

# Hazard Relation Diagrams: A Diagrammatic Representation to increase Validation Objectivity of Requirements-based Hazard Mitigations

Bastian Tenbergen<sup>†,‡</sup>, Thorsten Weyer<sup>‡</sup>, Klaus Pohl<sup>‡</sup>

<sup>†</sup> *Department of Computer Science, State University of New York at Oswego, United States*

<sup>‡</sup> *paluno – The Ruhr Institute for Software Technology, University of Duisburg-Essen, Germany*

**Abstract.** When developing safety-critical embedded systems, it is necessary to ensure that the system under development poses no harm to human users or external systems during operation. To achieve this, potential hazards are identified and potential mitigations for those hazards are documented in requirements. During requirements validation, the stakeholders assess if the documented hazard-mitigating requirements can avoid the identified hazards. Requirements validation is highly subjective. Among others, validation depends on the stakeholders' understanding of the involved processes, their familiarity with the system under development, and on the information available. In consequence, there is the risk that stakeholder judge the adequacy of hazard-mitigating requirements based on their individual opinions about the hazards, rather than on the documented information about the system's hazards. To improve the validation of hazard-mitigating requirements, we recently proposed a diagrammatic representation called Hazard Relation Diagrams [61].

In this paper, we extend the ontology of Hazard Relation Diagrams, present their notations and define well-formedness rules. We elaborate on the application of Hazard Relation Diagrams to visualize complex relationships between hazards and mitigations and present an automated approach to generate Hazard Relation Diagrams. Finally, we report on our empirical evaluations about the impact of Hazard Relation Diagrams on review objectivity, effectiveness, efficiency, and reviewer's subjective confidence.

*Keywords:* Safety Requirements, Hazards, Hazard-Mitigating Requirements, Safety Assessment, Validation, Reviews, Inspections, Mitigation, Adequacy, Modeling, Safety-Critical Embedded Systems, Model-Based Engineering, Hazard Relation Diagrams.

## 1. Introduction

Besides defining system functionality that is suitable to stakeholders, the engineering process of safety-critical embedded systems has to ensure that the system to

be developed is sufficiently safe [1]. Safety in this sense means that no harm can come to human users or external systems during operation. Hazards are operational situations that – given disadvantageous trigger conditions from the operational context [5] – can cause or contribute to harm (see Section 2 for the fundamental terminology adopted in this work). During early phases of safety assessment (see, e.g., [2], [3]), hazard analyses (such as Functional Hazard Analysis, see [4]) are conducted to identify possible hazards. In principle, each functional requirement defined and realized in the system can be the cause for one or more hazards to occur during system operation. Hazards are thus typically identified based on the functional requirements defined for the system.

To ensure sufficient system safety, the identified hazards must be mitigated. This is done by deriving safety goals from the identified hazards. In safety engineering<sup>1</sup>, safety goals are often understood as high-level statements about the safety of the system under development in a safety argument (see [8], [9]). However, in requirements engineering [6], the term “*goal*” refers to a top-level desired property or objective of the system under development, which is refined into system functionalities. For safety-critical embedded systems, this means that the safety goals derived during hazard analyses must be refined into mitigations fulfilling the safety goals [10]. Mitigations comprise specific functionalities to avoid or reduce the occurrence of the hazard’s trigger conditions, or control harm caused by the hazard. According to [5], the following types of mitigation strategies exist:

- **Hazard Prevention.** A hazard is mitigated by preventing the hazard’s trigger conditions from occurring during operation.
- **Hazard Reduction.** A hazard is mitigated by reducing the likelihood of the hazard to occur, e.g., by reducing the likelihood of the trigger conditions to occur.
- **Accident Prevention.** If a hazard cannot be prevented or sufficiently reduced, a mitigation can prevent the occurrence of a harmful accident due to a hazard.
- **Damage Control.** If a hazard can neither be prevented, nor sufficiently reduced, nor can an accident be prevented, a mitigation can aim to protect human users from injury, protect external systems from damage, or reduce the severity of such harm. This could be achieved, for example, by means of additional functionality intended specifically for damage reduction.

---

<sup>1</sup> In the following, we use the term “*safety engineering*” for the academic discipline, while we use the term “*safety assessment*” for the activities carried out during development.

In the simplest case, the hazard-inducing requirement can be omitted in order to avoid a hazard. Yet, typically, a functional requirement documents functionality required to achieve the system’s operational purpose. Mitigations must hence comprise concrete implementable measures to mitigate the hazard (see [1]) while maintaining the functionality desired by the stakeholders. In other words, hazard-mitigating requirements (cf. [7]) must be defined for the chosen mitigation strategy.

Defining *valid* hazard-mitigating requirements is one of the central challenges in the development of safety-critical embedded systems [11]. Hazard-mitigating requirements must not only adequately define the stakeholder intentions with regard to the system’s functionality (i.e. describe the *right* system), but must also define adequate<sup>2</sup> mitigations to address all hazards (i.e. describe the *right* hazard mitigations).

Typically, the validity of hazard-mitigating requirements is assessed during the validation of the entire requirements specification using, e.g., manual reviews [12]. However, manual requirements reviews are impaired by several challenges. For example, the quality and objectivity of the review mostly depends on the involved stakeholders and their understanding of the relevant processes [13], [14], their understanding of the system under development [15], [16], [17], and simply on the information available to them such that they may be able to make an educated judgment [12]. Specifically, the latter is crucial with regard to safety-critical systems, as the adequacy of hazard mitigations must be judged with regard to contextual information about the hazard, i.e. the safety goals to be achieved [8], [9] or the triggering conditions in the operational context of the system [1] under which the hazard occurs. In consequence, inadequate mitigations can be overlooked and thus hazards can still occur during operation, or additional hazards can occur.

### 1.1. Motivating Example

Consider the following example of a simplified adaptive cruise control system (ACC) from the automotive domain. An ACC is a driver assistance system, which maintains both the desired speed set by the driver and a minimal safe distance to a vehicle driving ahead. Fig. 1 depicts an activity diagram, which shows an excerpt of the

---

<sup>2</sup> Please note the difference between adequacy, i.e. an artifact’s suitability for the development process, and an artifact’s formal correctness, as outlined in [15]. Even formally correct requirements can be functionally, strategically, or legally inadequate to mitigate a hazard.

functional requirements of an ACC. The ACC in this example continuously monitors the *Driver's Desired Speed* and the *Own Vehicle's Current Speed* and computes the *Adjusted Speed* through the functional requirement “Compute Optimal Velocity.” Doing so allows the ACC to gradually reach the desired speed through smooth acceleration and deceleration, rather than using maximum engine power or brake force. Concurrently, the functional requirement “Measure Distance” continuously monitors some *Vehicle Ahead* that has been detected, e.g., by means of RADAR or LIDAR sensors, and computes the *Distance to Vehicle Ahead*. Based on the adjusted speed and the distance to the vehicle ahead, the functional requirement “Compute Closing Rate” determines whether the own vehicle is closing in on the vehicle driving ahead (*Closing Rate*). If the own vehicle underruns the minimal safe distance, the ACC computes the *Required Brake Force* to maintain the minimal distance. Else, the *Required Engine Torque* is computed to achieve the *Adjusted Speed*. More details on ACCs can be found in [18], [19].

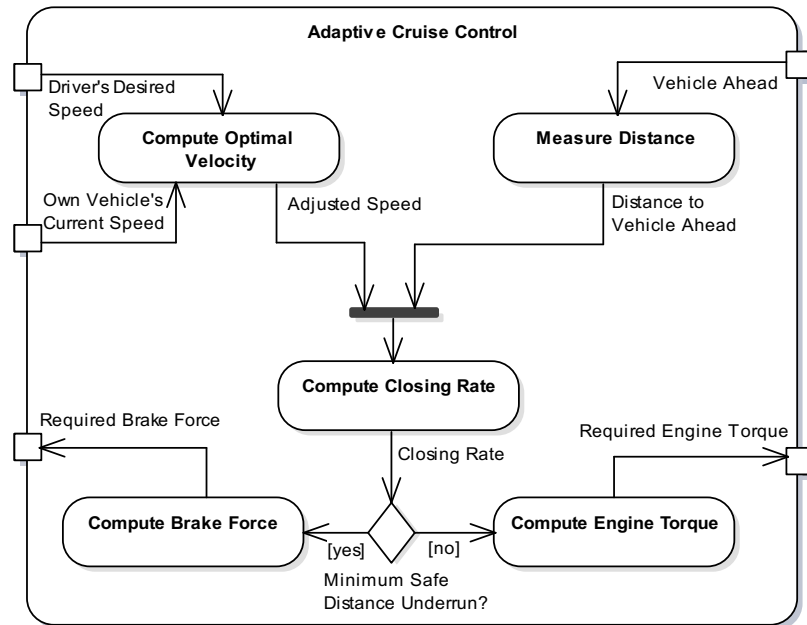


Fig. 1. Excerpt of the Functional Requirements of the Example ACC.

These functional requirements are subjected to systematic hazard analyses (e.g. Functional Hazard Analyses, FHA, see [4]) in order to determine potential hazards, their harmful effects, possible trigger conditions under which the hazards are induced, and possible safety goals in response to the hazard (see Section 2 for definitions of these terms). Table 1 shows an excerpt of a Functional Hazard Analysis for the functional requirement “Compute Brake Force” from Fig. 1.

Table 1. Functional Hazard Analysis of the Functional Requirement “Compute Brake Force” from Fig. 1.

System	Adaptive Cruise Control		Functional Hazard Analysis			
Func. Req.	ID	Hazard Description	Effect	Trigger Condition	ID	Safety Goal Description
Compute Brake Force	H1	Sudden Deceleration during Yaw	Loss of Control, causing Crash	Driving through Curve and (High Vehicle Velocity or Low Road Surface Friction)	SG1	Prevent Loss of Control in Curves
	H2	Understeering due to Skidding		Driving through Curve or Low Road Surface Friction	SG2	Use Anti-Lock Brakes to Prevent Wheel Skid
	H3	Brake Force Too Low due to Skidding	Rear-End Collision with Vehicle Driving Ahead	Low Road Surface Friction	SG3	Adjust Brake Force Independently for All Four Wheels

As depicted in Table 1, a hazard that could occur during operation is that a rapid deceleration command is issued by the ACC whilst the car is driving through a curve (hazard H1 in Table 1). This may cause the driver to lose control over the vehicle and crash. A simple mitigation would be to fulfill the safety goal “Prevent Loss of Control in Curves” (SG1 in Table 1). This safety goal could be satisfied by querying a yaw rate sensor to determine whether the vehicle is currently driving through a curve, and potentially limit the maximum brake force. At first glance, this may be an adequate way to mitigate the hazard. However, doing so might cause the vehicle to brake insufficiently, thereby underrunning the minimal safe distance, or possibly even rear-ending the vehicle driving ahead. More experienced automotive engineers might instead choose to compute the necessary brake force for each wheel and submit the result to the electronic stability program, thereby maximizing brake efficiency and maintaining a controlled yaw, while at the same time, keeping a safe distance to the vehicle driving ahead.

Both mitigation alternatives satisfy the safety goal, but only the latter adequately mitigates the hazard while at the same time maintains a safe trajectory through the curve. During validation, the inadequacy of the former mitigation alternative might be overlooked, since hazard analysis results, which are a work product of safety assessment and provide crucial information about the adequacy of the hazard-mitigating requirements, might not be available in the requirements phase (see, e.g., [11], [12], [17], [20]). In consequence, there is a severe risk that inadequate mitigations are overlooked and the hazard can still cause harm during operation.

## 1.2. Contribution and Outline

The adequacy of hazard mitigations must thus be judged against objective contextual information about the hazard, i.e. the safety goals to be achieved [8], [9] and the triggering conditions in the operational context of the system [1] under which the

hazard occurs. In [21], we have proposed a diagrammatic representation called Hazard Relation Diagrams, which displays hazard-mitigating requirements together with contextual information about the hazard to be mitigated. Hazard Relation Diagrams thus support judging the adequacy of the hazard-mitigating requirements objectively.

The contribution of this paper is threefold:

1. we extend the ontology and the notation of Hazard Relation Diagrams from [21] and define well-formedness rules, which support the creation of syntactically correct Hazard Relation Diagrams;
2. we outline the use of Hazard Relation Diagrams in non-trivial relationships between hazards and mitigations;
3. we present an automated approach to generate Hazard Relation Diagrams; and
4. we empirically investigate the impact of using Hazard Relation Diagrams during reviews on objectivity, effectiveness, efficiency, and subjective certainty.

This paper is structured as follows: Section 2 discusses the fundamental terminology and outlines the relationship between safety assessment activities and the requirements engineering process. Section 3 summarizes the principles of Hazard Relation Diagrams as first introduced in [21] and extends their ontology as well as the visual notation. In addition, Section 3 introduces well-formedness rules and discusses the applicability of Hazard Relation Diagrams in different relationship types between hazards and mitigations. Section 4 presents our approach to generate Hazard Relation Diagrams. Section 5 outlines the design of our empirical experiments. Section 6 discusses participant demographics. Section 7 presents the results of the empirical investigation. Section 8 critically discusses the threats to validity. Section 9 reviews the relevant state of the art. Section 10 concludes this paper.

## 2. Fundamental Terminology

In [22] it is reported, that certain terms are interpreted rather differently between practitioners and scholars in safety engineering and requirements engineering. To lay a common foundation, this section extends the terminology we have provided in [21] to define the basic concepts relevant for this paper.

The term “hazard” is of central importance to our work, but is used differently across standards and authors in the field of safety engineering. As outlined in Section 1, hazards are identified based on functional requirements during early phases of safety

assessment. For this paper, we have adopted the following definition based on [1] and [23]:

**Definition 1: Hazard.** *A hazard is an operational situation that – given disadvantageous triggering conditions in the operational context of the system – could lead or contribute to harm to come to humans or systems.*

Hazard analyses are not only concerned with identifying hazards, but also with identifying trigger conditions and safety goals:

**Definition 2: Trigger Condition.** *A trigger condition is an operational or environmental condition, which may occur during operation such that a hazard is caused and must hence be avoided or rendered sufficiently unlikely to occur during operation for the hazard to be mitigated (cf. [1]).*

**Definition 3: Safety Goal.** *A safety goal is a statement about the system’s safety or specific safety property that the system possesses or shall possess [10].*

Safety goals are inherently abstract and conceived in response to the hazard. In trivial cases, safety goals simply state the negation of an identified hazard (cf. [3], [24], [25], [26]). Like in goal-oriented requirements engineering, safety goals have to be refined into concrete, implementable functionality to mitigate a hazard.

The term “functional safety requirement” is used in two distinct, but related ways: In safety engineering literature, a requirement is often considered safety-critical when it gives rise to a hazard (see, e.g., [25]). In contrast, requirements engineering literature often considers safety requirements a type of quality requirement (e.g., [17], [27]), which is in place to achieve a certain level of safety. However, safety can only be achieved when concrete functional safety requirements (in the sense of [7]) are defined, i.e. concrete conditions and capabilities that, when implemented entirely and without error, mitigate the hazard. To address this dual role of requirements with regard to safety and to emphasize the functional nature of requirements documenting hazard mitigations, we adopt the following definitions:

**Definition 4: Hazard-Inducing Requirement.** *A hazard-inducing requirement is a functional safety requirement in the sense of [7], which is the origin of a hazard during operation, given the occurrence of trigger conditions from the operational context of the system.*

**Definition 5: Hazard-Mitigating Requirement.** *A hazard-mitigating requirement is a functional safety requirement in the sense of [7], which, possibly together with other hazard-mitigating requirements, mitigates a hazard.*

Hazard-mitigating requirements may themselves cause hazards. Therefore, safety standards (e.g., [2], [3]) demand iterative hazard identification and hazard mitigation, i.e. hazard-mitigating requirements themselves must be subjected to hazard analyses. Furthermore, it might be necessary to conceive hazard-mitigating requirements for hazards that arise due to a mitigation of another hazard.

Definition 5 suggests that minimally, the mitigation of a hazard can be defined in one hazard-mitigating requirement. In practice, this is rarely the case. Typically, multiple hazard-mitigating requirements must be defined in order to adequately mitigate a hazard. Obviously, incompleteness of hazard-mitigating requirements may impair adequacy: if one or more hazard-mitigating requirements are missing, the mitigation is likely to be inadequate. Assessing the completeness of hazard-mitigating requirements should thus be part of the validation process.

In the safety engineering literature, the term “*mitigation*” is used rather abstractly. For the technical aspects of Hazard Relation Diagrams (see Sections 3.3 and 4), a differentiation of this term is necessary. We therefore define the following two terms:

**Definition 6: Partial Mitigation.** *A partial mitigation consists of a set of hazard-mitigating requirements that are intended to mitigate a specific hazard.*

**Definition 7: Conceptual Mitigation.** *A conceptual mitigation consists of at least one partial mitigation, which refines a hazard’s safety goal into concrete implementable measures to avoid the hazard or reduce its harmful effects.*

The term “conceptual mitigation” refers to the chosen strategy of how to mitigate a hazard, while the term “partial mitigation” refers to the concrete hazard-mitigating requirements defined to implement the strategy. A conceptual mitigation therefore serves as a “bridge” between the abstract mitigation strategy (see Section 1 as well as [5]) and its technical realization (see Section 4.3). If two or more conceptual mitigations for the same hazard exist, these can be thought of as alternative strategies to mitigate the same hazard. In Section 3.3, the distinction between conceptual mitigations and partial mitigations is discussed in more detail.

Fig. 2 depicts the relationship between these terms and concepts using the running example from Section 1.1. The functional requirements of the ACC from Fig. 1 were subjected to hazard analyses. One hazard that could be *triggered* when driving through a curve at high velocity is that a sudden deceleration during yaw may cause the driver to lose control, potentially resulting in a crash (see hazard H1 in Table 1). In this case, “Compute Brake Force” from Fig. 1 is a *hazard-inducing requirement* for the hazard



“Sudden Deceleration during Yaw”. A possible *conceptual mitigation* for this would be to add requirements to achieve the *safety goal* “Prevent Loss of Control in Curves” (SG1 in Table 1). Following the strategy indicated in the conceptual mitigation, one *partial mitigation* was added containing *hazard-mitigating requirements* to determine the current yaw rate (“The ACC shall query the yaw sensor to determine the current yaw rate”) and limiting the brake force (“If the yaw rate exceeds 5°/sec, the maximum brake force shall be reduced by 15% per 5°/s of yaw”).

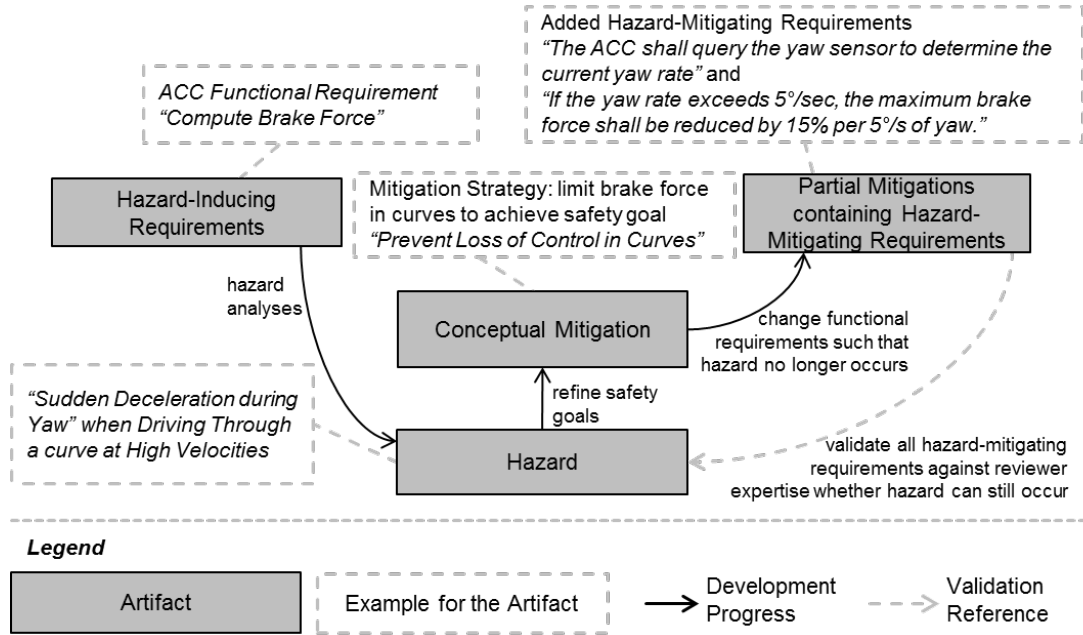


Fig. 2. The Relationship between Requirements, Hazards, Conceptual Mitigation, Partial Mitigation, and Validation illustrated through the ACC Example.

### 3. Hazard Relation Diagrams

As discussed in Section 1 validation is often subjective. Validation results depend, among other things, on the involved stakeholder’s understanding of the system under development, the process, and the information available to them [12], [16], [17]. In [21], we have introduced a diagrammatic representation intended to increase validation objectivity by integrating hazard analysis results and hazard-mitigating requirements. The representation is called “Hazard Relation Diagram.” In Section 3.1, we summarize the principle modeling concepts, ontological basis, and visual notation of Hazard Relation Diagrams from [21]. In Section 3.2, we build on these principles and present well-formedness rules of Hazard Relation Diagrams that arise from the ontological foundations and enable the automatic generation of Hazard Relation Diagrams. In Section 3.3, we illustrate the use of Hazard Relation Diagrams to visualize complex relationships between conceptual mitigations and hazards.

### 3.1. Principles, Ontology, and Visual Notation

The key idea of Hazard Relation Diagrams is to unify contextual information about the hazard (i.e. information from the hazard analysis results) with the hazard-mitigating requirements. Hazard Relation Diagrams structure the information necessary to conduct validation by visualizing trace concepts between hazard analysis results and hazard-mitigating requirements (see Moody’s Principle of Complexity Management and Moody’s Principle of Cognitive Integration in [35]). This reduces the cognitive load and subjectivity during validation. Hence, Hazard Relation Diagrams display the specific information needed to validate the hazard-mitigating requirements comprising the same conceptual mitigation, intended to mitigate one specific hazard. The language is based on an ontology, which extends UML activity diagram, as shown in Fig. 3. For a more detailed explanation of the modeling concepts and their relationships, please see [21].

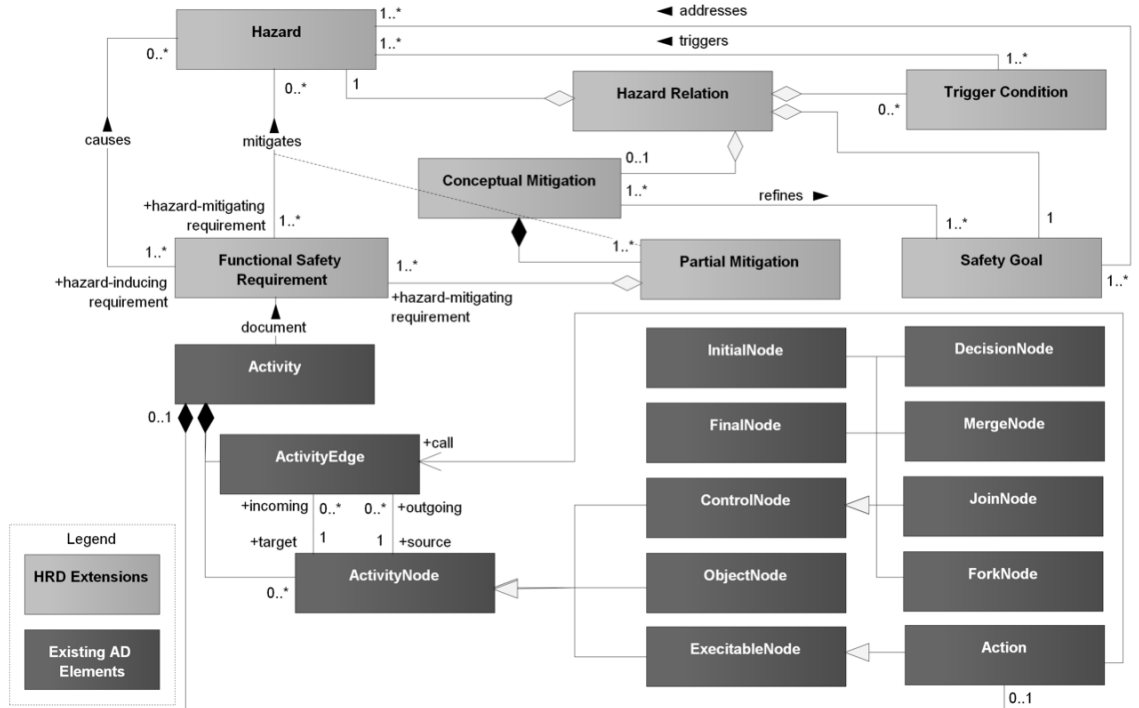







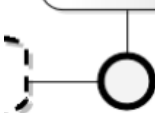


Fig. 3. Ontology for Hazard Relation Diagrams proposed in [21].

In Fig. 3, the modeling concepts specific to Hazard Relation Diagrams (see Section 2) are depicted in light grey boxes. Hazard Relation Diagrams extend UML activity diagrams with a minimal set of additional modeling elements (see Moody’s Principle of Graphic Economy in [35]). Dark grey boxes depict the elements of the UML meta-model for conventional activity diagram from [36]. In accordance with Definition 4 and Definition 5, the modeling elements represent both hazard-inducing and hazard-mitigating requirements (subsumed under the ontological element *Functional Safety Requirement*, see Moody’s Principle of Cognitive Fit in [35]). The core of a Hazard

Relation Diagram is the *Hazard Relation* which associates one *hazard* to its set of *trigger conditions*, *safety goals*, and the *conceptual mitigation*. Note, functional requirements for a system are in practice not specified in a single but in multiple activity diagrams. Conceptual mitigations could comprise hazard-mitigating requirements scattered across multiple diagrams, as is illustrated in Section 3.3. Therefore, our ontology depicted in Fig. 3 improves the ontology from [21] by introducing a differentiation between the ontological element *conceptual mitigation* and *partial mitigation* (see also Definition 6 and Definition 7 in Section 2). Hazard-mitigating requirements are part of the partial mitigation, which in turn is part of the conceptual mitigation of a hazard. This differentiation is necessary to accommodate generation of Hazard Relation Diagrams (see Section 4) and non-trivial relationships between the “mitigation” and hazard-mitigation requirements (see Section 3.3). Table 2 shows the notational elements of the modeling concepts of Hazard Relation Diagrams.

Table 2. Modeling Concepts of Hazard Relation Diagrams and their Visual Notation.

Modeling Concept	Diagrammatic Element	Description of Visual Notation	Example Visual Notation
Hazard	Hazard	Rectangle featuring a flash symbol in the middle and bearing the name of the hazard.	
Trigger Condition	Atomic Trigger Condition	Rounded edge rectangle with a dashed border bearing the name of the condition.	
	Trigger Condition Conjunction	Circle featuring two ampersand characters.	
	Trigger Condition Disjunction	Circle featuring two vertical lines (i.e. “pipe” operators).	
Safety Goal	Safety Goal	Rectangle bearing the stereotype <<Safety Goal>> as well as the name of the safety goal.	
Hazard-Mitigating Requirements	UML activity diagram Elements	Standard UML activity diagram modeling elements.	see [37]
Partial Mitigation	Mitigation Partition	Transparent rounded edge rectangle bearing the word “Mitigation.”	
Hazard Relation	Hazard Relation	Empty circle with a bold border.	
Hazard Association	Hazard Association	Line connecting a Hazard Relation with either a Hazard, with a Safety Goal, a Trigger Condition tree, and at least one Mitigation Partition. Corners and bends are permissible.	

For each ontological element one notational element has been defined in accordance with Moody’s principles of Semiotic Clarity and Semantic Transparency (see [35]). For example, *hazards* are depicted using a UML class with a bold arrow in the middle. *Safety goals* are modeled using a UML class stereotyped <<Safety Goal>>, thereby allowing future extensions to derive dedicated symbols in combination with other model-based development approaches (e.g., KAOS [38], i\*/Tropos [39], [40], or GSN [41], see also Moody’s principles of Cognitive Integration and Dual Coding in [35]). The hazard’s *trigger conditions* are documented using a binary tree of rounded-edge rectangles, like UML states (see Definition 2). Conjunctions and disjunctions between atomic trigger conditions are represented using node-elements bearing ampersand and “pipe” symbols reminiscent of modern programming languages (e.g. Java). Following Moody’s Principle of Perceptual Discriminability (see [35]), the modeling concept *mitigation partition* visually represents the hazard-mitigating requirements contained in a partial mitigation. Hazard Relation Diagrams contain one mitigation partition for each partial mitigation. Mitigation partitions are depicted using bold, dashed, rounded-edge rectangles which encapsulate the hazard-mitigating requirements that are subsumed by the corresponding partial mitigation. The *Hazard Relation* is the central modeling concept, which associates a *hazard* to its *trigger conditions*, *safety goal*, and the *hazard-mitigating requirements* subsumed by the *mitigation partitions*. A Hazard Relation can hence be thought of as an n-ary association between these modeling concepts. In UML class diagrams, n-ary associations are represented as diamond shapes, which are visually identical to UML activity diagram decision and merge nodes. Since decision and merge nodes can occur (as part of the UML activity diagram modeling elements) in Hazard Relation Diagrams, this may lead to confusion (see Moody’s principles of Visual Expressiveness and Semiotic Clarity in [35]). Hence, Hazard Relations are depicted using a circle with a bold border and associate the hazard to its safety goal, its trigger conditions and to the mitigation partition by means of individual hazard associations.

Fig. 4 shows an example of a Hazard Relation Diagram. The Hazard Relation Diagram features the ACC example from Section 1.1 and the hazard-mitigating requirements suggested in Section 2. The hazard “Sudden Deceleration during Yaw” (H1 in Table 1) and the associated *safety goal* “Prevent Loss of Control in Curves” (SG1 in Table 1) are depicted. The hazard’s trigger conditions have been documented as a binary tree using a Trigger Condition Conjunction as well as a Trigger Condition Disjunction. In Section 2, we suggested to limit the brake force depending on the rate of yaw to

mitigate hazard H1. This conceptual mitigation has been refined into several *hazard-mitigating requirements* surrounded by the bold, dashed, rounded-edge *mitigation partition*. In Fig. 4, the mitigation partition hence comprises the added input pin, which queries the car’s yaw rate sensor, and the added decision node, which checks if the brake force was appropriately limited.

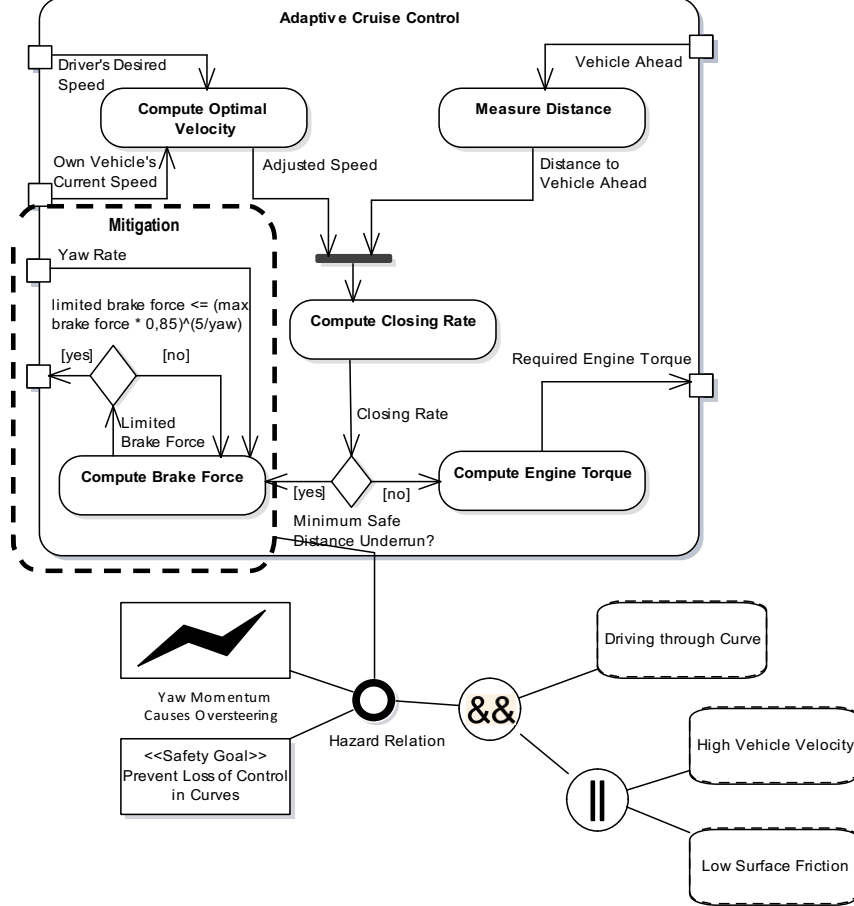


Fig. 4. Example of a Hazard Relation Diagram featuring Hazard H1 and Safety Goal SG1 from Table 1 and the Hazard-Mitigating Requirements from Section 2.

### 3.2. Well-Formedness Rules of Hazard Relation Diagrams

Well-formedness rules defined in this section further restrict the ontological foundations from Section 3.1 to support the definition proper Hazard Relation Diagrams. In Hazard Relation Diagrams, the hazard as well as its contextual information become reference artifacts against which the adequacy of the hazard-mitigating requirements for a specific hazard is validated. We define the following well-formedness rule:

**Well-Formedness Rule 1.** *A Hazard Relation Diagram contains exactly one hazard.*

During safety assessment, the safety of a system is argued by means of the adequate fulfillment of a safety goal (see Definition 3), which was defined in response to a hazard. Therefore, that safety goal must be contained in a Hazard Relation Diagram:

***Well-Formedness Rule 2.*** *A Hazard Relation Diagram contains exactly one safety goal.*

At least one hazard-mitigating requirement must be defined to fulfill the safety goal (see Section 2). Missing hazard-mitigating requirements are likely to cause the fulfillment and hence the conceptual mitigation to be inadequate. The adequacy of hazard-mitigating requirements depends on the adequate fulfillment of the safety goal that corresponds to the hazard. This can only be judged if a conceptual mitigation adequately fulfills the safety goal, if the safety goal and the hazard match:

***Well-Formedness Rule 3.*** *The safety goal in a Hazard Relation Diagram is specific to the hazard depicted in the Hazard Relation Diagram.*

In addition, whether or not a hazard still occurs during operation depends on the circumstance of in the operational context (see Definition 2). Therefore, validation must consider the hazard's specific trigger conditions:

***Well-Formedness Rule 4.*** *A Hazard Relation Diagram contains all trigger conditions identified during hazard analyses that are specific to the hazard depicted in the Hazard Relation Diagram.*

In practice, trigger conditions can rarely be expressed as atomic Boolean states. They often require a combination of operational situations. Therefore, multiple conditions must occur together for a hazard to be triggered. In addition, there may be alternative triggers for one hazard. Trigger conditions are thus typically defined as a tree of atomic states consisting of conjunctions and disjunctions of root nodes and atomic states in leave nodes:

***Well-Formedness Rule 5.*** *The trigger conditions are represented in a Hazard Relation Diagram in a tree structure.*

In principle, n-ary tree structures may be permissible. For the purpose of automatically generating Hazard Relation Diagrams (see Section 4), we restrict the tree structure to a binary tree, where the root of the binary trigger condition tree is either a Trigger Condition Conjunction or a Trigger Condition Disjunction (see Table 2). Binary trees have at most two leave nodes. We therefore define the following well-formedness rule:

**Well-Formedness Rule 6.** *The roots of the binary trigger condition tree must be either a Trigger Condition Conjunction or a Trigger Condition Disjunction and there must be at most two leaves.*

A trigger condition can in turn be recursively caused by some other trigger condition [5]. A binary trigger condition tree might thus contain further subtrees:

**Well-Formedness Rule 7.** *The leafs of the binary trigger condition tree must be atomic states from the operational context of the system under development identified during hazard analyses or the root for a subtree.*

In Section 3.1 we outlined that hazard-mitigating requirements in Hazard Relation Diagrams shall be documented using the same notational elements as conventional UML activity diagrams. We therefore define the following well-formedness rule:

**Well-Formedness Rule 8.** *The hazard-mitigating requirements depicted in a Hazard Relation Diagram are documented using the notational elements of UML activity diagrams.*

The hazard-mitigating requirements in a Hazard Relation Diagram must be syntactically correct, as syntactic correctness is a prerequisite for semantic validity (see Montague’s View of the role of syntax as described in [42]). However, syntactic validity of diagrams is beyond the scope of these well-formedness rules.

In Section 2, we have introduced the distinction between conceptual mitigations and partial mitigations: while conceptual mitigations refer to the strategy, how a hazard shall be mitigated, partial mitigations are used to subsume the concrete hazard-mitigating requirements. In Section 3.1, we have introduced the modeling element “mitigation partition” to visually subsume the hazard-mitigating requirements within a partial mitigation. Therefore, the following well-formedness rules follow:

**Well-Formedness Rule 9.** *A Hazard Relation Diagrams contains exactly one conceptual mitigation documented by at least one mitigation partition.*

**Well-Formedness Rule 10.** *A Hazard Relation Diagrams contains all mitigation partitions pertaining to the same conceptual mitigation.*

It follows, that there may be multiple mitigation partitions within one Hazard Relation Diagram. In this case, several sets of hazard-mitigating requirements are displayed pertaining to the same conceptual mitigation, for example, in different regions of the underlying activity diagram. Section 3.3 discusses this and other cases in more detail. Furthermore, hazard-mitigating requirements may overlap (e.g., if a hazard-mitigating requirement is used to mitigate two or more different hazards). In this case,

the hazard-mitigating requirement is contained in two or more partial mitigations. The partial mitigations are part of different conceptual mitigations, and therefore part of different Hazard Relation Diagrams. We thus define the following well-formedness rules:

***Well-Formedness Rule 11.*** *The conceptual mitigation depicted in a Hazard Relation Diagram is specific to the hazard depicted in the Hazard Relation Diagram.*

***Well-Formedness Rule 12.*** *The mitigation partitions depicted in a Hazard Relation Diagram subsume all hazard-mitigating requirements specific to the hazard depicted in the Hazard Relation Diagram.*

The central concept in Hazard Relation Diagrams is the Hazard Relation. It serves as a graphical representation of the trace links between the hazard (see Well-Formedness Rule 1), the safety goal (see Well-Formedness Rule 3), the trigger conditions (see Well-Formedness Rule 4), and the hazard-mitigating requirements (see Well-Formedness Rule 12). In each Hazard Relation Diagram, there can only be one Hazard Relation:

***Well-Formedness Rule 13.*** *A Hazard Relation Diagrams contains exactly one Hazard Relation.*

***Well-Formedness Rule 14.*** *The Hazard Relation contained in a Hazard Relation Diagram is associated with the hazard, the safety goal, the top-most element of the binary trigger condition tree, and all mitigation partitions depicted in the Hazard Relation Diagram.*

### 3.3. Relationship Types between Hazards and Conceptual Mitigations

Fig. 4 shows a Hazard Relation Diagram in which a single mitigation partition has been added to a activity diagram representing the functional requirements relevant for that hazard. Moreover, hazard-mitigating requirements which implement the conceptual mitigation were added in this activity diagram. However, in practice, the multiplicity between hazards and the conceptual mitigation is rarely 1:1 because diagrams depicting functional requirements are not “cut” according to potential hazards, but according to the level of detail needed for the specific development situation (see, e.g., [43], [44]). Therefore, there are several cases for interrelating hazards, conceptual mitigations, and partial mitigations (cf. Table 3).

Table 3. Types of Relationships between Hazards and Conceptual Mitigations in Hazard Relation Diagrams.

Case	# Hazards	# Conceptual Mitigations	# Mitigation Partitions	# Activity Diagrams	Description	Impact on HRDs
						Meaning
1	1	1	1	1	Exactly one hazard is addressed by exactly one conceptual mitigation	Default case shown in Fig. 4.



Case	# Hazards	# Conceptual Mitigations	# Mitigation Partitions	# Activity Diagrams	Description	Impact on HRDs
						Meaning
					and comprised in exactly one mitigation partition. The mitigation comprises hazard-mitigating requirements which were added to exactly one activity diagram.	
2	1	1	2..*	1	Exactly one hazard is addressed by exactly one conceptual mitigation, but the conceptual mitigation comprises hazard-mitigating requirements, which are scattered across geometrically distant areas within the same activity diagram.	Multiple partial mitigations are defined for the conceptual mitigation. For each partial mitigation, a mitigation partition is included in the Hazard Relation Diagram.
3	1	1	2..*	2..*	Exactly one hazard is addressed by exactly one conceptual mitigation, but the conceptual mitigation comprises hazard-mitigating requirements which are scattered across multiple activity diagrams.	Extension of Case 2: Multiple partial mitigations are defined for the conceptual mitigation. At least one partial mitigation is defined for each activity diagram. The partial mitigations contain hazard-mitigating requirements pertaining to the same conceptual mitigation. For each partial mitigation, the activity diagram and the corresponding mitigation partition is included in the Hazard Relation Diagram.
4	2..*	1	1..*	1..*	More than one hazard is addressed by the same conceptual mitigation, regardless whether the hazard-mitigating requirements are scattered across one or more activity diagrams.	The adequacy of the candidate mitigation must be validated with regard to each hazard, hence requiring one Hazard Relation Diagram for each hazard.
5	1	2..*	1..*	1..*	Multiple conceptual mitigations exist for the same hazard. Mitigations are independent from one another, i.e. represent alternative hazard-mitigating requirements, which might be scattered across different or the same activity diagram.	Reverse case of Case 4: The adequacy of each candidate conceptual mitigation must be reviewed with regard to the same hazard. Just like in Case 4, this requires one Hazard Relation Diagram for each conceptual mitigation.

Table 3 shows the trivial Case 1, which is the default case discussed in the previous sections. Fig. 4. depicts an example of Case 1.

Cases 2 and 3 in Table 3 represent more complicated versions of this interrelations. Several hazard-mitigating requirements that pertain to the same conceptual mitigation were added at different locations of one activity diagram (Case 2), to several activity diagrams (Case 3), or possibly a combination thereof. To allow for such non-trivial relationships between hazard-mitigating requirements, their conceptual mitigations, and hazards, the notational element “mitigation partition” (see Section 3.1) was introduced to represent each partial mitigation that is part of the same conceptual mitigation. This allows associating multiple mitigation partitions (which comprise hazard-mitigating requirements pertaining to the same conceptual mitigation, see Well-Formedness Rule 12) to the Hazard Relation of a Hazard Relation Diagram.

Fig. 5 shows an example for Case 2 and 3. In this example, the mitigation aims at detecting the yaw rate by querying individual wheels and delimiting the brake force

accordingly (the alternative conceptual mitigation suggested in Section 1.1). The conceptual mitigation consists of hazard-mitigating requirements depicted using three mitigation partitions, where one belongs to the ACC and two belong to the electronic stability program (ESP). The hazard-mitigating requirements for the ACC comprise functionality to query the current yaw rate as well as the current yaw momentum and comprise functionality to compute the optimal brake force for each individual wheel. In addition, hazard-mitigating requirements have been included in the functional requirements of the ESP (Fig. 5 shows an excerpt of a simplified ESP based on [45]). Specifically, a functional requirement to compute the current yaw momentum has been introduced. This functional requirement communicates the current yaw momentum to the functional requirement “*Compute Yaw Momentum Set Point*” and communicates the yaw momentum to the ACC. Furthermore, the ESP accepts input from the ACC regarding the necessary brake force for each wheel and the ESP double-checks the brake force distribution for each wheel before initiating the deceleration through the wheels’ brake actuators.

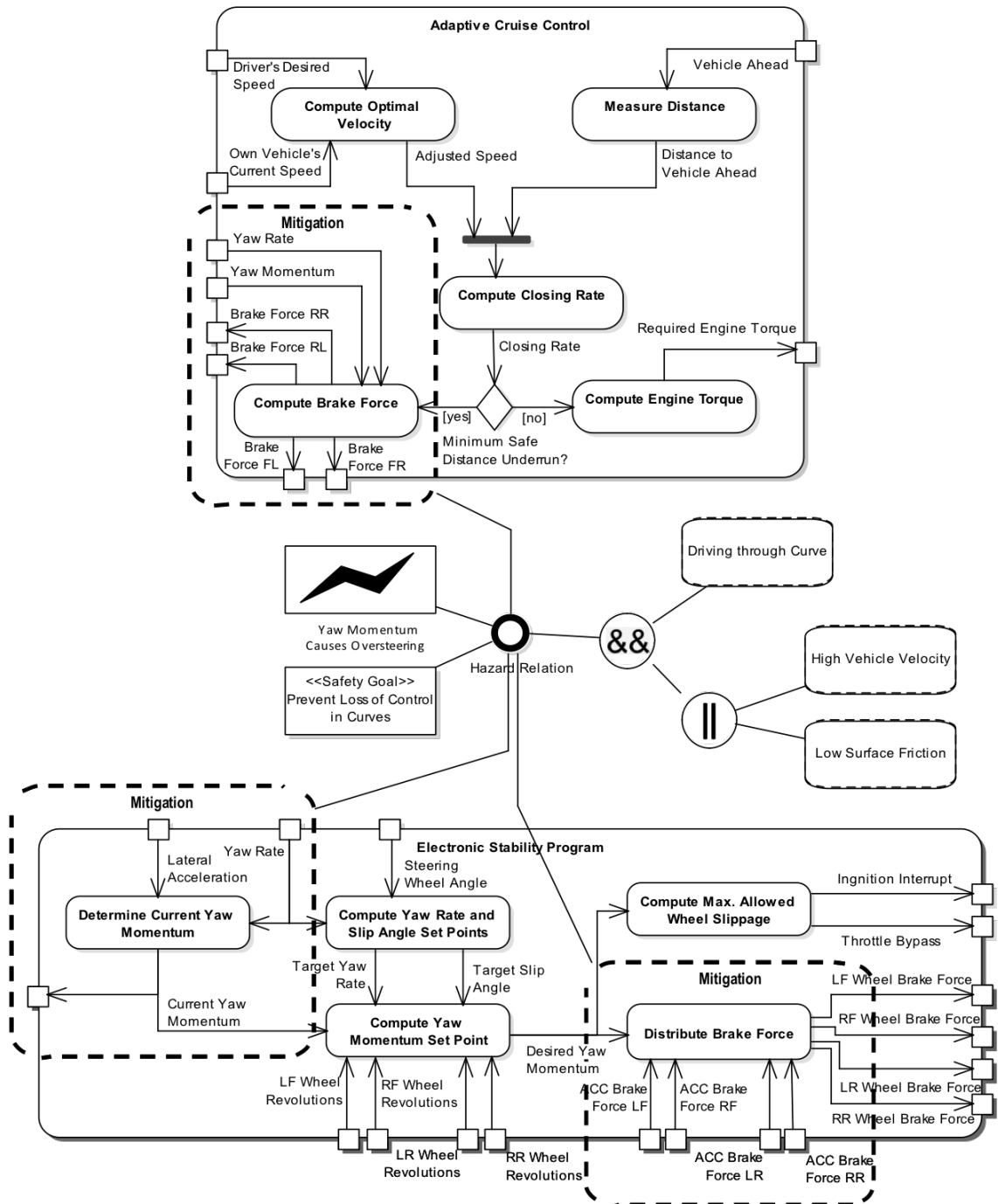


Fig. 5. Example of a Hazard Relation Diagram featuring Multiple Mitigation Partitions to fulfill Safety Goal SG2 from Table 1.

Table 3 furthermore depicts Case 4 and 5, in which several hazards are addressed by the same conceptual mitigation (Case 4), or several alternative conceptual mitigations exist for the same hazard (possibly with varying degrees of adequacy, Case 5). During development, combinations of these cases may occur frequently: For example, three hazards could be addressed by four conceptual mitigations, where one conceptual mitigation addresses two of the identified hazards and the other three conceptual mitigations address the last hazard. Since the purpose of Hazard Relation Diagrams is to focus the validators' attention on one conceptual mitigation with respect to a specific

hazard, each candidate conceptual mitigation must be validated with respect to each corresponding hazard (see Well-Formedness Rule 11). Thus, for Case 4 and 5 any combination of hazard and conceptual mitigation must be validated using an individual Hazard Relation Diagrams.

It must be noted that Cases 5 is limited to independent conceptual mitigation alternatives, i.e. each conceptual mitigation addresses the same hazard entirely, regardless of the respective other conceptual mitigations. The purpose of validation in this case is to assess not only adequacy of the conceptual mitigations, but also to assess *optimality* of conceptual mitigations, for instance to find a conceptual mitigation that is (1) adequate, (2) takes the least time to implement, or (3) is the most cost effective to implement. However, optimality is beyond the scope of this article.

The situation, where two or more conceptual mitigations are required to adequately address the same hazard does not constitute an incarnation of Case 5. What is assumed to be mutually dependent *conceptual* mitigations are in fact different *partial* mitigations belonging to the same conceptual mitigation (according to Definition 7). This constitutes an incarnation of Case 2 or Case 3. In consequence, combinations of Cases 4 and 5 with Cases 2 and 3 are likely to occur frequently during development.

In any other case which does not conform to Case 1 through 5 depicted in Table 3, no syntactically valid Hazard Relation Diagram can be generated, as in such cases, at least one of the well-formedness rules from Section 3.2 is violated. The case were multiple mutually dependent hazards require one or more mutually dependent conceptual mitigations (not depicted Table 3) can be expressed in terms of the above cases. This case might occur when one hazard “causes” another hazard. For such cases typically, Fault Tree Analysis is used to identify such interactions (i.e. to identify "minimal cut-sets" of hazards and their causes, see [4]). The identification of such interactions is not within the scope of this article.

## 4. Creating Hazard Relation Diagrams

This section presents an automatic approach to create syntactically correct Hazard Relation Diagrams. Section 4.1 provides an overview of the approach. Section 4.2 discusses the prerequisites for its applicability. Section 4.3 discusses the principles of generating Hazard Relation Diagrams and illustrate how the well-formedness rules from

Section 3.2 can be satisfied. The formal foundations to generate Hazard Relation Diagrams are not presented in this paper<sup>3</sup>.

#### 4.1. Overview

Fig. 6 depicts our approach for generating hazard diagrams as well as its interrelation with activities and artifacts of the development process of safety-critical embedded systems. Artifacts and activities of our approach are depicted using dark grey boxes with a black tag. Artifacts and activities belonging to the development process are depicted in white boxes and light grey tags. These development artifacts are prerequisites for the approach (see in Section 4.2).

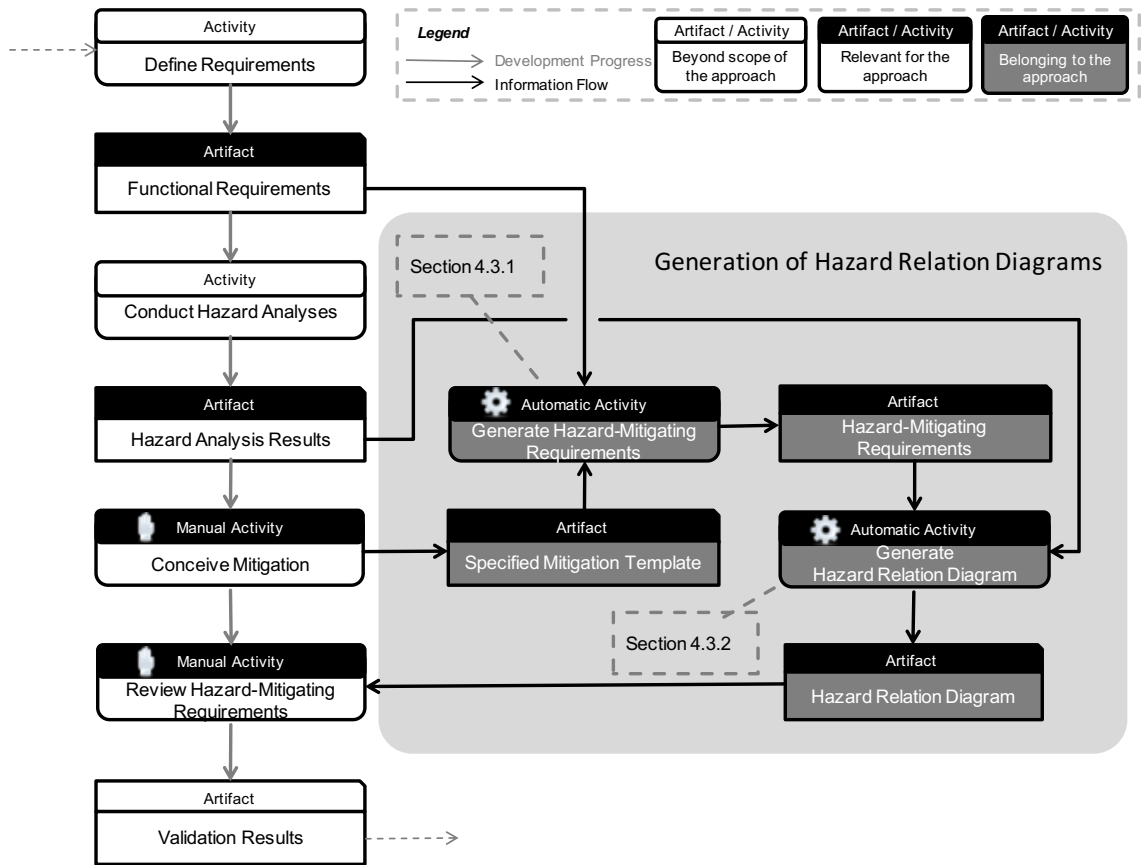


Fig. 6. Overview over the Approach to Generate Hazard Relation Diagrams

The left side of Fig. 6 shows an excerpt of the development process. During development, an initial set of functional requirements is defined, which is subjected to hazard analyses. Hazard analyses yield hazard-inducing requirements as well as hazard analysis results, consisting of the identified hazards, their trigger conditions, and necessary safety goals. Based on the hazard analyses results, possible conceptual

<sup>3</sup> The artifact formalizations are available as an appendix to this article at <https://goo.gl/FNfsB9>

mitigations for each hazard are conceived. Conceptual mitigations may make necessary specific changes, i.e. including adding, removing, or substituting functional requirements. The result is at least one conceptual mitigation containing at least one partial mitigation for each identified hazard.

Using all partial mitigations for the conceptual mitigation and the initial set of functional requirements, our approach first generates for each hazard one activity diagram, which includes the hazard-mitigating requirements (see Section 4.3.1). The activity diagram is then extended to a Hazard Relation Diagram for each hazard by automatically appending the contextual hazard information (i.e., the hazard's description, trigger conditions, and the conceived safety goal) from the hazard analysis results (see Section 4.3.2). Each resulting Hazard Relation Diagram (i.e., one for each hazard) is then validated to assess the adequacy of the hazard-mitigating requirements with regard to the hazard analysis results.

## 4.2. Prerequisites for Our Approach

The following prerequisites must be met by a development process:

- **The functional requirements of the system under development are elicited and documented using UML activity diagrams.** Functional requirements build the basis for safety assessment, as hazard analyses are concerned with identifying hazards based on the defined system functions (see [4]). Thus, a prerequisite for our approach is that all functional requirements have been elicited, i.e. are known. Functional requirements can be documented using natural language and/or different conceptual models and notations. We defined our Hazard Relation Diagrams as an extension of UML activity diagrams. Our approach for generating Hazard Relation Diagrams thus require that the functional requirements are documented using UML activity diagrams.
- **Results of hazard analyses are documented and are available.** Hazard Relation Diagrams combine the hazard-mitigating requirements defined in response to hazards and contextual information about them identified during hazard analyses. In different application domains, different types of hazard analyses are common. For example, in the avionics domain, Functional Hazard Analysis (FHA, see [2]) are mandatory for certification. The automotive industry often Hazard and Risk Assessment Analysis (H+R, see [3]) is used. As long as a hazard analysis produces the results described in

Section 3.1, our approach for generating Hazard Relation Diagrams does not depend on a specific type of hazard analysis.

- **For each hazard, partial mitigations are defined.** The safety goals identified during hazard analyses are refined into a principle strategy for mitigating the hazard. This constitutes the conceptual mitigation. The conceptual mitigation is further refined into concrete implementable measures to avoid the respective hazard or reduce its harmful effects. These measures constitute the hazard-mitigating requirements and have to be subsumed by at least one partial mitigation.
- **Partial mitigations and hazard analysis results are documented in template-based format.** For certifiably safe systems [11], it is required that hazard-mitigating requirements be traceable to the partial mitigation, which in turn must be traceable to the hazard analysis results. In order to support the generation of Hazard Relation Diagrams, hazard-mitigating requirements and their traces to hazards should be defined using templates. The template has to include all additions, removals, and substitutions of diagram elements from the activity diagrams as well as their relations to the conceptual mitigation of the hazard. The hazard, along with its trigger conditions and safety goal has to be documented in a similar manner (e.g., using hazard analysis worksheets, see [4], p. 276).

### 4.3. Artifact Generation

In the following subsections, we demonstrate how the satisfaction of the formalized well-formedness rules for Hazard Relation Diagrams from Section 3.3 can be ensured by our approach to automatically generate Hazard Relation Diagrams. To this end, our approach uses OMG's Query/View/Transformation Operational Mappings language (QVTo, see [47]). QVTo is a transformation language, which allows enacting UML model transformations by either manipulating specific model elements (i.e. adding, removing, or substituting model elements), or by converting diagrams on an ontological level. We use QVTo to first create an intermediate activity diagram containing hazard-mitigating requirements (see Section 4.3.1) and subsequently to append the diagram with the contextual hazard information (see Section 4.3.2). Thereby a Hazard Relation Diagram is created.

#### 4.3.1. Generating Hazard-Mitigating Requirements

As outlined in Section 3, hazard-mitigating requirements can be understood as specific changes to the functional requirements. Documenting these changes is done

using partial mitigation (see Section 2). Hazard-mitigating requirements are automatically enacted on the activity diagram containing the functional requirements  $ad^{fr}$  in order to generate an activity diagram containing the hazard-mitigating requirements  $ad^{hmr}$ :

$$(actD^{fr}, CM^h) \xrightarrow{q^{hmr}} actD^{hmr} \quad (4.1)$$

where  $CM^h = \{pm_1^h, pm_2^h, \dots, pm_n^h\}$  is a conceptual mitigation consisting of all partial mitigations referencing hazard  $h$  and  $actD^{fr}$  is an activity diagram containing functional requirements that is changed to  $actD^{hmr}$ , i.e. an activity diagram containing hazard-mitigating requirements and functional requirements. Generating hazard-mitigating requirements is done using a QVTo script  $q^{hmr}$  which contains code operations  $op$  to systematically perform all insertions, removals, and substitutions of elements in all partial mitigations  $pm_i^h \in CM^h$  that pertain to the same conceptual mitigation. It follows:

$$\begin{aligned} q^{hmr} &= (sig, op^{insert}, op^{remove}, op^{substitute}) | \\ op^{insert} &= I^{cm_1^h} \cup I^{pm_2^h} \cup \dots \cup I^{pm_n^h} \\ \wedge op^{remove} &= R^{pm_1^h} \cup R^{pm_2^h} \cup \dots \cup R^{pm_n^h} \\ \wedge op^{substitute} &= S^{pm_1^h} \cup S^{pm_2^h} \cup \dots \cup S^{pm_n^h} \\ \wedge pm_i^h &\in CM^h \wedge 0 < i < |CM| \end{aligned} \quad (4.2)$$

where  $sig$  is QVTo script signature, i.e. the head of every QVTo script which specifies input and output models and selects their meta-models (see [47] for more details on QVTo code). In (4.2),  $I$ ,  $R$ , and  $S$  represent modeling elements of the activity diagram  $actD^{fr}$  to be inserted, removed, or substituted, respectively<sup>4</sup>.

QVTo requires the meta-models of the input and output artifacts to be specified. This is included in the QVTo script signature  $sig$ , where conventionally, fictitious URI are used to point to the UML superstructure for use in QVTo scripts to be executed using the QVT Operational Mappings Plugin [48] available for the Eclipse Modeling Tool project [49]. Moreover, a fictitious URI points to the URI of an Ecore model representing the meta-model for the partial mitigations  $pm_i^h \in CM^h$  which must be part of the implementation. In addition, the source activity diagram containing functional

---

<sup>4</sup> The following algorithmic description references pseudo-code implementations, which we provide as an appendix to this article at <https://goo.gl/D6eGhR>



requirements  $actD^{fr}$  and the conceptual mitigation  $CM^h$  containing a set of partial mitigations  $pm^h$  are defined as inputs and the target activity diagram that contains the hazard-mitigating requirements  $actD^{hmr}$  is referenced. Before transformations occur,  $sig$  also checks if every partial mitigation references the same hazard. If not, an error is thrown and transformation is aborted by returning  $ad^{hmr} = \emptyset$ . If the referenced hazards are all the same, Well-Formedness Rule 10 is satisfied and transformation can continue.

Subsequently, the QVTo script  $q^{hmr}$  performs insertion, removal, and substitution operations  $op^{insert}$ ,  $op^{remove}$ , and  $op^{substitute}$ , respectively. Insertion operations to add additional modeling elements to the activity diagram are conducted for every partial mitigation in  $CM^h$ . For every insertion operation, it is checked what type the element to be inserted is (e.g., opaque action, association, fork node, etc., see [47]) and the appropriate set of diagram elements is created by unifying the old set of modeling elements already present in the activity diagram. Removing elements from the activity diagram is similar to their insertion: depending on what type the element to be removed is, a new set of such elements is computed that does not include the element in question. Removing modeling elements in this manner bears the risk that the removing operation results in some modeling elements (or cliques thereof) to be unconnected. Therefore, it is necessary to enforce the syntactic correctness of the remove operation. This, however, is beyond the scope of this article. To conduct substitution operations  $op^{substitute}$ , it is assumed that only elements of the same type can be substituted. In principle, non-homogenous substitutions may be permitted. In this case, a substitution operation can be understood as an insertion operation followed by a removing operation as outlined in  $op^{insert}$  and  $op^{remove}$ .

Once  $op^{insert}$ ,  $op^{remove}$ , and  $op^{substitute}$  have been executed, the hazard-inducing requirements are documented in UML activity diagrams (in accordance with Well-Formedness Rule 8). In principle, the developer can add (albeit most likely by accident) activity edges not connected to some other modeling element, or which cause other issues of syntactic invalidity. Therefore, basic syntactic validity checks must be conducted on the resulting activity diagram. Such checks can be incorporated into a QVTo code file. Executing  $q^{hmr}$  hence results in applying the changes specified in the partial mitigation to the functional requirements, yielding the hazard-mitigating requirements shown in Fig. 4. When implementing the QVTo scripts, certain dependency checks of insertion, removal, and substitution operations might be desirable. For example, such checks are

concerned with ensuring that no element is removed twice, inserted and immediately removed, etc.

#### 4.3.2. Generating Hazard Relation Diagrams

In Section 4.3.1, we have presented our algorithm to generate hazard-mitigating requirements from conceptual mitigations specified in partial mitigations. Since Hazard Relation Diagrams are extensions of activity diagrams containing hazard-mitigating requirements  $actD^{hmr}$ , a Hazard Relation Diagram  $hazRD$  can be automatically generated using the QVTo language as well:

$$\begin{aligned}
hazRD &= (hrd, AD^{hrd}, h^{hrd}, tc^h, sg^h, CM^{hrd}, hr^{hrd}, HA^{hrd}) | \\
h^{hrd} &\in fha(actD^{fr}) \wedge (actD^{fr}, pm_i) \xrightarrow{q^{hmr}} actD_i^{hmr} \\
&\wedge \forall pm_i \in CM^{hrd} \wedge \forall actD_i^{hmr} \in AD^{hrd}: \\
&\left( actD_i^{hmr}, CM^h, fha(actD^{fr}) \right) \xrightarrow{q^{hrd}} hazRD
\end{aligned} \tag{4.3}$$

In (4.3),  $hazRD$  is a Hazard Relation Diagram. Hazard Relation Diagrams are defined as an eight-tuple consisting of:

1. a name  $hrd$ ,
2. a set of activity diagrams containing functional requirements  $AD^{hrd}$ ,
3. a hazard  $h^{hrd}$ ,
4. a binary tree of trigger conditions  $tc^h$ ,
5. a safety goal  $sg^h$ ,
6. a set  $CM^{hrd}$  of all partial mitigations  $pm$ ,
7. a Hazard Relation  $hr^{hrd}$ , and
8. a set of hazard associations  $HA^{hrd}$ .

Furthermore, in (4.3),  $fha(actD^{fr})$  is a hazard analysis conducted on the functional requirements  $actD^{fr}$  yielding a list of Hazards that contains the hazard  $h^{hrd}$  referenced by the partial mitigations  $pm$  in  $CM^{hrd}$ . The purpose of the QVTo script  $q^{hmr}$  is to append contextual information about hazard  $h^{hrd}$ , thereby creating a Hazard Relation Diagram. Similar to the QVTo script from Section 4.3.1, the QVTo script  $q^{hmr}$  can hence be defined as:

$$q^{hmr} = \left( sig, append^{AD}, append^h, append^{sg}, append^{tc}, \right. \\
\left. append^{hr}, append^{CM}, append^{HA} \right) \tag{4.4}$$

where *sig* is QVTo script signature head and  $append^X$  are operations to append the activity diagrams containing hazard-mitigating requirements *AD*, the hazard *h*, the safety goal *sg* and the trigger conditions *tc* specific to *h*, a set of mitigation partitions for the partial mitigations in *CM*, the Hazard Relation *hr* and a set of hazard associations *HA*.

The QVTo script  $q^{hrd}$  takes three input parameters: an activity diagram containing hazard-mitigating requirements, a set of specified partial mitigations specific for one hazard, and the results of a hazard analysis containing, among others, contextual information about the mitigated hazard. The output of the script is a Hazard Relation Diagram. The same check for hazard specificity must be conducted in  $q^{hrd}$ , like in  $q^{hmr}$ , in order to satisfy Well-Formedness Rule 11. After this check is complete, it must be checked whether the hazard is part of the hazard analysis results  $fha(actD^{fr})$ . If the hazard is not part of  $fha(actD^{fr})$ , contextual information about the hazard cannot be added to the Hazard Relation Diagram and thus generation fails. Similarly, it must be checked if the activity diagrams containing the hazard-mitigating requirements have at least one corresponding partial mitigation. This check is required to ensure a mitigation partition, which subsumes the modeling elements representing the hazard-mitigating requirements, can be added to the Hazard Relation Diagram (see Section 3.3). If at least one activity diagram has no referencing partial mitigation, the generation of the Hazard Relation Diagram results in an error. These checks are subsumed in the signature *sig* of the QVTo script  $q^{hrd}$ .

Once the prerequisites to create a Hazard Relation Diagram are met, the notational elements outlined in Section 3.1 can be appended to the hazard-mitigating requirements by performing the operations  $append^{AD}$ ,  $append^h$ ,  $append^{sg}$ ,  $append^{tc}$ ,  $append^{hr}$ , and  $append^{CM}$  from (4.4). To this end, operation  $append^{AD}$  merges the hazard-mitigating requirements contained in the activity diagrams in  $AD^{hrd}$  from the input parameters in *sig*. Hazard-mitigating requirements are depicted using the modeling elements of UML activity diagrams using  $q^{hmr}$  (see Section 4.3.1). It follows that Well-Formedness Rule 8 is satisfied. Subsequently, operation  $append^h$  appends the hazard referenced in the partial mitigations to the Hazard Relation Diagram. Since *sig* has ensured that all hazards referenced in the partial mitigations are the same. Therefore, appending the hazard can be done by retrieving the hazard reference from any partial mitigation in  $CM^h$ . This fulfills Well-Formedness Rule 1. Next, operations  $append^{sg}$  and  $append^{tc}$  append the safety goals and trigger conditions to the Hazard Relation

Diagram. The prerequisite check in *sig* ensures that the safety goal and trigger conditions are specific for the hazard that was appended to the Hazard Relation Diagram using the *append<sup>h</sup>* operation. Thereby Well-Formedness Rule 2 to Well-Formedness Rule 4 are fulfilled. Appending safety goal and trigger conditions can be done by looking up the hazard in the hazard analysis results passed as an argument to  $q^{hrd}$ , retrieving the safety goal and the trigger conditions, respectively, and storing them in a local variable. Moreover, since the trigger conditions in  $fha(actD^{fr})$  are documented using a binary tree structure, Well-Formedness Rule 5 to Well-Formedness Rule 7 are satisfied. After all contextual information about the hazard are added to the Hazard Relation Diagram, *append<sup>CM</sup>* creates a mitigation partition for each partial mitigation in  $CM^h$  such that all inserted or substituted elements are subsumed by the mitigation partition. This way, Well-Formedness Rule 9 and Well-Formedness Rule 12 are satisfied. Lastly, exactly one Hazard Relation is created in *append<sup>hr</sup>* thereby fulfilling Well-Formedness Rule 13.

When  $q^{hrd}$  completes execution, all Hazard Relation Diagram components of have been created apart from the set of hazard association  $HA^{hrd}$ . This is done using operations in *append<sup>HA</sup>*. According to Well-Formedness Rule 14, there must be at least four hazard associations: between the Hazard Relation and the hazard, the safety goal, the trigger conditions, as well as at least one mitigation partition, respectively. Hazard associations can be seen as tuples of a source modeling element and a target modeling element. In this sense, the hazard associations between the hazard, the safety goal, and the trigger conditions all share the same source element, i.e. the Hazard Relation. The hazard association between the trigger conditions and the Hazard Relation must refer to the root of the binary trigger condition tree. Since Hazard Relation Diagrams can contain any number of mitigation partitions (however, at least one, see Well-Formedness Rule 9), all mitigation partitions must be associated with the Hazard Relation as well. This satisfies Well-Formedness Rule 14. All components are then appended into a tuple denoting the complete Hazard Relation Diagram.

## 5. Experimental Evaluation of Hazard Relation Diagrams

We have argued throughout this paper Hazard Relation Diagrams increase objectivity during validation. To validate this claim, we conducted two experiments with the focus on the following research questions:

- RQ1: What is the impact of Hazard Relation Diagrams on review objectivity?

- RQ2: Does the impact of Hazard Relation Diagrams on review objectivity come at a cost of decreased review effectiveness and efficiency?
- RQ3: Do Hazard Relation Diagrams impact the reviewers' confidence in their adequacy judgment?

More specifically, we investigated to evaluate the use of Hazard Relation Diagrams during individual reviews - the most common validation technique used [12]. To foster reproducibility and comparability to other studies, we present the empirical design of the two experiments in accordance to the guidelines from [50].

### 5.1. Evaluation Strategy: Between-Subjects vs. Repeated Measures

Initially, we designed our experimental investigation of Hazard Relation Diagrams using a between-group single factorial design [51] (in the following: Experiment 1). We favored this design over a repeated measures design since most participants required pre-hoc introduction into the fundamentals of safety engineering, hazard analyses, and Hazard Relation Diagrams. Repeated measures design would have greatly increased the training overhead per participant. This would have diminished our ability to draw conclusions based on the independent variable, as effects may have been caused by the impact of the training. Since we were able to recruit from a very large participant population (see Table 4 and Section 6.1), we deemed between-group single factorial design appropriate and validated our experimental design by means of a pilot experiment. The design and the results of the pilot experiment are discussed in [21].

After the Experiment 1, we had the opportunity to repeat the experiment (in the following: Experiment 2). In contrast to Experiment 1, the number of students was smaller. In addition, they had more advanced software engineering background (see Table 4 and Section 6.1). In order to maximize both comparability and empirical rigor, we designed Experiment 2 as a within-subjects repeated measures experiment [51], allowing for higher robustness against a small number of participants [52] and reducing overhead through additional training needed on safety engineering.

In an effort to maintain comparability of results in the experiments, we used the same experimental stimuli in both experiments and kept the experimental procedure as similar as possible. Hypotheses, measurements, data preparation, and the data analysis procedure was kept constant over both experiments, albeit some minor experiment-specific deviations were necessary. Table 4 summarizes the experimental configuration.

More details on the similarities and differences of the experiments are described in Sections 5.3 and Section 5.4.

Table 4. Summary of the Differences in both Experiments.

Property	Experiment 1	Experiment 2
Validation Technique	Individual Reviews	Individual Reviews
Design Type	Between-Subjects	Within-Subjects
Groups	2	1
Conditions	1 per Group	2 pro Participant
No. of Participants	133	31
Relative Experience	1 <sup>st</sup> to 3 <sup>rd</sup> Semester Undergraduate	3 <sup>rd</sup> to 5 <sup>th</sup> Semester Undergraduate
Instructional Background	Fundamental Requirements Engineering Course focusing on Validation, and Modeling using UML activity diagrams	Advanced Software Engineering Course focusing on Model-based Artifacts and Software Development Phases
Experimental Stimuli (see Section 5.2)	10 conceptual mitigations in either 10 activity diagrams with FHA excerpts (control group/condition) or 10 Hazard Relation Diagrams (control group/condition)	

## 5.2. Experimental Stimuli

In both experiments, we used an excerpt of a requirements specification for the Adaptive Cruise Control system shown in Fig. 1. This requirements specification was developed in close collaboration with industry partners from a larger, multi-site research project. It entailed one activity diagram comprising five hazard-inducing requirements, for which a Functional Hazard Analysis (FHA) was conducted (see Table 1 for an excerpt). A total of ten hazards were identified during the FHA, five of which were randomly selected and *adequately* mitigated. For each hazard, we derived a variation of the activity diagram from Fig. 1 containing the hazard-mitigating requirements. This yielded five activity diagrams containing adequate hazard-mitigating requirements. The other five hazards were *inadequately* mitigated. For each of those five hazard, a variation of the activity diagram from Fig. 1 was created, containing hazard-mitigating requirements with semantic mistakes allowing a hazard to still occur during operation. Consequently, the five additionally created activity diagrams contain inadequate hazard-mitigating requirements. We used all ten activity diagrams as control condition of the experiments (see Section 5.3). For the treatment condition, each adequate and inadequate activity diagram was extended into a corresponding Hazard Relation Diagram using our algorithms presented in Section 4.3. Deliberate semantic mistakes, which would not influence the safety of the ACC in operation, were included into the Hazard Relation Diagrams and activity diagrams, respectively. In addition, some FHA safety goals for inadequately mitigated hazards were replaced with nonsense safety goals. This was done to allow for decoy defects such that for every activity diagram and Hazard Relation Diagram, at least one correct rationale would exist which pertains to diagram semantics, diagram syntax, trigger conditions, safety goal, and the conceptual mitigation itself. The

introduced decoy defects were the same for treatment and control conditions. Overall, the experimental material<sup>5</sup> consist of ten activity diagrams for the control condition and ten Hazard Relation Diagrams for the treatment condition. The control condition furthermore included FHA results. All experimental stimuli for both experiments were in English.

### 5.3. Experimental Procedure

As outlined in Section 5.1, Experiment 1 was designed as a one-way between-subjects experiment [51]. Hence, participants were separated into a treatment group (reviews using Hazard Relation Diagrams) and a control group (reviews using conventional activity diagrams and FHA results) and only had to be instructed in one, rather than two modeling languages. In Experiment 2, all participants performed the same review task as in Experiment 1, but participated in both treatment condition (reviews using Hazard Relation Diagrams) as well as the control condition (reviews using conventional activity diagrams and FHA results). The design of the two experiments are discussed in the following sections.

#### 5.3.1. Experiment 1: Between-Subjects Individual Reviews

The experiment was implemented using the survey website SoSci Survey [53]. A short briefing was administered during a class session, which instructed participants on the fundamentals of safety engineering including Functional Hazard Analysis, embedded software, and function modeling. The unaltered ACC specification excerpt (akin to Fig. 1) was used during these pre-experimental briefings.

After all students were briefed, a link to the survey website was made available and the students were given five days to complete the experiment. Due to scheduling conflicts regarding the class from which participants were recruited (see Section 6.1), the experiment could not be conducted during class time. In consequence, the authors had little control over the manner in which the participants conducted the experiment. The experiment was designed such that it could not be discontinued nor resumed. The participants were informed during the briefing that they had to do the experiment in one session. Once a participant closed the survey, e.g., by closing the browser window, navigating away from the survey website, etc., this was recorded by the SoSci Survey tool as a discontinuation. Albeit during the briefing, participants were discouraged to

---

<sup>5</sup> For researchers interested in replicating our experiments, the experimental material can be downloaded from <https://goo.gl/XwJJQu>

“pause” participation, e.g., by leaving the survey open in the browser and leaving the computer, we could not control such behavior. Nevertheless, such behavior become quite apparent in several cases by the extraordinarily long time a participant needed to finish the survey. During data analysis, we controlled for such cases (see Section 5.5).

The experimental procedure consisted of six steps as shown in Fig. 7.

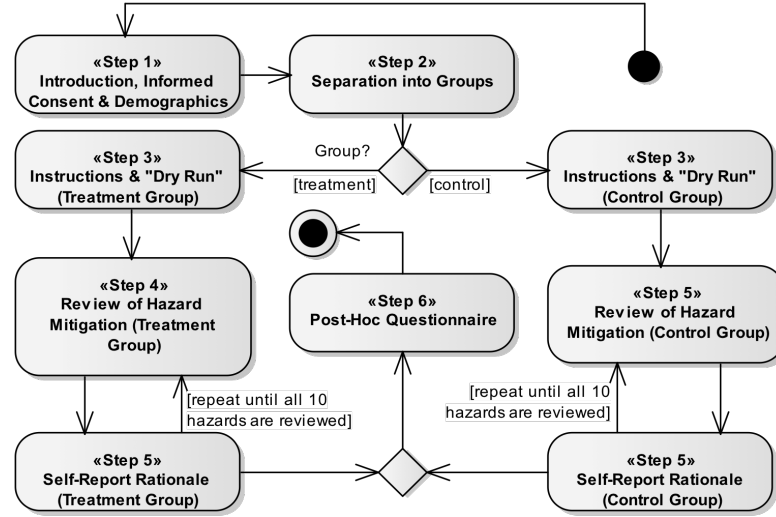


Fig. 7. Experimental Procedure of Experiment 1.

**Step 1: Introduction, Informed Consent, & Demographics.** Experiment 1 began with a short introduction, where informed consent as well as demographic data were collected. In this step, we told the students about their right to discontinue participation without penalty and assured that the collected data will be treated anonymously.

**Step 2: Separation into Groups.** Participants were randomly assigned to the *treatment group*, which conducted reviews using Hazard Relation Diagrams, and to the *control group*, which conducted reviews using conventional activity diagrams and FHA results. Assignment took into consideration the participants’ self-reported experience levels from the demographic questionnaire (see Section 5.4), such that both groups contained an approximately equal number of participants with corresponding experience levels. The assignment to the groups was done automatically using functionality provided by the SoSci Survey tool.

**Step 3: Instructions & “Dry Run.”** Both groups were presented with written instructions on how to review the experimental material once again. These instructions summarized the information given in the pre-experimental briefing. Furthermore, two example runs were performed in which the participants could rehearse the review task.

**Step 4: Review of Hazard Mitigation.** In this step, we asked participants to review the conceptual mitigations pertaining to one randomly selected hazard. In order to reduce



primacy, recency, and carry-over effects [51], for each participant the sequence in which the ten activity diagrams or Hazard Relation Diagrams were presented was randomized. We used the SoSci Survey tool for randomization of the order of the diagrams. In order to ensure that both participant groups reviewed approximately equally many information items, the control group was only shown one row from the FHA result table relevant to the hazard being mitigated in the activity diagram. Participants could review each activity diagram or Hazard Relation Diagram for an indeterminate amount of time. Their response times were recorded automatically by the SoSci Survey tool. Participants were asked to indicate “yes” if a hazard may still occur during operation and “no” otherwise. Participants could change their assessment as often as they wished before advancing to the next step.

**Step 5: Self-Report Rationale.** In this step, participants were asked to state a brief reason why they chose “yes” or “no” in the previous step. Due to technical reasons, this rationale could not be recorded in the step as adequacy judgments and confidence from Step 4. Therefore, participants were given the opportunity to return to the previous step and change their answer if thinking about the rationale made them change their mind. Furthermore, the experimental stimulus along with their decision from the previous step was shown for reference.

**Step 6: Post-Hoc Questionnaire.** After all ten conceptual mitigations were reviewed, both groups were presented with the post-hoc questionnaire in order to record participants’ subjective confidence when conducting reviews using the groups’ respective notation (see Section 5.4).

### 5.3.2. Experiment 2: Within-Subjects Individual Reviews

Like Experiment 1, Experiment 2 was implemented using the survey website SoSci Survey [53]. We copied the implementation from Experiment 1 and adapted it for a repeated measures design. An introductory briefing session was administered during a class meeting, in which we introduced the principles of software validation and safety engineering, including Functional Hazards Analysis of embedded systems, as well as UML activity diagrams. This briefing was identical to that of Experiment 1 (see Section 5.3.1). After the briefing, we informed the participants that during the next class meeting, an experiment will take place. To teach the participants how reviews are executed we used the unaltered ACC specification excerpt from Experiment 1. During the next class meeting, data collection for Experiment 2 commenced and consisted of the five steps depicted in Fig. 8.

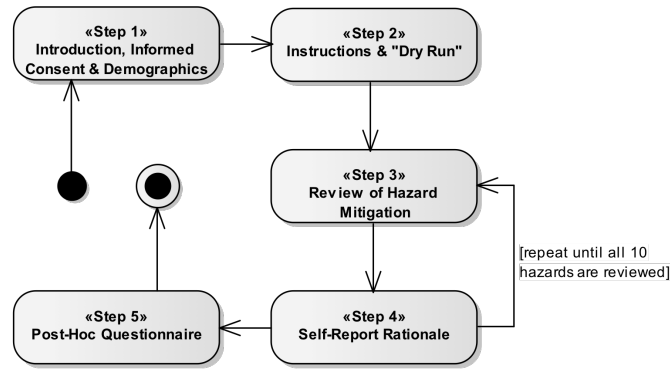


Fig. 8. Summary of the Experimental Procedure of Experiment 2.

**Step 1: Introduction, Informed Consent, & Demographics.** Like Experiment 1, Experiment 2 began with a short introduction, the retrieval of informed consent as well as the collection of demographic data. Students were again informed that participation is voluntary, anonymous, and that they are free to leave at any point without penalty.

**Step 2: Instructions & “Dry Run.”** In contrast to Experiment 1, participants were not separated into groups, as the review task would be conducted for both treatment and control condition. For this purpose, the review task was explained once again by summarizing the information from the briefing session. Two example runs were conducted using both treatment and control condition in order to rehearse the review task.

**Step 3: Review of Hazard Mitigation.** Like in Step 4 of Experiment 1 (see Section 5.3.1), this step entailed the review of conceptual mitigations pertaining to one random hazard. In total, ten conceptual mitigations were reviewed. In contrast to Experiment 1, Experiment 2 featured a within-subjects design. Thus, the conceptual mitigations presented to participants were equally distributed between the treatment and the control condition. In other words, five activity diagrams with FHA results and the five Hazard Relation Diagrams were presented. Like in Experiment 1, one conceptual mitigation was presented at a time. In order to reduce to primacy, recency, and carry-over effects [51], this was done by automatically randomizing the order of experimental stimuli for each participant. For each conceptual mitigation, an activity diagram or a Hazard Relation Diagram was available (i.e. a total of twenty stimulus diagrams, two for each hazard, see Section 5.2). Whether a conceptual mitigation was shown as an activity diagram or as a Hazard Relation Diagram was randomized as well. Each participant thus dealt with five activity diagrams and five Hazard Relation Diagrams (i.e. ten conceptual mitigations altogether). The diagrams used represented five adequate conceptual mitigations and five inadequate conceptual mitigations (regardless if these mitigations were presented as an activity diagram or Hazard Relation Diagram). Furthermore, we ensured that each

participant got adequate and inadequate conceptual mitigations in Hazard Relation Diagrams as well as activity diagrams. We used the SoSci Survey tool for the randomization and the balancing of diagrams used. As in Experiment 1, only the one row from the FHA result table relevant to the hazard being mitigated was displayed in the control condition in order to ensure that both conditions reviewed approximately equally many information items. Furthermore, like in Experiment 1, participants could review each conceptual mitigation for an indeterminate amount of time. We recorded the response times were recorded and participants were asked to indicate “yes” if the hazard reviewed may still occur during operation and “no” otherwise.

**Step 4: Self-Report Rationale.** Like in Experiment 1, participants provided a brief written reason why they chose “yes” or “no” in the previous step. If participants changed their opinion while documenting the reason for their decision about the conceptual mitigation they could return to the step 3 and change their yes-no answer.

**Step 5: Post-Hoc Questionnaire.** Like in Experiment 1, after all ten hazard mitigation were reviewed, the post-hoc questionnaire was presented to the participants in order to record participants’ subjective confidence when conducting reviews using the groups’ respective notation (see Section 5.4).

## 5.4. Hypotheses and Metrics

As outlined in RQ1, the primary aim of the experimental evaluation of Hazard Relation Diagrams was to investigate the influence of Hazard Relation Diagrams on objectivity. Therefore, we defined the following two-tailed hypothesis:

- **Hypothesis H1:** There is a difference in objectivity between treatment condition and control condition when validating hazard-mitigating requirements.

Table 5 summarizes the independent variable used in both experiments along with the dependent variables and metrics for hypothesis H1.

Table 5. Independent Variable (IV) and Dependent Variables (DV) as well as Metrics for Hypothesis H1.

Assoc Hypo	Variable		
	ID	Definition	Experiment
-	IV	Presentation Mode used during Validation. Two Levels: Using Hazard Relation Diagram (treatment) vs. Using Activity Diagrams and FHA Result Tables (control)	both
H1	DV1	# rationales mentioning semantics	both
	DV2	# rationales mentioning syntax	
	DV3	# rationales mentioning trigger conditions	
	DV4	# rationales mentioning safety goals	
	DV5	# rationales mentioning conceptual mitigations	
	H1a	# rationales mentioning diagram properties, i.e. DV1 + DV2	
	H1b	# rationales mentioning contextual information about the hazard, i.e. DV3 + DV4 + DV5	

We accept hypothesis H1 if there is a significant difference between treatment and control condition in the combination of DV1 and DV2 (in the following H1a) and the treatment condition has a lower mean for H1a. At the same time, there must be a significant difference between in the combination of DV3, DV4, and DV5 (in the following H1b) and the treatment condition has a higher mean for H1b.

Evaluating objectivity alone is not sufficient. In particular, validation using Hazard Relation Diagrams ought to be at least as effective and at least as efficient as validation using conventional activity diagrams and FHA results, as otherwise, a possible increase in objectivity may not be at the expense of reduced effectiveness and efficiency. To investigate these relationships, we defined additional two-tailed hypotheses for RQ2:

- **Hypothesis H2:** There is a difference in effectiveness between treatment condition and control condition when validating hazard-mitigating requirements.
- **Hypothesis H3:** There is a difference in efficiency between treatment condition and control condition when validating hazard-mitigating requirements.

Table 6 summarizes the dependent variables for hypotheses H2 and H3. Due to the difference in experimental design in Experiment 1 and 2, there are slight differences in the metrics for each DV, as can be seen from Table 6.

Table 6. Dependent Variables (DV) and Metrics for Hypothesis H2 and H3.

Assoc Hypo	Variable		
	ID	Definition	Experiment
H2	DV6	# of true positive answers	Exp. 1
		ratio of true positive answers	Exp. 2
	DV7	# of true negative answers	Exp. 1
		ratio of true negative answers	Exp. 2
	DV8	# of false positive answers	Exp. 1
		ratio of false positive answers	Exp. 2
	DV9	# of false negative answers	Exp. 1
		ratio of false negative answers	Exp. 2
H3	DV10	time needed for true positive answers	both
	DV11	time needed for true negative answers	
	DV12	time needed for false positive answers	
	DV13	time needed for false negative answers	
	H3total	overall time needed for all answers	

As outlined in Section 5.2, we differentiated the experimental stimuli into diagrams depicting adequate conceptual mitigations and diagrams depicting inadequate conceptual mitigations for both the control condition and treatment condition. For each diagram depicting an *adequate* conceptual mitigation, we recorded whether or not the participants indicated “adequate” during validation. This was considered a *true positive* answer (i.e. the answer was supposed to be “adequate” and the participants answered correctly, DV6

in the following). If participants indicated “adequate” during review of a diagram that depicted an *inadequate* conceptual mitigation, this was considered a *false positive* answer (i.e. the answer was supposed to be “inadequate” and the participants answered incorrectly, DV8 in the following). Equivalently, we recorded *true negative* answers (DV7) and *false negative* answers (DV9) as well. In Experiment 1, the participants’ effectiveness was determined by the amount of true positive, false positive, true negative, and false negative answers – a perfect score would therefore be five true positive and five true negative answers. In Experiment 2, there was an uneven amount of adequate and inadequate conceptual mitigations on both experimental conditions (see Section 5.3.2). We therefore determined effectiveness by measuring the ratio of true positive to false positive answers and true negative to false negative answers, respectively. In both experiments, we refrained from determining effectiveness by means of the number of identified defects in the rationales, as giving at least one rationale was mandatory in order to correctly measure hypothesis H1 in order to avoid skewing. We accept hypothesis H2, if there is a significant difference between DV6 through DV9, and the mean in the treatment condition is higher for DV6 and DV7 and lower in DV8 and DV9.

In order to measure efficiency of reviews, we measured the time needed for true positive, true negative, false positive, and false negative answers. This was done automatically by the SoSci Survey tool by measuring the time a stimulus was presented until an adequacy judgment was made (i.e. the user clicked the corresponding button). We accept hypothesis H3, if the mean time needed for true positive (DV10), true negative (DV11), false positive (DV12), and false negative answers (DV13) as well as the overall time needed (in the following H3total) is lower in the treatment condition.

The term “objectivity” does not only pertain to the validation results, but also to the individual validator’s confidence in their own review judgments. To account for this, we defined the following two-tailed hypotheses for RQ3:

- **Hypothesis H4:** There is a difference in self-reported confidence between treatment condition and control condition when validating hazard-mitigating requirements.

To measure the validators’ confidences, we selected items from the technology acceptance model version 3 (TAM3, see [55]) and the task-technology fit model (TTF, see [56]). TAM3 and TTF are questionnaire instruments designed to measure various aspects of technology acceptance and task appropriateness. In order to select the items that are relevant for our study, we applied the Goal/Question/Metric Method [54], as summarized in Table 7.

Table 7. TAM3 and TTF Item Selection using a Goal/Question/Metric Template based on [54].

Object	Purpose	Metric for	Role	Development Situation
Hazard Relation Diagram	Evaluate	Subjective Confidence	Requirements Engineer	Validation of the adequacy of hazard-mitigating requirements.
Quality Focus	Source	Adapted TAM3 / TTF Definitions		
Perceived Usefulness	TAM3	The degree to which someone believes that using Hazard Relation Diagrams or activity diagrams with FHA results to validate hazard-mitigating requirements adequacy enhances her performance.		
Self-Efficacy	TAM3	The degree to which an individual believes that she has the ability to validate hazard-mitigating requirements adequacy using Hazard Relation Diagrams or activity diagrams with FHA results.		
Result Demonstrability	TAM3	The degree to which an individual believes that the validation results using Hazard Relation Diagrams or activity diagrams with FHA results are tangible, observable, and communicable.		
“The Right Data”	TTF	The degree to which the needed information to validate hazard-mitigating requirements adequacy is maintained when using Hazard Relation Diagrams or activity diagrams with FHA results.		
Meaning	TTF	The ease of determining what a modeling element in a Hazard Relation Diagram, activity diagram, or FHA result means, or what information is depicted therein.		
Presentation	TTF	The degree to which a Hazard Relation Diagram or an activity diagram with an FHA result is understandable.		
Training	TTF	The amount of available training in order to use Hazard Relation Diagrams or activity diagrams with FHA results to validate hazard-mitigating requirements adequacy.		
Confusion	TTF	The degree to which the user is confused in using Hazard Relation Diagrams or activity diagrams with FHA results to validate hazard-mitigating requirements adequacy.		

To measure the quality foci of subjective confidence, we compiled a post-hoc questionnaire<sup>6</sup>. This post-hoc questionnaire consisted of a series of demographic questions as well as the questions pertaining to each selected TAM3 and TTF item. TAM3 and TTF questions were randomized. For Experiment 1, we measured participant responses on a 5-point-Likert scale [1: “disagree”; 2: “somewhat disagree”; 3: “neither agree nor disagree”; 4: “somewhat agree”; 5: “agree”], measuring the participants’ relative agreement with the questionnaire item with regard to the diagrammatic representation they received in their respective condition. For Experiment 2, we converted the Likert-scale to a semantic differential (see [57], [58]), which polarizes ordinates on measurement scales. For this purpose, we used Likert-style 5-point ordinates ranging from “1: true for activity diagrams” to “5: true for Hazard Relation Diagrams”. A median of “3” hence indicated indifference between both presentation modes. In both experiments, we adapted questionnaire questions to be neutral with regard to the used notations in order to minimize the validity threat of hypothesis guessing [51]. We calculated the degree agreement for each quality focus by the sum of the answers for each question pertaining to the respective quality focus (adding the inverse ordinate for negated questions) and dividing the sum by the number of questions in the respective quality focus. Table 8 summarizes the dependent variables and metrics for hypothesis H4.

<sup>6</sup>The post-hoc questionnaire is available at <https://goo.gl/XwJJQu>

Table 8. Dependent Variables (DV) and Metrics for Hypothesis H4.

Assoc Hypo	Variable		
	ID	Definition	Experiment
H4	DV14	degree of agreement of questions pertaining to “Perceived Usefulness”	both
	DV15	degree of agreement of questions pertaining to “Self-Efficacy”	
	DV16	degree of agreement of questions pertaining to “Result Demonstrability”	
	DV17	degree of agreement of questions pertaining to “The Right Data”	
	DV18	degree of agreement of questions pertaining to “Meaning”	
	DV19	degree of agreement of questions pertaining to “Presentation”	
	DV20	degree of agreement of questions pertaining to “Training”	
	DV21	degree of agreement of questions pertaining to “Confusion”	
	H4total	degree of agreement of subjective confidence	

## 5.5. Data Preparation

After each experiment, the data of all fully completed data sets were downloaded from the survey website. Incomplete data sets (e.g., due to participants who discontinued participation) was discarded, and data was transcoded into a data file for the statistical analysis tool SPSS.

Rationales were qualitatively analyzed and prepared for statistical analysis by means of constant comparison [59]. The constant comparison technique entails reading every statement and categorizing the statements according to constant concepts mentioned in the statement. This was done by counting the number of statements in each rationale (i.e., when a rationale included conjunctions such as “and” or “furthermore” or when multiple rationales were given). If a rationale included phrases such as “don’t know,” “just guessed,” or other indications that the participant did not provide a proper reason (e.g., placeholders such as “...,” “bla,” or a blank text field), this was seen as an invalid rationale and not counted. All valid rationales were categorized into exactly one of the following categories:

- **Semantics:** The rationale was based on the individual’s understanding of the semantics of the subject matter (DV1).
- **Syntax:** The rationale was based on the individual’s understanding of the diagram notation (DV2).
- **Trigger Condition:** The rationale indicated that for the hazard one or more trigger conditions are not avoided, or are not successfully avoided (DV3).
- **Safety Goal:** The rationale indicated that the safety goal is fulfilled, is not fulfilled or is semantically wrong (DV4).
- **Mitigation:** The rationale explained how and why the hazard-mitigating requirements do or do not lead to resolution of the hazard (DV5).

Rationales that could not be coded into any of those category were counted as invalid and discarded. The coding process was quality assured. Initial qualitative analysis and categorization was done by the authors, where each rationale was categorized independently from all other rationales. Each rationale categorization and transcoding was verified by three independent researchers. Two of the verifiers were members of the co-authors' research group in Germany and the third verifier was a member of the faculty of the first co-author's US affiliation. The purpose of independent verification was to identify erroneously coded data. Each derivation between the authors' and the verifiers' categorization was individually reviewed, bilaterally discussed, and rectified. In case of unresolvable conflicts, the rationale was discarded as invalid. Moreover, in order to reduce researcher bias (see Section 8), each rationale was categorized without knowledge about what condition (i.e. treatment or control) the rationale in question belonged to. To ensure this, a student assistant temporarily removed the information pertaining to group membership and experimental condition from the data file during rationale categorization. After the coding and verification process concluded, the same student assistant re-introduced the group membership and experimental condition information into the SPSS data file by using participant IDs.

For each adequacy judgment (i.e. yes/no indication for each diagram), it was recorded whether this was a correct or incorrect responses, thereby determining true positive (DV6), true negative (DV7), false positive (DV8), and false negative (DV9) answers. Furthermore, the time needed for all true positive (DV10), true negative (DV11), false positive (DV12), and false negative (DV13) answers was summed up and the total time needed for all answers was computed (H3total). In both experiments, the time needed was determined by the time a particular stimulus was presented on screen.

Using the prepared data, descriptive statistics and frequencies were computed using the statistical analysis tool SPSS. Moreover, to identify outliers and irregular responses the data was manually reviewed. Irregular responses entailed all participant data sets, which contained three or fewer valid rationales for all hazards or where participants answered in patterns (e.g., alternating yes/no answers, using always the same answer, etc.). Such irregular responses were discarded. In addition, since in Experiment 1 participants were able to run the experiment at home, we were unable to control for participants "pausing" participation (e.g., by leaving the survey open in the browser and leaving the computer) or not fully concentrating on the experiment and doing other things while participating (see Section 5.3.1). In addition, we removed a participant's data set,



if participants took less time than 5.5 minutes (since this was the minimal time needed to view each stimulus and questionnaire page once) or more than 40 minutes (since this was about twice the mean of total response times, which is a standard exclusion criterion for multivariate statistics).

Since the original TAM3 and TTF items and questionnaire questions were adapted to suit the needs of the experiments, it was necessary to validate questionnaire cohesion. For this purpose, Cronbach's  $\alpha$  [60] was computed on the post-hoc questionnaire items for the quality foci from Table 7. For further analysis, only quality foci were retained where cohesion was at least high ( $\alpha > 0.7$ ). The results for each experiment are shown in Table 9. Retained quality foci are marked in dark grey.

Table 9. Cronbach's alpha for all quality foci from Table 7 and all experiments.

TAM3 / TTF item		Cronbach's $\alpha$	
DV ID	Quality Focus	Experiment 1	Experiment 2
DV14	Perceived Usefulness	<b>0.906</b>	<b>0.930</b>
DV15	Self-Efficacy	<b>0.796</b>	0.405
DV16	Result Demonstrability	<b>0.793</b>	<b>0.774</b>
DV17	The Right Data	0.629	0.439
DV18	Meaning	<b>0.747</b>	<b>0.740</b>
DV19	Presentation	0.502	<b>0.928</b>
DV20	Training	0.641	< 0.01
DV21	Confusion	<b>0.736</b>	<b>0.786</b>
H4total	Subjective Confidence	<b>0.914</b>	<b>0.869</b>

## 6. Participant Demographics

In order to facilitate easier comparison between the experimental results presented in Section 7, we compare the participant populations. Section 6.1 discusses recruitment of the participants. Section 6.2 compares experience levels.

### 6.1. Recruited Participants

Participants for **Experiment 1** were recruited from an undergraduate requirements engineering course instructed by the authors at the University of Duisburg-Essen in fall 2014. The course featured a larger undertaking aimed at instructing the fundamentals of conceptual modeling in preparation of a grade-determining final exam. In particular, this included instruction in static-structural, behavioral, and functional models. Among others, activity diagrams were discussed in depth. A total of 123 students participated in Experiment 1. These were undergraduate ( $n = 110$ ) or graduate students ( $n = 13$ ). Graduate students were enrolled in the undergraduate requirements engineering course to fulfill a prerequisite for their graduate program. All students were enrolled in Software Systems Engineering or Business Information Systems degree programs. A pilot study was conducted with ten participants [21]. These participants were members from the

authors' research group. Participants from the pilot study were included in the sample for Experiment 1, which yielded a total of 133 data sets ( $n_{\text{treatment}} = 67$ ,  $n_{\text{control}} = 65$ ) after data preparation (see Section 5.5). Although gender and age were not assumed to impact the results of the experiment, we recorded a total of 117 male participants, sixteen female participants, and a participants' age between 18 and 36 years ( $\mu = 23.9$ ,  $\sigma = 3.66$ ).

Participants for **Experiment 2** were recruited from an undergraduate software engineering course instructed by members of the first author's department at the State University of New York at Oswego in fall 2015. The course featured all aspects of software engineering in industry projects, with a key focus on process models, software quality assurance, safety engineering, as well as semi-formal languages (specifically, activity diagrams). Prior to recruitment in the course, approval of the institution's research ethics board was obtained. A total of 31 students participated in Experiment 2, all of which were undergraduates enrolled in Electrical Engineering, Software Engineering, or Computer Science degree programs. The majority of eighteen participants were male with two female participants, eleven participants declined to provide gender information. Participants' age was between 18 and 45 years ( $\mu = 22.95$ ,  $\sigma = 5.995$ ). Again, gender and age are assumed to not impact the results of the experiment.

## 6.2. Participant Experience Levels

As outlined in Section 5.4, the post-hoc questionnaire administered in both experiments contained several demographic questions. The questions pertained to the participants' levels of experience with automotive software engineering (since the case example was from the automotive domain), requirements engineering in general, modeling using activity diagrams (since this is the foundation of Hazard Relation Diagrams), static requirements quality assurance using reviews or inspections (since this was the experimental task), dynamic quality assurance (i.e., testing and verification), and functional design and system architecture (since experience in this area may increase participants ability to think abstractly about diagrams in general). Each experience level question was measured on a 5-point-Likert scale [1: "no experience"; 2: "experience from academic homework"; 3: "experience from one or more academic projects"; 4: "experience from one industry project"; 5: "experience from multiple industry projects"]. We have deliberately chosen this scale, as these ordinates yield more objective categorizations than ordinates that, for instance, ask for "very little experience" to "very much experience."

To investigate differences in experimental populations, we computed the mean experience level ( $\mu$ ) and standard deviations ( $\sigma$ ) by summing up the numeric value of the ordinates. We then computed independent group T-Tests [61] to investigate the significance of the mean differences between groups. Afterwards, we calculated Cohen's  $d$  [63] for statistical power in T-Tests.

Table 10 shows the levels of experience of participants in both experiments. For each ordinate, Table 10 shows the total as well as the relative number of participants. Table 11 summarizes the means ( $\mu$ ), standard deviations ( $\sigma$ ), T-Test results ( $t$ ,  $dF$ ), significance ( $p$ ), effect size ( $d$ ), and statistical power ( $\eta^2$ ) of participants' levels of experience. Significant T-Test results and higher means are marked in dark grey in Table 11. Three participants in Experiment 1 and one participant in Experiment 2 declined to give demographic information. Hence, the total  $n$  for demographics in Experiment 1 is 130 and 30 for Experiment 2.

Table 10. Participants' Relative Levels of Experience.

Experience Level in...	Exp.	no experience		academic homework		academic projects		industry project		multiple industrial projects	
Automotive Software Engineering	1	62	47.3%	54	41.2%	11	8.4%	2	1.5%	2	1.5%
	2	23	76.7	2	6.7%	3	10.0%	0	0.0%	2	6.7%
Requirements Engineering	1	12	9.2%	69	52.7%	39	29.8%	8	6.1%	3	2.3%
	2	4	13.3	11	36.7%	8	26.7%	3	10.0%	4	13.3%
Modeling using Activity Diagrams	1	20	15.3%	61	46.6%	39	29.8%	10	7.3%	1	0.8%
	2	4	13.3%	12	40.0%	10	33.3%	2	6.7%	2	6.7%
Reviews and Inspections	1	45	34.4%	68	51.9%	14	10.7%	4	3.1%	0	0.0%
	2	8	26.7%	8	26.7%	7	23.3%	4	13.3%	3	10.0%
Req. Test and Software Test	1	38	29.0%	62	47.3%	25	19.1%	6	4.3%	0	0.0%
	2	8	26.7%	10	33.3%	5	16.7%	4	13.3%	3	10.0%
Design and System Architecture	1	46	35.1%	52	36.7%	27	20.6%	6	4.3%	0	0.0%
	2	6	20.0%	10	33.3%	8	26.7%	3	10.0%	3	10.0%

Table 11. Means, Std. Deviations, T-Test, and Power for Participants' Levels of Experience.

Experience Level in...	Exp.	$\mu$	$\sigma$	$t$	$dF$	$p$	$d$	$\eta^2$
Automotive Software Engineering	1	1.69	0.814	-0.701	31.104	0.488	0.162	0.609
	2	1.53	1.137					
Requirements Engineering	1	2.40	0.829	1.426	32.265	0.163	0.315	0.563
	2	2.73	1.230					
Modeling using Activity Diagrams	1	2.32	0.853	1.041	38.838	0.304	0.221	0.541
	2	2.53	1.042					
Reviews and Inspections	1	1.82	0.739	2.870	33.366	<b>0.007</b>	0.669	0.716
	2	<b>2.53</b>	<b>1.306</b>					
Req. Test and Software Test	1	1.99	0.818	1.905	34.383	0.065	0.441	0.625
	2	2.47	1.306					
Design and System Architecture	1	1.95	0.862	2.632	35.874	<b>0.012</b>	0.586	0.638
	2	<b>2.57</b>	<b>1.223</b>					

Post-experimental analyses revealed that both participant populations are largely homogeneous (cf. Table 10 and Table 11). The majority of participants in both experiments claimed either no experience from the respective areas or experience from academic work (see Table 10). A notable exception are three participants from

Experiment 2, who claimed industry experience in reviews, inspections, dynamic quality assurance, and architecture. These three individuals were students, who had a side-job in local software development companies. In consequence, “reviews and inspections” as well as “design and system architecture” were the only experience areas with a significant difference in means between both experimental populations (see Table 11). We assume that this difference in experience had little impact on the experimental results. This assumption is supported by the medium effect size and statistical power ( $0.3 < d$ ,  $\eta^2 < 0.8$ ) for both significant T-Test results (see Table 11). Moreover, independent samples T-Tests are sensitive to largely unequal  $n$ . Hence, the difference in population ( $n_{\text{Experiment 1}} = 130$ ,  $n_{\text{Experiment 2}} = 30$ ) size is likely to have exaggerated significance.

## 7. Results of the Experimental Evaluation

After data preparation and outlier identification (cf. Section 5.5), F-Tests were computed on the prepared data to test for normal distribution of measurements. To determine the mean differences between treatment and control groups in Experiment 1, independent group T-Tests [61] were conducted on all DVs shown in Table 5, Table 6, and Table 8. Given the repeated measures design of Experiment 2, a paired-sample repeated measures T-Test [62] was conducted for all variables pertaining to hypothesis H1 to hypothesis H3 (see Table 5 and Table 6). For hypothesis H4, the variables were measured by means of a semantic differential, hence only allowing for descriptive comparisons. Furthermore, to calculate effect size and achieved statistical power for the two experiments, Cohen’s  $d$  [63] was computed for all T-Test results.

We report on the experimental results for all hypotheses in the following subsections. Based on the results we state for each experiment, whether a hypothesis can be accepted or must be rejected. To foster easier discrimination, we added a suffix to each variable listed in Table 5, Table 6, and Table 8. For example, variable DV6.2 refers to the results for variable DV6 in Experiment 2. In Section 7.5, we comparatively discuss the experimental results with regard to the impact of Hazard Relation Diagrams on validation of hazard-mitigating requirements at large.

### 7.1. Results for Hypothesis H1: Rationale Objectivity

Results from **Experiment 1** indicate that the means for DV1.1, DV2.1, and H1a.1 are consistently higher in the control group than in the treatment group. This means that the control group based their adequacy judgment more often on diagram semantics or

diagram syntax than the treatment group. Conversely, the means for DV3.1, DV4.1, DV5.1, and H1b.1 are higher for the treatment group than for the control group. This indicates that the treatment group based their rationales more often on trigger conditions, safety goals, or conceptual mitigations than the control group. The results of the independent groups T-Test indicate that the differences in means between groups were highly significant for DV1.1, DV5.1, H1a.1, and H1b.1, respectively, yet not significant for DV2.1, DV3.1, nor DV4.1, respectively. Post-hoc power analyses revealed a very large effect size and statistical power for all significant T-Test results, while for all non-significant results, a small or medium effect size was revealed. High statistical power was achieved for all variables, except DV2.1 and DV4.1.

Due to the significant differences between treatment and control group for H1a.1 and H1b.1, hypothesis H1 must be accepted for Experiment 1. In other words, Experiment 1 shows that using Hazard Relation Diagrams adequacy judgments are more often based on contextual information about the hazard and, thus, review objectivity is increased. Table 12 summarizes the results pertaining to hypothesis H1. In this and the following result tables, significant results and better performing groups (in the sense of the respective hypothesis) are marked in dark grey.

Table 12. Means, Std. Deviations, T-Test, and Power for H1 in Experiment 1.

Var	Grp	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV1.1	treatment	1.44	1.393	-8.36	118	< 0.001	1.536	1.000
	control	4.79	2.751					
DV2.1	treatment	0.29	0.720	-0.71	118	0.481	0.123	0.172
	control	0.39	0.900					
DV3.1	treatment	1.10	1.410	2.614	118	0.10	0.477	0.860
	control	0.57	0.694					
DV4.1	treatment	1.44	1.755	1.239	118	0.218	0.223	0.355
	control	1.07	1.559					
DV5.1	treatment	5.37	2.304	7.024	118	< 0.001	1.278	1.000
	control	2.69	1.867					
H1a.1	treatment	1.73	1.799	-8.13	118	< 0.001	1.488	1.000
	control	5.18	2.742					
H1b.1	treatment	7.92	2.967	7.059	118	< 0.001	1.288	1.000
	control	4.33	2.593					

Results for **Experiment 2** show that the means in the treatment condition for DV1.2, DV2.2, and the combined variable H1a.2 are lower than in the control condition. This indicates when reviewing conceptual mitigations and using Hazard Relation Diagrams review rationales were less often based on diagram semantics and diagram syntax – as opposed to using activity diagrams and FHA result tables. Conversely, when using Hazard Relation Diagrams, review rationales were more often based on trigger conditions, safety goals, or the conceptual mitigation in the treatment condition than in the control condition (variable H1b.2). A pairwise T-Test shows that these differences

between treatment condition and control condition was significant for DV1.2, DV3.2, DV4.2, H1a.2, and H1b.2. For DV5.2, the differences in means approaches significance, while the differences in means for DV3.2 was not significant. Power analyses show very large effect sizes for DV1.2 and H1a.2, while for DV3.2, DV4.2, DV5.2, and H1b.2, a medium effect size was achieved. Statistical power was high for DV1.2, H1a.2, and H1b.2 and medium for DV3.2, DV4.2, and DV5.2. Cohen's d and statistical power could not be computed for variable DV2.2, since in the treatment condition, no rationales mentioned diagram syntax leading to a mean of zero.

These results require hypothesis H1 to be accepted in Experiment 2: there is a significant effect of Hazard Relation Diagrams to improve objectivity within participants when reviewing conceptual mitigations. Table 13 summarizes the results for hypothesis H1 in Experiment 2.

Table 13. Means, Std. Deviations, T-Test, and Power for Hypothesis H1 in Experiment 2.

Var	Cond.	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV1.2	treatment	1.52	1.087	-5.151	29	< 0.001	1.0099	0.9781
	control	2.74	1.318					
DV2.2	treatment	0.00	0.000	-1.795	29	0.161	n/a	n/a
	control	0.07	.267					
DV3.2	treatment	0.74	0.944	-2.359	29	0.009	0.6599	0.7617
	control	0.26	0.447					
DV4.2	treatment	1.04	1.160	3.275	29	0.005	0.5799	0.6765
	control	0.44	0.892					
DV5.2	treatment	1.96	1.285	2.249	29	0.063	0.3858	0.4029
	control	1.52	0.975					
H1a.2	treatment	1.52	1.087	-5.677	29	< 0.001	1.0756	0.9879
	control	2.81	1.302					
H1b.2	treatment	3.74	1.723	5.673	29	< 0.001	0.9404	0.9612
	control	2.22	1.502					

## 7.2. Results for Hypothesis H2: Effectiveness

For **Experiment 1**, the means in the control group for DV6.1 and DV7.1 was higher than the means in the treatment group. Conversely, the means in the control group for DV8.1 and DV9.1 were lower than the means in the treatment group. This shows that the control group scored more correct answers and fewer wrong answers than the treatment group. However, results from the T-Test indicate that none of the means are significantly different and effect sizes as well as statistical power was small for all tests except DV8.1.

The lack of significance for all dependent variable pertaining to hypothesis H2 in Experiment 1 shows that the alternative hypothesis must be accepted: There is no difference between groups with regard to the number of correctly or incorrectly identified adequate or inadequate conceptual mitigations between both groups. Table 14 summarizes means, standard deviations, T-Test and power analyses results.

Table 14. Means, Std. Deviations, T-Test, and Power for Hypothesis H2 in Experiment 1.

Var	Grp	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV6.1	treatment	3.03	1.299	-1.23	118	0.222	0.226	0.362
	control	<b>3.31</b>	<b>1.177</b>					
DV7.1	treatment	3.17	1.289	-0.19	118	0.850	0.032	0.071
	control	<b>3.21</b>	<b>1.226</b>					
DV8.1	treatment	1.93	1.298	1.214	118	0.227	0.216	0.341
	control	<b>1.66</b>	<b>1.196</b>					
DV9.1	treatment	1.78	1.260	0.187	118	0.852	0.033	0.072
	control	<b>1.74</b>	<b>1.196</b>					

Results for **Experiment 2** show that for adequately mitigated hazards, participants in the treatment condition judged adequate conceptual mitigations more often correctly (DV6.2) than incorrectly (DV8.2). This performance is superior to the control condition. The same relationship holds for inadequately mitigated hazards. Interestingly, participants averaged at the same success and failure rate in the treatment condition, but were slightly more likely to make incorrect judgment in the control condition. T-Tests show significance for all these differences. Moreover, Cohen's d shows a large or very large effect size and statistical power for DV7.2, DV8.2, and DV9.2. Effect size and power was medium for correctly judged adequate conceptual mitigations (DV6.2).

These results provide compelling evidence for accepting hypothesis H2 in Experiment 2: there is a significant difference in effectiveness within participants between the treatment condition and the control condition when reviewing conceptual mitigations and using Hazard Relation Diagrams for review results in higher effectiveness. Table 15 summarizes these results.

Table 15. Means, Std. Deviations, T-Test, and Power for Hypothesis H2 in Experiment 2.

Var	Cond.	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV6.2	treatment	<b>0.7037</b>	<b>0.3180</b>	3.077	56	<b>0.011</b>	0.6141	0.7197
	control	0.5062	0.3252					
DV7.2	treatment	<b>0.7037</b>	<b>0.2824</b>	2.718	56	<b>&lt; 0.001</b>	0.86	0.9297
	control	0.4259	0.3591					
DV8.2	treatment	<b>0.2963</b>	<b>0.3180</b>	-3.112	56	<b>0.011</b>	1.5469	0.9996
	control	0.4938	0.3252					
DV9.2	treatment	<b>0.2963</b>	<b>0.2824</b>	-3.225	56	<b>&lt; 0.001</b>	0.86	0.9297
	control	0.5741	0.3591					

### 7.3. Results for Hypothesis H3: Efficiency

Descriptive statistics for **Experiment 1** show that the mean response time for all dependent variables DV10.1 through DV13.1 was lower in the treatment group than in controls. This means the treatment group was able to correctly detect adequate (true positive answers, DV10.1) and inadequate (true negative answers, DV11.1) conceptual mitigations more quickly than the control group. Furthermore, the treatment group had a lower response time for incorrect answers (DV12.1 and DV13.1). Consequently, it follows that the overall response time H3total.1 was also shorter in the treatment group than in the control group. T-Tests did not reveal significant differences in variables

DV11.1, DV12.1, nor DV13.1. However, the difference in means for true positive answers (DV10.1) reached significance. Post-hoc power analyses on the T-Test results revealed a medium effect size and statistical power for DV10.1 and H3total.1 as well as a small effect size and low statistical power for all other variables.

In light of the absence of significance in the differences between the treatment and control groups for variable H3total.1, the hypothesis for H3 must be rejected for Experiment 1. In other words, there is no significant difference between the two groups with regard to the time needed to review hazard-mitigating requirements using Hazard Relation Diagrams. Nevertheless, there is a clear trend for Hazard Relation Diagrams to positively influence efficiency, given that the response times were consistently faster in the treatment group for all dependent variables. The results depicted in Table 16.

Table 16. Means, Std. Deviations, T-Test, and Power for Hypothesis H3 in Experiment 1.

Var	Grp	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV10.1	treatment	248.17	157.124	-1.92	116	0.05	0.353	0.646
	control	305.65	168.001					
DV11.1	treatment	326.38	224.416	-0.97	116	0.336	0.178	0.266
	control	373.48	298.171					
DV12.1	treatment	176.36	163.282	-0.66	116	0.508	0.123	0.172
	control	199.20	206.925					
DV13.1	treatment	187.24	212.735	-0.72	116	0.641	0.133	0.118
	control	214.93	203.705					
H3total.1	treatment	938.16	418.790	-1.89	116	0.281	0.348	0.635
	control	1093.27	470.493					

Response times in **Experiment 2** for correctly judged adequate (DV10.2) and inadequate (DV11.2) conceptual mitigations was faster in the treatment condition than in the control condition. However, response times in the control condition were faster than in the treatment condition for incorrectly judged conceptual mitigations, both adequate and inadequate (DV12.2 and DV13.2, respectively). Furthermore, the overall response time (H3total.2) was on average faster in the control condition. However, it can be seen that response times only differed by five to fourteen seconds for all variables, indicating that the average time to review any stimulus was only marginally impacted by experimental condition or conceptual mitigation adequacy. This is supported by the relatively high standard deviations for all variables. In consequence, none of the pairwise T-Tests indicated any significance of the mean response times for the variables. Furthermore, effect size and statistical power was low for all variables as well.

Due the lack of significance in the difference of response times in both conditions for variable H3.2total, we must reject hypothesis H2. In other words, there is no significant difference with regard to the time needed to review hazard-mitigating requirements using Hazard Relation Diagrams as opposed to conventional activity



diagrams and FHA result tables. However, it is noteworthy that when using Hazard Relation Diagrams for review, participants were faster for all correct judgements (i.e. DV10.2 and DV11.2), but slower when they incorrectly judged a conceptual mitigation (i.e. DV12.2 and DV13.2). Table 17 depicts the results.

Table 17. Means, Std. Deviations, T-Test, and Power for Hypothesis H3 in Experiment 2.

Var	Cond.	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV10.2	treatment	<b>68.9048</b>	<b>60.1711</b>	-1.349	22	0.366	0.0202	0.1583
	control	78.5635	30.8712					
DV11.2	treatment	<b>89.5463</b>	<b>54.3600</b>	-0.870	18	0.349	0.2766	0.2028
	control	103.6667	47.5184					
DV12.2	treatment	76.5000	60.0778	1.043	13	0.406	0.2421	0.148
	control	<b>63.8718</b>	<b>42.8037</b>					
DV13.2	treatment	85.3111	66.8607	0.188	15	0.781	0.1030	0.0854
	control	<b>79.6333</b>	<b>40.0520</b>					
H3total.2	treatment	86.3407	38.4851	0.497	28	0.374	0.1561	0.1403
	control	<b>81.1111</b>	<b>27.6392</b>					

#### 7.4. Results for Hypothesis H4: Subjective Confidence

Results for **Experiment 1** indicate that participants in the treatment group reported a higher degree of agreement with the quality foci “Perceived Usefulness,” “Computer Self-Efficacy,” “Results Demonstrability,” and “Meaning” while at the same time reporting less confusion. Moreover, the independent samples T-Test reveals that the differences in between the means of the two groups were significant for DV15.1, DV18.1, DV21.1, and the for the combined variable H4total.1, yet not significant for any other variable. The post-test power analysis revealed a medium effect size and very high statistical power for the significant difference in DV21.1. Medium effect size and statistical power became apparent for the other significant variables as well as for the non-significant variable DV16.1. A small effect size and low statistical power was revealed for DV14.1.

In light of these results, we can accept hypothesis H4 in Experiment 1: participants using Hazard Relation Diagrams to review hazard-mitigating requirements report higher subjective confidence compared with participants using conventional activity diagrams and FHA result tables. Table 18 depicts these results.

Table 18. Means, Std. Deviations, T-Test, and Power for hypothesis H1 in Experiment 1.

Var	Grp	$\mu$	$\sigma$	t	dF	p	d	$\eta^2$
DV14.1	treatment	<b>3.602</b>	<b>0.786</b>	0.944	130	0.347	0.151	0.217
	control	3.476	0.880					
DV15.1	treatment	<b>3.758</b>	<b>0.629</b>	2.001	130	<b>0.047</b>	0.323	0.580
	control	3.521	0.824					
DV16.1	treatment	<b>2.818</b>	<b>0.472</b>	2.001	130	0.552	0.322	0.577
	control	2.641	0.618					
DV18.1	treatment	<b>3.500</b>	<b>0.878</b>	2.228	130	<b>0.027</b>	0.356	0.651
	control	3.188	0.876					
DV21.1	treatment	<b>2.904</b>	<b>0.840</b>	-3.881	130	<b>&lt; 0.001</b>	0.625	0.973
	control	3.494	1.039					
H4total.1	treatment	<b>3.410</b>	<b>0.491</b>	1.900	130	<b>0.05</b>	0.301	0.538
	control	3.245	0.588					

In **Experiment 2**, the means for “Perceived Usefulness,” “Result Demonstrability,” “Meaning,” “Presentation,” and “Confusion“ (DV14.2, DV16.2, DV18.2, DV19.2, and DV21.2, respectively) are around the neutral ordinate of “3: indifferent”. However, the standard deviation for all variables was around one full ordinate, indicating that the mean difference between participants was between the ordinate “4: tendentially true for Hazard Relation Diagrams” and the ordinate “2: tendentially true for activity diagrams”. Yet, results also show that participants report a slight preference for activity diagrams with regard to the variables “Perceived Usefulness” (DV14.2), “Result Demonstrability” (DV16.2), “Meaning” (DV18.2), and “Presentation” (DV19.2), and reported slightly more confusion with Hazard Relation Diagrams (DV21.2). With regard to the overall variable for self-reported subjective confidence H4total.2, participants indicate a preference for activity diagrams and FHA result tables.

The fact that the means for each variable were consistently around the neutral ordinate with a fairly large standard deviation suggest that we must reject hypothesis H4 in Experiment 2: there is no difference in subjective confidence within participants when using Hazard Relation Diagrams or when using conventional activity diagrams with FHA result tables to review conceptual mitigations. Table 19 summarizes the means for each variable. In Fig. 9, the frequencies of responses for each ordinate and each retained quality focus is shown. Darker bars indicate stronger preference for Hazard Relation Diagrams.

Table 19. Means and Standard Deviations for Hypothesis H4 in Experiment 2.

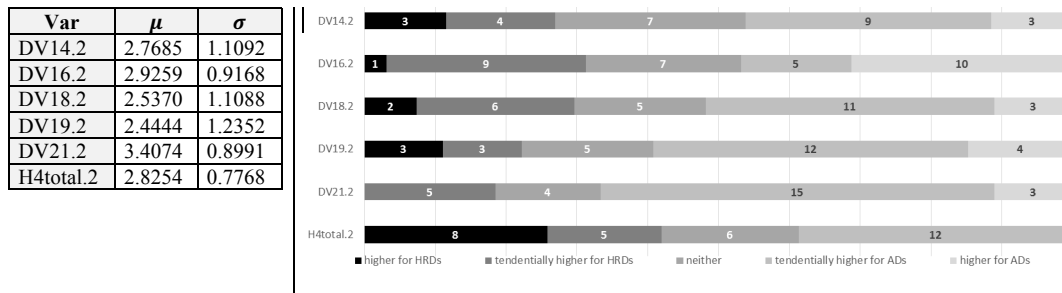


Fig. 9. Distributions over all Ordinates of the Retained Quality Foci for Hypothesis H4 Experiment 2.

## 7.5. Discussion of the Findings

In this section, we discuss the impact of Hazard Relation Diagrams on reviews at large. For this purpose, we compare the results from the two experiments to each another with regard to each hypothesis. To foster comparability of the overall results, Table 20 depicts the result of each experiment for each hypothesis. Considering the two-tailed nature of the hypotheses, Table 20 indicates whether Hazard Relation Diagrams (HRD)

or conventional activity diagrams with FHA result tables (AD) show superior performance. Conclusive evidence in favor of Hazard Relation Diagrams is depicted in dark grey.

Table 20. Summary of Hazard Relation Diagram Results in each Experiment wrt. the Hypotheses.

Hypothesis		Experiment 1	Experiment 2
		Hypothesis Accepted	
		Superior Performance of HRD/AD	
H1	There is a difference in objectivity between treatment condition and control condition when validating hazard-mitigating requirements.	yes	yes
		HRD	HRD
H2	There is a difference in effectiveness between treatment condition and control condition when validating hazard-mitigating requirements.	no	yes
		AD	HRD
H3	There is a difference in efficiency between treatment condition and control condition when validating hazard-mitigating requirements.	no	no
		HRD	HRD
H4	There is a difference in self-reported confidence between treatment condition and control condition when validating hazard-mitigating requirements.	yes	(no hypothesis testing)
		HRD	AD

**H1: Objectivity.** In both experiments, the treatment condition provided considerably more rationales mentioning contextual information about the hazard as opposed to diagram properties. These results are significant for all combined variables. In particular, the modeling concepts “mitigation partition” and “trigger condition” are the most pertinent to this effect, while there does not seem to be a significant effect on “safety goal” or “diagram syntax.” Yet, these results do not indicate whether or not this effect is caused by the combination of modeling elements in the Hazard Relation Diagrams or due to the mitigation partitions and trigger conditions alone. Further investigation is needed to ascertain whether this effect also holds when safety goals are not part of the Hazard Relation Diagram, i.e. whether or not surrounding the changed portions of the diagram with a dashed line is sufficient as long as trigger conditions are present. In addition, further investigation is required about the impact on reviewer background on the results. Albeit experimental populations were roughly equivalent with regard to their levels of experience (see Section 6.2), repetition is necessary with, for example, industry representatives. Nevertheless, we can confidently conclude that Hazard Relation Diagrams have a positive influence on reviews and improve the objectivity when reviewing hazard-mitigating requirements. This is supported by the differences in experimental design, as these results are true for between-subjects and for within-subjects designs.

**H2: Effectiveness.** Experiment 1 revealed negligibly better effectiveness of the control group, as the participants scored more correct answers while making fewer mistakes than the participants of the treatment group did. However, these differences

were insignificant and the measured effect was small in all cases. In Experiment 2, the results of Experiment 1 were reversed: when using Hazard Relation Diagrams for reviews, participants scored significantly more correct answers than in the control condition. These results can be explained by the experimental design: while in Experiment 1, participants were allowed to complete the experiment at home in an uncontrolled environment, we controlled the experimental conditions more strictly in Experiment 2 by inviting participants to complete the experiment in a controlled environment during a class meeting. It is thus possible that in Experiment 1, participants did not dedicate their full attention to the experiment, leading to more or less random adequacy judgments and hence an absence of significant differences. By contrast in Experiment 2, a higher degree of concentration lead to significant differences in favor of Hazard Relation Diagrams. Considering the lack of significance in all variables, we can assume that the mode of experimentation caused participants in Experiment 1 to be distracted, impacting their focus and motivation, and lead to equal effectiveness between Hazard Relation Diagrams and activity diagrams. In light of this possibility, the significant results in Experiment 2 represent more compelling evidence in favor of Hazard Relation Diagrams. Moreover, effect sizes were high or very high for all variables in Experiment 2, increasing the confidence in the results of Experiment 2.

**H3: Efficiency.** Results regarding efficiency in Experiment 1 indicate that all response times were slightly faster in the treatment group. Similar results were obtained from Experiment 2, where participants were faster in the treatment condition when answering correctly, but slower when answering incorrectly. However, in neither experiment, there was a significant difference between the two conditions. We therefore assume that Hazard Relation Diagrams have no significant negative impact and do not influence the efficiency during reviews. Nevertheless, in both experiments, a marginal improvement in efficiency when using Hazard Relation Diagrams became apparent. If this effect is increasing or decreasing when using experienced reviewers (e.g., industry practitioners) is subject of further studies.

**H4: Self-Reported Confidence.** The results of the post-hoc questionnaire from Experiment 1 show that the treatment group rated their confidence during reviews consistently higher than the control group, revealing an overall significant effect with medium effect sizes. In particular, participants in the treatment condition rated their ability to justify adequacy judgements and understand the modeling concepts contained in the diagrammatic representation significantly higher than in the control condition.

Moreover, they self-reported significantly less confusion about the review task than control participants. In Experiment 2, participants reported a slight preference for conventional activity diagrams. These results are surprising, as participants' lower subjective confidence using Hazard Relation Diagrams for reviews contrasts with their considerably higher degree of objectivity (see Section 7.1), higher effectiveness (see Section 7.2) and slightly improved efficiency (see Section 7.3). The topic of activity diagrams was taught in the course from which the participants were recruited just before Experiment 2 was conducted. It might thus well be that the participants preferred more familiar diagrammatic representations. In Experiment 1, activity diagrams were a topic earlier in the semester. Neither the experimental investigation nor the participants' self-reported levels of experience (which were comparable between both experiments, see Section 6.2) do not conclusively support such conjecture. However, Experiment 1 shows some positive influence on participants' self-reported subjective confidence for validation at large when using Hazard Relation Diagrams to validate the adequacy of conceptual mitigations.

## 8. Threats to Validity

Albeit utmost care was taken to design the experiments at hand, several threats to validity remain. These are discussed in the order suggested by [51] in the following.

**Internal Validity.** One critical issue for the study at hand is the suitability of the between-subjects design and the experimental procedure. To gain confidence in the design and experimental set-up in general, we conducted a pilot test (reported in [21]) that led to several improvements of the study. Another issue, which may impair internal validity, is the conveniently sampling of the participants from undergraduate university courses and their limited experience in the subject matter. "Conveniently sampled" means that students were recruited from courses where the authors had convenient access to and where students could be motivated to participate. Since the use of undergraduate students is somewhat controversial [64], we adhered to the best practices in this matter [64], [65], [66] to reduce this threat, while at the same time served to motivate students. This included integrating the experiments as a learning opportunity into the courses from which the participants were selected, allowing for student feedback on the effectiveness of experiments to facilitate learning, allowing for student introspection about the post-experimental learning progress, and providing a detailed debriefing concerning the

purpose of the study, including sharing preliminary data with participants to foster learning.

Another issue that has impact on the internal validity of the empirical evaluation is the difference in the mode of experimentation used for Experiment 1 and Experiment 2. In Experiment 1, students completed the experiment at home, in an uncontrolled setting, while in Experiment 2, students participated in the experiment in a class meeting. In consequence, there is a possibility that participants in Experiment 1 were distracted, did not entirely focus on participation, and may have had a lower level of motivation than in Experiment 2. This could have impacted their performance and skewed their results. We accounted for this possibility during data preparation (see Section 5.5) and strictly excluded potentially skewed results. Furthermore, we took a very conservative approach to present and discuss the results in Section 7 in light of differences in mean, significance, and statistical power. Nevertheless, we acknowledge a remaining influence of the uncontrolled factors in Experiment 1 as a threat to internal validity. In fact, this issue was one of the key reasons to design and conduct Experiment 2.

Language ability may also have had a influence on the results of Experiment 1, as the experimental stimuli were in English, but participants were mainly German. However, we are confident that this impact is negligible due to the generally high proficiency of German university students in written and spoken English. Nevertheless, we combatted this threat by allowing for questions pertaining to technical terminology used in the experimental stimuli during the briefing session.

***Construct Validity.*** The experimental material must be suitable to measure the desired effect. Errors or bias within the material must not influence the experimental results. In our experiments, we placed particular emphasis on the development of the experimental stimuli. To avoid bias, we specifically used a case example which is intuitively understandable whilst maintaining some degree of realism. We used the findings from the pilot test to improve the experimental material and the pre-experimental briefing. Participants were trained not only in the case example and in the intended review procedure, but also in safety engineering and in the syntax and semantics of Hazard Relation Diagrams, activity diagrams, and FHA. The same training material was used in both experiments. Moreover, through industry cooperation and pilot testing, we iteratively improved the experimental material, metrics, and questionnaires until they fit for experimentation in both conditions.

In addition, the mode of testing may have impaired the experimental results. While in Experiment 2, participation took place during a class meeting, scheduling conflicts prevented controlled participation conditions in Experiment 1. To combat this issue, we took great care in identifying irregular responses, by rigorously excluding incomplete data sets, data sets with pattern answers, or data sets showing skewed response times, etc. This resulted in the exclusion of several data sets as described in Section 5.5.

**Conclusion Validity.** Confirmation bias and low statistical power may impair conclusions drawn from results. To avoid confirmation bias, we have only accepted hypotheses based on a strict significance level of 0.05. We have taken a conservative approach in discussing the results and aimed to illustrate tendencies that are in favor of or against Hazard Relation Diagrams. We have conducted post-hoc power analyses on all T-Test results and have reported on effect sizes and power in order to increase credibility in the results of our analysis. Due to the possible impact on internal and construct validity of Experiment 1, we furthermore conducted Experiment 2 in order to minimize the impact on the “take-home” nature of Experiment 1 on the experimental results. We have conservatively analyzed the impact of Hazard Relation Diagrams in both experiments, and have carefully drawn conclusions. We have highlighted confirmatory, but also contrasting results, thereby increasing confidence in the findings.

In addition, the results in both experiments may have been impacted by the participants’ levels of experience. We have reported on a detailed comparison of the differences in experience (Section 6.2) and have discussed the possible impact of experience on the findings (Section 7.5). We have found little differences between the experimental populations, which presumably had little impact on the results.

Considering the conservative approach and strict criteria in accepting evidence in favor of Hazard Relation Diagrams, and considering rigorous experimental design and discussion of findings, we have confidence in our results. Nevertheless, the results from both experiments warrant further investigation into the impact of Hazard Relation Diagrams, particularly with regard to effectiveness, subjective confidence, and more experienced participants (e.g., industry practitioners).

**External Validity.** Another relevant concern is how the results generalize to circumstances beyond the scope of the study at hand, i.e. if the effects found in this study hold for reviews of any hazards in any project. To ensure external validity, we used an industrial case example developed by industry partners as the basis for the experimental material. The experimental material was rigorously quality assured. In addition, industrial

practice of validating requirements by means of reviews and inspections was discussed intensively with industrial partners. The development of the experimental procedures was guided by industrial practice. The industrial case example was reduced in complexity to fit this study's scope, generalizability with real-world examples may have been lost. The experimental material thus reflects a *realistic* example, but does not represent a *real-world* example.

***Student Participant Representativeness.*** The typical arguments regarding the employment of students in experimental evaluations can be brought forward [64], i.e. mainly that students may not be representative for industry experts and that the low experience level of undergraduate students may have an impact on the experimental results. Since the experimental design in both experiments balanced participants across groups and conditions to similar experience levels (see Section 6.2) and did not draw upon experience (with the exception of the material covered during pre-experimental training), we believe that the experimental designs in general alleviates this issue to some degree. Nevertheless, a repetition of these experiments with practitioners is desirable.

***Researcher Bias.*** The qualitative analysis regarding the objectivity hypotheses gives rise to the possibility of researcher bias during coding of the rationale statements. This is a serious issue that we took great care to avoid by conducting the qualitative analysis as objectively as possible: we have removed group membership information from the data set while coding rationale statements. This should reduce the influence of researcher bias on categorization. In addition, categorization was verified by three independent researchers. Yet, due to the nature of the quality assurance process of the qualitative data categorizations (see Section 5.5), we were unable to compute interrater reliability, which is an acknowledgeable threat to validity. Because of this, we have taken a very conservative approach to data preparation and analysis, adhered to a strict protocol reviewed by academic partners. Potentially biased results in both experiments have been rigorously reviewed, rectified, or discarded, where necessary. We are therefore confident that we took sufficient actions to minimize bias.

## 9. Related Work

As outlined in Section 1, in early phases of development, it is of particular importance to find objective means to ensure that all identified hazards are adequately mitigated. That means that:



1. Adequate functional hazard-mitigating requirements must be elicited for every hazard;
2. The adequacy of these hazard-mitigating requirements must be validated.

A plethora of different approaches to systematically elicit hazard-mitigating requirements exists, e.g. goal-oriented approaches (e.g., [38], [67]) scenario- and/or use case-based approaches (e.g., [68], [69]) or approaches that derive hazard-mitigating requirements from safety analysis results (e.g., [70], [71], [72], [73], [74]). Goal-oriented approaches (e.g., [38]) support the systematic derivation of resolutions (i.e. hazard-mitigating requirements in the sense of Definition 5) for adverse operational conditions (i.e. failures in [67] or hazard obstructions in [38]). Especially in the KAOS approach [38], these resolutions are the result of systematic refinements. In principle, safety goals are refined into hazard-mitigating requirements which are provably correct. However, some effort must be undertaken to assess their functional adequacy (see, e.g., [75]).

Scenario-based approaches find possible safety-critical deviations in the intended interaction between external users and the system (see, e.g., [68]) and guide developers in finding possible conceptual mitigations. Similarly, misuse cases (e.g., [76]), which have been conceived as a means to identify and mitigate security threats, have successfully been applied to safety-critical systems in order to identify unsafe interactions between the system and its context and in order to find candidate interactions to resolve them (see [77] for a comprehensive overview on the application of security engineering techniques on system safety). A number of approaches have been proposed to elicit hazard-mitigating requirements from safety analysis techniques (like FTA, see, e.g., [71], [72], or FMEA, see, e.g., [70], [73]). Most of these approaches, however, such as [70] and [71], become applicable in late phases of development (i.e. when part of the system design and/or implementation is already present, see [4]).

Some techniques focus on early phases of development. Most notably, the STPA approach in [33] provides process guidance for the identification of hazard controlling functions and fosters adequacy validation of such conceptual mitigations by investigating how such a hazard can occur (i.e. what are the trigger conditions in the sense of Definition 2). In [74], hazard-mitigating requirements are developed based on a combination of FHA and FTA as well as data and control flow analyses. The approach also aims to foster requirements verification in so far as to verify that the system design satisfies safety-relevant timing invariants.

In a state of practice report from 1994, Flynn and Warhurst [12] have indicated that unstructured reviews are the predominant technique to conduct validation. Among other things, the authors point out that lack of available reference information and inherent subjectivity is detrimental to review objectivity by practitioners. Although more recent investigations into the state of practice are desirable, it becomes obvious that these issues prevail to this day: for example, more recent empirical findings reported by Shull et al. [78] and Boehm and Basili [79] also indicate that peer reviews are about 60% effective in identifying defects in engineering artifacts. A number of techniques have been proposed to improve reviews, e.g., through reading techniques like perspective-based reading (see, e.g., [80], [81]), or value-based [82] or inspection-like techniques (e.g., [83], [84]), which can improve effectiveness and efficiency of validation considerably [85]. Yet, none of these techniques is specific to safety requirements. In [86], [87] a framework is suggested which aids safety engineers in validating safety requirements of composite systems (i.e. systems of systems) by proposing a number of metrics that allow assessing the architecture of the composite system. Among other things, the framework considers the correctness (in the sense that there are no safety constraint violations), yet not adequacy. A concrete process model to conduct validation within this framework is given in [88]. The process model focuses what is considered “sufficiency of requirements,” i.e. to what degree the hazards have been mitigated.

The overwhelming majority of approaches in the discipline of safety engineering at large deal with formal quality assurance. These approaches are mostly concerned with identifying safety-critical defects in formal specifications (e.g., [71], [70], [89], [90], [91], [92], [93]). These approaches focus on analyzing timing constraints (e.g., [71], [90]), behavioral constraints (e.g., [91], [92], [93]), design invariants (e.g., [70]), or event-based failure propagation (e.g., [89]) and rarely distinguish, whether the techniques are applied to hazard-inducing requirements, hazard-mitigating requirements, or both.

While goal- and scenario-based elicitation techniques are powerful means to elicit mitigation strategies for every hazard identified, guidance on how to refine high-level goals into concrete hazard-mitigating requirements is rarely given. They thus do not foster adequacy of the conceptual mitigation or of the hazard-mitigating requirements defined to mitigate the hazards. Furthermore, safety analysis-driven approaches to elicit hazard-mitigating requirements require that at least some portion of the system to be designed or even implemented. Because of this, the requirements that can be identified using these approaches are more akin to technical constraints that become apparent during

later development states rather than system functionality during early development stages (cf. Section 1.1). Formal quality assurance approaches can provide objective evidence by checking the formal correctness of requirements (i.e. *verification* in the sense of [94]). However, they do not support the *validation* of the adequacy of hazard-mitigating requirements.

## 10. Conclusion and Outlook

Hazard Relation Diagrams are a diagrammatic representation intended to improve objectivity when validating if the defined hazard-mitigating requirements are adequate. In this paper, we discussed the need of Hazard Relation Diagrams, outlined their principles, ontology, and notation, and described their application in settings with non-trivial relationships between the identified hazards and their mitigations. Moreover, we defined a set of well-formedness rules to facilitate checks about the syntactical correctness of Hazard Relation Diagrams. Based on the well-formedness rules, we presented an approach to automatically generate Hazard Relation Diagrams. Furthermore, we have presented a detailed empirical investigation on the impact of Hazard Relation Diagrams on reviews. Specifically, the research questions of the empirical evaluation are:

- RQ1: What is the impact of Hazard Relation Diagrams on review objectivity?
- RQ2: Does the impact of Hazard Relation Diagrams on review objectivity come at a cost of decreased review effectiveness and efficiency?
- RQ3: Do Hazard Relation Diagrams impact the reviewers' confidence in their adequacy judgment?

Regarding RQ1, experimental results show an effect on validation objectivity. Adequacy judgments in reviews using Hazard Relation Diagrams are more often based on contextual information about the hazard than in reviews using conventional activity diagrams and hazard analysis result tables. This shows a clear, positive influence of Hazard Relation Diagrams on objectivity during validation and shows that using Hazard Relation Diagrams to validate the adequacy of hazard-mitigating requirements leads to an improvement.

In addition, our empirical results pertaining to RQ2 indicate that validation may be more effective in validation settings and show a positive impact on the time needed for the validation. Yet, this effect was not noticeable in the between-subjects setting; an outcome that was likely impacted by differences in the experimentation mode.

Nevertheless, we can conclude that an improvement in review objectivity does not come at the cost of a decreased review efficiency or effectivity.

The effect on subjective confidence for RQ3 was marginal. Especially the empirical findings for the between-subject settings, show that participants perceive Hazard Relation Diagrams as more useful when validating hazard-mitigating requirements than conventional activity diagrams and FHA result tables. This supports the findings regarding RQ1, where Hazard Relation Diagrams improve the objectivity of adequacy judgements of hazard-mitigating requirements.

Ongoing work is concerned with additional empirical investigations and repetition of our studies to investigate the impact of Hazard Relation Diagrams on validation activities in more detail. Specifically, repetition studies with industrial professionals are desirable to investigate generalizability of the results presented in this paper.

The idea of unifying information to focus attention during validation is a principle that can be applied to other areas of software engineering as well. Due to the closely related nature of the disciplines, Hazard Relation Diagrams could be particularly beneficial in security engineering and reliability engineering – a promising avenue for future work.

## Acknowledgements

This research was partly funded by the German Federal Ministry of Education and Research under grant number 01IS12005C. We thank Arnaud Boyer (Airbus Defence and Space) for his consultation regarding FHA. We thank Dr. Frank Houdek (Daimler AG) as well as Peter Heidl and Jens Höfflinger (Robert Bosch GmbH) for their consultation on the Adaptive Cruise Control system. We thank Dr. Kai Petersen of Blekinge Institute of Technology as well as Marian Daun and André Heuer (University of Duisburg-Essen) for their feedback on the study design. We also thank all our participants in the experiments.

## References

- [1] Leveson NG (2011) Engineering a safer world: Systems thinking applied to safety. Engineering systems. MIT Press, Cambridge, Mass
- [2] SAE International (1996) ARP4761, Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment. <http://standards.sae.org/arp4761/> Accessed 7 January 2016
- [3] International Organization for Standardization (2011) ISO26262, Road Vehicles – Functional Safety. [http://www.iso.org/iso/catalogue\\_detail?csnumber=43464](http://www.iso.org/iso/catalogue_detail?csnumber=43464) Accessed 7 January 2016
- [4] Ericson CA (2005) Hazard Analysis Techniques for System Safety. John Wiley & Sons, Inc, Hoboken, NJ, USA
- [5] Leveson NG (1995) Safeware: System safety and computers: Addison-Wesley, Reading, Mass
- [6] IEEE Standards Board: IEEE Std. 610.12: IEEE Standard Glossary of Software Engineering Terminology, 1990.

- [7] Firesmith D (2004) Engineering Safety Requirements, Safety Constraints, and Safety-Critical Requirements. *J Object Technology* 3(3): 27–42. doi: 10.5381/jot.2004.3.3.c3
- [8] Bishop P, Bloomfield R, Guerra S (2004) The future of goal-based assurance cases. In: *Proc. Workshop on Assurance Cases*, pp 390–395
- [9] Wilson SP, Kelly TP, McDermid JA (1997) Safety Case Development: Current Practice, Future Prospects. In: Shaw R (ed) *Safety and Reliability of Software Based Systems*. Springer London, London, pp 135–156
- [10] Leveson N (2011) The Use of Safety Cases in Certification and Regulation. *Journal of System Safety* 47(6). <http://goo.gl/j9NW5Y>, accessed July 13, 2016.
- [11] Hatcliff J, Wassyng A, Kelly T et al. (2014) Certifiably safe software-dependent systems: challenges and directions. In: *Proc. Future Softw. Eng.*, pp 182–200
- [12] Flynn DJ, Warhurst R (1994) An empirical study of the validation process within requirements determination. *Inform Syst J* 4(3): 185–212. doi: 10.1111/j.1365-2575.1994.tb00051.x
- [13] Lisagor O, Sun L, Kelly T (2010) The illusion of method: challenges of model-based safety assessment. In: *28th International System Safety Conference (ISSC)*
- [14] Sun L (2012) Establishing confidence in safety assessment evidence. Dissertation, University of York
- [15] Glinz M (2000) Improving the Quality of Requirements with Scenarios. In: *Proc 2nd World Cong. Softw. Qual.*, pp 55–60
- [16] Gacitua R, Ma L, Nuseibeh B, Piwek P, de Roeck AN, Rouncefield M, Sawyer P, Willia A, Yang H (2009) Making Tacit Requirements Explicit. In: *2<sup>nd</sup> Intl. Workshop on Managing Requirements Knowledge (MARK)*, pp 40–44
- [17] Glinz M, Fricker SA (2015) On shared understanding in software engineering: An essay. *Comput Sci Res Dev* 30(3-4): 363–376. doi: 10.1007/s00450-014-0256-x
- [18] Mao J, Chen L (2012) Runtime Monitoring for Cyber-physical Systems: A Case Study of Cooperative Adaptive Cruise Control. In: *Proc. 2nd Int. Conf. Intell. Sys. Des. & Eng. Appl.*, pp 509–515.
- [19] Caramihai SI, Dumitrache I (2013) Urban Traffic Monitoring and Control as a Cyber-Physical System Approach. In: Dumitrache L (ed) *Advances in Intelligent Control Systems and Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 355–366
- [20] Lempia DL, Miller S (2009) Requirements engineering management findings report. Technical Report DOT/FAA/AR-08/34, Federal Aviation Administration
- [21] Tenbergen B, Weyer T, Pohl K (2015) Supporting the Validation of Adequacy in Requirements-Based Hazard Mitigations. In: *Requirements Engineering: Foundation for Software Quality, LNCS vol 9013*. Springer International Publishing, pp 17–32
- [22] Heimdahl MP (2007) Safety and Software Intensive Systems: Challenges Old and New. In: *Future of Software Engineering*, pp 137–152.
- [23] Stoneburner G (2006) Toward a Unified Security-Safety Model. *Computer* 39(8): 96–97. doi: 10.1109/MC.2006.283
- [24] Kelly T (2007) Reviewing assurance arguments-a step-by-step approach. In: *Workshop on Assurance Cases for Security-The Metrics Challenge, Dependable Systems and Networks (DSN)*
- [25] Johnson CW, Holloway CM (2006) Questioning the role of requirements engineering in the causes of safety-critical software failures. In: *IET Intl. Conf. Sys. Safety*, pp 352–361
- [26] Lempia DL, Miller S (2009) Requirements Engineering Management Handbook. Technical Report, DOT/FAA/AR-08/32, Federal Aviation Administration
- [27] Chung L, Nixon BA, Yu E et al. (2000) Non-Functional Requirements in Software Engineering. *International Series in Software Engineering*, vol 5. Springer, Boston, Mass.
- [28] Conway, M (1967) How Do Committees Invent? *Datamation* 14(4), pp. 28–31.
- [29] Finkelstein, A.; Kramer, J.; Nuseibeh, B.; Finkelstein, L.; Goedicke, M. (1992) Viewpoints: A Framework for Integrating Multiple Perspectives in System Development. *Intl. J Softw. Eng. & Knowl. Eng.* 2(1), pp. 31–58.
- [30] Cooper K, DePrenger M, Mattern S, McKinley A, Pajouhesh A, Shampine D (2010) Joint Software Systems Safety Engineering Handbook. United States Department of Defense, Version 1.0, 2010. <http://goo.gl/er2mWY>, accessed July 13, 2016.
- [31] Boehm BW (1981) *Software engineering economics*. Prentice-Hall, Englewood Cliffs, NJ
- [32] Hawkins R, Kelly T, Knight J, Graydon P (2011) A New Approach to creating Clear Safety Arguments. In: Dale C, Anderson T (eds) *Advances in Systems Safety*. Springer London, London, pp 3–23
- [33] Leveson NG (2004) A systems-theoretic approach to safety in software-intensive systems. *IEEE Trans.Dependable and Secure Comput.* 1(1): 66–86. doi: 10.1109/TDSC.2004.1
- [34] Wang J, Yang JB (2001) A subjective safety and cost based decision model for assessing safety requirements specifications. *Int. J. Rel. Qual. Saf. Eng.* 08(01): 35–57. doi: 10.1142/S0218539301000335.
- [35] Moody DL (2009) The “Physics” of Notation: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. *IEEE Trans. Soft. Eng.* 35(6), pp 756–779
- [36] Störrle H (2004) Semantics of Control-Flow in UML 2.0 Activities. In: *IEEE Symp. on Visual Languages and Human Centric Computing*, pp 235–242
- [37] Object Management Group (2015) *OMG Unified Modeling Language, Version 2.5*. OMG Document Number formal/2015-03-01. <http://goo.gl/7cQyPv>, accessed July 13, 2016.
- [38] van Lamsweerde A (2009) *Requirements engineering: From system goals to UML models to software specifications*. Wiley, Chichester
- [39] Yu E (1997) Towards modelling and reasoning support for early-phase requirements engineering. In: *Intl. Symp. RE*, pp 226–235
- [40] Giorgini P, Mylopoulos J, Sebastiani R (2005) Goal-oriented requirements analysis and reasoning in the Tropos methodology. *Engineering Applications of Artificial Intelligence* 18(2): 159–171. doi: 10.1016/j.engappai.2004.11.017
- [41] Kelly T, Weaver R (2004) The goal structuring notation-a safety argument notation. In: *Proc. of the Workshop on Assurance Cases of Dependable Systems and Networks*
- [42] Heim I, Kratzer, A (1998) *Semantics in Generative Grammar*. Wiley, Chichester
- [43] Finkelstein A, Kramer J, Nuseibeh B et al. (1992) Viewpoints: A Framework for integrating Multiple Perspectives in System Development. *Int. J. Soft. Eng. Knowl. Eng.* 02(1): 31–58. doi: 10.1142/S0218194092000038

- [44] Conway ME (1968) How Do Committees Invent? *Datamation*: 28-31
- [45] Reif K (2010) *Fahrstabilisierungssysteme und Fahrerassistenzsysteme*. Vieweg+Teubner, Wiesbaden
- [46] Awodey S (2011) From Sets to Types, to Categories, to Sets. In: Sommaruga G (Ed) *Foundational Theories of Classical and Constructive Mathematics*, Springer, pp. 113-125.
- [47] Object Management Group (2011) QVT: Meta Object Facility (MOF) 2.0 Query/View/Transformation, v1.1.
- [48] QVT Operational Eclipse Plugin, v3.5.0: <https://goo.gl/SglK1F> Accessed 7 January 2016
- [49] Eclipse Modeling Tools, Luna Package Distribution: <https://goo.gl/qo9Sf5> Accessed 7 January 2016
- [50] Jedlitschka A, Ciolkowski M, Pfahl D (2008) Reporting Experiments in Software Engineering. In: Shull F, Singer J, Sjøberg DIK (eds) *Guide to Advanced Empirical Software Engineering*. Springer, London, pp 201–228
- [51] Wohlin C, Runeson P, Höst M et al. (2012) *Experimentation in software engineering*. Springer, Berlin
- [52] Tabachnick BG, Fidell LS (2010) *Using multivariate statistics*, 5th ed. Pearson/Allyn and Bacon, Boston, Mass.
- [53] SoSci Survey Website: <https://www.sosicisurvey.de> Accessed 7 January 2016
- [54] van Solingen R, Berghout E (1999) *The goal/question/metric method: A practical guide for quality improvement of software development*. The McGraw-Hill Companies, London
- [55] Venkatesh V, Bala H (2008) Technology Acceptance Model 3 and a Research Agenda on Interventions. *Decision Sciences* 39(2): 273–315. doi: 10.1111/j.1540-5915.2008.00192.x
- [56] Goodhue DL (1998) Development and Measurement Validity of a Task-Technology Fit Instrument for User Evaluations of Information System. *Decision Sciences* 29(1): 105–138. doi: 10.1111/j.1540-5915.1998.tb01346.x
- [57] Osgood CE, Suci G., Tannenbaum P. (1957) *The measurement of meaning*. University of Illinois Press, Urbana, IL
- [58] Verhagen T., Hooff B. van den Meents S. (2015) Toward a Better Use of the Semantic Differential in IS Research: An Integrative Framework of Suggested Action. *J of the Association for Information Systems* 16(2): Article 1
- [59] Corbin JM, Strauss AL (2008) *Basics of qualitative research: Techniques and procedures for developing grounded theory*, 3. ed. Sage Publ, Los Angeles, Calif..
- [60] Cronbach LJ (1951) Coefficient alpha and the internal structure of tests. *Psychometrika* 16(3): 297–334. doi: 10.1007/BF02310555.
- [61] Student (1908) The Probable Error of a Mean. *Biometrika* 6(1): 1–25. doi: 10.2307/2331554
- [62] David HA, Gunnink JL (1997). The Paired t Test Under Artificial Pairing. *The American Statistician* 51(1): 9–12. doi:10.2307/2684684. JSTOR 2684684
- [63] Cohen J (1992) A power primer. *Psychological Bulletin* 112(1): 155–159. doi: 10.1037/0033-2909.112.1.155
- [64] Carver J, Jaccheri L, Morasca S, Shull F (2003) Issues in using students in empirical studies in software engineering education. In: *Proc. 9<sup>th</sup> Intl. Software Metrics Symp.*, pp 239–249
- [65] Hart C, Mulhall P, Berry A, Loughran J, Gunstone R (2000) What is the purpose of this experiment? Or can students learn something from doing experiments? *J. Res. Sci. Teach.* 37(7): 655–675. doi: 10.1002/1098-2736(200009)37:7<655::AID-TEA3>3.0.CO;2-E
- [66] Carver JC, Nagappan N, Page A (2008) The Impact of Educational Background on the Effectiveness of Requirements Inspections: An Empirical Study. *IEEE Trans. Software Eng.* 34(6): 800–812. doi: 10.1109/TSE.2008.49
- [67] Navarro E, Sanchez P, Letelier P, Pastor JA, Ramos I (2006) A goal-oriented approach for safety requirements specification. In: *13<sup>th</sup> Ann. IEEE Intl. Symp. and Workshop on Eng. Comp. Based Sys.*, pp 319-326
- [68] Allenby K, Kelly T (2001) Deriving safety requirements using scenarios. In: *5<sup>th</sup> IEEE Intl. Symp. RE*, pp 228–235
- [69] Chen D, Johansson R, Lönn H, Papadopoulos Y, Sandberg, A, Törner F, Törngren M (2008) Modelling Support for Design of Safety-Critical Automotive Embedded Systems. In: *Proc. 27th Intl. Conf. Comp. Safety, Reliability and Security*, pp 72–85
- [70] Guillerme R, Demmou H, Sadou N (2011) Combining FMECA and fault trees for declining safety requirements of complex systems. In: *Advances in Safety, Reliability and Risk Management*. CRC Press, pp 1287–1293
- [71] Hansen KM, Ravn AP, Stavridou V (1998) From safety analysis to software requirements. *IEEE Trans. Software Eng.* 24(7): 573–584. doi: 10.1109/32.708570
- [72] Tsuchiya T, Terada H, Kusumoto S, Kikuno T Eun Mi Kim (1997) Derivation of safety requirements for safety analysis of object-oriented design documents. In: *Proc. of the 21st Ann. Int. Comp. Softw. and Applications Conf*, pp 252–255
- [73] Troubitsyna E (2008) Elicitation and Specification of Safety Requirements. In: *Proc. of the 3rd Int. Conf. on Systems*, pp 202–207
- [74] Xu X, Bao X, Lu M, Chang W (2011) A study and application on airborne software safety requirements elicitation. In: *Proc. of the 9th Int. Conf. on Reliability, Maintainability and Safety*, pp 710–716
- [75] van Lamsweerde A (2009) Reasoning About Alternative Requirements Options. In: *Conceptual Modeling: Foundations and Applications*. Springer, Heidelberg, pp 380–397
- [76] Sindre G (2007) A Look at Misuse Cases for Safety Concerns. In: *Proc. IFIP WG 8.1 Conf.*, pp 252–266
- [77] Raspotnig C, Opdahl A (2013) Comparing risk identification techniques for safety and security requirements. *Journal of Systems and Software* 86(4): 1124–1151. doi: 10.1016/j.jss.2012.12.002
- [78] Shull F, Basili V, Boehm B, Winsor Brown A, Costa P, Lindvall M, Port D, Rus I, Tesoriero R, Zelkowitz M (2002) What We Have Learned About Fighting Defects. In: *Proc. 8th Int. Symp. on Software Metrics*, pp 249–258
- [79] Boehm B, Basili VR (2001) Software Defect Reduction Top 10 List. *Computer* 34(1): 135–137. doi: 10.1109/2.962984.
- [80] Basili VR, Green S, Laitenberger O, Lanubile F, Shull F, Sorumgard S, Zelkowitz M (1996) The Empirical Investigation of Perspective-Based Reading. *Empirical Software Engineering* 1(2): 133–164. doi: 10.1007/BF00368702
- [81] Shull F, Rus I, Basili V (2000) How perspective-based reading can improve requirements inspections. *Computer* 33(7): 73–79. doi: 10.1109/2.869376
- [82] Li Q, Boehm B, Yang Y, Wang Q (2011) A value-based review process for prioritizing artifacts. In: *Proc. Int. Conf. Softw. Syst. Process*, pp 13–23
- [83] Aurum A, Petersson H, Wohlin C (2002) State-of-the-art: software inspections after 25 years. *Softw. Test. Verif. Reliab.* 12(3): 133–154. doi: 10.1002/stvr.243

- [84] Porter AA, Votta LG, Basili VR (1995) Comparing detection methods for software requirements inspections: a replicated experiment. *IEEE Trans. Software Eng.* 21(6): 563–575. doi: 10.1109/32.391380
- [85] Lee K, Boehm B (2005) Empirical results from an experiment on value-based review (VBR) processes. In: *Proc. Of Intl. Symp. Empirical Softw. Eng.*, pp 3–12
- [86] Cruickshank KJ, Michael JB, Man-Tak Shing (2009) A Validation Metrics Framework for safety-critical software-intensive Systems. In: *IEEE Int. Conf. System of Systems Engineering*, 2009, pp 1–8
- [87] Michael JB, Shing MT, Cruickshank KJ, Redmond PJ (2010) Hazard Analysis and Validation Metrics Framework for System of Systems Software Safety. *IEEE Systems Journal* 4(2): 186–197. doi: 10.1109/JSYST.2010.2050159.
- [88] Driskell SB, Murphy J, Michael JB, Man-Tak Shing (2010) Independent validation of software safety requirements for systems of systems. In: *Proc. 5th Int. Conf. on System of Systems Engineering*, pp 1–6
- [89] Belli F, Hollmann A, Nissanke N (2007) Modeling, Analysis and Testing of Safety Issues - An Event-Based Approach and Case Study. In: *Proc. 26th Int. Conf. Comp. Safety, Rel. Security*, pp 276–282
- [90] Bitsch F (2001) Safety Patterns — The Key to Formal Specification of Safety Requirements. In: *Proc. of the 20th Int. Conf. on Comp. Safety, Reliability and Security*, pp 176–189
- [91] Bharadwaj R, Heitmeyer CL (1999) Model Checking Complete Requirements Specifications Using Abstraction. *Automated Software Engineering* 6(1): 37–68. doi: 10.1023/A:1008697817793
- [92] Heitmeyer C, Kirby J, Labaw B et al. (1998) Using abstraction and model checking to detect safety violations in requirements specifications. *IEEE Trans. Software Eng.* 24(11): 927–948. doi: 10.1109/32.730543
- [93] Zafar S, Dromey RG (2005) Integrating safety and security requirements into design of an embedded system. In: *Proc. 12th Asia-Pacific Software Engineering Conf.*, pp 629–636
- [94] Robertson S, Robertson J (2013) *Mastering the requirements process: Getting requirements right*, 3rd ed. Addison-Wesley, Upper Saddle River, NJ