Ontological Foundation of Hazards and Risks in STAMP

Jana Ahmad [a,*], Bogdan Kostov [a] Andrej Lališ [b] and Petr Křemen [a]

[a] *Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic*
E-mails: jana.ahmad@fel.cvut.cz, bogdan.kostov@fel.cvut.cz, Petr.Křemen@fel.cvut.cz

[b] *Department of Air Transport, Faculty of Transportation Sciences, Czech Technical University in Prague, Czech Republic*
E-mail: lalisand@fd.cvut.cz

Abstract. In recent years, there has been a growing interest in smart data-driven safety management systems comparing to the traditional ones. The demand for such upgrade comes from the frequent changes in our daily life and technological innovation which introduce new causes and factors of accidents. However, the increasing amount and heterogeneity of safety-related data introduces a new demand for their proper knowledge management to use them for detecting safety-related problems and predicting them. In this paper, we discuss the ontological foundations of hazards and risks which are represented by such data. We consider their representation in safety systems, specifically in the domain of aviation safety using the STAMP model. As a result, we propose a STAMP hazard risk ontology that could help in analyzing accidents and modeling control loop failures according to the theory of STAMP. For evaluation, we tested our ontology on realistic examples in the aviation safety domain as a use-case.

Keywords: Aviation Safety, Hazards, Ontology, Risk, Safety Engineering, STAMP

1. Introduction

With the advent of modern civilization, there has been a growing interest in building safer systems. High-risk industries and academic initiatives have been pushing the boundaries of how to view safety and how to improve upon existing solutions. Different safety models and safety analysis methods were proposed throughout years [1]. In this regard, we live an era of systemic models of safety that attempt to take the system-level view when explaining the etiology of safety, i.e. avoiding explanation of causality only with respect to separate component failures. In result, these models account for phenomena such as emergence, complexity and component interaction accidents. This is especially important for safety in socio-technical systems, where the interplay of humans, machines and software matters [2].

The two most recent systemic causation models of safety are the System-Theoretic Accident Model and Processes (STAMP) [3] and the Functional Resonance Analysis Method (FRAM) [4]. Both models decompose systems into components. STAMP models components as feedback control loops which allow classifying objects into three main categories, namely controllers, sensors and actuators (key parts of a feedback control loop). On the other hand, FRAM models components as functions avoiding descriptions of objects by design.

Both STAMP and FRAM have been constantly validated by other research [5–12]. These efforts are typically oriented to ad-hoc analyses in a real-world setup. Also, some software prototypes supporting modeling with STAMP [13–15] and FRAM [16] have been proposed.

Even though both causation models are intelligible and clear when used in simple applications, this ceases to be true for real-industry applications, as indicated by the ad-hoc analyses mentioned, where their usage can

[*] Corresponding author. E-mail: jana.ahmad@fel.cvut.cz.

be complex and hard to manage. Furthermore, in order to create STAMP/FRAM models, one needs both extensive amount of data [17] and significant expertise in safety [18]. Thus, the practical usefulness of STAMP and FRAM in large-scale industrial setups is still an open issue [7].

One of the key obstacles of adopting these models in the industry is the lack of their formalization. Usage of the same term in different concepts so as different terms for the same concept are example manifestations of this limitation. In this paper, we address these issues by ontological analysis and formalization of STAMP in the context of hazards and risks. We consider also the System-Theoretic Process Analysis (STPA) [3] methodology based on the STAMP model, that is intended for the use case of hazards analysis.

Our contribution includes two ontology modules: the STAMP Hazard and Risk Ontology (SHRO) presented in Section 4 and the STAMP Control Loop Hazard Profile (SCLHP) presented in Section 6. SHRO describes the concepts Hazard and Risk as understood in traditional safety as well as in STAMP and it is aligned with a novel reference ontology – the Common Ontology of Value and Risk [19]. The SCLHP formalizes common hazards associated with control loops proposed by the STAMP model. Additionally, we validate the Common Ontology of Value and Risk with industry use-cases. We adopt the Systematic Approach for Building Ontology (SABiO) [20] to develop the proposed ontology modules. The ontologies designed in this paper can be found online[1].

To validate our approach and results, we take the perspective of the aviation industry and its safety management. This work has been done within a research project in tight cooperation with two Czech aviation industry companies – Prague Airport and Czech Airlines Technics which trialed STAMP and STPA in their operations. Direct involvement of the two companies helped in assessing the usability and practical applicability of the proposed solutions. Industry experts also directly participated in the research activities.

The remainder of this papers is organized as follows. In Section 2, we detail STAMP, ontology engineering methodology, Foundational ontology and the Common Ontology of Value and Risk on which our work is based. Section 3 describes the ontology purpose identification and requirements elicitation. Section 4 shows the developed STAMP Hazard and Risk Ontology (SHRO). Section 5 describes the probabilistic risk assessment. Section 6 models the STAMP hazards modules according to our reference ontology. Analyzing hazard ontology in term of foundational ontology is in section 7. Section 8 shows the ontology validation and section 9 adds more details on the related work. Finally, section 10 concludes the paper.

2. Background

This section provides fundamentals for our research. It deals both with safety and ontology engineering, provides definition of key concepts and industry example.

2.1. STAMP: Hazards and Risks

The concepts of Hazard and Risk serve successfully for a couple of decades (since the invention of HAZOP methodology [21]) the very core of industrial safety management and are often part of industry standards (e.g. in aviation see [22]). HAZOP was one of the first methodologies actively using the concept of Hazard for the purpose of safety management. In the methodology, hazards were considered as deviations from normal procedures and the provided guide words (e.g. *more*, *less*, *early*, *late* etc.) assisted analysts with their identification, based on standard procedures description. The concept of Risk was used to prioritize identified hazards, by the means of expressing the probability and severity of potential hazard consequences, i.e. by standard risk matrix.

The new theory of STAMP provides updated methodology for hazard identification, the STPA. The methodology is not completely new as it builds upon the cornerstones of HAZOP, adopting core of hazard and risk conceptualization. However, it provides for some additional steps and guidance how hazards can be identified and treated, in line with the perspective of feedback control theory [23]. In fact, STAMP claims the ability to identify more hazards than it is possible with HAZOP and similar industrial methodologies, with improved support for risk estimation. On the other hand, the shortcomings of STAMP mentioned in the previous section hold for STPA as well. Furthermore, current industrial safety management using hazards and risks as in HAZOP is close to its limits, as there are already indications of the inability to progress any further on the safety of current operations [2]. Therefore, it is desirable to solve the conceptual issues of

[1] http://onto.fel.cvut.cz/ontologies/stamp-hazard-profile

STAMP so as other systemic models to allow for the progress.

As already mentioned, STAMP is a safety causation model that sees the problem of safety as a feedback control problem. With respect to this, the theory of STAMP proposes generic control loop issues that can be mapped onto a specific system (particular network of control loops) and is used to derive hazards or support accident/incident investigation. The generic control loop with classification of control problems is depicted in Fig. 1.

To allow the mapping of generic control problems from Fig. 1, accurate system description is needed according to the feedback control theory. This implies drawing complete set of control loops of the system (or its part under consideration) with their relationships thus establishing specific control loop network, which is then aligned with the proposed classification. This way, the model separates data from their interpretation; instead of encouraging merely descriptive statistics of the classified data, STAMP suggests to consider the control loop network at the first place, when the analyst attempts to identify weak points of the evaluated system.

The theory is not completely new but builds upon the heritage of Rasmussen [24], Perrow [25] and other successful practices in safety (as the already mentioned HAZOP). This is clear from how the theory of STAMP defines hazard [3]:

Definition 1. *A system state or a set of conditions that, together with a particular set of worst-case environmental conditions, will lead to an accident (loss).*

As an example, the adopted definition of hazard in civil aviation (by International Civil Aviation Organization) is "*A condition or an object with the potential to cause or contribute to an aircraft incident or accident.*" [22] The theory of STAMP updates conventional definitions of hazard to include the system perspective (system state), but does not reinvent the concept. Similar situation regards risk, which is defined by STAMP [3] as:

Definition 2. *A function of the hazard level combined with (1) the likelihood of the hazard leading to an accident and (2) hazard exposure or duration.*

This definition conforms to what is usually regarded as risk in different industries and does not introduce new notions (concepts).

To demonstrate the meaning of the concepts *hazard* and *risk* and their relation, consider the following example:

Example 1. *A bird strike is a type of incident in which a bird collides with an aircraft. Let's consider a specific bird strike incident. The cause/factor of that incident is the presence of a flying bird during landing of an airplane. The damage of the aircraft requires minor repair of the airplane fuselage.*

Based on example 1, we can say that birds flying near a runway during landing/takeoff of an aircraft is a hazardous situation. This hazardous event increases the risk (probability) of the occurrence of a bird strike event. Note that both hazard and risk are not associated with a specific event. They represent empirical knowledge used to predict the probability and the level of loss given a specific situation arises. Finally, this knowledge is always extracted from concrete events (e.g. investigation of an incident analysis of a hypothetical incident). Risk is based on the loss (the incident itself) while hazards are based on factors of incidents.

STAMP theory supports multiple use cases (designing a system, operations, investigation and similar), each with some specific perspectives, but we will not focus on these as they fall outside the scope of this paper.

2.2. Ontology and Ontology Engineering Methodology

The term ontology (in other words the study of existence) originates in philosophy. In computer science, there are several definitions of what an ontology is. We adopt the definition found in [26] – "*An ontology is a formal, explicit specification of a shared conceptualization*".

Ontology engineering is a complex process. There are many methodologies found in the literature, e.g., the agile methodology RapidOWL [27] and Methontology [20]. In this work, we are using SABiO [28] which understands ontology engineering as a five-steps process:

1. Purpose Identification and Requirements Elicitation,
2. Ontology Capture and Formalization,
3. Design,
4. Implementation and
5. Testing.

The first two steps build a *domain-reference* ontology which captures key knowledge of the domain. The next two steps focus on the design and implementation of

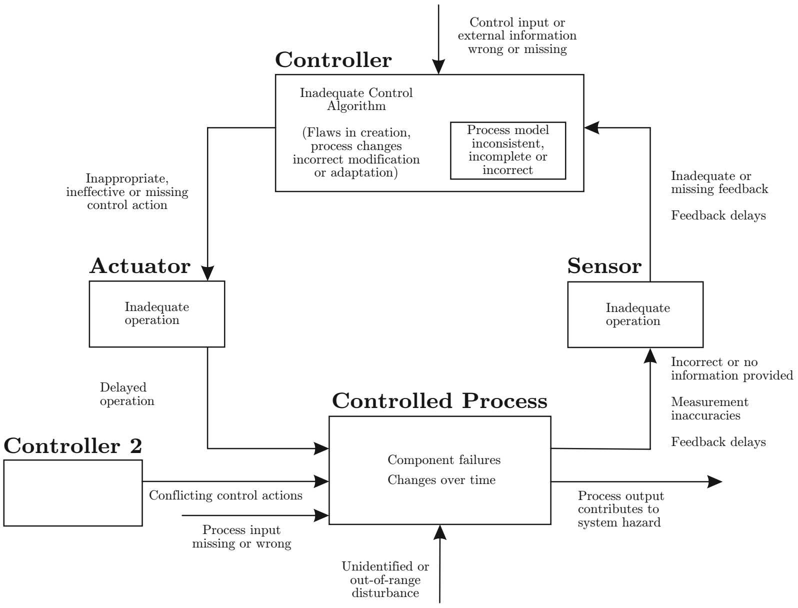
Control input or external information wrong or missing
Controller
Inadequate Control Algorithm
(Flaws in creation, process changes incorrect modification or adaptation)
Process model inconsistent, incomplete or incorrect
Inappropriate, ineffective or missing control action
Inadequate or missing feedback
Feedback delays
Actuator
Inadequate operation
Sensor
Inadequate operation
Incorrect or no information provided
Measurement inaccuracies
Feedback delays
Delayed operation
Controlled Process
Component failures
Changes over time
Controller 2
Conflicting control actions
Process input missing or wrong
Process output contributes to system hazard
Unidentified or out-of-range disturbance

Fig. 1. Generic control loop with issues classification as per STAMP (adapted from [3])

a *domain-operational* ontology into a formal machine-readable representation of the domain-reference ontology developed in the first two steps. The domain-operational ontology is designed to be used in software solutions. Finally, the last step evaluates the ontology w.r.t. to functional requirements defined in the first step or throughout the engineering process.

Furthermore, SABiO specifies five support activities which are parallel to the process described above. A short description of the these support activities is shown below:

Knowledge Acquisition in terms of interviews with domain experts, literature analysis,
Documentation of the engineering process, and
Configuration Management control of the artifacts such as source code produced by the individual phases, e.g. least change control and versioning,
Evaluation of intermediary artifacts
Reuse of existing ontologies/conceptualizations,

More information about SABiO methodology can be found in [20].

The engineering efforts were achieved by a team consisting of two ontology engineers, two domain experts and two potential ontology users. The ontology engineers and the domain experts have experience [29, 30] with building ontologies grounded in the Unified Foundational Ontology (UFO) [31]. Ontology users are represented by safety management departments of the two commercial partners participating in the research, i.e. Prague Airport and Czech Airline Technics.

The following subsections describe the foundational and reference ontologies used in this work.

2.3. Ontological Foundations

This section details the ontology engineering used in this work, i.e., the unified foundational (UFO) ontology so as the reuse of the Common Ontology of Value and Risk.

2.3.1. Unified Foundational Ontology (UFO)

UFO is a top-level foundational ontology that has been developed based on a number of theories from Formal Ontology, Philosophical Logic, Philosophy of Language, Linguistics and Cognitive Psychology [32]. Its main concepts fundamental for this work are sketched in the UML class diagram in Fig. 2. UFO describes endurants that are static objects (UFO-A) [31], perdurants/events (UFO-B) [33] and social agents (UFO-C) built on the top of UFO-A and UFO-B [34]. UFO splits entities into endurants and perdurants which are both individuals, i.e. entities that exist in reality and possess an identity that is unique

($\mathsf{Endurant} \sqsubseteq \mathsf{Individual}$) ($\mathsf{Perdurant} \sqsubseteq \mathsf{Individual}$)[2]. Endurants can be observed as complete concepts in a given time snapshot and they can be any object (e.g. an agent, aircraft) ($\mathsf{Object} \sqsubseteq \mathsf{Endurant}$), or its tropes or moments (e.g. speed, location, colors, etc.) ($\mathsf{Moment} \sqsubseteq \mathsf{Endurant}$), that exist as long as an object they inhere in exists ($\mathsf{Moment} \sqsubseteq (= 1\ \mathsf{inheresIn}{\cdot}\mathsf{Object})$) and situations ($\mathsf{Situation} \sqsubseteq \mathsf{Perdurant}$).

Perdurants only partially exist in a given time snapshot. They involve events ($\mathsf{Event} \sqsubseteq \mathsf{Perdurant}$) and object snapshots ($\mathsf{ObjectSnapshot} \sqsubseteq \mathsf{Perdurant}$).

Events can be either atomic or complex ($\mathsf{Event} \sqsubseteq (\mathsf{AtomicEvent} \sqcup \mathsf{ComplexEvent})$), they occur in time and have participants ($\mathsf{Event} \sqsubseteq (\geq 1\ \mathsf{hasParticipant} \cdot \mathsf{Object})$) and complex events have parts ($\exists\ \mathsf{hasEventPart} \cdot \top \sqsubseteq \mathsf{ComplexEvent}$) [33]. An event occurs in a certain $\mathsf{situation}$ at a certain point in time, and transforms it to another situation, they may change reality by changing the state of affairs from one (pre-state) situation to a (post-state) situation [36]. $\mathsf{ObjectSnapshot}$ is an immutable state description of an object within a situation. Situation is a snapshot of object states valid in the given temporal range.

Moreover, UFO defines Dispositions which are Intrinsic Moments ($\mathsf{IntrinsicMoments} \sqsubseteq \mathsf{Moment}$), i.e. existentially dependent entities that are realizable through the occurrence of an Event ($\mathsf{Dispositions} \sqsubseteq \mathsf{Moment}$). This occurrence brings about a Situation [37]. In other words, UFO considers dispositions as properties that are only manifested in particular situations or the occurrence of certain triggering events, and that can also fail to be manifested ($\mathsf{Dispositions} \sqsubseteq (= 1\ \mathsf{isManifestedBy}{\cdot}\mathsf{Event})$). Dispositions inhere in particular objects ($\mathsf{Dispositions} \sqsubseteq (= 1\ \mathsf{inheresIn}{\cdot}\mathsf{Object})$). For example, security flaw in an information system is manifested by event of stealing sensitive data that brings about non-safe situation.

Additionally, UFO introduces the notion of agents ($\mathsf{Agent} \sqsubseteq \mathsf{Substational}$), i.e. proactive objects with an intention, the propositional content of intention is a Goal. intentions cause the agent to perform actions ($\exists\ \mathsf{performs} \cdot \top \sqsubseteq \mathsf{Object}$) [38]. Finally, UFO also defines services [39], and powertypes, i.e. universal types whose instances are individuals in the subject domain [40, 41].

We selected UFO for this work because of (i) our experience with using UFO in various conceptual model-based domains [29, 30], (ii) UFO addressing many essential aspects for conceptual modeling, which have not received sufficiently detailed attention in other foundational ontologies [31], (iii) the availability of its formal translation to OWL [42] and (iv) the availability of OntoUML, an ontology modeling language that could be used to create ontology-driven conceptual models and domain ontology in a variety of existing UML tools. OntoUML aims to design a language for structural conceptual modeling [31].

2.3.2. The Common Ontology of Value and Risk

In the Common Ontology of Value and Risk [19], the authors have presented an ontological analysis of risk which clarifies the connections between the concepts of value and risk. The ontology is based on the analysis of several risk assessment methodologies, used by different industries and domains. The ontology is grounded with the Unified Foundational Ontology (UFO) [32] and as such provides for the most recent and complete conceptualization of risk. The ontology discusses three different perspectives of risk: (i) the relational perspective that describes risk as the relationship of ascribing risk, which the authors classified as Risk Assessment; (ii) the experiential perspective that considers risk as a chain of events that impacts on an agent's goals or intentions, which the authors labelled as Risk Experience, e.g., having your phone stolen, which puts one in a a phone-less situation, which in turn hurts one's goals of contacting people; (iii) and the quantitative perspective that describes risk as a quantitative notion which they labelled as Risk, i.e., it describes the risk by means of the Risk qualities inhering in particular relations. common examples of the quantitative perspective include a color of an apple, a person's weight, etc.

Furthermore, because the ontology aims to discuss the connections between risk and value, the authors presented an ontological analysis of using value by (i) discussing the impact of likelihood of events, (ii) describing value as experience, its structure and the objects that participate in this experience, and (iii) clarifying the role of dispositions in value creation.

From the previous different perspectives on risk and value, the authors propose the Common Ontology of Value and Risk, formalized in OntoUML [32].

Limitations of the Common Ontology of Value and Risk The current version of the Common Ontology of Value and Risk, however, does not completely describe the domain of risk management as it lacks safety-related concepts such as mitigation and control strategies.

[2] We reuse Description Logic formalization of basic UFO concepts introduced in [35]

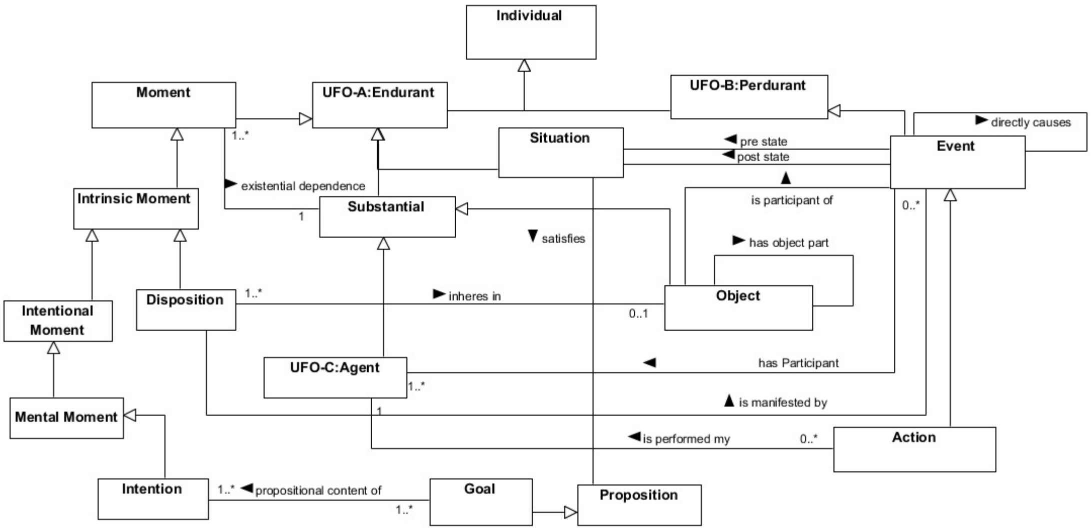

Fig. 2. Main concepts of UFO

In this paper, based on risk and value ontology and with respect to the principles of Unified Foundational Ontology, we present STAMP hazard and risk ontology which analyzes risks and hazardous states contributing to loss events, i.e. unsafe events such as accidents or incidents in the safety domain in accordance with STAMP. From this work we aim to describe the domain of risk management and assessment to solve the safety related problem in safety-based domain industries.

Moreover, to exemplify the limitation of the Common Ontology of Value and Risk so as the focus of our ontology work, we can use the concept of hazard as used in the aviation industry. Example of hazard are two aircraft in the air too close each other (also known as separation minima infringement). This situation implies that the requirement (constraint) for mininum aircraft separation was not enforced, or in other words violated by the hazard. This situation cannot be represented by the Common Ontology of Value and Risk.

3. Ontology Purpose Identification and Requirements Elicitation

3.1. Purpose Identification

Based on the knowledge acquisition activity documented in Section 2.2, we formulate the purpose of the ontology and draw representational requirements in the form of competency questions and non-functional requirements.

The purpose of the ontology is to allow for the representation of knowledge gained through a hazard analysis such as STPA. This knowledge is captured by the *risk/hazard model* which describes causality between future events (w.r.t. to a point in time) as well as their severity and likelihood. We recognize the existence of a similar causal model that describes historical events, referred to as the *historical causal model* in this paper. In contrast to the risk/hazard model, this model describes how events happened, what caused them and what was the loss associated with them, e.g. the friendly fire incident in Example 3.1.1.

Furthermore, there is a subtle connection between the two models. Instances of the historical model contribute to the formation of a risk/hazard model. For example, documented occurrences of incidents and accidents are summarized using statistical methods to asses the likelihood of causal links and the risk (i.e. the potential loss) of future events. Similarly, experts assess future events based on their experience.

Based on the discussion above we define:

Definition 3. *The purpose of the ontology is to represent knowledge of the STPA (hazard analysis) process. This knowledge is characterized by two main models,* historical causal model *and* risk/hazard model.

To exemplify the rest of the discussion we introduce an industry example for the application of the STPA methodology.

3.1.1. Industry example

Due to confidentiality restrictions, this section does not provide a real-world industry example from the environment of Prague Airport or Czech Airlines Technics, where this project was executed. We decided to exemplify our approach using an industry example provided directly by the author of STAMP [3], from the domain of military aviation, namely the friendly fire accident from April 15, 1994 that occurred in Iraq. On that day, two U.S. Air Force interceptors patrolled an area and mistakenly shot down two U.S. Army helicopters carrying 26 people, who all died in the accident.

Detailed investigation using STAMP principles is demonstrated directly by the author of STAMP. For the sake of practicality we take only the last three minutes of the accident, as follows:

- Time 0728: Lead interceptor pilot has visual contact with unidentified helicopter at 5 nautical miles.
- Time 0728: Lead interceptor pilot conducts identification pass and asks his wingman, using phraseology, whether he sees two enemy (Iraqi) helicopters.
- Time 0728: Wingman interceptor pilot confirms seeing two helicopters.
- Time 0729: Lead interceptor pilot instructs his wingman to disarm missiles, reports to the controllers of the operation (supervising flights in the area) that he engaged the targets.
- Time 0730: Interceptors fire at helicopters, they are hit by missiles.

The safety control structure involved in the last minutes is depicted in Fig. 3. Each of the figure elements consists of a separate control loop, where the simplified control structure above the pilots involves complex network of various control loops. Considering the definition of hazard from previous section, the system states and conditions can be derived from all involved control loops in the control structure so as from the relations between them. With regard to the last minutes of the accident mentioned above, example of hazard is early control action of the lead interceptor pilot who, being apparently in a rush, did not check thoroughly that his wingman in time 0728 actually did not confirm seeing two enemy helicopters but only two helicopters. Note that in Fig. 3 control-feedback relations are only hierarchical; there is only indirect coordination between the interceptors modelling the helicopters for which the mission control is responsible.

3.2. Requirement Elicitation

Designing and implementing an ontology impose several implicit non-functional requirements. The ontology should:

- *R1*: represent a shared conceptualization of the domain
- *R2*: be formalized, i.e. using a suitable formal language
- *R3*: include well documented concepts with their usage exemplified.
- *R4*: be compatible with the theory of the STAMP model
- *R4-a*: contain a STAMP Hazard profile based on STAMP failures (as shown in Fig. 1)

We employed several strategies to comply with the first requirement. First of all, as we mentioned before we ground our ontology in a foundational (or top level) ontology (UFO). This puts concepts into a well-founded conceptual framework and it should reduce conceptual interoperability problems compared to a design without a foundational ontology [31]. Second, the terminology used to represent the concepts of the ontology (i.e., the selection of relevant terms) is grounded in expert literature which allows for broader adoption, compared to defining new terms not aligned with the common language used by the community. Third, we try to find and reuse existing risk ontologies compatible with other risk models.

Formalization of the ontology is achieved in several steps. First, by modeling the domain using UFO we gain a formal representation of the model in modal logic. Furthermore, we use description logic formalization specifying an OWL ontology [42, 43]. The last requirement, documentation of the concepts, is achieved by annotation of the formal model with definitions and examples. The last two non-functional requirements are based on a broader initiative to formal-

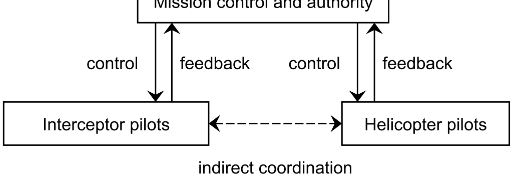

Fig. 3. Control of interceptors (fighters) and helicopters in the example mission (adapted from [3])

ize the entire theory of STAMP, which reuses the results of this paper (see Section 3.3).

Next we define the representational requirements of the ontology in terms of competency questions (CQs), which were derived from related STAMP literature [3, 44], by interviews with domain experts and ontology users discussing the STPA process:

- *CQ1:* What is the accident/What happened?
- *CQ2:* What are the hazards in the controlled system?
- *CQ3:* How does risk accumulate concerning specific hazard?
- *CQ4:* What are the factors of a given accident?
- *CQ5:* What is the STAMP failure classification?
- *CQ6:* Where is the potential for inadequate control actions (possible control flaws)?
- *CQ7:* Where can be identified responsibility for specific risks?
- *CQ8:* Which objects participate in a specific occurrence?

In the ontology validation section 8, we answered this questions by applying them on our running industry example 3.1.1

3.3. *Ontology Modularization*

To facilitate re-usability and interoperability with other ontologies we split the ontology into three main modules, namely the *Risk/Hazard causal* module, *Historical causal* module and the *STAMP Hazard* module. Compatibility with STAMP is ensured through the interaction of these modules with the rest of the STAMP theory. Fig. 4 shows a work-in-progress of the modularization of STAMP.

In order to fulfill requirement *R4* we need to examine the relation of the conceptualization designed here with the remaining conceptualization of the STAMP theory.

3.4. *Modeling*

With the help of experts and ontology users, we identify key terms in the example and the STAMP literature. The fragment of the STAMP terminology dealing with hazards and risks can be divided into two modules, the STAMP Hazard and Risk Ontology (SHRO) and the STAMP Control Loop Hazard Profile (SCLHP). The terms related to the SHRO ontology are shown in Tab. 3.4 and for SCLHP are the shown in Fig. 1.

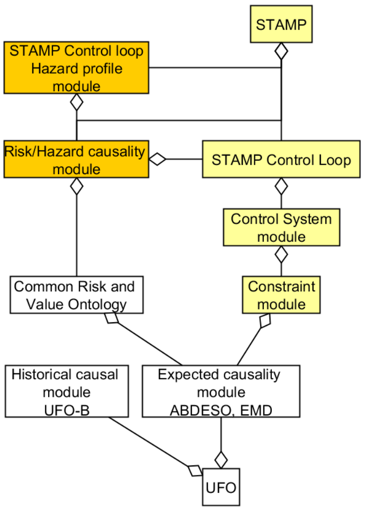
STAMP
STAMP Control loop Hazard profile module
Risk/Hazard causality module
STAMP Control Loop
Control System module
Common Risk and Value Ontology
Constraint module
Historical causal module UFO-B
Expected causality module ABDESO, EMD
UFO

Fig. 4. STAMP modules. (Legend: **orange** - modules discussed in this work; **yellow** - other STAMP modules; **white** - reused ontologies.)

Table 1

Terms referring to concepts and relations capturing the purpose of the SHRO ontology.

term
Accident
Factor
causes
contributes to
violates
Risk
Hazard
severity
likelihood
directly cause
mitigates
occurrence

4. STAMP Hazard and Risk Ontology (SHRO)

We designed SHRO for describing safety issues and increasing the awareness of analytic methods and tools for safety analyses in safety industry, focusing on the STAMP accident model, we tested this ontology in the domain of aviation, but it is not limited to the aviation industry. Our strategy is to analyze STAMP-based safety events that lead to incidents or accidents and explain STAMP-based hazards or factors, that contribute to safety events. Such approach ensures re-usability of the ontology for other high-risk industries. The ontological foundational model of SHRO is presented in

Fig. 5. The concepts are assigned colors as follows: *yellow* - concepts native to SHRO; *blue* - concepts reused from the Common Ontology of Value and Risk; *white* - UFO concepts reuse and light blue for SHRO relations.

According to definition 1 hazard concept describes any factor that causes or contributes to an unplanned and undesired loss event. That loss may involve human death and harm, but it may also include other major occurrences, including system or equipment damage, and information losses. Thus, there are different physical or social objects participating in the occurrence of hazard. In SHRO, we adopt three object roles that participate in a risk event, defined in the Common Ontology of Value and Risk:

Threat Object is a person or another object which poses danger to an asset (via threat event, e.g., attacks) , i.e. the objects participating in a threat event. An example of a Threat Object is a hacker of safety information.

Object at Risk is an object, which is exposed to potential damage. Objects at risk are constituted around traits such as loss, vulnerability, and need for protection, e.g. a person in an accident. Therefore, they deserve attention and care. For example, information should be protected from a hacker attack.

Risk Enabler is an object which is mainly responsible for risk event or accident to happen. It has inherent hazards in the sense that it refers to something that is identified as dangerous, e.g. the controller in STAMP model.

Axiom A1 captures this notion. A2 explains that, in our ontology: a hazard is manifested by a risk or loss event and this hazard inheres in an Risk Enabler object who is responsible for the loss and participates in this loss or risk event as in axiom A3.

$$\begin{aligned}\mathsf{RiskEvent} \sqsubseteq ((\geq 1 \;\; \text{hasParticipant} \cdot \mathsf{RiskEnabler}) \\ \sqcup (\geq 1 \;\; \text{hasParticipant} \cdot \mathsf{ObjectatRisk}) \\ \sqcup (\geq 1 \;\; \text{hasParticipant} \cdot \mathsf{ThreatObject}))\end{aligned} \tag{A1}$$

$$\begin{aligned}\mathsf{Hazard} \sqsubseteq ((\geq 1 \;\; \text{isManifestedBy} \cdot \mathsf{RiskEvent}) \\ \sqcap (= 1 \; \mathsf{inheresIn} \cdot \mathsf{RiskEnabler}))\end{aligned} \tag{A2}$$

$$\mathsf{RiskEnabler} \sqsubseteq (\geq 1 \;\; \text{participatesIn} \cdot \mathsf{RiskEvent}) \tag{A3}$$

As in the Common Ontology of Value and Risk, each loss and threat event is manifestation of some vulnerabilities, weak points or hazards that cause or lead to these events, i.e. accident's cause can be described, using STAMP, by identifying relevant safety constraints, that were violated by vulnerabilities or hazards. Example could be two aircraft violating minimum separation requirements [3]. However, there are situations in which the there is no violation of a constraint. One example is when an accident, such that the safety control structure does not account for occurs. Axiom A4 ensures that occurrence of any loss event is considered a constraint.

$$\mathsf{LossEvent} \sqsubseteq \exists \mathsf{eventToAvoid}^{-} \cdot \mathsf{AvoidEventConstraint} \tag{A4}$$

According to STAMP theory, a proper analysis and understanding of these hazards can resolve major part of safety issues and significantly reduce risk in everyday operations. In axiom A5, when a risk event happened, then it is a manifestation of a hazard and this hazard doesn't respect the safety constraints but violates them and that what axiom A6 defines.

$$\mathsf{RiskEvent} \sqsubseteq (\geq 1 \; \text{isManifestationOf} \cdot \mathsf{Hazard}) \tag{A5}$$

$$\mathsf{Hazard} \sqsubseteq (\geq 1 \;\; \text{violates} \cdot \mathsf{Constraint}) \tag{A6}$$

Furthermore, losses result from component failures as shown in Fig. 1, e.g. disturbances external to the system, interactions among system components, and behavior of individual system components. That leads to hazardous system states, which are denoted in Fig. 5 as STAMP Hazards (STAMP Failures). Example of hazards includes medical mistakes which are manifested by death of patients, where the loss event is caused by medical mistake hazard. Consequently, STAMP hazard and risk ontology must obey axiom A7 that hazard is manifested by risk event if and only if this hazard contributes to the risk event.

$$\mathsf{isManifestedBy.RiskEvent} \equiv \mathsf{contributesTo.RiskEvent} \tag{A7}$$

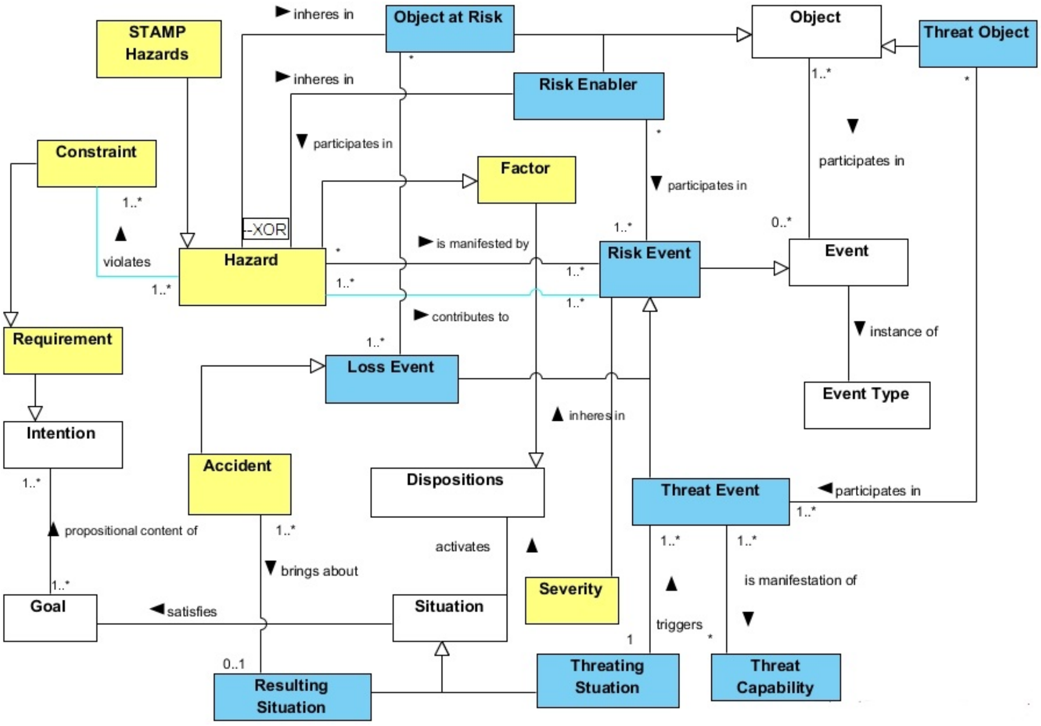

Fig. 5. Main concepts of SHRO grounded in the Common Ontology of Value and Risk and UFO.

As can be seen from Fig. 5, our ontology is mainly based on the Common Ontology of Value and Risk. It incorporates several terms that we explained before such as Risk Event, Loss Event, Threat Event, Object at Risk, Threat Object and Risk Enabler. However, there are many differences that need to be explained. SHRO aims to describe how hazardous states by violating the Safety Constraints contribute to loss event in the safety domain regarding specific accident model - the STAMP. Common Ontology of Value and Risk lacks safety-related concepts such as Hazards, Occurrence, STAMP Failures, violates, mitigates and Safety Constraints. Moreover, the Common Ontology of Value and Risk aims to explain the relations between value and risk, and how the Vulnerability could be considered as a positive and negative value in the same time according to the object's role that we discussed early in this section [19]. Since SHRO cares about safety issues, especially STAMP Failures or STAMP Hazards, it describes Hazard concept as unsafe concept. From our perspective, Vulnerability concept means that there are some weak points or features inhere in some object that are manifested by unwanted events. But Hazard is a safety related concept that is manifested by Occurrences of STAMP Failures which are safety events. Moreover, SHRO defines Safety Constraint concept that refers to acceptable ways the safety system has to follow to achieve its mission goals. However, in this paper, we don't describe this concept in detail, we only need the term Safety Constraint to fully describe hazards because they violate the Safety Constraint, and this violation causes a risk event in safety systems.

5. Probabilistic Risk Assessment

In this section, we describe a Risk as a future event, i.e. risk involving uncertainty about whether or not such a loss event will happen in the future.

Probabilistic risk analysis using event chains are used by the industry today to convey safety and risk information. In performing a probabilistic risk assessment (PRA), initiating events in the chain are usually assumed to be mutually exclusive. While this assumption simplifies the mathematics by combining probabilities of individual component failures and mutually exclusive events, it may not match reality.

In Fig. 6, we represent the likelihood of loss event and risk as a quality in terms of UFO [40, 41]. In [19], they differentiate between a Triggering Likeli-

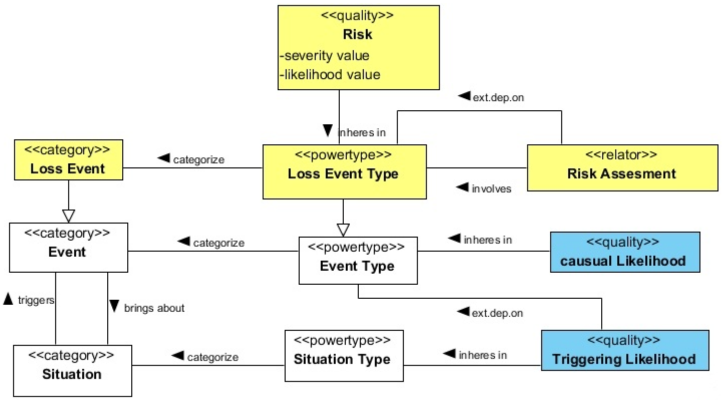

Fig. 6. Risk Likelihood in STAMP. Adopted and updated from [19].

hood, which inheres in a Situation Type and represents how likely a Situation Type will trigger an Event Type once a situation of this type becomes a fact, and a Causal Likelihood that inheres in an Event Type and represents how likely specific event e will cause another event type to occur. Risk as a quality should indicate two values in safety. First value is the severity that depends on the type of loss (e.g. if the loss event leads to death then the severity value is high but if it results in only small damages, the severity value is low etc.) and the second is the probability or likelihood, combining probabilities of individual failures in event model chain.

Regarding STAMP [3], probabilistic risk assessment (PRA) is not appropriate for systems controlled by software and by humans making cognitively complex decisions. There is no effective way to incorporate management and organizational factors, such as flaws in the safety culture, into PRA despite many well-intentioned efforts to do so. As a result, these critical factors in accidents are often omitted from risk assessment because analysts do not know how to obtain a “failure” probability, or alternatively, a number is pulled out of the air for convenience. The *ontological probabilities* are unknown and we can only study the probabilities of the existing factors to predict or specify the ontological probabilities. If we knew enough to measure these types of design flaws, it would be better to fix them than to try to measure them. But in a risk assessment, we analyze many instances of risk experiences, i.e. risk events that happened in the past and then measure the likelihood and risk values of these experience to qualify the Risk in the future. Another possibility for future progress is usually not considered, and so the conclusion in STAMP is that “Risk and safety may be best understood and communicated in ways other than probabilistic risk analysis” [3].

Nevertheless, STAMP does not reject probability value as a constituent of Risk in general. It only emphasizes that in complex systems this value is untraceable and for the purpose of achieving practical results of the Risk analysis, the value needs to be replaced by other variables, such as mitigation potential [45]. In this paper, however, we aim to propose formalization of the base PRA approach and so adhere to the standard probability inclusion. In future work concerning the overall STAMP ontology, we will address the need for offsetting the issues related to Risk assessment in complex systems, which will be possible by extending the ontological foundations provided in this work.

6. STAMP Control Loop Hazard Profile Ontology

In this section we model the STAMP control loop hazards from Fig. 1 according to the risk and value ontology. The motivation for this step is to assess and improve the appropriateness of the conceptual model of the hazard profile. By doing this we ensure for example that the hazards are properly associated with the components of the control loop as well as the events that are manifested by them. Furthermore, we uncover and discuss the limitations of the STAMP causal model.

Before we start modeling the STAMP hazards we first need to specify how we interpret the STAMP control loop hazards defined in Fig. 1, subsection 6.1. Next, we establish a minimal model for the control loop elements necessary to connect the hazards to control control loop components, subsection 6.2. Finally, we specify the ontological pattern used to represent our models(subsection 6.3). Subsection, present the result of our modeling of the “Inadequate Operation” for the Sensor component and discusses the modeling process.

6.1. Interpretation Of STAMP Hazards

The diagram in Fig. 1 is based on a well known concept from cybernetics, the feedback control loop. The diagram is composed of three types of elements, labeled boxes, arrows (representing control loop components) and labels representing the hazards of the control loop. We interpret the diagram as follows. The boxes represent the main components of the control loop labeled – controller, sensor, actuator and control-loop. The arrows represent communication (in a broad sense) between main control loop components. Hazards are written inside the boxes and along the arrows. All of the hazard labels are composed of two parts, the hazard’s subject and an adjective. The subject refers to a component or related concept of the component, e.g. part of, disposition, state and behavior. The adjective in the hazard label describes different kinds of inefficiencies. For example, the sensor’s “inadequate operation” has subject the “operation of the sensor” which is “inadequate”.

6.2. Minimal Conceptual Model of a Control Loop

Although this work focuses on the conceptualization of hazards and risks, we need to model to some extent the control loop components which are the main source of the hazards. We focus on categorizing the control loop components into the main UFO categories. This way we can take advantage of existing and well founded UFO patterns to capture the connection between the control loop components and their hazards.

We start the modeling with the box elements. We categorize the controller, sensor and actuator box elements as object types. The last box element is named “controlled process” but is also referred in STAMP **??** as a socio-technical system. We choose to model this element as an event type. This seams appropriate because it is the only way in UFO which allows to model together technical systems, resources, people, groups of people, their states, behaviors and their interactions. The arrows in the diagram represent means for the transfer of information/energy. We model arrows as objects.

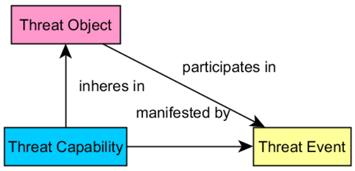

Fig. 7. Ontological Pattern from the Common Risk and Value Ontology used to model STAMP Hazards

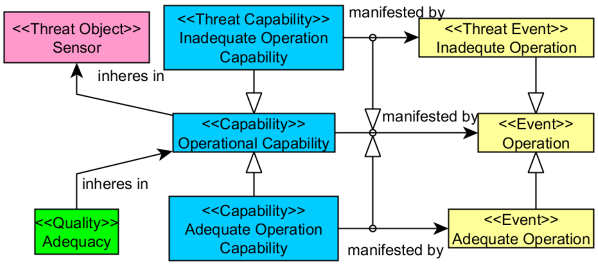

Fig. 8. Model of the Sensor’s Inadequate Operation as an Event according to the ontological pattern of hazards in Fig. 7

6.3. Ontological Pattern for Modeling STAMP Hazards

To model the STAMP hazards we are going to use the model of the feedback control as discussed above and the experience part of the Common Risk And value Ontology (CRVO). According to the experience profile of CRVO there are three event categories which we may choose from to model hazards. Namely, threats, enablers and losses. Losses correspond to events such as accidents and are not suitable for modeling the STAMP hazards. To model the STAMP hazards we mainly use the concepts preceding the losses, that is threats and enablers. Both the concepts of a threat and risk enabler are suitable in our modeling scenario from a structural perspective. Without loss of generality we select to model hazards using threats instead of enablers. Furthermore, note that according to the CRVO and our extension SHRO this modeling choice does not restrict one to classify threats as enablers and or losses if necessary. Fig. 7 shows the ontological pattern extracted from CRVO used to model STAMP hazards.

6.4. Modeling STAMP Hazards

Fig. 8 shows our model of the sensor’s “Inadequate Operation”. This model is obtain as follows. First, instantiate the ontological pattern from Fig 7 using the terms used to describe the hazard. Second, examine the possibility to generalize the hazard and extend the model according to the . First, we identify the control loop component in which the hazard occurs. In this case this is the “Sensor”. Next, we have to decide what is the

As we mentioned earlier, this hazard can be interpreted in different ways. Here we choose the interpretation where “Inadequate Operation” is an *Event Type*. According to the CRVO there should be a disposition (the hazard) which is manifested by the instances of the “Inadequate Operation” event type. The STAMP hazard model does not mention this disposition of the sensors (the same holds for some other components of the control loop). We name this disposition “Inadequate Operation Capability”. Finally there is a “participates in” relation between the “sensor” and the “Inadequate Operation” event type.

7. Analyzing STAMP Hazard and Risk in term of UFO

Building safer systems requires pouting emphasis on system hazards and eliminating or reducing their occurrence. Therefore, the Occurrence or Accident is a safety term that refers to the loss event that is caused by system hazards. Accident is a risk event i.e. a perdurant having endurants as its participants. Axiom A2 holds this notion. In UFO, an event occurs in a certain situation at a certain point in time and transforms it to another situation. In SHRO model we refer to this situation as Resulting Situation. Example of a type of occurrence that may occur in STAMP model-based safety system is two aircraft collision due to the lack of coordination between the airborne TCAS (collision avoidance) system and the ground air traffic controller, each giving different and conflicting advisories on how to avoid a collision. Another example is an *Accident* where one or several components failed, leading to a system failure. The last example could be crash accident due to coordination problems in the control of boundary areas [3]. In these three examples, regrading UFO principles, each of them are events that have starting and ending time.

Hence, UFO Events existentially depend on the objects that participate in them and an event is a manifestation of a disposition of an object, then a risk event occurs due to the dispositions of its participants, which are in STAMP model the Hazards (i.e. the dispositions). Therefore, we consider *Hazard* as dispositions in SHRO conceptual model. In UFO, dispositions are defined as properties that inhere in particular objects and are only manifested in particular situations of the occurrence of certain triggering events, and that can also fail to be manifested [37]. When manifested, they are manifested through the occurrence of resulting events and state changes. When dispositions enable undesired events, they are referred to as vulnerabilities or here in our model as hazards (Axiom A7 holds this notion). For example, the flaws in process creation in a safety system is manifested in system failure.

Accident causal analysis based on STAMP starts with identifying the safety constraints which are the requirements that the system should respect to achieve safety goals. Regarding STAMP, when these constraints are violated by hazardous states. UFO-C [34] defines requirement as an Intention and Goal, which is the propositional content of an Intention that inheres in an Agent. However, there is no obvious definition of constraints in UFO. STAMP defines safety constraints as part of system requirements that must be enforced to prevent hazard’s occurrences [3]. Axiom A5 explains this argument that having a hazard’s occurrence means that, a safety constrain is violated by this hazard. Therefore, in our model Constraint is a specialization of Requirement. For example, the safety-related design constraint might be “obstructions in the path of a closing door must be detected and the door closing motion reversed”. And the system safety requirement or constraint is that “the temperature in the reactor must always remain below a particular level” [3].

The previous analyzing of SHRO in terms of UFO falls into the the same alignment between SHRO and the Common Ontology of Value and Risk, which leads to verification of the SHRO in terms of the latter ontology.

8. Validation

For validation, we consider the SABiO guidelines methodology for ontology verification and validation [20].

8.1. SHRO Verification

To verify our ontology, we answer the constructed competency questions (CQs) by domain expert that he used to find the best answers directly from the STAMP theory [3], then we mapped these answers to the ontological axioms defined before, and check the conceptualization of our ontology by highlighting its concepts and relations in the answers, showing which elements of the ontology (concepts, relations, properties and axioms) answer each one of the Competency Questions (CQs). We highlight only SHRO concepts, we don't consider STAMP or UFO concepts. The results are shown in Tab. 2.

8.2. SHRO Validation

For validation, SABiO suggests that the ontology should be capable of properly representing real world situations. Therefore, we instantiated the ontology on the defined competency questions using the real world industry example from section 3.1.1, i.e., the helicopter shot down accident. Then, we tested these instances in our ontology by mapping between expected Outputs and SHRO matching concepts, if they exist. The selection of instances is done by the domain expert. Tab. 3 shows the results of the competency questions instances according to the helicopter shot down accident. The successful instantiating of SHRO in a real world situation indicates to the appropriateness of our proposed ontology as well as to the reference ontology. To defend our proposal and prove the appropriateness of our proposed ontology, we create a conceptual model for the helicopter shot down accident example, that analyzes the accident based on our ontology concepts. It is depicted in Fig. 9. The example concepts are in orange color.

9. Related Work

From the conceptual model perspective, we are not the first to analyze hazard and risk events. The Common Ontology of Value and Risk that we have discussed in detail in Section 2.3.2 was used as the base for this work. It formally characterizes the process of ascribing risk as a particular case of the process of ascribing value [19]. In [46], a well-founded ontology is provided for resources and capabilities modeling in enterprise architecture for ArchiMate. Modeling Enterprise Risk Management and Security with the ArchiMate Language paper identifies the Enterprise Risk Management (ERM) concepts, tests many standards and frameworks for ERM and security deployment, gathers a set of accepted risk by analyzing a representative sample of ERM, analyzes their semantics and describes the capabilities of the ArchiMate 2.1 [47]. In [48], the authors analyse the Risk and Security Overlay also of the ArchiMate language. Goal-Risk approach [49] is another related work which represents a goal-oriented approach for analyzing risks in term of requirements. In [47], enterprise architecture of risks by Archimate models is analyzed. In addition, in our previous work we proposed an aviation safety ontology that defines the basic concepts from the aviation industry and describes Ramp Error Decision Aid (REDA) Contributing Factors that cause some specific accidents [30].

Relation to our approach. Although each of the related works above presents a unique prospective in a risk analyzing and assessment, none of their approaches allows for integrating to a systematic model. In this paper, we analyze risk from the systematic approach based on foundational ontology (UFO) that puts concepts into a well-founded conceptual framework and it should reduce conceptual interoperability problems compared to design without a foundational ontology.

10. Conclusion

In this paper, we have discussed the ontological foundation of hazard and risk regarding the System-Theoretic Accident Model and Processes *(STAMP)* in aviation safety domain as a use case. As a result, we proposed STAMP hazard risk ontology *SHRO* which its implementation could help with creating semantic analyses of safety systems accidents and hazards. We followed the *SABiO* approach for identifying the purpose, eliciting requirements, formalizing, verifying and validating the ontologies. The proposed ontology describe loss events in both risk experience and risk assessment perspectives based on risk value ontology as a *Reference Ontology*. Moreover, we implemented *SHRO* in formal ontological language *OWL* which allows creating *SPARQL* Queries for testing our ontology by instantiating the competency questions*(CQs)* on realistic examples.

References

[1] EUROCONTROL, *A White Paper on Resilience Engineering for ATM*, European Organisation for the Safety of Air Navigation (EUROCONTROL), 2009.

[2] S. Dekker, *Safety Differently: Human Factors for a New Era, Second Edition*, CRC Press, 2017.

Table 2

Ontology verification

CQS	Answers with highlighting ontology relations and concepts	Axioms
CQ_1 : What is an accident?	**Accident** is an undesired and **unwanted event** or an **occurrence** that results in a **loss** of some **severity** (including **loss** of human life or injury, property damage, environmental pollution, and so on). Losses result from different **hazards** such component failures, disturbances external to the system, interactions among system components, and behavior of individual system components that lead to hazardous system states.	A5
CQ_2 : What are the hazards in the controlled system?	**Hazards** are system states or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to an **accident** or **loss**.	A2
CQ_3 : How does risk accumulate concerning specific hazard?	The basic STAMP concept is that most major **accidents** does not result simply from a unique set of proximal, physical **events** but from the migration of the organization to a state of heightened **risk** over time.	-
CQ_4 : What are the factors of this accident? (Why a specific accident happens?)	If there is an **accident**, one or more of the following **hazards** must have occurred: (1) the safety **constraints** were not enforced by the **controller**, (2) appropriate **control actions** were provided but not followed.	A5
CQ_5 : What is the STAMP failure classification?	Classification of **accident** causal **factors** starts by examining each of the basic components of a **control loop** and determining how their improper operation may **contribute to** the general types of **inadequate control** or **hazard**. The causal **factors** in **accidents** can be divided into three general categories: (1) the **controller** operation, (2) the behavior of **actuators** and controlled processes, and (3) communication and coordination among **controllers** and decision makers.	-
CQ_6 : Where is the potential for inadequate control actions (possible control flaws)?	**Inadequate control** includes cases where (a) the **control actions** necessary to enforce the associated safety **constraint** at each level of the socio-technical control structure for the system were not provided, (b) the necessary control actions were provided but at the wrong time (too early or too late) or stopped too soon, (c) unsafe control actions were provided that caused a **violation** of the safety **constraints**.	A6
CQ_7: Where can be identified responsibility for specific risks?	The responsibility for implementing each **requirement** needs to be assigned to the components of the control structure, along with requisite authority and accountability, as in any management system; **controls** must be designed to ensure that the responsibilities can be carried out; and feedback loops created to assist the **controller** in maintaining accurate process models.	A3
CQ_8: Which objects participate in a specific occurrence?	**Objects participating** in a specific **occurrence** are given by the safety control structure in place to control the **hazard** and enforce the safety **constraints**. This structure includes the roles and responsibilities of each component in the structure as well as the **controls** provided or created to execute their responsibilities and the relevant feedback provided to them to help them do this.	A1

[3] N. Leveson, *Engineering a Safer World: Systems Thinking Applied to Safety*, The MIT Press, Cambridge, Mass, 2012.

[4] E. Hollnagel, *FRAM: The Functional Resonance Analysis Method: Modelling Complex Socio-technical Systems*, Ashgate, Farnham, Surrey, UK England Burlington, VT, 2012.

[5] Y. Song, Applying System-Theoretic Accident Model and Processes (STAMP) to Hazard Analysis, Master's thesis, The School of Graduate Studies of McMaster University, 2012.

[6] R. Pereira, C. Morgado, I. Santos and P. Carvalho, STAMP Analysis of Deepwater Blowout Accident, *Chemical Engineering Transactions* **43** (2015), 2305–2310.

[7] P. Underwood, P. Waterson and G. Braithwaite, 'Accident investigation in the wild' – A small-scale, field-based evaluation of the STAMP method for accident analysis, *Safety Science* **82** (2016), 129–143.

[8] C.K. Allison, K.M. Revell, R. Sears and N.A. Stanton, Systems Theoretic Accident Model and Process (STAMP) safety modelling applied to an aircraft rapid decompression event, *Safety Science* **98** (2017), 159–166.

[9] Y. Zhou and F. Yan, Causal Analysis to a Subway Accident: A Comparison of STAMP and RAIB, *MATEC Web of Conferences* **160** (2018), 05002.

[10] R. Patriarca, J. Bergström and G.D. Gravio, Defining the functional resonance analysis space: Combining Abstraction Hierarchy and FRAM, *Reliability Engineering & System Safety* **165** (2017), 34–46.

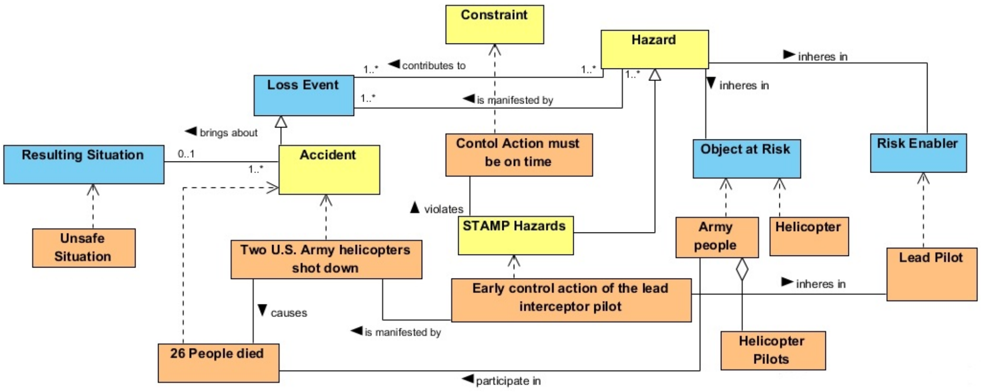

Fig. 9. Conceptual model of U.S. helicopters shot down accident

Table 3

Ontology validation

CQS Instances	**Input**	**Expected Output with SHRO matching concept if exists**
CQI_1: Who was responsible for visual identification of unidentified flying objects?(instantiated from CQ7)	visual identification event	The lead pilot and his wingman *(Risk Enabler)*
CQI_2: Which objects participated in the event of the helicopters shot down? (instantiated from CQ8)	helicopters shot down	Two F-15 fighter aircraft (interceptors) and two UH-60 helicopters *(Object at Risk)*
CQI_3 : How did the risk accumulate since the visual contact between the fighter aircraft and the helicopters? (instantiated from CQ3)	helicopters shot down	After the visual contact, the lead fighter pilot conducted visual identification pass and requested confirmation of identification from his wingman. This was received in rather ambiguous way (nor confirming visual contact with enemy helicopters), what was followed by instruction to disarm missiles and the shot down *(Event's Parts, causes)*
CQI_4 : What are the factors of this accident? (instantiated from CQ4)	helicopters shot down	Many inadequacies have been identified in the safety control structure (violated safety constraints, inadequate control actions etc.) at the time of the accident. All the control issues either directly caused or contributed to the accident *(Hazard).*

[11] K. Fukuda, T. Sawaragi, Y. Horiguchi and H. Nakanishi, Application of Functional Resonance Analysis Method to Evaluate Risks in Train Maneuvering, *Transactions of the Society of Instrument and Control Engineers* **52**(2) (2016), 68–76.

[12] R. Patriarca, A. Falegnami, F. Costantino and F. Bilotta, Resilience Engineering for socio-technical risk analysis: application in neuro-surgery, *Reliability Engineering & System Safety* **180** (2018), 321–335.

[13] A. Abdulkhaleq and W. Stefan, A-STPA: An Open Tool Support for System-Theoretic Process Analysis, *2014 STAMP Conference at MIT, Boston, USA* (2014).

[14] A. Abdulkhaleq and S. Wagner, XSTAMPP : An eXtensible STAMP Platform As Tool Support for Safety Engineering, *STAMP Conference* (2015). doi:10.13140/2.1.3862.0486.

[15] A. Abdulkhaleq and S. Wagner, XSTAMPP 2.0: new improvements to XSTAMPP Including CAST accident analysis and an extended approach to STPA, in: *STAMP Workshop (5th, 2016, Cambridge)*, 2016. doi:http://dx.doi.org/10.18419/opus-8749.

[16] H. Rees, FRAM MODEL VISUALISER, zerprize, 2016. http://www.zerprize.com/FRAM/index.html.

[17] T.-e. Kim, S. Nazir and K.I. Øvergård, A STAMP-based causal analysis of the Korean Sewol ferry accident, *Safety Science* **83** (2016), 93–101.

[18] E.H. och J. Speziali, SKI Report 2008:50: Study on Developments in Accident Investigation Methods: A Survey of the “State-of-the-Art", Technical Report, Swedish Nuclear Power Inspectorate (SKI), 2008.

[19] T. Prince Sales, F. BaiÃ£o, G. Guizzardi, J. Almeida, N. Guarino and J. Mylopoulos, The Common Ontology of Value and Risk, 2018.

[20] R. De Almeida Falbo, SABiO: Systematic approach for building ontologies, in: *CEUR Workshop Proceedings*, Vol. 1301, 2014, ISSN 16130073.

[21] T. Kletz, *HAZOP and HAZAN: Identifying and Assessing Process Industry Hazards*, Institution of Chemical Engineers, Rugby, Warwickshire, 2001.

[22] ICAO, *Doc. 9859: Safety Management Manual (SMM*, International Civil Aviation Organization (ICAO), Montreal, Quebec, 2018. ISBN 978-92-9249-214-4.

[23] J.C. Doyle, B.A. Francis and A.R. Tannenbaum, *Feedback Control Theory*, Dover, Mineola, N.Y, 2009.

[24] J. Rasmussen, Risk management in a dynamic society: a modelling problem, *Safety Science* **27**(2–3) (1997), 183–213.

[25] C. Perrow, *Normal Accidents: Living with High Risk Technologies*, Princeton University Press, Princeton, N.J, 1999.

[26] R. Studer, V.R. Benjamins and D. Fensel, Knowledge engineering: Principles and methods, *Data & Knowledge Engineering* **25**(1–2) (1998), 161–197.

[27] S. Auer, The RapidOWL Methodology–Towards Agile Knowledge Engineering, in: *15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises WETICE'06*, IEEE, 2006.

[28] R.D.A. Falbo, SABiO: Systematic Approach for Building Ontologies.

[29] P. Křemen, B. Kostov, M. Blaško, J. Ahmad, V. Plos, A. Lališ, S. Stojić and P. Vittek, Ontological foundations of european coordination centre for accident and incident reporting systems, *Journal of Aerospace Information Systems* (2017), ISSN 23273097. doi:10.2514/1.I010441.

[30] B. Kostov, J. Ahmad and P. Křemen, *Towards ontology-based safety information management in the aviation industry*, Vol. 10034 LNCS, 2017, ISSN 16113349. ISBN 9783319559605. doi:10.1007/978-3-319-55961-2_25.

[31] G. Guizzardi, Ontological Foundations for Structural Conceptual Model, PhD thesis, 2005, ISSN 13813617. ISBN 9075176813. doi:10.1007/978-3-642-31095-9_45. http://doc.utwente.nl/50826.

[32] G. Guizzardi and G. Wagner, *Towards Ontological Foundations for Agent Modelling Concepts Using the Unified Fundational Ontology (UFO)*, Lecture Notes in Computer Science, Vol. 3508, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 110–124. ISBN 978-3-540-25911-4. doi:10.1007/b136434. http://dx.doi.org/10.1007/11426714{_}8http://dl.acm.org/citation.cfm?id=2156041.2156051.

[33] G. Guizzardi, G. Wagner, R. de Almeida Falbo, R.S.S. Guizzardi and J.P.A. Almeida, Towards ontological foundations for the conceptual modeling of events, in: *Conceptual Modeling*, 2013, pp. 327–341. http://link.springer.com/chapter/10.1007/978-3-642-41924-9{_}27.

[34] G. Guizzardi and G. Wagner, Using the Unified Foundational Ontology (UFO) as a foundation for general conceptual modeling languages, in: *Theory and Applications of Ontology: Computer Applications*, 2010, pp. 175–196. ISBN 9789048188468. doi:10.1007/978-90-481-8847-5_8.

[35] A.B. Benevides, J.-R. Bourguet, G. Guizzardi and R. Peñaloza, Representing the UFO-B Foundational Ontology of Events in SROIQ, in: *Proceedings of the Joint Ontology Workshops 2017 Episode 3.*.

[36] G. Guizzardi, R. Falbo and R.S.S. Guizzardi, Grounding software domain ontologies in the unified foundational ontology (ufo): The case of the ode software process ontology, in: *In 1th Iberoamerican Workshop on Requirements Engineering and Software Environments (IDEAS'2008*, 2008.

[37] G. Guizzardi and G. Wagner, Dispositions and causal laws as the ontological foundation of transition rules in simulation models, in: *Simulation Conference (WSC), 2013 Winter*, 2014, pp. 1335–1346.

[38] G. Guizzardi and G. Wagner, Towards Ontological Foundations for Agent Modelling Concepts Using the Unified Fundational Ontology (UFO), in: *Proceedings of the 6th International Conference on Agent-Oriented Information Systems II*, AOIS'04, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 110–124. ISBN 3-540-25911-2, 978-3-540-25911-4. doi:10.1007/11426714_8. http://dx.doi.org/10.1007/11426714_8.

[39] C. Griffo, P. A João, J. Almeida, G. Guizzardi and J. Nardi, From an Ontology of Service Contracts to Contract Modeling in Enterprise Architecture, 2017.

[40] V.A. Carvalho, J.P.A. Almeida, C.M. Fonseca and G. Guizzardi, Multi-level ontology-based conceptual modeling, *Data Knowledge Engineering* **109** (2017), 3–24, Special issue on conceptual modeling — 34th International Conference on Conceptual Modeling, ISSN 0169-023X. doi:https://doi.org/10.1016/j.datak.2017.03.002. http://www.sciencedirect.com/science/article/pii/S0169023X17301052.

[41] G. Guizzardi, J.P.A. Almeida, N. Guarino and V.A.D.E. Carvalho, Towards an Ontological Analysis of Powertypes, in: *The Joint Ontology Workshops at the International Join Conference on Artificial Intelligence*, 2015.

[42] OWL 2 Web Ontology Language Document Overview, Technical Report, December, W3C Consoritum, 2012. http://www.w3.org/TR/owl2-overview/.

[43] I. Horrocks, O. Kutz and U. Sattler, The Even More Irresistible SROIQ., in: *KR*, P. Doherty, J. Mylopoulos and C.A. Welty, eds, AAAI Press, 2006, pp. 57–67. ISBN 978-1-57735-271-6. http://dblp.uni-trier.de/db/conf/kr/kr2006.html{#}HorrocksKS06{%}23my{_}node{_}{%}23.

[44] N. Leveson, A New Accident Model for Engineering Safer Systems, *Safety Science* **42**(4) (2004), 237–270.

[45] N. Leveson and N. Dulac, Incorporating Safety in Early System Architecture Trade Studies, *Journal of Spacecraft and Rockets* **46**(2) (2009), 430–437.

[46] C.L.B. Azevedo, M. Iacob, J.P.A. Almeida, M. van Sinderen, L.F. Pires and G. Guizzardi, in: *2013 17th IEEE International Enterprise Distributed Object Computing Conference*, 2013, pp. 39–48, ISSN 1541-7719. doi:10.1109/EDOC.2013.14.

[47] I. Band, W. Engelsman, C. Feltus, S. González Paredes, J. Hietala, H. Jonkers and S. Massart, Modeling enterprise risk management and security with the ArchiMate language, 2015, Document No.: W150.

[48] N. Mayer and C. Feltus, Evaluation of the risk and security overlay of archimate to model information system security risks, in: *2017 IEEE 21st International Enterprise Distributed Object Computing Workshop (EDOCW)*, 2017, pp. 106–116, ISSN 2325-6605. doi:10.1109/EDOCW.2017.30.

[49] Y. Asnar, P. Giorgini and J. Mylopoulos, Goal-driven Risk Assessment in Requirements Engineering, *Requir. Eng.* **16**(2) (2011), 101–116, ISSN 0947-3602. doi:10.1007/s00766-010-0112-x. http://dx.doi.org/10.1007/s00766-010-0112-x.