Received 6 June 2023; accepted 11 July 2023. Date of publication 17 July 2023;
date of current version 2 August 2023. The review of this article was arranged by Associate Editor Michael W. Sievers.

Digital Object Identifier 10.1109/OJSE.2023.3295946

BSafeML: A Model-Based Hazard Management Technique for Safety-Critical Systems Development

MICHAEL CUTAJAR [1] AND SIYUAN JI [2] (Member, IEEE)

[1]Nuclear Waste Services, Harwell Campus, OX11 0GD Didcot, U.K.
[2]Wolfson School of Mechanical, Electrical, and Manufacturing Engineering, Loughborough University, LE11 3TT Loughborough, U.K.

CORRESPONDING AUTHOR: SIYUAN JI (e-mail: s.ji@lboro.ac.uk)

The work of Michael Cutajar was funded by Nuclear Waste Services, and was conducted as a part of his study for the M.Sc. in Safety Critical Systems Engineering at the University of York.

ABSTRACT Effective management of hazards is at the heart of achieving acceptable safety for any safety-critical system. With the recent advancement in model-based systems engineering, various hazard management techniques have been proposed as a means to transition from a document-based paradigm, such as hazard logs implemented in a relational database to a model-based paradigm with standardized modeling languages. However, a review of the state-of-the-art has shown that the existing methods do not provide sufficient traceability to integrate hazard management with other system lifecycle activities. To address this gap, a new model-based hazard management technique, BSafeML, is developed. BSafeML is a unified modeling language profile, and a procedure extending the existing systems modeling language and SafeML profiles with language for modeling the behavior of hazards and mitigations. BSafeML integrates the structural and behavioral views of hazards, supporting traceability and semantic consistency over them and with the wider system-of-interest. Specific behaviors supported by BSafeML include accident sequences and ordered action of safety functions. BSafeML is evaluated in a case study of a waste package emplacement system in the context of geological disposal of radioactive waste. A hazard log, including a range of hazard types, is converted to model-based format with BSafeML. The evaluation is further supported by a stakeholder survey that revealed mostly positive attitudes toward the safety function modeling by BSafeML.

INDEX TERMS Geological disposal facility (GDF), hazard log, hazard management, model-based systems engineering (MBSE), SafeML, systems modeling language (SysML).

I. INTRODUCTION

The management of hazards is fundamental to safety engineering [1]. The principal artifact of hazard management, the hazard log, is used to structure hazard identification and assessment activities to capture the outcomes of these activities and to track the implementation of safety requirements derived from hazard assessment. The use of documents to record hazard logs relies on free-form text to describe hazards, text-based references between hazards and other safety artifacts, and manual processes to control the development of hazard information. These features pose significant challenges to consistency, traceability, and change control in hazard management.

Unsatisfactory solutions to these challenges in hazard management have inevitably contributed to catastrophic accidents. For instance, a seminal investigation of the loss of Nimrod XV230 [2] revealed a number of hazard records in the hazard log made reference to engineering mitigations that simply did not exist on the aircraft. Another notable exemplar accident was the explosion and fire at the Buncefiled oil depot in 2005 that resulted in 40 people being injured. The accident investigation [3] revealed a number of management failures, the first of which to be listed in the report was the absence of traceability between major hazards and "the safety-critical equipment and systems designed to control them".

Model-based systems engineering (MBSE) is increasingly adopted to realize benefits in consistency, traceability, and efficiency under change in the architecture of complex systems [4]. These are precisely the improvements that are needed in hazard management; however, there are not any well-established techniques for formal hazard modeling that can be integrated with MBSE. Various approaches have been proposed; however, they lack validation in real project environments. The adoption of model-based hazard management will depend on whether or not it can succinctly express all of the information recorded in the conventional hazard logs. Suitable research is required to demonstrate this.

This work also has a specific motivation from industry. Nuclear Waste Services (NWS) is the organization responsible for delivering a geological disposal facility (GDF) for radioactive waste arising from the U.K. nuclear industry [5]. As a major U.K. infrastructure project, the use of building information modeling (BIM) in the delivery of the GDF is mandatory [6]. BIM [7] is a fundamentally digital approach, establishing a coherent data model over all design artifacts. The increasing complexity of infrastructure projects has also motivated the introduction of systems engineering methods to the domain, with systems engineering guidance recently published by the Institution of Civil Engineers [8]. There follows a need to establish a digital thread connecting BIM to the systems engineering activities. This has previously been demonstrated in the context of BIM for space habitats [9]. Recent research has identified MBSE as a suitable basis for the digital thread [10], and MBSE adoption is increasing in the infrastructure and energy sectors [11].

A GDF is a safety-critical system, and the safety case for its operation and eventual closure is of paramount importance [5]. The digital thread must, therefore, extend into the safety domain if its benefits are to be fully realized in GDF delivery. This may be facilitated by a model-based description of safety information and of hazards, in particular, that can be integrated with MBSE.

Motivated by these industrial needs, this work offers a new model-based hazard management technique in a timely fashion. The technique, referred to as BSafeML, is a readily implementable profile that extends the systems modeling language (SysML) [12] and the SafeML profile proposed by Biggs et al. [13]. The novelty of this technique is that it supports the modeling of behavioral aspects of hazards in ways that are coherent with the structural aspects of the hazards, thereby establishing visualizable traceability and ensuring semantic consistency between digital artifacts, contributing to the development of digital threads for safety-critical systems, such as a GDF.

The rest of this article is organized as follows. In Section II, six use cases (UCs), compiled from standards in safety-critical industries, are presented. This is then followed by Section III offering a critical review of the existing model-based hazard management techniques against the UCs. Subsequently, In Section IV, we present the BSafeML profile in which its development is driven by the UCs and the identified issues in the existing techniques. The profile is evaluated through a GDF case study in Section V and further assessed through stakeholder questionnaire in Section VI. Finally, Section VII concludes this article.

II. HAZARD MANAGEMENT UCS

Taking a systems approach to the design of a model-based hazard management technique, we compiled six hazard management UCs from a in-depth review of hazard management standards and guidance in the aviation, nuclear, defense, and rail domains. The UCs describe the "problem space" of hazard management, and are, therefore, used to assess the existing model-based techniques for hazard management (see Section III) and will be used as the "requirements" that drive the development of BSafeML (see Section IV).

In the description of the six UCs, the concept of *hazard log* will be used, which is conventionally referred to as a document-based approach paired with a data management system, such as a relational database [14]. However, in this context, the description does not preclude model-based "hazard logs" as a solution to the UCs.

UC1. Record identified hazards: The primary use of a hazard log is to record the hazards revealed by hazard identification activities. The U.K. Civil Aviation Authority (CAA) defines a hazard log as "a structured way to record the hazards identified pertaining to a project or system" [15]. EU Aviation Safety Agency (EASA) guidance [16] recommends that the "organizations should wherever possible maintain a centralized log of all identified hazards" in its eight steps of risk assessment. Both sources of guidance recommend referencing the original hazard identification study in the hazard log by means of unique identifiers.

UC2. Group causes under hazards: Hazards can be considered to arise via several possible causes. This relationship is illustrated by the left-hand side of a bow–tie representation of a hazard. Hazard logs must be able to capture such grouping of causes under hazards. EASA guidance [16] provides a hazard log template wherein each hazard is presented as a table with rows of the table corresponding to causes ("safety events" in the EASA nomenclature). Office for Nuclear Regulation (ONR) guidance [17] also recognizes the use of hazard logs ("fault schedules" in the ONR nomenclature) to associate the subjects of hazard analysis with their causes. The importance of accurately capturing this grouping is stressed because inappropriate subdivision of hazards can result in "evasion" of any assessment criteria based on failure frequency [17].

UC3. Track hazards through the assessment: Hazard logs are used as live records to track the status of hazards and the status of the response to each hazard throughout a safety project. This UC is reflected in the U.K. Defense Standard 00-56 for Safety Management Requirements [18], which stipulates that "*the contractor should update the hazard log throughout the contract to ensure that it accurately reflects the status of the design, hazard analysis, safety analysis, and safety engineering activities.*" CAA guidance [15] provides more detail on the information that needs to be recorded to

fulfill this UC. In summary, the hazard log should record the current and next stages of the assessment process plus author and date information about the last update and next required update. Each hazard is effectively subjected to a controlled workflow, the status of which is recorded in the hazard log.

Furthermore, the hazard log should record whether each hazard is "open" or "closed" [15] where "closed" indicates that the hazard has been fully assessed and appropriate mitigation implemented. This is emphasized in the CENELEC 50128 standard for railway control and protection systems [19], which requires the duty holder to "ensure that the related hazard logs and remaining nonconformities are reviewed and that all hazards are closed in an appropriate manner through elimination or risk control/transfer measures." Hazard management must also accommodate the "reopening" of hazards or addition of new hazards, in response to design changes, modifications to operating systems, or occurrence of incidents [18].

UC4. Link hazards to other system and safety artifacts: The bow–tie representation of hazard management associates a hazard with three other types of system safety artifact: causes, consequences, and safety measures. EASA [16] and CAA [15] recommend also capturing the safety measures, termed *risk controls* and *mitigations* in the respective guidance documents and consequences in the hazard log. This ensures that each hazard can be traced to the subsystems that protect against it. ONR guidance [17] explicitly recommends tracing to safety measures, safety functions, and safety requirements, and also the "*operating mode, plant configuration, or plant state*" relevant to the hazard. The hazard log can, therefore, be used to trace between hazards and all key artifact types in a system description: functions, components, requirements, and states.

ONR guidance further describes traceability among causes, hazards, and safety measures as a "golden thread." This name emphasizes the importance of this UC to the development of a safety case. Defense Standard 00-56 provides similarly strong emphasis [18]: "*Traceability is fundamental, without it, it is not possible to understand how the results of low -level activities contribute to demonstrating satisfaction of requirements. If traceability is lost, then this can seriously undermine the validity of the hazard log, and hence the safety case for a PSS [product, service, or system].*"

UC5. Link hazards to safety arguments: Safety cases typically contain an argument over the hazards associated with a system [2], where each hazard is claimed to be addressed by the implementation of suitable safety measures. The set of identified hazards is the context for such an argument, and the hazard log provides a convenient reference when stating this context. Hazard logs may also be used to provide references from each hazard to specific claims in the safety case that address that hazard [17].

UC6. Communicate hazard information: The previous UCs describe how hazard information should be organized. The last UC concerns with communicating this information to stakeholders. Defense Standard 00-56 emphasizes the visibility of the hazard log throughout a safety project [18]. CAA guidance [15] describes the use of hazard logs at project milestones, for example, to demonstrate the completeness of hazard management to stakeholders prior to a system entering into operation. ONR guidance [17] also recognizes the "fault schedule" as a key regulatory interface, providing regulators with an overview of hazard management at a nuclear facility.

III. EXISTING PROPOSALS FOR MODEL-BASED HAZARD MANAGEMENT

Several approaches to model-based hazard management have been proposed in recent research. This section presents a review of these proposals and discusses the extent to which they exploit the benefits of model-based techniques and satisfy the hazard management UCs as formulated in Section II. This review does not cover attempts to conduct model-based safety assessment using standardized modeling languages as the dedicated safety model. The authors in [13] and [20] provide an account of relevant research on generating fault tree analysis and failure mode and effects analysis information from unified modeling language (UML) [21] and SysML models, but these are hazard assessment rather than hazard management techniques; hence, they will not be considered in this work.

A. MODELS OF HAZARD INFORMATION

An intuitive starting point in the investigation of model-based hazard management is to construct a model of hazard information with an established modeling language. Berenbach and Wolf [22] provide such a proposal in which the various artifacts involved in hazard management are represented in a UML class diagram. The focus of this research is on requirements management in general; however, the derivation of requirements from hazard analysis is addressed specifically. The model provides traceability among hazards, safety requirements, causes, and other system artifacts. These are all defined as UML classes where the attributes provide the data typically included in a hazard log, for examples, hazard description, probability, and severity. The model also associates causes with instances of the UML UC: using the system in a certain way and in a certain context gives rise to the hazards. Causes are also linked to the system component that is "implicated" in the hazard. The one-to-many relationship between the hazard and cause classes allows for modeling of hazard grouping as defined in the relevant UC for hazard management, as defined in Section I.

This model has been implemented in a modeling tool and used in a trial project [22]. The claimed benefits include specifically: improved traceability, the ability to view complex relationships between artifacts, and the model as a configuration-controlled single source of truth.

A limitation of this work is that it presents only a single model rather than a modeling technique as such. Each project following this approach would need its own UML model that may be difficult to compare to models in other projects. The presented model [22] is, therefore, effectively a

proof-of-concept that requires further development to deliver a standardized and reusable technique. The model also relies strongly on the UML class attributes to describe hazard characteristics, which are essentially free-form text. The model is, therefore, susceptible to semantic inconsistency in much the same way as a document-based hazard log.

There is ongoing work to apply a similar approach to the modeling of safety case information, including hazard information, in the U.K. nuclear industry [23]. This approach uses SysML block definition diagrams to describe all of the system and safety artifacts referenced in the safety case for operating a new nuclear reactor design in the U.K. The model provides traceability among hazards, safety measures, safety functions, safety requirements, and claims in the safety case. This example is, therefore, particularly relevant to the safety argument traceability UC.

B. PROFILES FOR SPECIFIC SAFETY STANDARDS

The approach described in Section III-A may be enhanced using the UML extensibility features. Creation of a modeling profile for hazard management, rather than just a model, supports reusability across projects. Modeling profiles may be created for general purpose hazard management or for alignment to specific safety standards. Examples of the latter approach are provided by Vepsäläinen and Kuikka [24] for the IEC 61508 functional safety standard and for its automotive adaptation ISO 26262 by Beckers et al. [25]. Both approaches define stereotypes for the safety integrity level (SIL) and automotive SIL concepts from the standards with enumerations capturing the various integrity levels and risk categories. These techniques demonstrate the relevance of model-based hazard management across domains, as well as the extensibility of UML to create domain-specific modeling techniques. Close alignment with a specific standard may support the safety argument traceability UC if conformance to the standard is an important safety case claim, but conversely, this may limit general applicability.

C. PROFILES FOR GENERAL HAZARD MANAGEMENT

Müller et al. [20] propose a generic, i.e., not standard specific, approach for integrating hazards into the modeling of a system. They propose a hazard analysis methodology based on a SysML model of the system together with a profile defining the hazard management extension. The profile includes hazard, cause, consequence, and safety measure stereotypes derived from the SysML block. Safety functions and malfunctions are also defined, derived from the SysML action, permitting explicit functional modeling of hazards; these are represented in an activity diagram on an alternate path to the normal control flow. The proposed methodology proceeds from UC definition to preliminary hazard identification to generate safety requirements, exploiting the SysML requirements syntax. This is followed by functional and physical architecture development within which malfunctions, hazards, safety functions, and safety measures are modeled using the new stereotypes.

A significant benefit of this approach [20] is the provision of a behavioral description of hazards using the malfunctions and safety functions. This permits modeling of a series of safety functions that act in some order, which is a feature of the bow–tie representation of hazards. The use of SysML is also beneficial for integration with the broader system model.

The profile [20] is demonstrated with a simple model and hazard analysis of a coffee maker. The technique still requires validation in a more complex context as the authors acknowledge. Furthermore, the relationship between the malfunction and hazard stereotypes is unclear; this may lead to inconsistency in how they are used in models, limiting the semantic control provided by the technique. The behavioral aspect of the technique also focuses on misuse and malfunction; it is not clear how external events would be accommodated.

An alternative hazard management based on SysML is the "SafeML" profile proposed by Biggs et al. [13]. This is informed by the IEC 61508 and ISO 26262 standards, as well as the ISO 12100 machinery design standard, but is intended to have more general applicability than the techniques referenced in Section III-B. SafeML models a harm context as the situation that converts an inherent hazard into an actual harm.

Harm contexts in SafeML [13] may be associated with SysML blocks, activities, and UCs, representing components, processes, and interfaces that may fail. This provides a flexible language of hazard causation, traceability to other system elements, and a means of fulfilling the hazard grouping UC. Safety measures are implemented by the stereotypes <<PassiveDefence>>and <<ActiveDefence>>that may be attached to the harm context and SysML requirements to provide a coherent description of derived safety requirements. Hazard attributes, such as probability and severity categories, are implemented as tagged values. These features of SafeML provide it with a relatively high degree of semantic control over the expression of hazard information, compared with the other techniques reviewed, in a manner compatible with the bow–tie representation of hazards.

A limitation of SafeML [13], particularly in comparison with the profile proposed by Müller et al. [20], is that it does not provide a means of explicitly modeling the functional behavior of hazards and safety measures. These are represented only structurally, although the safety measures can be linked to SysML requirements that may be functional in nature. It is, therefore, not clear how to model multiple safety measures acting in an order (as commonly expressed in a bow–tie diagram) or how to differentiate preventative and mitigative safety measures. SafeML also requires validation in more complex contexts than the presented test cases of a kettle and laser pointer.

D. WORKFLOW AND AUDIT ASPECTS OF HAZARD MANAGEMENT

The research discussed in Section III-A–III-C focuses on modeling the relationships between hazards and other system artifacts. An equally important aspect of hazard management is describing the provenance and workflow status of hazard

TABLE 1. Comparison of the Features of Existing Model-Based Hazard Management Proposals

Selected work	Language extension	Standard	Traceability improved	Semantic consistency	Function based	State based	Workflow modeling
Berenbach and Wolf [22]	✗	Generic	✓	Low	✗	✗	✗
Beckers et al. [25]	✓	ISO 26262	✓	High	✗	✗	✗
Jensen and Tumer [27]	✗	STAMP	✓	Low	✗	✗	✗
Müller et al. [20]	✓	Generic	✓	Low	✓	✗	✗
Fletcher [23]	✗	UK nuclear industry	✓	High	✗	✗	✗
SafeML [13]	✓	Bow–tie	✓	High	✗	✗	✗
SafeTIM [26]	✗	Generic	✗[1]	Low	✗	✗	✓
Vepsäläinen and Kuikka [24]	✓	IEC 61508	✓	High	✗	✗	✗

information. "SafeTIM" [26] is a UML model of such information. SafeTIM explicitly models the composition of a hazard log by a number of hazards, with defined relationships to the authors of the hazard records and the techniques used to identify the hazards. Versioning of hazard information is also modeled, although an explicit "open" versus "closed" status is not. SafeTIM does not attempt to model the relationships between hazards and other system artifacts.

SafeTIM [26] demonstrates that project and audit information for hazard management can be modeled. It is unclear, however, if a unified model of technical hazard information and project control information is desirable or if it is preferable to model only the technical hazard information while controlling it under a suitable configuration management tool.

E. SUMMARY OF REVIEW

Recent research has demonstrated a variety of approaches to modeling hazards, from creating models aligned to specific domains and standards to new modeling profiles for general hazard management. The most developed techniques use SysML with custom extensions to integrate hazard information into the system model. Both functional and structural approaches to describing hazards and their relationships with other system artifacts have been attempted, with advantages and disadvantages inherent to each. The reviewed techniques have been trialed in several domains, including process, nuclear, and automotive. However, validation in real projects with complex systems is so far lacking. This is understandable given that MBSE itself is a developing discipline and not fully established in all domains.

Table 1 compares the features of the reviewed techniques. All of the techniques reviewed provide traceability improvements over document-based approaches. Semantic consistency is improved to varying extents, more so in the case of the well-developed modeling profiles because they effectively specify new language for the description of hazards. The simpler models of hazards (using unextended UML class or SysML block diagrams) do not support reusability and standardization to the same extent and rely on the modeler to consistently describe hazard properties in class attributes. Improvements in traceability and semantic consistency are directly relevant to the hazard management UCs defined.

Work to date has not reproduced the expressiveness of the bow–tie method in full. The proposals of the authors in [13] and [20] capture different but separately incomplete the aspects of bow–tie. The former is capable of modeling safety functional behavior, but this feature may be difficult to apply to external hazards and lacks traceability to the structural view of hazards. The latter (SafeML) provides well-defined semantics for relating hazards to inherent harm potentials, causes, consequences, safety measures, and safety requirements; it lacks, however, any behavioral representation and cannot explicitly model safety measures acting in a given order.

Models of system behavior may include both function-based and state-based descriptions, as exemplified by the activity and state machine diagrams of SysML. While techniques implementing traceability between hazards and system functions have been proposed, no literature has been identified that explicitly considers the relationships between hazards and system states. The importance of traceability to system states is recognized in the relevant hazard management UC and the Nimrod accident [2].

Most of the reviewed research does not explicitly address the hazard management workflow UC. This aspect of hazard management may be modeled explicitly, as demonstrated by the SafeTIM model [26], but it is unclear if this is preferable over fulfilling the UC with appropriate tool selection instead. A model, such as SafeTIM, may perhaps be used in the design of a model configuration management tool.[1]

IV. BSAFEML

The design, development, and implementation of BSafeML is primarily driven by UC1, UC2, UC4, and UC6, as introduced in Section II, with the goal being enhancing current model-based approaches to hazard management as reviewed in Section III, while facilitating future works in managing hazards in real time (as per UC3) and linking model-based hazard artifacts to safety arguments in a safety case (as per UC5).

A. DEFINITIONS AND NOTIONS

To aid the understanding of the technical contents of BSafeML and corresponding discussions, the following definitions of key terms are being adopted.

Inherent Hazard: A part of a system or its environment with a potential to cause harm.

[1] SafeTIM provides traceability between hazard provenance and workflow information, but not between hazards and other system artifacts.

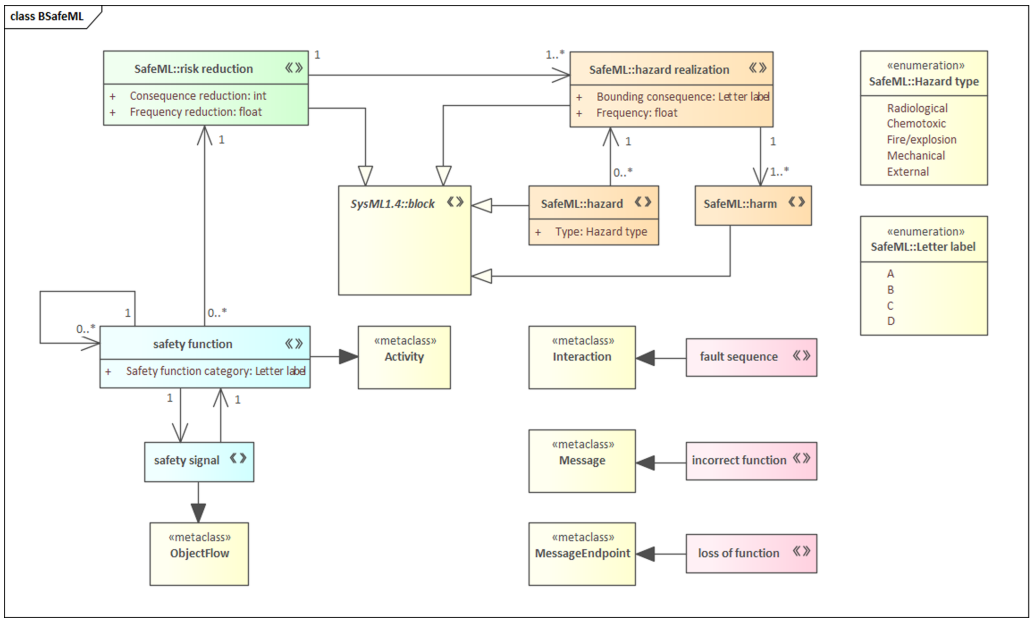

FIGURE 1. BSafeML profile.

Hazard Realization: A system state that may lead to an accident through, otherwise, normal behavior; it is also known as a "fault," "fault condition," or "fault group" in the nuclear industry.

Hazard: When used without further qualification, refers to both inherent hazards and hazard realizations.

Harm: A qualitative adverse effect of an accident.

Consequence: The severity of a harm.

Cause: An internal component failure, human error, or external event that may lead to a hazard realization.

Fault Sequence: The causal chain of events linking a cause to a hazard realization and subsequent accident.

Safety Function: A system behavior intended to prevent hazard realizations or reduce their consequences.

Safety Measure: A system, component, or person that implements a safety function.

Mitigation: Mitigation refers to both safety functions and safety measures, including those that prevent hazard realizations as well as those that mitigate consequences.

Further to the above definitions, the diagrams presented in BSafeML profile and models will follow the following consistent color scheme: yellow indicates the standard UML [21] and SysML [12] elements, orange indicates the SafeML [13] elements describing hazards, green indicates the SafeML elements describing mitigations, red indicates the elements of the new BSafeML technique describing hazards, and blue indicates the BSafeML elements describing mitigations. Finally, in the description of BSafeML profile and its procedure in the subsequent sections, metamodel elements are italicized.

B. BSAFEML—PROFILE

The BSafeML profile is depicted in Fig. 1. The structural description of hazards supported by BSafeML is inherited from SafeML [13]. Some of the inherited SafeML stereotype names are changed in BSafeML to match nuclear industry terminology. Inherent *hazards* are related to *harms* by *hazard realizations*. Both inherent *hazards* and *hazard realizations* can be traced to any relevant elements of a SysML [12] model of the system-of-interest using standard SysML connectors.

Mitigations in BSafeML are defined functionally, in contrast to SafeML where they are defined structurally. A BSafeML *hazard realization* is related to one or more *safety functions* via the *risk reduction* elements. Any *safety function* may be decomposed into lower level *safety functions* using the standard SysML *part association*. *Safety functions* can also be invoked on an activity diagram describing an activity of the system-of-interest and related to other *actions* using the standard SysML *object flow* and *control flow* connectors. If an *object flow* to or from a *safety function* represents a safety-critical input/output, the flow is stereotyped as a *safety signal*. *Safety functions* may also be allocated to structural elements of the system, i.e., safety measures, using the standard SysML *partitions* on the activity diagram.

Fault sequences are represented in BSafeML on sequence diagrams, with lifelines corresponding to the relevant elements of the system-of-interest. The *loss of function* and *incorrect function* stereotypes are used to show messages that should or should not have been sent, respectively, and thereby lead to a *hazard realization*. *Fault sequences* are grouped under *hazard realizations* as owned behaviors.

The behavioral features of BSafeML differ to those of Muller et al. [20] by inheriting the *safety functions* from SysML activity (so that they can be associated with *hazards* in the structural views) and by inheriting the *fault sequence* elements from SysML *interactions* (so that they can be owned behaviors of a BSafeML *hazard*). In [20], both *safety functions* and *malfunctions* inherit from SysML *action,* so they cannot be related to the structural view in a controlled way.

C. BSAFEML—PROCEDURE

The following steps explain how BSafeML is used. Prior to the application, it is assumed that the system-of-interest has been modeled using SysML and that this system model is accepted for conducting hazard-related activities.

1) Identify the inherent hazards and potential hazard realizations associated with the system-of-interest.
2) For each identified hazard realization, create a *package* in the system model.
   a) In the new *package*, create a block definition diagram and arrange on it the *hazard realization* as an *association block* between the inherent *hazard* and the *harm*.
   b) On the block definition diagram, link the *hazards* and *hazard realization* to the system elements that give rise to them.
   c) Consider the various causal sequences that lead to the *hazard realization* to:
      i) create a *fault sequence* for each and represent the sequence of events on the corresponding sequence diagram;
      ii) associate the *fault sequence* with the *hazard realization* as owned behaviors.
   d) Define the mitigations against the *hazard realization* as *safety functions* on the block definition diagram by:
      i) associating the *safety functions* with the *hazard realization* using a *risk reduction*;
      ii) decomposing the *safety functions*, as needed, into lower level *safety functions* on the block definition diagram.
   e) Place the *safety functions*, as invocations, into the activity diagrams that describe the general behavior of the system-of-interest:
      i) On each activity diagram, connect the *safety functions* to each other and to other *actions* or *parameter nodes* using the appropriate flow connectors;
      ii) Give any safety-critical *object that flows* the *safety signal stereotype*;
      iii) Allocate the *safety functions* to structural elements using *partitions* on the activity diagram.
   f) Place the *packages* corresponding to the various hazard realizations on a package diagram; this package diagram serves as the overall hazard log with the diagrams within a given package representing an individual hazard.

V. GDF CASE STUDY

A. GDF SYSTEM MODEL

One of the motivations for this project is to investigate the use of model-based hazard management in the specific context of a GDF for radioactive waste, as explained in Section I. A SysML model of a GDF is, therefore, constructed to provide a basis for the case study. The model is based on international consensus on geological disposal of radioactive waste [28], [29]. It is sufficiently generic so as to be representative of geological disposal in general and not of any country or organization specifically. The level of detail in the model corresponds to the system definition stage of a typical life cycle.

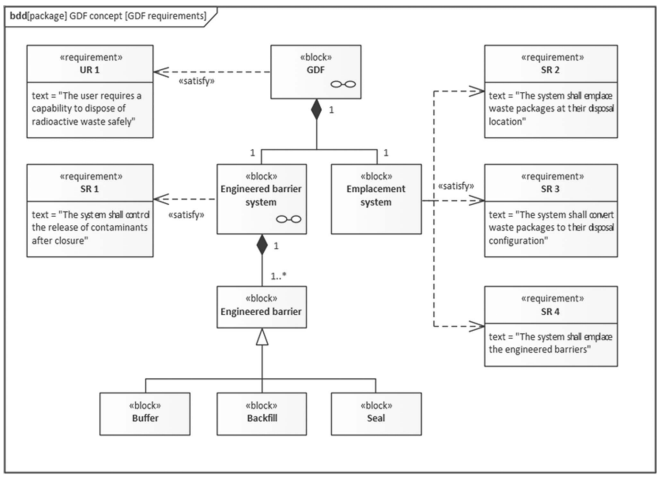

FIGURE 2. Principal components and requirements of a generic GDF.

The principal components and requirements of a GDF are defined in Fig. 2. These include an engineered barrier system that provides containment of waste underground and an emplacement system that moves the waste packages and the engineered barriers to the appropriate underground locations. The engineered barrier system comprises multiple individual barriers that may be buffers, backfills, or seals. These features of geological disposal are explained in International Atomic Energy Agency guidance [29]. Additional diagrams representing the generic GDF model are provided in the research data [30] for readers who are unfamiliar with geological disposal. These diagrams are not used directly in the BSafeML evaluation.

The case study focuses on the emplacement system and, in particular, the process of emplacing radioactive waste packages at the appropriate underground locations. The principal functions involved are shown in Fig. 3. Fig. 4 provides a view of the generic GDF facilities. Waste packages are imported to the GDF in shielded transport containers. The containers are moved underground through an access way, also called a "drift." The waste packages are removed from the transport containers and emplaced in an underground disposal vault, and the empty transport containers are decontaminated and exported off-site.

A sample of hazards associated with GDF emplacement operations is assembled to support comparison of BSafeML against a document-based hazard log. The generic GDF model itself is kept sufficiently generic so as to be a representative of geological disposal in general and not of any country or organization specifically. This sample is presented as a hazard log in Table 2. It includes both errors of commission and omission, human errors, internal component failures, structural

TABLE 2. Nonexhaustive Tabular Hazard Log for Generic GDF Waste Package Emplacement Operations

Hazard Realization	Freq. [/Year]	Hazard	Harm Consequence	Fault Sequences	Safety Functions / Requirements	CAT.	Sub- Safety Functions	Safety Measures
Exposure to waste package during unpacking operations	0.1	Radiation from packages	External radiation dose / C	Worker opens facility door while unshielded package present	Prevent access to facility during unpacking operations	B	Detect package presence in facility	Package detection system
							Lock facility door	Access control system
							Warn workers of unpacking operations	
				Facility door fails open	Single failure criterion	-	-	Redundant shield door
Exposure to waste package during container maintenance	0.01	Radiation from packages	External radiation dose / B	Loaded waste container returned to container maintenance facility in error	Prevent exposure to waste package during container maintenance operations	A	Determine container configuration (loaded/empty)	Package detection system
							If loaded container present, disable package handling	Package handling system
					If worker exposed to waste package, warn worker of radiation flux	C	-	Alarm system
Flammable gas build-up in disposal vaults	0.1	Flammable gas released from packages	Injury from fire or explosion / A	Loss of off-site power; loss of power to ventilation	If loss of off-site power, maintain power supply to ventilation system	A	Provide back-up power	Power supply system
					Maximum evacuation distance	C	-	Safe haven
Derailment of drift vehicle during transport of waste package to sub-surface	0.0001	Earthquake	Internal radiation dose / B	Earthquake derails drift vehicle during transport of waste	Prevent derailment of drift vehicle under seismic load	A	Detect seismic load on drift vehicle	Earthquake detection system
		Radioactive material in packages					If unsafe seismic load on drift vehicle, decelerate drift vehicle to stop	Surface-to-subsurface conveyance system
Tunnel collapse	0.001	Earthquake	Injury from falling debris / A	Earthquake induced tunnel collapse	Withstand seismic loads	-	-	Robust tunnel structure
		Tunneling explosives		Explosion induced tunnel collapse				

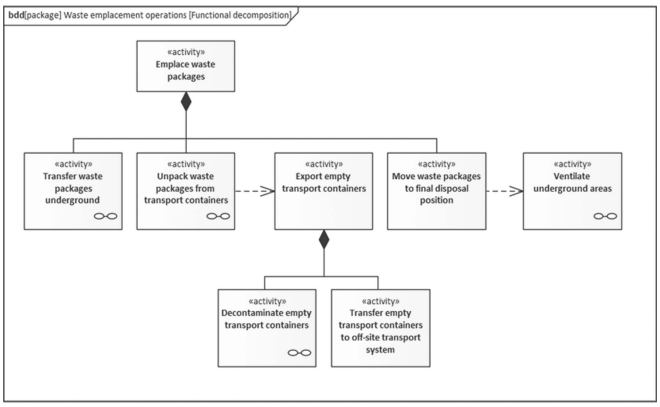

FIGURE 3. Functional breakdown of the waste package emplacement process.

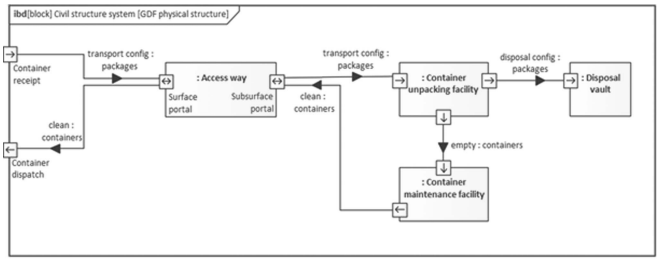

FIGURE 4. Generic GDF waste package handling facilities.

failures, external events, continuous processes, and discrete operations. This is reasonably comprehensive in the range of hazard types, and therefore suitable to evaluate BSafeML.

B. TABULAR HAZARD LOG TO BSAFEML MODELS

The GDF SysML model and sample hazard log correspond to the first two steps of the BSafeML application procedure, as defined in Section IV-C. The remaining steps of the procedure were followed to produce a BSafeML version (graphical models) of the (tabular) hazard log. For clarity, this section will only demonstrate diagrams produced for the "exposure to a waste package during unpacking operations" hazard (as in the first row of Table 2). BSafeML diagrams that were produced for the remaining hazards are included in the accompanied research data [30] for interested readers. The BSafeML hazard realization diagram for the "exposure to a waste package during unpacking operations" hazard is shown in Fig. 5. The hazard describes the potential for a GDF worker to enter the area where a radioactive waste package is being removed from its shielded transport container, thereby exposing the worker to radiation. The realization diagram shows two of the key features of BSafeML: these are namely the traceability between the hazard realization and the fault sequence, where the fault sequence is an owned behavior, and between the hazard realization and the safety functions. The diagram also demonstrates the key features of SafeML [13] that are retained in BSafeML: these include the traceability to other system elements (namely the "unpack waste packages from transport containers" activity) and the controlled semantics with which the hazard realization, inherent hazard, harm, and risk reduction are described.

The safety functions in this diagram describe an interlock system that prevents the worker from accessing the area when the waste package is present. The safety functions are shown

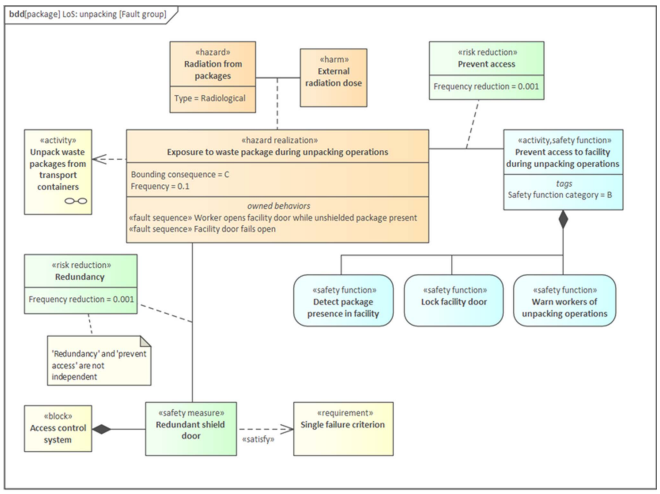

FIGURE 5. BSafeML hazard realization diagram for the hazard—"exposure to a waste package during unpacking operations."

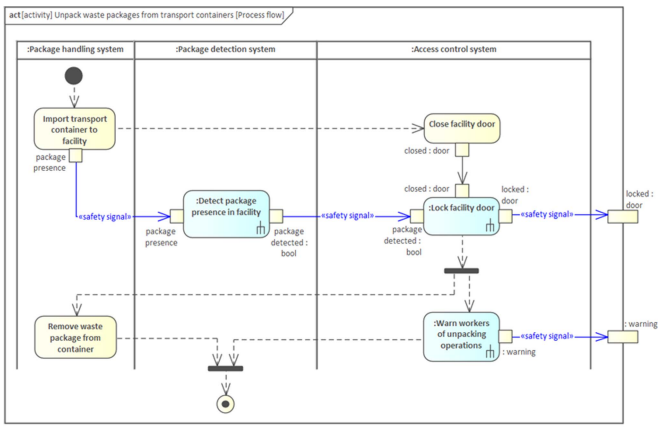

FIGURE 6. BSafeML activity for safety functional behavior associated with waste package unpacking operations.

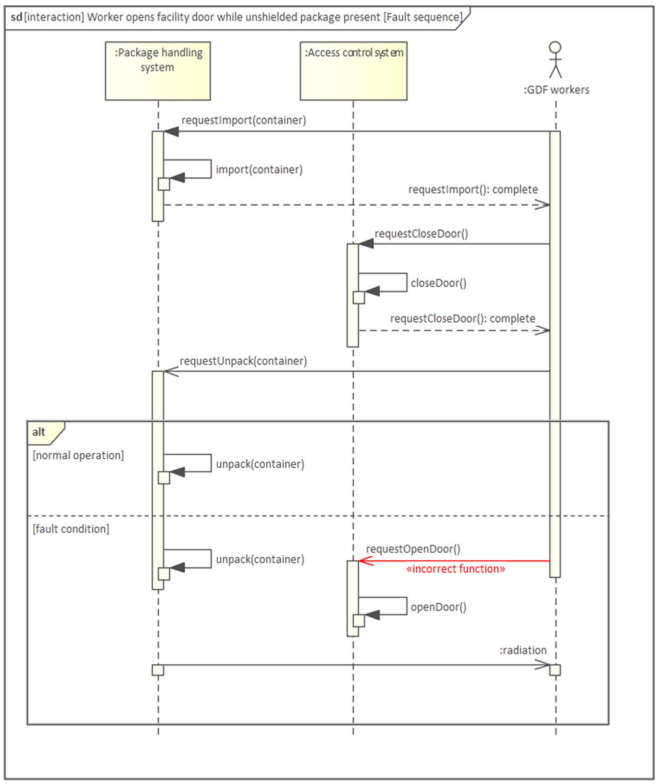

FIGURE 7. BSafeML fault sequence diagram for inadvertent entry into a hazardous area.

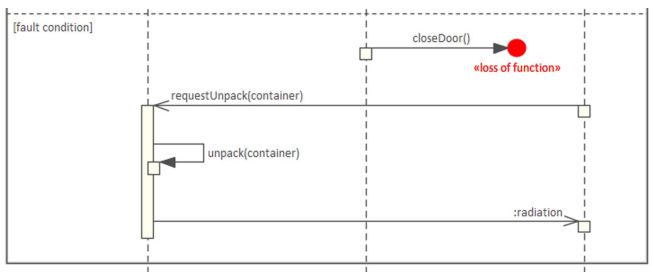

FIGURE 8. BSafeML fault sequence fragment (c.f. Fig. 7 showing a shield door failing open.

acting in the "unpack waste packages from transport containers" activity diagram, as presented in Fig. 6.

There are two fault sequences associated with the container unpacking hazard realization, as shown in Fig. 5. One involves the worker erroneously opening the facility door, and the other involves the door itself failing in the open position. The latter fault sequence does not introduce any new functional behavior, but rather a nonfunctional reliability requirement. This part of the mitigation is, therefore, modeled using elements from SafeML. The fault sequences are represented in BSafeML sequence diagrams in Figs. 7 and 8, respectively. Fig. 8 shows only the fault condition fragment, for brevity, as the rest of the sequence is already shown in Fig. 7. These diagrams demonstrate the BSafeML fault sequence modeling capability in the GDF context, showing, respectively, an error of commission and an error of omission. Traceability is provided through the lifelines, which are elements of the GDF system model, and through the owned behaviors of the hazard realization.

Finally, Fig. 9 shows a package diagram with one package defined for each of the modeled hazard realizations. Within each package are the relevant BSafeML diagrams. This package diagram serves as the model-based hazard log itself, with each package corresponding to an entry in a document-based hazard log. Stakeholders interacting with the hazard log in the tool can start from the package diagram and easily navigate to the desired hazard information.

C. DISCUSSION

Based on the BSafeML capability demonstrated in the GDF case study, we argue that profile has successfully satisfied UC1, UC2, and UC4 of the four model-based hazard management UCs that it intends to address. However, there are a few observed limitations during the GDF study.

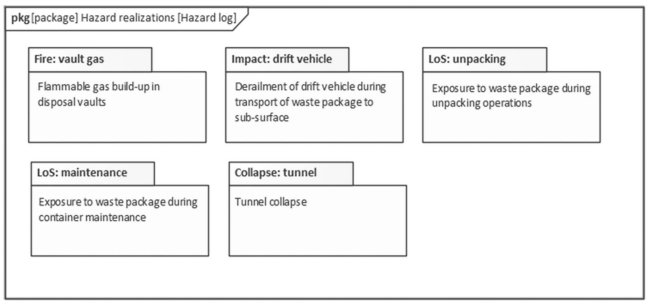

FIGURE 9. Model-based GDF hazard log implemented with BSafeML.

First, the interpretation of inherent hazard to hazard realization could be misleading when the hazard realization is associated with more than one inherent hazard (see the diagrams produced for the vehicle derailment hazard that is associated with the radioactive material in the waste packages and the earthquake, as detailed in the research data as an example [30]). This is because the hazard realization acts as a SysML association block between an inherent hazard and a harm. While one hazard realization may be related to many hazards or harms, the hazards and harms must be related one-to-one with each other and the relationship between a pair represented as an association block. This syntax breaks down when the relationship between hazards and harms is many-to-one. Currently, additional hazards are linked to the harm with a simple association. This is unsatisfactory because it leaves the additional hazards only indirectly traceable to the hazard realization. It may also be confusing to stakeholders of the model.

Another potential issue observed from the construction of Fig. 5 is the modeling of mitigations that are dependent. The risk reduction associating each mitigation with the hazard realization has a tagged value, "frequency reduction," specifying the reduction in frequency of occurrence of the hazard realization. This could be easily misinterpreted if the mitigations are dependent: a stakeholder might incorrectly assume that the stated risk reductions are provided separately and should be multiplied to give the total risk reduction.

VI. STAKEHOLDER EVALUATION

A pivotal claim for any model-based approaches, compared with document-based approaches, is that they offer better stakeholder communication through graphical models. The GDF case study conducted in Section V does not offer an evaluation of BSafeML against UC6 formulated in Section II. Therefore, a qualitative study is performed to assess how well BSafeML communicates hazard information to relevant stakeholders. The results are presented in this section.

A. EVALUATION STRATEGY

The stakeholder evaluation approach is primarily based on a predesigned questionnaire but administered to each respondent (stakeholder) in a one-to-one video call, lasting 1 h. The time limit is considered mandatory to assess the understandability of BSafeML models in a constrained time frame. The one-to-one format is chosen for certain advantages as identified by Gillham [31], specifically to permit clarification of the questions and diagrams used in the questions and to elicit additional unstructured feedback that might give deeper insight into the answers given.

The audience is carefully selected to ensure the validity of the result. The respondent must be the following:

1) be experienced safety engineers;
2) have a senior role;
3) practice or supervise hazard-related engineering activities, e.g., hazard identification, assessment, etc.;
4) have working knowledge in GDF waste package emplacement system;
5) not have significant experience in model-based approaches to hazard management;
6) little familiarity with SysML.

The first three criteria were defined to ensure that respondents have sufficient experiences in document-based hazard management techniques to be able to give comparative views to BSafeML, which is model-based. The fourth criterion is needed to ensure that the time used by the respondent to understand the safety aspects in the case study is minimized in the 1 h time frame such that most time can be used to assess the technique rather than the hazards themselves. In addition, the last two criteria were defined to remove potential bias toward favoring model-based approach before seeing BSafeML in action.

Due to the nonstandard format of the questionnaire administration, the specific criteria on respondents, and limited resources, seven candidates from the GDF safety case team at NWS were identified as appropriate and have all agreed to participate the stakeholder evaluation.

B. QUESTIONNAIRE DESIGN

The questionnaire is formed of three parts.[2] The first part is consisted of a set of five diagnostic questions posed alongside standard SysML diagrams of the generic GDF model. The purpose of these questions is to identify any correlation between respondents' attitudes toward standard SysML diagrams and their attitudes toward BSafeML diagrams, by splitting the respondents into groups who do and do not find the standard SysML diagrams to be clear. A statement is included asserting that the diagrams do not necessarily correspond to the GDF designs of any particular organization or country. This is to limit potential systematic bias due to respondents being sampled from members of the U.K.'s GDF project: it is important that respondents understand that their answers will not have an effect on their day-to-day work.

The second part of the questionnaire starts with Q5 that presents Table 2 to the respondent to establish mutual understanding of the system and hazards of interest. This is

[2]Full questionnaire containing a total of 20 questions is provided in the accompanied research data [30] for interested readers.

then followed by a set of 12 evaluation questions that ask about the respondents' attitudes toward BSafeML diagrams, e.g., the ones presented in the case study in Section V, when compared with what is presented in the table in Q5. These questions aim to establish the extent to which respondents find BSafeML diagrams to be clear, if the behavioral extensions introduced in BSafeML improve their understanding of the hazards, and if they prefer the BSafeML or the tabular views of the hazard log.

The third part of the questionnaire is formed of three reflective questions, probing reasons for the other answers given.

The evaluation and diagnostic questions use Likert scales to measure the strengths of respondents' attitudes. Five-point scales are used but the middle option is implied, so only four options are listed on the questionnaire. Respondents' general attitudes toward each BSafeML diagram are queried separately to their attitudes about particular aspects and to their preferences for (graphical) model versus table. The general question about the overall clarity of the models is asked first in each instance.

An effort is made to avert social desirability bias by wording questions in terms of the clarity of models rather than the knowledge of the respondents: the form "is the diagram clear?" is used instead of "do you understand the diagram?". Respondents are also told that critical and complementary views are of equal value to the study and suggest the normalcy of not finding the diagrams to be clear.

C. RESULTS OF THE EVALUATION

Respondents who answered positively ("agree" or "strongly agree") to at least two of the SysML diagnostic questions in the first part are considered to have passed the SysML diagnostic, i.e., they expressed a partial understanding of the SysML diagrams. The respondents who passed and failed the SysML diagnostic are treated as separate groups in the subsequent analysis.

The analysis on the second part of the questionnaire has revealed the following key highlights.

On Q6, Q7, Q11, and Q13 that are dedicated to BSafeML hazard realization diagrams, e.g., Fig. 5, the respondents were almost evenly split in whether they found the models to be clear. Most of them found the tabular hazard log (see Table 2) to be more effective at communicating information about the hazard than the hazard realization diagrams. It is worth noting that limitation in presenting one hazard realization-to-many hazards, as discussed in Section V-C, is also independently identified by several respondents. Respondents who failed SysML diagnostic are found to lean toward models being unclear.

On Q8, Q9, Q12, and Q16 that are dedicated to BSafeML activity diagrams, e.g., Fig. 6, responses are noticeably more favorable toward BSafeML, and even more, so only if respondents who passed the SysML diagnostic are considered. Most respondents agreed that the BSafeML activity diagrams improve understanding of the safety functions. Those that did not were confused by the name of the safety signal

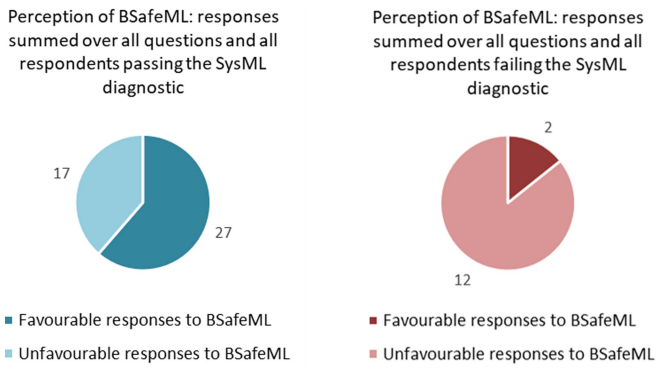

FIGURE 10. Responses summed over all questions.

stereotype: they interpreted this to refer to instrumentation specifically, rather than a safety-critical input/output generally, due to their previous experience of the term being used in that way. It is plausible that an alternative stereotype name would have resulted in all respondents finding the BSafeML activity diagrams to improve understanding of the safety functions. Several respondents commented on the capability of the BSafeML activity diagrams to support further hazard identification and to provide a subject for the HAZOP technique [32] specifically.

Finally, on Q14 and Q15 that are dedicated to BsafeML sequence diagram, e.g., Fig. 7, the responses are again mixed: roughly half the respondents found the fault sequence diagrams to be clear and to improve understanding, with the remainder finding the opposite. Positive comments regarding these diagrams emphasized their clarity over long textual descriptions, especially in the case of more complicated sequences. Negative comments mostly concern the text notation on the diagrams, which strongly reflects the software engineering heritage of UML, and the presentation of both the normal operation and fault condition as alt fragments: it was suggested that this would be clearer with more whitespace between the fragments.

In general, the responses show some correlation between the responses to the BSafeML diagrams and the responses to the SysML diagnostic questions. This is more apparent if the responses are aggregated over Q6 to Q16, as shown in Fig. 10. In this figure, answers that "agree" or "strongly agree" that the BSafeML diagrams are clear or improve understanding and answers that state a preference for the BSafeML diagrams over the tabular hazard log are counted as favorable toward BSafeML; answers that "disagree" or "strongly disagree" with those statements or express preference for the tabular hazard log are counted unfavorable toward BSafeML. The result here suggests that BSafeML is clear and adds value to most stakeholders who also find standard SysML diagrams to be clear, and vice-versa. These results suggest a need for further training in model-based techniques—a need that was reflected in comments made by several respondents.

For results on the third part of the questionnaire, it can be concluded that the overall attitude toward BSafeML is

mixed. The results, as per Q18, indicate that three respondents preferred the tabular presentation of the hazard log and three preferred the BSafeML presentation; one respondent abstained from this question. As per Q19, the most common reason given for preferring the tabular version was clarity and intelligibility with three responses. Reasons for preferring the BSafeML model were mixed with two responses each for clarity and intelligibility, expressiveness, and integration with the general system model. Most respondents, despite expressing a preference for one or the other, commented that, in reality, a variety of formats are important in hazard management (including text, tables, and diagrams). In the final question, Q20, five of the seven respondents believed that BSafeML would add more value as the maturity of the GDF design increased. One respondent abstained from this question and one respondent believed that BSafeML would add less value in a more mature design context.

D. LIMITATIONS OF THE RESULTS

Evidently, resulting from the relatively narrow selection criteria, the stakeholder evaluation is limited primarily by the small sample size. The statistical significance of any trends present in such a small sample is necessarily small. The criteria may introduce systematic bias, if the safety professionals are not representative of hazard management stakeholders generally. The extent of such bias (or any social desirability bias) cannot be determined without involving more respondents from other geological disposal organizations. Acknowledging these shortcomings, the respondents are nevertheless geological disposal safety experts, so the results have some degree of validity.

The small number of respondents also precluded pretesting of the questionnaire [33], [34] as that would require a separate test sample. Even with pretesting, there are fundamental limitations on determining the validity of attitude-based questions [35], [36]. A more rigorous evaluation would, therefore, require other measures in addition to the expressed attitudes of stakeholders. Other such measures could include project key performance indicators relating to hazard management, e.g., the time taken to progress a hazard log through a review process or the frequency with which hazard traceability problems are raised by project members.

Overall, a complete validation of BSafeML would require a large sample of randomly selected respondents, pretesting of the questionnaire, additional case studies from diverse projects and domains, and more objective measures than stakeholder attitudes alone. These will be the subjects of a future work.

VII. CONCLUSION

In this article, we have presented a new model-based hazard management technique, BSafeML, extending the existing SafeML technique to support coherent and integrated modeling of both the structural and behavioral aspects of hazards and their mitigation. This is a novel contribution to the field of model-based hazard management, providing a promising solution to the complete model-based description of hazards, thereby offering improved understandability of a hazard log and enabling wider traceability to digital artifacts produced in typical model-based approaches to the engineering of safety-critical systems.

The usage of activity and sequence diagrams in the modeling and management of the behaviors of hazards and their mitigation is particularly insightful. They offer plausible means to model, respectively, safety functions and fault sequences, which are, otherwise, poorly addressed in the current state-of-the-art. This claim is supported by the stakeholder evaluation in which the behavioral modeling attracted the most positive feedback. The interrelationships of these behavioral models to other system artifacts, such as test cases and safety case arguments, are of particular interest for a future work.

Limitations of BSafeML has been accounted for in the case study discussion and through the analysis of the stakeholder questionnaire. While acknowledging the BSafeML can be further improved to address issues identified through the evaluations, it is worth noting from the stakeholder engagement that a model-based approach is only as good as the competency of the engineers in practicing model-based approaches. This insight could raise a worrying concern when considering the fact that the evolution of model-based technique relies on better modeling languages, which themselves continue to evolve to becoming more capable, but likely also more complex and difficult to master by the practitioners. This would eventually defeat the original intention of any model-based approach, such as facilitating stakeholder management by means of offering intuitive diagrams.

REFERENCES

[1] Health and Safety Executive, "Risk assessment: A brief guide to controlling risks in the workplace," Bootle, U.K., 2014.
[2] C. Haddon-Cave, *The Nimrod Review: An Independent Review Into the Broader Issues Surrounding the Loss of the RAF Nimrod MR2 Aircraft XV230 in Afghanistan in 2006*. London, U.K.: The Stationary Office, 2009.
[3] Competent Authority for the Control of Major Accident Hazards, "Buncefield: Why did it happen," 2011.
[4] B. Beihoff et al., "World in motion–systems engineering vision 2025," 2014.
[5] Radioactive Waste Management, "Geological disposal: An overview of the generic disposal system safety case," Nuclear Decommissioning Authority, Rep. DSSC/412/01, 2016.
[6] Infrastructure and Projects Authority, "Government construction strategy 2016-20," London, U.K, 2016.
[7] ISO, "BIM–building information modelling," ISO 19650, Geneva, Switzerland, 2018.
[8] Institution of Civil Engineers, "A systems approach to infrastructure delivery," London, U.K, 2020.
[9] R. P. Casillas and S. A. Howe, "Virtual construction of space habitats: Connecting building information models (BIM) and SysML," in *Proc. AIAA Space 2013 Conf. Expo.*, 2013, Paper 5508.
[10] A. M. Madni, C. C. Madni, and S. D. Lucero, "Leveraging digital twin technology in model-based systems engineering," *Systems*, vol. 7, no. 1, 2019, Art. no. 7.
[11] R. Cloutier and I. Obiako, "Model-based systems engineering adoption trends 2009–2018," in *Guide to the Systems Engineering Body of Knowledge (SEBoK) v2.6*, SEBoK Editorial Board and R. J. Cloutier, Eds., Hoboken, NJ, USA: The Trustees of the Stevens Institute of Technology, 2019. Accessed: Sep. 12, 2022. [Online]. Available: www.sebokwiki.org

[12] Object Management Group, "OMG systems modeling language," OMG Document Number formal/2017-05-01, Needham, MA, USA, 2017.
[13] G. Biggs, T. Sakamoto, and T. Kotoku, "A profile and tool for modelling safety information with design information in SysML," *Softw. Syst. Model.*, vol. 15, pp. 147–178, 2016.
[14] R. Lewis, "Safety case development as an information modelling problem," in *Safety-Critical Systems: Problems, Process and Practice*. London, U.K.: Springer, 2009, pp. 183–193.
[15] Civil Aviation Authority, "CAP 760: Guidance on the conduct of hazard identification, risk assessment and the production of safety cases," London, U.K.: The Stationary Office, 2010.
[16] I. Maragakis et al., "ECAST guidance on hazards identification," European Union Aviation Safety Agency, 2009. [Online]. Available: www.easa.europa.eu/en/document-library/general-publications/ecast-guidance-hazards-identification
[17] Office for Nuclear Regulation, "Design basis analysis," Technical Assessment Guides NS-TAST-GD-006, Revision 5, Bootle, U.K., 2020.
[18] *Safety Management Requirements for Defence Systems—Part 1: Requirements*, Defence Standard 00-56 Part 1, Ministry of Defence, 2017.
[19] European Committee for Electrotechnical Standardization, "Railway applications—Communication, signalling and processing systems—Software for railway control and protection systems," UNE EN 50128:2012, 2011.
[20] M. Müller, M. Roth, and U. Lindemann, "The hazard analysis profile: Linking safety analysis and SysML," in *Proc. Annu. IEEE Syst. Conf.*, 2016, pp. 1–7.
[21] Object Management Group, "OMG unified modeling language (OMG UML) infrastructure," OMG Document Number formal, Needham, MA, USA, Aug. 5, 2011.
[22] B. Berenbach and T. Wolf, "A unified requirements model; integrating features, use cases, requirements, requirements analysis and hazard analysis," in *Proc. Int. Conf. Glob. Softw. Eng.*, 2007, pp. 197–203.
[23] S. Fletcher, "Application of model-based systems engineering (MBSE) in the U.K. nuclear sector," Mar. 2020, Accessed on: Sep. 4, 2021. [Online]. Available: www.innovationfornuclear.co.uk/safetyandsecurity.html
[24] T. Vepsäläinen and S. Kuikka, "Towards model-based development of safety-related control applications," in *Proc. ETFA2011*, 2011, pp. 1–9.
[25] K. Beckers, M. Heisel, T. Frese, and D. Hatebur, "A structured and model-based hazard analysis and risk assessment method for automotive systems," in *Proc. IEEE 24th Int. Symp. Softw. Rel. Eng.*, 2013, pp. 238–247.
[26] S. Nair, J. L. de la Vara, A. Melzi, G. Tagliaferri, L. de-la-Beaujardiere, and F. Belmonte, "Safety evidence traceability: Problem analysis and model," in *Proc. 20th Int. Work. Conf. Requirements Eng. Found. Softw. Qual.*, 2014, vol. 8396, pp. 309–324.
[27] D. C. Jensen and I. Y. Tumer, "Modeling and analysis of safety in early design," *Procedia Comput. Sci.*, vol. 16, pp. 824–833, 2013.
[28] International Atomic Energy Agency, "Disposal of radioactive waste," in *IAEA Specific Saf. Requirements SSR-5*, Vienna, Austria, 2011.
[29] International Atomic Energy Agency, "Geological disposal facilities for radioactive waste," *IAEA Specific Saf. Guide SSG-14*, Vienna, Austria, 2011.
[30] [] M. Cutajar and S. Ji, "BSafeML research data," Louhgborough Univ., Loughborough, U.K., 2023, [Online]. Available: https://doi.org/10.17028/rd.lboro.22795049.v1
[31] B. Gillham, *Developing a Questionnaire*, 2nd ed. London, U.K.: Continuum, 2008.
[32] British Standards Institution, "Hazard and operability studies (HAZOP studies)–Application guide," London, U.K., BS EN 61882, 2016.
[33] J. Blair, R. F. Czaja, and E. A. Blair, *Designing Surveys: A Guide to Decisions and Procedures*, 3rd ed. Newbury Park, CA, USA: Sage, 2014.
[34] W. E. Saris and I. N. Gallhofer, *Design, Evaluation, and Analysis of Questionnaires for Survey Research*, 2nd ed. Hoboken, NJ, USA: Wiley, 2014.
[35] N. M. Bradburn, S. Sudman, and B. Wansink, *Asking Questions: The Definitive Guide to Questionnaire Design—For Market Research, Political Polls, and Social and Health Questionnaires*, 2nd ed. Hoboken, NJ, USA: Wiley, 2004.
[36] P. Lietz, "Research into questionnaire design: A summary of the literature," *Int. J. Market Res.*, vol. 52, no. 2, pp. 249–272, 2010.

MICHAEL CUTAJAR received the Ph.D. degree in physics from Imperial College London, London, U.K., in 2012, and the M.Sc. degree in safety critical systems engineering from the University of York, York, U.K., in 2023.

He is a Senior Safety Systems Engineer with Nuclear Waste Services, U.K. His work is focused on developing digital safety case and model-based systems engineering capabilities for use in the design of a geological disposal facility for U.K. radioactive waste interests.

SIYUAN JI (Member, IEEE) received the M.Sc. and Ph.D. degrees in physics from the University of Nottingham, Nottingham, U.K., in 2011 and 2015, respectively.

He is a Senior Lecturer in systems engineering with Loughborough University, Loughborough, U.K. He was a Lecturer and the Programme Lead for M.Sc. in safety-critical systems engineering with the Department of Computer Science, University of York, U.K. His research is focused on model-based systems engineering and system safety assessments, and constraint-driven design algorithms.