Assessing the Usefulness of Assurance Cases: an Experience with the CERN Large Hadron Collider

Chris Rees[1], Torin Viger[2], Mateo Delgado[1], Rolf Lippelt[1], Jeff Joyce[1], Simon Diemert[1], Claudio Menghi[3], Marsha Chechik[2], Jan Uythoven[4], Markus Zerlauth[4], and Lukas Felsberger[4]

[1]Critical Systems Labs, Inc.
[2]University of Toronto
[3]University of Bergamo and McMaster University
[4]European Council for Nuclear Research (CERN)

April 14, 2023

Abstract

Assurance cases are structured arguments designed to show that a system functions properly in its operational environment. They are mandated by safety standards and are largely used in industry; however, they are typically proprietary and not publicly available. Therefore, the benefits of assurance case development are usually not rigorously documented, measured, or assessed.

In this paper, we present an assurance case for the CERN Large Hadron Collider (LHC) Machine Protection System (MPS). We used open-source documentation for its creation and used the eliminative argumentation, methodology for assurance case development. The development involved four authors with considerable experience in assurance case development, three of whom work for Critical System Labs, a small enterprise specializing in assurance cases.

Our experience shows that (a) the cost and time required to develop our assurance case is negligible compared to the effort needed to develop the system, and (b) eliminative argumentation helped identify 10 defeaters not detailed in the documentation used for creation of the assurance case. In this paper, we describe our experience and findings, and also discuss how the LHC assurance case helped accurately identify Key Performance Indicators for the MPS.

1 Introduction

Assurance cases (ACs) are arguments intended to show that a system will function as expected in its operational environment. ACs connect technical evidence about a system to high-level claims that a wide range of stakeholders can understand, enabling engineers and reviewers to assess whether proper risk mitigations are in place. ACs are used in many domains (e.g., automotive, rail, and control [1]), mandated by safety standards (e.g., ISO 15026-2 [2], ISO 26262 [3], and EN 51026 [4]), and are often represented graphically (e.g., [5, 6, 7]).

Despite the interest of the research (e.g., [5, 6, 8]) and industrial (e.g., [9]) communities in ACs, the benefits of the usage of AC are seldom empirically assessed or publicly shared. On one side, the research community is usually interested in the development of new notations for AC representation (e.g., [5, 6, 8]) and automated reasoning tools (e.g., [10, 11, 12]) that are typically evaluated on showcase examples which differ significantly from those developed in the industry. On the other side, practitioners extensively use ACs. However, companies often have strict nondisclosure policies; therefore, there is limited knowledge sharing about practical assurance strategies and industrial examples [13]. We are aware of only a handful of attempts (e.g., [1, 9, 14]) aiming to assess the benefits of using ACs in practice, and not aware of any works that evaluate the benefits of AC development over real, publicly available AC case study.

This paper presents an AC for the CERN Large Hadron Collider Machine Protection System (MPS) [15]. The LHC is a particle accelerator and collider built by the European Organization for Nuclear Research (CERN) [16]. The LHC is a cyber-physical system combining hardware and software components [17, 18, 19]. We selected the LHC since it is a sizeable industrial case study from the nuclear domain, and we could interact with CERN engineers to empirically assess the results of our study. We relied on open-source documentation for AC creation and used *eliminative argumentation (EA)* [20] as a graphical notation. EA explicitly supports modeling defeaters, i.e., reasons to doubt AC claims. We used EA since it is a well-known methodology for AC development; it is supported by existing tools [21] and considered by Critical Systems Labs (CSL) [22] in similar works [1].

CSL is a small-medium Canadian enterprise specializing in assessing and managing complex software safety and security risks. We developed our AC using Socrates [21], an industrial collaborative tool for AC development. Development took approximately three months and involved three engineers working at CSL and one Ph.D. student with four years of experience in the assurance case domain. Our AC is a significant example comprising 506 nodes. This AC is of medium size, according to industrial experience. We made our AC publicly available [23][1].

We collected metrics and reflected on the AC creation process with the goal of answering two research questions: *What is the effort needed to develop an assurance case for a complex system* (**RQ1**)? and *How useful is the creation*

[1]The assurance case is publicly available on the CERN website at the following address: `https://cds.cern.ch/record/2854725`

of an assurance case (**RQ2**)? In terms of effort, the time (91.9 days) and estimated cost required to develop an AC for the LHC MPS are significantly lower than system development (10 years and $\approx$ 4.4 billion USD for construction of the LHC, of which $\approx$ 5% was estimated to be spent on the MPS $\approx$ 200 million USD). To analyze effectiveness, we interacted with CERN experts to understand the impact of the defeaters that our AC creation process identified but that were not detailed in the documentation available to us. CERN experts confirmed all the identified defeaters and found only a handful of new defeaters that were not included in the argument. Therefore, we conclude that EA shows high precision and recall for identifying valid defeaters.

Finally, we reflect on the practical implications of these findings and discuss how EA enables the accurate identification of *Key Performance Indicators* (KPIs): detectable measurements of events that may gauge the performance of a system beneficial for a safety-critical system such as the MPS.

To summarize, this paper makes the following contributions:

1. We develop a medium-size AC for the LHC MPS by using EA;
2. We make publicly available all the resources we used for the AC development, the process we followed, and our AC enabling the replication of our experiments and results by other researchers or practitioners;
3. We empirically assess the effort and the usefulness of AC development in a large industrial case study.

The paper is structured as follows. Section 2 presents the LHC and the MPS component. Section 3 provides relevant background information on ACs and EA. Section 4 presents the methodology used for the AC development and to evaluate each research question. Section 5 describes the AC for the MPS. Section 6 presents our evaluation results. Section 8 discusses related work. Section 9 concludes by summarizing key results and describing plans for future work.

2 The Large Hadron Collider

The *Large Hadron Collider (LHC)* is a particle accelerator and collider built by the European Organization for Nuclear Research (CERN). The LHC enables testing theories and investigating unanswered questions in particle physics by observing collisions between highly accelerated particles. Building the LHC required approximately 10 years [24, 25] and costed approximately 4.6 billion SFr ($\approx$ 4.4 billion USD [24]). We selected the LHC as our case study since (a) it is a large and representative complex system; (b) it is carefully documented; (c) the documentation is publicly available, and (d) we had contact with CERN engineers that could help us answer our research questions.

The LHC consists of two 27-kilometer-long rings that accelerate particles to nearly the speed of light in opposite directions (see Figure 1). Particle beams travel around each ring in clusters (with particle-free gaps between them), and

Figure 1: The 27 km LHC tunnel, housing the LHC accelerator, here showing the superconducting magnets containing the two beam pipes. [26].

over 10000 magnets are used to bend and focus beams around the rings. During collision experiments, the trajectories of these beams are diverted so that they intersect at four collision points, and phenomena related to the collision are then detected and analyzed by a range of large-scale particle detectors.

The accelerated particle beams circulating in the LHC have high energy and destructive force and pose a significant risk of damage to the system if their trajectories become unstable (one proton beam within the LHC has the power of an aircraft carrier moving at 12 knots). Further, a substantial amount of energy is stored in the electrical circuits used to power the LHC magnets, and an uncontrolled release of even a small portion of this energy could result in damage to the LHC. Thus, the machine should be sufficiently protected from the damages described above.

The *Machine Protection System (MPS)* is comprised of inter-dependent components designed to ensure that the LHC does not become damaged during operation. It proactively protects the system by monitoring all conditions that could lead to damage, and issuing a beam dump (i.e., extracting all particles from the LHC rings) before hazardous scenarios occur. Each critical component of the MPS has redundancy so that, if a failure occurs, backups of the malfunctioning MPS component will be in place to extract the beam before damage is caused. The MPS is designed to protect the LHC from two main hazardous scenarios: beam loss and magnet quenches. In this paper, we focus on an assessment for uncontrolled potential beam loss only. A *beam loss* occurs when accelerated particles become unstable in their trajectory around the LHC. There are several factors that may cause beam losses, such as collisions between proton beams and residual gas molecules in the LHC ring's vacuum chamber, magnets used to bend and focus the beam around the LHC being out of tolerance, and failure to extract the beam from the one of the two LHC rings during a beam dump. This may result in a loss of containment, or particle collisions with the LHC itself. As these particles have very high energy, beam loss can cause significant damage to the LHC if it exceeds acceptable levels.

The MPS is responsible for detecting beam loss and performing beam dumps before potentially damaging conditions are reached. A *beam permit signal* is used by components of the MPS to communicate whether conditions are ap-

propriate to continue operating the LHC: if the beam permit signal is present, the LHC may continue operating; otherwise, a beam dump is required. For this study a simplified MPS is considered to consist of four main components: the Beam Loss Monitoring System (BLMS), the Beam Interlock System (BIS), the Beam Dumping System (BDS) and the Safe Machine Parameters (SMP).

The Beam Loss Monitoring System (BLMS) is responsible for monitoring the LHC to measure the beam loss in all portions of the ring. The BLMS consists of ≈ 4000 monitors distributed around the two rings, each of which is monitoring a specific region of the LHC. Monitors are more densely distributed in critical regions of the LHC, such as around the critical components required to perform a beam dump. When non-nominal beam losses are detected, the BLMS signals the BIS to initiate a beam dump by withdrawing the beam permit. There is triple redundancy and error detection in the optical transmission to the BIS, and redundancy in other areas of the machine protection system. The MPS is intended to extract the beams within $400\mu s$ of the occurrence of a failure state to avoid potential damage to accelerator components. To satisfy this requirement, the BLMS is designed to detect and communicate beam losses to the BIS within $80\mu s$.

The Beam Dumping System (BDS) is responsible for extracting the beams from the LHC rings without damaging the system. It consists of a large graphite block designed to absorb extracted beams, dilution magnets that spread out particle clusters to reduce the energy density when they impact the sink, pulsed kicker magnets and continuously powered septa magnets to divert the circulating beams from the main LHC ring towards the sink, and moveable absorbers that protect the machine in the case of errors during a dump. For a beam dump to occur in a loss-free way, the BDS is engaged during an *abort gap*, i.e., a particle-free gap of $3\mu s$ in the ring.

The Beam Interlock System (BIS) determines whether the BDS should initiate a beam dump depending on the values assumed by a set of permit signals. The BIS processes these signals and sends a continuous signal to the BDS depending on the values received by the so called User systems. The BLMS is one of these User system (in total there are about 200 connections to the LHC BIS). The BIS sends a beam permit with value `true` to the BDS if it receives a beam permit with value `true` from all subsystems; otherwise, it sets the beam permit to the value `false`. It may take between $20\mu s$ and $120\mu s$ for the BIS to receive, process, and redirect a beam permit signal. Note that the BIS connects to all systems that may cause damage to the LHC. If any of these systems enter an unsafe state, the BIS will trigger a beam dump before the BLMS detects a problem. The BLMS is an additional protection measure on top of the BIS.

If the abort gap is filled with beam or synchronisation with the abort gap is lost, the BDS will engage[2], the BDS will engage anyway and perform an asynchronous dump. Asynchronous dumps can be dangerous as any particles that pass by the kicker magnets while they are only partially engaged will not be

[2]There are rare cases where a malfunction may cause particles to de-bunch and travel around the ring with a more uniform distribution.

diverted to the proper extraction trajectory. Absorbers are placed to protect the LHC in asynchronous dumps by covering the possible trajectories that particles could be sent on if they pass by kicker magnets that are not fully engaged. The extraction time for a worst-case scenario beam dump is $178\mu s$ since it may take up to $89\mu s$ for an abort gap to synchronize with a withdrawn beam permit, and another $89\mu s$ for all the particles to be extracted from the beam.

The *Safe Machine Parameters (SMP)* computes the values of a set of parameters from the operational conditions of the LHC and the super proton synchrotron via the two safe machine parameter controllers: one for the LHC and one for the super proton synchrotron. Once the SMPs are derived, they are communicated either via a broadcast protocol through the general machine timing channel, or via direct serial cable communications. The SMP system is used to ensure that only low intensity beam is injected into an LHC without beam and that certain inputs of the BIS can be masked with safe beam intensities. It also distributes critical parameters to many systems. An example is the beam energy which is used by the BLMS to calculated the thresholds for requesting a beam dump.

3 Assurance Cases

This section provides relevant background information on ACs using a fragment of an AC of the LHC MPS.

Eliminative Argumentation (EA) [8] is a graphical notation for AC development that extends the Goal Structuring Notation (GSN) [5]. We selected EA from other alternatives (GSN, CAE [27], SACM [28]) since it has a graphical structure and enables us to express reasons to doubt AC claims using defeater nodes. EA has been shown to be easy to learn, facilitates independent review and emphases the importance of doubt [5].

Figure 2 presents a fragment of an EA for the LHC Machine Protection System. The EA has nodes of different types:

- *Claim nodes* express affirmative statements asserting that a system satisfies one or more properties. For example, the node C0081 in Figure 2 is a claim asserting that the system's Target Dump External (TDE) dump block (i.e., a large graphite sink) will safely absorb beams from the LHC.

- *Defeater nodes* express doubts about the validity of an assurance argument. Defeaters are unique to EA, whereas the other AC node types presented in this section are also included in other notations such as GSN. A defeater can be decomposed into nodes showing how it has been mitigated, or it may be left as residual risk that threatens the argument's validity. For example, the defeater D0091 in Figure 2 asserts that claim C0081 will not hold if the TDE block absorbs a beam with high energy density that causes it to exceed its maximum safe temperature threshold. This defeater is decomposed into an argument showing that the hazardous scenario has been sufficiently mitigated.

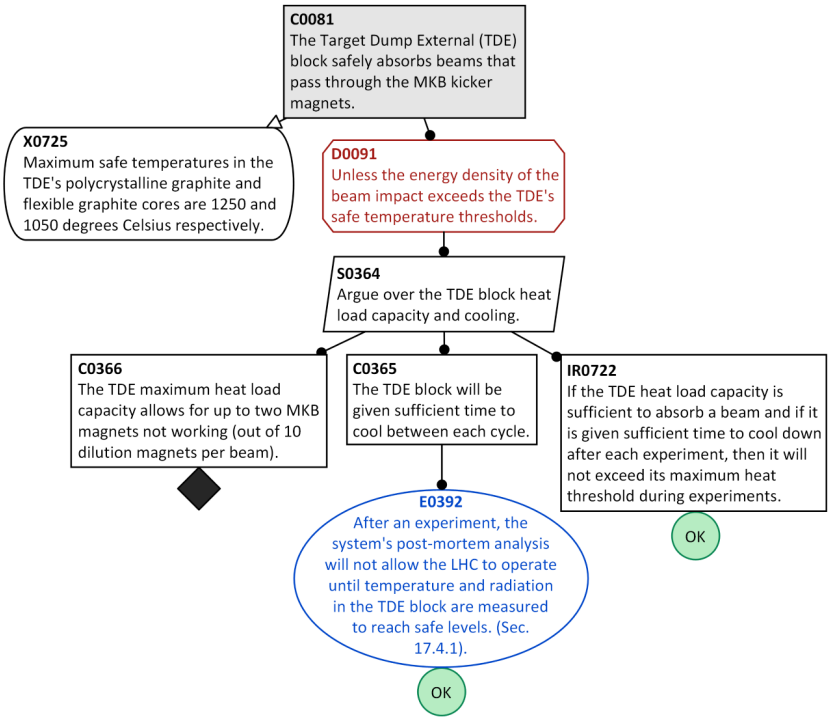

Figure 2: AC fragment for the LHC machine protection system

- *Strategy nodes* express reasoning steps used to decompose a claim into more refined subclaims. For example, node S0364 in Figure 2 decomposes defeater D0091 into subclaims related to the heat load capacity and cooling of the TDE block.

- *Context nodes* are used to provide background information or missing details that may be necessary to understand the argument. For example, context X0725 in Figure 2 provides information on the maximum heat loads for each type of core in the TDE block.

- *Inference rule nodes* are attached to strategy nodes, and are used to explain the rationale for why a strategy's child claims are sufficient to show that the parent claim holds. Inference rules may also be referred to as *justification* nodes (e.g., in GSN). For example, inference rule IR0722 in Figure 2 argues that if the TDE block's maximum heat load is sufficient to absorb a beam from the LHC and if it is given time to cool down each time it absorbs a beam, then it will never exceed its safe temperature threshold.

- *Assumption nodes* may be used to list conditions related to the system or its operational environment that are assumed to be true in the argument.

- *Evidence nodes* are used to support claims by directly connecting them to supporting evidence or documentation showing that the claim holds. For example, node E0392 in Figure 2 supports node C0365 by referencing a protocol in the MPS's post-mortem analysis in which it will never allow an experiment to commence when the TDE block is at a potentially unsafe temperature.

- *Residual risk nodes* are the residual uncertainties that cannot be completely eliminated by the argument, and thus they remain as potential sources of risk or uncertainty. These nodes may require further investigation or risk management strategies to mitigate their potential impact.

- *Undeveloped nodes* are aspects of the system that are not fully addressed or developed within the argument. These undeveloped nodes may require further investigation or analysis to fully understand their implications for the problem at hand. They represent areas of potential uncertainty or risk that may require further attention or consideration.

4 Methodology

Engineers typically develop an AC during the safety analysis following a precise methodology and development process. In this section, we describe the methodology we used to create the AC for the LHC MPS and empirically assess its benefits.

Our methodology (see Figure 3) follows two phases: *assurance case design* (1) and *feedback collection* (2). These correspond to our research questions: **RQ1** and **RQ2** from Section 1.

Assurance Case Design (1). Four authors of this paper designed the AC for the LHC. Three of them (Chris Ress, Mateo Delgato, Rolf Lippelt) are industry experts working for CSL. The team of CSL engineers has a combined experience in AC production of over 25 years. The other (Torin Viger) is a Ph.D. student at the University of Toronto with four years of research experience in AC development. From 2009 to 2012, CSL performed a series of technical audits for CERN covering particular aspects of the MPS. Knowledge gained from earlier work assisted the effort to develop this AC. Separate from these technical audits, CSL also collaborated with CERN and Cambridge University researchers on the formal verification of a critical component of the MPS [29].

The assurance case design proceeded as follows:

1.1 *Collection of Material.* The AC developers conducted a literature review of public documentation for the LHC MPS and identified eight relevant papers and four technical reports – see Table 1. These included engineering specifications for the system captured from various CERN internal documents and reviewed/discussed in project reports (i.e., [30, 31]), scientific papers (i.e., [32, 33, 34, 35, 36]) and a Ph.D. dissertation [37].

1.2 *Preliminary Analysis of the Material.* The AC developers studied the MPS documentation with the objective to better understand the MPS and its

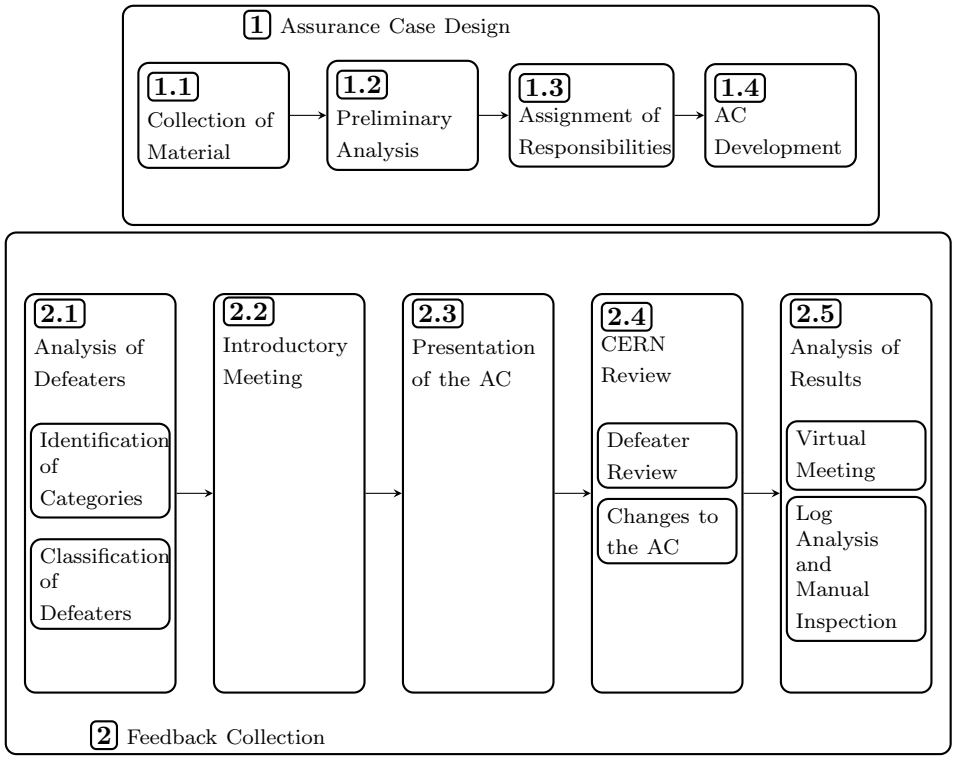

Figure 3: Methodology for creation and analysis of the MPS AC.

subsystems in order to determine how AC development tasks should be distributed among the team. Collection and analysis of this material took a combined period of two weeks.

1.3 *Assignment of Responsibilities.* An online session was performed to plan the AC development and define the tasks assigned to each member of the AC development team. Each member was responsible for developing a branch of the argument for one of the four main subsystems (i.e., the BLMS, BIS, BDS, and SMP) of the MPS (see Section 2).

1.4 *AC Development.* The AC design was performed using the collaborative web AC development platform Socrates [21] and took approximately seven weeks. Development was primarily done in parallel, with additional collaborative work sessions to review the argument and identify connections and interdependencies between its branches. These sessions lasted for around two hours and occurred twice weekly. Interdependencies between branches were mainly determined by considering defeaters in each argument branch and analyzing whether any other MPS subsystem performed a function that mitigated them.

The main argument creation phase was deemed complete once all branches of the argument were sufficiently decomposed so that they could be directly linked

Ref	Description
[30]	Operational report for the BIS
[31]	Description of the BLMS
[32]	Documentation of the LHC beam and power interlock systems
[33]	Technical overview of the BIS
[34]	Instruments and methods for measuring beam parameters
[35]	Statistics related to operation of the BDS
[36]	Upgraded BDS configuration and behaviour of beam dumps
[37]	Ph.D. dissertation describing the LHC MPS and its components

Table 1: Documents considered for the creation of the LHC AC.

to evidence from relevant CERN documents. The process left some residual defeaters where supporting evidence could not be identified from the publicly available documentation. The argument was reviewed internally by five additional engineers for consistency and quality for two weeks. We discuss the results of the assurance case design and provide the answer to **RQ1** in Section 6.1.

Feedback Collection (**2**). The evaluation of the AC proceeded as follows:

2.1 *Analysis of Defeaters.* We performed an internal review to analyze and classify the defeaters. The internal review had the following steps:

1. *Identification of Defeater Categories.* We identified a set of categories that classify how the doubts expressed by the defeater nodes were mitigated by the documents we analyzed. The categories were defined by two of the authors and discussed with the other members of the team. The categories capture the degree to which the defeaters and their corresponding mitigations were addressed by the publicly available documentation. Table 2 presents the categories we used to classify the defeaters. For example, the category NOT EXPLORED refers to defeaters for which corresponding hazard scenarios are not explored in CERN documents we analyzed.

2. *Classification of the Defeaters.* We analyzed each defeater and associated it with one of the defined categories. This process was conducted internally within the project, with suitable peer review, before being verified by CERN experts. The process involved a review of each defeater identified in the AC, determining whether this defeater was adequately addressed by the open source documentation available, to ensure that sufficient evidence was available to demonstrate that the scenario was satisfactorily mitigated.

2.2 *Introductory Meeting.* We provided a high-level presentation of the goal of our empirical study and outlined the goal of our evaluation to CERN experts.

2.3 *Presentation of the AC.* We presented the AC in detail to CERN experts to give them a general understanding of the argument we built. We then gave CERN experts access to the Socrates platform so that they could edit the AC themselves. We also shared with them the categorized list of defeaters.

Category	Description
Residual Risks	The risk captured by the doubt of the defeater is not mitigated.
Not Relevant	The doubt expressed by the defeater does not represent a significant risk for the system.
Not Explored	The hazard scenario is not explored in the documentation.
Some Understanding	The documents address the risk from the defeater without explicitly detailing it.
Understood	The documents detail the defeater. However, its mitigations are not simple or obvious.
Well Understood	The documents precisely detail the defeater and the corresponding mitigations.

Table 2: Classification of defeaters.

2.4 *CERN Review.* Three senior CERN experts reviewed and validated the argument against existing assessments and verified the evidence for identified claims. CERN engineers provided feedback in two different ways: by reviewing the categorized list of defeaters and by directly editing the argument.

2.5 *Analysis of Results.* We held a virtual meeting with CERN engineers and collected their feedback on the categorized list of defeaters, especially those classified as Not Explored, since they capture hazard scenarios not explored in the docu- mentation. To analyze the activity performed by CERN engineers in editing the argument, we collected and manually inspected the AC changes they made.

In the following sections, we first present the AC produced during *Assurance Case Design* (**1**), and then summarize the results from the *Feedback Collection* (**2**). Finally, during the *Feedback Analysis* review meeting, the project proposed the identification of *key performance indicators* (KPIs) from the AC. Specifically, it was proposed to use performance metrics within the AC to identify *leading* and *lagging* KPIs for the MPS. We then continued to identify subsequent KPIs for the MPS subsystems in the AC, noting areas of key performance metrics, residual or undeveloped nodes where monitoring of the system could aid mitigation of possible residual risks. Finally, the identified KPIs were shared with CERN experts for review. CERN suggested some minor typographical changes and noted that these KPIs corresponded to metrics already tracked by the post-mortem system (i.e., the system responsible for analyzing LHC data after an experiment completes).

5 The LHC Assurance Case

The AC for the LHC created by us during the *Assurance Case Design* (**1**) has 506 nodes. Table 3 gives a distribution of the number of nodes of each type in

Node Type	Number of Nodes
Claims	146
Evidence	70
Strategies	32
Inference Rules	29
Context	26
Assumptions	1
Defeaters	105
Residual	9
Undeveloped	15
Complete	73
Total	506

Table 3: Number of nodes of each type in the LHC AC.

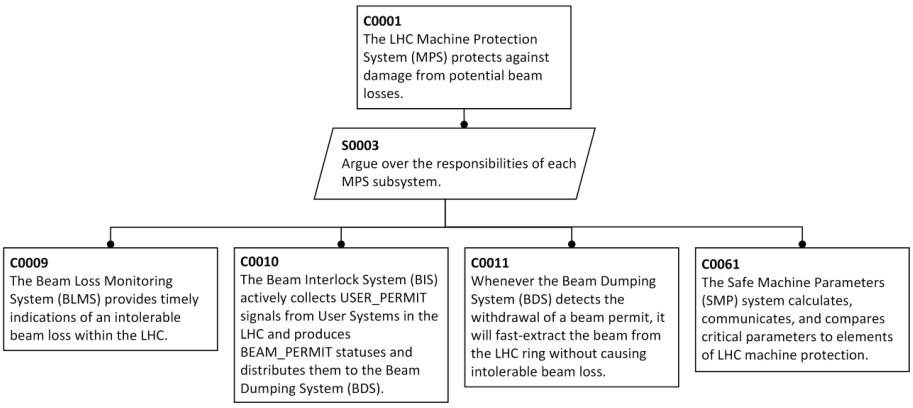

Figure 4: Overview of the high-level structure of the argument for the LHC.

the argument. Of these nodes, 105 are defeaters representing sources of doubt in the system. While most defeaters are mitigated by evidence, nine are left as residual risks within the EA.

Figure 4 presents an overview of the high-level structure of the argument. The top-level claim C0001 asserts that "The LHC Machine Protection System (MPS) protects against damage from potential beam losses" and is recursively decomposed into sub-claims, evidence, and other EA nodes. Specifically, the claim C0001 is decomposed using a strategy that splits it into four subclaims, one for each of the subsystems (i.e., BLMS, BDS, BIS, and SMP - see Section 2).

Each subclaim describes how the corresponding subsystem protects against damage from potential beam losses. For example, Figure 5 presents a fragment of the AC argument associated with a subclaim for the BIS subsystem. Claim C0030 argues that "The BIS will transmit loss of the beam permit to the BDS in

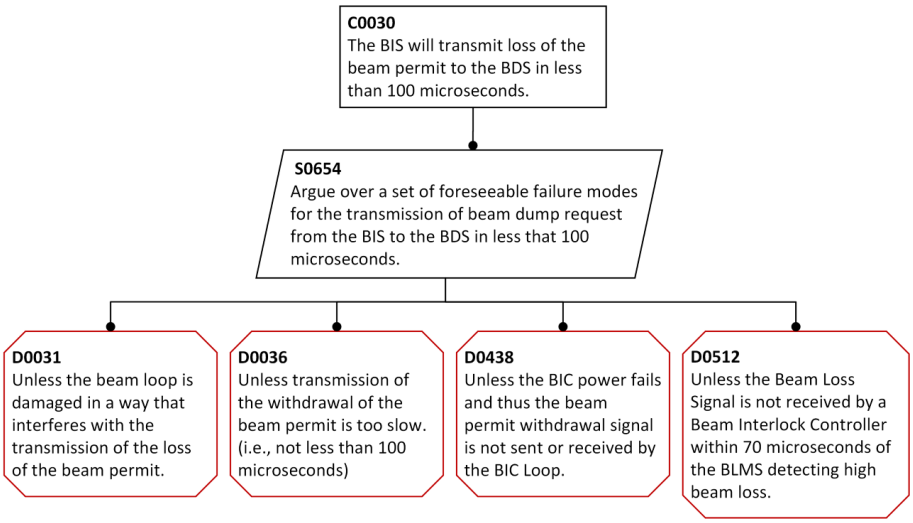

Figure 5: Fragment of the AC that refers to the BIS subsystem.

less than 100 microseconds". The strategy S0654 decomposes claim C0030 into four branches based on the foreseeable failure modes that could block, delay or otherwise interfere with the transmission of a beam dump request to the BDS. These failure modes are recorded explicitly by the defeater nodes D0031, D0036, D0438, and D0512. For example, the defeater D0031 argues that the BIS will transmit the loss of the beam permit to the BDS in less than 100 microseconds "Unless the beam permit loop is damaged in a way that interferes with the transmission of the loss of the beam permit".

EA expands defeater nodes into subclaims that terminate with evidence nodes and describe how the risks associated with the doubts introduced by the defeaters nodes are mitigated. For example, to mitigate the risks introduced by defeater D0031, the usage of redundant beam permit loops and the fail-safe design of the mechanism responsible for transmitting beam permits is considered. The evidence node E0534 of the AC asserts that "in the event of one or all transmission lines being damaged, the beam permit loop will have no 10MHz signal or noise and subsequently results in the request for a beam dump", which shows how the design of the beam permit loop mitigates risks from potential damages to the system's transmission lines.

6 Evaluation

As discussed in Section 4, after designing the AC (*Assurance Case Design* — 1), we empirically evaluated our results in collaboration with CERN experts (*Feedback Collection* — 2). This section presents the answers to our research questions: Section 6.1 assesses the difficulty of developing an AC for the system

Engineer Level	Activities	Days Booked
Junior A (full time)	AC creation	28.0
Junior B (full time)	AC creation	28.0
Senior (full time)	AC creation	31.4
Senior (part time)	Review, verification, validation	4.5
Total		91.9

Table 4: Total number of workdays spent developing the MPS AC by each engineer.

(**RQ1**), and Section 6.2 evaluates the usefulness of the AC (**RQ2**).

6.1 AC Development Effort - RQ1

Our AC consists of 506 nodes, which corresponds to a medium-sized artifact according to CSL engineers. Considering a recent paper that reports on the application of EA to seven different software-intensive systems [1], our AC is larger than six of the seven assurance cases; the size of the remaining assurance case (513 nodes) is comparable to ours.

Developing the AC required 2543 changes (additions, modifications, and removals of nodes) and took 91.9 workdays. Table 4 shows the total number of workdays spent developing the AC, with rows representing the time spent and activities performed by engineers at different levels. This metric includes time spent studying the MPS and its documentation. Therefore, the development time could be reduced if the AC were developed by engineers already familiar with the system.

Compared to the development time of the LHC (approximately 10 years [24, 25]), this AC development time is negligible. The AC building cost is also negligible when compared to the investment required to build the LHC MPS itself ($\approx$ 200 million USD).

RQ1 - Development Effort

The time (91.9 days) required to develop a medium-size AC for the LHC MPS is significantly smaller than the one needed for the system development (10 years). The cost estimated for developing the AC is also negligible compared to the cost of building the LHC MPS ($\approx$ 200 million USD).

6.2 Identifying Risk Scenarios - RQ2

To assess the usefulness of developing an AC for the MPS, we evaluate (a) whether the AC development enabled us to identify defeaters that were not explicitly detailed in the publicly available documentation and (b) the precision and recall for the identification of the defeaters of the MPS.

Table 5 (column *Analysis of Defeaters*) reports the number of defeaters that were classified in each of the categories from Table 2 during the *Analysis of Defeaters* phase (**2.1**) detailed in Section 4. Based on our initial classification, 24, 50, and 13 defeaters were classified as SOME UNDERSTANDING, UNDERSTOOD, and WELL UNDERSTOOD, respectively. Among the remaining 18 defeaters, nine were classified as RESIDUAL UNIDENTIFIED RISKS, three as NOT RELEVANT, and six as NOT EXPLORED. These defeaters were analyzed during the *Defeater Review* phase **2.4**. Table 5 (column *Defeater Review*) reports the number of defeaters that belongs to each category after *Defeater Review* phase. It also illustrates within brackets how this number if computed starting from the number of defeaters present in that category after the *Analysis of Defeaters*. For example, in the UNDERSTOOD row, 57 corresponds to $50 - 10 + 17$" which indicates that 10 and 17 defeaters were respectively removed and added to the 50 defeaters from the *Analysis of Defeaters*.

6.2.1 CERN Assurance Case Review

Recall that during the *Defeater Review* phase **2.4** described in Section 4, CERN experts directly reviewed and edited the safety argument. Their review raised a total of 20 technical and editorial comments in the AC, which provided extended descriptions and clarifying details related to the design and functionality of the MPS. As an example, CERN clarified that if the BIS has no power, the beam permit loop signal should have no signal or noise, which will be interpreted as dump request by the dumping system (claim C0442).
Reviewing these comments resulted in the following changes:

(a) minor modifications of the AC claims to more accurately reflect the design and functionality of the MPS,

(b) creation of two additional evidence nodes,

(c) revision of three nodes from claims to defeaters,

(d) creation of one new defeater for a previously unexplored branch of the BLMS argument focused on the potential for beam energy to not be processed correctly by the BLMS,

(e) one defeater being marked as UNDEVELOPED in the AC, which focused on a potential scenario involving a sudden loss in power to the BDS and available documentation,

(f) addition of two context nodes to the argument to expand on information provided by CERN experts on the operation of the BLMS.

6.2.2 Defeater Review

In total, 27 defeaters were reclassified following the Defeater Review by CERN experts. The impact on each category of defeaters was as follows:

Category	Analysis of Defeaters 2.1	Defeater Review 2.4
Residual Unidentified Risks	**9**	**7** $(9-2+0)$
Not Explored	**6**	**3** $(6-3+0)$
Some Understanding	**24**	**14** $(24-10+0)$
Understood	**50**	**57** $(50-10+17)$
Well Understood	**13**	**23** $(13-0+10)$
Not Relevant	**3**	**1** $(3-2+0)$
Total	**105**	**105** $(105-27+27)$

Table 5: Categorization and number of defeaters.

- Residual Unidentified Risks. CERN experts confirmed seven of the defeaters in this category but noted that the remaining two were mitigated by additional publicly available information about the MPS which they described during a review meeting.
  Resulting Changes: Two defeaters were removed from the Residual Unidentified Risks category and moved to the Understood category.

- Not Relevant. After consulting the additional references identified by CERN experts (research papers and supporting documentation from publicly accessible CERN resources), information was found related to the mitigation of two defeaters initially classified as Not Relevant category. The remaining Not Relevant defeater was confirmed by CERN experts as not a relevant risk to the LHC MPS.
  Resulting Changes: Two defeaters were removed from the Not Relevant category and moved to the Understood category.

- Not Explored. CERN experts explained the measures used to mitigate the risk associated with three defeaters that were classified into this category. They then confirmed the relevance of the remaining three defeaters as well as absence of the mitigation measures for them in the publicly available documents we considered.
  Resulting Changes: Three defeaters were removed from the Not Explored category and moved to the Understood category.

- Some Understanding. CERN experts explained how the measures reported in the documents we analyzed mitigated the risk associated with ten defeaters initially categorized under Some Understanding. They then confirmed the relevance of the remaining fourteen defeaters in this category, for which we could not find thorough mitigation measures in the documentation we analyzed.
  Resulting Changes: Ten defeaters were removed from the Some Understanding category and moved to the Understood category.

- UNDERSTOOD and WELL UNDERSTOOD. CERN experts confirmed the relevance of all defeaters in these categories and provided additional information which enabled us to improve the AC and expand on the mitigation measures for ten of the defeaters initially classified as UNDERSTOOD.
  Resulting Changes: Ten defeaters were removed from the UNDERSTOOD category and moved to the WELL UNDERSTOOD category.

The results from Table 5 (column **Defeater Review**) show that only one defeater was classified as NOT RELEVANT after the *Defeater Review*. Among the remaining 104 defeaters, $\approx 90\%$ ($95 = 14 + 57 + 23$) were classified as SOME UNDERSTANDING, UNDERSTOOD, or WELL UNDERSTOOD) and were known by CERN experts. This result is expected since the AC development was based on an existing operating system and publicly accessible online documentation containing information reported by CERN experts. Of the remaining $\approx 10\%$, seven were classified as RESIDUAL UNIDENTIFIED RISKS and three were classified as NOT EXPLORED. These defeaters were confirmed to be relevant by CERN experts, and not explicitly detailed in the publicly available documentation. This result is significant: it shows the usefulness of AC development in identifying real defeaters relevant to a system, and the level of precision of the approach. Therefore, we conclude that development of an AC using EA is useful to accurately identify doubts in a system.

To calculate the precision and recall of our identification of defeaters, we defined True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) as follows:

- TPs are defeaters that we classified as relevant during the *Analysis of Defeaters* phase (**2.1**) (i.e., those in the RESIDUAL UNIDENTIFIED RISKS, NOT EXPLORED, SOME UNDERSTANDING, UNDERSTOOD and WELL UNDERSTOOD categories) which were confirmed to be relevant by CERN experts during the *Defeater Review* phase (**2.4**). All defeaters in these categories were confirmed to be relevant, therefore TP = 102 ($9 + 6 + 24 + 50 + 13$).

- TNs are defeaters which we classified as NOT RELEVANT that were confirmed to be NOT RELEVANT by CERN experts after the *Defeater Review*. We have TN = 1, as only one of the three defeaters identified as NOT RELEVANT was confirmed by CERN to be not relevant.

- FPs are defeaters which we categorized as relevant defeaters (i.e., in any category except NOT RELEVANT), but that CERN identified as NOT RELEVANT. We had no false positives, therefore FP = 0.

- FNs are nodes which we did not categorize as relevant defeaters, but which CERN identified to be relevant. We have two FNs from the *Defeater Review* phase (**2.4**), as two defeaters we categorized as NOT RELEVANT were found to be relevant by CERN experts. Additionally, as noted in Section 6.2.1, CERN changed three nodes from claims to defeaters and added

an entirely new defeater during their review of the AC itself. Therefore, FN = 6 (2 + 3 + 1).

The precision of our defeater identification process is $^{TP}/_{(TP+FP)} = 1$. Our recall is $^{TP}/_{(TP+FN)} = {}^{102}/_{(102+6)} = 0.94$.

RQ2 - Usefulness

The answer to RQ2 is that AC development identified 10 defeaters that were not detailed in the documentation we considered. These defeaters were confirmed to be valid by CERN experts. The precision of the manual development process we used to identify defeaters is 1, the recall is 0.94.

7 KPI Discussion and Threats to Validity

During the interactions with CERN experts, especially during the *Defeater Review* phase, parts of our conversation led to the discussion and definition of Key Performance Indicators (KPIs). KPIs are quantifiable and detectable measurements of events whose rate of occurrence can be used to gauge the performance of a system. Identifying KPIs helps develop and manage a system, as they give concrete measurements that can be analyzed to determine whether a system or its subsystems are functioning correctly. Specifically, two types of measurements were discussed with CERN experts: *lagging indicators* and *leading indicators*.

- *Lagging indicators* track the occurrence rate of hazards and loss events, such as crashes, injuries, and fatalities.
- *Leading indicators* measure the occurrence of events that, while not themselves harmful, are expected to precede or indicate the potential for future failures in performance.

Lagging indicators can provide useful information, however they are heavily dependent on collecting data post a loss event. Thus, the use of lagging indicators alone to measure a system's performance is not ideal. Leading indicators can include near-misses, equipment malfunctions and faults/failures.

The two indicators are complementary: lagging indicators detect the presence of hazards and loss events when they occur, whereas leading indicators can process larger amounts of data and preemptively detect problems in performance.

For the MPS, we identified 21 KPIs (15 lagging and 6 leading indicators). They were identified by analyzing claims and defeaters related to measurable aspects of the system's performance, and events which can be monitored via the CERN MPS. The KPI were mainly derived from key *Claims*, *Residuals* and *Undeveloped* nodes, where monitoring the system could provide data to help measure and mitigate potential residual risks. As an example, the KPI *BLMS-KP2* (a lagging indicator) states that "*A failure of a single, or multiple, BLMS*

detector(s) would be reported to the control room, and thus a user permit not granted for operations" is derived from Claim C0140 - "*Detector failures will be identified and reported to the central control room*", within the AC.

Our experience shows that the EA has enabled these KPIs to be easily identified from the wider argument (approximately 500 nodes), due to the structure of the argument and the ease of identifying residual risks, claims and defeaters. These KPIs were shared, reviewed and discussed with CERN experts. CERN experts also confirmed that these KPIs are reasonable and largely addressed by their existing postmortem system. The MPS postmorten system will identify items such as missing redundancy between systems or within a system. Operations can only continue after redundancy has been re-established. The fact that the identified KPIs have been considered by the postmortem system of the LHC confirms that they are reasonable, demonstrating that AC development and the use of EA can help identify KPIs that mitigate residual risks in a system. In addition, the discussion related to some KPIs aided in identification of potentially unrealized loss events. We plan to more rigorously evaluate the usefulness of AC development in identifying KPIs as future work.

Threats to validity. The methodology used to collect the feedback from CERN, i.e., the defeater review and the changes to the AC via Socrates, is an internal threat to validity of our results. To mitigate this, we used two methods to collect feedback: the discussion of the defeater review and the analysis of the changes performed via Socrates on the AC. Another internal threat to validity is team composition and experience of team members. To mitigate this, we created a team composed of members with a mix of experience, including industry and academia.

The analysis of a single case study threatens the external validity of our results: the conclusions of our empirical investigation may differ for different case studies and systems. However, the fact that the MPS is a large safety-critical system and the involvement of experts in AC development from CSL mitigates this threat: a large safety-critical system is likely to share problems that are also encountered in other safety-critical systems, and the presence of CSL engineers ensured that the creation of the AC was grounded on previous experience. Usage of public documentation for the AC development and analysis is another external threat to validity since for other systems (still under development), this documentation may not be available, or might be incomplete. Therefore, when systems are not as mature as the one we analyzed, we expect precision and recall to be lower. Further, an AC is normally created during the system design and hence evolves over time, this would also likely affect the precision and recall.

Finally, the metrics used to measure the usefulness of the defeaters produced by the manual development process (TP, FP, TN, and FN) threaten our results' construction validity since they influence how well they represent or reflect a concept that is not directly measurable.

8 Related Work

There is significant research and industry interest in approaches that support AC development, including new notations [27, 8, 5], methodologies [9, 38, 39, 40], argument templates [41, 42, 43], domain-specific techniques [44, 14], and tools for formal reasoning over ACs [45, 11, 46, 10]. However, these techniques are often not assessed or only assessed over small showcase examples. For example, the work introducing EA [8] demonstrated its usefulness on three artificial examples with ≈30 nodes. We only identified a handful of works that analyze the implications and effectiveness of AC development techniques in practice. We summarize these works below.

Diemert and Joyce [1] discussed their experiences and lessons learned from using EA to create ACs for seven different industrial systems, including four automotive (149, 257, 484 and 513 nodes), two rail (40 and 95 nodes) and one industrial control (14 nodes) ACs. The authors report that EA increases confidence in ACs and helps with independent safety assessment; however, this insight is only supported by practitioner experience, as neither the ACs nor information on their development processes are publicly available. In addition, the lessons learned are presented informally, without any systematic empirical analysis of the developed assurance cases.

Sujan et al. [47] reviewed AC practices in six UK industries (automotive, civil aviation, defense, nuclear, petrochemical, and railway). Their analysis compares safety requirements and regulations from the healthcare domain and concludes that ACs may lead to more structured healthcare safety management practices; however, the authors note that further research studies are required to provide empirical evidence of the contribution of ACs to safety management.

Graydon and Holloway [48] reviewed twelve candidate proposals (in fifteen papers) for assessing confidence in ACs. Their goal was to assess the capabilities of the proposed techniques for quantifying confidence in assurance arguments. The authors searched for counterexamples to detect techniques that can produce implausible results. Where possible, the authors prioritized counterexamples that are variants of the original examples. For three out of twelve techniques, the authors reported some counterexamples showing that the technique outputs are untrustworthy.

Nair et al. [49] used evidential reasoning [50] to measure and aggregate confidence in ACs. Evidential reasoning requires safety analysts to attach confidence levels to evidence nodes to denote the evidence's trustworthiness and aggregates these values to derive a quantified confidence measurement in the AC. The authors evaluated their framework through a survey involving 21 participants with over two years of experience in safety assurance. However, the authors did not assess their framework in any case study.

Cyra et al. [51] proposed visual assessment to analyze an argument that relies on the Dempster-Shafer theory of evidence [52]. Dempster–Shafer's theory of evidence requires associating each evidence node with a value within the interval $[0, 1]$. Strategies are linked to functions that enable computing the confidence of the different claims from the confidence of evidence nodes. The authors analyzed

whether the functions associated with the different strategies are plausible. This analysis was conducted via an experiment involving 31 students from the last year of the Master's degree in information technologies. The results show that the accuracy of the aggregation rules is similar to the consistency from the answers of the participants.

Unlike these works, this paper empirically analyzed and assessed the benefits of AC developed using EA on a significant industrial example. We made our AC and results publicly available.

9 Conclusion

In this paper, we empirically evaluated the effort required for developing an assurance case for a large representative safety-critical system and assessed its usefulness. Our results show that the cost and time required to develop this AC are negligible compared to the system cost, the AC helped identify risk scenarios not explicitly considered in the documentation considered for the AC development, and the manual development was effective in identifying defeaters. Based on our knowledge, this is the first study that empirically assesses the usefulness of the AC and involved two teams of experts from the industries, one helping in the construction of the AC (experts from CSL) and one for its evaluation (experts from CERN). Therefore, our results will be relevant to both researchers, who need industrial case studies to assess their research solutions, and practitioners, who can rely on empirical results confirming the usefulness of AC development.

Our AC, made publicly available, is a significant contribution per se. Further, the fact that the AC and all the supporting documentation is openly available makes it even more useful to researchers looking to benchmark their methods on a realistic example. Based on our knowledge, this is the first (and only) AC publicly available that includes more than 100 nodes, developed in collaboration with safety experts with an extensive experience in safety analysis and revised by domain experts. This AC is also a significant contribution for the safety community: it will provide a reference case study that can be used as a benchmark for future works.

The development of a large and representative exemplar AC is part of our long-term vision of supporting AC developers by using AC development information as data [53]. We believe that by monitoring the AC development activities and treating assurance cases as data, we can learn suggestions to help safety engineers improve their AC. For this reason, the activity of the AC developers was monitored, and their activities during the creation of the AC were logged. We plan to treat these AC development activities as data and to propose techniques that can learn from these data and provide suggestions that can improve the design of the AC. For example, recommendations may help identify safety interdependencies between the components for the MPS overlooked during the AC design.

Acknowledgment

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) [funding reference numbers RGPIN-2022-04622, DGECR-2022-0040, RGPIN-2015-06366].

References

[1] S. Diemert, J. Joyce, Eliminative Argumentation for Arguing System Safety - A Practitioner's Experience, in: Proceedings of International Systems Conference, IEEE, 2020, pp. 1–7.

[2] ISO/IEC JTC 1/SC 7 Software and systems engineering, Systems and software engineering — Systems and software assurance — Part 2: Assurance case, https://www.iso.org/standard/52926.html (2011).

[3] R. Palin, D. Ward, I. Habli, R. Rivett, ISO 26262 Safety Cases: Compliance and Assurance, in: International Conference on System Safety, IET, 2011.

[4] BS EN 50126-1:2017 (2011).

[5] GSN Working Group, GSN Community Standard Version 2, http://www.goalstructuringnotation.info/ (2011).

[6] R. Boomfield, P. Bishop, Safety and Assurance Cases: Past, Present and Possible Future – an Adelard Perspective, in: Safety-Critical Systems Symposium, Springer, 2010.

[7] J. Rushby, The Interpretation and Evaluation of Assurance Cases, Comp. Science Laboratory, SRI International, Tech. Rep. SRI-CSL-15-01 (2015).

[8] J. Goodenough, C. Weinstock, A. Klein, Eliminative Argumentation: A Basis for Arguing Confidence in System Properties, Tech. Rep. CMU/SEI-2015-TR-005, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA (2015).
URL http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=434805

[9] M. Mohamad, A. Åström, O. Askerdal, J. Borg, R. Scandariato, Security Assurance Cases for Road Vehicles: An Industry Perspective, in: International Conference on Availability, Reliability and Security, ACM, 2020.

[10] T. Viger, L. Murphy, A. Di Sandro, C. Menghi, R. Shahin, M. Chechik, The ForeMoSt Approach To Building Valid Model-Based Safety Arguments, Software and Systems Modeling (2022) 1–22doi:https://doi.org/10.1007/s10270-022-01063-4.

[11] N. L. S. Fung, S. Kokaly, A. Di Sandro, M. Chechik, Assurance Case Property Checking with MMINT-A and OCL, in: Recent Trends and Advances in Model Based Systems Engineering, Springer, 2022, pp. 351–360.

[12] T. Viger, L. Murphy, A. Di Sandro, R. Shahin, M. Chechik, A Lean Approach to Building Valid Model-Based Safety Arguments, in: ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, 2021.

[13] J. Cheng, M. Goodrum, R. Metoyer, J. Cleland-Huang, How Do Practitioners Perceive Assurance Cases in Safety-Critical Software Systems?, in: International Workshop on Cooperative and Human Aspects of Software Engineering, ACM, 2018, p. 57–60.

[14] O. Jaradat, I. Sljivo, I. Habli, R. Hawkins, Challenges of Safety Assurance for Industry 4.0, in: European Dependable Computing Conference, IEEE, 2017.

[15] R. Schmidt, R. Assmann, E. Carlier, B. Dehning, R. Denz, B. Goddard, E. Holzer, V. Kain, B. Puccio, B. Todd, et al., Protection of the CERN Large Hadron Collider, New Journal of Physics 8 (11) (2006) 290.

[16] The Large Hadron Collider, `https://home.cern/science/accelerators/large-hadron-collider` (04 2022 [Online]).

[17] R. Andersson, E. Adli, E. Bargalló, A. Nordt, Machine Protection Systems and their Impact on Beam Availability and Accelerator Reliability, in: International Particle Accelerator Conference, 2015, p. MOPTY044. `doi:10.18429/JACoW-IPAC2015-MOPTY044`.

[18] E. B. Holzer, B. Dehning, E. Effnger, J. Emery, V. Grishin, C. Hajdu, S. Jackson, C. Kurfuerst, A. Marsili, M. Misiowiec, M. Nagel, E. N. D. Busto, A. Nordt, C. Roderick, M. Sapinski, C. Zamantzas, Beam Loss Monitoring for LHC Machine Protection, Physics Procedia 37 (2012) 2055–2062, international Conference on Technology and Instrumentation in Particle Physics. `doi:https://doi.org/10.1016/j.phpro.2012.04.110`.

[19] B. Dehning, LHC Machine Protection, in: Beam Instrumentation Workshop, 2008.

[20] J. B. Goodenough, C. B. Weinstock, A. Z. Klein, Eliminative induction: A basis for arguing system confidence, in: International Conference on Software Engineering, IEEE, 2013, pp. 1161–1164.

[21] Socrates Assurance Case Editor , `https://safetycasepro.com/welcome` (04 2022 [Online]).

[22] Critical Systems Labs, `https://www.criticalsystemslabs.com/` (04 2022 [Online]).

[23] C. Rees, T. Viger, M. Delgado, R. Lippelt, J. Joyce, S. Diemert, C. Menghi, M. Chechik, J. Uythoven, M. Zerlauth, L. Felsberger, CERN LHC MPS Assurance Case (2023).
URL `https://cds.cern.ch/record/2854725`

[24] Large Hadron Collider, https://en.wikipedia.org/wiki/Large_Hadron_Collider (04 2022 [Online]).

[25] R. Highfield, Large Hadron Collider: Thirteen ways to change the world, The Daily Telegraph (2008) 10–10.

[26] CERN Website Images, https://home.web.cern.ch/about (01 2023 [Online]).

[27] S. Gan, J. Ryan, Claims, arguments, evidence, International Nuclear Information System (INIS) 52 (2019).

[28] Y. Nemouchi, S. Foster, M. Gleirscher, T. Kelly, Isabelle/SACM: Computer-Assisted Assurance Cases with Integrated Formal Methods, in: Integrated Formal Methods, Springer, 2019, pp. 379–398.

[29] N. Ghafari, R. Kumar, J. Joyce, B. Dehning, C. Zamantzas, Formal verification of real-time data processing of the LHC beam loss monitoring system: a case study, in: Formal Methods for Industrial Critical Systems, Springer, 2011, pp. 212–227.

[30] B. Puccio, I. Romera Ramirez, B. Todd, M. Kwiatkowski, A. Castañeda Serra, The CERN beam interlock system: principle and operational experience, Tech. rep., European Organization for Nuclear Research (CERN) (2010).

[31] S. Gilardoni, E. Effinger, J. Gil-Flores, U. Wienands, S. Aumon, Beam loss monitors comparison at the CERN proton synchrotron, Tech. rep., European Organization for Nuclear Research (CERN) (2011).

[32] F. Bordry, R. Schmidt, K. Mess, F. Rodríguez-Mateos, B. Puccio, R. Denz, Machine protection for the LHC: Architecture of the beam and powering interlock systems, Tech. rep., European Laboratory for Particle Physics, European Organization for Nuclear Research (CERN) (2001).

[33] B. Puccio, R. Schmidt, J. Wenninger, et al., Beam interlocking strategy between the LHC and its injector, in: International Conference on Accelerator and Large Experimental Physics Control Systems, 2005, pp. 10–14.

[34] M. Gasior, R. Jones, T. Lefevre, H. Schmickler, K. Wittenburg, Introduction to beam instrumentation and diagnostics, arXiv preprint arXiv:1601.04907 (2016).

[35] E. Carlier, C. Bracco, C. Wiesner, L. Ducimetiere, N. Magnin, J. Uythoven, V. Senaj, LHC beam dumping system, Evian Workshop on LHC beam operation (2017) 215–220.

[36] J. Maestre, C. Torregrosa, K. Kershaw, C. Bracco, T. Coiffet, M. Ferrari, R. F. Ximenes, S. Gilardoni, D. Grenier, A. Lechner, et al., Design and behaviour of the Large Hadron Collider external beam dumps capable of receiving 539 MJ/dump, Journal of Instrumentation 16 (11) (2021) P11019.

[37] S. C. Wagner, LHC machine protection system: method for balancing machine safety and beam availability, Ph.D. thesis, ETH Zurich (2010).

[38] M. A. Javed, F. U. Muram, H. Hansson, S. Punnekkat, H. Thane, Towards Dynamic Safety Assurance for Industry 4.0, Journal of Systems Architecture 114 (2021) 101914.

[39] T. Viger, R. Salay, G. Selim, M. Chechik, Just enough formality in assurance argument structures, in: International Conference on Computer Safety, Reliability, and Security, Springer, 2020, pp. 34–49.

[40] J. Inge, The safety case, its development and use in the United Kingdom, in: Equipment Safety Assurance Symposium 2007, 2007.

[41] T. Chowdhury, C.-W. Lin, B. Kim, M. Lawford, S. Shiraishi, A. Wassyng, Principles for Systematic Development of an Assurance Case Template from ISO 26262, in: Proceedings of 2017 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), 2017, pp. 69–72. doi:10.1109/ISSREW.2017.14.

[42] M. Szczygielska, A. Jarzebowicz, Assurance case patterns on-line catalogue, in: Advances in Dependability Engineering of Complex Systems: International Conference on Dependability and Complex Systems, Springer, 2018, pp. 407–417.

[43] S. Yamamoto, Y. Matsuno, An evaluation of argument patterns to reduce pitfalls of applying assurance case, in: International Workshop on Assurance Cases for Software-Intensive Systems, IEEE, 2013, pp. 12–17.

[44] T. Myklebust, T. Stålhane, G. Hanssen, Agile safety case and DevOps for the automotive industry, in: European Safety and Reliability Conference and the Probabilistic Safety Assessment and Management Conference, 2020.

[45] P. Arcaini, A. Bombarda, S. Bonfanti, A. Gargantini, E. Riccobene, P. Scandurra, The ASMETA approach to safety assurance of software systems, in: Logic, Computation and Rigorous Methods: Essays Dedicated to Egon Börger on the Occasion of His 75th Birthday, Springer, 2021, pp. 215–238.

[46] E. Denney, G. Pai, Tool support for assurance case development, Automated Software Engineering 25 (3) (2018) 435–499.

[47] M. A. Sujan, I. Habli, T. P. Kelly, S. Pozzi, C. W. Johnson, Should healthcare providers do safety cases? Lessons from a cross-industry review of safety case practices, J. Safety Science 84 (2016) 181–189.

[48] P. J. Graydon, C. M. Holloway, An investigation of proposed techniques for quantifying confidence in assurance arguments, Safety science 92 (2017) 53–65.

[49] S. Nair, N. Walkinshaw, T. Kelly, J. L. de la Vara, An evidential reasoning approach for assessing confidence in safety evidence, in: International Symposium on Software Reliability Engineering, IEEE, 2015, pp. 541–552.

[50] J.-B. Yang, D.-L. Xu, On the evidential reasoning algorithm for multiple attribute decision analysis under uncertainty, Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 32 (3) (2002) 289–304.

[51] L. Cyra, J. Górski, Support for argument structures review and assessment, Reliability Engineering & System Safety 96 (1) (2011) 26–37. doi:10.1016/j.ress.2010.06.027.

[52] G. Shafer, Dempster-shafer theory, Encyclopedia of artificial intelligence 1 (1992) 330–331.

[53] C. Menghi, T. Viger, A. Di Sandro, C. Rees, J. Joyce, M. Chechik, Assurance Case Development as Data: A Manifesto, in: International Conference on Software Engineering: New Ideas and Emerging Results, 2023.

Declaration of interests

☐The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☒The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Torin Viger reports financial support was provided by Natural Sciences and Engineering Research Council of Canada.

A PDF representation of the full assurance case described in this work (also available online, as cited in the paper)