

# Strengths and Limitations of Existing STAMP Tools for Analyzing System of Systems Risks

**Marjorie Nawila Pettersson\***

Mälardalen University  
marjorie.pettersson@mdu.se

**Jakob Axelsson**

Mälardalen University  
jakob.axelsson@mdu.se

**Anna Johansson**

Mälardalen University  
anna.johansson@mdu.se

**Pontus Svenson**

Research Institutes of Sweden  
pontus.svenson@ri.se

## ABSTRACT

The System Theoretic Accident Model and Processes (STAMP) considers safety as a dynamic problem by focusing on interactions among system components rather than on component failure. However, analyzing complex system risks with STAMP can be challenging due to manual processes. Therefore, software tools supporting STAMP analysis have been developed. An evaluation of five tools (ASTAH, STPA, Capella, XSTAMPP, STPA Master Lite, and Visio Pro) was conducted, focusing on systems of systems (SoS) risk analysis. While all tools demonstrated compliance with the STAMP risk analysis procedure, with some tools offering additional features, a shortcoming in scalability limits the support for risk analysis of SoS. This makes risk analysis with the available STAMP tools challenging for SoS. Recommended enhancements include mechanisms to address SoS characteristics, scalability, collaboration, and improved usability. The study contributes to research on SoS risk analysis and provides direction for advancement in risk analysis tool support.

## Keywords

STAMP Tools, STAMP, Usage, Risk Analysis, Systems of Systems, SoS

## INTRODUCTION

The System Theoretic Accident Model and Processes (STAMP) (Leveson, 2004) is an approach that considers safety as a dynamic problem by focusing on interactions between system components and behavior. This is a shift from viewing safety as failure of component in a system which is a central in traditional safety system approaches (Leveson, 2004). The STAMP approach aims to enforce constraints on the entire system's behavior and interactions (Leveson, 2004). This makes STAMP highly suitable for analyzing risk in dynamic complex systems such as systems of systems (SoS). In such an SoS, the independent elements of the system (constituent systems) collaborate to provide a unique capability that none of the constituent systems can achieve by itself (Maier, 1998).

Originally used in the aviation domain, the STAMP methodological framework has been applied to other fields such as energy, health, and automotive (Zhang et al., 2022). Moreover, there has been an increase in integration of STAMP with other approaches in risk analysis (Zhang et al., 2022), which underscores its popularity. This increased use means STAMP could be a valuable framework for broader identifications of risk, thereby enhancing safety in complex systems domains such as SoS.

Despite its strengths, STAMP's limited tool support makes the risk analysis process challenging. Sun et al. (2022) highlight that STAMP requires expertise to conduct a comprehensive risk analysis. Further, techniques and tools can be developed based on STAMP to make analysis easier (Zhang et al., 2022). Therefore, it is important to evaluate the effectiveness, usability, and scalability of existing tools that support the STAMP approach, identify areas of

---

\*corresponding author

improvement, and explore opportunities for capability enhancement. While the literature on STAMP covers the use of the approach (Zhang et al., 2022), there is limited literature available on evaluating tools that have emerged to support the risk analysis process.

Additionally, evaluating STAMP tools is critical to identifying gaps and enhancements for strengthening risk analysis, especially for environments such as SoS, where systems are interconnected. Systems that worked in isolation are now intertwined with each other. Hence, modern engineered systems are required to execute missions within integrated systems, which encompass the integration of multiple systems (Keating et al., 2003). In SoS, multiple systems maintain operational and managerial independence (Maier, 1998), adding a layer of complexity for risk analysis. Additional aspects that drive complexity include the dynamic nature of SoS (Jamshidi, 2008); as the system evolves and increases, so do the sources of risk in the SoS. This implies that methods originally designed to analyze risk in other domains may not capture broader risk.

Key challenges highlighted in using STAMP include difficulties in performing risk analysis of complex systems (Aglave et al., 2018; Salmon et al., 2012). For example, STAMP may not provide a straightforward graphical representation for complex systems and their emergent behaviors and uncertainty (Bjerga et al., 2016; Underwood & Waterson, 2014). Such challenges could be amplified when STAMP is used in SoS, which are characterized by emergent behaviors and interactions of system components.

A study (Pettersson et al., 2023) found that one control structure was insufficient when STAMP is used to capture risk in a SoS. For example, a single control structure may be sufficient when analyzing risk at a point in time; however, it does not account for the evolving and dynamic nature of an SoS. As the SoS evolves, new risks between interactions can form emergent behavior during the system's existence. Therefore, tools must include broader procedures that capture a wider range of risks over time.

This paper evaluates the effectiveness of tools that support risk analysis using the STAMP approach. The study further explores the strengths and limitations of the tools and suggests possible enhancements for efficient risk analysis in the context of SoS. Thus, the study's objective is to assess the tool's adherence to the STAMP procedure, outline features for general user experience, and verify suitability for broader risk analysis.

## Research questions

This paper evaluates existing STAMP support tools by addressing the following questions:

1. What are the strengths and limitations of STAMP-based tools in supporting risk analysis, particularly in effectiveness, usability, and scalability?
2. How can STAMP-based tools be adapted or extended to address unique concerns in SoS, such as interdependencies, emergent behaviors, and dynamic system boundaries?

The study contributes to the research by enhancing the understanding of STAMP support tools, demonstrating criteria for their evaluation, and providing insights for tool enhancements as indicators for future research.

## Overview of paper

The next section discusses the foundation and theory of the study. This is followed by a description of the evaluation method used, and continues with a summary of the findings. The paper closes with the discussion and conclusion sections.

## BACKGROUND AND THEORY

The development of the STAMP model has a foundation in systems theory. Systems theory looks at a system as more than a sum of its parts, and the whole exhibits properties that none of its parts individually possess (Senge, 1990). A closely related concept is systems thinking, which looks at the system as a whole, its parts, and relationships (Anderson & Johnson, 1997; Arnold & Wade, 2015).

## Overview of the STAMP concept

Early works related to the STAMP model include that of Rasmussen, who studied adaptive risk management in which closed control feedback loops continuously enforce safety (Rasmussen, 1997). The work included human risk management with a focus on decision-making as impacting safety at different organizational levels. It also includes a focus on human factors in risk management, arguing that human decision-making at different levels of the organization impacts risk management (Rasmussen, 1997).

Leveson built on this legacy of control loops and hierarchical structures and proposed the STAMP model with human controllers and control actions as key concepts (Leveson, 2004). This creates a shift by providing a model that could broaden safety management. STAMP is an accident causality model based on system theory and system thinking. It addresses complex operations and unsafe interactions between elements, such as software, human factors, technology, and organizational factors (Leveson, 2004). This approach extends the traditional linear model beyond directly relating failure to a chain of events or component failure. Additionally, STAMP focuses on enforcing constraints to regulate system behavior by using control and feedback loops to enhance safety. Thus, safety is viewed as a dynamic problem where the system's unsafe actions can be adjusted (Leveson, 2016).

For analysis, the STAMP approach (Leveson & Thomas, 2018) has two analysis methods, STPA (Systems-Theoretic Process Analysis) and CAST (Causal Analysis using System Theory). STPA is proactive and focuses on analyzing potential causes of accidents to eliminate or control them. In contrast, CAST is the forensic analysis method used to investigate accidents after they have occurred. It identifies the causal factors involved in the risk analysis. This paper focuses on STPA to address the research questions.

## Evaluation of STAMP tool

Evaluating tools supporting the STAMP approach is key to determining their practical application and assessing their effectiveness. Additionally, it identifies the strengths and weaknesses of the tools for broader risk analysis.

Work related to this paper includes Underwood et al. (2016), in which practitioners evaluated the STAMP approach. They concluded that usability and graphical output were key aspects to improvement for the STAMP approach. However, the study does not evaluate the tools developed to support the STAMP approach, nor does it address the characteristics of SoS. The latter is critical for broader risk analysis.

Andrews et al. (2018) applied the STAMP approach to security in industrial control systems, and highlighted their effectiveness and limitations. Their findings emphasize the significant potential for software to reduce manual processes for the STAMP approach. They suggest better integration with existing systems engineering processes and advanced automation features. Similar to work by Underwood and Waterson (2014), the evaluation does not address how tools manage characteristics of SoS.

Other works such as Bjerga et al. (2016), Salmon et al. (2012), and Underwood and Waterson (2014) evaluate STAMP as an approach in comparison to other approaches. The works provide insight into the different aspects of the approaches, such as the ability to handle characteristics of SoS, such as emergent behavior and relationships between systems. Thus, the works do not focus on tools developed to support the STAMP approach.

Beyond the works of Bjerga et al. (2016), Salmon et al. (2012), and Underwood and Waterson (2014), a literature review by (Zhang et al., 2022) on STAMP provides insight into the limited availability of studies on the evaluation of tools that support STAMP. The review indicates that much of the research focuses on the STAMP approach's Methods and their usage for risk analysis across domains for specific case studies, rather than on software being developed to improve the approach's manual process.

This limitation of work on the evaluation of STAMP-based tools is a gap that presents a research opportunity, particularly in examining the tools for broader risk analysis in SoS, which is the focus of this paper. This gap presents an opportunity for future work to focus on developing evaluation frameworks and improving tools to cover domains such as SoS.

## Overview of the SoS concept

A SoS is a set of systems that collaborate while maintaining operational and managerial independence to achieve a common goal (Maier, 1998). The elements in a SoS are called constituent systems (CS). They interact to provide a unique capability that none can accomplish while acting by itself (Maier, 1998). Further, the SoS are complex and evolve (Maier, 1998).

Additional characteristics of SoS outlined by (Boardman & Sauser, 2006) include the following:

1. **Autonomy:** CS in the SoS have managerial and operational independence. This means that a CS is free to pursue its interests while contributing to the SoS objective.
2. **Belonging:** CS join an SoS based on a cost-benefit analysis relating to their system interests and needs.
3. **Connectivity:** The CS can link to other elements and can contribute to the overall SoS objective.
4. **Diversity:** A SoS should be highly diverse in its capabilities, and, for example, SoS system design should include capabilities that must respond to uncertainty.
5. **Emergent behavior:** New properties emerge from unforeseen system behavior, both good and bad. Further, Sage and Cuppan (2001) point out that emergent behavior in a system evolves from interactions of multiple participants, making it impossible to predict or envision from knowledge of individual elements or agents of the system.

Understanding the unique characteristics and types of SoS is important for effective risk analysis. The nature of SoS introduces challenges that are not typical for individual systems. As SoS evolve, interdependencies increase, which increases sources of risk, making unexpected emergent behavior and risk hard to predict (Axelsson & Kobetski, 2018).

Thus, the interconnected CS and their managerial and operational autonomy, along with emergent behavior, influence the identification and assessment of risks in SoS. To navigate these challenges, broader and holistic risk analysis tools for SoS are essential. This makes approaches like STAMP and associated supporting tools of interest for analyzing risk in SoS.

## METHODOLOGY

This study evaluated tools that support the STAMP risk analysis approach by examining the tools' alignment with the STAMP procedure, user experience, and capability for broader risk analysis. The study used design science methodology (DS), which is used in fields such as information technology and engineering and focuses on designing, validating, and evaluating artifacts (Hevner et al., 2004; Peffers et al., 2007).

These artifacts can be models, methods, frameworks, and systems that could be reusable to solve real-world problems (Hevner et al., 2004). For this study, the DS Framework for Evaluation (FEDS). Venable et al. (2016) is used to evaluate the software artifacts developed to support the STAMP risk analysis procedure. The DS evaluation, FEDS, was complemented by (ISO/IEC/IEE 25041, 2024), which guided the general features to evaluate, including functionalities to achieve the purpose of the system, usability for the user, and scalability.

## Evaluation

An Evaluation can cover different things, such as functionality, liability, usability (ISO/IEC/IEE 25041, 2024). Thus, this study's objective is to assess the tool's adherence to the STAMP procedure, outline features for general user experience, and verify suitability for broader risk analysis in SoS. In line with this study's objective, a DS evaluation framework, FEDS (Venable et al., 2016) and the STAMP procedure are used as criteria to assess the STAMP supporting tool. The key elements for evaluation include functionality, usability, and the capability to manage SoS characteristics, for broader risk analysis.

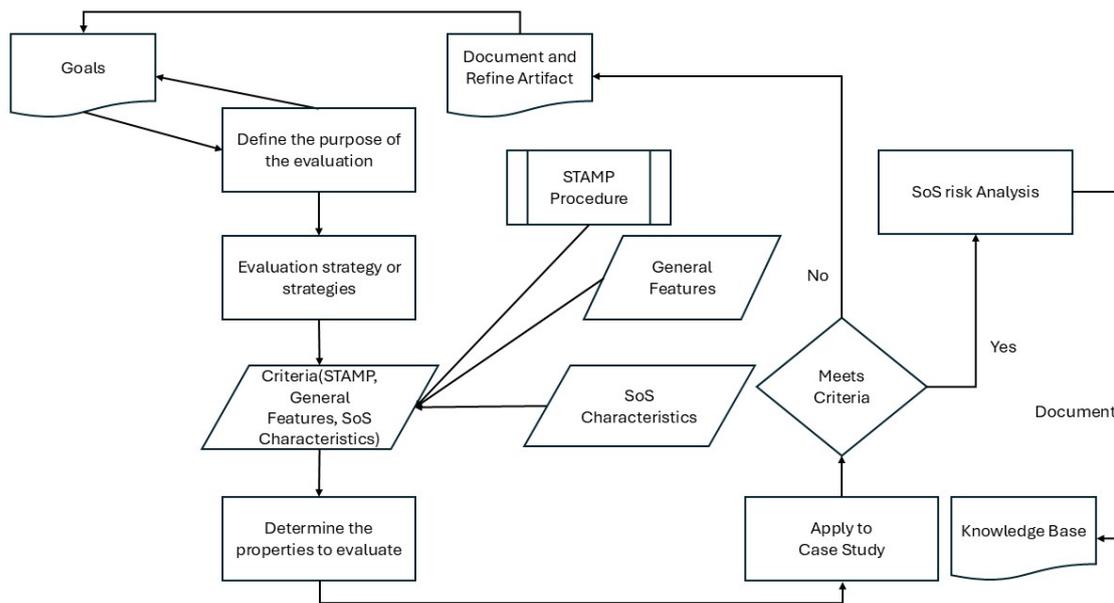
An evaluation framework such as FEDS provides a structured approach for the DS Methodology Section of this paper. The FEDS (Venable et al., 2016) steps and the activities carried out during the study are outlined below:

1. **Define the purpose of the evaluation:** The purpose was to evaluate existing artifacts, namely the STAMP tools, which are software developed to support STAMP analysis.

The study aimed to evaluate the software's functionality for alignment with the STAMP procedure, usability, and ability to manage the complexity characteristic of SoS.

2. **Evaluation strategy or strategies:** The strategy for this evaluation used was both formative and summative. Summative evaluations are often used to measure a completed development's results or appraise a situation before development begins. On the other hand, formative assessments yield empirically supported interpretations that serve as a foundation for effective action in improving the characteristics or performance and functionality (Venable et al., 2016).

Thus, in this study, a formative assessment is used as it aligns with the purpose of assessing the functionality of the tools and their ability for broader risk analysis in SoS. For each tool evaluated, the STAMP procedure



**Figure 1. Application of the Evaluation process**

and evaluation criteria for SoS, as outlined in Sections: Criteria for STAMP procedure and Evaluation criteria for SoS alignment, were applied to a case study.

3. Determine the properties to evaluate: The FEDS includes possible generic artifact properties to include to guide in determining properties to evaluate. This includes:

- Adapting levels of granularity, such as evaluating if a task was completed.
- Adapting context, input, process, and product; for example, adjusting goals, strategy, work plan, and outcome according to needs.
- Adapting criteria as design goals: Ensuring artifacts are usable, secure, efficient, correct, reliable, maintainable.
- Adapting both rationality and understanding such perform quality assurance to maintain standards.

Using this FEDS guide and adapting the evaluation for criteria as design goals and context, this study's sets the evaluation criteria outlined in the Sections: Criteria for STAMP procedure and Evaluation criteria for SoS alignment of this paper. Key properties for evaluation, as guided by FEDS, are derived from the study goal and design elements (ISO/IEC/IEE 25041, 2024). The design elements include functionality, usability, and scalability of supporting tools for broader risk analysis. Additionally, functionality, usability, and scalability properties were also guided by guidelines from (ISO/IEC/IEE 25041, 2024), which guides on key elements of software evaluation. Figure 1 illustrates the evaluation process of the FEDS Framework as applied in this paper.

Instead of evaluating all available tools, a subset was selected based on study objectives and time constraints. At the start of this study, there were 15 tools listed on the STAMP site<sup>1</sup>. The basis for selection along with selected tools were:

- Tools that have been possibly deployed in the industry: Visio Pro and STAMP Bench, which uses the commercial name ASTAH were included.
- The tools have been refined over time: Masterlite was included.
- Tools that indicate popularity, for example, the most downloaded tool: XSTAMPP was included.
- Tools that indicate in the description the possibility of software integration: STPA Capella was included.

<sup>1</sup>STAMP Tools, available at: url <https://psas.scripts.mit.edu/home/stamp-tools/>, accessed on January 23, 2025

A pandemic management SoS (Pettersson et al., 2024) was used as a case study. Each tool was used for risk analysis and evaluation for alignment in supporting the STAMP procedure. Additionally, overall systems functionality such as usability and scalability were evaluated to assess the tools used in the context of SoS.

### Criteria for STAMP procedure

The evaluation documented how each tool supported the STAMP analysis method by facilitating prescribed steps outlined in the STAMP handbook (Leveson & Thomas, 2018). The steps for performing the safety analysis that was systematically carried out using each tool are listed below:

1. Define the purpose of the analysis.
  - Identify losses
  - Identify hazards
  - Define system constraints
2. Model the control structure in the system.
3. Identify unsafe control actions.
4. Identify loss scenarios.

### Evaluation criteria for SoS alignment

Evaluation criteria for SoS alignment include assessing for capabilities to support characteristics such as evolution, interdependencies, and interconnections. Details of these characteristics are outlined in the results section.

## RESULTS

This section summarizes the identified strengths and limitations of existing tools regarding alignment with the STAMP procedure, general usability or user experience, and scalability for broader use in complex systems such as SoS.

### Strengths

The evaluation indicated that all evaluated tools demonstrated compliance with STAMP by providing basic functionality used for risk analysis by STAMP, as shown in Table 1.

Additionally, a positive step towards scalability is that the tools provide the functionality to define more hazards than STAMP recommends for a single analysis. Although there are no hard limits on the number of system-level hazards, STAMP recommends no more than seven to ten for a manageable list (Leveson & Thomas, 2018).

Further, of the tools evaluated, STPA-Capella, XSTAMPP STPA, and MasterLite are open source, while ASTAH and Visio Pro require licenses.

In addition, the evaluation found the following:

- Visio Pro, ASTAH, and Master Lite provided additional functions, such as an auto-generating feature from existing input data, a precondition table, a function to describe responsibilities, a definition of system actors, and base knowledge linked to the STAMP handbook for reference.
- Capella provided model-based integration functionality with the Capella MBSE tool, while STPAmaster Lite uses Google Docs.

Table 1 provides a summary of the results of software alignment to the STAMP approach.

Additionally, Table 2 shows a summary of the evaluation results on usability and functionality. Added usability includes features such as a lookup table and knowledge base. Additionally, reporting capability includes exporting directly to a user report in addition to the standard export functionality to the Excel package.

**Table 1. The results of mapping support tools to the STAMP procedure**

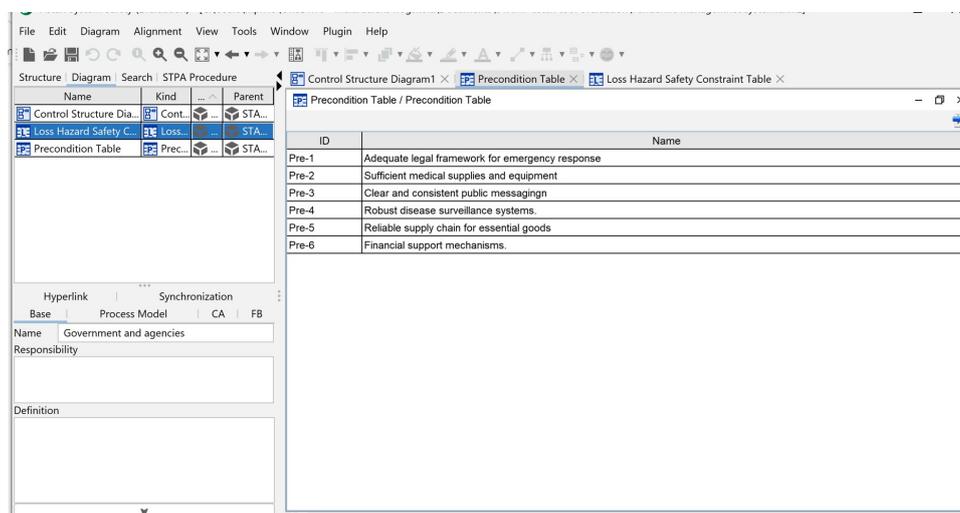
Step in STAMP method	ASTAH	STPA-Capella	XSTAMPP	Masterlite	Visio Pro	Comment
1. Define the purpose of analysis: Losses, Hazards, and System constraints	✓	✓	✓	✓	✓	Visio Pro has an extra feature for auto-generation, including reports. Astah can be used for a broader part of SysML modeling
2. Build a model	✓	✓	✓	✓	✓	Modeling limited, e.g, to a single entity control diagram as illustration Figure 2, integration functionality unavailable.
3. Unsafe control action (UCA)	✓	✓	✓	✓	✓	
4. Identify scenarios	✓	✓	✓	✓	✓	

**Weaknesses**

Although tools offer basic reporting features, the evaluation results revealed limitations in advanced features such as automatic report generation. Moreover, the tools that have advanced reporting capabilities lacked platform independence. This creates a trade-off between capability and functionality. Users who need enhanced features could consider using licensed software.

However, scalability, an important feature for SoS, was limited in all the evaluated tools. Figure 2 illustrates an analysis where only one control diagram can be generated at a time based on one precondition table and a set of Hazards. This constraint in the design of the software limits scalability.

Other limitations relate to scalability include difficulty mapping and linking many elements in ASTAH, STPA Capella, and Master Lite. Additionally, challenges were generating large models XSTAMPP and large diagrams in Visio Pro. These limitations make it difficult to manage numerous interconnected elements and can thus render risk management in SoS ineffective.



**Figure 2. Tool application of STAMP procedure**

**Table 2. Evaluation of Software Tools: Generic features**

Property	ASTAH	STPA-Capella	XSTAMPP	STPA Master Lite	Visio Pro	Comment
Basic input and output	✓	✓	✓	✓	✓	Users can create lists and tables for control actions, losses and loss scenarios.
Scalability limitations	linking many elements	elements	large models	elements	large diagrams	Generally limited for large systems.
Availability	licensed	✓	✓	a free version exist	licensed	Open Source
Report features					✓	Enhanced report capability.
Enhanced usability	✓			✓	✓	Allows detailed description of entities, lookup feature, Tool auto-generation of loss scenarios, validation of errors etc
Collaboration features						Collaboration not available.
Basic reporting features	✓	✓	✓	✓	✓	Basic reporting available.
Use assisted features				✓	✓	User assistance, e.g., auto-generation.

## DISCUSSION

This study evaluated tools that support risk analysis by verifying compliance with the STAMP procedure, evaluating generic features such as usability, and verifying the ability to perform broader risk analysis in SoS.

### Tool strengths

Notable strengths include the tools' provision of basic modeling capabilities that conform to the STAMP procedure, along with basic reporting functions. The tools evaluated allow flexibility on the number of hazards analyzed at a time; this enables more hazards to be included for large systems. These strengths offer a baseline for STAMP and the supporting tools in risk analysis for large systems like SoS.

Additionally, extra features provided by tools that enhance usability, such as auto-population from existing input data, provide a good start for SoS analysis. The added value is that large amounts of data can be entered once and provide information for the connection and relationship of systems in SoS.

Further, the availability of both licensed and open-source tools provides a solid starting point for the use of STAMP support tools in SoS risk analysis as this offers flexibility in software choices.

### Tool limitations

A solid foundation with basic modeling and reporting functionality was evident in the evaluation of STAMP support tools. However, the presentation of a large control diagram is limited. Further, multiple modeling can be created as different projects, but a provision for the integration or link of these projects was not available within a defined project, thereby limiting analysis.

This limited analysis in an SoS is unlikely to capture all risks. Sources of risk in an SoS are diverse; for example, (Pettersson et al., 2023) applied time slicing and created multi-control structures to capture risks at various points in

time as the SoS evolves. This indicates that risks are different for various timelines of SoS evolution, making a single assessment of SoS insufficient.

Further, using available STAMP supporting tools to create manageable groups of components for analysis is possible. However, functionality for creating and reporting an integrated structure does not exist in all the tools.

Additionally, a few tools offer user-friendly functions like auto-generation from input data. Without these, the efficiency is limited, especially when applied to SoS where interdependencies and loops increase as the system evolves. Given these characteristics of SoS, enhanced efficiency and accuracy could also be achieved with emerging technologies such as artificial intelligence (AI) that are not yet available in the tools.

A lack of collaboration features in the tools could contribute to subjectivity in the analysis. One of STAMP's shortcomings is that analysis can struggle with performing the analysis (Salmon et al., 2012). This subjectivity could be enhanced in complex systems such as SoS.

### Implications for SoS risk analysis

Existing tools supporting STAMP allow for step-by-step guidance and provide a structural workflow of analysis as prescribed by the STAMP method (Leveson & Thomas, 2018). This is significant for the STAMP approach because it is a starting point for a shift from a manual risk analysis method to risk analysis supported by automation. The advantages of automation, including improved user experience, are a positive feature. Noteworthy, during the evaluation of tools were enhancements for usability such as auto-generation from existing data. However, these enhancements do not exist in most tools. Moreover, scalability is limited for all support tools.

These findings show a good foundation to support risk analysis by the STAMP approach. An important implication of this study is that emerging STAMP tools need to incorporate features that address the characteristics of complex systems such as SoS. Since SoS are increasingly prevalent in modern society (INCOSE SoS Primer, 2018), STAMP support tools should provide scalability to allow better risk analysis of SoS.

### CONCLUSION

This study evaluated tools that support the STAMP approach with a focus on their alignment for SoS risk analysis. The tools' strengths include their capability to apply the STAMP procedure and some advanced generic features in reporting as well as auto-population. These strengths contribute to the efficiency of risk analysis using the STAMP approach. While the tools are useful in analyzing risk in complex systems, they face challenges in addressing dynamic, interconnected large systems. Hence, current tools that support the STAMP approach must be scalable and modified to address the needs of SoS risk analysis.

In the SoS domain, complexity requires broader risk approaches that address risk from interdependencies and emergent risk as the system evolves (Conrow, 2005). Hence, It is essential to ensure that tools adequately address SoS risk analysis needs.

This study indicates that STAMP tools provide basic modeling and reporting features, which form a good foundation. However, the tools need to be adapted for SoS risk analysis. The findings also reveal that for SoS risk analysis, STAMP manual procedures would still be required to be used either in isolation or in combination with some tools since scalability is limited in existing tools.

Future continuations of this work include a deeper analysis of the proposed tool improvements with a focus on artificial intelligence. The following outline will complete this work on tools supporting the STAMP approach:

- Investigate advanced features such as integration of the report and diagrams.
- Explore Artificial Intelligence (AI) and Machine Learning techniques to enhance scalability.
- Explore how technological enhancements can be applied to support available tools for risk analysis in SoS.

### ACKNOWLEDGMENT

This research was funded by the Swedish Civil Contingencies Agency (MSB) under grant no. 2021-13694.

## REFERENCES

- Aglave, A., Chaudhuri, D., & Johnson, S. (2018). Application of STAMP to process industry. *Abu Dhabi International Petroleum Exhibition and Conference*, D022S136R002.
- Anderson, V., & Johnson, L. (1997). *Systems thinking basics*. Pegasus Communications Cambridge, MA.
- Andrews, T., Moulton, A., & Madnick, S. (2018). An examination of software tool features needed to help secure energy delivery industrial control systems.
- Arnold, R. D., & Wade, J. P. (2015). A definition of systems thinking: A systems approach. *Procedia computer science*, 44, 669–678.
- Axelsson, J., & Kobetski, A. (2018). Towards a risk analysis method for systems-of-systems based on systems thinking. *2018 Annual IEEE International Systems Conference (SysCon)*, 1–8.
- Bjerga, T., Aven, T., & Zio, E. (2016). Uncertainty treatment in risk analysis of complex systems: The cases of STAMP and FRAM. *Reliability Engineering & System Safety*, 156, 203–209.
- Boardman, J., & Sauser, B. (2006). System of systems-the meaning of of. *2006 IEEE/SMC international conference on system of systems engineering*, 6–pp.
- Conrow, E. H. (2005). Risk management for systems of systems. *CrossTalk*, 18(2), 8–12.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 75–105.
- Jamshidi, M. (2008). *System of systems engineering: Innovations for the 21st century*. Deakin University.
- Keating, C., Rogers, R., Unal, R., Dryer, D., Sousa-Poza, A., Safford, R., Peterson, W., & Rabadi, G. (2003). System of systems engineering. *Engineering Management Journal*, 15(3), 36–45.
- Leveson, N. G. (2004). A new accident model for engineering safer systems. *Safety Science*, 42(4), 237–270.
- Leveson, N. G. (2016). *Engineering a safer world: Systems thinking applied to safety*. The MIT Press.
- Leveson, N. G., & Thomas, J. P. (2018). *STAMP handbook*. MIT Press.
- Maier, M. W. (1998). Architecting principles for systems-of-systems. *Systems Engineering: The Journal of the International Council on Systems Engineering*, 1(4), 267–284.
- INCOSE SoS Primer. (2018). *Systems of systems primer*.
- ISO/IEC/IEEE 25041. (2024). *ISO/IEC 25041:2024 – systems and software engineering — systems and software quality requirements and evaluation (SQuaRE) — evaluation guide for developers, acquirers and independent evaluators*. Geneva, Switzerland.
- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, 24(3), 45–77.
- Pettersson, M. N., Axelsson, J., Johansson, A., & Svenson, P. (2024). Risk analysis for system of systems management: The swedish covid-19 management case. *2024 IEEE International Systems Conference (SysCon)*, 1–8.
- Pettersson, M. N., Axelsson, J., Svenson, P., & Johansson, A. (2023). Towards a risk analysis method for systems of systems: A case study on wildfire rescue operations. *20th Global Information Systems for Crisis Response and Management Conference, ISCRAM 2023*, 530–545.
- Rasmussen, J. (1997). Risk management in a dynamic society: A modelling problem. *Safety science*, 27(2-3), 183–213.
- Sage, A. P., & Cuppan, C. D. (2001). On the systems engineering and management of systems of systems and federations of systems. *Information knowledge systems management*, 2(4), 325–345.
- Salmon, P. M., Cornelissen, M., & Trotter, M. J. (2012). Systems-based accident analysis methods: A comparison of Accimap, HFACS, and STAMP. *Safety science*, 50(4), 1158–1170.
- Senge, P. M. (1990). *The fifth discipline: The art and practice of the learning organization*. Doubleday.
- Sun, L., Li, Y.-F., & Zio, E. (2022). Comparison of the HAZOP, FMEA, FRAM, and STPA methods for the hazard analysis of automatic emergency brake systems. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, 8(3), 031104.
- Underwood, P., & Waterson, P. (2014). Systems thinking, the Swiss cheese model and accident analysis: A comparative systemic analysis of the grayrigg train derailment using the ATSB, Accimap and STAMP models. *Accident Analysis and Prevention*, 68, 75–94.

- Underwood, P., Waterson, P., & Braithwaite, G. (2016). Accident investigation in the wild. a small-scale, field-based evaluation of the STAMP method for accident analysis. *Safety science*, 82, 129–143.
- Venable, J., Pries-Heje, J., & Baskerville, R. (2016). FEDS: A framework for evaluation in design science research. *European journal of information systems*, 25(1), 77–89.
- Zhang, Y., Dong, C., Guo, W., Dai, J., & Zhao, Z. (2022). Systems theoretic accident model and process (STAMP): A literature review. *Safety science*, 152, 105596.