# A five-layer model for the analysis of complex socio-technical systems

TOMOKO KANEKO, National Institute of Informatics
NOBUKAZU YOSHIOKA, National Institute of Informatics

Current systems, which connect things and people through computer systems, have had a considerable effect on society. Not only are things connected, but systems, the people who use them, and organizations have a complex relationship. In order to model a complex system, we propose to use a variant of the layers pattern to de-scribe the system control structure using five layers according to the lifecycle of soft-ware and other system requirements. This model applies several techniques such as safety, security, risk, and incident analysis. We thus extend the System Theoretic Accident Model and Process (STAMP) model, proposed by Nancy Leveson, to pro-duce the System Theoretic Architecture Model and Process (STAMP S&S). The rea-son for the term Architecture rather than Accident is that it extends STAMP with a five-layer hierarchical pattern variant and a process, necessary for various analyses. A way to perform analysis in each layer and to generate specifications and standards ensuring safety and security or other qualities is proposed based on the STAMP S&S Five-layer model.

**ACM Reference Format:**

Kaneko, T. and Yoshioka, N. 2020. A five-layer model for the analysis of complex socio-technical systems. HILLSIDE Proc. of Conf. on Pattern Lang. of Prog. (October 2020), 7 pages.

## 1. INTRODUCTION

Current systems, which connect things and people through computer systems, have had a considerable effect on society. Not only are things connected, but systems, the people who use them, and organizations can have a complex relationship [1]. Although Artificial Intelligence (AI), is rapidly becoming part of commercial systems, it is challenging to ensure the safety and reliability of machine learning systems, including mission-critical systems such as automatic driving.

To ensure safety and reliability in complex systems, it is necessary to first understand the entire target system, and the effects of its components on each other, model them clearly and deal with its risks. However, it is difficult to model a complex system as a whole when its implementation has not yet been defined. Currently, the lifecycle model standards for systems and software specify the requirements for "what" is to be implemented during the planning, development, operation, and maintenance processes. However, the problem is that "this lifecycle does not specify how to build each stage." That objective requires the use of a development methodology and there are several of them intended to build secure systems using patterns [2]; however, there are no methodologies to build systems combining several quality factors.

Leveson proposed the System Theory Accident Model and Process (STAMP) methodology [3] and its analysis methods, such as System Theory Process Analysis (STPA) [4]. STAMP uses specifications, safety guide design, design principles, system engineering, risk management, management principles, and regulations of organizational design to analyze safety aspects. Our objectives are to extend the use of STAMP not only as an accident model but also as a model where we can analyze the impact of complex systems on society. For that purpose, we will apply the Layers pattern [5] to separate the different system concerns in the form of a five-layer model. This pattern illustrates components and their interactions as components of a society and analyzes erroneous control actions which are unsafe or insecure. Improvements in control actions are reflected in the output of the analysis in the system service

This work is supported by the Widget Corporation Grant #312-001.
Author's address: Kaneko, T; 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan; email: t-kaneko@nii.ac.jp; Yoshioka, N, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan; email: nobukazu@nii.ac.jp;.

stakeholders found in society. The impact of this model can be measured when its results are reflected in specifications and standards. This pattern's users are system engineers and analysts who work on systems that require complex and diverse considerations, including human and social aspects, such as autonomous driving and smart cities, and want to know how to build them. As a first step toward solving this problem, the authors aim to "model complex systems, analyze them to ensure qualities such as safety and security, and to establish ways to standardize the results." Traditionally, single devices, components of a complex system, have been analyzed in great detail. However, that method has its limitations for the analysis of a complex system that requires considering it as a whole.

We propose here a model using a five-layer pattern and processes using a control structure for complex socio-technical systems. Background is introduced in Section 2, while the new model is explained in Section 3. Section 4 presents some conclusions.

## 2. BACKGROUND

### 2.1 STAMP model and its related methods

Modern embedded systems are becoming increasingly larger and more complex due to the interaction among connected elements in addition to the advanced functionality of each component. Traditional safety analysis techniques (such as Fault Tree Analysis [6], FMEA [7], and HAZOP [8]) are based on accident chain event models. However, since these models seek causal relationships between individual events, they cannot capture a complex system as a whole. Therefore, to ensure the safety of these complex systems, Leveson proposed the System Theory Accident Model and Process (STAMP) [3] and its analysis methods, like System Theory Process Analysis (STPA) [4]. The STAMP model was proposed as an improvement of the conventional safety models such as the Domino and Swiss cheese models. The operation of STAMP is explained by focusing on the element (component) and the interaction (Control action) in Fig.1. Many of the system accidents are not only caused by the failure of the components, but also because of the interaction of the control elements (control and controlled element). As a process, STAMP uses specifications, safety guide design, design principles, system engineering, risk management, management principles, and regulation of organizational design. However, STAMP is originally an accident model for safety, not for security or reliability. It has not been established as a method for analyzing the impact of computer systems on society.

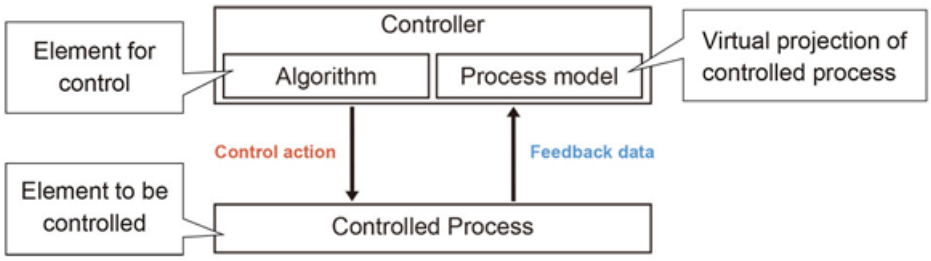


Fig.1. Control and controlled component in STAMP [3]

### 2.2 Traditional Methods of Security Analysis and development

Security analysis includes two aspects: 1) threat modeling and enumeration, 2) security evaluation.
Threat modeling and enumeration requires modeling how a threat is performed and several models have been proposed such as attack trees [9], misuse cases [10], and STRIDE [11]. Some HAZOP-based security analysis methods have also been proposed [12]. Security evaluation implies estimating the degree of security reached by a system;

there are no widely accepted methods for this purpose. Security evaluation is applied after a design stage is completed and to the complete system.

There are also several methodologies to build secure systems. A group of these is based on building secure code and includes Microsoft Security Development Life Cycle (SDL) [13]. SDL analyzes threats using STRIDE. A more effective group of methodologies uses model-driven engineering (MDE) [2]. MDE is necessary to deal with complex systems because it applies abstraction.

However, compared with traditional standardized safety analysis methods such as FTA and FMEA, no security analysis or development methodology has been standardized and become of widespread use.

### 2.3 Socio-technical Systems and Software Engineering

A system which includes nontechnical elements such as people, processes, and regulations, as well as technical components such as computers, software, and hardware, is called a socio-technical system [14]. These systems are so complex that it is impossible to understand them as a whole. Therefore, we have to study them using layers. The socio-technical systems stack is shown in Fig.2. Software systems are not isolated systems but are part of more extensive systems that have a human, social, or organizational purpose. Therefore software engineering is not an isolated activity but is an intrinsic part of systems engineering. Also, as shown in Fig. 2, software engineering includes business processes, application systems, communication and data management, and operating system layers, and system engineering also covers organization and equipment (hardware). The social layer is not included in software engineering and systems engineering [14]. However, although a complex system requires modeling based on such a hierarchy, the socio-technical systems stack does not show how to analyze it.

The Layers pattern is a common architecture pattern, used in many applications, e.g., the ISO standard for communication networks uses a 7-layer decomposition. Its main objectives are separation of concerns and decoupling of the system functions so they can evolve independently [5]. A variant is the Secure Layers pattern [15], where the layers hide sensitive parts of a system. Another important variant is the N-tier pattern, which is a business architecture describing the typical layers of IT systems [16]. Components within the layered architecture pattern are organized into horizontal layers, each layer performing a specific role within the application (e.g., presentation logic or business logic).


Fig.2.The socio-technical systems stack[14]

## 3. A FIVE-LAYER MODELING AND ANALYSIS PROCESS FOR COMPLEX SOCIO-TECHNICAL SYSTEMS

### 3.1 Motivation

Systems are becoming more and more complex with the introduction of the Internet and technologies such as AI and blockchain. There is a great need to develop safe and secure systems that meet the needs of society.

### 3.2 Problem

Complex systems cause safety and security problems, and accidents and incidents occur. We need to develop complex systems which are safe and secure, but it is difficult to develop such systems. Traditional models that capture accidents are models that consider isolated objects, such as the Swiss cheese and domino models. Also, typical hierarchical system models capture the system technical layers, but usually do not include the society layer.

The forces constraining our solution are:

A) Complex systems cannot be understood as a whole.
B) Computer systems affect not only systems but also various and complicated objects, such as people and organizations.
C) The analysis that captures the entire system, including people and organizations, can be performed using the STAMP model, but it has not been defined how that model can be divided into layers for detailed analysis.
D) A method for analyzing the influence of computer systems on society has not been defined.
E) Existing analysis methods vary depending on their attributes and targets, such as safety, security, risk, and accident.
F) Even if there is a suitable model, quality factor analysis cannot be done unless an analysis process is defined.

### 3.3 Solution

Define a modeling and analysis process with a 5-layer control structure diagram. This solution is called STAMP S&S five-layer model, which tries to capture different interacting elements, different perspectives, and needs [17] [18]. In order to do this, STAMP's Control Structure (CS) diagram, which is based on system theory, is a good choice. However, STAMP's CS diagram is mainly used as a basis for hazard analysis in a system layer, so it is necessary to devise a way to capture the interactions in a more hierarchical manner. Clarifying the hierarchy enables us to capture the characteristics of each layer and to conduct detailed analysis within each layer, which enables us to capture the whole system more accurately. In addition, STAMP S&S uses the analysis methods of STAMP in the safety and security engineering methodology for AI/IoT called CC-Case [19] [20].

Solutions to the forces are:

A) Use STAMP methodology that captures a complicated system as a whole.
B) Model a complex system hierarchically.
C) Divide the software layers into system layer, service layer, and stakeholder layer.
D) Determine the process of analyzing interactions and create a method that can include the analysis of the social layer, which is not included in software and systems engineering.
E) Analyze by focusing on the interaction of the various components.
F) Make a model with an analysis process.

Therefore, we propose to use a five-layer model. To show that the pattern is a complete solution constrained by the forces, the five layers define the hierarchical model solution and the analysis process solution in detail, as shown below.

#### 3.3.1 The hierarchical model solution

A model that captures complex systems as a whole is needed. This is because of the high-quality development of complex systems requires capturing the interaction of various elements, including different perspectives and needs. In order to do this, STAMP's Control Structure (CS) diagram, which is based on system theory, is a good choice. However, STAMP's CS diagram is mainly used as a base for hazard analysis in a system layer, so it is necessary to devise a way to capture the interaction in a more hierarchical manner. Clarifying the hierarchy enables us to capture

the characteristics of each layer and to conduct detailed analysis within each layer, which enables us to capture the whole system more accurately.
Therefore, we propose a five-layer model. These five layers consist of Society, Stakeholder, Service, System, and Software, which we call the STAMP S&S five-layer model. These five layers are in an inclusive relationship where the upper layers include the lower ones. This model uses a control structure diagram that shows the interaction between the components in terms of control actions and feedback. This CS diagram is a different graphical representation from UML, which is commonly used in software models; the elements of the CS diagram have been presented as interactions of control relations, i.e., what is controlled and what is done. However, the authors view this as an interaction of connections, rather than control.

### 3.3.2 The analysis process solution

The process of analysis defines the goals that are aimed at and model the interactions between the components involved in those goals. A component is an element of the 5-layer model. Then, the factors that support or hinder the goals are derived in the interaction, and scenario analysis is carried out to deal with them. A scenario is a knowledge representation used in a predefined sequence of events to determine the outcome of interactions between known entities. Various analyses, such as safety, security, reliability, risk, and accident, are conducted using scenarios. The output of analysis at each layer of software, systems, services, and stakeholders is the "specification." The output of the analysis is defined in the social layer.

## 3.4 Stages of the process model

The process model should define the activities in each lifecycle stage. For example, the requirements and analysis stage should enumerate the threats and hazards for the system under design. This is the stage where the 5-layer model is necessary.

### 3.4.1 Implementation of the hierarchical model

The specific contents of each layer are defined in Table 1.

Table1. Specific contents of each layer [17]

| Layer | Contents |
|---|---|
| Society | Human social life (rules, standards, customs) and its external environment (natural environment such as weather) |
| Stakeholder | Individual or organization having a right, share, claim or interest in a system or in its possession of characteristics that meet their needs and expectations such as business |
| Service | Actions performed by people, and services provided by people and organizations |
| System | Combination of interacting elements organized to achieve one or more stated purposes.<br>e.g. Hardware, which describes the physical aspects of a computer, communication equipment, semiconductor chip |
| Software | Set of instructions, data or programs used to operate computers and execute specific tasks.<br>e.g. Programs (application software, OS, and other software), cyber information, data, and AI. |

### 3.4.2 Implementation of the analysis process

The following is an overview of the scenario analysis process.
(1) Take the event to be analyzed as a goal and determine the scope of analysis.
(2) Find the loss or achievement conditions for the goal.
(3) Determine the components in each layer and diagram the control relationship.
-The control component issues control actions to the controlled component.
-The control component issues control actions to the controlled component, and the controlled component returns feedback to the control action.
-Each component has its own algorithm and process model.
-Each component has its own algorithm and process model.

-If the component is a person, the process model is a mental model.
If the component is a person, the process model is a mental model - The process model or mental model is aware of what the process (state) of the component under control is.

(4) Scenario-based state and cause analysis is performed on control actions.
-Problematic conditions occur when there is a flaw in the perception of the process model or mental model.
-Perform analysis to find and address this problematic state.
-It we cannot, find only the problematic state and also analyze the requirements of the achievement conditions.

5) The extracted requirements are reflected in specifications and standards and improved interactively.

* Depending on the attributes of the target, this can be safety analysis, security analysis, reliability analysis, privacy analysis, or maintainability analysis.

In summary, scenario-based analysis is conducted by modeling what is the goal and the state of affairs in a five-level CS diagram to determine the scope of the analysis and then focusing on the control actions.

### 3.5 Consequences:

The answers to the forces were presented in the problem section.

A) A complex system was modeled.
B) It was possible to model with layers.
C) The layered model allows us to perform various analyses based on one structure.
D) Components of the social layer were also presented and could be analyzed for interaction.
E) The analysis can be conducted with various emergent properties (various quality attributes).
F) Analyze various objects and conditions, such as risk analysis, accident analysis, business process analysis, and social needs analysis. Analyze by focusing on the interaction of various components.

The five-layer model can be seen as an interaction between a five-layer perspective: society, business, people, subsystems and devices, and software.

This interaction can be used to analyze by looking for non-secure states of control actions, as well as to analyze various quality requirements such as security, privacy, maintainability. The CS diagram of STAMP S&S should be an architectural model that captures the functional requirements of the entire system. However, it is a challenge to present in the future concrete examples of application of these functional requirements.

Although special structure and process for analysis have not been defined in all the layers, STAMP S&S has five layers for analyses of complex socio-technical and can analyze various emergent properties such as safety, security, reliability.

The overall view is that a wide range of systems and devices are connected, such as the IoT, from a highly abstract layer that broadly perceives society itself as one system. It can be handled hierarchically from one system and individual device layer, each component in the device, the functions in the software, and the relationships from the layer with high abstraction can be captured so that it can be tracked.

## 4. CONCLUSIONS

A particular combination of software, systems, services, stakeholders, and societies is modeled as a variant of the Layers pattern. Starting from a high layer we can propagate the analysis to the lower layers to understand the effects of events in each layer. Because of space restrictions we do not show examples here, those and more details of our approach can be found in a longer version of this paper (in preparation).

## 5. ACKNOWLEDGEMENTS

We thank Prof. Eduardo Fernandez for his great help as a shepherd and in preparing the final version of the paper.

REFERENCES

1. Information Promotion Agency (IPA), "IoT Safety/Security Development Guidelines (Second Edition)" Important Points to be understood by Software Developers toward the Smart-society https://www.ipa.go.jp/english/sec/reports/20160729-02.html.
2. Anton V. Uzunov, Eduardo B. Fernandez, Katrina Falkner, "Securing distributed systems using patterns: A survey", Computers & Security, 31(5), 2012, 681 - 703. doi:10.1016/j.cose.2012.04.005

3. Nancy G Leveson, 2012. Engineering a Safer World, MIT Press
4. "STPA handbook," http://psas.scripts.mit.edu/home/
5. F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. Pattern-Oriented Software Architecture: A System of Patterns, Vol. 1. J. Wiley, 1996.
6. IEC 61025:2006 Fault Tree Analysis(FTA), https://webstore.iec.ch/publication/4311
7. United States Military Procedure, "Procedure for performing a failure mode effect and criticality analysis," November 9, 1949, MIL-P-1629
8. IEC 61882:2001 Hazard and operability studies (HAZOP studies) - Application guide. http://www.iec.ch
9. Schneier, Bruce. "Attack Trees," Dr. Dobb's Journal of Software Tools, 24(12), (1999), 21–29.
10. Sindre, Guttorm; Opdahl, L. Andreas. "Eliciting security requirements with misuse cases," Requirements Engineering, Vol.10, No. 1, pp. 34–44 (2005).
11. Lipner, Steve; Howard, Michael. "The Trustworthy Computing Security Development Lifecycle," https://msdn.microsoft.com/en-us/library/ms995349.aspx.
12. Jingxuan Wei, Yutaka Matsubara,Hiroaki Takada,"HAZOP-Based Security Analysis for Embedded Systems: Case Study of Open Source Immobilizer Protocol Stack.", Recent Advances in Systems Safety and Security pp 79-96 2016
13. Shostack, Adam. "Threat Modeling: Designing for Security," Wiley., 2014.
14. Ian Sommerville, " Software Engineering-10ed", Pearson Education Limited (2016)
15. E.B.Fernandez, Security patterns in practice: Building secure architectures using software patterns, Wiley Series on Software Design Patterns, 2013
16. https://www.oreilly.com/library/view/software-architecture-patterns/9781491971437/ch01.html
17. Tomoko Kaneko, Nobukazu Yoshioka, "STAMP S&S: Layered Modeling for the complexed system in the society of AI/IoT," JCKBSE2020
18. Tomoko Kaneko, Nobukazu Yoshioka, Ryouichi Sasaki, "STAMP S&S: Safety & Security Scenario for Specification and Standard in the society of AI/IoT," The 2020 IEEE International Workshop on Cyber Forensics in Software Engineering (CFSE)
19. Tomoko Kaneko, Nobukazu Yoshioka, "CC-Case: Safety & Security Engineering Methodology for AI/IoT", 1st chapter of "A Closer Look at Safety and Security", 2020
20. Tomoko Kaneko, Shuichiro Yamamoto, Hidehiko Tanaka," CC-Case as an Integrated Method of Security Analysis and Assurance over Life-cycle Process", International Journal of Cyber-Security and Digital Forensics 3(1) 49 - 62 2014

Received July 2020; revised January 2021; accepted August 2020