23rd International Command and Control Research & Technology Symposium

Topic 9: Knowledge Systems

Framework for C2 Concept Development: Exploring Design Logic and Systems Engineering

Niklas Hallberg, Magdalena Granåsen, Anders Josefsson and Christina Ekenstierna

Niklas Hallberg
Swedish Defence Research Agency, Sweden

Magdalena Granåsen
Defence Research Agency, Sweden

Anders Josefsson
Swedish Defence University, Sweden

Christina Ekenstierna
Command and Control Regiment, Swedish Armed Forces, Sweden

Abstract

The conditions for military operations have changed due to, e.g., globalization, climate change, and nations' ambitions and actions. This has resulted in new demands on command and control (C2) capability. Further, the rapid evolution of information technology has provided vigorous opportunities to enhance the C2 capability, e.g., through advanced communication, information management, and decision support. However, the need to rely on modern technology also causes increased vulnerabilities. The sociotechnical nature of C2 systems means that the development of C2 systems is complex and challenging. Developing C2 concepts requires collaboration between people from different knowledge disciplines, traditions, and perspectives. Therefore, there is a need for elaborated concept development approaches and structures that promote collaborative efforts. The objective of this paper is a framework for the development of C2 concepts that enhance the collaboration of people from different traditions. The study was carried out as case study performed in two steps: theoretical development and formative evaluation. The case study targets the development of C2 concepts for future military operations of the Swedish Armed Forces. The framework includes terminology models, a development process, and system representations. The case study shows that in diverse teams, it is essential to agree upon terminology, development process, and systems representations used for the development to avoid misunderstandings and unnecessary rework. The framework explored in this paper is only in its first version. However, the development and the application of the framework was found to facilitate and focus the work of the multi-disciplinary team.

Keywords: Command and control, concept development, design logic, systems engineering, framework

Introduction

The feeling of the cold war is back, albeit in a different form with several superpower nations instead of two, a more tangible gray zone, and hybrid warfare challenges (Pogoson, 2018; Wirtz, 2017). Further, the conditions for military operations have changed, e.g., due to globalization, climate change, technological advancements, and ambitions and actions of different states (Ministry of Defence (UK), 2015). The competing states are devoted to ensure their sovereignty, borders, national interests, and prosperity. A key-enabler for states to defend their interests is to possess an adequate military capability, which has accelerated the procurement of military equipment (Stoltenberg, 2016). Hence, it is costly to uphold a competitive military capability.

Command and control (C2) is a vital *function* of military systems (van Creveld, 1985). It provides *direction* and *coordination* to the military efforts in order to produce military effects (Brehmer, 2007). According to Alberts and Hayes (2006) C2 is "about focusing the efforts of a number of entities (individuals and organizations) and resources, including information, toward the achievement of some task, objective, or goal." Ministry of Defence (UK) (2017) defines the purpose of future C2 as "[to] provide focus for individuals and organisations so that they may integrate and maximise their resources and activities to achieve desired outcomes." Further, Barry (2003) state that agility, flexibility, and responsiveness will be key abilities in NATO's future C2 systems.

The execution of military operations is complex and demanding (Brehmer, 2011). There, C2 is about handling uncertainty (Brehmer, 2013), which requires the ability to:

- achieve a combined affect,
- handle C2 inherent uncertainties, and
- produce effect faster than the opponent.

The C2 function is provided by so called *C2 systems*, which are characterized by complexity due to that the decision-making is performed in constantly changing environments with unpredictable events and based on contradicting information (Brehmer, 2013). This requires a C2 system design that allows adaptation to unpredictable situations by creating boundaries for an acceptable behavior (Rasmussen, Pejtersen, & Goodstein, 1994). In order to efficiently maintain a military capability, an adequate C2 system that uses the available resources strategically and efficiently is required (Barry, 2003). For this reason, governments and military organizations spend considerable amounts of resources on development, procurement, maintenance, and management to ensure sufficiency of C2 systems (Carroll, 2012). At a general level, military C2 systems are required to direct and coordinate the engagement systems so that the required effects is achieved that solve assigned assignments and tasks (Swedish Armed Forces, 2016b). Brehmer (2013) defines a C2 system as: “an artefact, which people designed to get support and help in managing a task.” In this case, it is the task of achieving direction and coordination for an operation in order to achieve its goal.

Sociotechnical design is a direction of interest for development of C2 systems, which follows of the fact that C2 systems are sociotechnical systems (Pilemalm, Lindell, Hallberg, & Eriksson, 2007, Walker, Stanton & Jenkins, 2017). The term sociotechnical stems from organizational and clinical psychology, in work carried out by the Tavistock Institute in the 1950s and 1960s (Emery, 1978). The aim of the sociotechnical school is to balance the *social* and the *technical* parts of systems (Mumford, 2006). There the technical parts are adopted to suit the preconditions of people, process, and organization. At the same time, people’s competences, work processes, and organizing are adopted to take advantage of the possibilities that the technology provides. Davis, Challenger, Jayewardene, and Clegg (2014) suggested that in the development of sociotechnical systems, aspects such as goals, people, processes, culture, technology, buildings, infrastructures, and their relationships should be considered.

To view a phenomena or an artifact as a *system* provides a measure to *understand* how it works, *predict* outcomes based on inputs, and *control* its behavior. There are numerous definitions of the term *system* (ISO/IEC/IEE, 2010). Von Bertalanffy (1968) defined a system as “a set of elements in interaction.” Other definitions implies that “elements in interaction” is not enough, elements need to be organized to accomplish a specific function or goal to be seen as a system (ISO/IEC, 2002; Brehmer, 2007). Systems consist of two or several components which themselves can be a system or an indivisible part (ISO/IEC/IEE, 2010). Further, systems exist in a context, which also can be seen as a system (Hallberg, Jungert, & Pilemalm, 2014) (Figure 1). In development of systems, it is vital to clearly define the scope of the system that should be developed. Hence, a clear border between the system and its context is essential (Sommerville, 2016). To distinguish between the system that is in focus and other systems, the terms *systems-of-interest* and *system-in-focus* are used. The system context includes the stakeholders of the system, and the physical and social environment in which the system will be operative (ISO, 1998).

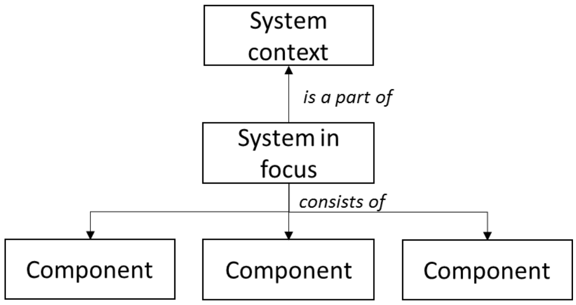

Figure 1. A system exists and operates in a context. The system consists of two or more components.

Systems can be seen as assembled of other systems, which is denoted systems-of-systems (SoS) or federations of systems (FoS) (Maier 1998). A SoS is a system whose components are self-sufficient systems, which can operate independently. The components in a SoS can, thereby, be handled separately (Maier 1998). In FoS, each system selects to participate in the federation or not (Sage and Cuppan 2001). FoS are characterized by significant autonomy, heterogeneity, and geographic distribution or dispersion (Sage & Cuppan, 2001).

The diversity of activities in development of complex systems requires that competences from different traditions are involved, including representatives of the users (Seffah & Metzker, 2004). However, the lack of understanding of and respect for other disciplines has been identified as an obstacle for efficient collaboration (Davis, 2013). An additional challenge is that the meaning of terms might differ between different disciplines. Consequently, the same term can be used with different meanings and different terms can be used for the same phenomena (Kaindl & Svetinovic, 2010; Hallberg, Timpka, & Eriksson, 1999).

To develop complex systems such as C2 systems is a challenging task that requires collaboration between people with different skills, provided from different knowledge traditions. However, people from different traditions of development may experience it difficult to collaborate due to differences in terminology, development methodology, and system representation. The objective of this paper is a framework for the development of C2 concepts that enhances the collaboration of people from different traditions. The construction of the framework, is based on exploring the two development traditions design logic and systems engineering.

Background

This chapter presents the theoretical basis of the study: design logic, systems engineering, and systems representations.

Design logic

Design logic is a tool for understanding existing C2 systems and for constructing future ones (Brehmer, 2007). It was developed by Brehmer (2013) inspired by work of Rasmussen (1984) about abstraction hierarchy and the work by Simon (1996) regarding theory and methods in order to understand how to use artifacts to achieve goals.

Design logic is a five level abstraction hierarchy (Figure 2) in which the first level, *Purpose* responds to the question *why* the system is needed, which provides the rational for designing the system. The second level, *Design criteria* responds to the question *in what way* the purpose should be fulfilled. This level represents the requirements on *the system to be developed*, imposed by *the external system* (responding system). The third level, *Function* responds to the question *what*, by describing what the system as an artifact must be able to produce, in terms of products or services, in order to fulfill the purpose. The fourth level, *General processes* responds to the question *what can be used*, by describing the constraints and opportunities regarding available knowledge, e.g., technology, command requirements, command possibilities, command culture, legal requirements, and C2 theory (Brehmer, 2009). The fifth level, *Form* responds to the question *how*, by describing the implementation of selected functions. Design logic can be applied to any system, not only C2 systems, and it enhances the understanding of the systems by providing different perspectives. When analyzing a new C2 system, it is convenient to express the fifth level, *form*, in terms of doctrine, methods, organization, personnel, and technology, before defining the requirements of specific units and technical systems.

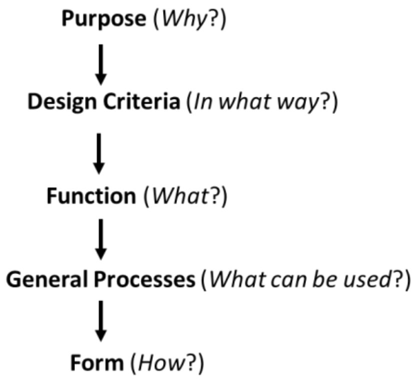

Figure 2. The five conceptual levels of the design logic (Brehmer, 2013).

Cognitive Work Analysis (CWA) has been composed as a framework for analysis of complex sociotechnical systems, inspired by the work of Simon (1996). The initial phase of CWA is the *Work Domain Analysis* (WDA) aimed to provide an understanding of the domain, by exploring constraints that govern the purpose and the function of the systems under analysis. WDA is used to define five constraints: (1) functional purposes, (2) values and priority measures, (3) purpose-related functions, (4) object-related processes, and (5) physical objects (Naikar, Hopcroft, & Moylan, 2005). The constraints defined by WDA corresponds to Brehmer's (2013) five conceptual layers of design logic.

Systems engineering

As a phenomena, *engineering* is old and can be traced back to probably 6000 B.C. (Kirby, 1990). The origin of *systems engineering* can be found in the 1940s and Bell Telephone Laboratories (Schlager, 1956). One of the driving forces for the development of the area of systems engineering was the understanding that working components put together as systems might not deliver the expected and desired effect. It is an interdisciplinary approach that is aimed at transforming a set of customer needs, expectations, and constraints into a solution and to support that solution over its lifetime (ISO/IEC/IEEE, 2010). This includes the management and operation of the solutions. Hence, systems engineering covers the management of systems from the earliest ideas to the disposal (NASA, 2007).

To be successful in systems engineering, a holistic view of the system is required. Further, systems have to be explored from the outside as well as from the inside and, thereby, systems engineers need to consider how the system interacts with the environment as well as how the components of the system interact (Kossiakoff, Sweet, Seymour, & Biemer, 2011; Kasser, 2009).

To be successful, a developed system must meet the needs of stakeholders, e.g., users, customers, users, and maintenance and operating staff. Hence, a holistic and coherent understanding of the stakeholder needs, transformation of their needs into requirements, and design of a solution that meet those requirements as are indispensable within development of systems (Hallberg, et al. 1999).

In the field of systems engineering, there is no unambiguous and comprehensive use of terms. Thereby, it is common that developers misunderstand each other, since they interpret and use terms differently (Enquist & Makrygiannis, 1998). These misunderstandings hamper the development.

Several processes for systems engineering has been suggested, such as the Waterfall model (Royce, 1970), Spiral model (Boehm, 1988), Vee-model (Blanchhard, 2008), and Soft systems method (Checkland, & Scholes, 2001). Systems engineering can be performed iteratively, incrementally, or evolutionary (Larman, & Basili, 2003). In iterative development, the process is carried out several times and the system is improved in each iteration. In incremental development, additional functionality and features are gradually developed and incorporated into the system. Evolutionary development is based on processes that are performed both iteratively and incrementally.

Kasser (2013) presents a framework that defines systems engineering as consisting of eight processes: (1) identifying the need, (2) requirements analysis, (3) design of the system, (4) construction of the system, (5) testing of the system components, (6) integration and testing of the system (7) operations, maintenance and upgrading the system, and (8) disposal of the system. An important initial step before the actual development is to define the borders between the system and the system context. Hallberg, Jungert et al. (2014) extended Kasser's framework in their description of the systems engineering process by adding an initial process called *Context analysis*, to emphasize the importance of determining the system's border relative its context.

System representations

There are several different ways to represent systems in the form of models, which should be seen as abstract descriptions of real or virtual phenomena (Buede & Miller, 2016). Different representations fit different purposes. Representations of C2 in the form of models are abstractions used to support understanding of C2. Especially in the case of complex systems, it is fruitful to remove aspects in representations of the systems that are not of interest. Enterprise architecture frameworks such as Zachmans framework, Department of Defense Architecture Framework (DoDAF) and NATO Architecture Framework (NAF) provide a number of different integrated views and perspectives of systems (Zachman, 1987; Schekkerman, 2004). In general, systems can be described from two perspectives: *structural* and *behavioral*. The structure of a system is a static property and consists of the system's components together with the relations between them. Behavior is a dynamic property and denotes the effect the system has on its environment when it is in operation.

Structural models

Structural models are used to describe the components of a system and the relationships between these components. Examples of such representations are hierarchical models and organizational charts (Tague, 2005). The structural models commonly provide an internal representation of the systems, but can also be used to view the relationships between the system and its context (Davis, Challenger, Jayewardene, & Clegg, 2014).

The Viable Systems Model (VSM) is a recursive structure based on cybernetic principles and organization theory, claimed to be suitable for modelling complex systems (Beer, 1981). The VSM represents a system of systems perspective in different levels. On each level, there is a control component connecting the system of interest to the control components of both superior and subordinate systems. The control component of the system of interest also connects to the environment in order to gather the information it is intended to act upon. The control system then decides how to use the resources of the system of interest in order to influence the environment. In each control component of a system of interest there are functional elements handling the system elements. These functional elements are: (1) coordination of all the system components in the system of interest, (2) internal control of the system of interest, (3) direction, and (4) information management within the system of interest, information, and intelligence gathering from the environment and communication to and from: (a) Control component elements, (b) Superior system level, and (c) System components of the system of interest.

Simon (1996) introduces the concepts of the *outer system*, the *inner system,* and the *interface* between the two systems (Figure 3). It is the outer system that should be impacted upon in such a way that desired effects are achieved and it is the internal system that will enable this by designing an interface that creates this effect (Flood & Carson, 1993). Hence, the purpose of the inner system is to be found in the outer system. Lawson (2010) on the other hand presents a system coupling diagram with three systems (Figure 3). Situations that require some sort of impact can be regarded as a set of cooperating elements. These cooperating elements are represented as a *situation system*. In order to interact with the situation system, the *respondent system* is created by the composition of a suitable set of elements from a pool of *system assets*. This pool of assets can be constituted by one or more systems. One of the system elements in the respondent system has to be a *control element* that receives input from the situation system and interacts with the respondent system to reach the desired impact on the situation system. Both the respondent systems and the pool of system assets are sociotechnical systems.

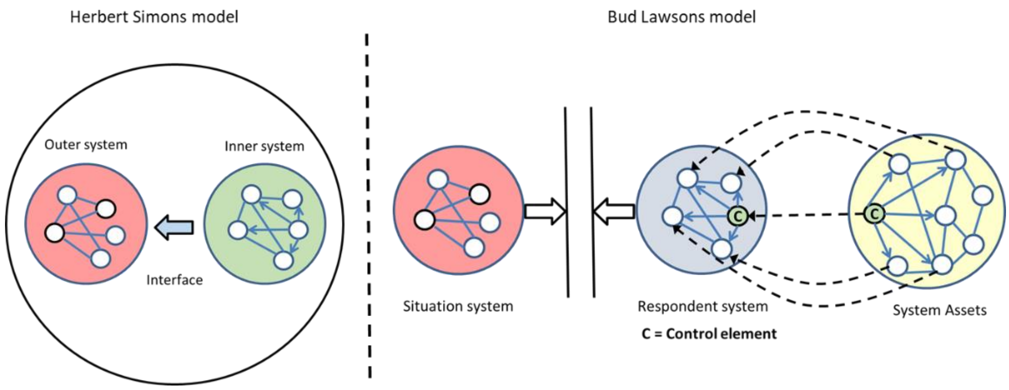

Figure 3. Herbert Simon's and Bud Lawson's system models adapted from (Lawson, 2010; Simon, 1996).

Behavioral models

Behavioral models describe how systems behave, e.g., as flow of activities or chain of functions that transform input to output. Examples of behavioral models are activity diagrams, process models, sequence diagrams, and use cases (Eriksson & Penker, 2000; Hallberg, Andersson, & Ölvander, 2010). These models can be used to represent systems behavior, both from an internal and an external perspective, e.g., activity diagrams and process models can be used for the internal behavior, while use cases can be used for the external behavior (Hallberg, Andersson et al. 2012).

There are several models that describe how C2 should be carried out. Boyd's Observe, Orient, Decision, Action (OODA) loop is probably one of the most well-known (Boyd, 1987). Several models extends the OODA loop such as The Cognitive OODA and Dynamic OODA (Rousseau, & Breton, 2004; Brehmer, 2005). In other areas, similar loops exist, e.g., the Plan–Do–Check–Act (PDCA) which is formulated for management and quality approaches. (Sokovic, Pavletic, & Pipan, 2010). PDCA has been seen as a vital measure in the approach named *continuous improvements* (Bhuiyan, & Baghel, 2005).

In their work regarding systems support for local emergency management and crises response, Hallberg, Hallberg, Granlund, and Woltjer (2014) described C2 as a continuous process including the activities: (1) assess the situation, (2) plan actions, (3) make decisions, and (4) activate resources. It is, however, argued that in C2 each activity has to be ongoing continually (Brehmer, 2005). Hence, one activity does not end, just as the next is initiated and, therefore, the relations between the activities in C2 systems should not be interpreted as flow of work, but rather as a flow of information artifacts that can be seen as products and services. Brehmer (2010) states that functions are means of analyzing what C2 systems produce internally without saying how output is produced. Hence, it is the output of a function, i.e., the product that is essential. Therefore, it is possible to view functions as "black boxes", which transforms input to output (Brehmer, 2010). Both design logic and systems engineering traditions encompass modelling notations that describe which functions that have to be included in a system. Such models describe how a product is the outcome of one function and are used by one or several other functions.

Functions can be as "theoretical constructs" of what C2 systems for military operation must perform, based on C2 theory (Brehmer, 2009). There are several sets of functions purposed for C2 systems. Alberts and Hayes (2006) claim that the following functions are essential for C2: establishing intent; determining roles, responsibilities, and relationships; establishing rules and constraints; monitoring and assessing the situation and progress; inspiring, motivating, and engendering trust; training and education; and provisioning. NATO ACT (2016) describes C2 as carried out by the following functions: connecting, sensing, processing, sense-making, deciding, and effecting. Meanwhile the Ministry of Defence (UK) (2017) sees the following functions as essential in future C2 systems: creating shared awareness, allocating resources to create effects, assessing progress, and recognizing the need to change the approach to C2 and/or the plan of action.

Josefsson, Marklund, and Hanson (2007; 2008) suggested a C2 concept for integrated, dynamic C2 (IDC2) within the framework of a Network Based Defence, which has been implemented in the Swedish Armed Forces on the tactical level. IDC2 consists of four basic elements: monitor, evaluate, plan, and coordinate. The basic elements are supported by information management to achieve C2. An adapted version of IDC2 has been introduced at the Swedish military strategic and operational level as the *Swedish method for planning and C2* (SPL) (Josefsson & Lavman, 2017; Spak, 2017). SPL consists of three activities: estimate to aim for decision about direction; plan to aim for coordination, monitor to aim for continuous situation awareness; and assess to aim for understanding of the situation in relation to the mission. Further, SPL contains two supporting activities: information management and communication.

Methods

The study was carried out as a case study performed in two steps: theoretical development and formative evaluation (Yin, 1994; Hallberg, Pilemalm, & Timpka, 2012). The theoretical development was based on exploring the traditions of design logic and systems engineering.

Three issues in the theoretical development were considered to enhance the collaborative work of C2 development:

- To define a common language, with unambiguous definitions of terms related to the development of C2 systems.
- To define a light weight process for development of C2 concepts.
- To define system representations that enhance the development of C2 concepts.

In the formative evaluation, outcomes of the theoretical development were used for the development of a generic C2 concept for the Swedish Armed Forces. Findings and experience was documented in reports, notes and models. The development of the C2 concepts, initiated early 2016, was carried out through one-day workshops on a regular basis. During the workshops both development of the generic C2 concept and improvements of the approach for the development was performed. The development was carried out by a core team. Four times per year, the work of the core team was presented and discussed with experts, both from military and academia. The core team consisted of four persons, two female and two males. Two of the participants have military ranks. Two of the persons in the core team were educated and trained in the tradition of systems engineering, one trained in the tradition of design logic, and one in both. All four have long experience of being engaged in development of approaches for engineering of C2 capabilities and systems.

Results

The results of this study are presented in relation to the following three aspects: terminology, development process, and systems representations. Each aspect is described regarding the rational for its theoretical development, and the implementation and formative evaluation.

Terminology

The initial idea was to come up with one shared comprehensive terminology for collaborative development of C2 systems. This to enhance the collaboration and ensure the avoidance of misunderstanding that would hamper the development of the C2 concepts. However, this was not achievable. Instead, models that present the definitions and relationships of the terms was constructed.

Implementation:

Initiating the collaborative development of C2 concepts, the need for a common language in the core development group was seen as vital. However, as the work proceeded it was found difficult to agree up on a shared terminology. After long and time consuming discussions, a redirection of the attempt was made, to increase each other's understanding for the other tradition's use of terms. This was done by exploring how the different terms related to each other. For that purpose, three models were constructed to describe the interpretation of the core terms of the two traditions. The first model relates the core terms used in design logic (Figure 4). The second model relates the core terms used in systems engineering (Figure 5). The third model relates the terms of the two traditions (Figure 6).

The core terms in design logic includes: *purpose, design criteria, functions, general processes,* and *form*. The model extends these core terms with the terms *situation systems* and *respondents system*, which are essential for defining the purpose (Figure 4). The *purpose* describes *why* the system exists and is to be found in the situation systems and respondent systems. The *design criteria* are derived from the purpose and describes *in what way* the system and its functions must fulfill the purpose. The *functions* are abstract components that describe *what* the system should produce. The *general processes* are constraints of how the form can realize the functions and can be expressed as best practices in different knowledge areas. The *form* presents *how* the system should be or is implemented.

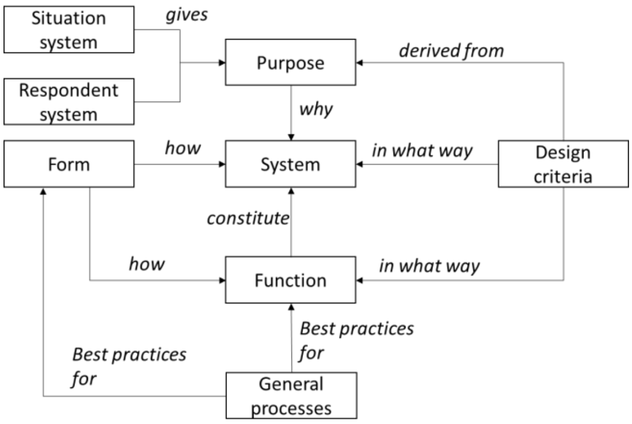

Figure 4. Interpretation of how the core concepts of design logic are related to each other, to the system in focus, and to the situation system and the respondent system.

The following definitions are grounded in the development of an ontology of definitions of system engineering terms (Hallberg, et al. 2014). The core terms of relevance for the work in this paper are: *needs*, *requirements*, and *design*. The model also extends the core terms with the related terms: *context*, *stakeholders*, and *statements* (Figure 5). The *context* is the environment in which the system will be operative. *Stakeholders* are part of the systems context and consist of those who are affected by, or are able to affect, the system. *Statements* can be collected from the stakeholders and are expressions that contain information that could be used in the development. They can consist of descriptions of the stakeholders, the actual or wanted situations of use, perceived problems, business goals, and visions. *Needs* are identified based on the statements and are a condition or capability needed by stakeholders to solve a problem or achieve an objective. *Requirements* are defined based on the needs and are a specification of *what* systems should accomplish. These requirements include *functional requirements*, *non-functional requirements*, and *domain requirements.* Functional requirements specify what systems must produce, while non-functional requirements specify what qualities the system must possess in order to perform well. *Domain requirements* specify requirements that are inherent of the operative context, e.g., laws and regulations. *Design* is created based on the requirements and specifies *how* functionality and features in the system should be implemented and realized, i.e., how the requirements should be fulfilled.

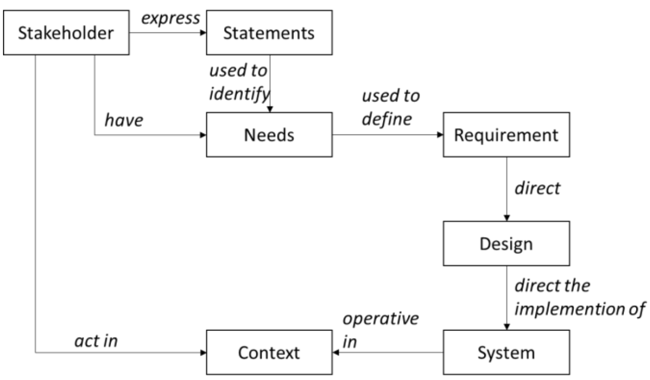

Figure 5. Interpretation of how the core concepts of systems engineering are related to each other, and the system in focus.

In the military C2 domain, situation systems and the responding system constitute the systems context (Figure 6). The purpose corresponds to the C2 needs and the design criteria corresponds to the requirements. The design defines functions and form, and the general processes constrain the design, by providing best practices.

Formative evaluation:
To agree upon on a common language was, initially, seen as a necessity by the core team. Ideally such work should result in a universal shared and agreed upon language for C2 development. However, the discussions of the different definitions became time-consuming and endless without getting closer to the desired common language. The way forward was found in exploring how the terms in the different traditions related to each other and expressing that in a model (Figure 6). Then presenting the C2 concepts to people outside the core team, the need for being able to use the language of both traditions for the communication become even more evident. Hence, it was found that the core group needed to be skilled in both of the tradition's terminology to express themselves. Further, both the work with the models and the models themselves was found to contributed to overcome the language barrier.

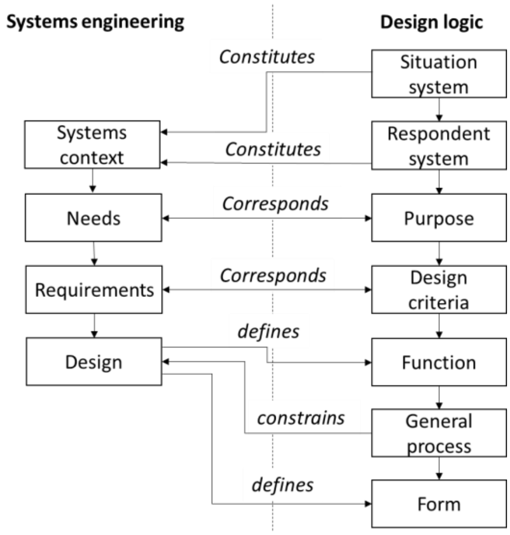

Figure 6. Interpretation of how the terms of systems engineering and design logic are related to each other. In the design logic track the situation system and the respondent system are included, which not is included in Brehmer's (2013) design logic.

Development process
There is a vast amount of different development processes available in the literature. For the development of C2 concepts, a straightforward and light weight process, which the team had experience of was selected. The initial development process used was based on a traditional process including: context analysis, needs assessments, requirements analysis, and design (Hallberg, et al. 2014). This process was supplemented by an additional activity aimed at the exploration of *Existing knowledge* in order to make use of work already performed regarding warfare, C2, C2 systems, and C2 concepts. The existing knowledge is in the design logic referred to as general processes.

Implementation:
The development was carried out in six activities: (1) Exploration of existing knowledge, (2) Context analysis, (3) C2 needs assessment, (4) C2 requirements analysis, (5) C2 concept design, and (6) C2 concept evaluation (Figure 7).

Exploration of existing knowledge is aimed at identifying knowledge that contributes to the development of the C2 concept. This includes knowledge of previously performed C2 needs assessments, C2 requirements engineering, C2 concepts, C2 principles, and warfare principles. *Context analysis* aims at getting an understanding of the operational prerequisites, including own units and resources. The outcome of the context analysis is statements, which express, e.g., challenges, demands, and possible operational solutions related to conducting future military operations. *C2 needs assessment* aims at, based on the statements collected in the context analysis, identifying and documenting C2 needs, which express the C2 needed by the operative units. Thereby, the C2 needs constitute the detailed purpose of the C2 system. *C2 requirements analysis* is based on the C2 needs and aims at, defining how (in what way) the C2 system's products and services will meet the needs of the system. C2 requirements encompass for example time frames, operating environment, conflict level, protection, and logistics. C2 requirements are also used to define the requirements on the C2 system components: doctrine, methods, organization, personnel, and technology. *C2 concept design* aims at, based on the C2 requirements and existing knowledge, design of C2 concepts. *C2 concept evaluation* aims at validating the concept by, e.g., presenting it to domain experts and applying it in relevant scenario-based games. The C2 concept is an idea of how a C2 system should be designed to best fulfill the needs from a foreseen operating environment with its problems and opportunities. The C2 concepts in this work were described with respect to: the main idea of the concept, how it fulfills the principles of C2 and warfare, and the functions required to produce the required products and services.

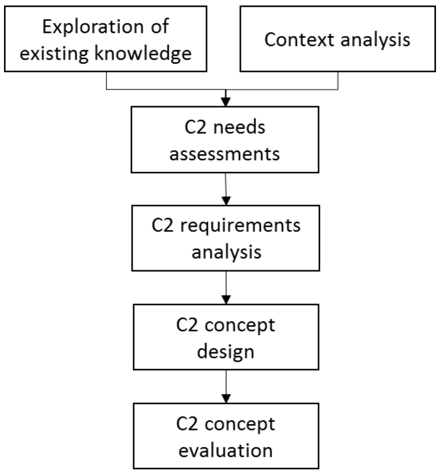

Figure 7. The initial development process, which guided the work of developing C2 concept.

The development process was carried out in three iterations resulting in four concepts. Each iteration was carried out completely differently due to project preconditions. The *first iteration* was aimed at generating two alternative concepts, called *Concept 1* and *Concept 2* (Figure 8, iteration 1). The work was focused on exploring existing knowledge, scrupulously assessing of the C2 needs, designing two alternative concepts, and evaluating those concepts to choose one for further development.

The *exploration of existing knowledge* was based on a literature review, which resulted in eight C2 principles, twelve warfare principles and six C2 functions (Granåsen, Hallberg, Josefsson, & Ekenstierna, 2017). Examples of C2 principles are: *C2 demands situation awareness* and *C2 system shall continuously be able to adapt to the situation*. Examples of warfare principles are *determine and hold on to objectives* and *act in such a way that means surprise for the opponent*.

During the *context analysis* in the first iteration scenario-based games were utilized to collect and explore information that could be used to determine C2 needs and C2 requirements regarding a future C2 concept. Scenarios were derived from the content of the Swedish government's latest *Defense Policy Bill* (Regeringen, 2015) and it was assumed that a sufficiently large part of the potential future possibility space was taken into account. All games were based on simple but unique scenarios. The scenarios were organized according to strategy, war deterrence (WD), and war fighting (WF) (Table 1). The main variable was strategy and could have the dimensions of offensive or defensive and war fighting or war deterrence (Edström & Josefsson, 2016).

Table 1: Categorizing of the scenarios of the games.

Strategy	**War Deterrence (WD)**	**War Fighting (WF)**
Offensive (O)	WD + O (Game 1, 160308)	WF + O (Game 3, 160524)
Defensive (D)	---------------------------	WF + D (Game 2 160405)

The participants in the game consisted of teachers and managers at Swedish Defence University (SEDU) whose skills were linked to strategic, operational, and tactical (land, maritime, air) command levels. Each game was divided into two phases: one focused on the operations execution perspective and the other focused on related C2 perspective (Swedish Armed Forces, 2017).

The *C2 needs assessment* was based on literature studies, interviews with high ranked Swedish military commanders, and the resulting information of the scenario-based games in the context analysis. The outcome of the needs assessment was C2 needs in eight categories of C2 needs, with several more detailed C2 needs in each category. An example of a C2 needs category is: *the need for C2 of operations in different environments*, which is detailed by 14 C2 needs such as *the need for C2* of *operations in urban environments*. The C2 concept design was to design two distinguishing C2 concepts, *Concept 1* and *Concept 2* that meet the C2 needs.

The main idea of *Concept 1* is that each unit has a wide range of capabilities. Concept 1 is characterized of uniform C2 with the ability to adapt the C2 method according to the situation. The broad range of capabilities makes it possible to change between direct command and mission command. Resilience is based on each unit having a wide span of capabilities and ability to carry on with their task even during loss of communication with its higher command. The concept entails that each unit can be trained together and can solve a variety of assignments on their own. However, there will be a risk of sub optimization since resources tied to a unit might be of better use elsewhere.

The main idea of *Concept 2* is based on highly specialized units where the required interoperability in order to access and utilize resources is based on services (Pilemalm & Hallberg, 2008). When needed, resources are integrated in temporary organizations in order to solve given assignments and benefit from emerging opportunities. When an assignment is completed, the temporary organization is dissolved. C2 can change between direct command and self-organization based on local decisions without involvement of a higher command. The outcome of the evaluation of the concepts led to an understanding that none of the concepts fully met the needs of the Swedish Armed Forces.

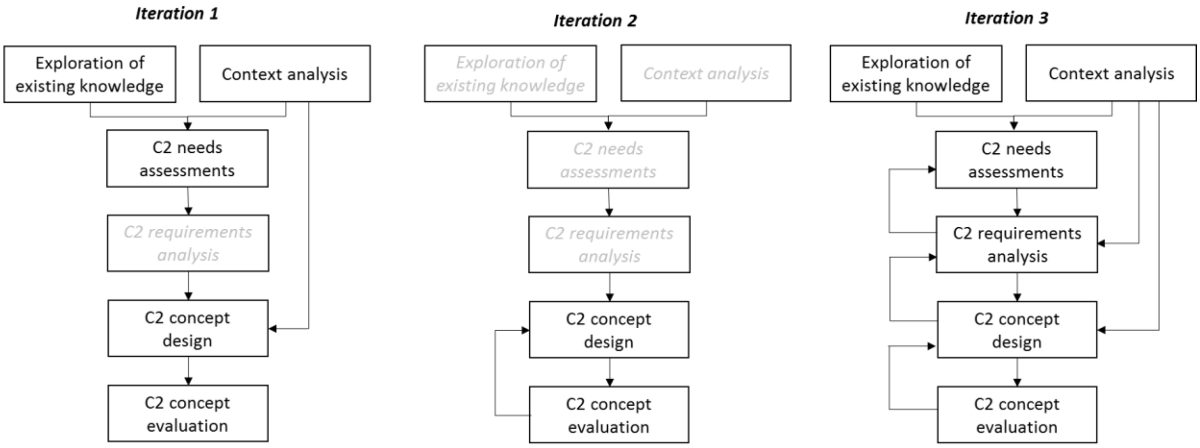

Figure 8. How the process was carried out in each of the three iterations. Iteration 1 resulted in Concept 1 and Concept 2, Iteration 2 resulted in Concept 3, and Iteration 3 resulted in Concept 35.

Since none of the two concepts was found to fully meet the Swedish Armed Forces' need of future C2 system a second iteration was performed. The aim of *iteration 2* was, based on the result of the evaluations of Concept 1 and Concept 2, to generate and evaluative a third concept, *Concept 3* (Figure 8, Iteration 2). This iteration therefore contained the steps concept design based on the previously made evaluation, followed by an extensive concept evaluation of Concept 3. The third concept prescribes C2 functions that achieve effect by utilizing own C2 resources together with resources of other units', by using service procedure, as a compliment. The C2 concept should enable a dynamic and agile organization that uses its resources in the most effective way in each situation. In Concept 3, C2 is performed based on an effect perspective rather than on a resource perspective.

The *third iteration* aimed to improve and detail Concept 3 (Figure 8, Iteration 3). This iteration contained all the development activities in the suggested development process. The context analysis was complemented based on the operational environment described in the Swedish Armed Forces perspective study that brings forth the Swedish military strategic concept for year 2035 (Swedish Armed Forces, 2018). Additional C2 needs were identified based on the military strategic framework and the proposed military strategic concept describing how military efforts can be executed in 2035. Examples of additional C2 needs where: *need to lead operations to oppose an attacker's denial zone*, *need to lead long-range precision targeting against an enemy´s critical vulnerabilities* and *need to lead operations conducted with systems in collaboration at multiple command level*s. The identified C2 needs were incorporated with the C2 needs from the previous iteration 1. The updated set of C2 needs were used to redefine C2 requirements, which were used to define requirements regarding doctrine, methods, organization, personnel, and technology. The outcome of the third iteration was named *Concept 35* based on the main idea that C2 capability encompasses achieving effect by commanding own resources and use others' resources through services. C2 will allow a dynamic and agile organization, which uses its resources in the most efficient way in every given situation. C2 is conducted with an effect perspective.

Formative evaluation:

The development process started with a minimalistic and straightforward approach. However, during the work it was found impossible to be completely loyal to the process for two reasons: (1) Limitations of time to develop the C2 concepts and (2) that new and more insightful information regarding future military operations was continuously added. Therefore, the activities C2 needs assessments, C2 requirements and C2 concept design became an iterative ongoing process as a whole (Figure 9). Further, the C2 needs were not found to be sufficiently informative to be used to define distinct C2 requirements. Beside the C2 needs, deep insight about military context was found necessary to define requirements.

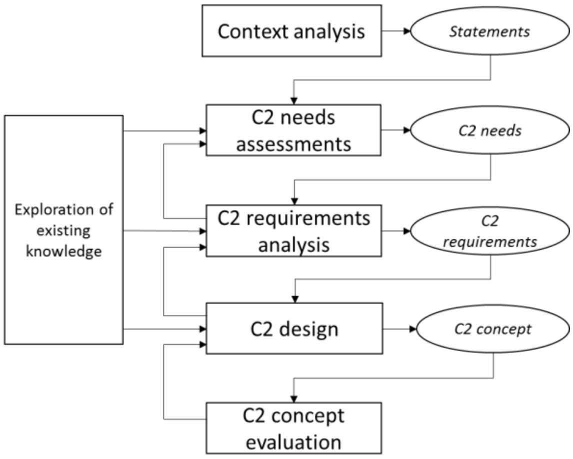

Figure 9. The iterative development process.

Systems representations

The situations to be met by respondent systems are commonly complex. According to Yates (1978) the complexity is denoted by five attributes: (1) significant interactions, (2) high number of parts, degrees of freedom, or interactions, (3) nonlinearity, (4) broken symmetry, and (5) nonholonomic constraints. Systems should be considered complex, if they fulfill one or several the attributes. The attributes of these systems opens up for a variety that has to be taken care of in military responding systems. Regulating a complex situation system requires a responding system with at least the same complexity as the regulated system according to Ashby's law of Requisite Variety (Ashby, 1958). Hence follows that regulating complex responding systems consequently requires complex control systems.

Implementation:

The case study applied four views as system representatives: (1) operative view, (2) recursive view, (3) functional view, and (4) systems elements view. These views are needed to understand the C2 context, to define the C2 borders, what the C2 system has to produce, and guide how it should be built.

The operative view combines characteristics from both Simon´s (1996) inner and outer systems model and Lawson´s (2010) system coupling diagram (Figure 10). The operative view contains the operative challenges or problems that have to be addressed within the situation system (e.g. the enemy) with the respondent system (own resources), which contains the armed forces with its different services. The respondent system creates effects in a situation system. The C2 system gives the respondent system continuous direction and coordination. C2 needs can be deducted from the course of action that the respondent system needs to execute in order to create desired effects in the situation system. The C2 system should be designed to meet the C2 needs and to produce what Ashby (1963) called the "requisite variety" to match the respondent system.

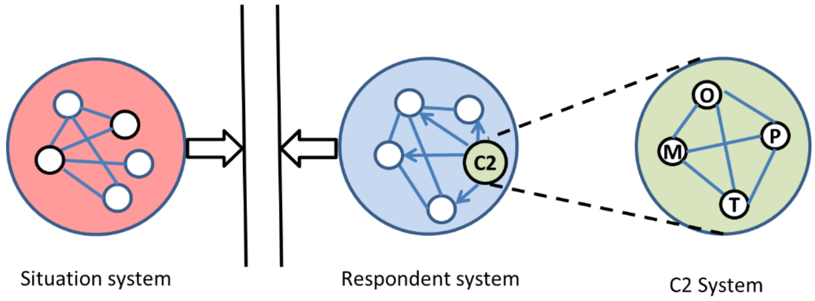

Figure 10. Operative view

C2 systems are not perfectly coherent systems, but rather consist of several interconnected systems that can be found at different command levels and in different services.

A recursive model view influenced by the Viable Systems Model, VSM (Beer, 1981) also makes it possible to study the C2 system's interactions with the respondent system as well as with the situation system where the given tasks are performed in order to create the desired effects (Figure 11). The recursive model depicts a respondent as a system-of-systems with different levels of command. In each level of these systems (system-of-interest), there is a C2 system connecting the system-of-interest to the C2 systems of both its superior and subordinate systems. The C2 system connects to the situation system (often via an intelligence system with different sensors) in order to gather the information to act upon. The C2 system then decides how to use the resources of the respondent system in order to influence the situation system. Since the different levels of command may differ between nations or organizations, we have used the most common classification of these command levels:

- Strategic level
- Military strategic level
- Operational level
- Tactical level
- Units

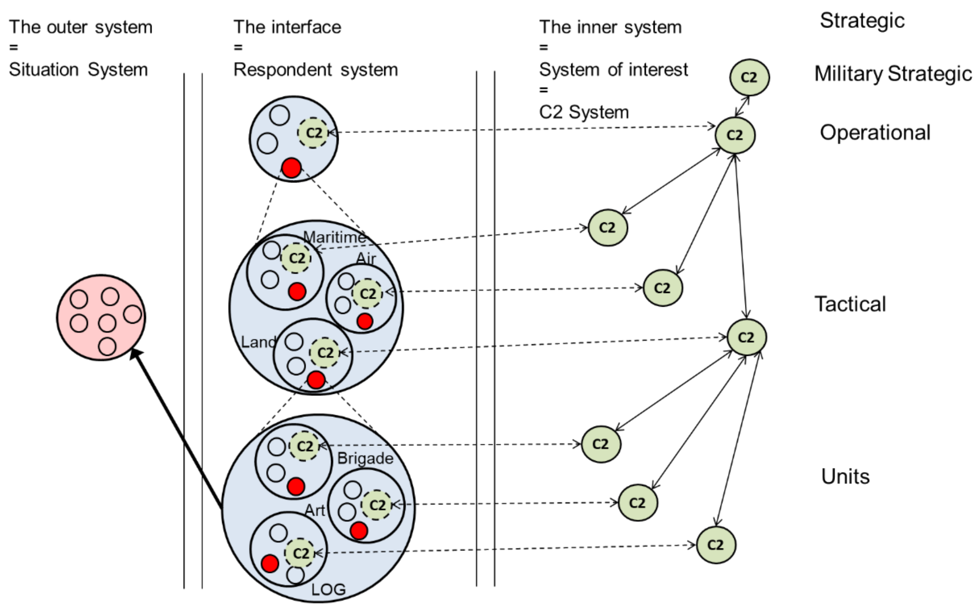

Figure 11. The recursive view

It is on the units' level in this system of systems model that the interaction occurs with the situation system where the desired effects are to be created.

In order to design the C2 system, five functions were included to answer the question of *what* the C2 system needs to do to achieve the C2 needs (purpose). The functions that are needed to provide the purpose of C2, namely direction and coordination, are: Data providing, Assessing, Estimating, Planning, and Communicating (Figure 12).

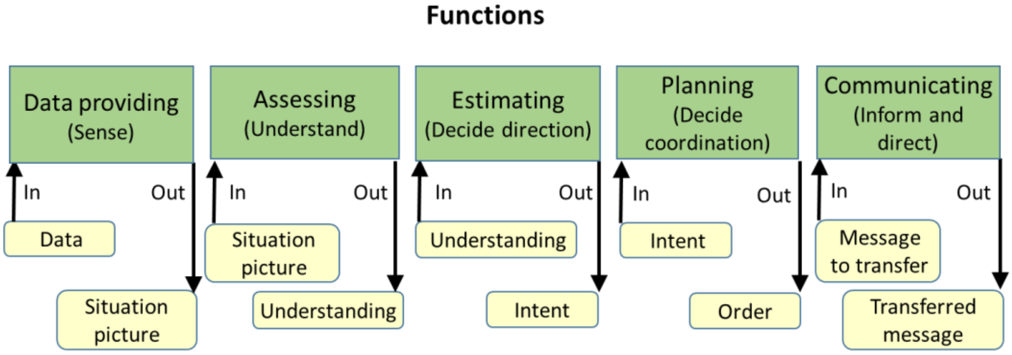

Figure 12 The functional view presents the five functions included in the C2 concepts. These functions are needed to understand what products the C2 systems have to produce in order to accomplish its purpose to direct and coordinate military operations.

The *data providing* function produces a *situation picture*, by *sensing* changes in the situation and thereby triggers the whole C2 process. The *data providing* function also receives orders from higher command levels, which also can be seen as changes in the situation. The input to the data providing function is *data* and the output is a *situation picture* that describes the operational situation encompassing the operational environment (situation system) and own forces (respondent system).

The *assessing* function produces an *understanding* of what the situation means to the respondent system due to its role and its overall mission. The input to the assessing function is a *situation picture* and the output is an *understanding* of what the situation means.

The *estimating* function produces a *decision of direction* of what the respondent system should do to fulfill the mission according to the understanding of the situation and create desired effects in the situation system. The input to the *estimating* function is the *understanding* of the situation and the output is an *intent* that describes what has to be done to fulfill the mission based on the understanding of the situation.

The *planning* function produces a *decision* of *coordination* of the elements of the respondent system. Coordination takes place in time and space, in order to meet the decided direction and to create desired effects in the situation system. The input to the *planning* function is the *intent* that describes what is to be done and the output is an *order* to the proper C2 element in the respondent system.

The *communicating* function informs and directs other elements of the system according to the produced situation picture, the understanding of the situation, the decided direction of the mission as an intent or an order that coordinates the elements of the respondent system. The input to the *communicating* function is a *message to transfer* and the output is a *transferred message*.

C2 systems are sociotechnical systems and the form elements that can provide needed output from the C2 functions are: C2 methods (including doctrine), C2 organization, C2 personnel, and C2 technology (Figure 13). Those elements are used as a foundation for the design of the C2 concept.

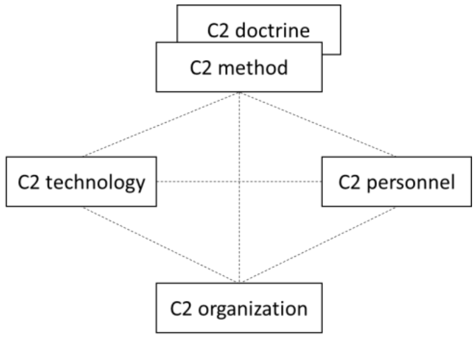

Figure 13. The systems elements view presents the form elements to be used as foundation for the design of the C2 concept.

The systems elements view shows all the elements within the sociotechnical system that can produce the different products and services needed from the C2 functions for the C2 system to fulfill its purpose, which is to direct and coordinate a military operation. The elements C2 method, C2 organization, C2 personnel, and C2 technology are chosen since they are the easiest form of describing a C2 system. Another important reason for choosing these form elements is that the Swedish Armed Forces (2016a) uses them for C2 systems.

Formative evaluation:

During the case study the operative view, recursive view, functional view, and systems elements view was elaborated and evaluated as support for the development of the C2 concepts. These representations were sufficiently supporting the work for understanding: the context to the C2 system, which C2 systems to develop, which functions are fundamental in C2 systems, and to outline the design of the elements of C2 systems.

Discussions

The conditions for conducting military operations are constantly changing. Currently, due to more tangible gray zone and hybrid warfare challenges together with a rapid technological development (Wirtz, 2017, Ministry of Defence (UK), 2015). Thereby, the preconditions for C2 changes, which demands new concepts for development of C2 systems. The objective of this paper is a framework that supports the development of C2 concepts by enhancing the collaboration of people from different traditions. Development of C2 systems, due to its sociotechnical nature, is complex and challenging, which requires the involvement of people from different disciplines and with different perspectives (Yates, 1978; Pilemalm et al. 2007, Walker et al. 2017). However, people from different traditions have different languages, which might hamper efficient collaboration (Hallberg, Jungert et al. 2014; Davis, 2013).

An initial ambition with the framework was to provide a uniform and comprehensive terminology that enhanced the development of concepts for C2. This was, however, found too difficult to accomplish within the scope of this study. Instead, the focus was shifted to models that helped clarify the meaning of terms used in each tradition and a model presenting how terms relate to terms in the other tradition. The resulting models were perceived to enhance the understanding of each other’s languages and at presentations of the C2 concepts for different audiences, the language used could be adopted. Still, the terminology models presented in this paper should, however, not be seen as an attempt to clearly define the final meaning of the terms. With the lower ambition, the aim was changed to an increased understanding of the terms used by the two traditions in the development of C2 concept.

To develop actual C2 systems requires well elaborated processes, which account for the sociotechnical nature of C2 systems (Pilemalm, et al. 2007). However, as far as the development is restricted to C2 concepts, a straightforward light-weight process was found applicable, even though it had to allow flexibility regarding how and which activities were applied. The tradition of systems engineering contributed with the development process, but the idea of general processes for the design logic was adopted as an activity to make use of existing knowledge during the development. A future improvement of the development process might be to use *work domain analysis* where the design logic supports limiting the complexity of C2 systems (Naikir, et al 2005). When comparing the system's output in form of products and services with the constraints in form of requirements during systems analysis, it can be possible to decide whether a revised set of functions and form elements are required, and if the set of general processes has to be changed.

The general purpose of C2 is to produce direction and coordination to the military effort and the purpose with the military effort is to create effect so the objectives with the whole mission are achived. Therefore, it is necessary to understand both the situation system and how the respondent system is going to behave to create the desired effects whithin the situation system (van Creveld, 1985, Brehmer, 2007). Further, the general purpose of C2, producing direction and coordination, is too abstract to guide the design of C2 systems. Therefore, *C2 needs* was used to operationalize the purposes of specific C2 systems. A valid set of needs provides a necessary foundation for defining the C2 requirements and, thereby, for the development of appropriate C2 systems. However, in the case study, the C2 needs were not experienced as, informational enough to support derivation of a clear solid set of *C2 requirements*. A possible solution to this is to perform an in depth revision of the operational prerequisites for each C2 need supported by aspects such as the possibility space by Waldenström (2016): (1) own mission, (2) own troop's abilities, (3) enemy troop's abilities, (4) enemy troop's plausible courses of actions, (5) terrain, (6) doctrine, and (7) legal constraints. To further enrich the descriptions of the C2 needs, they should be analyzed and documented on the following aspects:

- *When* should direction and coordination be applied?
- *For how long* should direction and coordination be applied?
- *How fast* should changes in direction and coordination be applied?
- *Where* should direction and coordination be applied?
- *Who* should *have* direction and coordination?
- *Who* should *give* direction and coordination?

A valid set of needs provides a good foundation for defining the C2 requirements and, thereby, for the development of adequate C2 systems (Hallberg, Hallberg et al. 2014). However, to process all information and aspects to create a sufficient set of C2 needs, demands that a considerably large amount of information from several different sources, such as future technology development, warfare strategies and tactics, and political development has to be taken into consideration.

In the case study, the development started with a straightforward development process. However, it was found difficult to strictly adhere to the process. However, since the process only prescribes general activities, the implementation of each activity can be based on the situation. The two reasons for the need of this agility are: (1) limitations in available time to carry out each iteration completely and (2) that new and more insightful information regarding future military operations constantly was added.

Systems engineering has its strength in the engineering process and the design logic in setting various constraints on systems in which the system can react and adapt to changes in its context. However, systems engineering and design logic to a large extent follows the same logical development pattern. They share the same idea of what is important and what should be in focus during development of systems. Instead of seeing design logic and systems engineering as two alternative approaches to develop C2 systems, it is more productive to see the relationship as: design logic as a meta-level development that guides how to think when applying systems engineering for development of C2 systems. There, the purpose in design logic states that C2 needs are limited to scope of direction and coordination of a military operation and that they ought to be derived from the systems context that consists of the situation system and the respondent system. The design criteria in the design logic means that the C2 requirements should be seen as constraints, which do not impose stricter requirements than justified by the needs. The general process in the design logic states that when designing the C2 system, in form of established functions, best practices, available technology, command culture, legal requirements must be considered in addition to the C2 requirements. Hence, those traditions should be able to smoothly elaborate and enrich each other and thereby enhance the development of future military C2 systems.

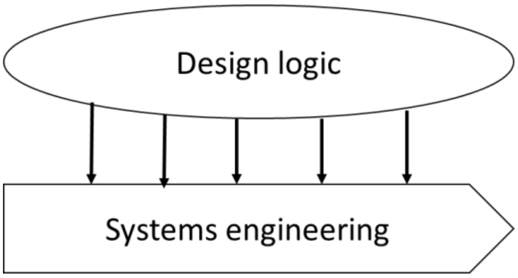

Figure 14. Design logic could be used as a means to guide how systems engineering should be applied to conduct C2 systems development.

Conclusions

To define a unified language, to be used by people of different traditions, requires ambitious agreement on definitions of terms, which is difficult to achieve. The work presented in this paper circumvents this problem by providing models that bridge the language barrier between design logic and systems engineering. It was found to be a more feasible approach to create a bridge between the two traditions' terms, rather than trying to develop a uniform language. Future work is needed on the language models, but for the development of C2 concepts suggested models did work satisfactorily.

Many processes for development have been proposed over the years. In this study, a plain process was adopted to fit the development of the C2 concepts. It is based on six activities: (1) exploration of existing knowledge, (2) C2 context analysis, (3) C2 needs assessment, (4) C2 requirements analysis, (5) C2 concept design, and (6) C2 concept evaluation. This process needs to be further developed with methods and tools that support each activity. However, the process worked for the development of C2 concepts, as long as a flexibility was permitted in how the process was applied. Exactly how the development is performed is not critical. The important thing is to understand the logic of the development, e.g., that C2 needs are found in the situation system and the respondent system. The study contributes with four views of C2 systems, which in the work provides a useful tool to understand and discuss the context of C2 systems, what C2 systems in general has to produce, and what they consist of.

The development of sufficient military C2 systems is an urgent issue. Due to its complexity, it requires collaboration of people with various qualifications and skills, who comes from different disciplines and development traditions. The developed framework is intended to be a contribution to ease the collaboration of these people. Further, it is our hope that the framework will strengthen those who will work together across the borders of traditions.

Acknowledgements

The work presented in this paper was funded by the *C2 functional study* within the Swedish Armed Forces' study program.

References

Alberts, D. S. & Hayes, R. E. (2006). *Understanding Command and Control.* DoD Command and Control Research Program (CCRP). CCRP Publication Series.

Ashby, W.R. (1963). *Introduction to cybernetics*. John Wiley and Sons Inc.

Ashby, W.R. (1958). Requisite variety and its implications for the control of complex systems, *Cybernetica, 1*(2), 83-99. [http://pcp.vub.ac.be/Books/AshbyReqVar.pdf]

Barry, C. L. (2003). Transforming NATO Command and Control for Future Missions. *Defense Horizons*, (28), 1-11.

Beer, S. (1981). *Brain of the firm. (2nd ed)*. John Wiley & Sons Ltd.

Bhuiyan, N. & Baghel, A. (2005). An overview of continuous improvement: from the past to the present. *Management Decision, 43*(5), 761-771.

Blanchard, B.S. (2008). *System Engineering Management*. John Wiley & Sons.

Boehm, B.W. (1988). A spiral model of software development and enhancement. *Computer, 21*(5), 61-72.

Boyd, J. (1987). *A discourse on winning and losing*. Maxwell Air Force Base, AL: Air University Library Document No. M-U 43947 (Briefing slides)

Brehmer, B. (2005). The Dynamic OODA Loop: Amalgamating Boyd´s OODA Loop and the Cybernetic Approach to Command and Control. *Proceedings of the 10th International Command and Control Research and Technology Symposium,* McLean, VA.

Brehmer, B. (2007). Understanding the functions is the key to progress. *International C2 Journal, 1,* 211-232.

Brehmer, B. (2009). From function to form in the design of C2 systems. *Proceedings of the 14th International Command and Control Research and Technology Symposium,* Washington, DC.

Brehmer, B. (2010). *The evolution of C2. Proceedings of the 15 th International Command and Control Research and Technology Symposium,* Santa Monica, CA.

Brehmer, B. (2011). Harmony Rather than Unity. A Command Concept for Complex Endeavours. *Proceedings of the 16th International Command and Control Research & Technology Symposium (ICCRTS),* Quebec, Canada, 21-23 June 2011.

Brehmer, B. (2013). *Insatsledning.* Swedish Defence University.

Buede, D.M., & Miller, W.D. (2016). *The engineering design of systems: models and methods*. John Wiley & Sons.

Carroll, G. (2012) Command, control and intelligence to 2021: the global C2/C4ISR market. Army Technology. [https://www.army-technology.com/features/featuredssi-icd-research-c2c4isr-market/]

Checkland, P.B. & Scholes, J. (2001). Soft Systems Methodology in Action, in J. Rosenhead and J. Mingers (eds), Rational Analysis for a Problematic World Revisited. Wiley.

Davis, A. (2013). *Just enough requirements management: where software development meets marketing*. Addison-Wesley.

Davis, M.C., Challenger, R., Jayewardene, D.N., & Clegg, C.W. (2014). Advancing socio-technical systems thinking: A call for bravery. *Applied ergonomics, 45*(2), 171-180.

Edström. H. & Josefsson, A. (2016). *Vem kan leda operationer… och vem bör?* Kungl. Krigsvetenskapsakademiens Handlingar och Tidskrift, 1/2016, s. 67-86.

Emery, F. (1978). *The Emergence of a New Paradigm of Work*. Australian National University.

Enquist, H. & Makrygiannis, N. (1998). Understanding misunderstandings [in complex information systems development]. *Proceedings of the Thirty-First Hawaii International Conference on System Sciences* (pp. 83-92).

Eriksson, H.E. & Penker, M. (2000). Business modeling with UML. *New York*, 1-12.

Flood, R.L. & Carson, E.R. (1993). *Dealing with Complexity*. Plenum Press.

Granåsen, M., Hallberg, N., Josefsson, A., & Ekenstierna, C. (2017). *Två grundläggande koncept för ledning av framtida militära operationer,* (FOI-R--4467—SE). Swedish Defence Research Agency.

Hallberg, N., Andersson, R., & Ölvander, C. (2010). Agile architecture framework for model driven development of C2 systems. *Systems Engineering, 13*(2), 175-185.

Hallberg, N., Hallberg, J., Granlund, H., & Woltjer, R. (2014). Exploring the Rationale for Emergency Management Information Systems for Local Communities. *International Journal of Information Systems for Crisis Response and Management, 6*(2), 16-37.

Hallberg, N., Jungert, E., & Pilemalm, S. (2014). Ontology for systems development. *International journal of software engineering and knowledge engineering*, *24*(03), 329-345.

Hallberg, N., Pilemalm, S., & Timpka, T. (2012). Quality Driven Requirements Engineering for Development of Crisis Management Systems, *International Journal of Information Systems for Crisis Response and Management*, *4*(2), 35-52.

Hallberg, N., Timpka, T., & Eriksson, H. (1999). The medical software quality deployment method. *Methods of information in medicine*, *38*(1), 66-73.

ISO 9241-11, *Ergonomic requirements for office work with visual display terminals* (Geneva, International Organization for Standardization, 1998).

ISO/IEC 15288, *System Engineering System Life Cycle Processes* (International Organization for Standardization/International Electrotechnical Commission, Geneve, 2002).

ISO/IEC/IEEE (2010) 24765:2010, *Systems and software engineering — Vocabulary* (The Institute of Electrical and Electronics Engineers, New York, 2010).

Josefsson, A. & Lavman, M. (2017). *Svensk planerings-och ledningsmetod* (SPL 2017) [Swedish method for planning and C2]. Swedish Armed Forces.

Josefsson, A., Marklund, J., & Hansson, L.H. (2007). *Ledningskoncept för integrerad dynamisk ledning (IDC2)* [C2 concept for integrated dynamic command and control]. Swedish Armed Forces.

Josefsson, A., Marklund, J., & Hansson, L.H. (2008). *IDC2 - a new C2 concept within the framework of a Network Based Defence Concept*. Proceedings of the 13th International Command and Control Research and Technology Symposium, Seatle, WA.

Kaindl, H. & Svetinovic, D. (2010). On confusion between requirements and their representations. *Requirements Engineering, 15*(3), 307-311.

Kasser, J. (2013). *A framework for understanding systems engineering*. CreateSpace Independent Publishing Platform.

Kirby, R.S. (1990). *Engineering in history*. Courier Corporation.

Kossiakoff, A., Sweet, W.N., Seymour, S.J., & Biemer, S.M. (2011). *Systems engineering principles and practice*. John Wiley & Sons.

Larman, C. & Basili, V.R. (2003). Iterative and incremental developments. a brief history. *Computer, 36*(6), 47-56.

Lawson, H.W. (2010). *A journey through the systems landscape*. College Publications.

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.

Ministry of Defence (UK) (2015). *Strategic Trends Programme: Future Operating Environment 2035*.

Ministry of Defence (UK) (2017). *Future of Command and Control*. Joint Concept Note 2/17. The Development, Concepts and Doctrine Centre (DCDC).

Mumford, E. (2006). The story of socio-technical design: reflections on its successes, failures and potential, *Information Systems Journal*, *16*, 317–342.

Naikar, N., Hopcroft, R., & Moylan, A. (2005), *Work domain Analysis: Theoretical Concepts and Methodology*, DSTO-TR-1665.

NASA (2007). *Systems Engineering Handbook*, NASA/SP-2007-6105 Rev 1. https://www.nasa.gov/sites/default/files/atoms/files/nasa_systems_engineering_handbook.pdf.

NATO ACT (2016). *Command and Control Focus Area. Functional Analysis*.

Pilemalm, S. & Hallberg, N. (2008). Exploring service-oriented C2 support for emergency response for local communities. In *Proceedings of the 5th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2008)* (pp. 159-166).

Pilemalm, S., Lindell, P.O., Hallberg, N., & Eriksson, H. (2007) Integrating the Rational Unified Process and participatory design for development of socio-technical systems: a user participative approach. *Design Studies, 28*(3), 263-288.

Pogoson, A. I. (2018). *Issues, Trends and Challenges in an Emerging Global Power Structure. Canadian Social Science*, *14*(2), 5-15.

Rasmussen, J. (1984). The role of hierarchical knowledge representation in decision making and system management. *IEEE Transactions on Systems, Man and Cybernetics, SMC-15 2, 4-243*.

Rasmussen, J., Pejtersen, A.M., & Goodstein, L.P. (1994). *Cognitive Systems Engineering*. Wiley-Interscience.

Regeringen (2015). *Försvarspolitisk inriktning – Sveriges försvar 2016-2020* (prop. 2014/15:109)

Rosseau, R. & Breton, R. (2004). The M-OODA: A model incorporating control functions and teamwork in the OODA loop. *Proceedings of the Command and Control Research and Technology Symposium (ICCRTS)* (pp 14-16).

Royce, W.W. 1970. Managing the development of large software systems: Concepts and techniques. *Proceedings of IEEE WESTCON* (pp 1–9).

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.

Schekkerman, J. (2004). *How to survive in the jungle of enterprise architecture frameworks: Creating or choosing an enterprise architecture framework*. Trafford Publishing.

Schlager, K.J. (1956). Systems engineering-key to modern development. *IRE Transactions on Engineering Management*, (3), 64-66.

Seffah, A. & Metzker, E. (2004). The obstacles and myths of usability and software engineering. *Communications of the ACM*, *47*(12), 71-76.

Simon, H.A. (1996). *The sciences of the artificial (3rd ed)*. MIT Press.

Sokovic, M., Pavletic, D., & Pipan, K.K. (2010). Quality Improvement Methodologies – PDCA Cycle, RADAR Matrix, DMAIC and DFSS. *Journal of Achievements in Materials and Manufacturing Engineering*, *43*(1).

Sommerville, I. (2016). *Software engineering*. Pearson.

Spak, Ulrik. (2017). The common operational picture: A powerful enabler or a cause of severe misunderstanding. *Proceedings of the 22nd International Command and Control Research and Technology Symposium*, Los Angeles, CA.

Stoltenberg, J. (2016). Keynote speech by NATO secretary general Jens Stoltenberg at the Oxford Union [online].

Available from http://www.nato.int/cps/en/natohq/opinions_137882.htm [Accessed 16 December 2018].

Swedish Armed Forces (2016a). *Handbok Nomenklatur Ledning (H Nomen Led)*. FM2016-7616:1

Swedish Armed Forces (2016b). *Militärstrategisk doktrin för Sveriges militära försvar*. FM2016-7616:1.

Swedish Armed Forces (2017). *Huvudstudie Ledning, Delrapport 2016*. FM2017-162:1

Swedish Armed Forces (2018). *Slutredovisning av försvarsmaktens perspektivstudie 2016-2018, Tillväxt för ett starkare försvar.* FM2015-13192:15.

Tague, N.R. (2005). *The quality toolbox*. ASQ Quality Press.

van Creveld, M. (1985). *Command in war*. Harvard University Press.

von Bertalanffy, L. (1969). *General System Theory: Foundations, Development, Applications*. Braziller.

Waldenström, C. (2016). För att förstå vad man ska göra måste man se möjlighetsrummet. In Andersson, I. *Operativ ledning: Slutrapport 2013-2015*. Stockholm: Försvarshögskolan (FHS)

Walker, G.H., Stanton, N.A., & Jenkins, D.P. (2017). *Command and control: the sociotechnical perspective*. CRC Press.

Wirtz, J.J. (2017). *Life in the "Gray Zone": observations for contemporary strategists*. Defense & Security Analysis, 33(2), 106-114.

Yates, F.E. (1978). Complexity and the limits to knowledge. *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology, 235*(5), R201-R204.

Yin, R.K. (1994). *Case Study Research: Design and Methods* (Second edition). Thousand Oaks: Sage Publications.

Zachman, J. (1987). A Framework for Information Systems Architecture, *IBM Systems Journal, 26(3)*, IBM Publication G321-5298.