Situated cognitive engineering: Developing adaptive track handling support for naval command and control centers

Mark Neerincx
TNO

Guido Te Brake
TNO

Josine van de Ven
TNO

Henryk F. R. Arciszewski
TNO

Situated cognitive engineering: Developing adaptive track handling support for naval command and control centers

M.A. Neerincx[1,2], G.M. te Brake[1], J.G.M. van de Ven[1], H.F.R. Arciszewski[1], T.E. de Greef[1,2], and J. Lindenberg[1]

[1]TNO Human Factors, P.O. Box 23, 3769 ZG Soesterberg, The Netherlands
<mark.neerincx><guido.tebrake>@tno.nl

[2]Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

ABSTRACT. *Future naval missions set high operational, human factors, and technical demands for a system that enhances human-machine teams' capabilities to cope with complex and potentially hazardous situations. This paper presents a situated Cognitive Engineering (sCE) methodology for an integrated analysis of these three types of demands, the derivation and maintenance of a requirements baseline with its rationale, and the refinement and validation processes of the requirements (including human-in-the-loop evaluations of simulation-based prototypes). Application of this method provided a coherent and concise compilation of design knowledge for an Adaptive Automation (AA) module for track identification: a theoretical and empirical founded Requirements Baseline with its design rationale consisting of core functions, claims, scenarios and use cases. The core functions focused on coping with workload dynamics and maintaining situation awareness; for each core function we specified claims on the operational effects and requirements for the system, which were added to the use cases specifications. The evaluation showed that the specified AA-module helps to reduce workload and maintain adequate situation awareness during critical naval missions.*

1. Introduction

Naval missions are changing substantially, setting new demands for advanced cognitive support systems. There is a general increase in operation diversity, and in complexity and amount of information flows that have to be processed, while team sizes are decreasing. The smaller teams must have the capabilities to cope with ambiguous and dynamic situations under high time pressure (e.g., for surveillance of littoral regions). In such situations, the team members have to collaborate optimally within their team and with 'external' partners who may come from other defense units and countries. To cope with all these changing demands, the Royal Netherlands Navy (RNlN) aims at flexible crews and systems that adapt to dynamic demands. The research program 'Human-System-Task Integration' (HSTI) identified and developed a broad set of design knowledge—consisting of theories, concepts, models and methods—to support the (re)design of crews, work and systems.

This paper presents a situated cognitive engineering method that helps to systematically acquire, refine and validate such design knowledge for the derivation of a theoretical and empirical founded Requirements Baseline. This method has originally been developed for the space domain, and was recently used to establish a Requirements Baseline for a system that supports distributed human-machine teams during planetary space missions (Neerincx et al., 2006). The method seems transferable to a lot of other domains as well. The naval domain is a case in point, because of the correspondences between space and defense missions. Both take place in hostile and complex environments where human operations and safety are highly dependent on adequate technology performance. Our situated cognitive engineering method fits in the trend of simulation-based acquisition for the development of complex and critical systems (US Army, 2007). Especially in the military domain, use of simulation technology and modeling across integrated acquisition phases and programs is considered important to save costs and develop products that meet users' expectations.

2. Situated Cognitive Engineering

Technological developments are causing a fundamental change of the machine's role in complex work environments, e.g. in the defense and space domain. Machines are becoming part of cognitive systems that consist of human and synthetic actors who collaborate for successful attainment of their joint operation objectives (e.g., Hoc, 2001). Neerincx & Lindenberg (2008) developed a situated Cognitive Engineering (sCE) method for the design of such Human-Machine Collaboration (HMC). It follows an iterative human-centered development process corresponding to recent human-factors engineering methods and standards (e.g. ISO 13407 "Human-centered design processes for interactive systems"), aiming at an incremental development of advanced technology. Corresponding to the 'classical' CE methods (Hollnagel and Woods, 1983; Norman, 1986; Rasmussen, 1986), it consists of an iterative process of generation, evaluation and refinement. In addition, the sCE method combines operational, human factors and technological analyses to establish a sound and practical design rationale, and applies simulation-based acquisition and assessment techniques for refinement and validation. Figure 1 shows the general structure that consists of three components: the Work Domain and Support (WDS) analysis, the generation and maintenance of the Requirements Baseline (RB) with its rationale, and the review and refinement activities to improve and validate this Requirements Baseline. In general, the cognitive engineering activities start with a—possibly preliminary—WDS analysis, followed by a first RB specification and, subsequently, a series of refinement processes. It should be noted that during this process, new insights in the operational demands, human factors or technology can be acquired and used to further improve the RB. Furthermore, the methodology allows for an incremental development approach, in which successive prototypes include more functions.

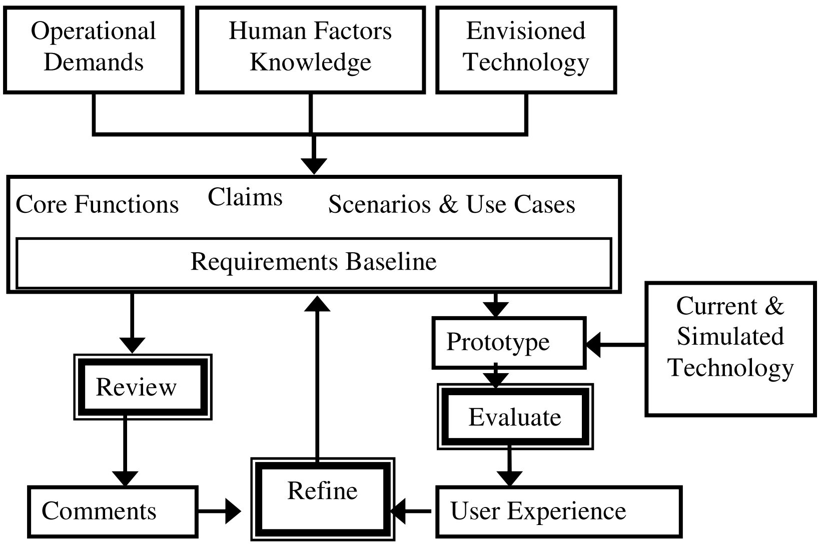

Figure 1: *The iterative process of requirements analysis.*

First, the WDS analyses identify operational, human factors and technological demands that set specific objectives or constraints for the support of future naval missions. Current and future *operational demands* can be acquired via scenario analysis (Rosson & Carroll, 2001; McCrickard & Chewar, 2006) and cognitive work analysis (Vicente, 1999; Naikar and Sanderson, 2001). With respect to *human factors knowledge*, the methodology advocates to systematically address state-of-the-art practical theories and models, which include accepted features of human cognitive and affective processes and which can be supported by technological artifacts (cf. Neerincx, 2003). This paper will provide an example of such a theory on situation awareness (Endsley, 2000). The analysis of *envisioned technology* is included to set a feasible design space, and to systematically address the adaptive nature of both human and synthetic actors with their reciprocal dependencies (Maanen et al., 2005).

The second component of the sCE methodology consists of the construction and maintenance of the Requirements Baseline (a table with all requirements) and the general design rationale that consists of the core functions, claims, and scenarios & use cases. For the specification of this rationale and the actual requirements baseline, we distinguish three steps that should be followed both from top and bottom (i.e., a top-down, goal-directed approach to work out core functions and a bottom-up, event-driven approach to address contextual or situational demands in scenarios (Neerincx, 2003):

- The *core functions* of the system are derived from the WDS analysis. For example, to better cope with workload dynamics during track identification in the Command and Information Center, an adaptive track-handling support system can be proposed which brings about an adequate operator's situation awareness. Corresponding core functions can be the prevention of mode errors or automation surprises.
- For each core function, one or more testable *claims* on its operational effects have to be specified; such a claim can be assessed unambiguously in review or evaluation processes. Both positive and negative claims can be specified. An example of the proposed system is that the operators will show faster adequate responses to all critical tracks in high workload conditions. Furthermore, for each core function, one or more requirements have to be specified for the future system (i.e., what the system must do). The complete set of requirements comprises the *requirements baseline*.
- *Scenarios & use cases*. Scenarios are coherent and situated stories about how a specific team of operators and the system behave or will behave in specific circumstances with the operational consequences, e.g. for track identification in littoral waters. Scenarios are often visualized via annotated storyboards to present and discuss the design rationale, and used for the evaluation of prototypes. Use cases describe the general behavioral requirements for software systems or business processes, and have a specific specification format. According to our methodology, each use case should explicitly refer both to one or more requirements and to one or more claims. In addition, each claim and each requirement should be included in one or more use cases. A scenario can be viewed as an instance of one or more use cases. Use cases and scenarios are very useful when discussing a not-yet-existing system with different stakeholders. With minor help most people are able to understand these design specifications.

Third, the sCE methodology distinguishes two types of refinement and validation processes. Domain, human factors and technical experts have to *review* the requirements baseline and its rationale, checking for consistency and validity. Furthermore, simulation-based prototypes can be built for human-in-the-loop *evaluations* of the claims. Both the environmental conditions and the system functions can be simulated in such evaluations, partly controlled by a human operator (e.g., to trigger alarms and change the level of automation according to a fixed set of rules). Review and evaluation results should be used to confirm, refine or remove design specifications.

The HSTI program applied the sCE methodology to the development of an Adaptive Automation (AA) module for track identification in order to establish a theoretically sound and empirically proven Requirements Baseline, and to realize a good transfer of research activities to the future system development and software engineering activities. It is important to note that this study focused on a future naval system that is in an early development stage, in which human factors aspects, operational demands and technology are systematically explored. According to current system development approaches, requirements are assessed and refined from rather high-level specifications in early development stages to detailed definitions in late development stages. Consequently, the present requirements baseline contains relatively abstract specifications that will be assessed and refined further.

3. Work Domain and Support Analysis

3.1. *Operational demands*

Missions for navies around the world are becoming increasingly complex, more-and-more taking place in littoral waters with involvement of several parties that may have unclear intentions and loyalties (see Joint Vision 2010 and 2020). Furthermore, reductions in crew sizes may cause reduced resilience to high workload situations (e.g., due to diminished opportunities for task re-planning). To deal with these trends, the amount of automated systems on board of naval ships is increasing, especially for command and control tasks, the development of situation awareness (SA), and weapon deployment. In future naval command and control, humans and systems will need to collaborate in an effective manner to obtain optimal performance. Problematic of automation in these environments is the large variability in workload. For prolonged periods, the amount of work may be rather small, but there will also be intense periods with high workload and stress. Overload might appear due to a competition for the operator's attention that is going on between different information items. If many tasks are handled by automated systems, the operator can deal with high workload circumstances, but will suffer from severe underload during quiet periods, probably losing his or her situational awareness.

3.2. Human Factors knowledge

In addition to the operations analyses, a literature study was conducted on Human Factors of complex high-demanding task environments, in which well-trained human operators may act in extreme and hostile situations (such as the defense and safety domains). This study provided key issues that the system should address to cope with the workload dynamics:

- *Cognitive task load* (Neerincx, 2003). Due to the large variety over time in number and complexity of tasks, the momentary mental load of the operators can be suboptimal. The system should support an adequate load scheduling over time and available human-machine resources based on a model of cognitive task load that distinguishes three load factors: percentage time occupied, number of task switches and task complexity.
- *Situation awareness* is most relevant in naval command and control environments (Endsley, 2000). A large part of sensors, automation, and operators, are used to build a common operational picture showing all relevant tracks, and providing the officers with the information required to make sense of the situation (Weick, 1995).
- *Trust* (Parasuraman, 1997; Adams et al., 2003). If users rely too much or too little on human or technology, performance will be suboptimal. Appropriate

trust depends on understanding of capabilities of the system, colleagues, and oneself. Users are not very good at estimating how much to trust a machine.

- *Diversity of cognitive capacities* (Scerbo, 2001). The human operators have different expertise and experiences, and will perform their various tasks in different environments. This causes differences in the momentary capabilities, levels of attention and available modalities. The system needs to be aware of these factors to be able to help the operator effectively by tailoring the level of automation to the available attention resources and modalities.
- *Decision making* (Klein, 1998). Rational decision making ought to be supported by exhaustive evaluation of options, collecting and providing an overview, ranking the options, and possibly proposing the best. Naturalistic decision-making is supported by functions that assess the situation based on patterns capturing experience and preference of the crew, recommend actions based on the patterns, check that the execution of the course of actions is according to expectancies, and test assumption underlying human naturalistic decision-making.
- *Collaboration* (Mohammed and Dumville, 2001). Adaptive teams help to avoid gaps and overlap in individuals' assigned work (i.e. support coordination), to obtain mutual benefits of human and machine actors by sharing or partitioning work (i.e. support cooperation) and to achieve collective results that the participants would be incapable of working alone. Furthermore, it helps to support the generation and maintenance of a shared mental model within human-machine teams, which contains both team knowledge as well as situation knowledge. By mediating between actors, insight will be provided into the other actors' goals, intentions behavior and needs.

3.3 Envisioned technology

In addition to the operations and human-factors analysis, we conducted a technology assessment. On naval ships, the combat management system (CMS) assists the crew in its assessment of the tactical situation, its decision making with respect to the evolving situation, and the execution of its subsequent actions. Four types of technology development will influence future CMSs.

- Current cross-disciplinary Research and Development (R&D) communities bring forth enabling technology that can be integrated, and possibly embedded, in such environments in order to sense, interpret and anticipate individual human conditions and behaviors (e.g. to improve safety and health) around themes like affective computing and augmented cognition (e.g., Picard, 1997; Satyanarayanan, 2001; Schmorrow et al., 2005).

- Context-sensing systems are being developed that employ diverse smart sensors to assess the momentary context of operation.
- Developments in Artificial Intelligence, such as Multi-Agent Systems, bring forth real or virtual machines that can act autonomously in dynamic environments and take the initiative in joint human-machine operations (e.g. Clancey, 2004). The demand for unmanned vehicles (aerial, land, above, under water) is rising sharply. It's a response to the increasing danger from highly mobile threats, and the need to rapidly engage high-value targets of opportunity, especially deep in unfriendly territory and dangerous environments.
- Future naval operations will be network-centric. Sensors and shooters are connected in a big network, enabling ships and planes to use equipment on other platforms. This capability will come with a sharp increase of the amount of available information and planning complexity.

Future combat management systems and sensors will contain many intelligent algorithms for correlating, fusing, and interpreting data. Many low-level tasks related to correlation and sensor fusion will be handled automatically, and more and more automation will enter the realm of more cognitive tasks. This trend will produce domain algorithms that can handle tasks such as classification, identification, navigation, planning, et cetera. The systems will be able to deal with incompleteness and uncertainty in information. A major challenge is to design for failure: realize resilience for both machine and human failures (Grant et al., 2007).

Adaptive automation has been proposed for dynamic task environments such as naval command and control. Adaptive systems adjust their level of support to the circumstances: low when possible, high when required, with the goal of keeping the operator near optimal workload levels in all circumstances (Scerbo, 1996, Rouse, 1988). If an operator is starting to get overwhelmed by the situation, a CMS that is capable of autonomous decision making could intervene and reallocate less critical parts of the work to itself so that the workload of the operator is reduced and he or she is again up to the task. When the situational demands are reducing, automation levels are lowered, shifting authority and responsibility back to the operator, hence improving operator's awareness and skills.

4. Design rationale: scenarios, core functions, claims and use cases

Development of systems that use innovative human-machine concepts such as AA, in which there is a critical balance between situated and mutually-influencing benefits (such as workload reduction) and costs (such as reduced situation awareness), warrants the use of the sCE method. We have applied this method to develop and test AA for a task that is conducted on many naval ships, track identification.

4.1 Scenarios

For future naval missions, several high-level scenarios have been developed over the years. For the specific support system of this paper, four scenarios were developed in cooperation with training experts of the Royal Netherlands Navy. All scenarios were intended to bring a reasonable workload on the operator and included various threats or suspicious-looking tracks that contributed to the workload. Two scenarios were developed around more or less traditional air and surface warfare in a high-tension peace-enforcing situation while the other two scenarios were situated against a civilian smuggling background.

4.2 Core functions

When sensors detect an object in the vicinity of the ship, for example another ship or aircraft, a track is built as a representation of the object within the CMS. Tracks that are initially unknown must be classified (is it an airliner, a fighter, or a helicopter?) and subsequently identified (is it a neutral platform, one of our own forces, or possibly a hostile one?) . Track identification can to a large extent be handled automatically by our prototype CMS. When the number of unknown tracks becomes too high, AA increases its support level and handles the ‘easy’ tracks (i.e., neutral airliners and commercial vessels). The complex tracks (for example, tracks with strange behavior or suspicious characteristics) are still handled by the human operator. If things get really busy, most or all tracks are identified by the system, warning the user only of the very important, complex, or hostile tracks.

The main advantage of AA is improved workload levels for the human operator, potentially increasing overall human-machine performance. Possible drawbacks of AA are lowered SA, increasingly complex systems, mode errors and automation surprises. The perils related to SA have been described by Endsley, who called them demons of SA (Endsley 2003). To tackle these demons, we have used them explicitly in the development of our design rationale. For each demon, we identified a corresponding core function, such as ‘Prevent out-of-the-loop problems’. We completed these SA-based functions with other core functions, such as the increase of performance and the prevention of some cognitive biases, making a total of 18 core functions (see Table 1).

Table 1: *Core functions for our system.*

Prevent out-of-the-loop problems	Maintain situational awareness
Prevent skill degradation	Ensure Transparency
Prevent mode errors or automation surprises	Prevent undesirable system behavior
Prevent misplaced salience	Prevent complexity creep
Prevent/limit increases of system demands	Prevent errant mental model of the situation
Prevent unwanted modes of operation	Prevent tunnel vision
Extend human memory capacity	Accomodate workload, stress & fatigue
Prevent data overload	Prevent cognitive lockup
Prevent change blindness	Improve performance

4.3 Claims

For the core functions of Table 1, AA features were derived with corresponding claims on the expected effect to the performance of the tasks in the scenarios. These claims are formulated in terms of operational consequences, which can be empirically evaluated with end-users (e.g., via standard methods for measuring human performance, effort and learning). **Table 2** shows an example for a core function, to which two claims are associated.

Table 2: *An example of a core functions with two associated claims.*

FUNCTION 1. Prevent out-of-the-loop problems
CLAIM 1. *Feature*: When the task load decreases (e.g. few tracks to handle), a lower level of automation is triggered. *Result*: The user does (almost) everything and handles more tracks, so that (s)he is sufficiently engaged in the current operation (e.g., adequate eye movements and medium arousal level), detects relevant objects in time (e.g., adequate identification performance) and is not involved in unrelated and irrelevant activities (e.g., mainly task-related behavior). CLAIM 2. *Feature*: When task load is high (e.g. more pending warnings, and higher volume and complexity of tracks), the system takes over the least important tracks and leaves the user the more important and potentially more threatening ones. *Result*: The user handles the critical tracks adequately and in time (e.g., identification performance), so that (s)he is sufficiently engaged in the current operation (e.g., adequate eye movements and medium arousal level), detects relevant objects in time (e.g., adequate identification performance) and keeps adequate knowledge of the most critical situational changes (e.g., adequate situation reports).

4.4. Requirements Baseline

Based on the core functions and corresponding features of the envisioned system, we specified the requirements (i.e. what the system is supposed to do), taking notice of the following characteristics of a good requirement specification (IEEE, 1993):

- **Unambiguity** Every requirement should only have one interpretation. It is best if each characteristic of the final system will be described using a single unique term.
- **Completeness** Inclusion of all significant requirements for the main function of the system, including the user interface.
- **Verifiability** A requirement should be verifiable; i.e. one should be able to check that the system meets the requirements.
- **Consistency** A requirement specification is consistent if no set of individual requirements described in it conflict.

- **Modifiability** A requirement specification should be modifiable so that changes to the requirements can be made easily, completely, and consistently without introducing inconsistency.
- **Traceability** A requirement specification should be traceable so that the origin of each of its requirements is clear and it should facilitate the referencing of each requirement in future development or enhanced documentation.

Each requirement can be annotated with an indication on the importance of that requirement to implement the functionality of the use case. The MoSCoW list is often used to indicate the importance. MoSCoW stands for **M**UST have this, **S**HOULD have this if at all possible, **C**OULD have this if it does not affect anything else, and **W**ON'T have this time but WOULD like in the future.

Table 3. *Explanation of the use case fields (based on Cockburn, 2001)*

[UC_Nr] [UC_name]	*Number used to link requirements to use case.*	*Example:* UseCase 3 Increasing SA after decreasing LOA
Goal	*What is achieved by carrying out the use case.*	Limit out-of-the-loop problems
Actor	*Main human (or possibly machine) actors.*	Team member of Command and Control Centre
Precondition	*Contains the state of the system or user just before using the functionality.*	AA is at the medium or high level; User has a limited view of tracks as some are handled by the system, limiting his situational awareness to ‘dangerous’ tracks.
Post condition	*What is achieved using the functionality, describes the state of the system or user.*	AA is set at a lower level More tracks will be handled by the user from now on, increasing his or her overall situational awareness.
Trigger	*Defines the event (e.g., time, alarm, …) when a user needs the functionality or how the system knows that the function needs to be carried out.*	Amount of work (pending tracks, tracks requiring user attention) is below a preset threshold level.
Main Success Scenario	*A top-to-bottom description of an easy to understand and fairly typical scenario in which the actor’s goal is delivered.*	1. After decrease of automation level, more tracks of multiple categories will be handled by the user 2. In doing so, the user quickly gets good situational awareness.
Alternative Scenario	*Other ways to succeed, and the handling of the most important failures*	……
Satisfies claim	*List of claim-numbers that link to this use case*	Claim 1, Claim 25
Satisfies requirement	*List of requirement-numbers that link to this use case*	Requirement 13

4.5 Use cases

Based on the scenarios, core functions, claims and requirements of the previous subsections, we specified a number of use cases according to a fixed format. Table 3 shows the structure that was used, extending the description of Cockburn (2001). It should be noted that the specification of the requirements with its rationale (i.e., scenarios, functions, claims and use cases) is an iterative process. During the process, specifications of all elements are refined, and harmonized (e.g., use case refinements may lead to requirement refinements).

To facilitate the transfer of the AA design specification from our research team to the system development group of the RNlN, the specifictions were worked out into an information, a dynamic and a functional model, including state transition and class diagrams (Object-Oriented Analysis of Shlaer and Mellor, 1992). Parts of the functional model were described in pseudo code.

5. Refinement processes

5.1 Review

The established core functions, claims and use cases, were reviewed by domain experts (both operational and technical) from the Royal Netherlands Navy. Based on their comments, requirements were added or updated. However, considering the system is a research and development platform and not a realistic CMS, a thorough review by domain experts was not appropriate at this time. For real space or military systems, thorough reviewing by domain experts forms a major component in the refinement of the design rationale and requirements baseline, due to the criticality of the software and the problems of proper testing in realistic situations.

5.2 Evaluation

A CMS prototype was built to evaluate the adaptive identification design (De Greef et al., 2006; 2007). This prototype proved to be sufficiently realistic to conduct a series of experiments to evaluate the claims and refine the requirements baseline. Besides identification, the prototype consisted of additional functionality, such as classification of tracks, weapon deployment, and navigation, to provide the complex environment required for proper testing of the adaptive identification module. These other modules have not been designed according to the sCE method, but although some interaction exists between these modules and the claims related to the identification task, we feel this did not effect our evaluation.

5.2.1 Participants

The participants were four warfare officers (WO) and four warfare officer assistants (AWO) of the Royal Netherlands Navy with several years of operational experience using naval combat systems on naval ships. At the time of the trials the subjects were either involved in the design and development of new Combat Management Systems for the RNLN or engaged in the training of officers at the operational school of the RNLN. The participants were available for two days and participated in teams of two, one WO and one AWO.

Figure 2: *Naval domain experts participating in the evaluation of the prototype.*

5.2.2 Tasks & Apparatus

The participants worked with a workstation called the Basic-T attached to a simulated combat management system (see Figure 2). The simulated ship resembles a modern RNlN air defense and command frigate in its sensor and weapon suite; the CMS was newly designed and built to encompass new functionality and adaptivity.

The four scenarios of section 4.1, bringing about relatively high workload situations, were used for the evaluation. The participants were given mission goals and were instructed to defend the frigate against any threats. In all cases the primary mission goal was to build a recognized maritime picture (RMP) of the surroundings of the ship and to detect and identify all platforms present. Building the RMP amounted to monitoring the operational space around the ship, classifying and identifying contacts. Decision making involved moving the ship and its associated helicopter and possibly neutralizing hostile entities. As the sensor reach of a modern naval ship extends to a large region around the ship, this represented a full-time job. In addition, the participants were responsible for the short-term navigation of the ship, steering it toward whatever course was appropriate under the circumstances and avoiding collisions with other ships.

5.2.3 Experimental design

The automation mode was either adaptive or fixed. System support in the fixed mode was comparable to the lowest level in the adaptive mode. It is important to remember that in this way even in the fixed mode the system was still giving advice and drawing attention to tracks that merited such attention and as such offered more assistance than current combat management systems would do. The major difference was the fact that in the adaptive mode the automation would scale up and the system would acquire more responsibility in handling tracks autonomously when the user's workload increased. It would scale back again as soon as the workload decreased.

Seven dependent variables were measured to test the claims:

- The *subjective workload* ratings as rated during each scenario on a one dimensional rating scale from one to five, one meaning heavy underload and boredom, three a comfortable and sustainable workload and five an overload of the operator (Zijlstra, 1993).
- The expert was closely observing the participant and estimated the *workload*, the *situation awareness*, and the *quality* and *timeliness* of the actions of the participants every two minutes using the same one-dimensional rating scale. In these cases, a one indicated a bad performance, a five an excellent performance and a three a sufficient performance.
- The *performance* in terms of tracks handled and reaction time on signals of the machine was measured.
- Lastly the *communication* served as the seventh dependent variable.

5.2.4 User experience

The experiments have shown an increase in overall human–system performance. Tracks were handled quicker, and on average much less pending tracks remained. Situation awareness of the participants in the adaptive and fixed condition was similar, as were the observed workload levels. The latter was somewhat surprising, as a decrease was expected. However, the time that became available because the automation took over work was spent on extra communication, and presumably also on other more reflective tasks related to sense making. Opinions were divided on the matter of transparency and automation surprises. Some participants said the system seemed to read his mind, because the automation was set higher at exactly the right moment. Others, however, said that automation seemed random at times.

5.2.5 Refinement

The evaluation was the first time domain experts used the system. Apart from implementation features and bugs, several problems and omissions within the prototype and the underlying claims and uses cases became evident. Two examples will be presented.

During the evaluation some observations were made with respect to system modes and automation surprises. Sometimes user decisions were overruled by the system even though the user had already inspected the system view and had discarded its conclusion in favor of his own. The sequence of system action and user action seems to be the deciding issue: if a user action occurs before system action, the user's decision should be safeguarded from unwarranted changes by the system. Apparently, the claims related to system modes were not appropriately met, and a refinement of the requirements baseline with its rational was required.

A second observation was the wish of users to be able to inspect tracks that were handled by the system autonomously during periods of high workload as soon as things were calming down and automation levels were lowered. A claim was added related to the out-of-the-loop function (see table 4) and a new corresponding use case was constructed.

***Table 4**: New claim for the core function related to out-of-the-loop prevention.*

CORE FUNCTION 1. Prevent out-of-the-loop problems
CLAIM 3. *Feature*: When the automation level is lowered, the user is made aware of tracks that have been handled by the system. *Result*: Tracks that were handled by the system at high automation levels are labeled as such, so that the operator can inspect them after the level of automation has lowered (e.g., user behavior) to improve his or her momentary knowledge of the situation (e.g., adequate situation reports).

6. Conclusions & Discussion

Future naval missions set high operational, human factors, and technical demands for a system that enhances human-machine teams' capabilities to cope with complex and potentially hazardous situations. This paper presented a situated Cognitive Engineering (sCE) methodology for an integrated analyses of these three types of demands, the derivation and maintenance of a requirements baseline with its rationale, and the refinement and validation processes of the requirements (including human-in-the-loop evaluations of simulation-based prototypes). It follows an iterative human-centered development process corresponding to recent human-factors engineering methods and standards (e.g. ISO 13407 "Human-centered design processes for interactive systems"), aiming at an incremental development of advanced technology.

Application of this method provided a coherent and concise compilation of design knowledge for an Adaptive Automation (AA) module for track identification on naval ships: a theoretical and empirical founded requirements baseline with its design rationale consisting of core functions, claims, scenarios and use cases. The core functions focused on coping with workload dynamics and maintaining situation awareness; for each core function we specified claims on the operational effects and requirements for the system, which were added to the use case specifications. The evaluation showed that the specified AA-module helps to reduce workload and maintain adequate situation awareness during naval missions. The current requirements baseline can be used *both* to start implementation of an adaptive track-handling support module *and* to extend the AA support to other naval tasks.

The sCE methodology was recently developed and applied for the design of systems for manned space missions (Neerincx et al., 2006). Also in the space domain, it provided a practical, coherent and extendable requirements baseline for crew support that can be incrementally developed and implemented. It is interesting to note that—for a part—similar human factors theories and models were applied in the work domain and support analysis. By explicating the design rationale in a similar way, it is rather easy to identify which support elements can be applied to the different domains and which elements are really domain specific.

References

Adams, B.D., Bruyn, L.E., Houde, S. and Angelopoulos, P. (2003). *Trust in Automated Systems: Literature Review*. DRDC Report No. CR-2003-096. Toronto, Canada: Defence Research and Development.

Clancey, W.J. (2004). Roles for Agent Assistants in Field Science: Understanding Personal Projects and Collaboration. *IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications and Reviews*, Vol. 34, No. 2, May 2004.

Cockburn, A. (2001). *Writing Effective Use Cases*. Addison-Wesley, ISBN 0201702258.

Endsley, M.R.. (2000). Theoretical Underpinnings of Situation Awareness. In M.R. Endsley & D.J. Garland (eds). Situation Awareness Analysis and Measurement. LEA, Mahwah, NJ, USA.

Endsley, M. R., Bolte, B., & Jones, D. G. (2003) Designing for Situation Awareness: An approach to human-centered design. London: Taylor & Francis.

Grant,T., van Fenema, P., van Veen, M. & Neerincx, M.A. (2007). On regarding 21st century C2 systems and their users as fallible ePartners. *12th International Command and Control Research and Technology Symposium*. Newport, Rhode Island, June 19-21, 2007.

De Greef, T.E., Arciszewski, H.F.R.; Lindenberg, J., Van Delft, J.H. (2007). Adaptive Automation Evaluated; Report Number TBO-DV 2007 A610; TNO Defence Security and Safety, Soesterberg, the Netherlands.

De Greef, T.E., Arciszewski, H.F.R., Van Delft, J.H. (2006). Adaptive Automation using an object-orientated task model. Proceedings MAST conference, 4-6 september 2006 Acropolis Convenion & Exhibition Centre, Nice, France.

Hoc, J.-M. (2001). Towards a cognitive approach to human-machine cooperation in dynamic situations. *International Journal of Human–Computer Studies*, 54(4), 509–540.

Hollnagel, E., and Woods, D. D. (1983). Cognitive systems engineering: New wine in new bottles. *International Journal of Man–Machine Studies*, 18, 583–600.

IEEE (1993). *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Std 830-1993.

Klein, G.. (1998). *Sources of power: How people make decisions*. Cambridge, MA: MIT Press,

Schneider-Hufschmidt, M., Kühme, T., & Malinowski, U. (1993). Adaptive user interfaces : principles and practices (*Human factors in information technology*, 10 ed.). Amsterdam : North-Holland.

Maanen, P.P. van, Lindenberg, J., and Neerincx, M.A. (2005). Integrating human factors and artificial intelligence in the development of human-machine cooperation. In H.R. Arabnia, and R. Joshua (Eds), *Proceedings of the 2005 International Conference on Artificial Intelligence (ICAI'05)* (pp.10-16). Las Vegas, NV: CSREA Press.

McCrickard, D.S. and Chewar, C.M. (2006). Designing Attention-Centric Notification Systems: Five HCI Challenges. In: J. Chris Forsythe, Michael L. Bernard, and Timothy E. Goldsmith (Eds.). *Cognitive Systems: Human Cognitive Models in Systems Design*, pp. 67-89. Lawrence Earlbaum.

Mohammed, S. & Dumville, B.C.. (2001). Team mental models in a team knowledge framework: Expanding theory and measurement across disciplinary boundaries. *Journal of Organizational Behaviour*, 22(2), pp. 89-106.

Naikar, N & Sanderson, P.M. (2001). Evaluating Design Proposals for Complex Systems with Work Domain Analysis, *Human Factors*, 43(4), 529-542.

Neerincx, M.A.. (2003) Cognitive task load design: model, methods and examples. In: E. Hollnagel (ed.), *Handbook of Cognitive Task Design*. Chapter 13. Mahwah, NJ: Lawrence Erlbaum Associates, pp. 283-305.

Neerincx, M.A. & Lindenberg, J. (forthcoming). Situated cognitive engineering for complex task environments. In: Schraagen, J.M.C., Militello, L., Ormerod, T., & Lipshitz, R. (Eds). *Natural Decision Making & Macrocognition*. Aldershot, UK: Ashgate Publishing Limited.

Neerincx, M.A. (2007). Modelling Cognitive and Affective Load for the Design of Human-Machine Collaboration. In: D. Harris (Ed.). *Engineering Psychology and Cognitive Ergonomics, HCII 2007*, LNAI 4562, pp. 568–574. Berlin Heidelberg: Springer-Verlag. ISBN 978-3-540-73330-0

Neerincx, M.A., Lindenberg, J., Smets, N., Grant, T., Bos, A., Olmedo Soler, A,. Brauer, U., Wolff, M. (2006). Cognitive Engineering for Long Duration Missions: Human-Machine Collaboration on the Moon and Mars. *SMC-IT 2006: 2nd IEEE International Conference on Space Mission Challenges for Information Technology,* pp. 40-46. Los Alamitos, California: IEEE Conference Publising Services.

Norman, D. A. (1986). Cognitive engineering. In D. A. Norman, and S. W. Draper (Eds) *User-Centered System Design: New perspectives on human–computer interaction*. (pp.31-62). Hillsdale, NJ: Lawrence Erlbaum Associates.

Parasuraman, R., & Riley, V. (1997). Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39, 230-253.

Picard, R.W. (1997). *Affective computing*. MIT Press, Cambridge, MA.

Rasmussen, J. (1986). *Information processing and human–machine interaction: an approach to cognitive engineering.* Amsterdam, Elsevier.

Rosson, M. B. and Carroll, J. M. (2001). *Usability engineering: Scenario-based development of human-computer interaction*. San Francisco, CA: Morgan Kaufman.

Rouse, W.B. (1988). Adaptive aiding for human/computer control. *Human Factors* 30, pp. 431–443.

Satyanarayanan, M. (2001) Pervasive computing: vision and challenges. *IEEE Pers Commun* 8(4):10–17.

Scerbo, M.W. (1996). Theoretical perspectives on adaptive automation. In: Parasuraman, R. and Mouloua, M., Editors, 1996. *Automation and human*

performance: Theory and applications, Lawrence Erlbaum Assoc, Hillsdale, NJ, pp. 37–63.
Scerbo, M.W. (2001). Stress, workload and boredom in vigilance: a problem and an answer. In Hancock, P.A. & Desmond, P.A. (ed.), *Stress, Workload and Fatigue*. Mahwah, New Jersey, Lawrence Erlbaum Associates.
Schmorrow, D., Stanney, K.M., Wilson, G. and Young, P. (2005). Augmented cognition in human-system interaction. In: G. Salvendy (Ed.), *Handbook of human factors and ergonomics* (3rd ed.). New York: John Wiley.
Shlaer, S. and Mellor, S.J. (1992). *Object Lifecycles: Modeling the World in States*. New Jersey: Prentice Hall.
US ARMY PEOSTRI (2007). http://www.peostri.army.mil/PRODUCTS/SBA/ visited December 29, 2007.
Vicente, KJ (1999). *Cognitive Work Analysis: Toward Safe, Productive, and Healthy Computer Based Work*, Lawrence Erlbaum Associates, London.
Weick, K. (1995). *Sense-making in Organisations*. Sage, Thousand Oaks, CA, USA.
Zijlstra, F.R.H. (1993). Efficiency in work behavior. A design approach for modern tools. (PhD thesis, Delft University of Technology). Delft, The Netherlands: Delft University Press.