Mediating thoughts and streams of actions.

Complex intuitive interactions in a skilled worker environment.

Sturla Bakke

Thesis submitted for the degree of Philosophiae Doctor

Department of Informatics
Faculty of Mathematics and Natural Sciences
University of Oslo

July 2018

Acknowledgements

First of all, I thank my supervisor, Professor Tone Bratteteig who, without hesitation, accepted to guide me through this rather long and non-linear explorative process, and back on track when it went astray into the socio-technical wilderness. In addition to the regular way of doing a PhD, this one is the result of our long conversation, spanning quite some years. Her patience is nothing less than baffling.

I thank my co-supervisor, Jo Herstad, not only for expanding my views of what UIs can be, and the rather diverse character of mediated interaction that might follow, but also his philosophical approach to the somewhat contradictory relationship between humans and technology altogether, which he generously shared with me.

Without their careful and patient mentoring and guidance this dissertation would probably not have seen the light of day.

I must thank the design group at the Department of Informatics, University of Oslo, for taking me in as one of their own, into the warm and professional shelter. It has been a privilege.

I thank my employer, the former Norwegian School of Information Technology (NITH), now a part of Kristiania University College, for giving me the opportunity to do this research, and for providing the financial support, not to mention the sufficient, and dynamic, amount of time. Their patience, too, is rather astonishing.

I am indebted to Odfjell Chemical Tankers in Bergen, who generously allowed me to study the ORCA-project over such a long period of time, and especially Leif Gunnar Alvær and Olav Skintveit who generously and patiently gave me of their time during the fieldwork sessions.

I thank my research colleagues at the Department of Technology at Kristiania University College, for valuable input at defining moments during this process. A special thank you to Lars Sydnes for valuable assistance with the survey data.

Also, I would take this opportunity to thank the many reviewers and discussants in the community for their time, and comments on papers and paper drafts that, in various and edited ways, have diffused into this dissertation.

I am very grateful to Victor Kaptelinin for his valuable comments and suggestions about activity theory and intuitive interaction. The remaining ambiguities, and unintentional omissions of any sort are purely my own doing.

I spent many years as a graphic designer before I chose to pursue an academic path. This transition might, for many like-minded, be perceived as somewhat ambiguous. Having one foot among the practitioners one is not really academic, and with the other foot in academia, one try theorise about the not so theoretical «praxis».

I thank my family; my wife, Hege, for her never-ending support and love, patience and fruitful discussions, while keeping the base camp in order. My son, Sondre, who, on a regular basis, dragged me out the bubble of insanity, as finishing a PhD-dissertation can be - and luring me into his imaginary world of play and fantasy, and the occasional game of football, which I usually lost. I cannot help thinking that he sometimes really did it for my sake, rather than his own, sensing dad's need to come out and play. Smart little fellow! I still beat him in an occasional game of chess, though.

To my mother, Marit Bakke, who proudly knew this day would come, but sadly passed away before she could experience it herself. I see myself as a rational person. I do believe in science, but I sometimes catch myself thinking that she nods acknowledging from her spot in the realms of eternity.

Thank you.

Oslo, July 22, 2018

Sturla Bakke

Abstract

This thesis explores contextually various aspects of *intuitive use of user interfaces*. By viewing use-activity as an object in itself, and as such something one might relate to as familiar, it attempts to situate the understanding of *intuitive use* within an activity theoretical framework by leaning on a task and goal based approach to the human-computer [inter]activity.

The research in this thesis is undertaken in a, controlled, skilled worker environment, and discusses two aspects of the case study; (1) the comingling of developers and users during the development process of this narrow, niche software where the main use-activity, following the established practice among the users, is centred around sketching preliminary test solutions, and (2) human use of the software when it was released.

Based on the literature and the empirical findings, this thesis proposes that intuitive interaction, in addition to the momentary recognition of artefacts, emerges when a sequence of single operations that make up a compound action, through a person's practice and skill acquisition, merge into one automated operation.

And that the close proximity between the users and the developers during the development period may have lead to this system that was immediately usable by skilled users and easily learnt by novice users, by the minute adherence to the established practice among the users, where the main task is sketching and testing preliminary solutions. Thus being designed to treat mistakes and breakdowns as a source for development rather than something that is unwanted and has to be avoided.

Table of Contents

CHAPTER 1.

Introduction

The primary human approach to computing technology is through a user interface. There is a consensus that graphical user interfaces have made computing technology more accessible to the general public. To fulfil its task of translating complex technology into understandable, non-intimidating possibilities for action. Concurring with the guidelines for usability in user interfaces, a user interface should be easy to learn, easy to remember, and facilitating ease of use (ISO, 1998; Nielsen, 1993; Shneiderman et al., 2009).

Regular individuals often refer to ease of use as intuitive, equalling intuitive use to instinctive use, i.e. immediately knowing what to do, based on a hunch or gut feeling. Among researchers in the field of HCI, on the other hand, intuition is based on skills, deep knowledge, and experience. This contradictory understanding of the term ’intuitive’ makes it vague, and problematic since it means different things to different people. Current research on intuitiveness and intuitive interaction is presented and discussed in Ch. 3.

The term ‘best practice’ might be regarded as consultancy speech. It is, however, occasionally used in this thesis as a term representing either what has become a set of standards and established design patterns within the field of human-computer interaction and interaction design, or an established practice of work. With the introduction of usability thinking in technology development, we see that the conception of user interfaces being tools that should be intuitive to use have gained wider adoption. Also, the software that was studied in this thesis was designed as a collection of vocational best practices, including a significant portion of participation from the user.

The chosen object of study has been the design and use of a vocational information system for stowage planning of chemical tankers in one of the world's largest shipping companies within the field of sea transport of chemical liquids. Stowage planning of chemical tankers is a rather specific vocational niche. It is a highly complex task, in a highly skilled working environment demanding a strong domain-specific task competence of the workers involved in doing it. Stowage planning is a dynamic process, not only because no plans are identical; they may also be continuously revised, and as such a good candidate for a longitudinal study.

The framework of activity theory has been chosen to explore the relations between experience and intuitive interaction. Activity theory offers a set of concepts for analysing human activity, not as a momentary, static doing, frozen in time, but as a reflected and goal-directed, evolving sequence of mediated activity, developing over time, in a given context. This thesis utilises these principles in the discussion of what intuitive interaction can be, and how it might be designed. Activity theory, its roots, concepts and as a framework for HCI research, is reviewed in ch. 2.

1.1 Human-Computer interaction

During the last couple of decades of human socio-technical context, the number of computing devices that affect human activities, whether being in contact with the public through a website, paying bills in an online bank, operating the alarm-setting panel in their home, or being at work doing vocation-specific tasks, has increased considerably. From being a tool for the few, it has become a ubiquitous aspect of societal practice, to the extent that most human activities are mediated by a tool that facilitates a multitude of perspectives and approaches regarding flexibility and contextual awareness.

Human-computer interaction (HCI) is the study of how to design the patterns of how we mediate our interaction with the special tool of computers. As such, the field of HCI has grown into a rather broad and diverse discipline in the intersection of computer science, ergonomics, design, and social and behavioural science (Rogers, 2012). The amount and level of usage that screen-based media have reached in current work life and societal practice, where a large and increasing part of human activity, are conducted through computer-mediated activity, shows that screen-based user interface artefacts or other computer-related interfaces, increasingly, are being regarded as 'real' tools for action. This has led to many approaches in defining what HCI is. It includes a vast array of interfaces, from, e.g. tangible, gestural, brain-computer, and screen-based GUI, in both embodied interaction patterns, and interaction with a screen-based user interface, covering most aspects of human life, where the exposure to the complexity of computers (Norman, 1998) has reached an ever-increasing number of people who mediate their interaction with the world through computers and their user interfaces and what the user interfaces afford them to do. While originally termed by Gibson (Gibson, 1979) as the direct perception and action between animals and their environment; an affordance would, in the current HCI discourse, describe the perceived notion of action possibilities present in a computer user interface. It would explain how screen artefacts as symbolic representations in the user interface, based on the perception of similarity with the objects and actions from the real, physical world, are to be used or operated.

Until the early '90s, there was no firm agreement of a single, unified theory of HCI (Hewett, 1992, p. 5); rather a set of underlying principles and best practices that comprised the underpinnings for an understanding of human-computer interaction, based on cognitive psychology as a multidisciplinary field of both work and research (Löwgren and Stolterman, 2007). The ACM Special Interest Group on Computer-Human Interaction (SIGCHI) developed a definition of HCI:

> 'Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use in a social context, and with the study of major phenomena surrounding them.'
>
> [The Acm Sigchi Curricula for Human-Computer Interaction
>
> (Hewett, 1992, p. 5)].

In classic HCI theory in the early 80s, utilising a model based on human limitations and capabilities, like e.g. perception, attention span, memory (Card et al., 1983),the perspective of cognitive science saw the individual as a human information processing unit, consisting of three connected systems: perceptual, cognitive and motor, with their individual memory and processor (Rogers, 2012, p. 25), as a design guidance for user interfaces. In acknowledging the limitation of the cognitive approach to HCI, the theoretical framework of activity theory is brought into the discourse by Bødker (1991) in what Kaptelinin and Nardi call 'Post-cognitivist HCI: second-wave theories' (Kaptelinin and Nardi, 2003), as a tool for analysing actual use, including participation in design, the user interface as mediator, and expertise as a whole; where the individual is, as Bannon states, not a factor, but an actor, with needs and objectives.

A principal concept within the HCI discourse, which relates to human-technological interactivity in a greater sense than just computer related interaction, is usability. It applies to human-artefact activity in general, including work, workflow, cooperative activity in addition to human organisations as open systems, and the reciprocal impact computer-based systems and people may have on each other (Hewett, 1992). As a fundamental concept within the HCI discourse, *usability* has gained recognition to the extent that it is settled as part of ISO standard 9241, and is defined as:

> Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Where effectiveness, efficiency and satisfaction are described as:

> **effectiveness:** the accuracy and completeness with which specified users can achieve specified goals in particular environments
>
> **efficiency:** the resources expended in relation to the accuracy and completeness of goals achieved
>
> **satisfaction:** the comfort and acceptability of the work system to its users and other people affected by its use
>
> (ISO, 1998)

This relates to design processes of all technical artefacts of interactive systems with the intent of enhancing human-technological interaction (Ghaoui, 2005), which is reflected, partly overlapping in e.g. Nielsen's usability heuristics (Nielsen, 1995)[1] and Shneidermans golden rules (Shneiderman et al., 2016)[2], which presents a set of main characteristics of human-UI interaction that help developers in informing the design of user interfaces, i.e.: A user interface should always, and immediately or as quick as possible, respond to user input by showing what's currently going on [Visibility of system status]. Screen artefacts, i.e. icons, terms or concepts should be

1. These heuristics have been published in different versions in a variety of channels of the years, both on the web, and in several papers.
2. The eight golden rules have been published in many variations, both on the web, in papers, and the many editions of "Designing the User Interface: Strategies for Effective Human-Computer Interaction". The version cited here, is taken from section 3.3.4 in the 6th edition (2016).

familiar, and logic, to the user [Match between system and the real world]. The user interface should prevent errors from happening, by guiding the user, either by providing hints on correct procedure or limiting the number of operations to the ones that are relevant in the given context [Error prevention, Help and documentation]. The user interface should be consistent and coherent; that e.g. icons, symbols and terms would have the same meaning throughout the interface [Consistency and standards]. It should be easy to recognise and recover from mistakes, both by swift system feedback-alerts not leaving errors unnoticed, and by undo and redo possibilities [User control and freedom, Help users recognize, diagnose, and recover from errors]. Reduce the cognitive workload by making the screen artefacts in user interface visible and easily accessable [Recognition rather than recall]. These are in themselves examples of what would be regarded as 'best practices' within the field of HCI and interaction design, guiding developers in designing that are effective, efficient, satisfying, and - intuitive to use, although the focus errors lies within the traditional view that errors and mistakes should be avoided. Primarily relating to the discourse of momentary interaction, and is, as such, highly valid, it misses the value of mistakes as sources of learning and development, and subsequent competence derived from development as a source from which intuitive use-activity might emerge.

1.2 Problem area: the perceived character of intuitiveness in human/user interface activity

There has been an extraordinary lack of a theoretical discourse about the intuitive human use of computing devices through user interfaces. While a computer, and its user interface, is something that almost everyone has got a relationship with these days, it is a paradox that very few would be able to give any precise description of intuitive interaction with it, if asked. A paradox, due to human use of computing devices has seen an exponential growth for approximately the last two decades. Or is it a paradox? The Oxford English Dictionary defines intuition as 'The ability to understand something instinctively, without the need for conscious reasoning.',[3] which means that the term intuition would mean different things to different people, and in various contexts. Intuition, then, would be primarily relational, and unfocused. Also, professionally and academically, intuition as a term has been described as vague and problematic (Naumann et al., 2007).

However, since the 2000s, there is research literature that discusses evidence- or experience-based approaches to what constitutes the foundations of intuitive interaction in user interfaces. This research has focused on finding precise and objective criteria for how something, in a socio-technical setting, can be understood as intuitive. Established usability standards and guidelines, (ISO, 1998; Nielsen, 1993; Shneiderman et al., 2009) refer to being «user-friendly

This thesis will discuss intuitiveness as an aspect of activity, and from an activity theoretical approach. A central aspect in this framework is the cultural historical element of the use of tools - mediation (Vygotsky, 1978; Kaptelinin, 2015).

1.3 What is intuitive interaction?

An expression that has gained a foothold among developers and users, is «intuitive». However, the term might be perceived as imprecise, as it could mean different things to various groups of people, and in different contexts. These ambiguities have not been sufficiently debated, and the term has become one of the watchwords that have slipped into the vernacular and developer communities as something everyone wants, and that some developers strive to achieve. The problem is, what? And for whom? When regular users, i.e. ordinary people without any specific experience

3. https://en.oxforddictionaries.com/definition/intuition. | Accessed: April 1. 2017

with the use of computers, use computers talk about «programs being really intuitive», do they really mean *instinctive*? That they click and drag on boxes and arrows instinctively? (Blackler, 2006; Blackler et al., 2011; Blomberg et al., 1996). And how do they know that boxes and arrows are to be dragged and clicked on? If not being a learned operation, from being computer literate as a member of a modern culture, as e.g. discussed by Mohs (2006a).

According to Raskin, in a situation where a user recognises resemblance or something as identical, "intuition" would be "*an almost exact synonym of 'familiar'* ". Thus, interfaces being perceived as intuitive, depend on whether a person recognises the user interface elements as familiar (Raskin, 1994).

HCI research has primarily been focused on interaction, thus intuitive interaction, in a momentary perspective (Kuutti and Bannon, 2014), while the perception of intuitveness stems from experience (Blackler et al., 2011) and thus develops over time.

1.4 Research goals

Intuitiveness has been described as a rather vague term in the relatively limited literature in the field. This may be due to its ambiguous character, which, if it should be able to have a precise meaning, depends on context. One of the arguments in this thesis is related to the visual character of abstractions as graphical elements in the user interface being mediated affordances and situated in a context where the recognition of visual abstractions as a familiar trigger experience-based recollection of an activity comprising previous operations and activities. This is in part supported by Hurtienne et al.'s claim regarding Image schemas mainly as mediated affordances (Hurtienne and Israel, 2007), and Blackler et al.'s main conclusions that the perceived intuitiveness in performing tasks, stems primarily from previous experience (Blackler et al., 2011).

In this thesis I will connect intuitive interaction and recognising and remembering previous activities as the basis for performing use-activities as sequences of tasks, in an user-explorative approach. The empirical material, gathered from a skilled worker environment as an activity system, is analysed and discussed by the theoretical frameworks of activity theory and activity theoretical HCI, see literature review, ch. 2, and the present research on intuitive use of user interfaces, see literature review ch. 3.

In addition, the empirical material includes findings from the software design and engineering period, thus, aspects of the possible connection between the design process and to which extent the interaction was perceived as intuitive, are explored.

1.4.1 Research questions

The main question in this dissertation deals with intuitive interaction, and as such it must, necessarily, discuss some momentary aspects of interaction, but the choice of case, namely computer mediated interaction in a skilled worker environment, opens for a discussion of issues that also include human interaction with a user interface over time. In a discussion of human intuitive use of user interfaces, where the distinction between instantaneous interaction and interaction resulting from development over time, the intuition concept itself would have to include different, and partly contradictory, approaches to what it means or contains, both in relation to context, and an experiental knowledge. This thesis explores some possible elements of perceived intuitive interaction in human use of a computer user interface as tools in doing targeted work over time. In this thesis I will discuss that process through the concepts of activity theory, i.e. the human-computer-functional organ and its role in the developmental stream of actions in computer-mediated activity. My first research question is:

1. What is intuitive interaction?

In order to address this question, and how intuitive interaction can be *evaluated*, we need to discuss how intuitive interaction might be *described* and *recognised.* In the following parts of the thesis, when 'intuition' or 'intuitiveness' is mentioned without 'perception' or 'perceived' attached, it is still always the perception of something that is being describe as intuitive, thus on every occasion that interaction is described as intuitive, it is always the perception of the interaction that is described as intuitive.

It is not possible to design another person's *user-experience*, but for the many developing communities of socio-technical artefacts, in this thesis, mediating artefacts in a human-computer interaction context, an important aspect of intuitive interaction is:

2. How intuitive interaction can be designed?

It is easier to be critical of something that does not work as planned or is a mediocre solution to task solving and practice when users get blocked or delayed in performing tasks they are supposed to do than explaining specifically why something works brilliantly.

In computer-mediated activities, in a skilled worker context, a non-best, problematic, practice could mean that the vocation-specific software does not support established work practices. When something does not work as intended, the troubleshooting will be problem-oriented, in the sense that one can focus on the areas where the errors occurred. It is possible to pinpoint the exact cause of a why something did not work.

However, this dissertation presents a best practice case study, and it is significantly more difficult to pinpoint, among several possible explanations, the one, single solution, or a set of solutions, that might explain why something works well. In a best case context, it is necessary to search wide in the sequence of use-activities that together comprise the task-domain, in order to understand the character of a best practice, and how, in both content and order, it is composed.

In the case study, the best practice was based on patterns of continuous testing and probing various solutions, constantly fine-tuning the plans and processes. The case, presented in ch. 5, applied a 'best practice' approach to the design of the new software meant to facilitate extended simulation possibilities and letting the operators access these testing facilities by a user interface mediation that would be either immediately familiar, similar, or resemble this familiarity or similarity. This meant that the operators could transfer their domain-specific knowledge and experience; their acquired 'best practices', into operation of the program through a user interface.

1.5 Thesis structure

This section presents the overview of the thesis. Here, the chapters are described sequentially.

Chapter 1. Introduction

Perceived intuitive interaction has, for a long time, been a buzzword in the HCI community. However, among researchers it has been described as vague and ambiguous, meaning different things to different people, thus becoming a problematic term to use with a kind of certainty. In this first chapter, the research area of intuitive interaction, its background and research questions are presented.

Chapter 2. Activity Theory

The cognitive science/human-as-information processor-approach dominated the HCI discourse for many years, until the early nineties, when the Russian socio-cultural psychological framework of activity theory was brought to the west, as agile and potent analytical tool for analysing human-computer interaction contexts or computer-mediated activities. This chapter presents a short introduction to activity theory and a historical overview. The basic concepts and principles of activity theory are presented, followed by its potential as an analytical framework for HCI-research.

Chapter 3. Intuition

This chapter presents problematic aspects of 'intuition' or 'intuitive use', as being vague and meaning different things to different people, followed by a description of its etymological roots and current definition. This is continued by connecting 'intuitive' to graphical user interfaces, before presenting the current research (post 2000) on intuitive interaction. The chapter is concluded with a brief discussion of metaphors as representations in user interfaces.

Chapter 4. Method

In order to study the relation between humans and their use of computers through user interfaces, several methods and techniques are available. In this chapter the research approach, -strategy and -methods are presented. Non-participant observation and semi-structured interviews have been the main methods. To supplement the qualitative data, a small survey directed towards a group of hard-to-reach users: naval personnel at sea, was undertaken.

Chapter 5- Case - designing the stowage planning software in a shipping company

In this chapter, the case of design of the new stowage planning software, the company, and its domain-specific workflow are presented.

Chapter 6. Design, development and participation

In this chapter the design and development process, and the degree of user-participation, with co-locating the designer-team, is presented.

Chapter 7. Use of ORCA

This chapter analyses and discusses the part of the empirical material that is related to the use of the ORCA software, by the operators at the company headquarter on shore and the naval personnel, in their daily work of planning the stowage and transportation of chemical fluids at sea -utilising the activity theoretical HCI-discourse; analysing human activity over time, discussing development and streams of actions.

Chapter 8. Mediation in ORCA

This chapter analyses and discusses ORCA as a mediating tool. First the ORCA user interface, and the most frequently used screen elements is presented, with the interaction pattern connected to each screen artefact. Followed by a discussion of user interface familiarity and transparency, in an immediate, momentary approach to interaction, before relating to the ORCA user interface as a dialectical approach to handle contradictions.

Chapter 9. Concluding remarks and limitations

The conclusions from the research, together with contributions to theory and potential implications for practice are presented, followed by a presentation and a brief discussion of the limitations of the study. Additionally, some possible directions for further research are described.

The term *use-activity* is used throughout the thesis meaning general human doing without placing it the hierarchical structure of activity, in order to avoid a possible mix-up with the activity term connected to the top level in the structure of activity.

Literature

Chapter 2 and 3 sums up the central body of literature of activity theory related to HCI, and, at the time of writing, the seemingly, main portion of the current research within the field of *intuitive interaction in user interfaces,* respectively. First, the activity theoretical framing with its historical background, and basic concepts, principles, and structure are presented, before relating it to the HCI discourse. Then, the most current body of research on intuitive interaction in an HCI-context since 2000, and the aspect of knowledge in action, including the tacit dimension of knowledge as a base for intuitive interaction is reviewed.

The work of Aleksei N. Leontiev is extensively referenced in this dissertation. Several spellings of his surname are found in the literature, i.e. Leont'ev, Leontyev, Leontjew, Leontiev and so on. In the text, I consequently use "Leontiev", but the referenced literature might use alternative spellings.

CHAPTER 2.
Activity Theory

The concept of activity arising from human [actors] making use of consciously chosen artefacts is of particular importance, and therefore especially valuable when exploring human use of technology, as a conscious, purposeful activity (Nardi, 1995b, p. 7; Vygotsky, 1978).

My entrance into activity theory has been through activity theoretical HCI, introduced by Kaptelinin & Nardi (1995b), and Bødker (1991), working backwards through Leontiev (1978) to Vygotsky (1978) and back again, briefly including Engeström (1987), here presenting it historically chronological.

There are several varieties of activity theory.[4] The discussion in this thesis leans mainly on the approach that was developed by A. N. Leontiev (1975),[5] based on the cultural-historical psychology of L. S. Vygotsky (Vygotsky, 1978), and the body of literature, related to activity theory and HCI, initiated by Bødker (1991) and Bannon (1991), with Kuutti's work on activity theory as a framework for HCI research (Kuutti, 1995), and underlining of object dependent transformative activity structure (Kuutti, 1999), in addition to Kaptelinin and Nardi's foundational activity the-

4. 57, according to Rogers (2008).
5. English version. Russian version published 1959.

oretical HCI approach (Kaptelinin and Nardi, 2012a; Kaptelinin and Nardi, 2006; Nardi, 1995b).

The activity theoretical perspective of interactive action *extended* the cognitive science approach, where a user of software would be regarded merely as a cognitive information processing unit, which was rather usual within cognitive psychology at the time. In his much cited paper 'From human factors to human actors: The role of psychology and human computer interaction studies in system design', Liam Bannon presented a set of possible solutions to these issues. 'Within the HF (human factors) approach, the human is often reduced to being another system component with certain characteristics, such as limited attention span, faulty memory, that need to be factored into the design equation for the overall human-machine system. This form of piecemeal analysis of the person as a set of components de-emphasises important issues in work design. Individual motivation, membership in a community of workers, and the importance of the setting in determining human action are just some of the issues that are neglected' [...] 'By using the term human actors emphasis is placed on the person as an autonomous agent that has the capacity to regulate and coordinate his or her behaviour, rather than being a passive element in a human-machine system' (Bannon, 1991, pp. 27-29).

During the last two decades, activity theory has become a growing and central post-cognitivist approach and concept in HCI research (Bannon, 1991; Bødker, 1991; Kaptelinin et al., 2003; Kaptelinin and Nardi, 2006; Kuutti, 1995; Nardi, 1995b), questioning the previous information processing approach, introducing human needs, agency, thoughts and practice as aspects of human-computer interactivity.

Together with findings from previous research in the field of intuitive action, the theoretical foundation for the main discussion in this thesis, is situated at the junction of recent research of intuitive interaction in user interfaces (Blackler et al., 2011; Blackler and Popovic, 2015; Blackler and Hurtienne, 2007; Blackler et al., 2005; Bærentsen, 2000; Mohs et al., 2006b; Naumann et al., 2008; Mohs et al., 2006b; Blackler et al., 2011; Hurtienne and Blessing, 2007), and what has come to be termed as activity theoretical HCI, where activity theory was established as a theoretical framework from which HCI would benefit Bødker (1991), Bannon (1991), and continued by e.g. Nardi and Kaptelinin (Kaptelinin and Nardi, 2012a; Kaptelinin and Nardi, 2006; Kaptelinin et al., 2003; Nardi, 1995b). As such, the discussion of one of the most used terms or concepts related to human users of technology – intuitive interaction – leans on concepts from the activity theory framework in the analysis of the empirical material.

This chapter is structured as follows: first, a short introduction to activity theory as a framework for understanding the relation between consciousness and activity is presented. The introduction is followed by a historical overview, presenting its origin in Soviet cultural-historical psychology and development into a set of principles for an understanding of how human cognition and activity are related to context, and the diffusion from a primarily Russian field of research into the western realm, and subsequent adoption by other fields of research, in this case, human-computer interaction. The chapter continues by presenting the basic concepts and central elements found in the activity theory literature and concludes with presenting the central body of literature related to activity theory as an approach human-computer interaction research and practice.

2.1 Activity theory - a short introduction

The cultural-historical theory of activity, CHAT, or simply activity theory, is in the literature described mainly as a theoretical framework rather than a unified theory. As such, activity theory has surfaced as a philosophical framework for the study of human thought and agency, both as individuals and as members of a group (Leontiev, 1978, pp. 119-120; Engeström, 1999a, p. 19). Kuutti presents, briefly, activity theory as:

> "Activity theory is a philosophical and cross-disciplinary framework for studying different forms of human practices as developmental processes, with both individual and social levels interlinked at the same time" (Kuutti, 1995, p. 25).

The central tenet of activity theory is to understand the unity of consciousness and activity – in a complex context of motives and activities, goals and actions, tasks and operations, in a hierarchical structure that guides the way we mediate our interaction with the world (Leontiev, 1978), in an approach that acknowledges and includes the cultural and contextual aspects of human activity and development (Vygotsky, 1978).

An important element of activity theory is to understand the concept that activity emerges from human actors utilising consciously chosen artefacts (Leontiev, 1978; Vygotsky, 1978). It is, therefore, especially valuable or useful when exploring human use of technology, i.e. human activity mediated by a computer, as a conscious, purposeful activity (Kuutti, 1995; Nardi, 1995a, p. 7).

The activity theoretical framework is an integrated system of conceptual tools and theoretical approaches, and provides thus a conceptual model for analysing the relationship between humans as subjects, and the external world.

Activity theory focuses on practice, which obviates the need to distinguish 'applied' from 'pure' science – understanding everyday practice in the real world is the very objective of scientific practice (Nardi, 1995b, p. 7).

Within the HCI research community, Activity Theory gained increased attention in the early to mid 90s (Bannon, 1991; Bødker, 1991; Kuutti, 1995; Nardi, 1995b), and has been adopted, not as a fully developed, predictive theory, but as a framework for comprising the many elements of human activity, i.e. personality, thought processes, actions, goals and needs in the study of human practice and behaviour, both as individuals and as members of a collective. Through a set of principles, activity theory affords exploring human use of technology, as a conscious, purposeful activity, through an abstracted or metaphorical tool like i.e. the screen-based graphical user interface, which constitutes key areas for what is regarded as the fundamental unit of analysis in activity theory: human activity and its three main characteristics in being goal- or object-directed, mediated by artefacts, and contextual within societal practice. In this thesis, activity theory affords the analysis of the elements of a purposeful, *computer mediated human activity,* as its unit of analysis.

2.2 Historical overview

The term "activity theory" can, historically, trace its roots to the ideas of human thought and agency in Activity Theory originate from the cultural historical school of Soviet psychology in 1920-30s, founded by Lev Vygotsky (1978), and further developed by A. N. Leontiev (Leontiev, 1978), and A. R. Luria (Luria and Cole, 1978; Luria et al., 1979) as the founding group of researchers, pioneering the concept of mediation of human activity. Succeedingly developed further by some of their students (Davydov, 1999b; Zinchenko and Gordon, 1981), and subsequently introduced to and developed in western psychology, and educatoion and learning by e.g. Engeström (1987), Cole (1996) and Wertsch (1998) in addition to the, already mentioned works of Bødker, Kuutti, Bannon, and Kaptelinin and Nardi in the field of human-computer interaction.

Vygotsky introduced the central concept that human activity originates from having an objective and being mediated through tools. Mediation is a dominant theme in Vygotsky's work, and he states that human consciousness is characterised by the use of psychological tools, like symbols and languages - of cultural-historically conditioned representations, and that our interaction with the real world is indirect and mediated through societal and culturally constructed tools.

> The communication of consciousness can be accomplished only indirectly, through a mediated path. This path consists in the internal mediation of thought first by meanings and then by words. Therefore, thought is never the direct equivalent of word meanings. Meaning mediates thought in its path to verbal expression. The path from thought to word is indirect and internally mediated (Vygotsky, 1987, p. 282).

Also, a central element in Vygotsky's mediation concept, is the reciprocality with which we relate to tools, e.g. technology, where we are being shaped, and develop as human beings, by employing what he coined 'higher psychological functions' - by our use of cultural artefacts. This notion of culturally mediated actions would lean on the principle of internalisation, where external mediation develops into actions based on internally mediated psychological tools (Kaptelinin and Nardi, 2012a; Leontiev, 1978; Vygotsky, 1978).

> The tool's function is to serve as the conductor of human influence on the object of activity; it is externally oriented; it must lead to changes in objects. It is a means by which human external activity is aimed at mastering, and triumphing over, nature. The sign, on the other hand, changes nothing in the object of a psychological operation. It is a means of internal activity aimed at mastering oneself; the sign is internally oriented (Vygotsky, 1978, p. 55)

In his universal law of development within the cultural-historical tradition, he states that we reflect on the employment of technical (physical), functional, and culturally founded psychological tools, such as e.g. patterns of notation or tutorial schemas. This mediational triangle structure of behaviour, where the relation between stimulus and response is mediated by tools, is shown in figure 1.

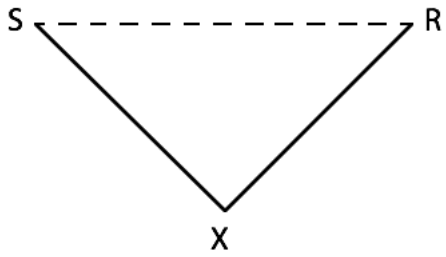


Figure 1. Vygotsky's depiction of mediated action, where he introduces 'sign operations' and the requirement for an 'intermediate link between stimulus and response', establishing the conception of a 'complex mediated act' (Vygotsky, 1978, p. 40).

In its most simple original version, the starting point is the activity; an action undertaken by a human, the subject, towards an object in order to achieve an outcome, and where the action is mediated by a tool. (figure 2).

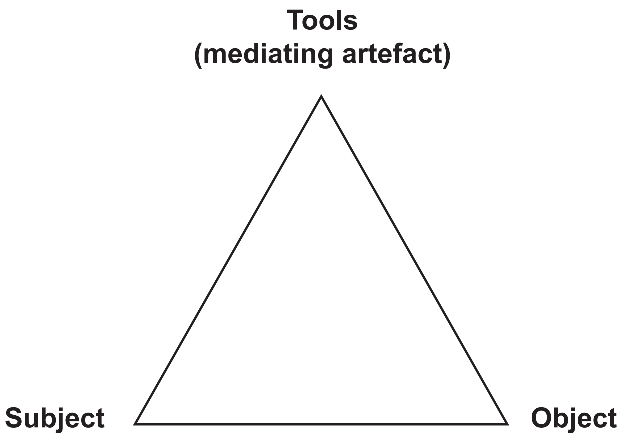


Figure 2. Revised version of Vygotsky's mediational model of human interaction with the environment. There exist a number of depictions of this version of the mediational model, with the tool on top.

The foundational concept of activity theory is the dialectical and developmental unity of consciousness and activity, which must be perceived as an interaction between a conscious, enacting subject, and an object, that must be recognised as an object positively existing in the real world (Vygotsky, 1978).

In addition to this primary concept, there are the following basic principles; object-orientedness, hierarchical structure of human activities, internalisation-externalisation, and mediation, on which activity theory is based (Kaptelinin and Nardi, 2006; Leontyev, 2009; Wertsch, 1981).[6] Vygotsky did not present his work as activity theory, but the cultural historical school and activity theory share the views on consciousness and mediation, with the slight difference in cultural historical direction focus on mediation by mental artefacts such as language, while activity theory has emphasised mediation by tools, without being incompatible (Zinchenko, 1995).

Leontiev extended Vygotsky's individual focus into situating human activities in a collective context - an activity system, showing the differentiation of individual action and collective activities through the division of labour (Engeström, 1999c, p. 5).

For an extended period of time it was largely a Soviet 'phenomenon', until Vygotsky's and Leontiev's work was translated into English and introduced to an international psychology audience (Luria and Cole, 1978; Luria et al., 1979; Cole, 1990) and Wertsch (Wertsch, 1981). During the last two decades — the post-cognitivist period (Kaptelinin and Nardi, 2012a) activity theory has been described and adopted as a useful analytic framework for HCI and interaction design, and human agency in a human-computer interaction context (Bannon, 1991; Bannon and Bødker, 1989; Bødker, 1991; Kaptelinin, 1996a; Kaptelinin et al., 2003; Kaptelinin and Nardi, 2006; Kuutti, 1995; Nardi, 1995b).

Vygotskys concept of tool and sign mediation (Vygotsky, 1978), and Leontiev's concept of activity (Leont'ev, 1974; Leontyev, 2009) was extended by Engeström (1987), in his development of the Scandinavian direction of activity theory, thus, bringing it, alongside Cole (1988; Cole and Griffin, 1980), and also Wertsch (1981), into the western realm, primarily in the context of organisational learning. Engeström (1987) uses the core ideas from Vygotsky (1978), and further develops

6. A. N. Leontyev, The Development of Mind, 2009 published by Marxists Internet Archive Publications. This is a reproduction of the Progress Publishers 1981 edition, plus "Activity and Consciousness," originally published in "Problems of Dialectical Materialism. Philosophy in the U.S.S.R." by Progress Publishers, 1977.

Leontiev's activity theory framework (Leontyev, 2009), by re-conceptualising our view on activity - as a system of collective processes, and thereby expanding its mediational model as an activity system, a model that considers both the cooperative and shared character of human activity, by utilising concepts such as division of labour, community and rules, in addition to subject, object and mediating artefacts.

The historical presentations of when and how activity theory originated and evolved is usually explained as a linear process (Engestrøm 1987 in Kapelinin 2012), developed by Vygotsky, Rubinshtein, Leontiev, Luria and other Russian researchers (Kaptelinin et al., 1995), before the theory is brought to the West by Engestrøm into other areas, such as organisational learning and pedagogy, and subsequently moved on with Bødker and Bannon (Bannon and Bødker, 1989; Bannon, 1991; Bødker, 1991) to the field of human-computer interaction where researchers in the late 80s had acknowledged the need for a new theoretical framework that not only discussed human-computer interaction in an information processing approach, but a theoretical foundation for how to include human needs and activity related to the context of HCI research and development. This linear presentation, which makes traversing the activity-theoretical literary landscape slightly less difficult (but still not easy) is, according to Kaptelinin (Kaptelinin and Nardi, 2012a), a bit too simplistic regarding the actual development-historical process associated with AT. I have still chosen this simplistic genealogy since it would make the historical presentation somewhat easier to follow while not having any decisive impact on the discussion.

2.3 What is activity?

Davydov categorises activities as e.g. productive, developmental, playful, object-manipulating, and comprise a hierarchy in order of 'importance' which he presents as: work, artistic activity, activity linked moral issues, followed by activities of law, with religious activities near the bottom of the list, and lastly: sport (Davydov, 1999a, p. 44).

An activity will arise from something being done by someone for the purpose of achieving a result of one kind or the other which means that activities differ from each other by being targeted at various goals, based on different needs and their accompanying motivation. The successful achievement following the transforming of an object into a result is the foundation for the entire activity. The principle of objectiveness is essential in Leontiev's theory of activity, as he states that an activity cannot exist without a purpose (Leontyev, 1977, p. 182), and that an activity is characterised by its duality, that an objective exists, initially, independent of the activity, subsequently transforming the activity by giving it a specific direction, reciprocally also transforming the subject in the subject's process of reflecting on the idea or notion of the object, but also that the object of an activity may be recognised by the subject's motive (Leontiev, 1978). Therefore, human activity is characterised both by its objectiveness, and its subjectiveness in that the activity of the subject must consistently be directed towards the transformation of an objective which is capable of meeting a specific need. (Leontiev, 1978, p. 12).

The process of transforming an object into an achievement is characterised by being a dynamic process that forms an 'ebb and tide' kind of motion between the subject and the object. Leontiev describes this as a circular process,

> The subject-activity-object transitions develop a kind of circular movement, so it may seem unimportant which of its elements or moments is taken as the initial one. But this is by no means movement in a closed circle. The circle opens, and opens specifically in sensuous practical activity itself. Entering into direct contact with objective reality and submitting to it, activity is modified and enriched; and it is in this enriched form that it is crystallised in the product.(Leontiev, 1978, pp. 10-11)

Leontiev (1978), expanded Vygotsky's work by conceptualising the 'theory of activity', and structuring activity as comprised by a need-motive, goals and conditions, in a hierarchical structure, with the 'main' activity and the actions and operations at three different hierarchical levels that constitute the activity. These three hierarchical levels consist of 1. motivated activity, 2. goal-directed actions focused on fulfilling the individual goals of the actions and which will together fulfil the need for the activity, and 3. the single operations comprising the actions, and which are linked to the prerequisites for the undertaking of the operations (see chapter 2.4.2).

Human activity consists of actions or a group of actions in sequence, in a conscious process of achieving or fulfilling the object of the activity. The goal of an activity encourages the activity itself. Leontiev argues that actions comprising an activity are connected to the notions of needs and motives (1978). This is also discussed by Davydov (1983, 1999a), who claims that actions are only connected to needs, and that this need is based on a desire to fulfil a task:

> Actions as integral formations can be connected with nothing but needs based on desires, and the actions aimed at fulfilling certain tasks stem from motives. Motives in their tum are specific forms of needs in the case when a person has set himself a task and is undertaking certain actions to fulfil it. Thus motives are consistent with actions. Actions are based on motives, and acting is possible if certain materials or sign and symbol means are available. (Davydov, 1999a, pp. 42-43)

According to Davydov, an object is not an independent, physical, thing in itself, but something that a subject acts towards, as a goal or purpose regardless of the activity being internal as a reflection, or external, following the character of the motive, that can be both material or conceptual, and as such will facilitate an activity being able to encompass several approaches in ensuring an object's ability to guide the direction of an activity. This may be affective elements i.e. emotions, needs, will, tasks, plans, means, motives. In a structure with several possible approaches, according to Davydov, it will be the need that will be the most important (Davydov et al., 1983).

In recognition of the importance of the collective aspect of human activity, Leontiev argues that,

> 'This first description now, after a quarter century, appears in many ways unsatisfactory and too abstract. But it is exactly owing to its abstractness that it can be taken as an initial departure point for further investigations. Up to this point we were talking about activity in the general collective meaning of that concept. Actually, however, we always must deal with specific activities...' (Leontiev, 1978, p. 62).

Leontiev argues that activity is a system of hierarchical levels where he distinguishes between collective activity and individual action, structuring human activity as a 'division of labour', as an important aspect in the process of the development of mental functions (Leontyev, 2009). Davydov discusses this as communicative action, arguing that communication reveals the cultural and interpersonal relations in an activity system, and inseparably linked to the activity itself (Davydov, 1999b, pp. 46-47).

2.4 Basic principles

In this chapter I will present and describe the basic elements of activity theory. I will present the concepts of object orientation, and point towards the underlying motivation for the activity, and how the process of human activity is divided and distributed in a hierarchy of levels, showing how an activity comprises consciously chosen sub-tasks, which in turn are undertaken by a string of single operations. These operations are internalised and externalised in a back and forth dialectical pattern, mediated by mental and physical tools towards the achievement of a primary goal.

Activity theory is, rather than an established theory, composed of a collection of principles that, together, constitutes a conceptual system. An activity is, broadly speaking, any instance of a subject's interaction with the world. In activity theory, this act of interaction is characterised by the subject's purpose and contextual motive, and is oriented towards an objective.

Activity theory separates between processes at different levels in a hierarchical order. An activity consists of a set of actions which are guided at an intended result which, in turn, are realised through operations (Leontiev, 1978; Kaptelinin and Nardi, 2012a; Kaptelinin, 1996a; Kaptelinin and Nardi, 1997), see figure 3, 'Leontiev's hierarchical structure of activity', page 35.

Activity theory differentiates between activities in the mind and external activities, but that the two cannot exist separately. This dual character of activities, i.e. they have both an internal and external side that are connected, is a crucial aspect of 'activity' within activity theory, and central to the principle that actions in the mind and actions towards the world - internal and external actions - mutually affect us.

The notion of humans making and using tools in order to mediate their interaction with the world is one of the key principles within activity theory (Vygotsky, 1978). According to Vygotsky, there are two categories of tools: technical ones and psychological ones. The purpose of technical tools is to manipulate physical objects, for instance, a pencil, while psychological tools are used by a person to influence another person or, reciprocally towards themselves, for example, a note containing a symbolic or linguistic help of some sort.(Vygotsky, 1987)

In an HCI research context, *activity* is a primary theoretical construct (Kaptelinin and Nardi, 2012a), and represents a relational context in which a [human] subject (Leontiev, 1978), interacts, physically or cognitively, with an object, positively existing in the world. This relationship is characterised by the subject's needs being met through the subject's interaction with the world, where the subject's activity towards an objective, and world and mutually influence each other (Kaptelinin and Nardi, 2012a).

Also, the relational aspect of the unity of consciousness and activity is a foundational principle of activity theory, where this relation is characterised as being asymmetrical, where [conscious] subjects possess *agency*, and have needs, unto which performing activities are necessary in order for the needs to be fulfilled. It is through this interaction that consciousness, the human mind, becomes an element in the activity 'matrix'.

Building on Wertsch (1981), Kaptelinin and Nardi (2006), identified the main principles in activity theory. They are here described in the order presented by Kaptelinin and Nardi.

2.4.1 Object-orientation

As the object of activity, represents one of the most basic concepts in activity theory, it plays a vital role in research that is utilising activity theory as the analytical framework. According to Leontiev, the main characteristic of activity is that it has a purpose - an object, and that all activity is object-oriented:

> The expression "objectless activity" has no meaning at all. Activity may appear to be objectless, but the scientific investigation of activity necessarily demands the discovery of its object. Moreover, the object of activity appears in two forms: first, in its independent existence, commanding the activity of the subject, and second, as the mental image of the object, as the product of

> the subject's "detection" of its properties, which is effected by the activity of the subject and cannot be effected otherwise (Leontyev, 1977, p. 182).

One initial problem of the object-oriented character of activity in activity theory is its somewhat linguistic lack of a distinct direction - in that the English word *activity* does not, in itself, contain or provide any connotation or guidance as to whether an activity has a certain direction or goal. Also, as stated by e.g. Kuutti (1995) and Kaptelin (2005), the typical English use of the term activity is too encompassing for the term to have any direction in itself, and that activity, as in object-oriented activity, is invariably connected to *agency* and associated with a transformation process. The notion of purposeful activity may be better expressed in German by the terms *tätigkeit* and *handlung*, which are closer to the original Russian term *deyatel'nost*. Likewise, the term object, as the foundation for purposeful activity, can be translated from the two Russian words *objekt* and *predmet*, which have almost the same and, often interchangeable, meaning. Leontiev (78,81)[7] in Kaptelinin (Kaptelinin, 2005, p. 6) describes this difference with the meaning of *object* relating to physical things existing independently of the mind, while the meaning of the word *predmet* would relate to a target or purpose in the mind, in the character of a thought or an action. Subsequently, Leontiev stated that the object of an activity should be understood as primarily related to "something at which an action is directed", i.e. the predmet approach to the term object (Leontiev, 1978; Leontyev, 2009, p. 29), where activity would be object-related, as in purposeful - *predmetnaja dejatel'nost* (Kaptelinin, 2005, p. 7).

Human activity is characterised by transforming the object of an activity into a planned or wanted outcome. The main characteristic of an activity is that it has a purpose - an object, and that all activity is object-oriented. An 'object' in activity

7. This book was reproduced in 2009, as The Development of Mind (Leontyev, 2009). Available from Marxist Internet Archive, Pacifica, CA. (https://www.marxists.org/archive/leontev/works/development-mind.pdf)

theory relates to the objective of an activity, as a source of motivation and guide in the process of transforming the object into an outcome (Kuutti, 1995).

According to Leontiev (Leontyev, 1977), the principle of object-orieation is:

> The expression “objectless activity” has no meaning at all. Activity may appear to be objectless, but the scientific investigation of activity necessarily demands the discovery of its object. Moreover, the object of activity appears in two forms: first, in its independent existence, commanding the activity of the subject, and second, as the mental image of the object, as the product of the subject's “detection” of its properties, which is effected by the activity of the subject and cannot be effected otherwise (Leontyev, 1977, p. 182).

An activity occurs to satisfy a motive or a human being's *objective* with the subsequent need to engage in activity. Therefore, the objective of an activity can be identified through the motive. However, even as the objective of an activity stimulates the activity, it is not given that it leads the direction of the subsequent activities that may exist within that activity. Therefore, the motive, coming from need, represents the necessary precondition for an activity to occur. A motive could be explicit or inexplicit (material or ideal), it can be perceived or imagined. This explanation highlights the view that there is no such thing as a motiveless activity. Therefore, "an activity does not exist without a motive; 'non-motivated' activity is not activity without a motive but activity with a subjectively and objectively hidden motive," (Leontiev, 1978, pp. 62-63). An activity is therefore driven towards the satisfaction of the motive or need. However, Kaptelinin (2005), claims that this object relatedness is not without “uncertainties and inconsistencies”, pointing to the characteristics of a lack of separation between the objective of an activity and the motive of an activity as problematic.

Within activity theory, the relationship between a *subject* and an *object* is mediated by tools. A *subject* is a living entity with intentions and agency to act upon things and objectives, to fulfil its needs or goals, as opposed to other elements within a particular context, that are, however useful, graspable - physical or cognitively, without agency. Within this context, the concept of agency points to the capability of an individual *subject* to, independently, act upon an object in the world, and change the context within which the objective is situated. Leontiev describes this reciprocal process:

> In activity an object is transformed into its subjective form or image while at the same time activity passes into its objective results and products. In this regard activity emerges as a process that effects a reciprocal transformation between the subject-object poles. […] Activity that is internal in form, having arisen out of external practical activity, is thus not separated from it and does not rise above it, but retains its basic, two-way connection with it.(Leont'ev, 1974, pp. 9, 22).

Collectively, the unity of subject and object is part of the framework of activity, as e.g. defined by Davydov et al. in their analysis of Leontiev's activity approach:

> "[…] human activity is characterized not only by its objectiveness but also by its subjectiveness: the activity of the subject is always directed toward the transformation of an object that is able to satisfy some specific need. Activity brings together in a unity such opposing principles as object and subject." (Davydov et al., 1983, p. 32).

Kuutti, focusing on the structure of activity states:

> "An activity is a form of doing directed to an object, and activities are distinguished from each other according to their objects. Transforming the object into an outcome motivates the existence of an activity. An object can be a material thing, but it can also be less tangible (such as a plan) or totally intangible (such as a common idea) as long as it can be shared for manipulation and transformation by the participants of the activity." (Kuutti, 1995, p. 27).

Object orientation is a term that has a different meaning within applied computer science than within the activity theoretical framework. Since [activity theoretical] human-computer interaction is situated within the computer science discourse, it would possibly facilitate a good reading experience to clarify that the meaning of the *object* in activity theory relates to the objective of an activity, as a source of motivation and guide in the process of transforming an object into a *wanted outcome* - as the purpose of an activity, and not necessarily a physical thing that exists in itself, as described by e.g. Leontiev (2009), and Davydov (1999a).

In a human approach, most of the objects we are surrounded by, and use, when working towards achieving a goal, are culturally mediated, either as a thing or its abstraction - a symbol or, as in an HCI context - a screen element in a user interface with which humans interact. Also, the character of objectiveness could be an established or a priori fact, which means that not only tangible artefacts but also social and cultural properties can be as real as physical ones, or other kinds of non-disputable entities, which means that the character of an object, its colour, temperature or surface, is both a measurable or perceived quality (Davydov, 1999a; Leontyev, 2009; Kaptelinin and Nardi, 2012a).

2.4.2 Hierarchical structure of use

Activity is defined by the contextual motive and necessitated by the objective, in the subject-object interaction, and can be represented as a hierarchical structure (Leontiev, 1978). Activity consists of consciously directed sub-tasks, actions, with goals representing the result of the action, which is put together by single operations that are not characterised by conscious direction, or even reflected upon, but are related to the conditions for the operations. (Leontiev, 1978; Zinchenko and Gordon, 1981; Kuutti, 1995, p. 30)

In Leontiev's three-levelled hierarchical structure, the mid-level is relating to goals being the objective of actions that, together with other actions, can form a group of actions, each with their own goals connected to each action in the group. While actions are the result of conscious doings, they are consisting of single *operations*, a part of human activity which may be undertaken in a kind of automatic approach, due to being situated in context, or dependent on what in activity theory is defined as a condition. For instance, when reading a book, I don't have to consciously think of turning the pages in order to continue to read. Unless I have a physical handicap, the turning of the page is not a goal in itself. Also, I read the book without thinking that I am technically reading the words on the page. As a literate adult, reading as a single technical operation in itself is without a goal, but dependent on the conditions in which it occurs. Can I sit down while reading? Is there a sufficient amount of light, so I can see the printed words on the page? On top of the hierarchy, we find the primary task and base for human activity, and that is motivated by an objective [of achieving an outcome]. The objective of an activity stimulates the activity, but it is not given that it leads the direction of the subsequent activities that may exist within that activity. The motive therefore, represents the necessary precondition for an activity to occur.

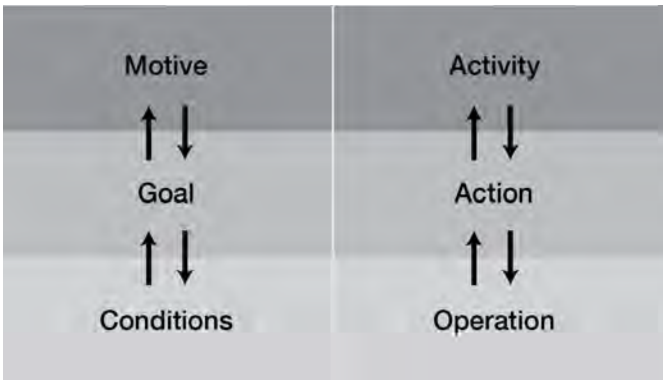


Figure 3. Leontiev's hierarchical structure of activity as continuous reciprocal transformation between the parts of activity (Davydov et al., 1983, p. 36; Leontiev, 1978; Leontyev, 2009). Depiction derived from Davydov et al. (Davydov et al., 1983, p. 36), and visually based on fig. 2.3 in Kuutti (Kuutti, 1995, p. 30). We see that an activity is initiated by a motive, and is comprised by a set of actions based on a conscious direction for each specific action, for which undertaking in turn is dependent on a sequence of operations that is not goal-directed, but related to the conditions for the operations.

As an activity occurs to satisfy a motive or a human being's objective need to engage in activity, the objective can be identified through the motive. A motive could be explicit or inexplicit (material or ideal), it can be perceived or imagined. This explanation highlights the view that there is no such thing as a motiveless activity. Therefore, "an activity does not exist without a motive; 'nonmotivated' activity is not activity without a motive but an activity with a subjectively and objectively hidden motive," (Leontiev, 1978, pp. 62-63).

Human **activity** exists only as actions or a sequence of actions, and is undertaken as a consequence of a purpose or a need - an objective towards which all human activities are directed (Leontiev, 1978). It is the contextual foundation for understanding the actions, or group of conscious actions that point towards the goals that together with other actions comprise the activity that leads to the fulfilment of the objective. However, this structure can change. An activity may lose the motive that generated the objective (Leontiev, 1978). This may lead to the object of an activity instead becomes the goal of an action, thus transforming the activity into an action, leading to the borders between activity and actions becoming fuzzy (Davydov et al., 1983). This is a bidirectional process; a mutual pattern of transformation, making the different parts of a use activity structure flexible and mobile, and in a continuous state of change and development, as a constant, reciprocal relationship between subject and object, where the subject reflects over, and transforms, the object, and is equally transformed (Leont'ev, 1974; Davydov et al., 1983, p. 36; Kuutti, 1995, p. 27).

Also, an action may evolve into a separate objective, thus turning into an activity. An action may also become a resource for reaching a goal, or as a result of skill acquisition, thus transforming into an operation (Leontiev, 1978, p. 104). This bidirectionality is further explained by Davydov, Zinchenko, and Talyzina:

> Thus an activity can lose its motive and become an action, and an action can become an operation when the goal changes, or as a result of user practice. The motive of some activity may become the goal of an activity, as a result of which the latter is transformed into some integral activity. Hence, mutual transformations are constantly taking place: activity ⇄ action ⇄ operation and motive ⇄ goal ⇄ conditions. The mobility of the constituents of activity is also manifested in the fact that each of them may become a part of a unit or, conversely, come to embrace previously relatively independent units (for example, some acts may be broken down into a series of successive acts, and, correspondingly, a goal may be broken down into subgoals) (Davydov et al., 1983, p. 36).

Actions are conscious goal-directed processes that must be undertaken to fulfil the objective of an activity, made up of a series of singular *operations*, related to condition and context, that users, in its sequential character of processing the operations, and thus influence on the conditions for attaining the goals. (Kaptelinin and Nardi, 2012a; Kuutti, 1995; Leont'ev, 1974). Initially, however, an operation would be the result of a conscious act: an action, until the subject has gained sufficiently experience, having internalised it, and no longer needs to reflect on it; thus it would subsequently be transformed into an operation, undertaken unconsciously and automatically, and placed into the sequence of other automated operations. This "action-operation dynamic" and the increase of sense of utilisation by actions resulting from this reciprocal internalisation-externalisation is a typical aspect of human development (Kuutti, 1995, p. 31). A typical example of this process is, learning to play an instrument. In order to produce an audible tone, one must concentrate on putting the fingers in the right place. To produce not only an audible tone, but also a tone of a certain quality that others would, presumably, enjoy listening to, one would also have to put the finger in the right place, in the correct way, depending on the instrument. When playing on a string instrument, e.g. the guitar, a correctly placed finger, would, in addition, have to press the string with a sufficient amount of power against the fretboard, and simultaneously, pluck the string with the other hand in order to produce a good sounding guitar tone. At a novice level, every one of these individual operations would be an action. The first goal would be to place one's finger in the right place, i.e. play the correct tone. The goal of the next action could be that the tone should sound good or with a certain amount of sustain. For an inexperienced novice, all these single operations would be quite complex actions, that would include intricate coordination involving both arms and hands, and mental functions. While, assuming all the conditions were set and stable, an experienced musician would, automatically and unconsciously, combine them into a single smooth operation.

Likewise, changing conditions could alter the foundation for an automated undertaking of operations, and lead the subject into conscious reflection, transforming the operation into a conscious action (Kuutti, 1995). In the instrument example above, a changed condition could be an unplanned change from an electric to an acoustic setting, forcing the musician, here: a guitar player, to be more aware of using an increased force in the plucking of strings, and reflect on how much the plucking style has to be changed in order to reach the goal that everybody should be able to hear the music. In this example, the automated operation of plucking a string was transformed into a conscious action, until this new way of plucking strings would be internalised and fully adapted to the new condition, when it would transform back into an operation.

2.4.3 Mediation

The principle of mediation is a central concept that human activity originates from having an objective that is fulfilled by being mediated by the conscious use of tools as an intermediate link between the subject and object. Mediation is a dominant theme in Vygotsky's work, and he states that human consciousness is characterised by the use of psychological tools, like symbols and languages - of cultural-historically conditioned representations, and that our interaction with the real world is indirect and mediated through societal and culturally constructed tools.

> The communication of consciousness can be accomplished only indirectly, through a mediated path. This path consists in the internal mediation of thought first by meanings and then by words. Therefore, thought is never the direct equivalent of word meanings. Meaning mediates thought in its path to a verbal expression. The path from a thought to a word is indirect and internally mediated. (Vygotsky, 1987, p. 282).

Also, a central element in Vygotsky's mediation concept, is the reciprocity with which we relate to tools, e.g. technology, where we are being shaped, and develop as human beings, by employing what he coined as 'higher psychological functions' - by our use of socially and culturally mediated artefacts. This notion of socially and culturally mediated actions would lean on the principle of internalisation, where external mediation develops into actions based on internally mediated psychological tools (Vygotsky, 1978; Leontiev, 1978; Kaptelinin and Nardi, 2012a).

> The tool's function is to serve as the conductor of human influence on the object of activity; it is externally oriented; it must lead to changes in objects. It is a means by which human external activity is aimed at mastering, and triumphing over, nature. The sign, on the other hand, changes nothing in the object of a psychological operation. It is a means of internal activity aimed at mastering oneself; the sign is internally oriented (Vygotsky, 1978, p. 55)

Kaptelinin et al. argues that activity theory in itself is "built upon the concept of mediation", which makes it particularly suitable for HCI "exploration" (Kaptelinin et al., 1995, p. 190).

Both in societal and organisational life, at work or socially, an increasing part of human activities is performed through an interface, as a unique and, seemingly, ubiquitous tool in mediating human interaction with the world. Kaptelinin discusses the theoretical implications this would have for the field of HCI, and argues that it is not the interaction with computers that is the primary goal in itself, leading it to become what Kaptelinin terms as Computer-Mediated Activity (Kaptelinin, 1995b). Thus, the notion of mediation is an important concept within the cultural-historical psychology (Vygotsky, 1978; Leontiev, 1978), as it represents not only the foundation for all human praxis but also what makes us uniquely human, also including the comprehension of 'similarity' within the process of transforming objects, that would be central in transforming existing knowledge into new knowledge.

Every aspect of what might be called the successful by the human species - nearly all human interactivity with the world and everyone and everything in it is culturally and technically mediated in some form, either by physical objects or signs, or in the internal plane as language or other symbolic artefacts (Kaptelinin and Nardi, 2012a; Leontiev, 1978). We can build cities and live in them. We can communicate without being physically or simultaneously present. We may work towards a target, by acting indirectly, using either abstract symbols, or by combining actions, that separately do not lead to the achievement of a goal, but combined, constitute an activity of a logical and sequential string of actions that lead to the achievement of a desired goal. The construction and use of tools and language are described as defining properties of higher mental functions and, therefore uniquely human (Vygotsky, 1978; Wertsch, 1991). The distinction of human interacting agency as it unfolds in all its complexity, societal and technological, the physical and the intangible is powered by mediation. Leontiev describes this relationship of humans and the enabling tools that extend human abilities into higher achievements that would not have been possible without functional organs (Leontyev, 2009).

Zinchenko (1995), discusses the mediation of a human-tool relationship and specifically the relationship that arises out of extensive and longitudinal interaction to such an extent that the activity, the interaction, becomes dissolved into an intuitive singularity; i.e. that the human and the tool melt together through a "master's introspection", and presenting the conception that all elements that comprise the modern socio-technical contexts and materiality, are contributing to the experiential pattern repository, consisting of both previous actions and their representative metaphors which we can access and recognise in similar or related artefacts and their related contexts and recollecting previous praxis.

The special character of the user interface, mediating human interaction with the world, is versatile in the sense of being able to mediate a multitude of activities, thus encompassing several relations (Kaptelinin, 1995b; Kaptelinin, 2015), as a tool into which the user, in the internalisation process, can externalise and distribute simulation capacity, forming a functional organ, subsequently influencing learning and development.

2.4.4 Internalisation and externalisation

Internalisation was introduced by Vygotsky as "the internal reconstruction of an external operation" (Vygotsky, 1978, p. 56), as an intermediate approach to the conceptualisation of the relation between internal and external activity, which previously had been approached either from a Cartesian point of view, focusing on the internal mental activity as the sole domain of psychological study, or the contrasting behaviouristic approach focusing on external behaviour as the only point of departure in psychological study (Wertsch and Stone, 1985).

Internalisation describes how external elements or activities in the real world *becomes* an internal plane, "hosting" the internalised processes (Leontiev, 1978). By transforming external activities into internal ones, this process provides a possibility to simulate potential interaction with the world in the mind, in the form of considering various strategies, by mentally, and initially, testing possible actions and operations - in the mind, making reflections that had not previously been present. Leontiev states:

> Internalization is thus not a process by which external activity is *transferred* onto a pre-existing inner "plane of consciousness": it is the process by which the inner plane is *formed* for the first time.(Leont'ev, 1974, p. 20)

Internal activity, originating from external, physical activity cannot, however, be separated from this external activity, but is expressed as a constant *reciprocal* relation with external activity, where human mental processes develop and redevelop. These continuous reciprocal processes are possible because both the internal (mental, in the mind), and the external (practical, physical) activities have fundamentally the same structure, as there are no boundaries between the internalisation and externalisation processes (Leont'ev, 1974, pp. 21-22).

This dual character of human activity is also expressed by Cole (1996, pp. 136-137) who, in his discussion of human activity and the relationship between internal and external activity, states that human and activity cannot be separated from each other, since the physical aspect of activity is the externalised form of the person's activity in the mind.

Internalisation is presented as a simulation process that takes place in the *internal plane of action*, without actually performing them in reality. This internalisation process - the planning of *what* is going to be done, *why* it is needed, the simulation of *how* it is going to be done, and the consideration of the potential outcome of the activity, prior to actually performing an action in the real world (Leont'ev, 1974).

Leontiev, describing the transition process as going from something physical or sense-able to a thought or idea, stated that

> "Internalization is that transition by which external processes with external material objects are transformed into processes carried out on the intellectual plane, the plane of consciousness. When this happens, the processes are subjected to a specific transformation: they are generalized, verbalized, abbreviated, and, most importantly, become susceptible of further development that exceeds the possibility of external activity." (Leont'ev, 1974, p. 18).

Davydov et al. (1983, pp. 34-35), and Kuutti (1995), states that human activities are not fixed or constant as they are included in processes in which skill acquisition takes place as a consequence of previous actions and operations, and in a constant reciprocal relation of patterns of internalisation and externalisation, forming new actions and operations (Leontiev, 1978). This is achievable since both internalisation and externalisation, according to Davydov et al. essentially have a corresponding structure (Davydov et al., 1983, pp. 34-35).

"A person's internal activity 'assimilates the experience of humanity' in the form in which it manifests itself in the corresponding external activity (Kuutti, 1995, p. 33) [...] It means that a person's mental processes acquire a structure necessarily linked to socio-historically formed means and modes, which are transmitted to him by other people through teamwork and social intercourse" (Kuutti, 1995, p. 33; Leont'ev, 1974, p. 19).

The relational aspect of the unity of consciousness and activity is a crucial principle within activity theory, and reflected in being asymmetrical, where [conscious] subjects possess *agency*, and having needs, unto which performing activities are necessary in order for the needs to be fulfilled. It is through this interaction that consciousness, the human mind, becomes one with activity, forming meaning.

Externalisation is the typical unique human ability, transforming activities in the internal plane into external activities by the help of tools. It could be the sketching of an idea, jotting down procedures and possibilities, that earlier have been merely a thought, and employing a mediating artefact, in order to manifest or materialise a reflection in the internal plane into something visible or physical. Externalisation could also be a necessary action when conditions change and mental processes in the internal plane need adjustments, or in need of being communicated to others. This, resulting in what, in turn, would provide a transformative feedback, as a result of the operations or actions, is in line with Kuutti's statement about externalisation being the mediated fulfilment of an objective, and a process that reciprocally, transform the individual (Kuutti, 1995). This is also reiterated by Kaptelinin and Nardi, in presenting the internalisation procedure as "a process during which phenomena external to the subject, both physical and social, become both individual and internal", and describe this procedure as a dynamically, bidirectional redistribution of functional elements with the likely transformation of both the internal and the external processes, as the individual learns and develops (Kaptelinin and Nardi, 2012a, pp. 16-17).

As the internalisation process relies on the use of cognitive tools such as signs and language, to comprehend contextual information about a particular activity, the externalisation process employs both physical and psychological tools. In an example of the case study, presented in ch. 5, an expert user or super-user externalised his previously internalised knowledge about a particular activity, like stowing a ship, by initiating or deciding on how the new software (psychological tool) should handle the stowage activity, thus influencing an alteration in the way other employees carry out that activity.

2.4.5 Development

The significant increase in human use of digital technologies, renders human activity increasingly, and exclusively, 'mediated'. The development of skills which is essential in mediated operations is central to the notion of familiarity, and the perception of intuitiveness based on previous experience, stems from learning in a culture where everything is mediated, in an approach related to a computer-mediated communication discourse.

Leaning on Vygotsky (1978), the cultural aspect of human actions, that is, the knowledge that accumulates from being a member of a culture and the human use of cultural artefacts that naturally follows; wherein an almost ubiquitous socio-technical context would include that simple interaction with a computer, utilising a user interface, could be regarded as basic knowledge.

Also, Leontiev links experience and development by situating the source of internal activities by practical activity: "inner mental activity derives from practical activity historically shaped as a result of the formation of human society based on labour and that these processes are formed anew during ontogenetic development in the separate individuals of each new generation." (Leont'ev, 1974, p. 19).

Vygotsky emphasised the importance of using a developmental method to understand human mental functioning, and this applied to mediation in all its forms no less than any other topic. In this connection, he argued that a hallmark of the relationship between sign and behaviour, as well as between word and thought, is that it undergoes a fundamental change (Daniels et al., 2007, p. 186).

Dialectical development

As human use-activities are parts in an overall interacting context, where related activities might change the conditions in such a way that users, during a working process, may experience problems that may cause downtime or a stop in ongoing activities. Within activity theory this is labelled as contradictions i.e. events that manifest themselves as complications and mismatches - breakdowns. This kind of misalignment could be represented by, e.g. not sufficiently trained users or inadequate design of interfaces that fail to support the processes of skilled workers intuitive use. As opposed to the traditional (Western) logic that regards contradictions as indications of problems that have to be solved, activity theory regards contradictions as options for learning; as problems that have to be overcome in a process that is dynamically and dialectically developing; transforming use-activities and tools related to the hierarchical structure, and reciprocally changing cognitive actions on the internal plane, over time (Vygotsky, 1993, pp. 159-160; Leontiev, 1978; Kaptelinin and Nardi, 2006; Ilyenkov, 2009). Davydov, referencing on Ilyenkov, states,

> Il'enkov showed that the ideal is revealed in the human capacity to reproduce or re-create a material object resting on a word, a draft, or a model. The ideal exists in constant intertransitions of the activity elements according to the following scheme: object—action-word-action—object. The ideal is the existence of an object in the phase of its formation, in the subject's activity manifesting itself as a need and a goal. (Davydov, 1999b, p. 50)

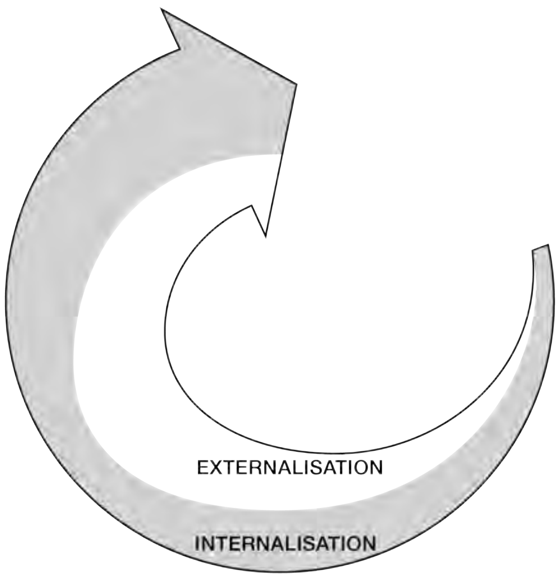


Figure 4. Engeströms expansive cycle of learning, and the relation between internalisation and externalisation during the development process. Engeström (1999a, p. 34).

Engeström (1987, 1999a), introduced the dialectical relation between internalisation and externalisation as a long-term perspective on learning as an expansive cycle of the development of the activity (Engeström, 1999b, p. 64), starting with an emphasis on internalisation, exercising for new skills, or changing to new work methods, e.g. in a context of introducing new software in an organisation, gradually, through experiencing breakdowns, by reflecting on this character of the actions comprising an activity, users would become competent and experts while the activity, with its inherent operations and actions increasingly externalised in the user interface. This form of learning by expanding, by experiencing breaks in a planned sequence of actions and overcoming them, leads to an evolutionary kind of development that through contradictions, breakdowns and workarounds in a context where the borders between action and operations are dynamic and flexible, and the border between actions and activity is blurred (Davydov et al., 1983; Kuutti, 1995), will contribute to the development of both subject and object. Engeström states: "In activity-theoretical terms, activity systems travel through zones of proximal development, […] a terrain of constant ambivalence, struggle, and surprise" (Engeström, 1999b, p. 90).

Engeström (1987) uses the core ideas from Vygotsky (1978), and Leontiev (1978), by re-conceptualising our view on activity - as a collective process, and thereby expanding its mediational model as an activity system, a model that considers both the cooperative and shared character of human activity, by utilising concepts such as division of labour, community and rules, in addition to subject, object and mediating artefacts.

2.5 Functional organ

> ”The hope is that, in not too many years, human brains and computing machines will be coupled together very tightly, and that the resulting partnership will think as no human brain has ever thought and process data in a way not approached by the information-handling machines we know today”.
>
> *J. C. R. Licklider, 1960*[8]

The processes of internalisation and externalisation exist simultaneously and continuously during human activities, and connected by a mediating tool, forming a functional organ intended at achieving or fulfilling an objective (Leontyev, 2009; Kaptelinin, 1995b, p. 51).

The idea of functional organs is a key concept in the context of human-computer mediated activity, since the computer, within activity theory, is considered, as a mediating and converging tool capable of incorporating the representation of a multitude of aspects and elements of the real world, a “special kind of tool” (Kaptelinin, 1995b, p. 49). The capabilities of the computer as a mediating tool allow for the distribution of cognition, and through its thus an ability to represent and simulate other mediating tools. This might also pose a challenge as to how it would allow a user to internalise real world elements using both physical and mental tools such as the computer itself, and representational symbols and language. This context would also constitute the foundation for how humans transform mental representation of an activity into an external activity. Thus, in an HCI-related context a human-computer functional organ would require tool-related competencies, knowing about the tool’s functionality and operating procedures, but also task-related knowledge about the functional limits, i.e. recognising what would be pos-

8. Man-Computer Symbiosis. IRE Transactions on Human Factors in Electronics, HFE-1 4-11. (Licklider, 1960)

sible to achieve, i.e. being dependent on goal and ability to situate the various possibilities that the functional organ provides into the structure of activity, which would indicate that internal processes guide functional organs. According to Kaptelinin, this internal guidance might also be mediated, necessitating the development of new internal skills to master the new tool (Kaptelinin, 1996b, pp. 21-22; Kaptelinin and Nardi, 2006, pp. 64-65).

The activity theoretical concept of *internal plane of action* is defined by Kaptelinin as "[...] a concept developed in activity theory that refers to the human ability to perform manipulations with an internal representation of external objects before starting actions with these objects in reality" (Kaptelinin, 1995b, p. 51), a process termed the *orientation* phase (Leontyev, 2009; Kuutti, 1995). Kaptelinin views the user interface as an extension of the internal plane of action, and by letting the user interface become an extension of the internal plane of actions, it becomes a part of what activity theory describes as a functional organ, introduced by Leontiev (2009), and described as the necessary combination of human capabilities and external artefacts facilitating the accomplishment of goals (Kaptelinin, 1995b). Kaptelinin defines functional organs as “functionally integrated, goal-oriented configurations of internal and external resources. External tools support and complement natural human abilities in building up a more efficient system that can lead to higher accomplishments” (Kaptelinin, 1995b, p. 50).

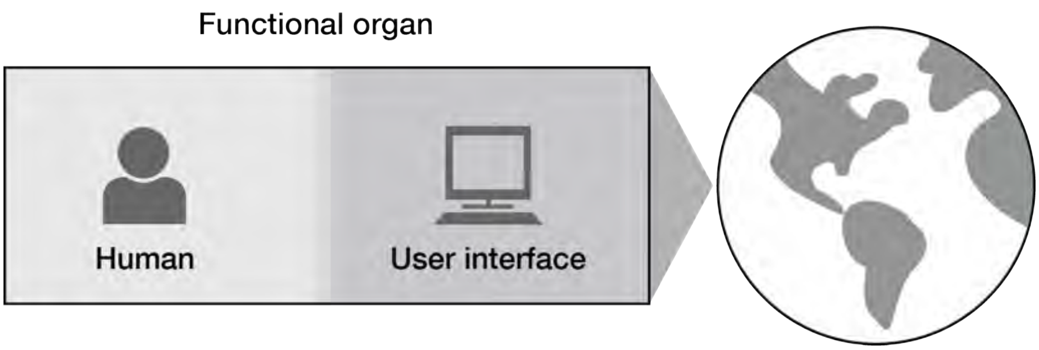


Figure 5. Functional organ. The user interface as an extension of the internal plane of action, forming a human-computer functional organ (Kaptelinin, 1995b, p. 50).

Another dialectical approach to the relation between subject and tool, as a framework for designers building the mediating tools is presented as The Human-Artifact model by Bødker and Klokmose (2011). This model calls for a 'theoretical framework to address the gap between culture, experience, and the practical role of artefacts in embodiment and mediation' (p. 318). The model's notion of reciprocality describes how human operations are inscribed into artefacts that gradually transform both humans and artefacts in a dialectical development process, where skills are acquired, and goals are transformed according to the accumulation of agency, and provides a framework for theorising around "the dialectics between the whole of the use of an artifact in relation to activity, and its parts" (Bødker and Klokmose, 2012, p. 101), focusing on action possibilities related to levels in the activity hierarchy, grounding the analysis from the three perspectives of human activity. The model is might help designers to inform their work, and shorten the distance to theory, by focusing on the questions: *Why?* - as related to the motivation for the activity, *What?* - as related to the goal of an action, and *How?* - as related to the single operations within a use-activity (Bødker and Klokmose, 2012, p. 100). For designers occupied by designing "good UX", the model's focus on the relation between the user's expectations of interaction possibilities, and what the designers' intended possibilities of use, i.e. the expectations among tool-competent users versus the planned use pattern inscribed in the artefact (Bødker and Klokmose, 2012, p. 101).

2.6 Activity theory and human-computer interaction

The 'human factor' in HCI analysis was previously process- and task-oriented (Card et al., 1983), by focusing on people's interaction with computers, achieving goals without considering typical human aspects such as an underlying *purpose* to act, what the task would *mean*, or *reflections* over the potential achievement, i.e. *why* and *how* they interact, and the context within which they act (Kaptelinin, 2015; Kaptelinin, 1992). The cognitive science approach to human-computer interaction, viewing humans as *information processing units* had been the foundation of HCI research. Within the approach of information processing psychology, it would prove difficult to guide the design of usefulness, usability and recently, user experience. The activity theoretical perspective of interactive action extended the cognitive science approach with its quantitative view of humans, as just a set of attributes, and its opposition to the vision of humans as thinking, reflecting and participating actors with needs, and objectives, was discussed by Bannon (1991, pp. 27-29).

The considerations of an activity theoretical approach to HCI grew out of the notion of a need for analytical tools in the direction of a human activity discourse. During the last couple of decades the number of computing devices that affect human activity, whether being in contact with the public through a website, paying bills in an online bank, operating the alarm-setting panel in their home, or being at work doing vocation-specific tasks, has increased considerably. The number of contexts in which humans use screen-based computing devices to mediate both tasks at work and during leisure, show that the number of functional elements in screen-based user interfaces acting as mediating tools that represent possibilities for operations and actions, and in turn activities, has, to a large extent become ubiquitous, in what Kaptelinin refers to as the age of computer-mediated activity (Kaptelinin, 1995b).

Activity theory was, initially, brought into the human-computer interaction discourse by Bødker (1991), and Bannon (1991; Bannon and Bødker, 1989). During the last two decades, activity theory has become a growing and central post-cognitivist approach and concept in HCI research. The main body of activity theoretical literature, related to HCI, has continued with important contributions from e.g. Kuutti (1995), Kaptelinin and Nardi (Kaptelinin, 1995b; Kaptelinin, 1995a; Kaptelinin et al., 2003; Kaptelinin and Nardi, 2012a; Kaptelinin and Nardi, 2006; Kaptelinin and Nardi, 2018), and Bertelsen (Bertelsen and Bødker, 2003) to name but a few.

Bødker states that "a computer application, from the user's perspective, is not something that the user operates on but something that the user operates through on objects or subjects of interest in the work activity. In other words, human beings operate through computer applications, as well as other tools, on materials that they are turning into products with the help of others" (Bødker, 1989, p. 173).

However, there are situated contexts where the user interface ceases to be transparent, e.g. during moments of breakdowns, when the mediating tool becomes visible during our reflections over the character of the breakdown, and adjustments of the user interface. Bødker discusses three possible, basic instances of human-interface relations (Bødker, 1991, p. 38): first, when the objective only resides within the computer artefact. Bødker mentions a spread sheet as an example, that in itself has no direct relation to objects outside the computer, with the argument that a print-out of the spread sheet would not have the same functionality as the spread sheet in the computer. Another example could be a 3D object, that in itself would not have a direct relation to physical objects in the real world other than the visual representation. A print out would, similar to the spread sheet, not have the same capabilities as the computer version. Second, when the objective exists in the real world, but is present exclusively as an activity in the software. Bødker discusses the use of a word processor as an example, where the object is a letter which, at least prior to print out, exists only within the use activity, mediated by the computer, where it can be edited. Another example, closer to this thesis, would be ORCA, the software that has been part of the research. Here the object, a successfully stowed chemical tanker is a physical object, while the use-activity of stowage planning is only present 'on the screen'. The user interface is the mediator between the object of the use-activity and the object in the real world, and that it is *how* the user interface relates to the process of coupling the final object and screen object, i.e. between the object-oriented use-activity on the screen and the physical object which determines the quality of the user interface in this specific constellation. This relation is elaborated in ch. 7.8 (Mediation in ORCA). According to Bødker, the third type of situation, occurs when the object exists, physically, outside the artefact but is controlled by operating the artefact, such as e.g. a control panel (Bødker, 1991, p. 38). An example of this type of situation is the physical stowage process of a chemical tanker as it happens when the ship is at the quay. This is done by the maritime staff on board by manoeuvring physical pumps and pipes. This is done by operating physical artefacts such as control panels with knobs and levers. However, this part of the stowage process is outside the scope of analysis in this thesis, and will not be elaborated further.

Within the discourse of information systems and human-computer interaction research, activity theory is described as a framework that acknowledges several perspectives regarding flexibility and contextual awareness. A theoretical framework that considers whole systems of organisations, work practices, technology and artefacts, but also the cognitive capacity of individuals, with all of their culture, history, experience, understanding and contextual complexity that goes into the tasks we, as human agents, undertake or must relate to, in our quest to fulfil our needs or reach our objectives. Thus, it is well suited as a theoretical lens for analysing the character of activity in a human-computer context as it provides a methodological approach and the tools for understanding patterns in the context of goal-oriented human use of technology (Kaptelinin and Nardi, 2012a). It is a framework that recognises the transformational processes of internalisation and externalisation, and the related concept of mediation (Vygotsky, 1978; Leontiev, 1978), including the character of development that is related to internalisation and externalisation connected to human activity, and in increasing numbers, the process of internalisation and externalisation that is computer mediated (Kaptelinin, 1995b). Activity theory is especially suitable as an analytical tool in a human-artefact interaction context since this relation is, by definition, always mediated, and therefore is particularly suitable for HCI "exploration" (Kaptelinin et al., 1995, p. 190).

Wertsch claimed that human-machine systems appear as primarily an extreme version of the human-environment interaction term, and that HCI is no more than one kind of human-environment interaction, where humans interact with mediated tools rather than other people (Wertsch, 1981, p. ix). However, these reflections were made ahead of the period where activity theory was brought in as a potential theoretical framework for HCI (Bødker, 1991; Kuutti, 1995; Nardi, 1995b; Bannon, 1991; Kaptelinin, 1995b). Nardi, says: "We have recognized that technology use is not a mechanical input-output relation between a person and a machine; a much richer depiction of the user's situation is needed for design and evaluation" (Nardi, 1995a, p. 8). Kaptelinin would argue that due to the special character of the user interface as a mediating tool, humans, rather than interacting with the computer, would interact with the world - and other humans, *through* it, thus describing human-computer interaction as being primarily computer-mediated activity (Kaptelinin, 1995b, pp. 49-50).

In an activity theoretical approach to HCI research, *activity* is the primary theoretical construct, and represents a relational context in which a [human] subject interacts, physically or mentally, with an object, positively existing in the world (Leontiev, 1978). The notion of context in activity theory provides a room for discussing human use of user interfaces based on human activity in a specific context that includes connected aspects, such as a motive, goals, needs and conditions, and also previous experience and the user's ability to undertake situated actions. This relationship is characterised by the subject's needs being met through the subject's interaction with the world, where the subject's activity towards an objective, and world and mutually influence each other (Kaptelinin and Nardi, 2012a).

Focusing of the structure of activity itself, Kuutti explains, "An activity is a form of doing directed to an object, and activities are distinguished from each other according to their objects. Transforming the object into an outcome motivates the existence of an activity. An object can be a material thing, but it can also be less tangible" (Kuutti, 1999), and describes activity theory in a wider perspective as a philosophical framework endowing a multitude of perspectives in studying human agency, whether this agency, both procedural or developmental, as individuals or as members of an organisation or society, underlining the tool approach of mediation (Kuutti, 1995, p. 25).

These are key elements in the discussion about the activity theoretical structure, since the theory's ability to absorb various aspects of human agency in a socio-technical context is formalised through a conceptual framework that must be made strong enough to encompass both humans and technology within the same conceptual models (Kaptelinin, 1995b). The different levels of the hierarchical structure of activity in activity theory allows for evaluating single elements, or categories of elements in the structure based on context or perspective, related to the varying goals of actions, differing associated conditions (Zinchenko, 1981). According to Kaptelinin, this could affect design choices and strategies for skill acquisition, and facilitates an understanding of the computer as a mediating tool strongly influenced by both social and cultural elements, and hence an awareness for technological and contextual differences, both socially and culturally (Kaptelinin, 1992).

Within an activity theoretical HCI/computer-mediated activity framework, the unit of analysis regarding intuitive interaction in user interfaces would be focusing on actions and operations in the interaction context with the user interface, and how skills and expertise are utilised. All human-computer interactions are mediated by the user interface, thus user interface mediation revolves primarily around the contextual relations between human and computer, on operations and actions level.

As a conceptual framework, activity theory is a valued lens in qualitative approaches (Fjeld et al., 2002), as it provides a method for finding and analysing patterns of operations, actions and activities, by its inherent and integrated repository and system of terms and language. Activity theory facilitates an understanding of an activity as a subject's purposeful interaction with an object through the use of a mediated tool, *and*, the subsequent transformation and development that emerges on the various levels of activity, action and operations, from the interaction (Leontiev, 1978).

2.7 Affordances in activity theory

Mediated affordances in the user interface explain how the user interface artefacts are to be used or operated. This proposes the concept of a, per-element, 'user guide' as an inherent part of the user interface, and not solely as possibilities for action that lie in every object.

Kaptelinin and Nardi's concept of handling affordances and effecter affordances, where they split the affordance concept into affordances that facilitate possibilities for handling *with* the technology–namely *handling affordances*, and *effecter* affordances that specifies possibilities for acting *on* the technology, which combined will facilitate the possibility of acting *through*, (Kaptelinin and Nardi, 2012a) shows that functional user interface artefacts ought to be regarded as more than just 'symbolic communication', i.e. as mediated affordances. Mediated affordances are artefacts that afford, not only motivated [inter]action. According to Kaptelinin and Nardi's discussion of handling affordances, mediated affordances also encompass a how-to functionality, through mediated abstractions in a user interface, following their concept of learning affordances (Kaptelinin and Nardi, 2012b), which creates an immediate understanding in performing tasks through 'abstracted tools' in the user interface.

While Bærentsen and Trettvig (2002) focus on a general activity theoretical approach to affordances, the approach of Nardi (1995a), Kaptelinin and Nardi (Kaptelinin and Nardi, 2012b), and Kuutti (Kuutti, 1995) is centred around the various diversities and incompatibilities regarding the use of the affordance concept within the HCI discourse. They do this by pinpointing the main limitation in Gibson's framework in an HCI context, stating that 'it lacks an appropriate conceptual apparatus for understanding technologies as a special type of object, that is, tools mediating human interaction with the environment'.

Affordances are not objects in themselves, but represent, through the visual character of the objects, instructions on how the objects are to be used. Again, a central aspect of user interface elements is *use*. If an affordance represents an instruction for use, it must also be an element in facilitating intuitive use of user interfaces. This must be situated in a goal directed, human activity centred, vocational context. This is also supported by Bødker who states that, 'The user interface cannot be seen independently of the goal or object, or of the other conditions of the use activity' (Bødker, 1991, p. 141). This supports the argument of a task- or activity-based understanding of what constitutes intuitive use of user interface elements.

We might regard all interfaces, analogue and physical as well as digital, as some kind of mediating tool through which people might perform work or communicative activity. A screen-based user interface can, then, be regarded as a framework for mediation; a mediated whole, in which to situate functional elements and the adhering affordances that might be linked to them in order to give the user the possibility, or sense, of immediate understanding.

Leontiev describes this inscription of behaviour as crystallisation, and argues that artefacts are crystallisations of modes, methods and operations of use, and not activity and goals (Leont'ev, 1974, p. 26; Leontiev, 1978, p. 102), bearing a certain resemblance to the notion of 'affordances' (2002; Gaver, 1991; Gibson, 1977; Kaptelinin and Nardi, 2012b; McGrenere and Ho, 2000; Norman, 1999).

Just as our world evolves, the terms we use to describe it should evolve with it. In the networked and digital modernity that most of us live in, screen-based user interfaces are ubiquitous. They have become a natural and almost transparent part of both our private and professional life. Therefore, leaning on, e.g. Kaptelinin and Nardi (2012b), and Gaver's non-static sequential affordances (Gaver, 1991), observing that also screen representations like buttons, sliders, metaphors or abstractions of real-world objects, represent affordances as well. A button affords clicking. A slider affords sliding. Even as elements on a screen, mediated as they are, they afford goal-directed use-activity–actions. They are what Kaptelinin and Nardi coin as mediated affordances (Kaptelinin and Nardi, 2012b). This is also supported by McGrenere and Ho, who argue that a screen element that affords acting upon is 'an affordance that is built into the software' (McGrenere and Ho, 2000, p. 6).

2.8 Chapter Summary

In this chapter I have presented activity theory, its historical roots and central tenets. I have described its basic principles, with the concepts of object orientation, pointing towards the underlying motivation for the activity, and how the process of human activity is divided and distributed in a hierarchy of levels, showing how an activity consists of actions–consciously chosen sub-tasks, which in turn are undertaken by a string of single operations, tasks that could be undertaken unconsciously; mediated by mental and physical tools towards the achievement of a primary goal. Also, the notion of functional organs, that within the context of computer-mediated activity is a key concept, is presented, followed by a review of a core collection of the literature within activity theoretical HCI.

CHAPTER 3.
Intuition

Within the HCI and IxD field, *intuition* or *intuitive use* as terms, have been viewed as professionally and academically problematic, and with a reduced epistemological awareness of its use among practitioners. With some exceptions (Bærentsen, 2000; Hummels et al., 1997), research prior to the early 2000s, was scarce, even as researchers became increasingly aware of the rising numbers of computers and the expanding areas into which computing powers played a crucial part.

There are some rather paradoxical aspects of the notion of intuition in that it has played such an important role in human thought and reflections about the character of knowledge throughout history. Already Plato and Aristotle describe intuition as knowledge residing in 'the eye of the soul' (omma tēs psuchēs) (Heidegger, 1988, p. 109), and Plato, in 'The Republic', states that intuition is a foundational ability of the human mind to grasp the inner, pure, character of reality (Plato, 1969).

In Descartes's writing about intuition, the meaning of intuition is referred to as a source of truth or something that is apparently true (Fishbein, 1987) that define the view of intuition as a form of intellectual perception towards the modern era.

It appears to be that there are more eclectic views than proper consensus among the researchers in the field, of what intuition is (Hoffman, 1992) in (Shanteau, 1992), although it might seem, that an approach where it is regarded as based on experience, learned skills, and expertise, rather than hunches, gut feeling, and magical unlearned 'knowledge by heart', is claiming precedence; however, with a large number of definitions.

Following Bastick (1982), who presents intuition as 'universal experience' it is not, perhaps, unnatural that the term intuition is being used in several contexts, which renders it rather ambiguous, or even contradictory of what researchers in the field possibly would recognise or define as intuitive.

3.1 Some etymological elements

Etymologically, the term intuition stems from around mid-1500. In the Merriam-Webster Dictionary, it is stated that the word stems from Middle English, *intuycyon*, (denoting spiritual insight or immediate spiritual communication): *'from Late Latin, intuitiō a contemplation, from Latin intuērī to gaze upon, from tuērī to look at'.*[9]

An excerpt from the collection of current synonyms for intuition, collected from Roget's online thesaurus (3 ed.), thesaurus.com, lists some of the terms: 'hunch, instinct, feeling, perception, premonition, gut reaction, innate knowledge'.[10] Mostly related to the lay people's perception of the term, being 'just knowing' without proof, or prior skill or expertise. Adding to this understanding of the word, the terms of knowledge, reasoning and reason are described as antonyms.

However, in the current literature on intuition in an HCI context, intuitive use is empirically shown to be closely related to experience and familiarity (Blackler and

---

9. intuition. Dictionary.com. Collins English Dictionary - Complete & Unabridged 10th Edition. HarperCollins Publishers. http://dictionary.reference.com/browse/intuition (accessed: October 17, 2014).
10. http://www.thesaurus.com/browse/intuition?s=t. | Accessed: October 17, 2014.

Popovic, 2015; Blackler et al., 2011; Hurtienne et al., 2008; Mohs et al., 2006a; Raskin, 1994). A selection of current synonyms for experience could be: 'skill, knowledge, training, maturity, practice, understanding, wisdom, familiarity'[11] to name but a few.

In most of the current research on intuitive use of user interfaces, there is little connection with what, according to Roget's Thesaurus, is perceived as intuitive, i.e. the meaning that has gradually inhabited the intuition term, and the content of what the researchers in the field, according to the literature, regard as the correct interpretation of 'intuition', namely experience, leading to an 'etymological' disconnection between researchers and regular individuals.

Bunge states that there are few words as ambiguous as intuition (Bunge, 1962), and claims that the unqualified use of the term intuition was delusive to such an extent that an exclusion from the dictionary was proposed, and that the reason for not going through with such a grave procedure was probably because of its impracticality, precisely because of the word's assimilation into the everyday speech to such an extent that one only had been forced to introduce a host of new concepts to replace it. Bunge (1962) describes the duality that the term of intuition originally comprised, namely the philosophical discourse or context and the scientific discourse and context, which, in the philosophical discourse, has evolved into what has become its meaning in the modern vernacular: immediate knowledge without qualification as a faculty of the human mind, an understanding of the term intuition that is elusively reflected in current dictionaries, like e.g. Oxford English Dictionary which defines intuition as 'The ability to understand something instinctively, without the need for conscious reasoning.'[12] while within the [non-philosophical] science discourse another approach has been developed, one where intuition may be

11. http://www.thesaurus.com/browse/experience?s=t | Accessed: October 17, 2014.
12. https://en.oxforddictionaries.com/definition/intuition. Accessed. October 17, 2014

the result of meticulously accumulated, and tested, knowledge (Bastick, 1982; Bunge, 1962).

Encyclopedia Britannica states that

> '***intuition***, in philosophy, is "the power of obtaining knowledge that cannot be acquired either by inference or observation, by reason or experience". As such, intuition is thought of as an original, independent source of knowledge, since it is designed to account for just those kinds of knowledge that other sources do not provide. [...] Two further technical senses of intuition maybe briefly mentioned. One, deriving from Immanuel Kant, is that in which it is understood as referring to the source of all knowledge of matters of fact not based on, or capable of being supported by, observation. The other is the sense attached to the word by Benedict Spinoza and by Henri Bergson, in which it refers to a supposedly concrete knowledge of the world as an interconnected whole, as contrasted with the piecemeal, "abstract" knowledge obtained by science and observation.[13]'

The current daily speech of intuition as knowledge without previous reason and experience, has been regarded as the 'magical' sense of knowing 'by heart' rather than mind, it has been a source of many discussions within philosophy and the philosophy of science.

3.2 Intuition: pre-HCI research

Richman, Gobet, Staszewski, and Simon (1996, p. 180) claim that 'intuition is synonymous with the familiar process of recognition', while Herbert Simon conceded that intuition would also be similar to *pattern* recognition. In de Groot's well-known chess player case study (de Groot, 1978), it was explained that chess players' approach to the game of playing chess was based on their ability to recognise previ-

13. Encyclopedia Britannica, fifteenth edition, 1998. Micropaedia vol. 6, pp. 360-361.

ous positions and how these positions were played; better players would be capable of recognising more patterns than inexperienced players, thus being able to play moves and positions quick and without having to analyse every step (Simon, 1955). Simon's approach to familiarity as pattern recognition, which he called chunks, was, however, criticised by Dreyfus and Dreyfus (1986, pp. 33-34) for being assessed as single 'chunk' patterns, defined independently, and detached from the totality of comprehensive positions and the sequentiality of related moves which, according to Dreyfus and Dreyfus, would pose a problem since most chess positions would comprise more than one chunk, as several possible moves and patterns would lay inherent in a position.

There are two distinct directions in the multifaceted realm of literature on intuition and expertise within the psychology and cognitive science discourse. Meehl's comparative meta-study on forecasts made by clinical psychologists compared to predictions based on statistical models (Kahneman and Klein, 2009, p. 517; Meehl, 1954), inspired by the 'heuristics and biases'-approach (HB) by Tversky and Kahneman, where they claim that intuitive decisions are based on pseudo-algorithmic logic (heuristics) and subsequent cognitive bias (Kahneman and Tversky, 1982; Tversky and Kahneman, 1974). The HB approach is less favourable towards expert judgements. The HB community is more focused on people making mistakes, and the cognitive limitations of man. Instead they are quantitatively focusing on performance based on rules and formal models. From the HB perspective, it is expected that decisions based on forms and algorithms will outdo experts making informal ad hoc decisions based on experiential patterns.

This perspective is contrasted by the 'naturalistic decision-making'-approach (NDM) by Klein (1999) which emerged from the work by de Groot on chess master's intuitive decision-making (de Groot, 1978). This work presents intuitive action as the result of experienced-based decisions made under changing ad hoc circumstances thus becoming a flow of actions and reactions in a dynamic real-time setting. Concurring with Simon, he presents pattern recognition as an explanation to how people can make decisions seemingly without a prior rational analysis (Klein, 2003). Klein's approach to intuition is based on human judgement and expertise at work; naturalistic decision-making, in which he describes the process of how users construct a what-if simulation, similar to the internalisation process we find in activity theory, in order to explain how something might happen, as 'simulation heuristics', but based on experience and 'natural decision-making' (Klein, 1999). Klein states that this also provides a base and structure for solving similar, but not entirely familiar, tasks (Klein, 1999).

Additionally, these two different approaches differ in the view on *research contexts:* field settings vs. laboratory settings for decision-making and performance standards, where the experiment approach preferred by the HB community would endorse the laboratory as a research context, whereas the overall, qualitative methods approach preferred by the NDM community, such as e.g. real world observations, would endorse field studies (Kahneman and Klein, 2009).

Also Bowers et al. (1990), leaning on Dreyfus and Dreyfus (1986) state that their 'model of intuition implies the role of memory and experience in judgement and problem solving', suggesting that clues and hints in a user interface could activate knowledge structures in memory, guiding the user towards an implicit solution that would be recognised as a preliminary perception of coherence, a hunch or preceding notion, that would otherwise, e.g. without the mediated screen elements and functional trigger elements, not have been recognised. Bowers et al. state that this must not be misinterpreted as what they coin as 'indivisible and unmediated apprehension of an elemental truth' (Bowers et al., 1990, p. 74), a view on intuition that ordinary people, as we will see, look upon as instinctive, natural or unlearned (Cappon, 1994).

Instead, the approach by Bowers et al. presents intuition as having two stages; first a *guiding stage* – the implicit perception of coherence, directing thoughts, through activating a network of recollection processes towards an explicit understanding within a context, and; second, the *integrative stage* that involves merging this representation of coherence into the consciousness, across what Bowers et al. call a 'threshold of awareness', and state that this transition, from the guiding to the integrative state, is often perceived as becoming knowledgeable in an almost non-rational, self-validating way, thus considering intuition as informed judgement in the context of discovery (Bowers et al., 1990, pp. 73-75).

Most of the literature, ontologically and epistemologically related to this thesis, is focused on intuition as an individual experience, and presents intuition as 'arriving at decisions or conclusions without explicit or conscious processes of reasoned thinking' (Gregory and Zangwill, 1987, p. 389). Kahneman and Tversky details the use of terms 'intuition' and 'intuitive' connected to activity into three different areas of meaning; divided on (1) knowledge is called intuitive if it is reached by an informal and unstructured mode of reasoning, without the use of analytic methods or deliberate calculation, (2) a rule or a fact of nature is called intuitive if it is compatible with our 'civilian, non-expert' model of the world and (3), a rule or a procedure can be said to be a part of our repertoire of intuitions when we apply the rule or follow the procedure in our normal conduct (Kahneman and Tversky, 1982, pp. 124-125). Here, there are different contextual meanings for the use of the concepts intuition or intuitive interaction, where the main distinction is activity/non-activity. While Kahneman and Tversky, in their rather quantitative approach to expert behaviour, are not particularly concerned about relating expertise to intuitive behaviour, Klein claims that intuitive action is based on experience and expertise.

3.3 Early works on graphical user interfaces

Intuitiveness in user interfaces has been something users have been craving for, and designers strived to develop almost since the invention of the graphical user interface; from Engelbart's work on the oN Line System (NLS) and the augmentation project (Engelbart, 1962), Ivan Sutherland's seminal work on Sketchpad (Sutherland, 1963), Alan Kay's early work on FLEX and Smalltalk, and later when the Xerox Alto, and subsequently Xerox Star was developed and built by Xerox PARC (Kay, 2001). When this emancipatory approach to human-computer interaction was later commercialised by Apple, first with the Apple Lisa and later the Macintosh (Levy, 1995), the demand from the public, for easy-to-use computing technology increased, and has continued to do so. The public's access to computing power surged by the change of the human-computer user interface, from the command line to the visual point and click WIMP paradigm, i.e. the graphical interface. In Scandinavia, the emerging vision of the democratisation of computing power, especially with the introduction of new technologies in the workplace, lead to the participatory design strand within Information Systems research, where workers as users, who were to use the technology, should also be involved in developing it.

3.4 The novice-expert continuum

To resolve the characteristics of what constitutes an expert, Shanteau (1992) states that the decision of who is an expert, should be turned over to their peers within the domain; and defines an expert as: 'Experts are operationally defined as those who have been recognized within their profession as having the necessary skills and abilities to perform at the highest level' (Shanteau, 1992, p. 255), thus ensuring that the vocational competence is categorised and situated correctly in the skill acquisition continuum.

The novice user is identified by a strict following of rules, with a rather rudimentary situational perception, and without the capability to relate to situational adjustments. The competent user is able to handle tension and complexity, and is capable of evaluating actions as part of, or appropriate, within a larger context while adhering to a set of standards or routines regarding processes. The expert practitioner, no longer relies on regulations and guidelines, and maintains an intuitive approach to work processes based on experience and tacit knowledge, only employing an analytic approach in extraordinary or problematic situations (Dreyfus and Dreyfus, 1986).

Popovic (2000) explores, using [think aloud] protocol analysis, the varying competences between beginners and experts by relating the interaction with technical devices to cognitive and ontological categories in addition to categories of how knowledge is represented. Popovic (2000, 2003) describes a set of features connected to the models of, and the transition process from experts to novices:

- <u>General Knowledge (GK),</u> knowledge level of a novice; emerging from general learning and experience which appears to be a set of general skills acquired through generic choices and actions, to which something all regular people, or users, will be able to relate; without any specific education or experience. It is, however, required in order to acquire what Popovic calls domain-specific knowledge and become an expert.

- Domain Specific Knowledge (DSK), specific, vocational knowledge in a particular field. A novice, typically needs, and learns from, externalised, knowledge in order to do tasks and develop into an expert. This category of, specifically learnt, knowledge is important for a persistent successful solving of task assignments, and thus for the process of transforming from novice to expert.

- Task Experience and Expertise (TE), experience which ensures that an expert in the domain uses his knowledge in a more efficient manner than novices [Kolodner (1983) in Popovic (2000)] and would analyse potential problems at a deep association level [Chi et al.(2009) in Popovic(2000)], thus being able to better assess the relevant procedures for performing a task.

- Task Execution (TEX), which is described as a task or sub-task completion outcome.

- Transition (T), the transformation moment, when users make changes between their thoughts, or protocol categories or knowledge categories. It occurs in the particular problem space in which the processing takes place, thus acting as a processor, where the development from novice to expert emerges.

The transformation from being a novice and a level of general knowledge, to becoming an expert, occurs by gaining domain-specific knowledge, and task experience and expertise:

Novice	Transition	Expert
General Knowledge	> > >	General Knowledge + Domain-Specific Knowledge + Task Experience and Expertise

Table 1: Knowledge levels. Features of the knowledge levels for novices and experts. Excerpted from (Popovic, 2000, p. 935).

At various points during this transition process from being a novice to becoming an expert, knowledge changes from being recently gained and new to becoming previous knowledge and experience, thus becoming a part of the domain-specific knowledge base (Popovic, 2000; Popovic, 2003; Simon, 1955).

Herbert Simon, in comparing the performance of highly skilled individuals to the performance of novices, claims that domain-specific knowledge is adamant in order to build expertise, and that this knowledge is built and resides within a knowledge repository (Simon, 1955), upon from which an individual can draw. Glaser states that, 'The performances of highly competent individuals indicate the possession of, rapid access to, and efficient utilization of an organized body of conceptual and procedural knowledge; [...] a major component of expertise is the possession of this knowledge' (Glaser, 1987, p. 82).

By also pinpointing the knowledge *source*, Mohs et al. (2006a) describe the sources in a novice-expert continuum going from: innate, sense experience, cultural, expertise, to the use of tools; recognising the closer we approach the tool plane, the more domain-specific the knowledge will become, as shown in figure 7, p. 72.

Dreyfus and Dreyfus' framework was developed as an argumentative element in the discussions of the limits of artificial intelligence and was not constructed as a general learning model as such. This framework was based on 'the dynamic processes of human skill acquisition' (Dreyfus and Dreyfus, 1986, p. 19). Their skill acquisition matrix, shown in table 2, describing the various levels of competence that comprise the process of going from novice to expert.

Skill level	Components	Perspective	Decision	Commitment
1. Novice	Context-free	None	Analytical	Detached
2. Advanced beginner	Context-free and situational	None	Analytical	Detached
3. Competent	Context-free and situational	Chosen	Analytical	Detached under-standing and deciding. Involved in outcome
4. Proficient	Context-free and situational	Experienced	Analytical	Involved under-standing. Detached deciding.
5. Expert	Context-free and situational	Experienced	Intuitive	Involved

Figure 6. The Five Stages of Skill Acquisition framework by Dreyfus and Dreyfus (1986, p. 50).

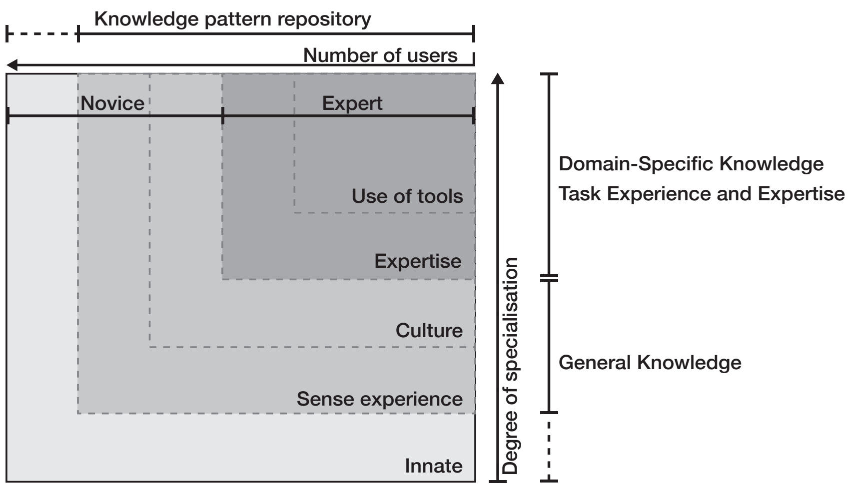


Figure 7. Expert-novice continuum. Levels of prior knowledge in the expert-novice continuum derived from Mohs et al. (2006a, p. 131), adapted by including the feature categories by Popovic (2000), shown in table 1, p. 71, adding Simon's notions of knowledge pattern repository (Simon, 1955), not taking into account that various conditions and degrees of specialisation in Mohs et al.'s 'continuum' might overlap in the transition process from novice to expert.

A resulting difference among practitioners according to the level of experience, described by e.g. Popovic (2000)(2000), Mohs et al.(2006a) and Shanteau (1992), is exemplified by e.g. Swann's study of airport security screeners (2014), where findings showed that experienced screeners, experts, interacted with interface functionality more efficient. They solved occurring problems more effectively than the screeners without the same level of knowledge who, with the help of functional user interface screen elements, would employ a trial-and-error approach that would result in inefficient problem solving and breakdown in the interaction with the screening system.

Dreyfus and Dreyfus (1986) characterise the *novice* as a user strictly following externalised information such as rules and regulations, with a limited perception of the operational possibilities in a context, and with an additional weak awareness of the need for situational adjustments. The *competent* user is capable of coping with complexity and pressure, and is also able to assess operations and actions as part of an overall solving of a task, or as suitable or appropriate within a larger conceptual context, while employing a set of procedural standards or routines. The *expert*, however, is no longer dependent on rules and guidelines, and can rest in the security inherent in having acquired skills and tacit knowledge; what Popovic calls domain-specific expertise (Popovic, 2000), and knowledge to use tools in an expert-like approach in situated circumstances, only employing an analytic approach in extraordinary or problematic situations. According to Dreyfus and Dreyfus, traversing the levels in the skill acquisition matrix, starting from level 1 reaching level 5, involves an accumulation of knowledge, thus, leading to expert-like skills being described as *accumulated experience* (Dreyfus and Dreyfus, 1986). This paradoxically provides the necessary platform from which to reflect, since Dreyfus and Dreyfus claim that experts employing intuitive action cannot really explain why, and sometimes not even how, they do what they do.

3.5 Research on intuitive interaction after 2000s

Intuitive or *intuitive use* is a widely used, perhaps overused, term in the field of human-computer interaction; in relation to most socio-technical artefacts involving human use in a human-computer interaction context, be it either regular software, information systems, web-based systems that have some socio-technological connection or relevance, or tangible user interfaces. In daily life, where we often employ this term as 'quick and easy understanding', or 'immediate apprehension or cognition'. In everyday life, then, *intuition* is encompassed by the sense of 'just knowing' without a rationale or previous experience or knowledge, as in *instinctively*, i.e. some kind of non-rational gut feeling; e.g. something along the lines of the definition of 'a natural ability or power that makes it possible to know something without any proof or evidence, a feeling that guides a person to act a certain way without fully understanding why, and something that is known or understood without proof or evidence', found in e.g. Merriam-Webster, see ch. 3.1.

The body of scientific literature on intuitiveness within the HCI discourse has, up until the early 2000s, been rather scarce. This can be explained due to the term's ambiguity and vagueness (Israel et al., 2009; Naumann et al., 2007).

Prior to the current research initiatives, Raskin (1994), stated that the term 'intuitive interface' in a layperson's understanding, was being perceived as normal, human, and being 'intuitive' would be enough for any user to use a technological appliance, i.e. understanding something without any previous experience and rational thought. Raskin claimed, however, that interfaces being *perceived* as intuitive, depends on the recognition of the user interface elements as familiar. He does, however, state that it is the experience that leads to something being recognised as an opportunity to act, which is the basis for an intuitive approach to activity (Raskin, 1994).

In much of the early literature that describes intuitive perception of interfaces, the term is, to a great extent, linked to people's cognitive understanding in a setting of human-technology relations. (Barnard, 1987; Card et al., 1983; Green et al., 1996; Newell and Card, 1985; Norman, 1987; Payne, 2003). Also, In 'Subconscious and Conscious Systems of Cognition', Norman relates what we regard as typical features of an intuitive approach to problem solving, e.g. speed, automation and multiple sources of knowledge, to the subconscious, while the task solution, where actions are characterised by or based on reflection, emerges as sticky, rule-based and with few or limited knowledge sources (Norman, 2013, p. 49).

In the research undertaken post 2000s, however, Raskin's familiarity perspective (Raskin, 1994) is presented as technological familiarity, and characterised as the primary aspect of intuitive interaction (Blackler and Popovic, 2015; Blackler et al., 2011; Hurtienne et al., 2008; Mohs et al., 2006a; Naumann et al., 2008); focusing on intuitive use, Blackler states that 'using familiar labels and icons and possibly positions for buttons help people to use a product quickly and intuitively the first time they encounter it' (Blackler et al., 2010, p. 1). Here, the term intuition and intuitive use approach comprising physicality, embodiment or tangible user interfaces, and presents intuition as a matter of experience of activity (Blackler et al., 2005; Dreyfus and Dreyfus, 1986; Hurtienne et al., 2008; Raskin, 1994).

The present and complementary definitions, on intuitive use within the field of HCI (Blackler and Popovic, 2015; Blackler and Hurtienne, 2007; Mohs et al., 2006a; Naumann et al., 2007), rest on the recognition of familiarity in a technological perspective, based on recollection, i.e. knowledge and experience gained by use of prior technology. Leaning on e.g. Simon (1955) and Klein (1999), this would also include the recognition or recollection of a *similar* technology.

Blackler et al. (2005) present three principles of familiarity, in order for designers to develop user interfaces that are intuitive to use: *firstly*, to use familiar symbols/words in expected positions for functions that are the same or similar features that the users already know. *Secondly*, metaphors that represent something that is already known, should be linked to new functionality in the process of creating familiarity with something that is similar but not identical. *Finally*, knowledge and metaphorical content and meaning should be coherent in all parts of an interface (Blackler et al., 2005).

The Intuitive Use of User Interfaces (IUUI) group at TU Berlin in Germany has devised a definition of intuitive use based on a literature review of usability design criteria [Scholz (2006) in Blackler and Popovic (2015)] and a series of interviews and workshops with users, usability specialists and user interface design practitioners, where intuitive would describe a *relation* between an individual and an artefact:

> A technical system is intuitively usable if it leads to effective interaction by unconscious application of prior knowledge by the user. (Mohs et al., 2006a, p. 130)[14]

This is also discussed by Israel et al., (2009), where intuition would be related to activity rather than describing the attribute of an artefact. In Israel et al., it is emphasised on mental efficiency by a discussion of mental focus on problem solving, focusing on how attention shifts from the ’interface-oriented’ to becoming ’task-oriented’ Mohs et al.(2007), in Israel et al.(2009, p. 351).

Also, Naumann et al. focus on *intuitive use* rather than the user interface itself being intuitive, by discussing physicality and repeated tangible interaction diffuse into, in

14. Original text: Ein technisches System ist intuitiv benutzbar, wenn es durch nicht bewusste Anwendung von Vorwissen durch den Benutzer zu effektiver Interaktion führt.

addition to discussing whether, or possibly how, intuitive use relates to the visual part of the user interface design (Naumann et al., 2008).

The overall view from recent and current research on intuitive use of user interfaces (Blackler and Popovic, 2015; Blackler et al., 2011; Hurtienne et al., 2008; Mohs et al., 2006a; Naumann et al., 2008), indicates that we might find grounds for intuitive use through *recognition of metaphors* which, in the goal of human interactivity with user interfaces in a domain-specific context, i.e. skilled worker environment, and based on what a human intends to do is laying the foundation for intuitive perception of functionality, rather than a precise cognitive understanding of an artefact.

The thesis aims to situate intuitive use of user interfaces within the activity theoretical approach to HCI, in addition to gain a better understanding of the rather diverse and vague concept of intuition or intuitiveness within the HCI field, by situating it within the activity theoretical HCI discourse.

3.6 Intuitive interaction in user interfaces

The terms intuitive, intuition and intuitiveness have been described as vague in the literature (Naumann et al., 2007). Instead, their connection to everyday language has made it a buzzword related to user-friendliness and usability, where 'use' should be self-explanatory, or at least 'usable' in an 'easy to learn, easy to use, easy to remember, easy to correct mistakes' manner; an immediate-use version of the user interface; the visual, graphical, point and click, and graspable one. (ISO, 1998; Nielsen, 1993; Shneiderman et al., 2009).

In working towards a theory of intuitive interaction, the QUT (Queensland University of Technology) research group's definition is based on a meta-study comprising a literature review into intuitiveness and diverse fields related to intuitive interaction in a socio-technical context, including HCI, cognitive psychology, usability and interaction design.

Intuitive use of products involves utilising knowledge gained through other experience(s). Therefore, products that people use intuitively are those with features they have encountered before. Intuitive interaction is fast and generally non-conscious, so people may be unable to explain how they made decisions during intuitive interaction (Blackler and Popovic, 2015).

According to Blackler et al., intuitive use of products involves utilising knowledge gained through similar experience(s). Therefore, products that people use intuitively are those with features they have encountered before. Therefore, Blackler et al. argue, intuitive interaction is fast and non-reflective; people may be unable to explain how they made decisions during intuitive interaction (Blackler, 2008; Blackler et al., 2002, 2003).

3.7 Intuition and HCI

Human-computer interaction (HCI) is the study of how to design the patterns of how we mediate our interaction with the specific tool of computers. As such, the field of HCI has grown into a rather wide and diverse discipline in the intersection of computer science, ergonomics, design, and social and behavioural science (Rogers, 2012). The amount and level of usage that screen-based media have reached in current work life and societal practice, where a large and increasing part of human activity, are conducted through computer-mediated activity, shows that screen-based user interface artefacts or other computer-related interfaces, increasingly, are being regarded as ‘real’ tools for action. This has led to many approaches to defining what HCI is, as it includes a vast array of interfaces, from, e.g. tangible, gestural, brain-computer, and screen-based GUI, in both embodied interaction patterns, and interaction with a screen-based user interface, covering most aspects of human life.

Until the early ’90s, there was no firm agreement of a single, unified theory of HCI (Hewett, 1992, p. 5); rather a set of underlying principles and best practices that comprised the underpinnings for an understanding of human-computer interaction, based on cognitive psychology as a multidisciplinary field of both work and research (Löwgren and Stolterman, 2007). The ACM Special Interest Group on Computer-Human Interaction (SIGCHI) developed a definition of HCI:

> ‘Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use in a social context, and with the study of major phenomena surrounding them.’
>
> [The Acm Sigchi Curricula for Human-Computer Interaction (Hewett, 1992, p. 5)].

In classic HCI theory in the early ’80s, utilising a model based on human limitations and capabilities, like e.g. perception, attention span, memory (Card et al., 1983), within the perspective of cognitive science the theorisations saw the individual as a human information processing unit, consisting of three connected systems: perceptual, cognitive and motor, with their individual memory and processor

(Rogers, 2012, p. 25), as a design guidance for user interfaces. In acknowledging the limitation of the cognitive approach to HCI, the theoretical framework of activity theory is brought into the discourse by Bødker (1991) in what Kaptelinin and Nardi call 'Post-cognitivist HCI: second-wave theories' (Kaptelinin and Nardi, 2003), as a tool for analysing actual use, including participation in design, the user interface as a mediator, and expertise as a whole; where the individual is, as Bannon states, not a factor, but an actor, with needs and objectives.

3.7.1 Recognition of metaphors

Metaphors are based on objects and actions from the real, physical world, and are utilised as semantic representations in the user interface to guide users in understanding, and thus using 'computing machinery'. The direct manipulation of semantically abstracted objects in the user interface has been discussed by e.g. Hutchins et al.: "The systems that best exemplify direct manipulation all give the qualitative feeling that one is directly engaged with control of the objects – not with the programs, not with the computer, but with the semantic objects of our goals and intentions" (Hutchins et al., 1985, p. 318)

The use of metaphors representational screen elements is, however, not entirely uncontroversial. A well-known example of how human-computer interaction experiences is added to this pattern repository is the early metaphorical Macintosh waste bin icon which represented an underlying, and somewhat ambiguous, functionality that could both *delete* a document or a folder, and *eject* a floppy disk. Many early Mac-users used logical inference and were afraid and apprehensive of ejecting the floppy by dragging it to the waste bin, believing that its content would be erased. Later, that anomaly was no longer a problem. After being clarified that the floppy would just be *ejected* and not *erased*, the users were no longer afraid of accidentally erasing floppies. Today, of course, we no longer use floppies, and the recollection of this anecdote of early computer trivia fills us with a blend of mild despair and chuckling indulgence.

Figure 8. Early use of metaphors on the web: The Southwest Airlines homepage, 1996.
(Source: The Internet Archive, [http://web.archive.org/web/19961106122431/http://iflyswa.com/], accessed: Oct. 15. 2016.)

Early in the web era, the reliance on metaphors eventually took to extreme directions. Nearly all physical artefacts were reproduced in the user interface, and represented by screen elements, such as this early site for Southwest Airlines in 1996. An understanding of this early approach to the use of metaphors on the web could be explained by a statement by Ericsson 'A metaphor is an invisible web of terms and associations that underlies the way we speak and think about a concept' (Erickson, 1990) in (Herstad, 2007).

By conceptualising the user interface with use of metaphors, we tap into a repository of recognised artefacts; a repository that will evolve with use. Svanæs describes the accumulated use of metaphors as:

> We conceptualize the computer through metaphors (e.g. information systems, hyper media, communication medium), and externalize this understanding in the conceptual model underlying the systems software (e.g. desktop metaphor, World-Wide-Web, e-mail). We thus 'freeze' a certain un-

> derstanding of the nature of the computer, and this understanding is reinforced every time a new piece of software is created within one of the existing structures. (Svanæs, 2000, p. 3)

According to Lakoff and Johnson (1980), metaphors, as concepts, are not only a matter of linguistic significance, but a way of internally, and visually, guide our comprehension, when interacting with the real, outer world, by letting these 'ontological icons' represent events, activities and states.

Acknowledging how the abstraction level, i.e. the use of representational metaphors and icons, in graphical user interfaces have become more internalised and implicit among users, and also more dense and information packed as they have matured over time. This implies that user interfaces have acquired a character of an increased complexity, which is somewhat of a paradox since GUIs initially were meant to pack complexity in a 'box' of comprehension.

Systems have, over time, grown large and complex, with a similarly complex user interface and have, in spite of the GUI, become increasingly difficult to learn and cumbersome to use. A steep learning curve and a certain degree of awkwardness will eventually lead to a proneness to make mistakes and a possible difficulty to correct them. An ideal planned user experience, using software of any kind, should be quite the opposite. It should be difficult to make mistakes, and if mistakes are made, they should be easy to correct (Nielsen, 1993; Shneiderman et al., 2009). These are key elements in the fields of human-computer interaction and interaction design, and central when working with usability and user experience on web development projects.

The number of computing devices and the contexts in which they are used, adding to the amount of action and work processes conducted through screen-based user interfaces, have increased considerably during the last decades. This, in turn, has made focusing on the users, and the contexts of use becoming increasingly important in modern software development.

Since the use of technology has increased considerably in society in general, due to the invention and development of graphical user interfaces, screen artefacts and metaphors have become more internalised and matured over time. This implies that they have also become part of a common, established, understanding, a sort of cognitive 'black boxing', of representational screen abstractions, their meaning and subsequent functionality. Although not a part of the theoretical framing for this thesis, the concept of black-boxing socio-technical complexity (Latour, 1999) is still useful as a depiction of the abstraction and mediation of tools comprising the user interface in a human-computer interaction context, as it would contribute in our understanding of complexity as an impacting element in how intuitive use of user interfaces might be perceived.

Gibson's original concept of affordances is primarily about framing the direct perception and action between animals and their environment. This perceived notion of action possibilities includes three basic features; an affordance relates to the capabilities of an actor but is independent of the actor's capacity of perceiving it, i.e. experience and culture, and, finally, it does not evolve as means and objectives of the actor change (Gibson, 1979).

Don Norman brought the concept of affordances into the HCI discourse (Norman, 1988). Norman differs from Gibson in the definition of affordances; while Gibson describes affordances as independent of a user's capacity to utilise them, while Norman claims that the *perceived* and physical/real character of an affordance (Norman, 1988), determines the usability of a technical system by stating that an affordance is a relationship between an individual and the object, displaying what possible actions it affords the individual to take upon it (Norman, 1998, p. 123).

The amount and level of usage that screen-based media have reached in current societal practice, where a large and increasing part of human activity, being either private, societal or as a part of ordinary work processes are conducted through screen-based media, shows that screen-based user interface artefacts are being internalised. This is further developed by Gaver's notion of affordances encompassing screen elements in user interfaces, as he sees affordances as 'properties of the environment relevant for action systems'. By exploring Gibson's bidirectional 'able-ness' (Gibson, 1986, p. 128), and original concepts of affordances for the design and evaluation of user interfaces, Gaver defines affordances as: ' … properties of the world that are compatible with and relevant for people's interaction. When affordances are perceptible, they offer a link between perception and action; hidden and false affordances lead to mistakes' (Gaver, 1991, pp. 79-80).

He states that "People perceive the environment directly in terms of its potential for action, without significant intermediate stages involving memory or inferences. For instance, we perceive stairways in terms of their 'climbability' " (Gaver, 1991, p. 79). This is possible because of the availability of attributes. In a user interface setting this is transferable to UI elements, like buttons, sliders, unequivocally designed hyperlinks and the like, where the potential for action is unambiguous.

3.8 Chapter Summary

In this chapter I have presented some problematic elements of the term 'intuition' or 'intuitive use', including a description of its etymological roots and current definition, followed by a presentation of parts of the discussion within the research community around intuitive interaction, and the connection of 'intuitive' to graphical user interfaces, leaning on the current research (post year 2000) on intuitive interaction. The character of, and the categorisations within, an expert-novice continuum, including the Skill Acquisition framework have been discussed. The chapter is concluded with a brief discussion of metaphors as representations in user interfaces.

CHAPTER 4.
Method

This chapter describes the research approach, research context and strategy, and research method applicable to this thesis.

This thesis reports from a case study of the development and use of a software system as a foundation for discussing the elements of intuitive interaction through a user interface in a skilled worker environment. The empirical material was gathered during the three phases of design, development, and later implementation and use in the period from 2012–2016.

4.1 Selection of case

The ORCA project at Odfjell Chemical Tankers in Bergen, Norway, was chosen as the case because the project represented a distinctive development approach and a defined goal for the end users, that would coincide well with the theoretical approach and defined research goal of the thesis: 'what constitutes intuitive interaction in user interfaces?' The ORCA system is a corporate information system used by human operators to govern the planning and management of stowing and transporting chemical liquids on ships, as the main pattern of use. As such, the environment in which the software would be put to use, could be regarded as a skilled worker environment, where the operators would undertake specific and repeating sets of tasks, and where one group of software operators within the shipping company had previous, extensive field operative experience within their vocational area, as ship officers. A specific experience with hands-on processes that might, or at least ideally, should have an impact on the degree of intuitiveness with which they could operate the system.

The case chosen for the study, presents an example of intuitive interaction of user interfaces. This dissertation will try to explain why interaction with ORCA is intuitive by going into details about the human-ORCA interaction in real use, and how this interaction was designed.

The development of the new stowage planning software in the company was done applying a best practice approach. It is supposedly more challenging to explain why something works really well, rather than finding an explanation to a problem, since the underlying causes for something that works well might be several – and related.

There are many examples of technology that works well; websites that give the users a sense of immediate understanding of how to interact; services that are available and accessible to all. It is even increasingly expected that regular users, through rather complex technological information infrastructures, should be able to solve tasks that previously had to be undertaken by skilled personnel, such as net banks, travel sites, hotel bookings. The activity needed to achieve one's goals has been transferred from a kind of human agency by proxy to a generalised human agency by technology, in a socio-technical context where everybody, ideally, should be able to participate. On the other hand, there has been a significant process-related shift in the workplace. An increasing amount of work tasks have been digitised, demanding an added set of skills from the workforce – the skill of interacting with computing machinery, in-house technology that is aimed at a small group, but still fail to provide systems that support the task-competence of the users.

Therefore, in order to grasp the fabric of best-case causality, it is necessary to search widely in the sequence of use-activities that together comprise the task domain, in order to understand the character of a specific best practice, and how, in both content and order, it can be designed.

4.2 Research approach

Yin (2008) states that a case study might employ both qualitative and quantitative methods for collecting data, and the empirical material in this dissertation consists of nonparticipant observation, interviews, archive material and survey results.

The research strategy has been carried out as a longitudinal case study (Yin, 2008), where designers and users of a newly developed stowage planning software for chemical tankers were followed.

The research method has been slightly mixed, but has *primarily* consisted of interviews with and observations of operators on land. Interviews with and observations of personnel at sea were not possible as time, and geographical reasons, prevented a stay on board from one harbour to another. Therefore, data from the personnel at sea were collected through a simple survey. Survey data does not provide the same depth of information as being present, observing actual use-activity. It is only the users' own descriptions of their own impression of how they experience the interaction with the user interface in ORCA, post factum. This is, however, an experienced user group, consisting of mainly captains and first mates, who have the necessary skills, i.e. tool- and task-related competencies, to assess the opportunities and limitations inherent in the mediating tool. Nevertheless, there is, admittedly, such a significant degree of uncertainty, and possible heuristic bias connected to the pattern arising from the answers, however distinct, that it would be premature to conclude from these findings with anything but uncertainty.

Interpretivism and *positivism* are two mutually excluding approaches to the understanding of what knowledge is. *Positivism*, in one end of the epistemological spectrum, is solely based on empirical data being observable, objective and quantifiable i.e. countable. Positivism proclaims that empirical data is the only source of true and verified knowledge and rejects the cognitive value of interpretation. Data collection methods are primarily quantitative; measuring, counting and statistical analysis of the empirical material.

The *interpretive* approach in the other end of the spectrum represents how we *perceive* the elements and aspects of what we are studying in order to provide an understanding of the *reciprocal* process of user, system and context (Walsham, 1993; Klein and Myers, 1999). This implies that interpretive researchers regard that our knowledge of reality is obtained exclusively through social constructions such as language and communication, consciousness and reflection, documents, mediating tools, and other cultural artefacts to which humans have given a purpose and a name, thus advocating the notion that reality, our perception of it, is purely subjective, and reciprocally so. Walsham states that 'Interpretative methods of research start from the position that our knowledge of reality, including the domain of human action, is a societal construction by human actors and that this applies equally to researchers'. (Walsham, 1993, p. 5)

Although the research topic revolves around issues that to a large degree are contextual, and is influenced by the participants' previous knowledge and experience which, necessarily, are based on and influenced by social and cultural construction and comprehension, i.e. an interpretive approach, some firm evidence that linked them together was required. In this thesis, that is the conception that something purely ontological, the mediated functional elements in the user interface, exist, and that these elements will be perceived differently by different participants, based on the participants' background and experience. In addition, the data collection was a bit challenging, to the extent that an additional survey was undertaken in order to collect some complementary data. A (non-positivistic) mixed method approach was therefore chosen. A simple analysis was made on the questionnaire answers and is presented in ch. 8.6.7.

4.2.1 Research strategy

There are several strategies, in addition to a case study that can form the basis for doing research, e.g. surveys, experiments and archival analysis. Yin presents a case study as an empirical inquiry; a longitudinal and thorough investigation of a limited number of events that, 1. investigates contemporary activities in depth and within its real-life, everyday context, when the boundaries between the phenomenon of the study, and context are not necessarily distinctly evident. 2. relies of multiple sources of evidence, of which Yin identifies six: Documentation, archive records, interviews, observations, participant observations, and physical artefacts (Yin, 2008, pp. 101-102). Yin states that a case study might employ both qualitative and quantitative methods for collecting data, and the empirical material in this thesis consists of nonparticipant observation, interviews, archive material and a survey.

Observation sessions , 45 - 60 min.		7
Interviews, 45 - 90 min		
	Developers	5
	Company	7
	Total	12
Archival material:		
	Company Quarterly magazine, mid-2011 - mid-2013	8
	Project documentation and meeting minutes*	23
Survey	No. of respondents	26

Table 2: Empirical material, categories overview.

In this case, the inquiry has been undertaken by following the development and implementation phases of software in a shipping company, which extends from the end 2011, until the summer of 2013. After the implementation period, data about use among operators were collected. The empirical data comprises observations of the cooperative process during the development period, i.e. the communication between members of the development team, project owners and participant super-users. In addition to observations, interviews with members of the core team of developers, management, and super-users were undertaken. Also, some archival material has been provided from the company and core team, such as the company quarterly magazine, and reports and meeting minutes from the development phase.

During the development phase, I have been present at the company headquarters in a back-and-forth manner, from early in the project, early 2012 and onwards, observing and interviewing SCRUM teams, project management, participant [super users] and also the regular operators, after the system was up and running.

During the subsequent implementation period, from spring 2012 to the summer of 2013, I have observed user participants reflect on needs and activities, transform tasks into actions, and perform operations in the vocational context that the system was released into, and meant to mediate. In late 2015, the empirical material was complemented with follow-up interviews and observation sessions of operators, who then had had the opportunity to work with the system for almost two years. Also, survey data regarding interface features and task process support, i.e. the mediation, in the finished software through a survey among captains and other ship officers at sea was collected.

4.2.2 Research context

During this longitudinal study, I have followed the progress of a system development process from the beginning, following SCRUM teams, project management, participants and super users. Regular users have been observed during the development period from early 2012 until well after the system was released. This long-term access has given me the possibility to follow a development process with a goal that closely matches my overall motivation for doing research within the field of HCI: computers should support people in their everyday tasks in a way that is easy to learn and easy to use.

The empirical data have been collected on location at the Odfjell Tankers' headquarters in Bergen, among cargo brokers in their daily work, and among visiting naval personnel during training sessions in ORCA. It was conducted in three stages, first during the approximately 15 months development period from January 2012, second, during the implementation period of the software, which lasted from March to August 2013, and finally, additional gathering of empirical material in the period October 2015 – February 2016. The empirical material consists of field notes, video recordings, audio recordings, survey data and archive material from the company. The survey was conducted as a web questionnaire in order to be independent of physical presence.

The empirical material may seem narrow, caused by the fact that the access to the field area was not continuously available, thus not being ‘hands on’ when the action happened, since the stowage planning process occurs when clients contractually agree and order transportation of cargo. This activity, with all its sub-actions and sub-operations happens at a typical regular irregularity, i.e. randomly and anytime, and have made the fieldwork difficult to plan in any detail. User interaction simulations were thus, eventually, required in order to gather observational data about the various aspects and elements in the operator-system interaction. In return, this became very effective and compact sessions, revealing numerous interactive operations in a controlled but realistic scenario, instead of waiting for another random cargo stowage process to happen, the operator could go straight into a typical everyday stowage process. The simulations were based on previous, real stowage plans.

The number of operators who could be released to such session duty was also limited. Therefore the observation data and interview data were supplemented with a survey, directed towards the officers on all naval vessels. The questionnaire was linked directly to how the personnel perceived the interaction with the different parts of the system and its user interface.

4.3 Research methods

Data collection has been carried out in a primarily qualitative approach as a case study (Yin, 2008). It consists of semi-structured interviews with stakeholders, developers, project management, participant users and regular users. Also, field observations of regular users during the testing period and after implementation has been undertaken, recognising the fact that different perspectives including users of varying experience and competence, technical artefacts and corresponding user perception and acceptance are needed in order to adequately cover the research area.

Due to observations of user activities being difficult to plan because of the random character of when they are done, in addition to the observation field itself, the Odfjell offices in Bergen, is located on the other side of the country, an additional small survey was conducted.

Non-participant observations have been targeted towards developer teams, SCRUM sessions, communication between developers and project management, communication between developers and participants and regular users. Also, observations of users' interaction with the system, i.e. regular work, where participants carried out actual tasks, in both real and demonstration contexts. The observation sessions varied in length, from 30 minutes to three hours depending on the amount of work tasks.

Semi-structured interviews have lasted for a period of 45 min-1,5 hours. A digital recorder was used. Targeted observation sessions of regular users lasted for approximately 1,5–2 hours, and involved observing the employee's work tasks. Un-targeted observations, observation by looking around, have been continuous while I have been present in the field.

Data has also been gathered through a survey directed towards personnel on board the vessels, centred around the participants' perception of process support for regular work tasks, and intuitive interaction through the system's user interface.

Some data related to the development phase has been gathered from archived documents, such as sprint reports, meeting minutes, project reports and the company's quarterly magazine and the original vision scope document.

The software studied is part of a substantial infrastructure that, to a large extent is influenced by a heterogeneous array of decisions on a higher, more abstract level. This opened up for a broad set of methods, including interviews with both developers and users, observation of users at work, maintaining their daily operations, in addition to a survey that was mainly based on quantitative methods but with added extended qualitative possibilities for comments. The mixed methods approach opens up for a broad set of methods, including interviews with developers, users and managerial staff, observation of users working, maintaining their daily operations, in addition to surveys based on quantitative methods.

It consisted of individual observation sessions of a group of five users working from the company headquarters on shore, with different levels of experience and also varying background expertise, as some of them had previous experience as captains and senior officers on the ships. I have conducted interviews with six members of the core team, which lasted between 45 minutes and 1,5 hours. Two of the team members, the main super-user and the project owner/leader were interviewed twice. The interviews were conducted as semi-structured conversations, where the respondents, in addition to talk about their career in the company, and relation to technology in general, were asked to talk extensively about their work processes. A digital recorder was used. Un-targeted observations have been a continuous process while being present in the field. Additional follow-up observation sessions were conducted, and additional survey data was gathered in early 2016.

4.3.1 Survey

Data collection was influenced by the fact that observations could not be planned, in any distinct fashion, in advance. This, combined with the access to the observation premises depended on air transport, suggested that the qualitative material had slightly less size than an ideal. Aiming to collect additional data that could contextualise the material, a simple survey was conducted in February 2016 among naval personnel. The survey was based on the QUESI questionnaire (Questionnaire for the subjective consequences of intuitive use). The QUESI questionnaire was developed by the German research group at the Technische Universität Berlin in

2009 (Hurtienne and Naumann, 2010; Hurtienne et al., 2009). The questionnaire builds on the IUUI description of intuitive use as the user's instinctive use of prior knowledge leading to effective and unhindered interaction (Mohs et al., 2006a; Naumann et al., 2007), presented in the literature review ch. 3. This survey model has been tested towards a variety of interactive systems. i.e. websites, navigation systems and mobile phones, with more than 500 users evaluated (Hurtienne et al., 2009, p. 2). The QUESI form consists of 14 questions, divided into five subgroups, see table 3, p. 98.

The response scale is a symmetric five-point Likert agreement scale, with a balanced label interval: 1=Fully disagree, 2=Disagree, 3=Neutral, 4=Agree, 5=Fully agree.

The questionnaire was distributed as an interactive form in pdf format by email to 39 vessels. Twenty-six officers returned answered pdf forms.

Sub-groups	Question no.	Characteristics
W - Perceived mental workload	1, 6, 11	1. I can use the program without thinking about it 6. The program is not complicated to use 11. I barely must concentrate on using the program
G - Perceived achievement of Goals	2, 7, 12	2. I achieve what I need by using the program. 7. I am able to achieve my goals in the way I expect. 12. The program helped me recognise what I had to do to completely achieve my goals.
L - Perceived effort of Learning	3, 8, 13	3. The way the program works is immediately clear to me. 8. The program was easy to use from the start. 13. It is clear to me straight away how the program is used.
F - Familiarity	4, 9, 14	4. I can interact with the program in a way that seems familiar to me. 9. It is always clear to me what I have to do to use the program. 14. I automatically do the right thing to achieve my goals.
E - Perceived Error rate	5, 10	5. I can easily fix problems if they should occur. 10. The process of using the program goes smoothly.

Table 3: Question numbers distributed on subgroups. The answer scale ranged from 1:'Fully disagree' to 5: 'Fully agree'. See appendix 2, part 3.

Sub-groups	Question no.	Characteristics
W - Perceived mental Workload	15, 20, 25	15. The ORCA interface uses the same 'images' as I am used to in the regular stowing process, therefore I can use the program without thinking about it. 20. Since ORCA works the way we usually do when stowing the ship, the program is not complicated to use. 25. Since the program operation resembles regular stowage task ow, I must barely concentrate when using ORCA
G - Perceived achievement of Goals	16, 21, 26	16. Because of my experience with regular stowage task flow I achieve what I need by using the program. 21. The ORCA interface uses the same 'images' as I am used to in the regular stowing process, therefore I am able to achieve my goals in the way I expect. 26. Because of my experience I could detect it when the program helped me recognise what I had to do to completely achieve my goals.
L - Perceived effort of Learning	17, 22, 27	17. ORCA works the way we usually do when stowing the ship, therefore it is immediately clear to me the way the program works. 22. Since ORCA works the way we usually do when stowing the ship, the program was easy to use from the start. 27. Because of my experience with regular stowage task flow, it is clear to me straight away how the program is used.

F - Familiarity	18, 23, 28	18. Since the program operation resembles regular stowage task flow, I can interact with the program in a way that seems familiar to me. 23. Since the program operation resembles regular stowage task ow, it is always clear to me what I have to do to use the program. 28. The ORCA interface uses much of the same 'images' as I am used to in the regular stowing process, therefore I automatically do the right thing to achieve my goals.
E - Perceived Error rate	19, 24	19. If problems occur, I can easily fix them, in the same way I fix problems that occur during a regular stowing process. 24. Because of my experience with regular stowage task ow, the process of using ORCA goes smoothly.

Table 4: Distribution onto sub-groups. The questions and corresponding numbers distributed on sub-groups in table 5 extended with aspects regarding the understanding of the user interface connected to previous experience and recollection of stowage operations. The answer scale ranged from 1:'Fully disagree' to 5: 'Fully agree'. See appendix 2, part 4.

The QUESI survey was adapted and employed to measure intuitive interaction in a skilled worker environment in relation to the use of one specific software. The first group of questions (questions 1–14 in the survey) were directed towards the perception of intuitive interaction in general. An additional group of questions was added, extending the number of questions to 28 (questions 15–28 in the survey). The added group of questions was analogue to the first, but requiring reflections from the participants on their vocational experience when answering (see appendices 2–6 for survey details). All questions follow the phrasing pattern of higher scores representing an increased possibility of intuitive use.

The questionnaire also contained an introductory part regarding the users' age, experience and education in addition to current position, prior to the questions directly aimed at aspects regarding the participant's use of the ORCA program through the program's user interface, see appendix 2, parts 1 and 2.

4.3.2 Data collection

The initial data were collected during the design phase and consisted of field notes from scrum retrospective meetings and presentations of sprint deliverables with developer team and company representatives present. In addition to interviews with the core development team, the project manager, a representative from the company management, the main super user and another very experienced user, and the head of the development team, and one developer from the developer team, see table 2, p. 92.

Role in project	**Interview topics/keywords**
Project manager	Project vision, software needs, development process, perceived cooperation, user participation
Management rep.	Software needs, previous software history
Main super user	Stowage process, previous software history, software needs, development process, perceived cooperation, user participation,
Very experienced user	Stowage process, user experience
Head of developer team	Development process, user participation, perceived cooperation, co-location aspects
Developer	Perceived cooperation, perceived co-operation, co-location aspects

Table 5: Overview —interviews with core team members and related topics.

The collection of observational data proved to be challenging in the sense that it was rather difficult to plan ahead when to observe regular stowage planning at the cargo broker floor. Requests and orders for transportation of cargo are normally received and accepted on an infrequent basis; thus it became rather unpredictable when a stowage planning session would be available for observation. This combined with the need for air travel in order to be present in the field led to a less than ideal setting for the gathering of the empirical material.

The solution to this dilemma of arbitrary access was to plan field observations at the same time as tutorial classes were scheduled and then schedule simulation sessions in the same period. The observation sessions of operators had a non-participant character, however, the researcher, and the participant were, occasionally, communicating in an inquiry- and interview-like manner during the sessions. These sessions, which simulated everyday work tasks, had the character of being both observation and interviews, as the operators acted dialectically in the observation session, by talking about what they were doing, and commenting their reasoning while performing operations, following the contextual inquiry approach presented by Holzblatt et al. (1993), an observation technique where a researcher is together with the participant in the users' regular environment, being in the background as in a regular overt non-participative observation manner, but where the researcher may ask clarifying questions. In these sessions, in order to collect all the data produced in these information-dense sessions, a video-camera was being used, primarily for preservation of data beyond what was possible to do with field notes alone.

Other data collection sources were archival material like sprint reports and retrospectives, and meeting minutes from meetings in the core group, in addition to scope documents and company quarterly magazines.

Lastly, a survey was conducted, both in order to reach a user group that was otherwise difficult to reach: the officers at the vessels, and provide a set of data extending the data from the observation sessions.

4.3.3 The participants

This following section presents the people that have been involved, describing, briefly, their background and part in the development process, in addition to the users of the software. What characterises this group of people is an *initial* heterogeneity, naturally, due to the difference in background, education, and project approach ranging from developers, project owners, expert users, and regular operators – gradually transforming into a homogeneous group.

Management representative

A member of the governing group for the ORCA development process. He was involved in the development of the previous system, as a project manager. He has participated as an advisor, and as a member of the steering committee. Interviewed twice; once, early in the project, and once after the system had been in use for about a year.

Management representative and a member of the governing group for the ORCA development process. He is the current project owner's immediate superior, and was involved as a project manager in the development of the previous system, which will replace ORCA. He has not been actively involved in the development of ORCA, but participated as an advisor, and as a member of the steering committee, thus helping to make decisions related to system capacity and functionality constraints,[15] but without being directly involved in the development process. Inter-

15. Hard and soft rules, determining which rules are allowed to 'bend', and which are not - and <u>how</u> to 'bend' the soft, bendable ones.

viewed twice; once, early in the project, and once after the system had been in use for about a year.

Project owner. Involved in the project from start, the middle of 2011. Project manager for the development of the new stowage system, ORCA. Interviewed three times; once, early in the project, once towards the end of the development period, and finally after the system had been in use for about a year.

Background within the Odfjell system: project management – eleven years. Involved in the project from mid-2011. Project Manager for the development of the new stowage system, ORCA, and related to the developers, he is the project owner, in SCRUM terminology. Went on to head the project and training department after the project was finished, and the software implemented on all ships and in company headquarters.

Super-user. The main super-user and most experienced employee, who acted as a premise provider in terms of which functionality that should be included in the software, and as a participating user representative. His role in the project has been what one in actor-network theory would have called an Obligatory Passage Point; the one person that all involved participants, users, developers, scrum master and project leader/owner alike, would go to with wishes, ideas and suggestions for how the new software should function and be operated. Educated as Captain; Twenty-five years of naval experience, 11 of them as Chief Officer or Captain at Odfjell Tankers. The super-user has also worked as an operator and having had the role as a super-user at the company headquarters for the last 12 years. Interviewed three times; once, early in the project, once towards the end of the development period, and finally after the system had been in use for about a year.

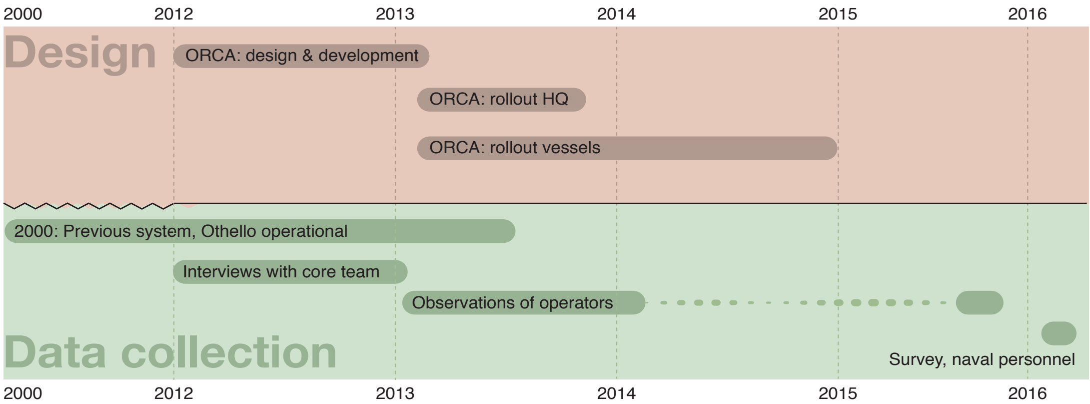
Operators. On land, the users of the new system will be operators in Bergen, and Dubai offices, in addition to the company's port operators in major ports like Houston and Rotterdam. At sea, it will be the captains and chief officers aboard the ships.

Development team. *Head of the Development team:* programmer and SCRUM master in the project. Interviewed twice; once, early in the project, and once towards the end. *Developer* and co-located member of the SCRUM team. Interviewed once, midway through the development process.

4.4 Chapter summary

In this chapter, the research approach, –strategy and –methods have been presented. The empirical material has been categorised and the research context, of non-participant observation and semi-structured interviews in the user location, in an addition to a small web-based survey directed towards a group of naval personnel at sea, have been presented. Also, this chapter has described the challenge connected to the data collection process, and how it was solved.

CHAPTER 5.
Case - design and use of stowage planning software in a shipping company



In this chapter the case study is presented. It describes the empirical case study undertaken in one of the world's largest shipping companies within the field of sea transport of chemical liquids, where the top-down *initiated,* but user/expert *driven* development of a vocational information system for stowage planning has been followed. The development process was characterised with continuous and extensive, bottom-up, communication between developers and users, with developers not only co-located in the same building but also located on the same floor as the users. The structure of this chapter is presented as follows: first, the company and participants are introduced, and subsequent motivation and goals for developing the new system.

5.1 The company

Odfjell Tankers is a global organisation, headquartered in Bergen, Norway. It is represented through overseas offices in 12 countries as of October 2016. In its field, the company is regarded as a world-leading company in the global market for ocean transportation, and storage, of industrial liquids, i.e. chemicals, oils and other products. The company owns and operates a large fleet of chemical tankers, both globally and regionally in addition to a network of tank terminals. Their customers range from small businesses with single and occasional shipments to large multinational businesses with large and diverse requirements for seaborne transportation.

The company's major trade streams cover the United States, Europe, Asia, India, the Middle East and South America.

> '[…] the company's main business thus includes transporting and storing organic and inorganic bulk liquid chemicals, acids, animal fats, edible oils, drinkable alcohols and clean petroleum products, and the company also operates four modern distillation coloumns. Some of the cargoes represent hazards as they may be flammable, toxic or corrosive. Examples of cargo would be alcohols, aromatics (benzene, toluene, xylene), Hydrocarbon solvents, Naphtha, Gasoline components, Gasoil and fuel oil components, Petroleum distillates, and so on.' (Odfjell Annual Report 2012).

The fleet of 74 chemical tankers;[16] owned, time chartered, commercially managed, and additionally managed on a pool basis, ranges in carrying capacities from 4,000 to around 75,000 DWT, and of a variety of tank configuration and coating. Most of the ships are equipped with zinc and epoxy coated cargo tanks unless made of stainless steel. The main reasons for coatings are easier cleaning and less risk for contamination of the cargo. However, tanks made of stainless steel are needed for the cargo that would chemically react with the coating. The smallest ships are oper-

16. As of November 2016.

ating the so-called regional trade in the Far East, and are operated from the Singapore office. The main tanker fleet, the large ships, are operated from Bergen. In their newest office, in Dubai, has product tankers carrying chemicals of a lighter variety, in addition to the main transport oil products, e.g. oil tankers that can also carry chemical class. The total capacity of the current fleet is around 2.75 million DWT, and transports 18.5 million tons of cargo to and from almost 500 different ports worldwide, with almost 5000 port calls. The process of loading and discharging more than 600 different types of products can be extremely complex. (Odfjell Quarterly March 2012)

For eleven years the company have based their daily stowage operations on one specific application, Othello that, although sophisticated, contained features that have rarely been utilised, and in principle just had made the software more complex and slow to use. The software was intended for both vessels and on land, but was never implemented on the vessels, which in turn added even more logistic complexity with managing the stowage processes. The company decided to develop a new stowage tool, where great emphasis was given to the user:

> ‘The new stowage tool will allow the operator and shipboard management to work on the same stowage and use the same computer software program. In order for the new system to be successful, high priority will be given to the user.»
> *- Odfjell Quarterly Magazine, December 2011*

5.2 Terminology and daily practice

In the following, the basic terminology is briefly presented but is described more elaborate in relation to the presentation and discussion of regular use in ch. 7.

Stowage plan.

The stowage plan is managed by the software. A stowage plan contains the following information:

- A list of calls, with voyage information (also called Rotations).
- A list of operations (mainly cargo-operations), happening in the calls from the Rotations.
- A tank plan and the links between operations and tanks in the tank plan.
- Additional ship information used to calculate ship utilisation and trim/list, or to run the COF checks.

By 'publishing' the plan the operator makes it available for online co-workers with access to the vessel, in addition to the naval personnel on the vessel, which will receive it by email. After stowing the cargoes, the captain of the vessel will promote the plan into a Masterplan, and send it back to the operator. (see figure 9, p. 115)

Rotation

The Rotation list is a list of sorted calls. In ORCA two levels of calls are managed: Port (level 1, top level) and Berth (level 2, sorting under a port). Each port can be assigned to a voyage.

In the following, a brief explanation of the field's terminology is presented.

Rotation and rotation list are parts of the terminology, but is presented more elaborate in relation to the presentation and discussion of regular use in ch. 7.

Operations

A cargo-operation in the ORCA user interface can be:

- A cargo-operation, such as loading, unloading, transferring or in transit. Transit operations are used to represent cargo-operations without other activities in a call, and are only displayed on the port level.

- An operation not related directly to the cargo, i.e. a port activity or a blocked tank.

Operation is a vocational term related to a specific set of use-activity in the ORCA user interface, as described above. 'Operation' is also the activity theoretical term for the bottom level of use-activity in the hierarchical structure of an activity in Activity Theory. An 'operation' in ORCA can be both an operation and an action in an activity theoretical sense. Thus, for the sake of clarity, the vocational term of *operations* is 'rebranded' as *cargo-operations* throughout the text of this thesis. However, the user interface elements related to cargo-operations in ORCA will still use the term *operations* as it is part of an established terminology, and therefore the user interface.

Operation list

Displays a list of the cargo that the company has agreed to transport, but not yet placed on board a vessel.

Tank plan

The tank plan in ORCA represents all the cargo tanks and their contents based on the selected calls. The size and position of the tanks on the screen are abstractions of the current ship, and not exact, pixel-perfect, renditions of the real tank plan. However, the information displayed in the tank plan can be regarded as accurate.

Violations

Violations are generated by the application when an incorrect situation has been detected. Violations are classified into 2 main categories:

<u>Soft rule violations.</u> These are warnings aimed at the Operator or the Captain, often requiring verification, but not preventing the stowage.

Hard rule violations. There are either errors in the stowage plan, or incompatibilities that need to be removed to have a valid stowage plan.

Cargo list:

Displays a list of cargo already on board.

Trim & List:

A ship's trim & list data describes the ship's position and stability in the water. It is important that the software take these parameters into account in order to precisely control the liquid cargoes on board the vessel.

Cargo Manifest

The Cargo Manifest is a report showing information about all loads, their origins and destination for a specific trip, including all IMO classifications and US Coast Guard classifications. The cargo manifest is a standard form, intended for customs personnel.

5.3 Stowage workflow: the process of stowing a chemical tanker

The handling of chemical cargo is a complicated subject. The products that a ship can carry are strongly regulated by national and international legislation, and in addition the different charterer organisations have their demands as to how their cargoes shall be handled. Adding the fact that chemicals have their own physical properties that influence the handling, the complexity increases.

The stowage planning of chemical tankers is a process by which the stowage locations of cargoes on board a vessel has to be meticulously planned prior to loading in order to reduce the costs as much as possible while at the same time exploiting the vessel's capacity. The requirements of the stowage planning process are becoming more challenging and complex with the development of new chemicals and ships with an increased number of tanks, in addition to rigorous regulations.

There are several aspects related to the process of stowing a ship other than vast numbers and complexity itself. One of them is quality, i.e. commercial quality in the sense that if done in a sloppy manner, the company will lose money on underutilised ships. Another aspect has to do with security quality. If cargoes were stowed in a manner that would make it possible for them to react with each other, the worst-case scenario would be exploding ship tanks. The 28th of February 2004, the chemical tanker Bow Mariner exploded and sank just 45 miles off the coast of Virginia. Twenty-one crew members died, while six were saved.[17] On June 4th the same year, another chemical tanker, M/T NCC Mekka, exploded while the ship was at anchor in Rio de Janeiro.[18] Two crew members died. On July 26th the chemical tanker Bunga Alpinia exploded off the coast of Malaysia.[19] Five crew members died. On the 7th of March 2013 the chemical tanker Harbour Krystal suffered an explosion in one of her tanks, of the coast of Lisbon.[20] One crew member died. The list of catastrophes could continue. Therefore it is of vital importance with software systems that would aid the users in the process of stowing the ships in a safe manner.

A chemical tanker differs from an oil tanker in almost all aspects, except for the fact that they both transport liquids. An oil tanker transports one type of cargo in five or six huge tanks, from the point of extraction to a refinery.

A chemical tanker, on the other hand, is significantly smaller, and transports liquid chemical materials in a multitude of smaller, specialised tanks, between various port

17. http://www.cerc.usgs.gov/orda_docs/Assets/UploadedFiles /CaseDocuments/V A_Bow-Mariner_USCG-Investigation_12-14-2005.pdf. Accessed: Sept. 5, 2013.
18. http://www.c4tx.org/ctx/job/cdb/precis.php5?key=2004040 4_001. Accessed: Sept. 5, 2013.
19. http://officerofthewatch.com/2012/07/27/tanker-bunga- alpinia-inferno/ Accessed: Sept. 5, 2013.
20. http://gcaptain.com/crewmember-missing-blast-board/ Accessed: Sept. 5, 2013.

terminals, in a multi-point to multi-point pattern called a *rotation*. Another, perhaps more immediate issue to handle when transporting various loads of chemicals is compatibility; how the chemicals react, either with specific tank coatings or each other. The system has to support the operator to avoid putting the wrong load in the wrong tank. For example, if an operator puts an acidic chemical in a tank with a coating, the coating will be 'burned' off. Then, a stainless tank, or a tank with a coating that could withstand acid must be used. Likewise, there are rules for the filling of adjacent tanks–as to which chemical liquid that may be transported alongside each other. If a mistake is made in this situation, the ship will be withheld in the port by the authorities. And the reason that the two loads are not allowed to be transported side by side with only one bulkhead in between is that if there is, or arises, a crack in the tank wall, there might be a chemical reaction that ultimately could destroy the ship. This is an operational practice associated with a significant degree of complexity, and sometimes critical complexity. Therefore, a chemical tanker must adhere to international regulations that require chemical tankers to follow the International Bulk Chemical Code (IBC Code),[21] regarding transport of hazardous liquids at sea like e.g. SOLAS ch.VII and MARPOL Annex II.[22] The examples at beginning of this chapter are all actual, real-world events, caused by the failure to adhere to these rules.

21. International Convention for the Safety of Life at Sea (SOLAS). Chapter VII - Carriage of dangerous goods
22. International Convention for the Prevention of Marine Pollution from Ships (MARPOL). Annex II - Regulations for the Control of Pollution by Noxious Liquid Substances in Bulk

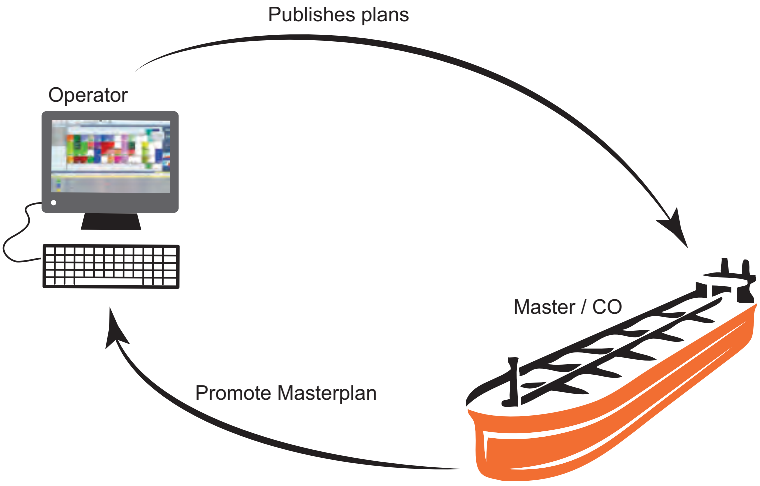


Figure 9. *Practice*: the workflow of making and publishing stowage plans.

Step 1: The Operator synchronises the stowage plan. This will import new rotations, with calls and cargoes, and update the existing ones.

Step 2: The Operator verifies the data in the stowage plan and adjusts it if required.

Step 3: The Operator publishes the stowage plan.

Step 4: The Captain receives the published stowage plan and stows the cargoes.

Step 5: The Captain promotes (or publishes) the stowage plan.

Back to step 1.

The operators on land are extensively involved in various tasks related to vessel performance, including stowage, coordination of port calls, transshipments, bunkering and bunkering purchases, supervision of tank cleaning, performance monitoring, weather routing, freight collection and demurrage management, and supervising the adherence to rules and regulations related to cargo and cargo handling.

Role	Tasks
Ship Operator	Primary user of stowage tool ashore
Vessel, shipboard	Primary user of stowage tool onboard
Ship Management-internal	Responsible for vessels, crew, ICT systems installed onboard
Ship Management-external	Responsible for T/C vessels, crew, ICT systems installed onboard

ICT operations	Responsible for operations and support of ICT tools used within the company
OT Shipping project	Responsible for new OT application portfolio, incl. onboard solution

Table 6: The six elements or roles active in the process of stowing a chemical tanker.

Viewing stowage planning as an activity system, includes the subject that consists of those who have direct contact with the software, i.e. the operator on land, and the captain and first officer/mate on board the ship who works towards the object which is the most optimal position of the load at any time. This is complicated by the fact that what is regarded as an optimal stowage might be changed by each port arrival during a rotation. This activity is mediated by the ORCA software. As an activity system, it would also be composed of a set of rules, the community that consists of everybody that is related to the context, such as maritime personnel, colleagues on shore–including support personnel, customers, in addition all the roles that are directly involved the stowage process as a division of labour (fig. 10, p. 116).

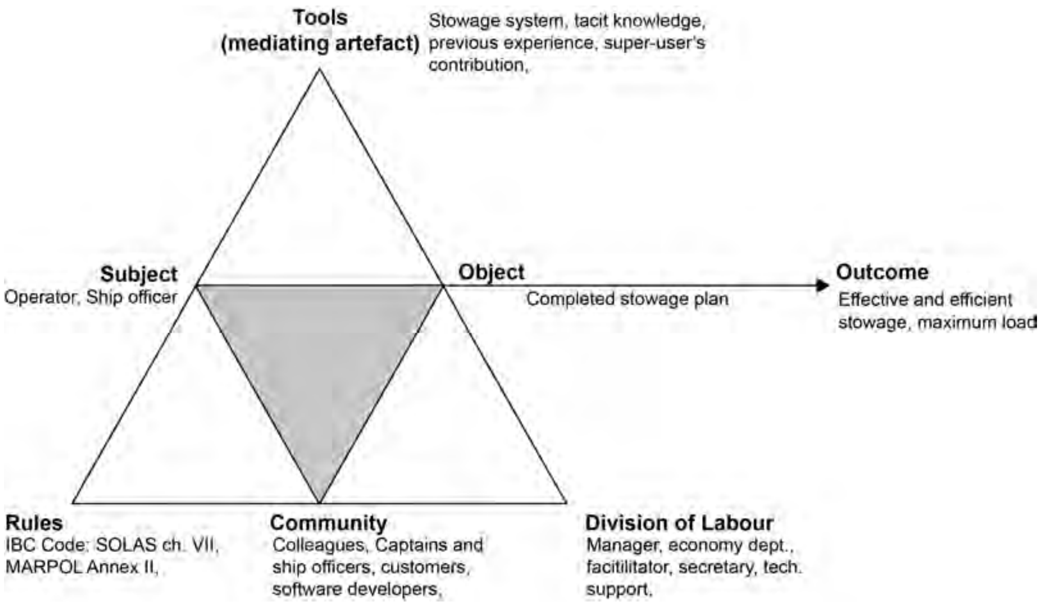


Figure 10. The collective activity of making stowage plans as an activity system.

The main responsibility for a safe stowage of the ship lies within the role of 'The Master', the highest-ranking officer at the shipboard, normally the captain of the ship. In the 'Odfjell Tankers Requirements and Guidelines Manual' it is stated that

> 'When new cargoes are added to the Cargo/Voyage Order, and new stowage or changes in stowage are required, the Master shall send an updated stowage plan to the Ship Operator.
>
> The Ship Operator may comment on the Master's stowage, or himself suggest stowage (many Ship Operators are former Chief Officers or Masters), but it remains the Master's responsibility to safely and efficiently stow the ship.
>
> If more cargo is available for a voyage than the ship can carry, the Master shall seek advice from the Ship Operator to optimise the cargo intake and the voyage result. The Ship Operator may take commercial considerations unavailable for the Master into such planning.'

When a ship arrives at a port, the captain must hand over the cargo manifest to the port authorities. The cargo manifest is a report containing the information about all the cargo aboard, where the various 'packets' were loaded, how much there is, all IMO classifications, US Coast Guard classification and so on. There is a lot of information that needs to be included. It is a standard form. In addition, the captain must produce an overview of the work on board, and so, while planning a rotation, there must be a plan for every pier. In addition, tanks might need cleaning, which must be planned much in the same way as loading or unloading cargo. This means that there are many processes the ORCA system must cover. Also, it must be able to view and print out plans to port authorities on request. The experienced, former captain – now super-user, explains by using the port of Rotterdam as an example:

'In Rotterdam, for example, the port authorities come on board when the boat docks, retrieving the stowage plan for the ship, checking the cargo when the ship arrives in the port, and also when it departs. They really go thoroughly through the stowage plan, for all the various quays in the port. And then they check that we are following the rules and regulations when stowing the ship. The rules are mainly IMO (International Maritime Organisation). However, we relate very much to the US Coast Guard compatibility list, which is even easier to work with. In this system the various loads are enumerated within different groups of chemicals. e.g. sulphuric acid is a group of two, phosphoric acid is group number 1, oil products belong to group 33, etc. So, when stowing, we just look up the table, check what goes to where. Group 2 is not really compatible with anything. Apart from group 33 [smiling], see attachment 2. This list makes it very easy to see what you are allowed to do and how, and what you're not allowed to do.'

5.3.1 Rules and regulations

There are a large number of rules that govern how a chemical tanker is allowed to operate. In addition to all of these rules, the operators must consider the ship's stability. If stowing only the middle parts of the ship, or just in the fore end, the ship might break. There are numerous factors that come into play in the daily stowage processes. The ship is loaded in one harbour, and unloaded in another, but the ship might also utilise loading and unloading operations in the same port, or loading and cleaning operations. Also, the charter operation itself could be a rather complex process. The largest ship in the fleet contains 56 tanks, which would, theoretically, give the operator 7×10^{74} possible configurations (a vessel with 56 tanks = $56! = 7 \times 10^{74}$). This would indicate that there is no single correct stowage of a ship. In addition to that, the charterers have the option of changing the load with e.g. ±10%, which in turn would further increase an already astronomical number of possible combinations. The system's main purpose is to simplify the planning of the stowing process, for marine personnel and operators on land. Its purpose is to ensure that operators don't make stowages that are illegal according to rules and regulations, like e.g. stowing two loads of chemicals, side by side, that is not allowed to be stowed alongside each other. This would have the system inform you about it, by issuing a soft rule violation – or even prevent the operator from making the stowage altogether, by issuing a hard rule violation. The system has been planned to analogously follow the established work practice, so by utilising the ORCA software, planning different scenarios by changing the stowage combinations should come naturally for the operators by associating the various operations as familiar with or similar to previously acquired knowledge and experience. A little change of volume here, adjusting the temperature there; one might free up a tank and be able to carry more cargo.

5.3.2 Stowage of chemical liquids

The stowage operation itself requires a lot of expertise among the operators, due to the many variables involved. Prior to the early 2000s, this was done with pen and paper.

The material conditions are given: the operator has a list of shipments that the company has agreed to carry, then the operator will find available ships that can accommodate loads, which also meets restrictions on how deep a ship can go in the harbours it has planned to visit. In addition to the material requirements, there are all the rules and regulations related to the transport of chemical liquids at sea.

The new system is developed in order to support the rather complex process of stowage of chemical tankers and the order in which the ship will enter the ports. The company operates ships that vary greatly in size, ranging, at the time of writing, from 14 to 56 tanks. In principle, there can be 56 different loads of chemicals on board. Some chemicals cannot be stowed adjacent to each other. Some require a specific temperature, some require heat and some require refrigeration. Some require transportation under pressure. Some chemicals react with other chemicals, and some loads will not be accepted by the charterer if certain loads have been in the tank in advance. For example, one cannot fill oil intended for use in food production on a tank that has recently been used to transport toxic chemicals, even if it has been cleaned.

5.4 The previous systems

There have been two prior stowage planning systems in use before the development of ORCA started; the *Othello* system, which was a web-based, stowage planning software that was developed in the early 2000s. Othello was a rather comprehensive system that contained quite a large set of functionality. This system was used by the operators on land. The other system was *Super Cargo*, a light standalone application for stowage planning on board. They are briefly presented here.

5.4.1 Othello

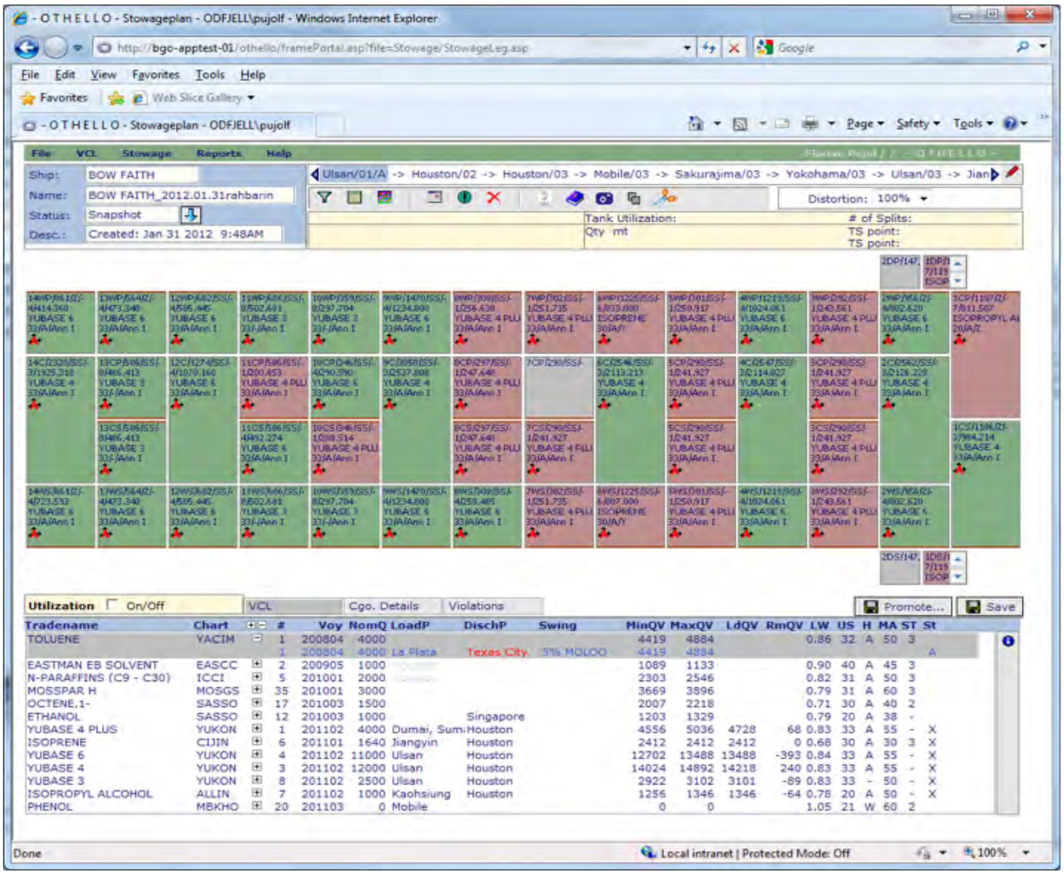


Figure 11. The main user interface in the web-based Othello system.

Othello was the first computer-based stowage system for chemical tankers that was developed. During the development process and, eventually, reaching the last iterations, it was clear that the software had become more and more complex, and with an increasing lag in response time. The rather slow performance due to an increased response time was regarded as an obstacle for rapid employment during the initial implementation case of Othello, with operators being forced to accept 20 to 40 seconds waiting time being the normality, leading to the operators into forced idleness and a sense of frustration. In addition, the software did not show proper error messages, and thereby failing to give the operators the necessary system status feedback. As one of the operators said:

> 'You did not know that anything was wrong. You just waited and waited and waited. The system looked like it worked, while in reality, it had silently crashed.'
>
> - An operator on the Othello system

There are several aspects related to the process of stowing a chemical tanker that might be very complex, with e.g. cargo-splitting, which means that an operator can load one specific cargo several times in a tank in several ports in order for the tank to be filled up, and also unloaded again. e.g. unloading the tank in several stages; this is split loading and split unloading. Another complicating issue is transshipment, where a cargo is unloaded at an intermediate destination and later transported to another destination, and loads can also be mixed. It's very complex in a system technical approach. The Othello software related to this level of complexity in a manner that forced people into long and laborious work processes. In addition, they had chosen to develop Othello as a web-based system, which also added to the technical complexity. The project manager for the Othello system said:

> 'We had chosen to develop Othello as a web-based system, so we had to tweak a hybrid system with some .NET stuff for it to run a little faster.

> Therefore, the technology that made up Othello was very complex, which was one of the main arguments that we eventually had to drop it, and start over again, because we knew we had to replace Othello. After all, we knew how bad things could become if something went wrong. We needed a system supporting the operator to weed out the errors leading to critical conditions. It was a kind of safety motivation, which perhaps is the strongest motivation of all, but optimisation purposes is also an important part. And then there was the fact that we had trouble getting hold of skilled maritime personnel, who could stow a ship based on experience.'
>
> - The project manager for the previous system, Othello.

In the initial phase, when starting to plan the development of the new software, the company checked whether it was possible to buy an off-the-shelf system, an easy-to-use program that would facilitate the required flow of communication, related to stowage planning processes, between the personnel aboard the ships and the operators on land. The Othello software did not meet the new requirements for facilitating the exchange of information needed by the operators and cargo brokers at the company headquarters. Therefore, the company had to search, whether there was an off-the-shelf software available, but no one had a ready-made software that we could use. All three software vendors that were contacted claimed that their software could be further developed.

5.4.2 Super Cargo

Since it would not be feasible for Othello, being a web-based system, to run efficiently on computers at sea, the company installed a different program, Super Cargo, on computers on board, and users were reported to be rather satisfied with it, although it had become increasingly outdated over the years. However, Super Cargo was a system that was designed by one single developer, and without a proper organisation behind it, which were regarded as a drawback, as it could possibly pose a vulnerability during support issues, and fell short when the company decided to develop a new and modern stowage system. Sources in the company could tell that if there had been a proper software engineering company behind it, they would most likely have opted for developing the Super Cargo-solution further, since the maritime personnel knew it well, and were quite pleased with it to such an extent that they did not initially wish to change software at all.[23] However, the company turned down Super Cargo, and received a lot of criticism from the naval personnel.

The company also communicated with a major software company in the US which came to Norway to demonstrate their system. The program was regarded as sufficient, but not very different from what they already had. The American software company claimed they could develop it, but could not meet the Odfjell's wish to have a continuous partner during the development period as they were located in the US. Instead, the choice of a software development partner fell on a local company, also being the same company that developed Othello, which meant that they already had an extended understanding of the specifics of the legislative framework that regulate the process of transportation chemicals at sea. However, due to the wide acceptance of, and also satisfaction with Super Cargo among the important user group consisting of the maritime personnel, it was decided to transfer what had shown to be the important functionality in Super Cargo to the new system, ORCA.

23. Comments from maritime personnel. see ch. 5.1 - Planning the new system.

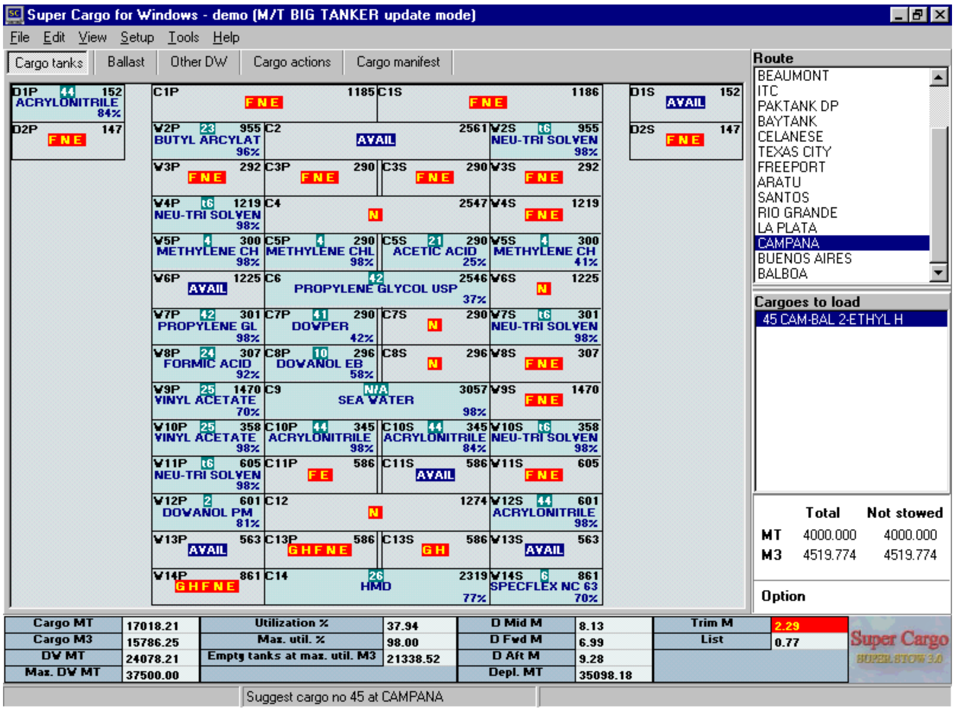


Figure 12. The main user interface in Super Cargo.

5.5 Goals for developing the software

The motivation for developing the software came from within the organisation. They decided early in 2011 to develop a new ICT platform in order to provide one common software solution to support Odfjell's commercial shipping activities onshore as well as on board the vessels. The new stowage software, named ORCA (Odfjell Resource Control Application), is part of that platform. The new system was meant to support a new workflow, allowing the operator on land and the shipboard officer to work on the same stowage plan with the same software. This development period lasted from late 2011 to early 2013. From the very beginning, the company vision was to facilitate a high degree of user centredness, and the key personnel within the company discussed what would be the most important requirements for the new software they were about to develop, such as support for an 'intuitive' workflow for skilled users, and allowing operators on land and officers at sea to work on the same stowage plan with the same software.

An additional motivation for creating a new system was that the old one had become slow and outdated. Also, they were about to replace their business system, which would render the stowage system they had, obsolete. So, in deciding on building a new system the question was: should they change and adapt what they already had, which was old, slow and a bit dull. Or should they buy something new? They decided to go for a new system, and to buy off-the-shelf software. The 'project owner', says:

> 'We made a list of experiences–both positive and negative aspects, and so we made a requirement specification that was relatively little detail. And so we sent it out to three companies that have made stowage system earlier. And then it turned out that there was no off the shelf we could buy, we had to develop it ourselves.'

In an interview, the 'project owner' stated that '… one of the goals for developing the system was that it should facilitate communication between operators on land and the personnel at sea'. The intention behind the development of the previous system, Othello, when it was created, was that it should have been implemented onboard the ships as well as at the headquarters on land, but that turned out rather difficult, and so difficult that they probably would never have been able to do so. It simply turned out to be too complex and cumbersome. According to the project owner, Othello was a system where it seemed as if they had utilised the existing technology at the time to its limits, and, maybe even a bit more than what it was made up to do. So, creating a system that would be able to provide for the communication between personnel on land and at sea, was quite an important issue, when planning the development of the new ORCA system. The 'project owner' also stated that 'we had never been able to install Othello on board the ships today, so we had to do something else. The actual stowage is one of the most important things done on board.'

When Othello was made, it was the first time anyone had made that kind of system. The task of a system like this is to simplify a rather complex job onboard. According to the current project owner, the development of Othello suffered a bit from a 'nice-to-have' approach; where lots of features appeared little by little, and one had perhaps not quite such a clear vision of what they wanted the software to do. The 'project owner' stated that

> 'we fumbled a bit in the dark. So we invented the wheel once again, and while we did, lots of nice features appeared; features that were never used. You may very well call it "scope creep", because that's what happened. Now we know what we want. We know what worked and what did not. So, in that respect we have been better off this time, since we can draw on prior experiences.'

The initial underlying vision for this software development project was that systems should be immediate and self-explanatory for the *skilled* users that are going to operate it on a regular daily basis.

From the company's annual report from 2011, we are informed that:

> 'Fleet composition, fleet scheduling and optimal vessel utilisation are thus critical success factors. Flexibility and interchangeability of ships between geographical areas and trade lanes are an integral part of Odfjell's business model, and are facilitated by our large and diversified fleet.' (Odfjell Tankers Annual Report, 2011).[24]

The company decided in early 2011 to develop a new ICT platform in order to provide one common software solution to support Odfjell's commercial shipping

24. http://www.odfjell.com/InvestorRelations/AnnualReportsAndPresentations/Documents/ODFJELL%20Annual%20Report%202012%20NY%20Final.pdf. Accessed: 17.10.2014

activities onshore as well as on board the vessels. The new stowage software, named ORCA, that makes up this case, will be a part of that platform.

During the last eleven years they have based their daily stowage operations on one specific application, Othello, which, although sophisticated, contained features that were rarely used, and in principle just made it more complex and slow to work with. In addition, it was intended for both vessels and operators on land, but was never implemented on the vessels, which in turn added even more to the logistical complexity of managing the stowage processes, as they had to run various stowage simulator systems on board, thus needing parallel systems, not directly communicating with each other, in order to complete regular stowage processes.

The new stowage tool, ORCA, was intended to support a new workflow, allowing the operator on land and the shipboard officer to work on the same stowage plan with the same software. This software was developed from late 2011 to mid-2013. From the very beginning, the company vision was to facilitate for a high degree user centredness.

The company developed a Vision Scope document for the project, describing the requirements and features of the future stowage software, stating that the new software, apart from assisting the personnel in obtaining the highest possible vessel utilisation, and ensure high quality stowage, should be flexible and user-friendly and also adapt to organisational changes. In addition, the system should enhance proper communication and contribute to better collaboration regarding stowage-related issues. It should be flexible in regard to both existing and upcoming limitations and opportunities that will influence the stowage of vessels and finally assist users in further developing skills and knowledge of stowing chemical tankers. (ORCA Vision Scope document, unpublished)

Functional requirements of the system include:

> Importing information from IMOS[25] voyage plans, including
> Voyage Itinerary
> Cargo information (volumes, options, etc.)
> Making stowage plans, including alternatives
> Checking stowage restrictions and the possibility to overrule violated restrictions
> Trim, list and draft calculations
> Generating suggestions for tank selection for cargoes
> Transfer of stowage plans between office and vessels
> Stowage tool to be available on both Odfjell owned and T/C vessels
> Maintaining all basic data to be able to operate independently
> from other systems
>
> (ORCA Vision Scope document, p. 5, unpublished)

At the start of this project, the key players within the company discussed what might be the most important requirements for the new software they were about to develop. Focusing on the users, they established a set of principles that should ensure, as much as possible, acceptance of the software in the user group. The main super-user in the core team described the initial setting as:

> 'Would the users accept the program? This was the first and largest requirement, it had to be user-friendly. We had to have the users in mind. The first thing we looked at was what the screens should look like; how the information that users should work should be presented. What we did was to combine the two programs we already knew and used, Othello and Super Cargo. We had already used Othello for over ten years and knew it well, while Super Cargo was the system that was used onboard the ships.'

So then we involved representatives from the maritime personnel, who had used Super Cargo and knew it well, and a group from the office who knew Othello, and, without hurrying the process, discussed various solutions with them before we decided who should actually deliver the system.

25. Institute of Maritime Operations and Systems

5.6 Chapter summary

This chapter has presented the case of the shipping company and the rationale of the design and use of a new stowage planning software. It has described the case study of a top-down initiated, user/expert driven development of a vocational information system for stowage planning, characterised by continuous and extensive, bottom-up, communication between developers and users, with developers co-locating with the users on a long-term basis.

CHAPTER 6.

Design, development and participation

This chapter presents, analyses and discusses the part of the material that is related to design and user participation in the development process of the ORCA system.

This thesis did intend to explore aspects of intuitiveness in graphical user interfaces (GUI) with an activity theoretical approach, primarily from a use perspective. However, it became clear that how intuitive interaction can be designed would be an important question. Thus, how such software ought to be designed had to be included.

In this chapter the initial planning, design and development of the ORCA software are presented. It describes the assembly and organisation of the core team, from choosing the participants to the co-location of the development team, in addition to the decision process and the underlying vision for how to transfer a set of vocational best practices into a best practice supporting software.

The pioneering Scandinavian participatory design discourse was initially focused on the social and political transformation of working conditions, also an increased influence by workers regarding introduction of computer technology at the workplace was incorporated (Bansler and Havn, 1994; Bjerknes and Bratteteig, 1987; Bjerknes and Bratteteig, 1995; Bratteteig and Wagner, 2014; Ehn, 1993; Greenbaum and Kyng, 1991; Nygaard, 1979; Schuler and Namioka, 1993; Simonsen and Robertson, 2017).

'Mutual learning' is among the basic characteristics of participatory design, representing a procedure where all parties, including developers and users cooperate in sharing the adequate amount of knowledge in order to understand each other's approaches when developing a future IT system (Bratteteig et al., 2013). Therefore, even though the main focus in this thesis is about the research and discussion of the foundation for the perception of intuitive interaction when doing work through mediated artefacts like a user interface on a computer screen, parts of the contribution lies in the discussion of the methods and approach of design and development. Particular attention is paid to the level and character of the physical proximity between users and developers. The analysis for patterns in organisational structure, and the discussion of the formal and informal knowledge exchange of the mutual learning process, as it unfolded during the development period, has also been previously published in Bakke and Bratteteig (2015).

Before the 2000s, there was no planning software for stowing chemical tankers available that suited the company, who wanted to utilise the expertise of their experienced personnel. All stowage plans were made with pen and paper, which required a particular set of skills, acquired by years of training, as freight planning operators or, preferably, as ship officers. Making one single stowage plan for a ship could take hours while the need for working with multiple stowage plans rose as the demand for efficient and economic utilisation of ship resources increased. The company management realised the need for electronic stowage planning, and started to look at various electronic devices for stowage control in the early 2000s. At the University of Delft in the Netherlands, as a part of a master thesis, it had been developed a prototype of a stowage planning system, JoStow,[26] for a competing company, who wanted Odfjell Tankers to join. However, the Delft-system was a fully automated system where the operator just had to input the list of loads and then the system would produce as many stowage propositions as the operator would request. The company assessed that system, but came to the conclusion that it did not, in a sufficient manner, utilise the expertise of the operators on land nor the competence of the naval personnel. Neither would it take into account the constraints of the load order of the cargo, the trim[27] of the vessel and other aspects of importance during the stowage process of a chemical tanker. Therefore, the Delft approach was abandoned rather quickly, and instead the company agreed to cooperate with a competitor, to develop a set of requirements for a stowage system. This set of requirements, a 600-page report, eventually became Othello, which today, is the *previous* stowage system used by the company's operators on land.

6.1 Planning the new system

There will always be some processes that are more important than others when planning to change or replace large technical systems, perhaps especially in technical development projects involving humans and human agency. It is precisely in the meeting, or the interaction, between humans and technology, that it is meaningful to discuss human agency through the intuitive use of user interfaces.

It was an explicit objective from the start that the system should be very user-friendly, almost self-explanatory. This would preclude the need for a lengthy train-

26. M. de Bruijn. (2001) *JoStow: a decision support system for stowage planning of chemical tankers*. Masters thesis, Report 96.3.LT.4729, Transport Technology, Logistic Engineering.

27. The vessel's position in the water. See marking no.[5] in figure 14, p. 135.

ing period. The idea was that users should be almost immediately operative. User participation in the development period was seen as a necessity.

During the initial and early phases, the company ran two user surveys, one during startup, and one a bit later when the project was up and running. These surveys were directed towards regular users. Further in the development process they ran workshops to find out what the users really wanted and which task supporting functionality would be optimally preferable. These workshops made it possible to involve a rather large group of people, both in order to collect as much first-hand stowage process competence as possible, but also to sort out disagreements, since there are two main groups of users, the nautical personnel and the cargo brokers on shore.

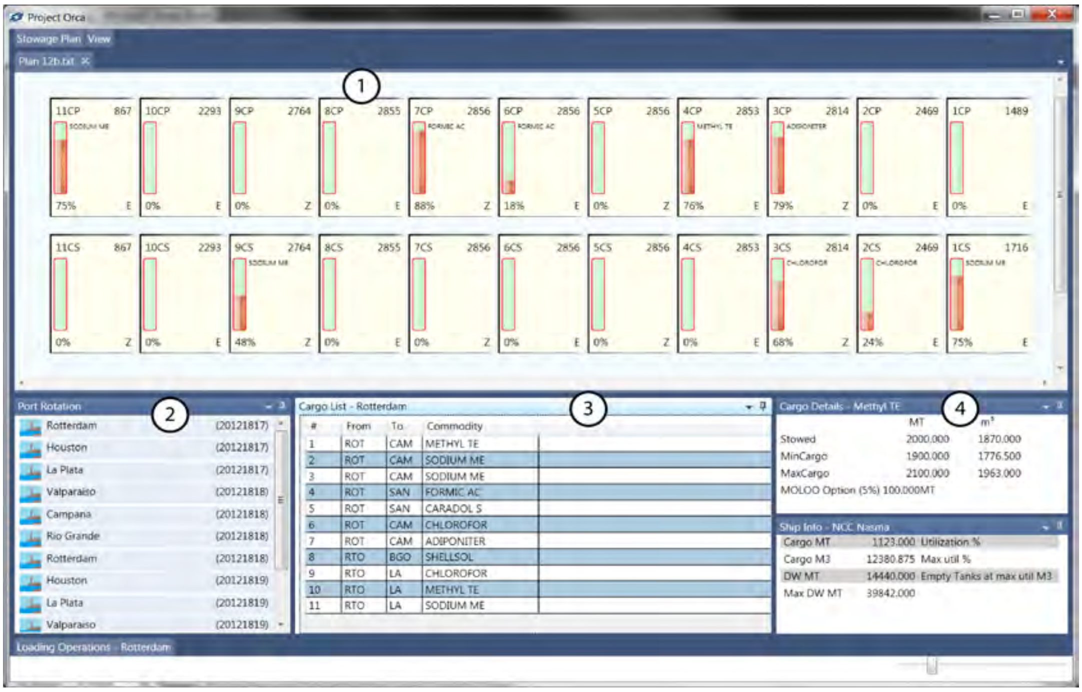


Figure 13. The first iteration of the tank plan in ORCA showing the main screen elements in the user interface: the main tank plan (1), the rotation list (2), the cargo list (3), and cargo detail/ship info (4).

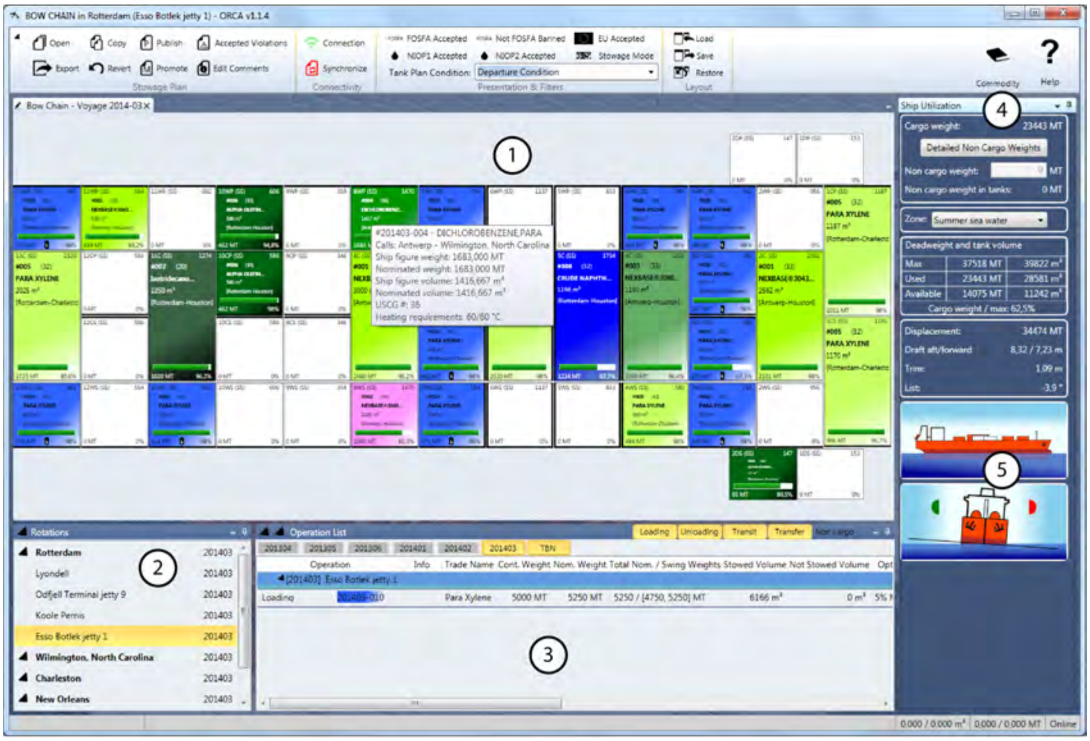


Figure 14. The last iteration of the tank plan in the final release of ORCA. Although metaphorically matured, it has kept the main functionality from the first iteration: the main tank plan (1), the rotation list (1), the operation list (2) are all in the same screen area of the screen. The screen elements called cargo details and ship info have been collected and recategorised into 'ship utilisation' (4). The new element is Trim & List; visualising the balance of the vessel in the water, based on tank fillings (5).

Some visual details have undergone slight changes, and some small portion of the information that is in there, changed a slightly, but the main principles that the company decided early on, that is, of making the ORCA system as user-friendly and task familiar as possible, would be the same. In an early interview, the project manager explained:

> 'As project manager, and project owner, I was very concerned that a good user experience should be our final state. And that the specification for that had to be defined well in advance. We have not defined success criteria yet, considering how quickly the system should respond and stuff like that, but we focus strongly on it. So – the user experience has really been a driving force. We, and also the developers, have been very focused, and aware of this being the user's system.'

The main view or vision that reflected the developers and stakeholders in the case study was that a specific work task-related software should reflect the profession- and field-specific discourse it was going to support, i.e. if one already knows a profession and its work processes, one should be able to operate the supporting software without the need for longitudinal tutorials. Thus, the new program would show the same parameters and elements, loading list and rotation list, regardless of whether the plan was made in Norway or Dubai or wherever, and the crew should be able to stow cargo immediately. In e.g. Super Cargo everything, including the entering of loads into the system, had to be done manually. The users even had to determine which controls the program should enforce. In the new system, the categories and type of controls it should enforce and categories of functionality it should offer would be done automatically.

The company had to overcome scepticism from the users, who were critical to change something they already knew well. Especially from the maritime personnel, who had to abandon Othello early on because of Othello's complexity made it useless with demanding requirements regarding technical infrastructure. Instead, they used the simpler and technology efficient Super Cargo software on board the ships, with which they had grown accustomed to, and liked. Now, they were anticipating new and complicated software tools with steep learning curves, and it would take some time, arguing and explaining, and a period of doing presentations of what the new system could do and how to be approached, task flow-wise, until their opinion changed. One of the core team members recalls:

> 'From the maritime personnel we really got a lot of criticism. With phrases like "Don't mess with Super Cargo", and "Don't make another portlog", for example. We'd previously worked on a portlog system, which did not work properly. They were also terrified of getting anything like what we had in the office, like Othello. Eventually, when we finally got the project going, and

> they were presented for what we really worked on, the "naval" opinion against what we were doing turned'.

Interviews with management and members of the core team revealed that it had become clear to the company that they needed a new system in order to streamline the port stays. In Houston, for instance, there could up to 15 berths on one port stay. It is a puzzle, which includes many parameters. When a ship is finished at a pier and going to the next, there might be another boat at that pier, leading to either an extended stay, or a forced change of plans. It is in order to make the planning procedure easier that the company chose to develop a new planning system, to replace the old system, Othello, that ended up becoming too complex and time consuming, and subsequently became an additional burden in the daily work tasks.

The project manager during the development of the previous system, and present management representative explains:

> 'Summing up, we might say that the essential difference between Othello and ORCA, is that it looks simpler. Kind of iPad-like. You do not need as much training in how to use it. It is more graphical. Othello was also graphical, but the users had to understand a lot prior to operating it.'

6.1.1 Early phase

During the initiation phase that preceded the actual software development period, and in deciding what to develop, a group of five Odfjell employees arranged a project workshop at a secluded place in the Norwegian mountains. Here, they stayed all together, and in three intense and focused days with discussion and working 'around the clock', having dinners into the night, and eventually came up with the main features of the system. The company's main super-user said:

> 'Many complex problems got disentangled up there. Many decisions were made during these discussions. Most of them were mainly focused around

> the user interface. Who is the user? How will the system be used? What should we try to get it to look like?'

Their initial main focus was the user interface. Which has led to the fact that the screen elements from the first mockups is more or less identical to what it looks like now. There have been some slight changes in user interface details, and some of the information that is in there has been changed, but the main principles of functionality are the same. The project manager for the ORCA project says:

> 'As a project manager I was very concerned that a good user experience should be the final state of the software. This must be defined in advance. We have not yet defined what should be the criteria for success, regarding the speed and responsiveness of the system, but we focus strongly on this aspect. So, the user experience has really been a driving force in the process. We are very aware of the fact that this is the user's system. […] In the first phase, before the software company was even involved, when we went through the requests, and made a specification list, we were one project manager, a guy from IT, and four from the user group, I think.
>
> I have been 100% occupied by the project which, among other things, includes attending daily scrum meetings. The IT department provides technological resources when needed.'

One of the aspects used as a starting point was the very program that had been implemented on the ships, Super Cargo, with which its users were very satisfied. It became a natural way to look in order to bring some of what the Super Cargo-users regarded as user-friendliness, e.g. the abstracted tank plan representing the ship in the user interface in 'Super Cargo', as shown in figure 12, p. 124, that became the tank plan in ORCA, shown in figure 40, p. 197.

6.2 Designing relations

Findings from interviews with the project owner, super-user, and development team, describes how the core team: personnel from Odfjell and the developers have participated and co-operated in the development work.

> 'When we started the project, it was in addition to me as a project manager, there was one representative from the IT department, a super user using representing the users from the company headquarters, and one officer representing the naval personnel. They were all involved because of their experience and expertise. My job was coordinating this.'
>
> (*from interview with the project manager/project owner*)

The team from the software company comprised three developers plus a project manager/SCRUM master that was staffed up with additional people when needed. None of the developers had worked with stowage systems before, but, since it was the same company that also developed the system that was to be replaced, Othello, the developer team was extended by a quality controller who had worked with the development of the old system when needed.

During the first phase, the users were very much involved in presenting ideas and suggestions for different functions, drop-down menus, the way it might look, what they need on board. When things grew complicated, they resorted to doing workshops in order to define what the users really wanted and the optimal kind of interaction patterns in given situations. By running workshops like these, they managed to engage a large group of people in-house.

Three naval officers were involved from the start. From them, the developers received some initial comments on e.g. procedures. They were also involved in the preliminary project, when they ran the initial workshop, discussing user interfaces and which features were adamant to include. The group of naval personnel was actively involved in the pilot phase, but not to the same extent in the implementation phase. Brief clarifications in advance were gathered by sending user surveys to the two main groups of naval users, the ship operators and chief officers, by that managing to engage these groups as well, in spite of not being present at the company headquarters.

One special aspect of this particular development process was the role of the company's main super-user. With his long and extensive experience, both as a naval officer and later cargo broker on shore, he was the company's most experienced employee. His experience and competence gave him the role as an intermediary between users, company management and developers. While using the old system, Othello, the users could contact him as immediate support when encountering difficulties, which quite a number of users did. The super-users stated:

> 'We have had many problems with Othello, and there are many users who eventually gave up using it.'
>
> When the new system was planned, it quickly became clear that the users had to be a part of the development team.

About having direct access to skilled workers, the head of the development team said:

> 'What's so special about this project is that we know the users extremely well, since we are here, in the same place. We can identify the users and talk with them, and identify expertise. […] We have rather strong users. They have extensive experience with the process, and we don't know how it works (yet). We have people, super-users, to help, telling us what is important. Occasionally, when a user says that he wants it and that, we can, if uncertain, check with the super-user. And then we can discuss with the users and the project owner.

During a longitudinal software development period, the internal processes and procedures might drift. After the system were released, and implemented in the organisation and during implementation on the ships, the head of the development team reflected over the agile character of user participation.

> 'It has been almost as agile rendering some of the sprint reviews unnecessary. Everyone knew what was implemented in the release anyway, so those who tested the build, plus product owner knew what was in the sprint and what was clear and what was not clear. It was more like information to the other stakeholders who did not follow the project that closely.'

Even if regular users were included early on, the main pattern of the process was that they were represented by the super-user. Most of the regular users that were participating were just present at the sprint demos and communicated their ideas and wishes through the super-user. According to the head of the development team, it was the main super-user who really influenced the development during a sprint. He said:

> "It's good to get feedback on a sprint demo, but it is even better to get feedback during the process, while you work, and it was in this regard the main super-user was important. He was involved in conveying single features from the users at 'the broker floor' when they came up. A bit like 'what should we do next? And how.'

In addition, there were a number of naval officers involved from day one. In the beginning, they were three officers. From them the development team received early comments. They were also involved in the preliminary project, and participated in the early workshops, which worked as a forum for discussing user interfaces and what features was mandatory to include. The naval part of the group was actively involved in the pilot phase, but not so much when the implementation phase started.

In the pilot phase, the naval group sent the same type of comments as the main super-user, but from the 'seaside' approach, and a bit later than the participating operators on land. The leader of the development team said they did not want to send a product to the ships that were not good enough, or too unfinished, and then get bad feedback later. Therefore, they did one round with testing among operators on land first, before testing it among the naval personnel, with another round of feedback, changes and improvements.

A management representative described, related to the new focus on user involvement that the biggest difference between the development process for the old system, Othello, and the new system, ORCA, lied in the actual location of the developers.

> 'Just the awareness of them sitting on the same floor … the operators know that the guys in the room on the other side of the wall are working on 'their' new stowage system, makes it so much more real, instead of the abstract notion that "someone out there" is making a new program. Actually, I think that made a certain impact.'

He further describes that although they did not have a clear requirement specification, they did have a clarification of the expectations, especially by the personnel from the ships, and by the 'nautical personnel' at company headquarters, in addition to ship management. This led the base of user representatives to being quite extensive. According to him, in its inception it was even more extensive in the ORCA project than the Othello project. Also, during the Othello process, they relied on the requirement specification that was developed by a management consulting firm.

6.2.1 Common goals and visions

In this project, the primary goal focused on developing what they called 'a user's system', meaning that their main focus was a heightened user experience. The users were included from the very beginning through both formal and informal communication. A group of people was formally picked to contribute to the development, but the initial brainstorming session had the rather informal character of a mountain summer trip. Other techniques for collaboration included mail, from officers at sea, and mail correspondence between regular users and the super-user, and face-to-face meetings between super-user and the developer team about functionality and interface issues.

There had been some signs of discontent with the previous system, and it was important for the company that input from the users were taken care of in a different manner than what had been the case during the development project of the previous software. During the last eleven years, they have based their daily stowage operations on one specific application: Othello, that, although sophisticated, contained features that were rarely used, and in principle just made it more complex and slow to work with. The character of co-location experienced in the ORCA project is contrasted with the development of the Othello system, in which the software company did not co-locate. The Othello Project was met a great deal of resistance, with heated discussions among the group of people involved in the development of the software, and the operators that were going to use it. During the Othello development period, the management struggled in convincing people to 'accept' what they were about to design.

This time, both the management and the users wished to develop a software system that would support the natural task flow of how humans would stow a ship, meaning that operators with experiential skills in stowage processes should be able to understand how to use the system without the presence of [influential] breakdowns. The project manager and -owner explained:

> 'Our intention with developing with the use of the software was, if you know how to stow a chemical tanker, you should, intuitively, understand how the system works. If you do not know how to stow a chemical tanker, then you don't have the experiential knowledge of using the system. Experi-

enced personnel should not need long training. The system should be intuitive.'

Securing the transfer of vocational experience, from the experienced operators, to the developers of the software, and ensuring that the design choices were grounded within the user group was achieved by co-location. The first central design decisions were made by a core team of hand-picked, particularly skilled, employees and the project management, already in the pre-study period. This decision-making helped establish a *common* goal that remained unchanged throughout the process with just minor changes. This concurs with Bratteteig and Wagner's discussion of participation in decisions, where a project leader and his core team make the decisions, but based on design suggestions by the users and best practice (Bratteteig and Wagner, 2014, pp. 99-104).

A small detail showing an early categorisation mistake in the screen element representing the rotation list (2) in fig. 13, p. 134. It was initially, and wrongly, named port rotation, which is a term for rotating the berths in one single port. 'Port rotation' would, perhaps, not have been an illogical term for an unskilled cargo broker/ operator such as a software developer. In the released version, shown in fig. 14, p. 135, this was corrected.

6.2.2 Organisational and procedural structure of participation

As described in ch. 6.1, the users were extensively involved during the first period but later it was noted that too many user participants could cause the development process to be unclear and messy. This is analogously described by Brooks (1975). Resolution of this problem was done by running user surveys and conduct what the core team labelled back-on-track workshops employing a bigger group of people than would be practical for regular development work. Running these surveys and workshops engaged a large group of people in a controlled manner. The surveys provided brief clarifications while the workshops brought out the engagement within the user group and gave the users an outlet to present ideas and suggestions to different functions or menus and how it could possibly look.

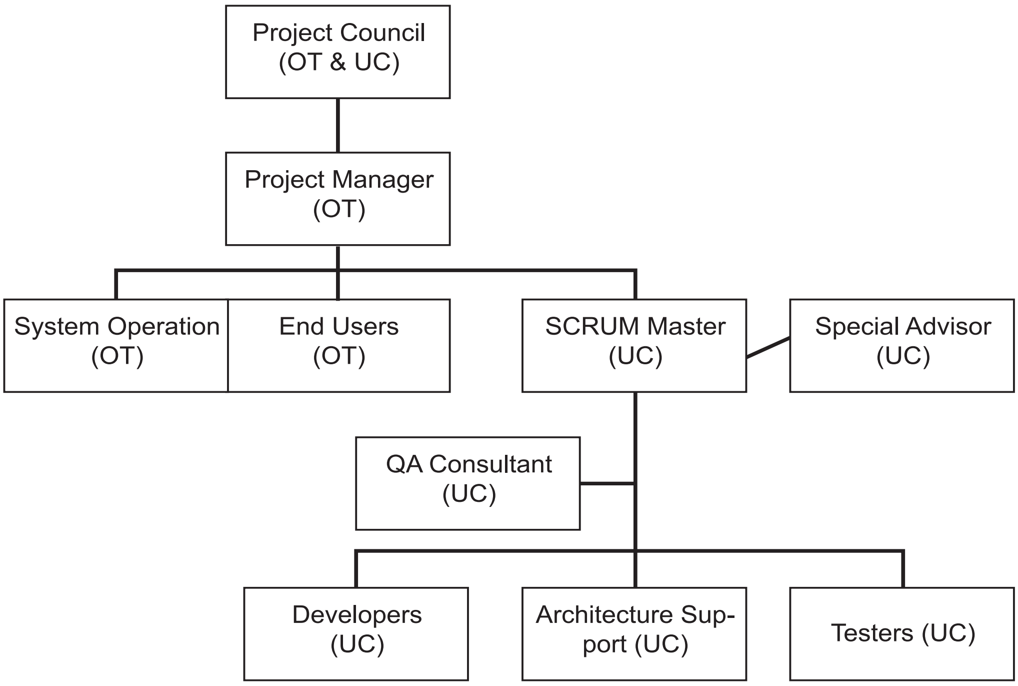


Figure 15. The project's formal organisation in three main levels, with a project council, consisting of employees from the company (OT) and the developers (UC). OT= Odfjell Tankers; UC= Umoe Consulting.

Figure 15 shows the project's formal organisation in three main levels, with a project council, consisting of employees from the company (OT) and the developers (UC) directly connecting with end users and system operation (super-user). We also see the SCRUM Master/Head of the developer team operated as a gatekeeper for the user participants' possibilities for communication with the developers. The developer team was, however, located in within a minute walking distance, the communication between the project manager and main super-user was informal, continuous and direct.

For employing an agile development method, the mixture of formal organisational structure and informal knowledge exchange and creation seemed well suited. In addition, development processes where so much emphasis had been directed towards the users, SCRUM, as an agile development method was well suited for encompassing the informal liaison between developers and users. Feedback received by the developers from the users during a Sprint could take on a rather informal character but this was not the case regarding all information exchange. After the iterations, there would be a sprint review, a sprint retrospective, and a sprint planning. Also, the workshops, where the theme would be certain specifics, would, to some extent, have a formal character.

The development process was described as extremely agile after the system was released and implementation on the ships had begun. Agile to the extent that parts of the Sprint reviews were considered unnecessary as everyone already knew what was in the release through the continuous presence of the development team. The developer team got continuous feedback, from all participant users, from the regular users to the main super user, during the process.

Salient in the project was a strong focus on mutual learning, knowing its outcome would rely on a shared, easy access to the discourse of the field (Bjerknes and Bratteteig, 1987; Bødker et al., 1987) through being close together, side by side, meeting each other several times a day by the coffee machine, or for lunch, or simply by being in the same place – mutual learning by walking around – which provided a window of opportunity to exchange knowledge both formally and informally.

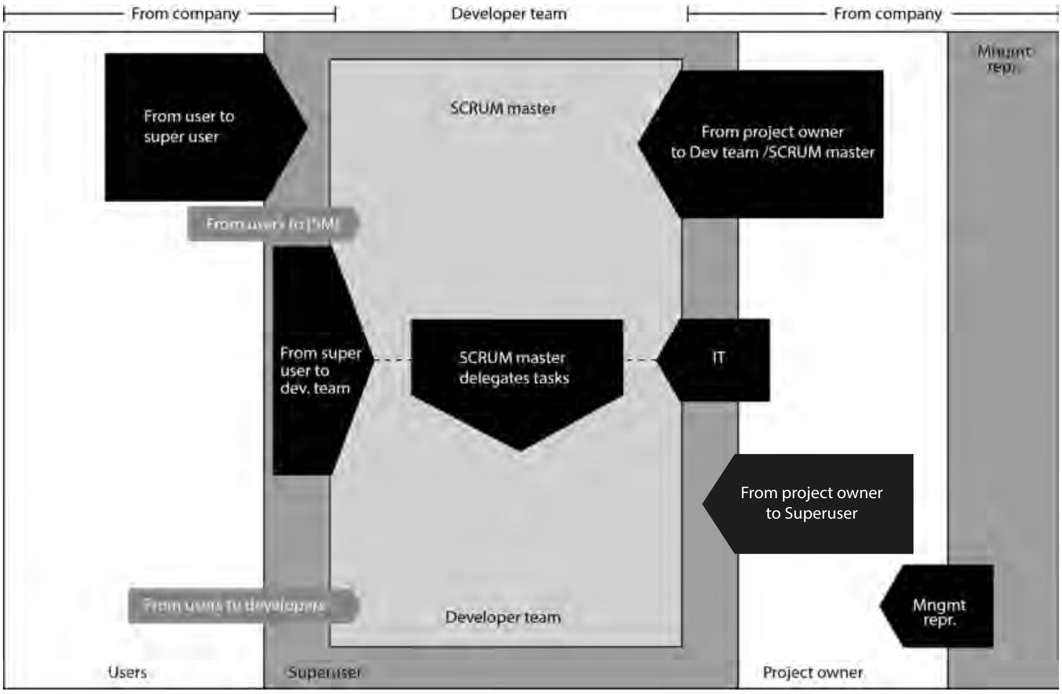


Figure 16. Information flow and direction between the members of the core team, including the user participants. This figure aims to visualise, also, to which extent the company's utilisation of the one person in the company with the most extended experience as operator/cargo broker/maritime personnel – the super-user, and how most decisions were channelled through or in different ways supervised by him.

6.2.3 Accountability and task distribution

In this project, the responsibility of giving the developers a certain direction was divided between the project owner and the super-user. Apart from project owner providing regular interests and input of software functionality, the super user was influential in giving the developers a guide and determining the vocational specifics that needed to be inclusive in the release.

In this thesis, the term 'accountability' is referring to the fact that the users and developers collaborated so close, for sake of the program's clarity and user-friendliness. In an interview, the project manager for the development of the previous system, who is the present management representative, said:

'The fact that the developers are sitting here, on this floor, makes the operators aware that the guys in the room next door develop the new stowage planning system. Instead of the rather abstract "Somebody out there is working on a program for us", it becomes much more concrete.'

During the ORCA system development, the users were well informed that the developers were sitting right next door working on a new stowage system for them. Seemingly, this sense of awareness managed to make the project more immediate and real contrary to the conception that an 'unknown' group, somewhere, is making a new program. Both developers and users expressed that this reciprocal accountability made an impact on the knowledge exchange that occurred during the development process making it possible for them to develop an explicit requirement specification and rely on communication for clarifying the expectations. The ship management and the personnel from the vessel and maritime personnel made their expectations clear from the start. The user representative base was extensive, and in its inception it was more extensive in the ORCA project in comparison to the Othello project. During the development process of Othello, the company based the development on a requirement specification developed by a management consulting company contrary to the workshops and dialogues between involved parties in the case of ORCA.

This proximity made the software development team look like colleagues and the fact that they could bump into regular users on their way to the coffee and lunch breaks, brought the sense of being accountable for choices made during the development phase and for the contribution in the process of empowering the user with a good tool.

6.2.4 Choosing user participants

One important aspect of user participation is the choice of contributors. The company is a large entity in the field and so had access to user participants with previous experience, being operators with substantial expertise, also with previous nautical experience. First the super-user was directed into an active role. According to company management, his role during the development of the ORCA system has been very important, being not only interested in taking part in the development but also in the educational aspects of implementing the new software, besides having very strong opinions about how the software should work and operate. As a long-term employee, he also knew the former system, Othello, and was very involved in improving it from a rather difficult start.

After such an extensive initial contact with users that continued while being co-located, the development team reported a clear vision of the system being as easy to learn and work with as possible; looking at aspects that could simplify the process and make it intuitive to operate the program. The head of the development team said:

> We know that if the process is simple and intuitive, users will enjoy working with the program, and then it might be a success. So, it has to be easy to use. Although the process can be a bit complicated because the business logic underneath is a bit complicated. That must not be visible to the user. And that could be a challenge for us.

6.2.5 Co-location of the development team

In participatory design, mutual learning is discussed as a key aspect. Developers should learn from users about their needs and wishes for a new solution, but users should also learn about technical possibilities so that they can participate in generating design ideas. The main aspect of mutual learning is the dynamics of reciprocal exchanging knowledge in action; i.e. dynamically, through the entire design process as a kind of 'ebb and flow' movement influencing the users as the process unfolds, something that cannot be replaced by a static requirement specification in advance (Bratteteig et al., 2013).

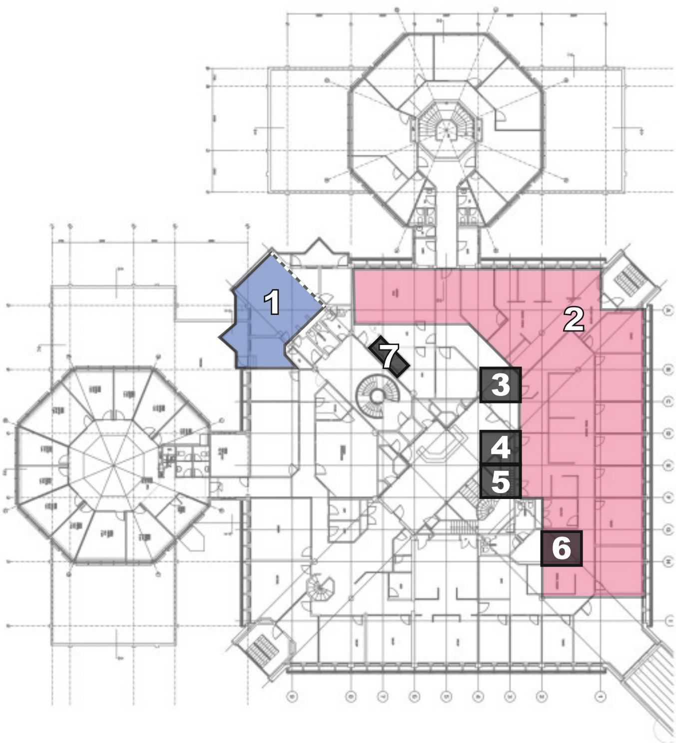


Figure 17. Proximity between users and the core team: developers and users located on the same floor. The development team, in the blue zone (1). The operators/cargo brokers in the red zone (2), project owner, (3), management representative, (4), super-user (5), expert user (6), and coffee machine (7).

By being co-located as if they were colleagues, sitting next to each other, 'only a coffee machine trip away', the designers were allowed to seamlessly tap into the rightmost levels in figure 18 of what is presented as 'expertise' and 'use of tools'-

levels by Mohs et al. (2006a), resembling Dreyfus and Dreyfus' 'competent', 'proficient' and 'expert'-levels (Dreyfus and Dreyfus, 1986), situated in the levels of domain-specific and task expertise area (Popovic, 2000) of Simon's knowledge repository (Simon, 1955).

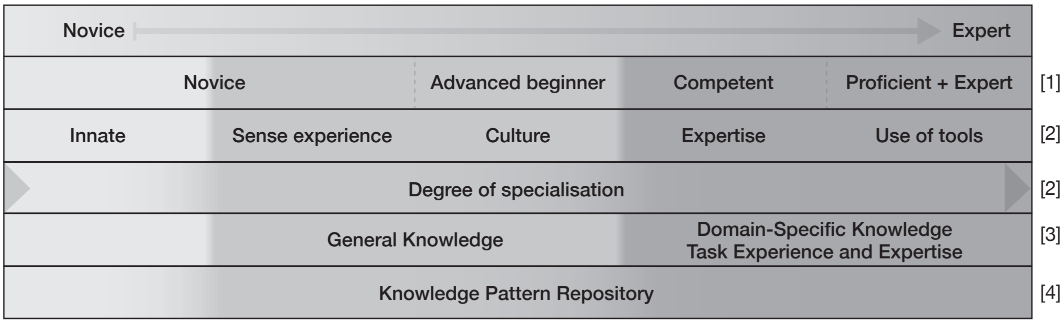


Figure 18. Levels of knowledge in the expert-novice continuum presented in figure 7 consisting of category sources by Mohs et al. [2] (2006a), adapted by including Dreyfus and Dreyfus' skill acquisition categories [1], and the featuring categories by Popovic [3], shown in table 1, p. 71 (Popovic, 2000), and adding Simon's [4] concept of a knowledge repository (Simon, 1955), in relation to the aspect of development over time.

A main characteristic of the software development process, from a participatory design approach, is that the software developers and the users moved in and stayed together from the conceptual model phase until the end of the implementation period almost two years later when operators in both HQ on land and ship officers started to use it full scale. Bringing users and developers together is not a new or revolutionising move within software engineering or development processes related to technical innovation. There are, however, aspects of this kind of co-location, related to the level of the proximity and co-operation that are interesting to explore. The previous system was developed in a traditional user-developer setting, where the developers did not co-locate but were remotely situated; which is in contrast to the dynamics experienced by the co-location of the developer team in this project.

Additionally, the reason for the joint decision to co-locate, was to obtain a certain degree of informal information exchange. The two super-users in the core team were present in the proximity and could continuously check the iterations for errors and give immediate, face-to-face feedback. This facilitated instantaneous, informal feedback, both ways. The main super-user had his own office and 'expert user' worked on the 'operator floor'. This is important since there is a vast body of experience and competence continuously accessible for the developers, giving the project an agile approach. The co-location arrangement was continuous for the whole development period. This development pattern turned out to be so agile and with such efficiency that the development team could skip some weekly project update meetings.

One of the developers talked about the possible impact of sitting so close to the future users, he underscored the convenience of being able to just walk into the room next door, to the users. With an expressed proximity like this, a developer can receive immediate answers on the questions he might have, without having to wait for any significant amount of time. In an interview, he said:

> 'For our part, this has been positive. It is very convenient to be able to just walk into the room next door, to the users. If I've got a question, I will most likely get an answer right away. And if they're not present at that specific time; they could e.g. be in a meeting, then I haven't spent that much time anyway. There was some talk about moving us to a room down in the basement, and then it would have taken me five minutes to go up to the operators, and if the person I need to speak to isn't there, I have to walk all the way down again, and there are limits as to how many times you can bear to do that every day. Then we would have ended up using the phone anyway, and then there's no point in sitting together, is there?'

According to the developers in the core team, this level of co-location, of sitting among your users, did, continuously, remind them to make the software easy to use. In addition, it made it easier for the project owner to work very closely with the developers, and to come by often, i.e. several times a day. For the developers, this way of co-operating seemed rather different from previous projects with which they had been involved, where they might have to write a list of questions or suggestions and send it to the developers sitting somewhere else and get the answer days later.

Figure 19. Pair programming and co-locating: developers, together, in the same room, room 1, on the same floor, right next to the operators who reside on the other side of the far back wall to the right, which corresponds to the dotted line on field 1 in the architect drawing in figure 17 on page 150.

By co-locating with the users, the developers had immediate access to information related to the field and the users for which and whom they were developing the support systems. However, this sense of being in the immediate vicinity could also, potentially, be double-edged. The developers become equally accessible to the users, and thus in a possibly vulnerable position, being interrupted at work. There are, however, no findings in the empirical material that this aspect of being in the immediate vicinity did pose any problems during the ORCA-project. In an interview with the leader and scrum master of the development team, he explained how they perceived this co-habited way of working:

> 'When the main super-user come by, we must always find time. We never say no, even if we are busy. After all, we're here to work with them, so we stop working, to answer their questions. Co-location is all about being available. It is not enough to sit in the same place. If you can't take the time to answer questions on the spot, you lose the whole point of co-location.'

By always having to find time, using co-location as method shows how user participants were prioritised; confirming co-location as a design method. The empirical material suggests that a co-location context where mutual learning is facilitated might be necessary for establishing a common understanding of how the workings of a use-centred software system should be, by continuously sharing the knowledge repository, build up over years of accumulated experience, as what Schön describes as 'intelligent practice as an application of knowledge to instrumental decisions' (Schön, 1983, p. 50).

Also, co-location of developers and designers, provides continuous access to skilled workers' knowledge-in-action in a dynamic and reciprocal manner, and thus the ability to inscribe how expert behaviour should be taken care of, into the system, which could pave the way for experts to work in a way that they perceived as intuitive and based on, specifically, craft-experience and action-experience, enabling skilled workers to connect the abstracted, metaphorical and mediated user interface to the experiential, physical and real (Bakke and Bratteteig, 2015).

6.3 Report: ORCA installation and training on board vessel.

Location: port of Rotterdam, The Netherlands.

Date: December 9–10, 2013

The collection of empirical data had until the end of 2013 taken place at Odfjell's headquarters in Bergen. This, primarily qualitative, material consists of interviews with users and super users, developers and project owners, together with observations of meetings between the developers, users, super users and owners. Also, observations of course participants in a classroom situation – at a tutorial workshop, were undertaken.

Empirical data collection from installation on board vessel

The super-user was the sole person within the company who installed and conducted training in the ORCA software on the vessels they operate. During this trip to Rotterdam, the new system aboard would be installed onboard the ships, JB Sapphire, a time-chartered vessel, and Bow Engineer, one of Odfjell's own ships.

The aim of the trip, in addition to installing the software onboard, was to provide basic training for the captain and chief officer while loading and unloading cargo at the Odfjell Terminal in the Port of Rotterdam, Netherlands.

Date: December 9, 2013.
Vessel: JB Sapphire. As a time-chartered boat, it was not among the prioritised vessels scheduled for installation of the ORCA software.

The installation was done by installing on each vessel via a separate USB stick with the program, which is uniquely marked for each ship. It is not distributed not via the Internet for safety reasons, and not because time-chartered vessels do not share online resources with the head office in Bergen. After a not entirely unproblematic installation, Odfjell's representative gave the boat captain an introduction of the ORCA program. The training session was estimated to last approximately one hour and was done by the super-user, who explained the most common and used functions to the ship's captain. We went through an old plan transmitted to ORCA. The ship's officer who was observing, commented: 'It makes a lot of sense.' During the training session, he believed that the program seemed logical and easy to understand. The captain went through regular stowing procedures like loading, unloading, split loads, editing port rotation, etc., through interacting with the mediated tank elements in the software user interface, making himself acquainted with the functionality and mediating tools provided by the software. He missed 'his target' on a couple of occasions, but still managed to achieve the given tasks.

The company's super user showed a subtle, and a little cunning, detail in the software that showed a technique for getting out of 'frozen program' situations; when ORCA freezes, you can 'grab' the program window with the mouse or the touch pad, and 'shake' it. The software would unfreeze and continue to run, without having to reset the entire computer. There was a comment from the observing chief officer about the warning triangle that appears when you get a so-called soft rule violation, which he claims should have been larger and more visible. He also mentioned that there should be possible to send emails directly from the program.

It is the chief officer's first contact with this software, and he understands rather quickly. The drag-and-drop functionality implemented in the software is known from the operating systems and do not represent any ambiguities. Nor do other interaction patterns that are recognisable through gradually accumulated knowledge about 'standardised' interaction through graphical user interfaces, like mouse clicking on buttons, dragging of scroll bars, etc. The officer practices, on loading, unloading and moving cargo. He gets into some slight trouble with an earlier tank plan that is imprecisely marked, and not vacant after all, forcing him to stow the cargo in a different, non-adjacent tank, an action that, seemingly, has little significance, and the overall perception during this observation of the operation of the program is that it aligns with the experiential process of how an operator in Odfjell would stow cargos.

Date: December 10, 2013.

Vessel: Bow Engineer. Vessel owned by Odfjell.

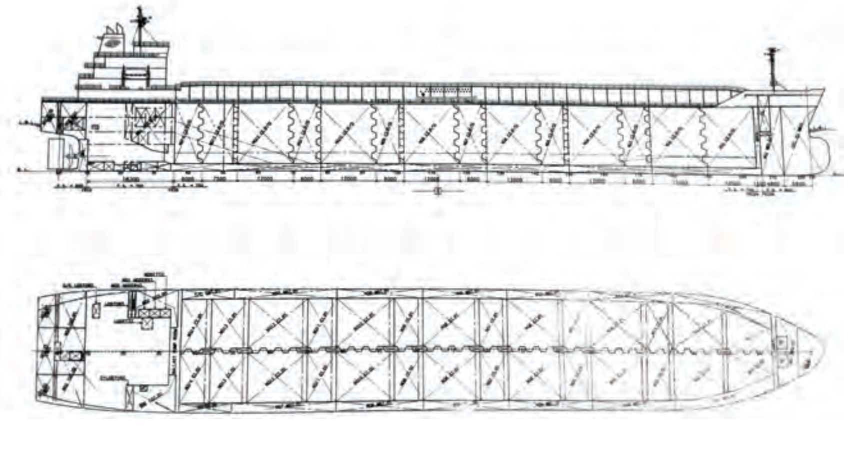


Figure 20. Capacity plan, Bow Engineer.

The software has previously been installed, but has 'stopped working' and is out of service due to the program being time-locked. For safety reasons the system must be re-unlocked with updates within a certain time after updates are made. There are two aspects connected to this procedure: in the case of theft, an un-updated program will stop working some time after an update is made available, and it also ensures that all units use the same updated software. Before a training session can take place a board the ship, the software has to be upgraded.

Since there are several new officers on board, there is a plenary session on the ship bridge where the program is demonstrated by showing a set of regular stowing tasks, such as stowing a cargo by dragging its representation from the operation list unto the main tank plan or rotations list, testing right-clicking on screen artefacts, change rotation by moving ports or berths up or down in the rotations list, split-unloading cargo destined for two berths or ports, and so on.

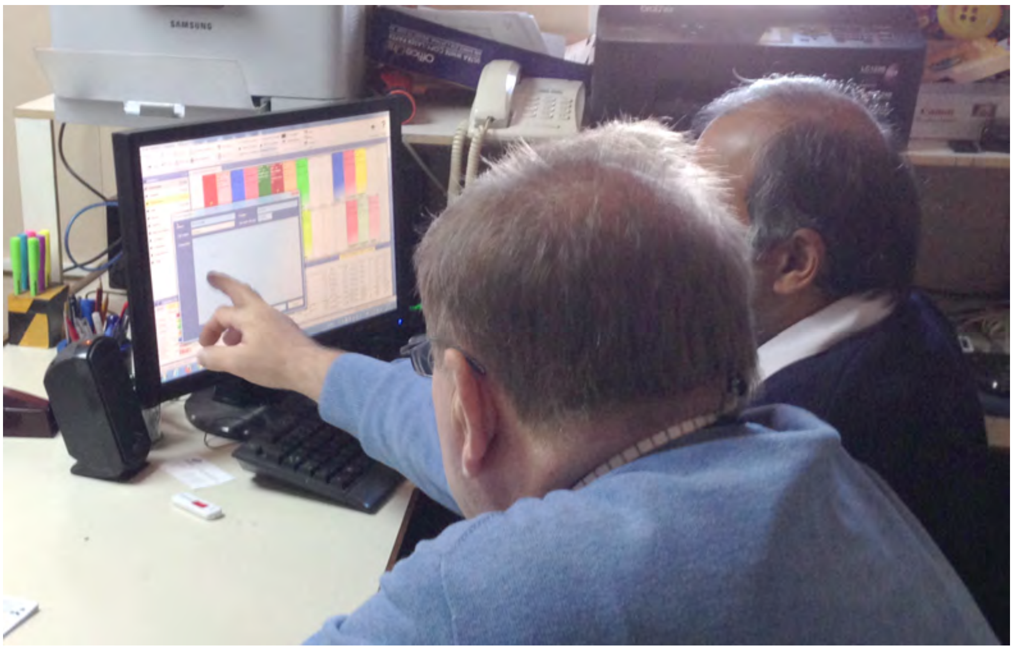


Figure 21. One-on-one teaching with the ship's captain. The coloured fields in the user interface are the abstracted tank elements in the main screen artefact, (see fig. 28, p. 178) for details.

After the plenary session, a one-to-one training is scheduled. The ship's captain has seen the program before, but not received formal training. This session is therefore more affected by the captain's questions about things he is already wondering about than first-time users training.

6.4 Discussion - development and participation

There was a need for software that could handle the complexity of stowing chemical tankers, in a fashion that would be as self-explanatory as possible for experienced users, i.e. operators who were skilled at planning the stowage of a vessel, prior to using ORCA. The main view or vision that reflected the developers and stakeholders was that a specialised software should reflect the professional and field specific discourse it was going to support, that is, if you already know a profession and its work processes, you should be able to operate the supporting software without the need for longitudinal tutorials. The software had to be used both on land, in the company HQ (operations), and on the bridge on all ships loading and unloading cargo (fleet). This meant that the group of users would be skilled, but heterogenous.

The findings suggest that the proximity had an impact on the continuous knowledge exchange, the organisational structure and the experience of being accountable in the software development process. Moreover, these elements of the software development process influenced the quality of the end result by a constant communication of priorities and targets.

The empirical material shows that a co-location context where mutual learning is facilitated might be necessary for establishing a common understanding of how the workings of a use-centred software system should be, by continuously sharing the knowledge pattern repository, build up over years of accumulated experience.

As a result, the new stowage software utilises abstractions in the user interface that, in short, are based entirely on previous practice. User participants have been involved from the very beginning, drawing on experience from a very specific work practice and environment, and also with an established division of work tasks, in addition to ten years experience with the previous software.

This resulted in a set of specifications that relied solely on former practice – intention and action. On top of the user requirements list was 'Good user experience/friendliness'. The next important points on the list of requirements were

- Support ease of communication between Vessel – Operator
- Support ease of communication between Operator – Broker
- Emphasis on an intuitive Graphical User Interface (GUI)

During the development of the ORCA software, the users were aware that the developers were sitting right next door, working on 'their' new stowage system. According to the project manager, this awareness made the project more immediate and real, rather than the kind of an abstract notion that someone is making a new program. Both developers and users expressed that this reciprocal accountability made an impact on the exchange of knowledge that took place during the development process and made it possible for them not to develop an explicit, frozen, requirement specification from the start but rely on continuous communication with the users for clarifying the expectations. The mutual learning principle, emphasise that all participants 'are accounted for', and that designers learn about the vocational settings of use from the users, and that the users learn about technological possibilities and solutions from the designers (Bratteteig et al., 2013).

The ship management and the personnel from the ships and 'nautical personnel' at company headquarters made their expectations clear from the start. The base of user representatives was quite extensive, in its inception it was more extensive in the ORCA project than the Othello project. During the Othello process, the company based the development on a requirement specification that was developed by a management consulting firm, not on workshops and dialogue between the involved parties as in the case of ORCA.

Findings presented in ch. 7.2, related to how operators do regular daily work, i.e. accepting cargo, and making test stowage plans, suggest that the user interface should be designed to support established workflows in a way that is based on experiential knowledge stemming from participating users. One operator described, e.g. the process of moving one cargo from one tank to another – a cargo transfer. Instead of just dragging the cargo from one tank to another, as would have been the logic, but unskilled thing to do. The correct process would be to first un-stow the first tank and then re-stow the cargo in another tank. Leaning on this experiential knowledge of previous actions from participant users, the system was instead designed to prohibit a direct drag-and-drop between tanks, providing the un-stow/re-stow procedure as default.

6.5 Chapter summary

In this chapter the initial planning, design and development of the ORCA software have been presented. It has described the assembly and organisation of the core team, from choosing the participants to the co-location of the development team, in addition to the decision process and the underlying vision for how to transfer a set of vocational best practices into a best practice supporting software. In this chapter the particular attention that was paid to the level and character of the physical proximity between users and developers has been described, and the formal and informal knowledge exchange of the mutual learning process, as it unfolded during the development period was analysed.

The analysis in this chapter suggested that this level of proximity in a user-developer relationship could possibly lead to a system that would be instantly usable by skilled users and easily learnt by novice users.

CHAPTER 7.
Use of ORCA

This chapter analyses and discusses the part of the empirical material that is related to the use of the ORCA software, by the operators at the company headquarters on shore and the naval personnel, in their daily work of planning the stowage and transportation of chemical fluids at sea and the observations relevant to the understanding of how skills and experience might influence their interaction with the software. Included locations are field sites, of which the main site is the headquarters in Bergen, Norway, and two of the ships and the ship bridges, during software installations, and a brief introduction to the software among ship operating personnel, where the empirical investigation was conducted.

The discussion of use-activity in ORCA is structured following approximately the order of basic principles of activity theory presented by Kaptelinin and Nardi (2006). First presenting the operators' general use-activity in the ORCA software and the ORCA user interface, followed by situating the use-activity in ORCA within the hierarchical structure of activity. Then, this is connected to a discussion of specific actions and operations in the user interface, together with reflections around the human-computer functional organ, and the perception of intuitive interaction emerging from development and practice.

In the following, the findings related to how regular, but skilled, operators are interacting with the software tool is presented and discussed in relation to the hierarchical structure of activity, transformations and aspects of development. Cargo brokers on shore and maritime personnel on the vessels interact with the ORCA software that is described in this dissertation. As presented in ch. 4, the empirical material consists of interviews with and observations of operators on land. Interviews with and observations of personnel at sea were not possible as time, and geographical reasons, did not allow a stay on board from one harbour to another. Therefore, data from the personnel at sea were collected through a simple survey. Survey data does not provide the same depth of information as being present, observing actual use-activity. It is only the users' own descriptions of their own impression of how they experience the interaction with the user interface in ORCA, post factum. This is, however, an experienced user group, consisting of mainly captains and first mates, who have the necessary skills, i.e. tool- and task-related competencies, to assess the opportunities and limitations inherent in the mediating tool, required for the operator-ORCA constellation to work as what activity theory refers to as a functional organ, becoming a part of what Kaptelinin describes as an extension of the internal plan of action (1995b, p. 51), (see chapter 2.5 - Functional organ). Nevertheless, there is, admittedly, a significant degree of uncertainty, and possible heuristic bias connected to the pattern arising from the answers, however distinct, that it would be premature to conclude from these findings with anything but uncertainty.

Previous research on intuitive interaction in interfaces (Blackler et al., 2005; Blackler et al., 2011; Hurtienne et al., 2008; Mohs et al., 2006b; Naumann et al., 2007) has shown that the experience of recognising something as familiar is a decisive factor in how people experience interactions in user interfaces. Perception is an inner, mental process related to the internal plane of action and is, as such, a part of the processes that take place within the functional organ.

From the empirical material, it is problematic to be precise and specific on whether a series of operations, seemingly performed intuitively, are perceived as one intuitive whole as one action, or if the individual operations are perceived as serially intuitive, i.e. intuitively one after another, as sequential operations. In practice, this may not necessarily matter, since it, in this case – in the real world, is the sequence of operations, and the perception of how it was undertaken which is at the centre of the human-ORCA interaction, but theoretically, this may facilitate an interesting discussion. For example, in an activity-theoretical approach, it may be significant as to which level in the hierarchical structure of the activity the analysis relates, i.e. whether a series of operations should be analysed as one [compound] action, at the middle level, or if individual operations should be analysed separately, even if they comprise one sequence, also in an example where the composite action has been automated and transformed into an operation. That actions and operations can be transformed, bi-directional; that an action can be automated into an operation or that the conditions associated with an operation might change leading an operation to become an action that is reflected over. This makes the distinction between operations and actions dynamic – and more complex.

This chapter structures the findings, analysis and discussion by the activity theoretical core principles of activity in context as follows:

“The systemic analysis of human activity is an analysis of levels” (Leont'ev, 1974, p. 32). Thus, findings in the empirical material related to use-activity are discussed in relation to levels, and the human-ORCA constellation as a functional organ.

The orientation phase and the reciprocality of internalisation/externalisation process in the human-computer constellation that is created as a functional organ: presenting the operator-ORCA as a functional organ and discussing the reciprocal process of internalisation/externalisation that is perceived through the observation of human use.

7.1 Use-activity of ORCA

The ORCA software allows an operator and shipboard management to work with the same software and work on the same, and shared, stowage plans in managing the logistical complexities of stowing chemical tankers. From an activity theoretical approach, ORCA is the software that is the developmental result from the many contradictions that occurred from the two previous systems. These systems, Othello and SuperCargo, were misaligned related to each other, as the Othello software was used at the company headquarters while the Super Cargo program was used on board the ships, without the operators on land and the maritime personnel being able to work on the same stowage plan simultaneously (see chapter 5.4, p. 120). Assessed according to Engeström's three variables for assessing development: destruction of the previous, development as a collective transformation, and horizontal development encompassing several activity systems (Engeström, 1996), the design of ORCA representing the hitherto 'maximum' level of expertise.

During observations, operators test and simulate stowage procedures, and mentally evaluate a variety of patterns to stow the cargoes on a chemical tanker. The software is designed for operators to be able to mediate this internalisation process by externalising the ideas and possibilities, they virtually tested in the mind, by the operation of screen elements in the user interface. These evaluations, that are done on the internal plane, are rooted in the real, external world and grounded in experience, which links user-interface mediation to an experience-based interaction with the user-interface through the activity theoretical notion of internalisation and externalisation. This would also, in return, transform and, over time (see fig. 62, p. 237), further develop the operator into a higher skill level as the internalisation/externalisation process unfolds (Kuutti, 1995; Leont'ev, 1974).

While being aware of being watched, the operators comment on use-activities, during the interaction with the user interface, and dialectically verbalising their operations, reflecting over how tasks are solved by human-ORCA interaction. Affected by feedback from the user interface, the interaction has the character of being a dialectical, contextual, reflective process (numbers [x] relate to numbers in figure 22, p. 168):

> I have to decide which of the tanks the 1500 m^3 [cargo] will be unloaded from. Then I have to drag 'this' into the tank [2] operator pulling a cargo unloading order, from the Operation list and up to the tank containing that cargo [that is ready for unloading] to show that she unload it from that [specific] tank [3]. This is So, when I arrive at Santos [4] the rest will be unloaded there. But here [in this case, auth. remarks.] (The operator clicks on the Rio Grande port in Rotations [5]), we look at the possibility of going to Rio Grande first, because we are going to unload the cargo.

In this example, the described use-activity of unloading the cargo from a tank, is an action that is comprised of three operations: the use-activities of clicking a line in the Operation list representing the correct unloading task [1] , dragging it towards the tank plan artefact above the Operation list [2] and dropping it on the chosen tank [3]. This was one unloading action from a tank that shared a large volume of the same cargo, where the remaining volume would be unloaded later, at different harbours: Santos [4] and Rio Grande [5]. The operator reflects on which of the two harbours the ship should sail to first. In the current rotation, and normally, the ship would sail to Santos first, but in this case, she decides that unloading in Rio Grande first would be preferable; thus she must change the order of arrivals in the rotation list. She accesses the contextual menu by right-clicking on the Rio Grande entry in the rotation list [1]. Choosing 'Move Up' [2] will move the Rio Grande entry up one place, above Santos in the list (see fig. 23, p. 169). The moving of the Rio Grande entry is one action comprised of one right-click operation and one menu choice operation. The complete use-activity, i.e. the action of unloading this specific cargo consists of two sequential actions, comprising five operations.

Figure 22. Drag and drop operations. Operator dragging a cargo unloading order, from the Operation list and up to the tank containing that cargo.

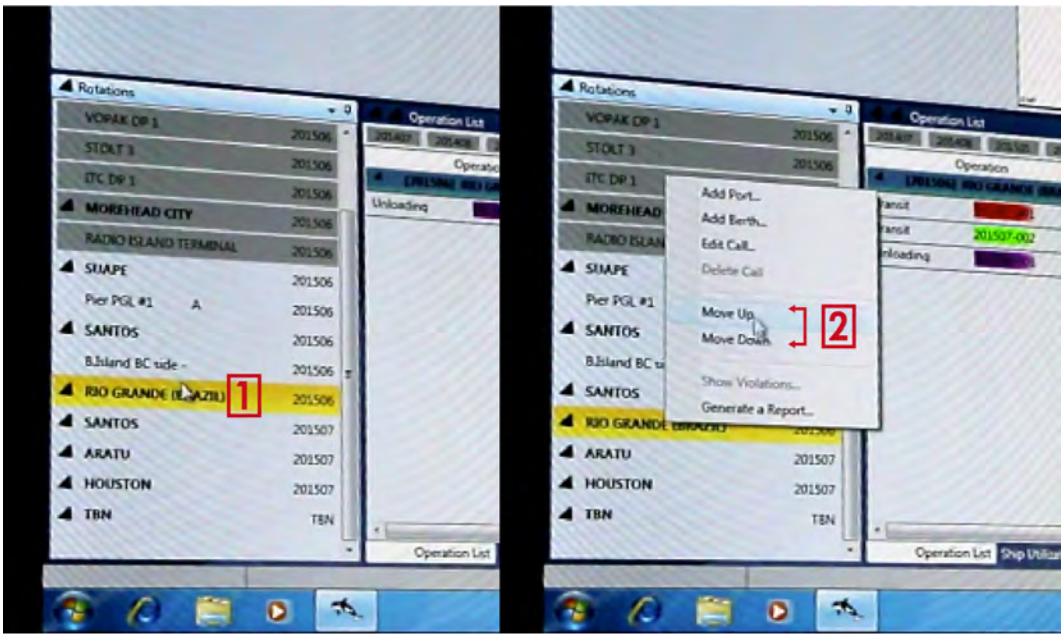


Figure 23. Contextual menus in Rotation list. The operator right-clicks on the Rio Grande: contextual menu pops up. Clicking on Move up [1], the screen element of the Rio Grande port is moving upwards in the Rotation list [2].

> Operator:
>
> Then we have changed that rotation, and can bring something ... retain something, where you plan to go, you know. Then, we go directly from Suape to Rio Grande, and then we go to Santos. With the planned stowage we now have, this will be fine. It is just one load, which is to be unloaded in Rio Grande. So, things like this I might be able to do very easily, and see if there are conflicts; to see whether the tank in which you are going to put the cargo is vacant, or not, at the time.

During another observation session, an operator clicks back and forth between the tank plan, rotation list and operation list, to check for availability. She continues the stowage process, doing what operators do all the time – testing and simulating rotation and tank logistics, and possible scenarios in the user interface; 'is *that* tank available in *that* berth in *that* port? For *that* rotation?'

Operations mediated by screen artefacts, such as clicks, holds, and dragging are done in various parts of the user interface, interactively testing and simulating a variety of rotation and tank logistics in the user interface. Initially, simulating variations of Rotation, or testing specific cargo distributions in the tank plan comes forward as an action, but could, during a process without problems – contradictions – be regarded as an operation. Nearly all functionality in ORCA, except functionality such as open, archive and close, that is common in all software, is concentrated around testing and simulation of different stowage solutions. Testing a simulation is in its nature explorative and reflective. Nevertheless, it appears that certain actions performed in such a way that they appear as operations. The fact that ORCA is designed for operating in an explorative manner appears to have an impact on users who, in a given situation or in repeated patterns of action, do not perceive reflective use-activity as a halt, but as a natural and necessary part of their interaction with

ORCA, which becomes part of the tool–and task competency that underlies the human-ORCA combination's ability to serve as a functional organ.

It is specifically the ease of testing and simulating solutions, resembling that of a sketch book. This sketch book analogy is not far-fetched; as 'sketching on copies' is one best practice when making stowage plans, and a foundation for the design of ORCA. Once, it was pen and paper that was the primary planning tool for the stowage of chemical tankers. Pen and paper were replaced by the previous system, Othello in 2000. When it was time for that system to be replaced, all domain-specific knowledge, both from the pen and paper era, and the experiential knowledge from using Othello, were put into the design and development of the new system, making it into an experience-based, domain-specific tool, tightly integrated into an established practice. An example of the practice-oriented interaction that the system facilitates was noticed during the observation of an operator testing a complicated scenario revealing a typical quay-order manipulation process involving a number of cargoes, where one cargo should be on one specific pier in the Houston port, with its four quays, while another cargo that is to be loaded also comes from OTH. The process of switching the order of quays in a harbour from which a cargo is to be loaded consists of numerous single operations that together constitute an action that also involves decisions related to tank suitability, such as coating, temperature, adjacent tanks and so on. The user interface in ORCA facilitates this quay rotation testing by virtually moving quay order in the user interface to test whether the tank that was intended to be used for the specific cargo to be loaded in this port could become available if the ship sailed to another dock and unloaded there first. This action involved several cargoes; cargoes to be unloaded and cargoes to be loaded, and the deciding factor for the operator's planning and subsequent decision is the order for when it can be done. Here, practice would suggest checking timely availability first which is related to port or quay. In stowage practice, ports and quays belong to and constitute the order of a Rotation, thus, based on that domain-specific knowledge, the program facilitates altering the rotation order in a simple well-known manner, in this specific situation, moving the quay-element in the rotation list (see figure 23, p. 169).

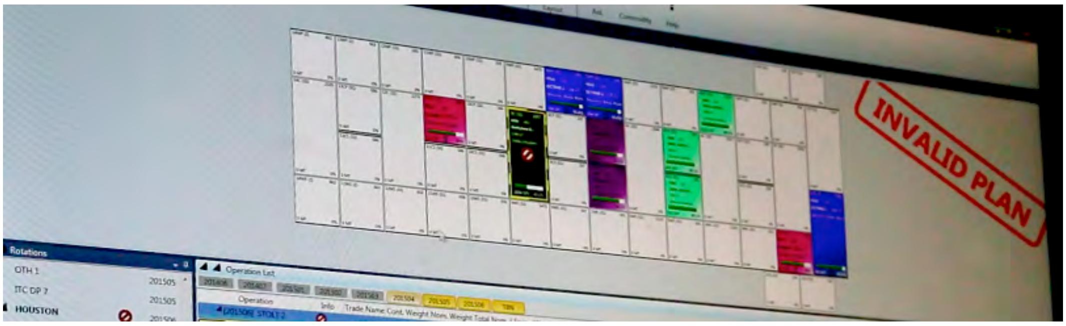


Figure 24. Invalid plan error: The operator tries to load a tank that was not unloaded:

> 'If I were going to use *that* tank, which I unloaded *here*; if I tried to load *that* tank earlier, I would have got an invalid plan error, if it was not unloaded first.'

The operator could either grab hold of it, or drag and drop it in the direction of the future berth of the vessel. The operator points at the Houston port screen artefact in the Rotation list, and at a specific cargo screen-element in the Operation list, demonstrating a drag and drop operation towards the OTH berth in Houston, and then showing another way of moving by dragging and dropping the cargo from one berth to another in the Rotation list, explaining:

> If I wish to change the rotation and want to go to LBC first, I just 'Move up' [accesses the Rotation list contextual menu] and if I then should have used the tank that I also unloaded ... *there*, for example, if I were employing *that* tank on *that* berth, I would have got an invalid plan. You cannot load a tank until you've unloaded it first. I use this a lot to change a rotation.

As previously described, ORCA is designed around a simulation and probing paradigm, i.e. the larger part of stowing a ship is centred around simulation and probing. This is manifested on two levels;

1. all stowage plans are *tentative* as default and must be actively promoted to *valid* prior to being used as an actual stowage plan.

2. externalising stowage solutions as a chain of actions, and the operations of which they are constituted, by distributing simulation capacity to the user interface, extending the internal plane of action.

During observation sessions, operators probing various stowage configurations in the abstracted tank models in the user interface, made possible by the risk-free, default 'work-on-copies' approach in the ORCA system (see fig. 25, p. 172).

Figure 25. Duplicate stowage plans. The operator can make as many duplicates as needed. Here, the operator can try out three different stowage plans: the initial plan [1], that the operator preserved by making a copy [2], which could be used to explore other stowage constellations. This copy was, in turn, preserved by making yet another copy [3], providing the operator with an extended number of possibilities to mediate the internalisation process.

In an activity theoretical object-oriented approach to human-interface interaction, the object of the activity can, according to Bødker (1991, p. 38), have three instances: 1. it resides exclusively within the computer artefact, 2. the objective exists in the real world, but is present exclusively as an activity in the software, and 3. the object exists, physically, outside the artefact but is controlled by operating the artefact, such as e.g. a control panel.

From the operator's point of view, a published stowage plan exists as a real-world object, but also as an object in the mind, and mediated by interactive artefacts in the user interface. Bødker claims that 'The quality of the user interface for this type of application must relate to how the user can couple the final object and the object "on the screen" to each other' (Bødker, 1991, p. 38). In the ORCA case, this quality relates to the use of visual representations and functional capabilities of the user interface.

Maintaining a metaphorical proximity with the real world of vessels in the ORCA user interface was a central element in the design of the software. In this chapter, the abstracted screen elements that make up the graphical user interface in the program, and the interaction they afford, are presented. In addition, the operator-interface interactions, are briefly described and placed within the activity theoretical hierarchical structure.

7.2 Mediating practice

In ORCA, where the system is based on *simulating* actions and activities, the operator never works on 'originals', and all operations, actions, and even activities are, in an activity theoretical perspective, internalisations until the goal of the activity, an agreed upon stowage plan, is reached by fixating, and then 'promoting', the plan, as the last stowage iteration, into a published final plan. This aspect of user interaction has shaped how the system in its core is designed to mediate activity.

Abstracted representation of the vessel, with manipulation and interaction directly onto the tank representations, which provide direct system feedback, providing immediate feedback. The Operation list beneath the tank plan has direct drag-and-drop functionality, both related to the Rotation list to the left and to the tank plan above. By leaning heavily on the extensive base of competence that existed in the organisation, the co-located development team had access to what had developed into best practice examples the activity of making stowage plans.

As previously described, assembling a stowage plan is a complicated process that requests an extensive amount of reflection and testing before a plan is 'frozen' and publishable.

Observations during stowage sessions suggest that the best practice, activity centred, 'sketching' approach characterises the interaction in ORCA, as the core design pattern. This facilitates or maybe seeks possible contradictions in an attempt to visualise incompatibility in an action. In the dialectical logic perspective often associated with activity theory, contradictions are regarded as sources for development. Thus, I argue that the core interaction pattern in ORCA is specifically designed to maintain and develop the task- and subsequent tool-competence among its users.

The ORCA software gives immediate alert-feedback. The operator tests possible stowage scenarios by changing the rotation. This is done in the Rotation list contextual menu, accessed by right-clicking a berth under a port in the Rotation list.

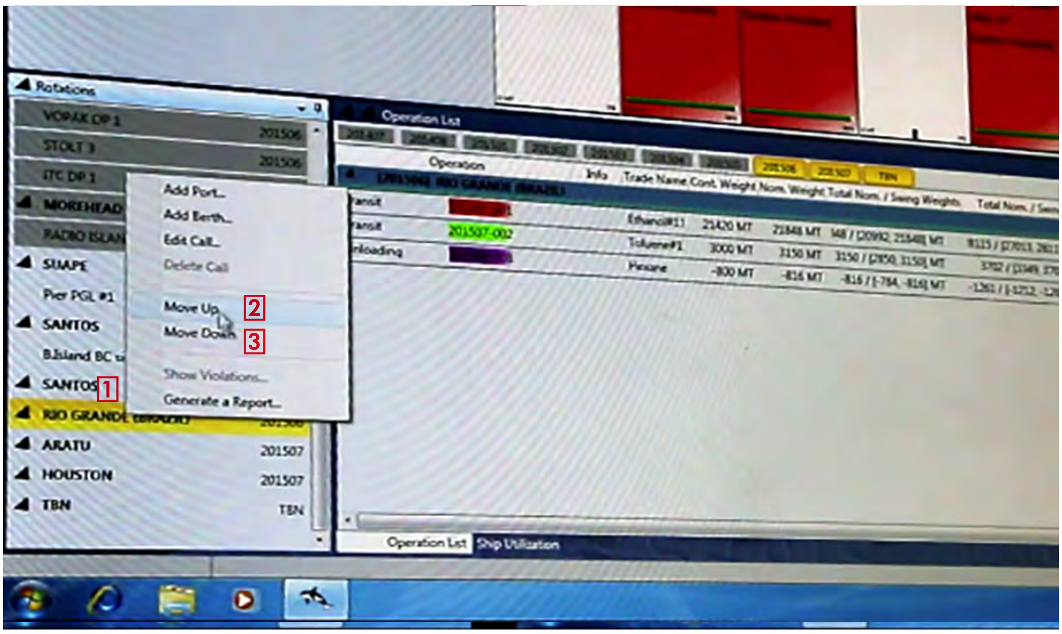


Figure 26. Contextual menu in Rotation List.

Moving the berths up or down, changing the order of the vessel's arrival at the berth. Changing the rotation is a sequential click-operation: 1. right-click on a berth to access the contextual menu [1] and 2. click either 'Move Up' [2] or 'Move Down' [3]. The operator describes:

> For example, I have a pack of (… (2) I can stow it *there* and put *that* package in *that* tank there. If I come *here* (Operator points to the screen element [Stolt 2][28] under Houston), and I'll drop this (cargo) in that tank, which I unload on the previous berth. I go to Stolt 2 first (right-click on [Stolt 2] to access contextual menu, then clicks 'Move up'). Then the program will show 'Illegal commingling' because I tried to load the tank before it had been unloaded.
>
> (1) Then we have to change that stowage, by moving the berth (downwards in the Rotation list, by clicking 'Move down' in the Rotation list contextual menu. The Rotation list contextual menu is accessed by the operation of right-clicking the related port or berth element in the rotation list).
>
> (2) Then I can just stow that tank somewhere else. [Clicks on the OTH berth that is placed beneath the Stolt 2 berth, and thus later in the Rotation list]

The operator explains how the program responds to a mistake, and warns about a mix of cargo, 'Illegal commingling.' Previously, the operator had to find out about commingling cargo manually. With ORCA, the operator can concentrate on other aspects of the stowage that can only be managed by a human, e.g. considering overrides of soft warnings. This is further elaborated in the discussion of the human-ORCA as a functional organ in ch. 8.6, p. 220.

Terminology: Rotation and Rotation list

A *rotation* is a sequence of ports, such as e.g. Texas City and Freeport, all in the US Gulf, continuing to South America after having finished all loading procedures in

28. Stolt-Nielsen Limited is a large Norwegian shipping company. Stolt2 is one of their terminals in Houston.

the Gulf, to e.g. the port of Aratu, and the port of Santos in Brazil, then further to down to Argentina. This is called the *rotation f*or the ship. Another rotation could also start from the US Gulf, continuing through the Panama Canal, and out into the Pacific, going to different ports in the East, which is another typical rotation. There are also trans-Atlantic rotations, when a ship goes from for example Rotterdam/Antwerp to e.g. some quays in Charleston on the east-coast of the US, before ending up in Houston. Then they bring some loads at various quays there, and then they go back to Europe. That could also be a rotation, in addition to being a *port rotation* since the boat loaded at several berths in a port.

A *rotation list* displays a sorted list of calls. The program manages the calls on two levels: on port level (level 1) and Berth level (level 2).

Port rotation

It is called a port rotation when there is a rotation within a harbour. For example, when a ship is coming to Houston, she could go to five, ten or fifteen quays in the same harbour. Then, it is important to rotate the ship around the harbour in an efficient way so that it does not remain berthed. Every stay in every harbour must be cost-effective. In order to streamline the stay, it is important to know how many loads and packages that will arrive at the various quays.

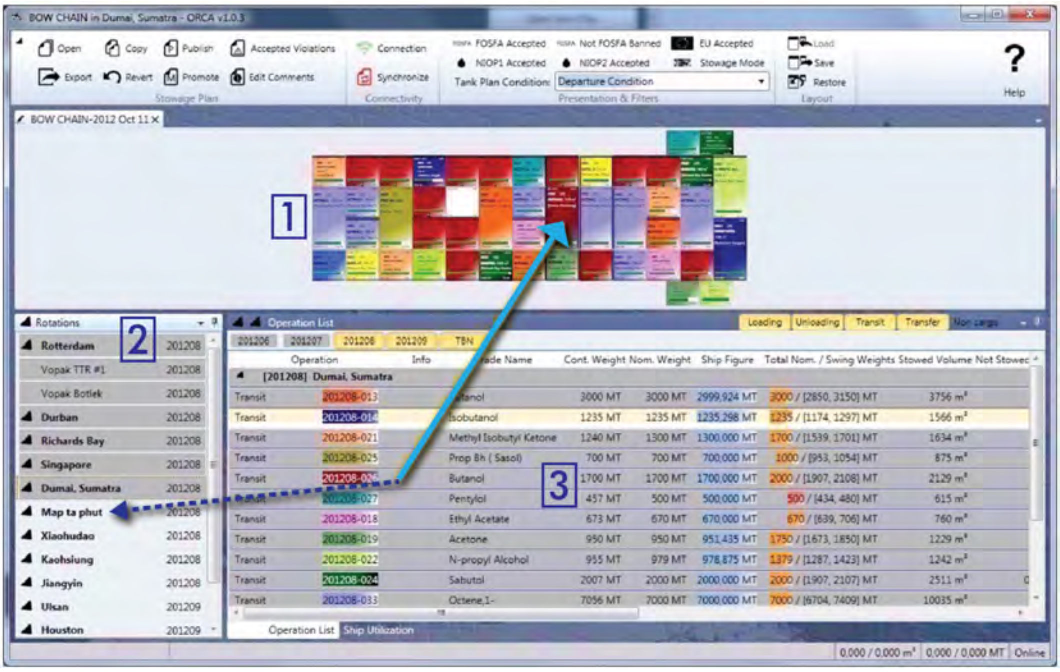


Figure 27. Tank plan in ORCA. Mediated representation of a vessel's tank structure, with drag-and-drop interaction directly unto the tank element [1], port element [2], and cargo representations in the operation list [3].

The Operation list [3] beneath the tank plan [1] following the direct feedback paradigm, has direct drag-and-drop functionality, both related to the Rotation list to the left [2] and to the tank plan above. The blue arrows are not part of the UI, but a marking made by the author to visualise the drag-and-drop activity done by the operator. The three operations of 'click and hold–drag–drop' constitute a 'filling tank'-action in ORCA.

Figure 28. Details of tank plan in ORCA.

The system provides direct feedback on each abstracted element: recipient port, weight, grade of filling, warning border for adjacent hazard, and so on. It supports drag and drop functionality that facilitates direct interaction in the GUI, thus maintaining support for an unhindered, established work flow.

During these observation sessions, the number of breakdowns appearing during the regular stowage processes is rather few and quickly solved. The immediate recognition of a screen artefact is present in the findings from the user observations; however, it would seem that the real intuitive character lies, not necessarily in this artefact recognition, but the connection to previous similar activity, as the foundation for the mental probing of operations and actions, and its distribution to the user interface as a perceived intuitiveness emerging from. Many of the operators abandoned the previous stowage tool, Othello, and resorted to stowing on paper. By being able to stow efficiently through the new system, an operator compares:

> It saves me a lot of time compared to making the stowage plan at one, or more, sheets of paper, with an eraser and... You save time, but must still pay attention. You still need to know what you're doing. Also, we save a great

> deal not having to wonder *whether* we can go to this and that pier. We can easily see that in the program. Very easy.

In a system, where virtually all interactions in the interface are based on probing/simulation, the ORCA user interface facilitates the externalisation of what could resemble a prolonged orientation phase, supporting this continuous simulation, and is based on a combination of recollection of previous manual processes that have been developed during years of stowage planning, and familiarity with artefacts. The degree of familiarity with the screen artefacts in the user interface, and the operations and actions they represent, lays the foundation for the internalisation undertaken in the user interface that leads to the experience of intuitive interaction.

This is supported by an example of the seemingly most logic cargo-operation of all: moving one cargo directly from one tank to another, by dragging a cargo from one tank screen artefact and dropping it on another. For a lay person, the cargo would just have been moved, but for a skilled, experienced operator the first tank must first be un-stowed, and then re-stowed in a new tank. This transfer action is what infra-structurally will happen 'beneath' the user interface as the result of a direct dragging and dropping between two tank screen artefacts in the user interface; a logic interaction that retains the vocational structure of how it was previously done with pen and paper. As an operator describes:

> It is registered that the cargo was loaded into the first tank, and then moved to another afterwards. If you're just going to move the stowage, you have to un-stow first, and then stow it again [in the new tank].
>
> For an operator that is logical. If you move the load, then you have, in a way, made a different stowage action than if you stow it straight into another tank.

Related to vocational practice, un-stowing a cargo from one tank and re-stowing it in another tank, is what an operator would have to do in order to move a cargo.

Thus, even if the program could have supported a simple, and seemingly intuitive, drag-and-drop operation in the user interface that moved one cargo directly from one tank to another while all the underlying functionality of un-stowing cargo from one tank and re-stowing it in another could have been handled by the program, it would have been counter-intuitive according to practice. The correct way of intuitively mediate this specific action, according to practice, would be to let it consist of these two operations, also in the user interface. This exemplifies how domain-specific experience could make the internalisation process more comprehensive in relation to established practice, and thus, with a certain degree of probability, laying the foundation for the operator to perceive the interaction as skill-based intuitive. This is an example of how intuitiveness might have two different meanings, in relation to task competency, in that the first, logical, approach would, instinctively, be a direct drag-and-drop operation, while the correct, and intuitive way according to practice, is to unload the cargo in the first tank prior to loading it in another.

7.3 Hierarchical structure of use activity in ORCA

How does one decide what something is if the borders between the different 'somethings' are unclear or indeterminable? How does one reason when deciding whether something is an activity or an action? Utilising activity theory as a framework for analysing observations of use activity, it may, at times, be unclear what an activity is and what an action is, and distinguishing a *complex* action and an activity may be significantly difficult. Kuutti states:

> The flexibility of the basic [activity theory] concepts makes them useful in describing developmental processes. On the other hand, it also means that it is impossible to make a general classification of what an activity is, what an

> action is, and so forth because the definition is totally dependent on what the subject or object in a particular real situation is (1995, p. 32).

This means that the hierarchical structure that activity theory uses to explain and discuss the relationship between mind and action is in continuous motion and leads to the understanding of what is perceived as intuitive actions as dynamic and contextual. Utilising activity theory concepts to understand how actions and operations can be perceived as intuitive, to get a better insight of which contexts might facilitate this understanding, and how it might inform can be done in practice.

As described in chapter 2.4.2, the border between action and activity is unclear (Kuutti, 1995; Davydov et al., 1983), as the structure of the use activity might change. While the objective of optimal stowage might seem ideal and fixed, the *character* of 'optimality'–what 'optimal' means–will change continuously, and development in itself would be an aspect of the activity. Therefore, the identification of the actions and operations at various levels of use activity, mediated by the user interface in ORCA, represents how they appeared and were perceived during the period of observation.

As described in chapter 5.3, p. 112, the stowage planning of a chemical tanker is a process by which the locations of cargoes on board a vessel has to be meticulously planned prior to loading in order to reduce the costs as much as possible. At the same time it is important to exploit the capacity of the vessel. This includes controlling a large set of parameters, such as compatibility, either with other chemicals or the coating of the variety of tanks on board, the cargo requires a specific temperature during transport, distribution of cargo on several tanks due to weight balance of the ship (trim & list).

The process of stowing a chemical tanker is associated with a significant degree of complexity, and sometimes critical complexity. Therefore, a chemical tanker must adhere to international rules and regulations. An elaborate description of the complexity of stowing a chemical tanker is presented in the case description in chapter 5.3. In this chapter, the stowage process is analysed according to the hierarchical structure of activity, on an individual operator level. Some illustrations used in the case presentation are reused as background illustrations with the operator's interaction with the user interface is visualised as a numbered sequence in an overlay on top.

	Activity outside UI	Interaction with UI
Activity Level ↑↓	Stowing a chemical tanker	Planning the stowage of a chemical tanker
Action level ↑↓	Filling tanks	Occupying tank elements in GUI
Operation level	Opening valves Attaching pipes Pushing physical buttons / Embodied interaction	Utilizing screen elements representing real artifacts

Figure 29. The stowage process as a hierarchical structure, with examples of activities, actions and operations, linked to the empirical material. Depiction based on fig. 2.4 in Kuutti (1995, p. 30).

Activity

<u>Motivation and object</u>**:** the main motivation, and the object of the stowage is the optimal or most [cost] efficient stowage, i.e. the optimal utilisation, of the ships tank configuration, in the form of a stowage plan.

The activity is characterised by *planning* the optimal or most [cost] efficient stowage, i.e. the optimal utilisation, of the ships tank configuration. A completed stowage plan is visualised in the software and 'published' as a 'full' tank plan. This is, however, a process that is continuously repeated as new contracts for transporting cargo is received and accepted, and the state of optimal utilisation of ship tanks changes and must be revised, making a new completed stowage plan that can be published, changing both the rotation and utilisation of tanks. This process of revision, as what would be an optimal stowage plan changes, and continues in an almost perpetual manner within the rotation cycles, which gives the activity the character of being in development, in that the object, the optimal state of tank utilisation, is not fixed.

Figure 30. Distributing cargo onto several tanks.

Example of action

<u>Goal:</u> identifying and loading, at the time of action, the right tank or set of tanks in the current configuration of a tank plan for a single cargo.

In order to achieve an optimally efficient stowage plan, e.g. the most efficient usage of the tanks available, the operator must find and choose the correct tank for that specific cargo. One example, in figure 30, of distributing cargo onto several tanks is e.g. *cargo order #018, Ethyl acetate (pink colour)*. In order to stow this cargo, the operator must move the cargo order, visualised by one line of text, from the operation list beneath the main tank plan artefact [1], by clicking, holding and dragging the cargo order artefact and drop it on the designated tanks by clicking on the target tanks in successive order [2],[3], until the cargo has been fully distributed. This is an action pattern that rather quickly became automated into an operation.

Example of operations

Identifying the correct tank for a specific cargo and subsequently loading the cargo in a single tank, or set of tanks, in the tank plan. This is comprised of a series of operations where the conditions are related to the vacancy, and requirements of the transportation of cargo, such as e.g. moving a cargo from the cargo list, by clicking, holding and dragging the screen artefact, in the form of a line in the cargo list, onto a vacant tank screen element [1] in the tank overview. (See figure 31).

Figure 31. Single tanks in the tank plan. User interface screen elements showing a vacant tank and an occupied tank in the tank plan.

Another requirement related to transportation of chemical cargo is a compatible tank coating, i.e. the cargo must not react with the surface in the tank. This tank metadata is checked by a constant system feedback [2], but requires an active check from the operator prior to initial loading of the cargo. There is no coating in tanks 11WP and 10WP in fig. 31, as the tank element in the user interface shows the metadata (SS); (SS) meaning 'Stainless Steel'.

Many dialog boxes in ORCA are accessed by a right-clicking operation. They might contain a rather substantial number of functional elements that the operator may interact upon. Inside a dialog box, the human-ORCA interaction will consist of one or a sequence of operations, that together, might be regarded as an action– until the sequence of operations has been practised into becoming automated, thus being transformed into an intuitive operation, as a part of the domain-specific best practice.

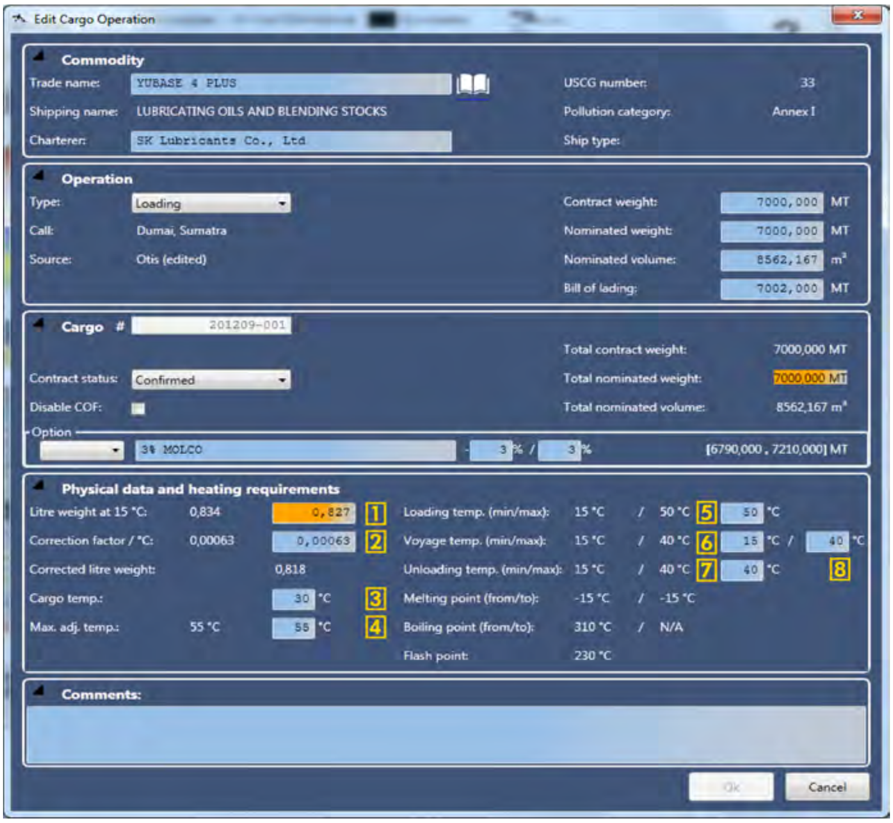


Figure 32. Edit Cargo Operation dialog box. Controlling and adjusting cargo physical data and temperature. For hierarchical dynamics, see fig. 33, p. 186.

After the cargo, in the form of the screen artefact represented by a visual line of text in the cargo list, has been loaded onto the single tank artefact, visualised by a vacant (colourless) tank screen artefact being coloured and filled with the primary textual metadata (fig. 31, p. 184), the operator must continue to plan and test various parameters related to the current cargo. This could e.g. include physical data and temperature control. The operator will check or edit the litre weight, and tank temperature according to the requirements related to the cargo, by right-clicking the single-tank artefact in the tank plan overview (fig. 40, p. 197), activating the pop-up dialog box accessing this set of metadata related to the cargo. Physical data and temperature are controlled by eight parameters [Numbers 1–8 in figure 32]. Many of them are pre-entered by the system, but the operator may edit them. At the time of observation, a rather short time after the program was released and installed on the operator's computers, early findings seemed to indicate that these particular use-activities had the character of being operations. Later observations showed that this sequence of operations, that initially would have the character of an action, became an operation, like many other similar operator-interface operations undertaken in dialog boxes. Thus, intuitive interaction would seemingly emerge during the transformation of a use-activity from being an action, comprised of a group of operations, into a single operation performed as one automated movement.

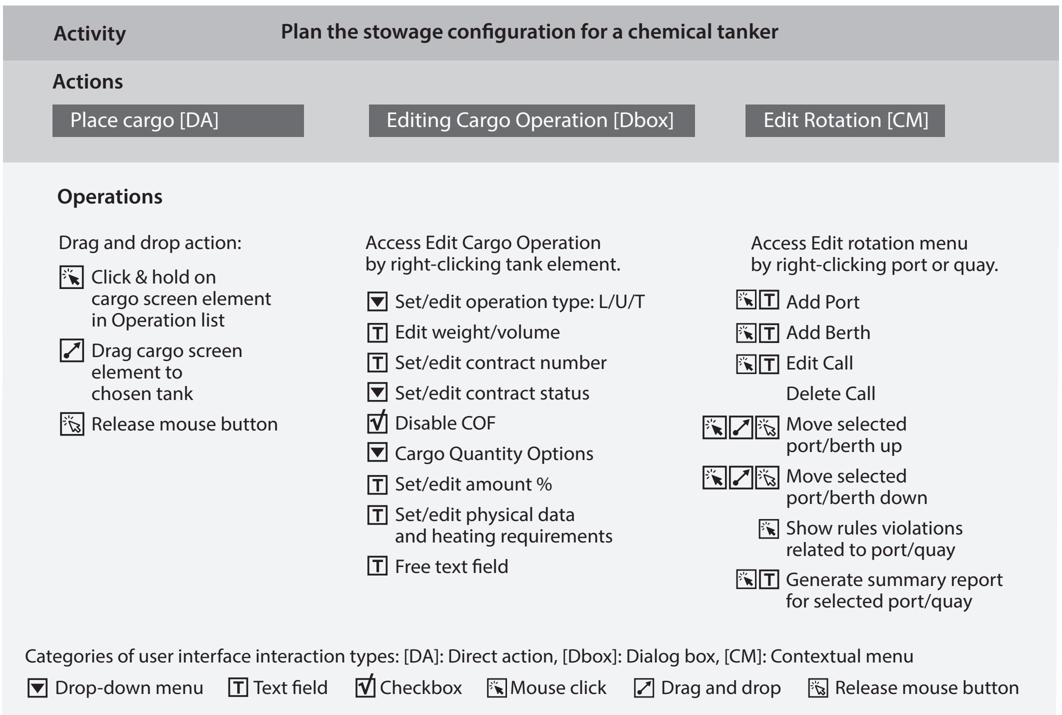


Figure 33. Activity structure. Examples of action <->operation dynamics related to user interface interaction in the three most common use-activities in ORCA: placing a cargo, editing cargo metadata and stowage parameters, and changing the order of ports and quays in a rotation cycle. Abbreviations: Operation type: L/U/T refers to Loading, Unloading and Transfer. COF stands for Certificate of Fitness.

According to Leontiev, an activity occurs to satisfy a motive, through which the objective can be identified, and that human activity, exists only as actions or a sequence of actions.

As presented in ch. 2.6, 'a computer application, from the user's perspective, is not something that the user operates on but something that the user operates through, towards objects or subjects of interest in the work activity. Human beings operate through computer applications, as well as other tools, on materials that they are turning into products with the help of others' (Bødker, 1989, p. 173), making the user interface just the mediating tool, and not the target itself, (Bødker, 1991), leading the user interface to become 'transparent', as in not itself being the goal of an action. However, exceptions would be in situations of breakdowns, when an operator using ORCA encounters a problem leading to a change of conditions that could have been caused by a variety of reasons, i.e. The testing of a possible stowage by placing a cargo in a tank-representing screen artefact, by dragging it from the operation list and receiving a system feedback. In the ORCA user interface this is often a warning, requiring the operator to adjust, either a condition for an operation, the goal of an action, or perhaps even the motivation for an activity, although the motivation that guides the activity of making a stowage plans might seem stable in its core. This would raise the question of whether making a stowage plan for a chemical tanker is a large complex action or an activity.

In the ORCA case, the overall objective is a completed stowage plan. The activity of making one consists of many sequential actions and operations e.g. the action of placing a cargo in a tank, or changing the order of port arrival, or moving a cargo from one tank to another, or 'meta' actions like e.g. customisation of the tool itself. All consist of drag-and-drop operations, and contextual menu operations that include clicking in UI elements like check boxes, accessing drop-down menus, editing temperature requirements, adjusting volumes and trim & list, or assessing whether the cargo is compatible with either the tank coating or the cargo in the adjacent tank. A complex action might be seen as an activity, or be perceived as both, by different people and at different times and in different contexts. The complexity that characterises the activity of making a completed stowage plan involves dealing with various contexts and changing conditions, but still has a final outcome as the object. Final until another cargo-operation in another port must be added, which involves changing both the tank plan and the rotation, where the need is adjusted, and the activity/action distinction is renegotiated and subsequently settled once again.

Leaning on Kuutti's statement of the flexibility of basic activity theory concepts, making them, on one hand, useful in describing developmental aspects of activities, while, on the other hand, seeing this flexibility as blocking the possibilities of defining a standard set of categories of what constitutes an activity and a clear distinction between an activity and an action, would lead these categorisations to become entirely contextual (Kuutti, 1995).

Therefore, I argue that it is the aspect of the final need – the overall motive, that defines when a use-activity becomes an activity, even if what an activity is and what an action is may interchange. In the end, because there will finally be an end, it will ultimately be the overall objective that would determine what would be an activity.

7.4 Chapter summary

This chapter has analysed and discussed the part of the empirical material that is related to the *use* of the ORCA *software.*

This chapter has presented the operators' general use-activity in the ORCA software and how the use of the ORCA user interface is situated within the hierarchical structure of activity and its connection to the discussion of specific actions and operations in the user interface. It presents reflections around the human-computer functional organ, and the perception of intuitive interaction that might emerge from the development and transformation process that can be identified in the action-operation dynamics in the hierarchical structure of activity.

CHAPTER 8.
Mediation in ORCA

This chapter analyses and discusses ORCA as a mediating tool. First the ORCA user interface, and all the most frequently used screen elements are presented, together with the interaction pattern connected to each screen artefact. Followed by a discussion of the user interface familiarity and transparency, in an immediate, momentary approach to interaction, before connecting user participation during the development phase, and turning to utilising the ORCA user interface in a dialectical approach to solving contradictions. As a mediating tool, the computer and the human form a functional organ.

The empirical material from the ORCA case indicates that the user interface transparency described by Bødker occurs primarily on the operation level in the hierarchical structure of activity, and that the visibility or conscious reflection of the user interface is varying with the complexity of the actions, i.e. the awareness of the user interface is increasing with the increasing structure of compound actions and associated complexity.

Norman states that PCs are complex and hard to use (Norman, 1998), and one might get the impression that, by spending a disproportional amount of time and resources in finding one's way in and around the user interface, that it is the user interface itself that, unplanned, becomes the objective by being unnecessarily complex. This might be problematic since it is not the user interface that is the objective of the activity. Bødker (1991), however, states that the user interface is not the objective for our actions and operations, and is, as such, merely a tool one is utilising to mediate the activity in order to reach the objective. According to Bødker, one only works through the interface, towards a goal, resulting in the user interface becoming transparent, i.e. out of focus of the user. This requires that the actions and operations are known and that it runs smoothly, i.e. without breakdowns. However, breakdowns are possibilities for development and may even be desirable in some contexts. Breakdowns may be linked to the object of the activity that, for instance, when assumptions change, which is in line with Kuutti's statement that activities are unstable, making the use-activity unpredictable and situational (1995). There may also be conditions related to the mediating tool itself, in this context, the user interface, which will force the user to stop, and reflect on which operations and actions that are both available and adequate. In the context of making a stowage plan, as it appears in the empirical material, such breakdowns may be e.g. that the cargo from an accepted transport request, does not reach the quay(berth) at the right time, and that plans must be adjusted. According to Suchman (2010), situated actions would have the character of workarounds, fulfilling the objective that was originally planned, but undertaking it in using a different approach. During breakdowns, the object of the activity remains, but the use-activity will have the character of situated action mediated by the user interface, which temporarily, 'looses' its transparency and moves into the operator's focus, until things are 'settled', meaning that the user interface is 'transparent' during settled, regular, uninterrupted use and becoming visible only when contradictions occur; that it is through these contradictions, when use-activity is interrupted and becomes unsettled, it is being developed, and that the development emerges through trial and error. A user interface designed for users to easily test and simulate; controlling mistakes by 'embracing' mistakes, making them visible, contributes to the transformation of actions and operations, and the development of both the subject and the object.

Kaptelinin et al. (1995, p. 190), state that human agency through mediation is an important aspect of activity theory as a theoretical framework, providing the conceptual tools needed in order to analyse how people relate to e.g. software, and also how, through adherence to 'rules' and participating in the 'community', a 'subject's knowledge becomes extended.

Mediation, in an activity theoretical HCI context, comprises an additional relation; as a tool with which the user can combine. The user interface being an extension of the internal plane can externalise and distribute actions in the mind, thus influencing the levels of competence.

In this chapter, the empirical material regarding the abstracted screen elements in ORCA is presented. This is followed by the presentation and discussion of externalisation as being a part of the functional organ, and linked to a mediated action perspective to affordances (Kaptelinin and Nardi, 2012b).

The ORCA user interface mediates operations and actions in a sequential structure. Thus, findings indicate that intuitive interaction emerges as an event between the operator and the user interface, that is as a result of a development process, as opposed to the common focus on momentary interaction in HCI–research and practice. According to Kuutti and Bannon, a substantial amount of HCI research has been centred around 'momentary and ahistorical HCI situations that are not crucially connected to a particular time and space; the focus is on the snapshot of the interaction at the moment' (Kuutti and Bannon, 2014, p. 3543).

8.1 The ORCA user interface

The user interface consists of five main screen elements. The user interface follows a gradually established modern interaction pattern, with drag and drop functionality between the operation list and the tanks in the tank overview, and right-click access of contextual menus for detail editing for each tank element.

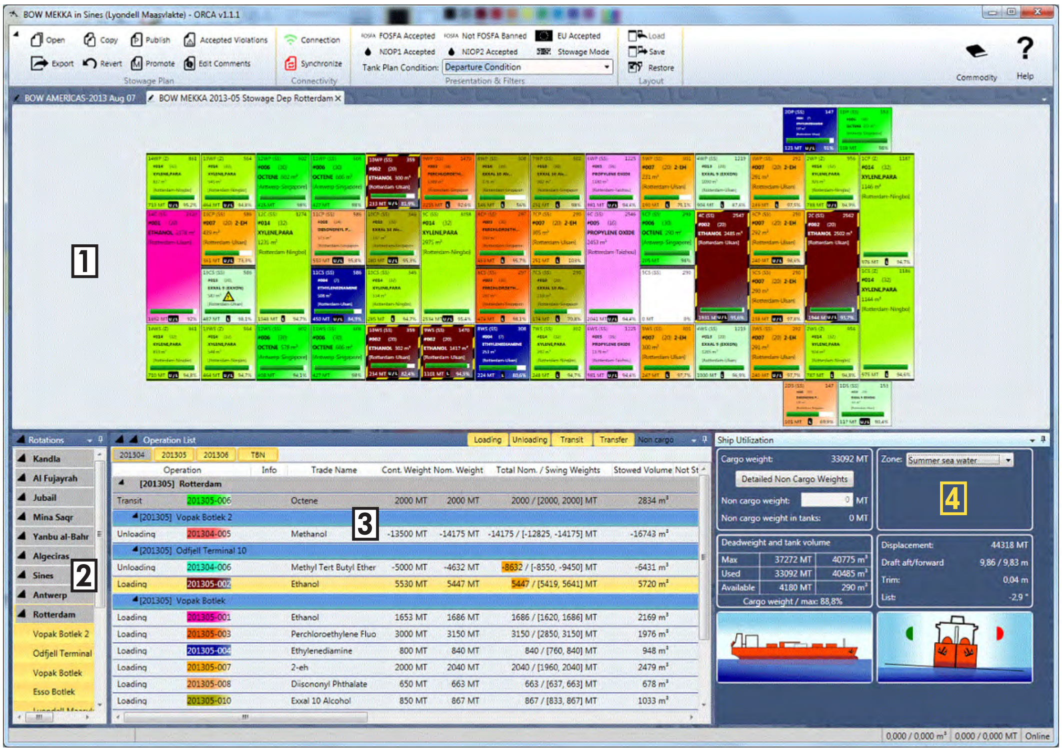


Figure 34. The user interface consists of a top menu and 4 sub-windows: Tank Plan [1], Rotations [2], Operation List [3], Ship Utilization [4]

8.1.1 Screen area

The main window in the user interface contains the five main screen elements. The user interface follows a gradually established modern interaction pattern, with drag-and-drop functionality between the operation list and the tanks in the tank overview, and right-click access of contextual menus for detail editing for each tank element. The size and the position of the Rotations, the Operation List and the Ship Utilisation are configurable by the user. The position of the windows can be changed by clicking on the top of the window and dragging it. Small but clearly visible screen artefacts mark the change process while it is undertaken.

8.1.2 Top menu

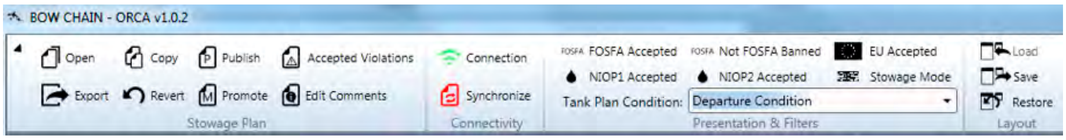


Figure 35. The top menu shows non-cargo functionality of a stowage plan.

The top menu shows main non-cargo functionality of a stowage plan, like open, copy, publish, export and so on, in addition to synchronisation between an operator/cargo broker and maritime personnel, and viewing filters based on regulations, mode and categories of condition (cargoes in transit, loading or unloading).

8.1.2.1 Stowage plan

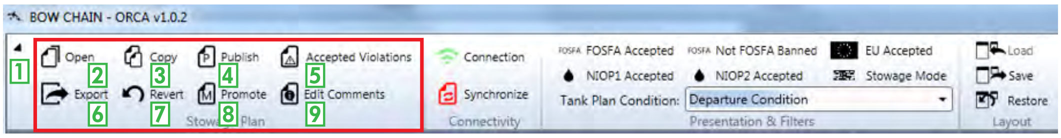


Figure 36. Stowage plan area of the top menu

1. The small black arrow opens and hides the menu at the top.
2. Open: opens a new or stores plan.
3. Copy: copies the open plan as many times as you click 'Copy'.
4. Publish: publishes a plan in the common database if the program is online at the office, or exports into a file if offline. Every plan, or copy of a plan, is regarded as just a preliminary plan, i.e. sketch, test, simulation
5. Accepted Violations: displays all accepted soft rule violations.
6. Export: saves a stowage plan in a local or shared folder.
7. Revert: rolls back to the originally open stowage plan.
8. Promote: same as publish. 'Promote' will promote the published plan to the Master Plan.
9. Edit Comments: displays and edits comments entered by an operator or any other person having modified the current plan.

8.1.2.2 Connectivity

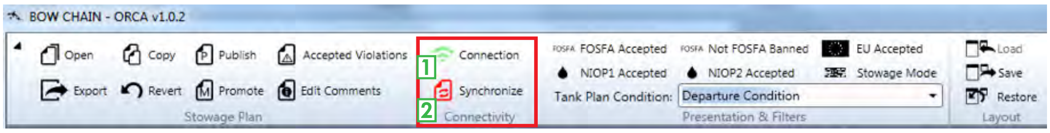


Figure 37. Server connection management in top menu.

1. Connection shows the connection status to the server if connected to the office (only to be used by operators). A green colour is displayed if connected, black if not connected. Clicking on 'Connection' disables the connection if already connected, or tries to connect if not connected.
2. Synchronize retrieves the latest data from IMOS[29], if connected, and updates the current stowage plan automatically. A red colour is displayed if the current plan is not synchronised with the latest data from Otis, black colour is displayed otherwise.

8.1.2.3 Presentation and filters

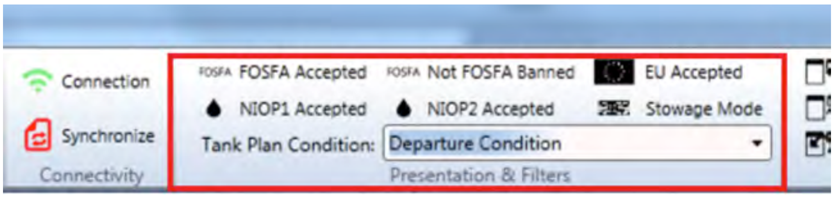


Figure 38. Presentation filters for visualisation of tank plan overview.

1. FOSFA – EU–NIOP[30] filters: highlights the tanks based on the chosen filter. Several filters can be combined.
2. Stowage Mode: highlights only the tanks that can be connected to a given operation. The operator must select a [cargo]operation in the operation list to activate this filter in the tank plan.
3. Tank Plan Condition: two tank plan conditions can be selected (at arrival or at departure). In addition, tank plan filters can also be selected (operations in call, loading cargoes, cargoes in transit and unloading cargoes).

29. IMOS: Integrated Maritime Operations System (Veson Nautical)
30. FOSFA: Federation of Oils, Seeds and Fats Associations. EU: European Union, NIOP: National Institute of Oilseed Products

8.1.2.4 Screen layout management

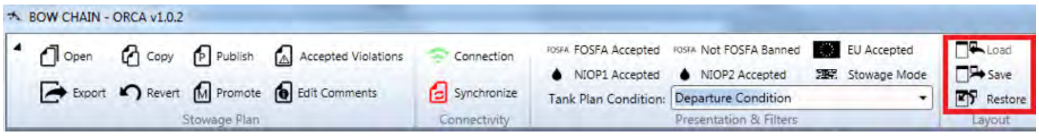


Figure 39. Operators can configure and save their own custom screen layouts by clicking on 'Save' in the Layout section of the top menu. The standard, 'factory settings' can be restored by clicking on 'Restore' in the Layout section of the top menu.

8.1.3 The tank plan

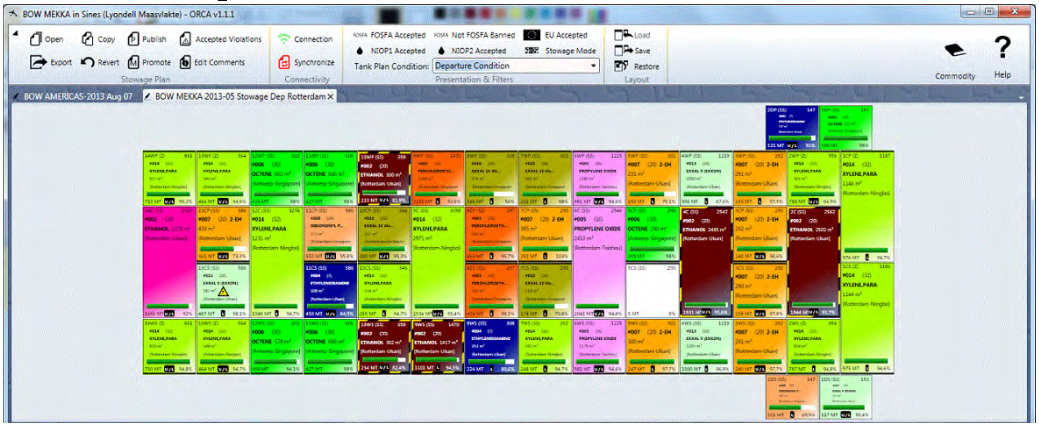


Figure 40. Tank plan overview. The tank plan presents abstracted overview of all the tanks and their location on the ship. The single-tank elements display the content of the tanks based on the selected operations in the Rotations.

If several operations are selected, the tank plan will display a summary of the operations for the selected tanks (using the L (Loading), U (Unloading) or T (Transfer) letters in each tank where a [cargo]operation is assigned). Visual marks are added to the tank elements under certain conditions: Arrival Condition, if the Arrival Condition is selected in the tank Plan Conditions drop-down list, Filter Active, if a filter is selected in the Tank Plan Conditions drop-down list. Invalid Plan if a hard rule violation is generated in a non-departed call. The cofferdams[31] are represented with an emboldened tank limit.

31. A Cofferdam on a ship is an empty compartment between tanks to prevent two different liquids from mixing.

For each tank, the following information is presented:

a. Tank name and coating
b. 98% filling volume (in m^3)
c. Cargo weight in tank (in MT) or 0 MT if no cargo
d. Filling ratio (in %) or 0% if no cargo

In addition, if a tank contains a cargo, information related to the cargo information is also displayed. This information is customisable by the Content Presentation in the tank contextual menu (see figure 41).

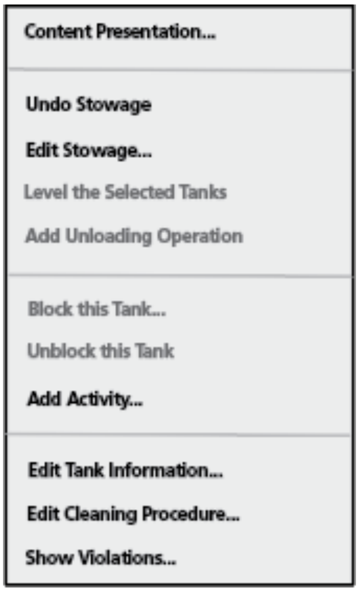


Figure 41. Tank plan content presentation menu

8.1.4 Rotations

A rotation is a sequence of harbours, and their quays (berths), between which one specific ship sails, either during a specific period of time, or an agreed-upon number of ports. This is visualised as the Rotations palette screen artefact where the port and quays are represented by palette sub-elements (figure 42).

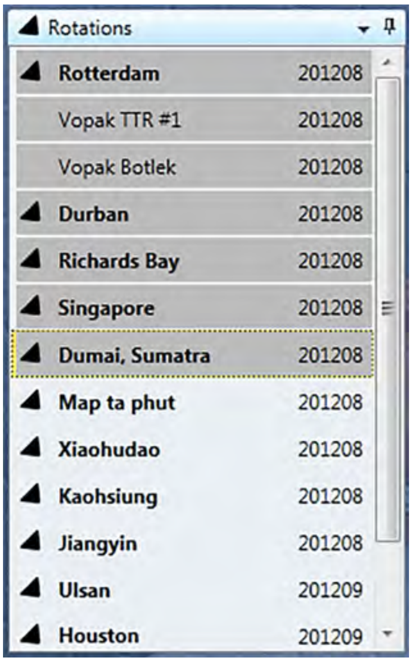


Figure 42. The Rotation list shows the route, called Rotation, that the ship will sail, and also the list of ports and berths included in the current stowage plan.

The Rotations screen element displays the list of ports, and berths included in the current stowage plan. Ports and berths can be added and deleted from this window. Various reports can be generated from this list by right-clicking on a port or a berth. Ports and berths can be moved up or down in the rotations by right-clicking and selecting ‘Move Up’ or ‘Move Down’. Ports and berths can be edited by right-clicking and selecting ’Edit Call…’ (see fig. 43, p. 200). A TBN call is added at the end of the rotation list (in a TBN voyage). It will include all the operations that are not yet assigned to a given call.

Figure 43. The contextual menu for the Rotations list is invoked by right-clicking a port or berth in the menu.

This contextual menu contains the following options:

1. Add Port: adds a new port in the rotation list.

2. Add Berth: adds a new berth under a port.

3. Edit Call: displays and edits calls details.

4. Delete Call: removes a call without any cargo-operations. A port can only be deleted if it has no berths.

5. Move Up: moves a call upwards in the rotation list.

6. Move Down: moves a call downwards in the rotation list.

7. Show Violations: displays violations present in the call.

8. Generate a Report: offers a set of various PDF reports to the user.

These use activities are undertaken as sequences of single operations, first by right-clicking on a port or berth in the Rotations window to invoke the contextual menu, and then click on the relevant menu item (numbered 1–8 above). Common procedures would e.g. be editing a rotation, where changing the port order is done by right-clicking the port that the operator needs to rotate (change order) and then choose 'Move Up' or 'Move Down'. The operator can add a new port by [a]: right-click a port in the rotation list – [b] click 'Add Port...' – [c] write the name of the port, voyage number that needs to be changed. Within a port, the operator can add a new berth by right-clicking on the port where the berth needs to be added, and write the name of the berth and any relevant comments. Both ports and berths can be edited, by right-clicking on the port or berth screen element, and choose 'Edit Call…'. The operator then makes the changes and click 'OK'.

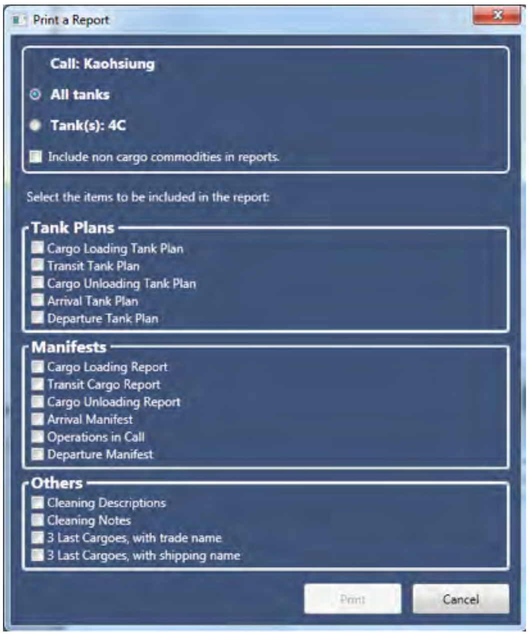


Figure 44. The Contextual menu for Rotation List settings.

In situations when a cargo has to be transferred from one tank to another, in another port or at another berth than the cargo was loaded, the operator must first chose the cargo, and then chose the tank from which the cargo has to be moved out of, and drag and drop it in another tank. Both the tanks will now display a 'T' to indicate that the cargo has been transferred in this call.

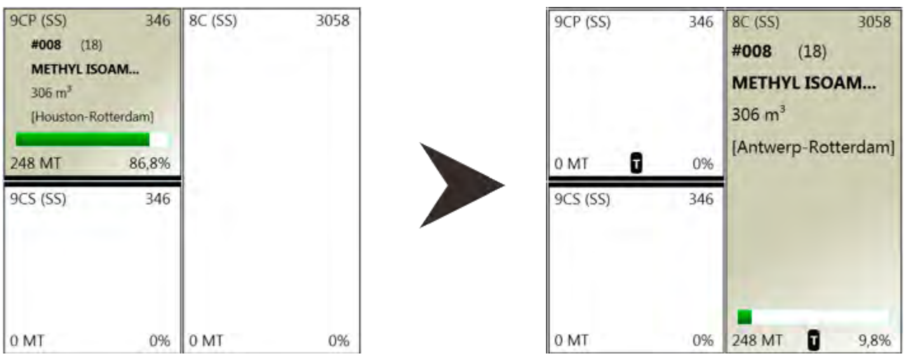


Figure 45. Before and after successful transfer of cargo from tank number 9CP to tank number 8C.

A new [cargo]operation is created in the operation list, with the type 'Transfer'. Clicking on this line will highlight the two cargo tanks. If the operator wishes to undo this operation, he can do this by right-clicking on the line representing the transfer operation, then select 'Delete Operation', and confirm the deletion, or while still at the same selected call, drag and drop the cargo back to the original tank.

Figure 46. Dialog box for generating a report for selected call.

The operator can choose to generate a report for all tanks, i.e. the complete tank plan, or filter on arrival or departure state of the tank plan. In addition, the operator can filter on cargoes that are loading, unloading or in transit if the report is required to only include information about specific or selected tanks.

Additionally, the menu allows non-cargo commodities to be included in the report. The operator would finally specify the reports to be generated. Several reports can be generated at the same time. In this case, they will be included in the same PDF file. The operator must click on 'Generate'. The requested report(s) will be generated, and the user will be asked where to save the PDF file.

8.1.5 The [cargo] operation list

The Operation List displays the operations in the selected calls in the Rotations. Operations can be filtered based on their types by clicking on the Loading/Unloading/Transit/Transfer/Non-Cargo buttons. The filter is active when the button is pressed (yellow).

Figure 47. Cargo operation list - Overview screen element. Each line in the Operation List can be dragged and dropped onto a tank element in the tank overview. This operation marks the tank element in the tank plan as occupied, and 'fills' it with cargo content.

Cargo Operation, which is related to all parameters connected to a selected cargo. Operations related to the editing of these parameters are divided into the following categories, accessible through the screen artefact of a dialog box (see fig. 49, p. 204):

The displayed columns can be configured by right-clicking on the column headers, see figure 48.

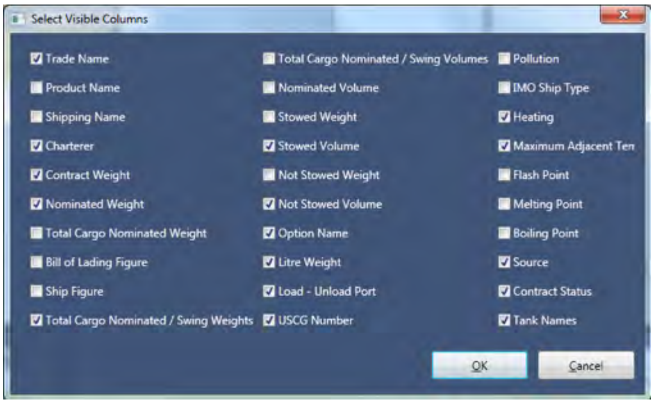


Figure 48. Contextual menu for the Operation List settings. All editing is binary since operations are restricted to ticking on or off choices, represented in the user interface by checkboxes.

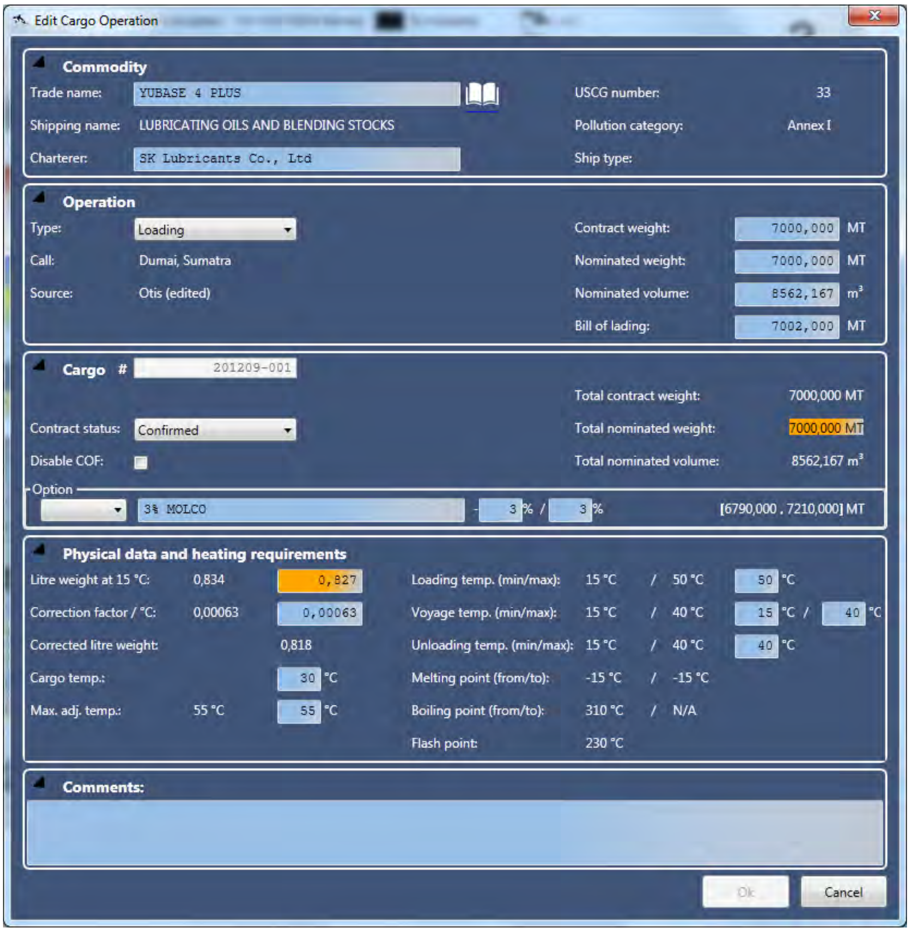


Figure 49. Cargo Operation editor. Mediates detailed editing of the Tank plan, such as contract status, weight, temperatures in tank and adjacent tanks and so on. This screen element is accessed by right-clicking a single tank element.

- **Commodity**. This category provides administrative information about the chemical cargo, such as 1. trade name, imported from the Commodity book, which is an external database where all chemical cargo is registered. 2. Shipping name, name of category under which the cargo belongs and is shipped. 3. charterer, 4. compatibility number in the US Coast Guard compatibility chart, 5. pollution category, 6. ship type.

- **Operation data.** This category provides information about operation type, whether the cargo is to be loaded or unloaded (op>ui)[32] the call (location), the source of operation details, the agreed and nominated weight and volume to be transported (op>ui), and the actual volume that was loaded (op>ui) .

- **Cargo contract meta data.** This category provides information about cargo meta data, such as cargo contract number, contract status (pending, confirmed, etc.) total contract weight, nominated weight and volume.

- **Cargo physical data and heating requirements.** This category provides information and editing capabilities related to temperature requirements, e.g. litres weight at 15 °C (op>ui), correction factor divided by temperature (op>ui), setting cargo temperature (op>ui), max temp adjustment (op>ui), minimum and maximum loading temperature (op>ui), minimum and maximum voyage temperature (op>ui), minimum and maximum unloading temperature (op>ui), temperature range for the melting point, temperature range for boiling point, and flash point.

The 'Edit Cargo Operation' dialog box appears by right-clicking the cargo screen artefact in the operations list (see fig. 47, p. 203). In order to set operation type, the

32. (op>ui) indicates that this screen element is *partly* operator-controlled, i.e. data is retrieved from a database but can also be edited by operator interaction.

operator must click the 'Type' pop-up menu in the Operation category, and choose either Loading or Unloading from the menu.

The observed human-interface interaction in this dialog box consists of single sequential operations that collectively would constitute an action.

The displayed columns can be configured by right-clicking on the column headers, see figure 48.

Columns can be moved in the operation list (fig. 47, p. 203) by dragging and dropping the titles in the column headers. The first line of the Operation List contains a list of voyages. Voyages can be selected by clicking on their respective buttons. To select several voyages, operators may either (1) click on a voyage and then control click to add another voyage to the selection, or (2) click on the first voyage, hold shift and then click on the last voyage or TBN. Selected voyages appear with a yellow background colour. Calls from the selected voyages are automatically added in the Rotations. The black arrows (triangles showing or hiding cascading menus) before a call can be used to hide operations in a given call. Hiding operations for a port does not hide operations in its berths. The black arrows in the window menu can be used to hide operations in all ports (first arrow) or all berths (second arrow). When an operation is selected, the tanks linked to this operation are highlighted with a yellow and black border in the tank plan.

Figure 50. Contextual menu for the Operation List, accessible by right-clicking the [cargo]operation element.

This menu contains the following options (numbers corresponding to figure):

1. **New Cargo:** adds a new manual cargo and cargo-operation in the Operation List.
2. **Edit Operation:** displays and edits information about the operation.
3. **Delete Operation:** deletes an operation. A loading operation cannot be deleted if deleting it results in more unloading weight for this cargo than loading weight. In practice, unloading operations must be deleted (or adjusted) before loading operation can be deleted.
4. **Split Operation:** splits a cargo-operation in two cargo-operations of the same type. A stowed operation cannot be split.
5. **Add Loading Operation:** adds a new loading operation for the selected cargo. A new loading operation can only be added when selecting a transit operation.
6. **Add Unloading Operation:** adds a new unloading operation for the selected cargo. A new unloading operation can only be added when selecting a transit operation.
7. **Undo Stowage:** removes all the links to the tanks for the selected cargo-operation.

8. **Edit Bill of Loading:** displays and edits the B/L figure.
9. **Edit Cleaning Procedure:** edits the cleaning procedure for all the tanks where the selected cargo-operation is stowed. The cleaning procedure can then be adjusted tank by tank in the tank plan.
10. **Add an Activity:** adds a non-cargo activity.
11. **Show Operations for this Cargo Only:** enables a cargo filter in the Operation List. Only operations related to the selected cargo will be visible. The filtered cargo number will be displayed in the first line of the Operations List window, beside the list of voyages.
12. **Show Operations for All cargoes:** if the cargo filter is enabled in the Operation List, disables the cargo filter.
13. **Show Violations:** displays violations caused by this operation.
14. **Generate a Report:** the same command as the Rotations context menu.

8.1.6 Ship utilisation

Figure 51. The Ship Utilization window visualise the degree of utilisation of the ship for the currently selected call. (Numbers in the figure correspond with the numbers in the following paragraph.)

This part of the user interface displays information about the total weight on-board, i.e. total cargo weight, total non-cargo weight, total non-cargo weight in cargo tanks . For ships with trim/list data enabled, it is also possible to specify the detailed non-cargo weight per tank and store for each non-cargo tank or group of non-cargo tanks **[1]**. The operator can set the meteorological zone, which is used to determine the max deadweight **[2]**. A table summarising the deadweight and tank volume, with max used and available values in addition to total ship utilisation (cargo weight/max deadweight) [3], and displacement, trim, list and draft information [4], with a visual representation of the ship's balance and position in the water [5].

8.1.7 Two examples of non-cargo specific functionality

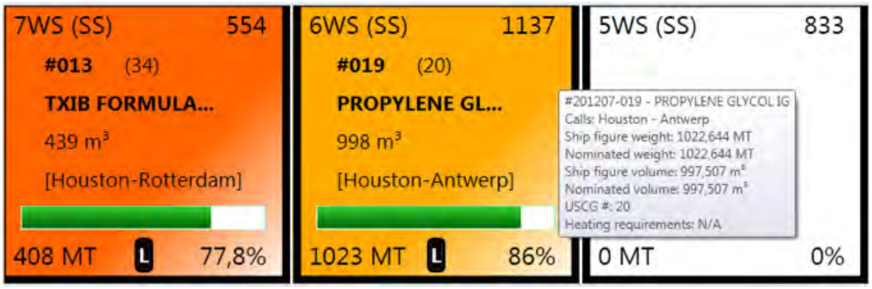


Figure 52. Tooltip functionality. Displaying additional information on single tank elements in the tank plan, becomes visible when moving the mouse over a tank. Tooltip functionality by hovering is an operation in the hierarchical structure of activity.

The tooltip functionality in ORCA is used both to provide extended information of a single cargo-operation in the tank plan element (fig. 40, p. 197), and as clarifying hover.

Another common use-activity connected to the rotation cycle, since it is a procedure that is undertaken while at port, is e.g. tank cleaning, which is a procedure undertaken between the cargo-operations of unloading and next loading. This is a part of a normal cargo-operation procedure, and will be an operation within an unloading action.

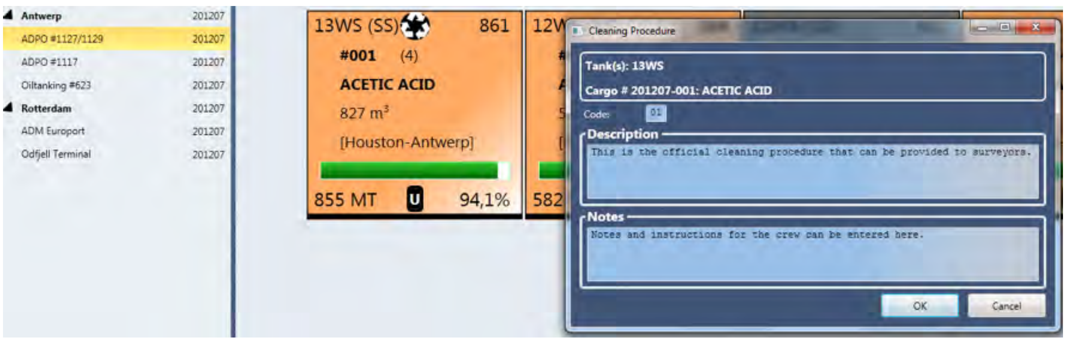


Figure 53. Dialog box for cleaning a tank after it has been unloaded.

Here the operator must select a call in the Rotations, and set the 'Tank Plan Condition' to 'Unloading cargoes' (black U-symbol in the bottom of the tank screen element), then right-click on the cargo tank to be cleaned and click 'Edit Cleaning Procedure'. In the Cleaning Procedure dialog box, the operator must enter a cleaning code, which will appear in the cleaning procedure tank plan report. It is also text fields for writing the official cleaning procedure in 'Description', which can be printed and given to e.g. surveyors, and for writing the instructions to the ship's crew in 'Notes', which can also be printed in a separate report.

The various sequences of operations in the user interface may constitute an action, or the sequence of use-activites, as a whole, might be perceived as an operation, depending on the complexity of choice, or whether these operations together and with experience have become one automated operation.

8.2 User interface familiarity

The non-commonsense, experience-based, notion of intuitiveness might be situated within the activity theoretical framework, by leaning on Leontiev's idea of a structural similarity between internal and external processes [Leontiev (75)[33] in, and translated by, (Kaptelinin and Nardi, 2012a, p. 19)]. It would seem analogous to the familiarity concept of e.g. Raskin (1994) and Blackler et al. (2005, 2011), describing humans' ability to recognise similarities, and exploit this ability to fill in the gaps in a setting or context that would seem familiar but not identical, leading to a functional intuitive flow.

The analysis of the findings in chapter 6 and 7 is grounded on Kaptelinin and Nardi's descriptions of mediating affordances (Kaptelinin and Nardi, 2012b), and Naumann et al.'s discussion on *intuitive use* rather than the interface being intuitive

33. Kaptelinin & Nardi, 2012, reference the russian version of Activity, Consciousness, and Personality from 1975, while this dissertation, of natural causes, references the english translation from 1978.

(Naumann et al., 2008). The overall view from findings and conclusions in the most recent and current research on intuitive use of user interfaces (Blackler and Popovic, 2015; Blackler et al., 2011; Hurtienne et al., 2008; Mohs et al., 2006a; Naumann et al., 2008), could indicate intuitive use through *recognition of metaphors* which, in the goal of human interactivity in user interfaces in a domain-specific context, i.e. skilled worker environment. In combination with the recollection of operations undertaken in similar contexts and based on what a human intends to do, this is laying the foundation for intuitive perception of functionality, thus contributing in how computer mediated activity is designed and perceived.

An observed feature of experienced operators' use of ORCA was that although they ran into problems on the *operation* and *action* levels, they did not suspend *activity* as a whole, even if the objective had to be temporarily adjusted. When the user interface did not offer the immediate functionality for what seemed to be the proper execution of an operation, thus causing a breakdown, the experienced operators circumvented the problem by adjusting the order of operations within an action by testing in the user interface. As expert users they probed and circumvented their way around breakdowns, as situated actions, an ability originating from previous actions of what activity theory would describe as operationalisation, described by Bødker as 'the dimension that allows the user to form repertoires of operations through which the instruments are handled, on the one hand, and to consciously reflect on dealing with the components on the other' (Bødker, 1991).

The capability, of the internal, intellectual plane exceeding the possibility of the external in-the-world-activity, aids the learning process that is a requirement for the level of knowledge and experience necessary to perceive operations and actions as intuitive and immediate, which rests on domain-specific competence, i.e. a skilled worker context. The notion of intuitiveness can also apply to an action, where the sequence of operations, as an action, has turned into an automated operation. Findings indicate that the perception of interactions as intuitive does not necessarily happen through momentary interactions, but emerges through the transformations of reflective actions into automated operations.

8.3 User interface transparency

As presented in ch. 3, a limited amount of research has been done on intuitive use of user interfaces. One reason for this could be that the concept of intuition has been regarded as vague and ambiguous (Naumann et al., 2007), and therefore means different things to different people. For regular people, non-academics, thesauruses and dictionaries, 'intuition' means gut feelings and a sixth sense, and knowledge without prior learning. Within the field of human-computer interaction, it is regarded as rather uncontroversial to assert that intuition is based on experience. Knowledge, which, in turn, will be the grounding for recollecting previous processes as part of the task competency of knowing the possibilities and limitations of a task, that are necessary in forming a human-computer functional organ.

The special character of the computer user interface, in being able to mediate such a large variety of use-activities, and a subsequent high degree of complexity points towards a continuum in which there is a transition between when a user-interaction is reflected upon, as an action in an activity theoretical sense, which might also include a conscious interaction with the user interface as a technical reflection on the mediating tool, and towards the user interface becoming transparent.

How a user interface might facilitate the handling of this kind of complexity is e.g. discussed by Kuutti from an activity theoretical perspective, in which he states: ' … at the activity level, information technology can form the matrix of activities; it can be the principal enabler for an activity. At least two different possibilities can be identified. First, information technology may make an activity possible and feasible (e.g. by linking the participants by a network or system). Second, information technology may enable an activity to have an object that would otherwise have been impossible to grasp" (Kuutti, 1995, p. 35).

	Activity outside UI	Interaction with UI
Activity Level	Stowing a chemical tanker	Planning the stowage of a chemical tanker
↑↓		
Action level	Filling tanks	Occupying tank elements in GUI
↑↓		
Operation level	Opening valves Attaching pipes Pushing physical buttons / Embodied interaction	Utilizing screen elements representing real artifacts

Figure 54. The stowage process as a hierarchical structure, with examples of activities, actions and operations, linked to the empirical material. Depiction based on fig. 2.4 in Kuutti (1995, p. 30)

Together, groups of single operations form actions, which together, subsequently, constitute an activity. The mediating layer is an abstraction of any socio-culturally mediated entity, but is, in an activity theoretical HCI context, which Kaptelinin has renamed computer-mediated activity (Kaptelinin, 1995b), where humans and mediating tools are integrated into functional organs (Kaptelinin, 1995b; Leontyev, 2009). This means that user interaction in a user interface may require the user to possess a certain amount of knowledge, i.e. to have some insight or expertise, but that the interface aids the user in solving tasks that would have been difficult or impossible to perform without the emancipating power of the user interface. In ORCA this is exemplified and visualised in the system's alert feedback functionality, like form variables on UI elements i.e. tanks with a bold outline implying a double hull meaning that otherwise incompatible cargo types can be loaded next to each other, or pop-up windows warning about illegal or non-viable actions.

8.4 Mediated action perspective on affordances

The concept of affordances (Gibson, 1979), in a human-computer interaction context (Norman, 1999), can be seen as a built-in help functions, and screen elements in the user interfaces of the socio-technical world; i.e. the technological device in front of them at any given time. This, 'inscribing optimal action' into an artefact, has helped and guided members of the general public in their approach to, and subsequent accept and understanding of 'computing machinery'; which, in the era of computer-mediated activity, i.e. since the invention of the graphical user interface (Engelbart, 1962; Kay and Goldberg, 1977; Sutherland, 1963).

In ORCA, this is visible in how the system was developed; by the close collaboration between users and developers, and a focus on knowledge transfer made possible by co-location. The developers would understand how the interface in ORCA could absorb the users' work habits by, on a continuous, daily basis, having access to the users' knowledge pattern repository of how operations and actions should be carried out and inscribe these patterns – these best practices – into the screen elements of the user interface, leading them to become what Kaptelinin and Nardi describe as mediated affordances (Kaptelinin and Nardi, 2012b), and 'verified' by participation. The guiding principle of affordances lies in their facilitation of recollection of a process through recognition of artefacts. That the character of the form of the artefact, of which some, the vocation-specific ones, are based on domain-specific knowledge, while others, generic, standard user interface elements in modern computing, have been adopted by growing up in and being a member of a culture where computers have become ubiquitous.

Affordances are representational 'clues', that through their visual character and shape, explain how the objects to which they belong, are to be used. A central aspect of the user interface elements is *use*, as mediated use-activity. If an affordance, as a 'crystallised' instruction of use, represents an instruction for use, it must also contain an element of facilitating intuitive use of user interfaces, since the comprehension of the affordance, then, stems from knowledge that is derived from the accumulated repository of culturally and domain-specific mediated actions, that is, from practice. This concurs with Bødker, who states that, 'The user interface cannot be seen independently of the goal or object, or of the other conditions of the use activity.' (Bødker, 1991, p. 141). This supports the argument of a task or activity-based understanding of what constitutes intuitive use of user interface elements; making intuitive interaction by imitating previously known physicality possible, and by that, familiarising it, by recognition and contextualising physical actions, letting users remember or associate previous behavioural patterns and processes into the mediating artefact, as internalisation tools, which is how the ORCA user interface is designed.

A ship-officer stated: 'I recognise "actions" in the GUI,' explaining that the various screen elements in the user interface, made him remember *previous* and similar operations and methods when making judgments in the *present*. This is supported by e.g. Leontiev's statement of specific modes, operations and methods being crystallised in the mediating artefact (Leontiev, 1978), and as mediated affordances (Kaptelinin and Nardi, 2012b).

As presented in ch. 3, intuitiveness stems from experience that the user perceives something as familiar (Blackler et al., 2005; Blackler et al., 2014; Naumann et al., 2008; Naumann et al., 2007; Raskin, 1994). This requires that one remember previous experience, meaning that an event within the action or operation level in the hierarchical structure of activity, is based on recognising artefacts triggering the recollection of 'process'. Recollection of process as a prerequisite for intuitive interaction leans on the task- and tool competency required for forming a functional organ. The perception of intuitive interaction with a user interface may be expected to emerge, as the use of specific software in a work context, like e.g. the ORCA system, which operators, as participants of a skilled worker environment, would use every day in their work. The use-context would be characterised by domain-specific competence, and a shared knowledge pattern repository.

In e.g. Blackler et al. (2005, 2011), familiarity is related to experience with products with similar features, while in Hurtienne et al.'s studies (Hurtienne et al., 2008; Hurtienne, 2009; Hurtienne, 2017; Hurtienne and Israel, 2007) of the role of image schemas as representations of interaction patterns, familiarity relates to the interaction with the physical world and how it facilitates an embodied understanding of the world around us (Johnson, 1987). Findings among the naval personnel, presented in ch. 8.6.7, suggest that the 'metaphorical' recognition of artefacts, as familiar, would benefit from encompassing familiarity as a recollection of prior processes, based on skill and experience. This is in line with Dreyfus and Dreyfus, who claim that an *intuitive understanding* must come from previous education and accumulated experience (Dreyfus and Dreyfus, 1986). In relation to intuitiveness, as an *immediate understanding* of what to do, findings in ORCA showed that both experienced and not-so-experienced operators worked with a certain fluidity. Even though the experienced operators worked slightly quicker, supposedly due to having had the time to acquire more skills, thus relying less on clues in the user interface, the less experienced operators also worked with a certain flow, although slightly less. Image schemas parallel mediated affordances, where they, in an operation, trigger experiential recollection of a previous process, i.e. that a present process is not solely based on recognition of the artefact, but that the mediating artefact triggers the recollection of previous action making the internalisation process intuitive. This seems to be analogous to the observations of operators' use of ORCA, which indicate that interaction with the ORCA interface might be perceived as intuitive either by [1] through the affordances in the mediated artefact, and [2] by relying on domain-specific competence based on recollection of previous actions, or a combination of means that could be a part of what Bødker and Klokmose (2011) describe as an artefact ecology.

8.5 Artefact ecology and mediated artefacts

Concurrent research on intuitive use of user interfaces (Blackler and Popovic, 2015; Blackler et al., 2011; Hurtienne et al., 2008; Mohs et al., 2006a; Naumann et al., 2008), and findings in the empirical material from this case study indicate that we might find grounds for intuitive use through recognition of metaphors (ch. 3.7.1, p. 80), as elements in the artefact ecology, in the goal of human activity.

Rather than concluding that the cause of what may be perceived as intuitive action through a user interface is triggered by mediated artefacts alone, or by recollection of previous operations and actions alone, the findings in the study suggests that it is the totality, as e.g. Bødker and Klokmose (2011) call artefact ecology that trigger what would appear as intuitive action. This creates a diversity of circumvention possibilities when human users experience breakdowns facing technically mediated tools such e.g. a software user interface, and that it is the diversity of the artefact ecology that would facilitate the variety of approaches that might give users the perception of e.g. performing an operation as something as intuitive.

There are several mediators in the process of transporting chemicals at sea. One is the software tool used in the planning process, another tool that is needed in order to achieve the agreed goal, is the transporting vessel; the third and fourth mediators in the stowage processes in a shipping company operating chemical tankers are the rules governing allowed cargo and placement of shipment in relation to each other (see the USCG Compatibility Chart in appendix 1.), and the organisation itself, providing the platform into which all other mediations are collected, of which facilitating the possibility of collecting the operators' knowledge repository into a common, vocational best practice provides the base for employing operations and actions with expertise. It is, however, the user interface as a mediating tool and part of the human-computer functional organ, being the centre which facilitates the primary activity of stowage planning, as well as being the mediator of learning and development, as a computer-mediated activity that is the main objective of this dissertation.

The vocabulary, tasks and processes related to the skilled worker context described in the empirical material, first utilised by working with pen and paper, later in Othello, and continued in ORCA. The abstracted set of symbols, i.e. the artefacts that first would inhabit the user interface of Othello was further developed in ORCA, while representing the same experiential knowledge.

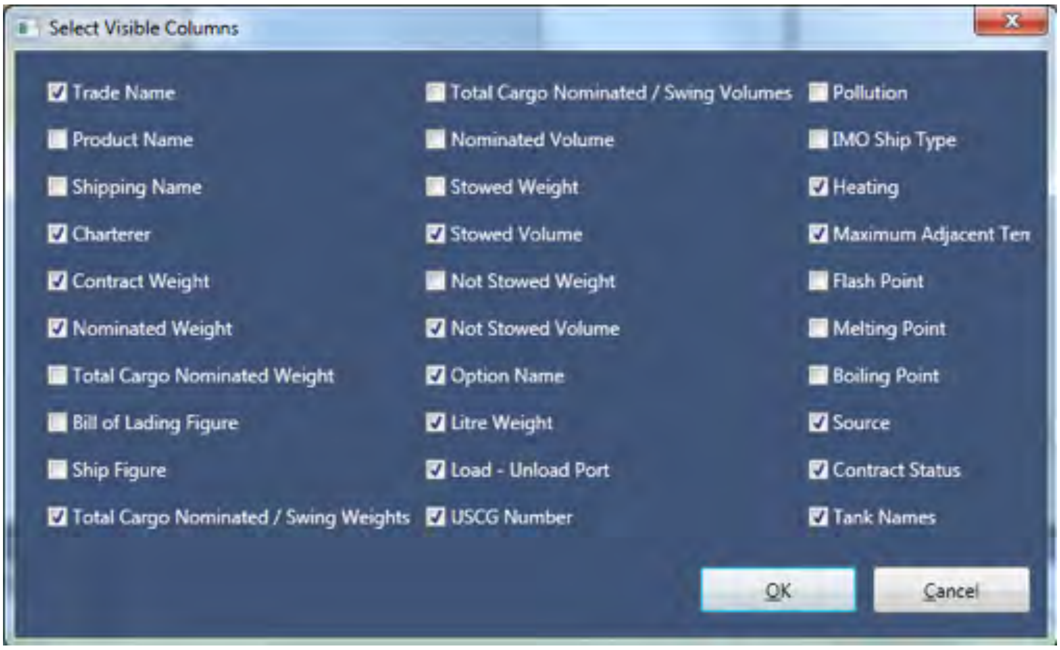


Figure 55. User-customisation by setting preferences in a dialog box. Interface elements are excluded or included as to suit individual task flow. Accessing the dialog box is an operation, setting the preferences will, during practice, transform from an action into an operation. This is a direct action operation

The notion of artefact ecology in ORCA includes being able to individually adjust the mediating capabilities to fit the patterns in the operator's knowledge repository. One example of this would be K. customising the user interface. She could adjust the interface elements of a cargo entry in the Operation list by invoking the contextual menu where the user interface could be adjusted to the optimum workflow relative to her individual level of skill and competence by setting the user interface she preferred, to show all the modules she required, at a glance without having to scroll back and forth. The operator can then change the order of how the selected cargo parameters are visualised in a direct action operation, see fig. 61, p. 235. The main operating procedure follows the standard modern drag-and-drop design pattern. This type of direct action stems from Ivan Sutherland's work on the Sketchpad (Sutherland, 1963), and later conceptualised as a click-and-drag paradigm by e.g. Jef Raskin (2000). The research work by Douglas Engelbart at Stanford Research Institute and later Xerox PARC (Engelbart, 1962; Levy, 1995), was instrumental in the development of the modern GUI and, and subsequently the tangible interaction discourse in exploring prior experience in spatial, embodied and engaged seamless human-tech-interaction (Dourish, 2004; Hornecker, 2012; Hornecker and Buur, 2006; Ishii, 2008). Also, there are aspects of learning and development related to the customisation functionality, see ch. 8.8.

8.6 Human-ORCA interaction as a functional organ

Internalisation is, within activity theory, presented as transforming external processes into processes undertaken in the internal plane, and is described as simulating use-activities, and its potential outcome, by mentally testing actions and operations for a possibly optimal outcome, without actually carry them out in reality. Kaptelinin and Nardi define internalisation as "a process during which phenomena external to the subject, both physical and social, become both individual and internal", and describe this procedure as a dynamically, bidirectional redistribution of functional elements with the likely transformation of both the internal and the external processes, as the individual learns and develops (Kaptelinin and Nardi, 2012a, pp. 16-17).

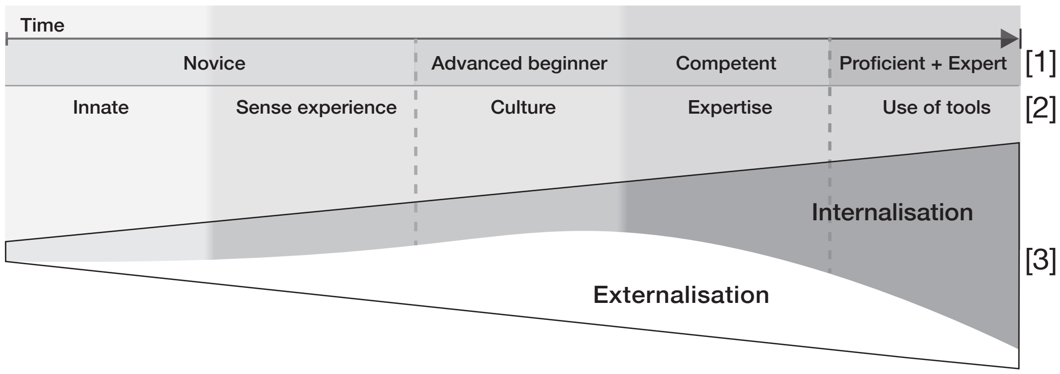


Fig. 62 (copy from p. 237 for reading convenience)

Both internal and external processes contribute as the individual transforms and develops. In figure 62, p. 237, (copied here for convenience), Engeström's model of expansive learning [3] (1987) is visualised together with Dreyfus and Dreyfus' skill acquisition matrix [1] (1986) and Mohs' knowledge category sources [2] (2006a).

As presented in ch. 2.4.3, Zinchenko (1995), discusses the mediation of a human-tool relationship that emerges through extensive and longitudinal interaction to such an extent that the interaction dissolves into an intuitive singularity; i.e. that the human and the tool melt together through a "master's introspection", and contributing to the experiential pattern repository, consisting of both previous actions and their representative metaphors that we can recognise as similar or related artefacts and their related contexts and recollecting previous praxis.

Kaptelinin (1995b) views computer tools, accessed by a user interface, as an extension of the internal plane of action. By utilising the user interface in ORCA, which was developed primarily as a simulation and planning tool, supporting a process that activity theory refers to as the orientation phase (Leontyev, 2009; Kuutti, 1995), the operators could distribute, the processes undertaken on the internal plane of action into the user interface, using the user interface as a facilitator for "sketching".

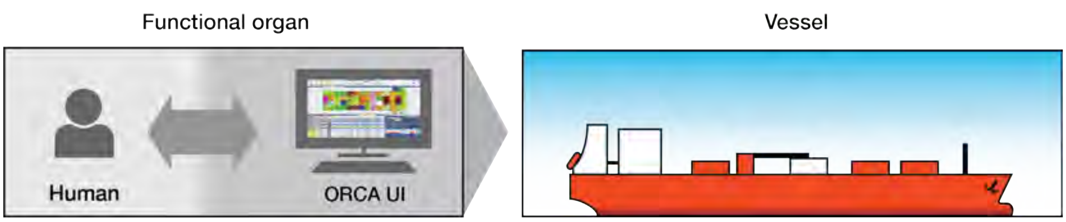


Figure 56. The operator and the user interface forming a functional organ, letting operators plan and test stowage outcomes that, without the ORCA user interface, would have been possible.

By situating the user interface tool as an extension of the internal plane of action, it also becomes, what in activity theory is defined as, a functional organ (see ch. 2.5, p. 48), needed in the activity of stowing vessels, where the user interface's externalising capabilities encompass the functionality of 'easing the burden' of internalising possible scenarios - in the mind. Conceptually, this is not radically different from what Douglas Engelbart wanted to achieve with his inventions of computer tools in the augmentation of the human intellect project, predating the Xerox Alto and Xerox Star (Engelbart, 1962).

Mediated artefacts, that is, the functional screen elements in the user interface, mediate the mental simulation in the operation- and action-levels, thus becoming an extension of the internal plane of action. Externalisation, thus, is the process of transforming the ideas, thoughts, goals, and possibilities; elements of the internal plane of action, into objects and fulfilled objectives in the real world, and lies at the very heart of human nature, and so laying the foundation for human culture. Virtually all human activity is, in one way or another, mediated.

Findings may seem to show ORCA as an enabler for the distribution of the mental probing in the mind, by the user interface's mediating power, enabling the externalisation of the mental simulation of a series of operations and actions, which is in line with e.g. Kuutti (1995), and Kaptelinin (1995b, 2015).

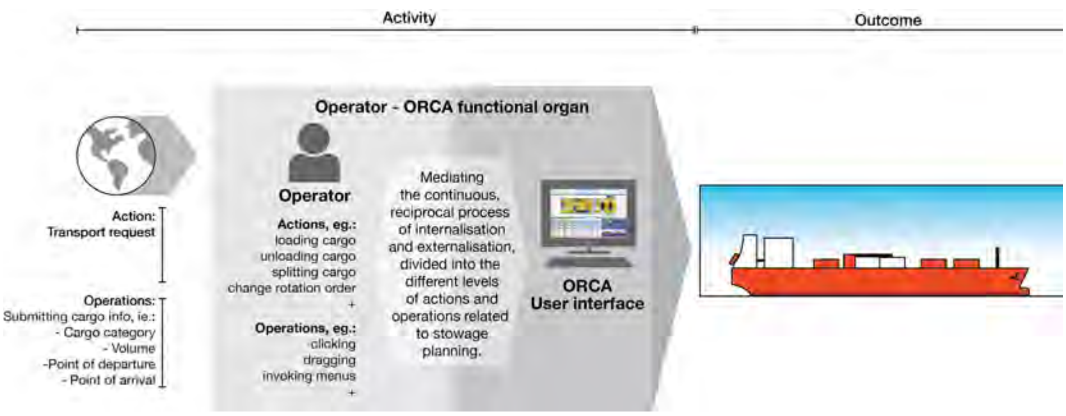


Figure 57. Human-ORCA functional organ related to levels. The functional screen elements in the user interface, mediate the mental simulation in the operation- and action-levels, thus becoming an extension of the internal plane of action, forming a functional organ.

The functionality of mediating the *planning process* is a central aspect of the user interface in ORCA as a 'special' mediating tool, encompassing the possibilities of both sequential probing and artefact recognition. In this specific 'computer-mediating' context, the user works with the user interface, by forming a functional organ, using it as a probing tool. The operators are mediating the planning process by a sequence of situated operations and subsequent actions in order to test whether something works before a final decision is being made, in ways that would not have been possible without the ORCA program, and as a mediating tool of an activity where the user works through the user interface towards an objective, as shown in figures 56 and 57.

A computer model is an abstraction, and might simultaneously represent a physical object in the real world. Thus, during computer mediated activity, the user interface mediation can have the character of being both internalising and externalising, and that this reciprocal 'flexibility' distinguishes user interface mediation from other types of mediation; by having an added directional relation by being directed both inwards towards individual operations and actions [fig. 58:1] and outwards towards an overlying goal, in the real world [fig 58:2] (Kaptelinin and Nardi, 2012a).

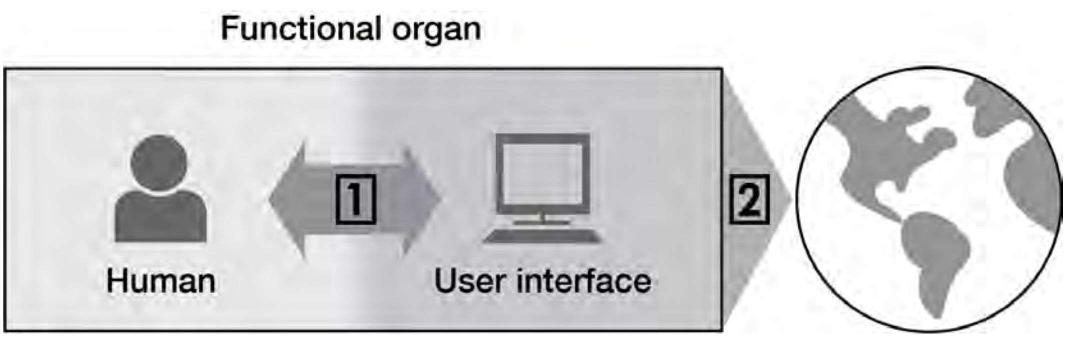


Figure 58. The "duality" of the functional organ is expressed by the subject-user interface mediation within the functional organ [1], and then between the functional organ and the real world [2].

The ORCA user interface allows for externalising thoughts and plans made in the internal plane, and it is how the cognitive action on the internal plane is facilitated by the formation of the functional organ, including its mediating concepts and symbols in the user interface, which determines the perceived intuitiveness in human-ORCA interaction. The ORCA interface's ability to 'absorb' and visualise the work made by the user on the internal plane, supports the integration of the human-ORCA functional organ. Whether the mediating concepts and symbols in the user interface aligns with the user's thoughts and ideas related to practise, seems to have a significant influence on the integration of the human-ORCA functional organ, as functional organs are dependent on the user interface representing a task- and tool competence on the human side. A well-integrated human-user interface functional organ will, or a least facilitate, a human-interface interaction that could be perceived as functional and intuitive.

Thus, the character of internalisation in the human-ORCA functional organ is based on domain-specific skills, and from an activity theoretical HCI perspective we could see this activity as a chain of operations and actions that have been practised sufficiently enough to become intuitive, and also reflected in how the user interface facilitates the externalisation of the processes in the mind, as the very extension of the internal plane stated by Kaptelinin (1995b). In the domain specific context of skilled practice, experience and recollection of previous use-activity provides the foundation for the tool- and task competence necessary for an innate understanding of limitations and possibilities of the mediating tool which forms the basis for intuitive actions in the user interface.

8.7 A small survey - findings "at sea"

As described in ch. 4, data collection could not be planned in advance in an ideal manner due to the character of the work: stowage planning includes processing transportation requests that are received at an arbitrary rate, combined with the observation premises being located in another city, reachable primarily by plane. This fact, and to collect data from maritime personnel, with experience from a different stowage tool, a small survey was conducted in February 2016, which could contribute to the data already collected.

Intuitive use relates to prior knowledge, reducing the cognitive workload during work activities. The QUESI model measures the subjective elements of intuitive use, and was developed by the IUUI group based on the groups research approach to intuitive use of user interfaces. Signs of intuitive use as this is defined by the IUUI group is based on a user's subconscious application of prior knowledge that leads to effective interaction (Mohs et al., 2006a; Naumann et al., 2007). This survey model has been tested towards a variety of interactive systems. i.e. websites, navigation systems and mobile phones, with more than 500 users evaluated (Hurtienne et al., 2009, p. 3).

The questions in the QUESI survey are divided into 5 subcategories: subjective mental workload, perceived achievement of goals, perceived effort of learning, familiarity, perceived error rate. The QUESI survey was used as the model for the ORCA-survey, however, since the context of use was different from the generic setting of the QUESI survey; the use context in this thesis was a skilled worker environment, and the mediated artefact that was being utilised was a user interface for a specific information system. Therefore, the set of questions were duplicated, in which the second set, activity was linked to the ORCA interface and the respondents perceived vocational experience. The correlating values are shown in the plots in figures 59 and 60.

A simple correlation between part 3, that was related to the questions 1–14 - on the sense of intuitiveness based on the respondent's general computer competence, and part 4 was made on the whole group, on the subgroups W-E, showing the compared variations in the pattern for intuitive interaction among the participants in fig. 60.

Sub-groups	Question no.	Characteristics
W - Perceived mental workload	1, 6, 11	1. I can use the program without thinking about it 6. The program is not complicated to use 11. I barely must concentrate on using the program
G - Perceived achievement of goals	2, 7, 12	2. I achieve what I need by using the program. 7. I am able to achieve my goals in the way I expect. 12. The program helped me recognise what I had to do to completely achieve my goals.
L - Perceived effort of learning	3, 8, 13	3. The way the program works is immediately clear to me. 8. The program was easy to use from the start. 13. It is clear to me straight away how the program is used.
F - Familiarity	4, 9, 14	4. I can interact with the programin a way that seems familiar to me. 9. It is always clear to me what I have to do to use the program. 14. I automatically do the right thing to achieve my goals.
E - Perceived error rate	5, 10	5. I can easily fix problems if they should occur. 10. The process of using the program goes smoothly.

Table 7: The questions and corresponding numbers distributed on subgroups. The answer scale ranged from 1:'Fully disagree' to 5: 'Fully agree.' See appendix 2, part 3.

Sub-groups	Question no.	Characteristics
W - Perceived mental workload	15, 20, 25	15. The ORCA interface uses the same 'images' as I am used to in the regular stowing process, therefore I can use the program without thinking about it. 20. Since ORCA works the way we usually do when stowing the ship, the program is not complicated to use. 25. Since the program operation simulates regular stowage task flow, I must barely concentrate when using ORCA

G - Perceived achievement of goals	16, 21, 26	16. Because of my experience with regular stowage task flow I achieve what I need by using the program. 21. The ORCA interface uses the same 'images' as I am used to in the regular stowing process, therefore I am able to achieve my goals in the way I expect. 26. Because of my experience I could detect it when the program helped me recognise what I had to do to completely achieve my goals.
L - Perceived effort of learning	17, 22, 27	17. ORCA works the way we usually do when stowing the ship, therefore it is immediately clear to me the way the program works. 22. Since ORCA works the way we usually do when stowing the ship, the program was easy to use from the start. 27. Because of my experience with regular stowage task flow, it is clear to me straight away how the program is used.
F - Familiarity	18, 23, 28	18. Since the program operation simulates regular stowage task flow, I can interact with the program in a way that seems familiar to me. 23. Since the program operation simulates regular stowage task ow, it is always clear to me what I have to do to use the program. 28. The ORCA interface uses much of the same 'images' as I am used to in the regular stowing process, therefore I automatically do the right thing to achieve my goals.
E - Perceived error rate	19, 24	19. If problems occur, I can easily fix them, in the same way I fix problems that occur during a regular stowing process. 24. Because of my experience with regular stowage task flow, the process of using ORCA goes smoothly.

Table 8: The questions and corresponding numbers distributed on subgroups in table 5 extended with aspects regarding the understanding of the user interface connected to previous experience and recollection of stowage operations. The answer scale ranged from 1:' Fully disagree' to 5: 'Fully agree.' See appendix 2, part 4.

Figure 59 shows the variation of perceived intuitiveness while working with the ORCA user interface. By correlating between interaction independent of previous experience (x-axis) and intuitive interaction based on previous stowage experience (y-axis), the trending pattern towards interaction with a user interface being perceived as slightly more intuitive when related to previous experience.

Figure 59. The total average values interaction with the ORCA system, correlating between interaction independent of previous experience (x-axis) and the perceived intuitive interaction based on previous stowage experience (y-axis). The plot indicates a slight increase in perceived intuitiveness when the participants were asked to relate user interface interaction with previous experience.

The range in age among the participants was 24–53 (n=25, one participant did not answer the question about age), while the range in years of experience ranged from 0 (just started right after finishing education) to 15. Correlating between age and experience, one anomaly emerged: one respondent reported current position as Master, with 10 years of experience, and being 24 years of age. This respondent reports current position as being Master; being a vocational term that is presumably not the result of a typo. Being Master concurs with having 10 years of stowing experience, which leaves us with the assumption that this respondent has made a typing mistake when answering the question about age. This respondent's answers to the other questions in the questionnaire have been retained since the analysis does not take age into account, but only the participant's perception of the interaction with the user interface in ORCA based on stowage experience, and not. The reason for this is that almost all participants in the survey (n=20) have equal or equivalent education, university college or naval university college, as their highest education, while two participants' highest education was a university degree and one participant had a degree from naval school. Two respondents did not answer the question about education.

The plot of the total score findings in figure 59, p. 228, shows that on average, a significant number of participants' perception of intuitive interaction range from slightly above 3.00 (neutral), and with the majority of the findings up towards 4.00 (mainly satisfied), which would indicate that the perception of interaction as intuitive was slightly higher, i.e. the users' perception of interaction was slightly more pronounced towards having an intuitive character, when related to previous stowage experience.

When distributed on subgroups W-E, the average number in the findings range from 3.26 to 4.01 on questions 1–14 in table 9, while average numbers on questions 15–28 in table 10 are shown to range from 3.35 to 3.70. This shows that, apart from subgroup G, perceived achievement of goals, the numbers from questions 15–28, where the answers are given based on the participants reflecting on previous experience, follows the trending pattern of the total score shown in figure 59 (p. 228), and shows a more pronounced perception of being able to interact intuitively through the ORCA user interface. The findings also show a greater degree of homogeneity and a smaller window of variation in the answers on questions 15–28 in table 10.

Subgroup	Characteristics	M	SD	Min	Max
W	perceived mental workload	3,320512821	0,458859748	1,00	5,00
G	perceived achievement of goals	4,012820513	0,05875097	1,00	5,00
L	perceived effort of learning	3,371794872	0,270144968	1,00	5,00
F	familiarity	3,525641026	0,256089543	1,00	5,00
E	perceived error rate	3,269230769	0,435142635	1,00	5,00

Table 9: Average findings distributed by subgroups, questions 1–14 in part 3 in the questionnaire.

Subgroups	Characteristics	M	SD	Min	Max
W	perceived mental workload	3,358974359	0,32711156	1,00	5,00
G	perceived achievement of goals	3,692307692	0,038461538	1,00	5,00
L	perceived effort of learning	3,576923077	0,341853631	1,00	5,00
F	familiarity	3,705128205	0,197369286	1,00	5,00
E	perceived error rate	3,423076923	0,271964147	1,00	5,00

Table 10: Average numbers distributed by subgroups where the findings are related to the participants reflecting on previous vocational experience, questions 15–28 in part 4 of the questionnaire.

Comparing the findings distributed on subgroups, the following patterns emerge:

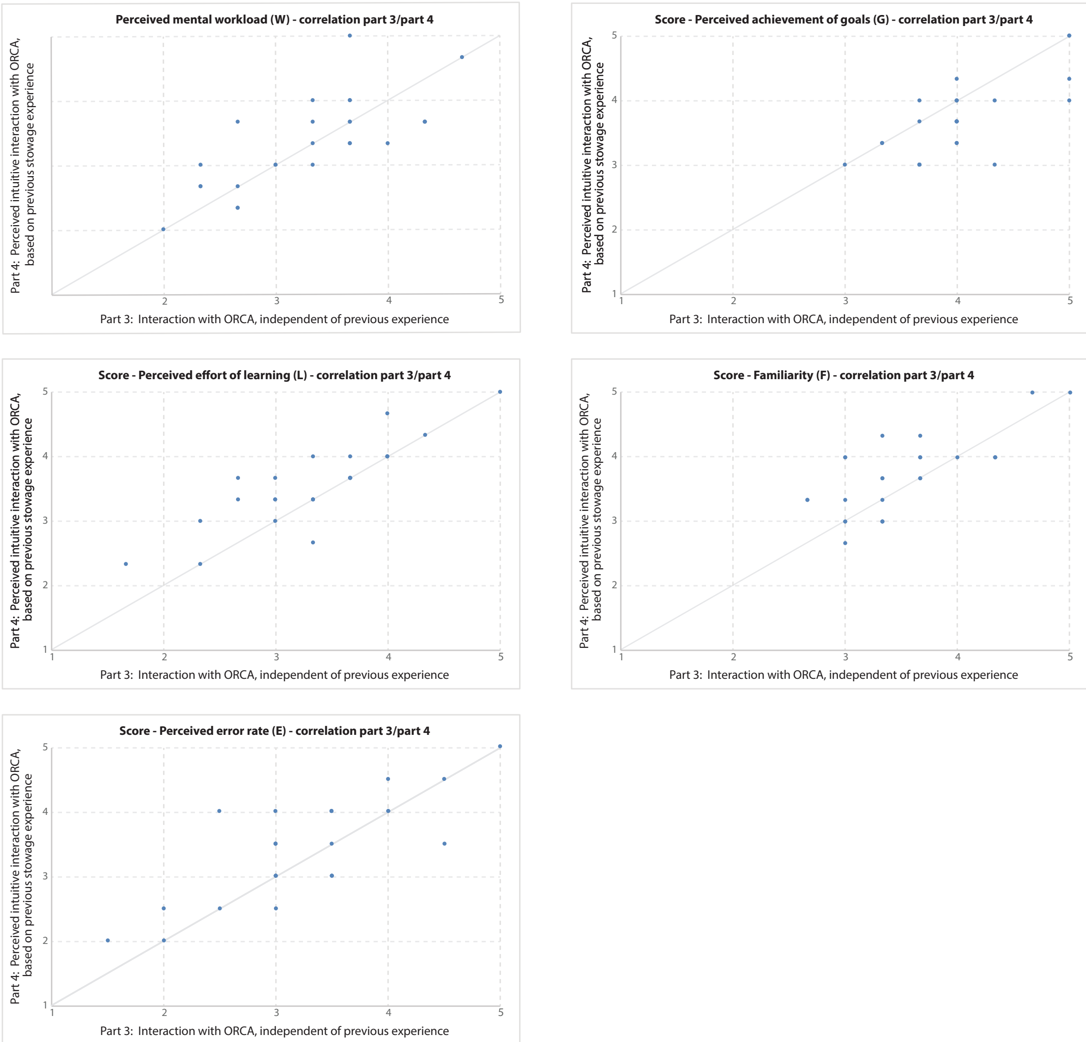


Figure 60. Average values distributed on subgroups W, G, L, F and E. (See larger versions, with a higher resolution, of these plots in appendix 6).

The average value in interaction with the ORCA system, correlating between interaction independent of previous experience (x-axis) and the perceived intuitive interaction based on previous stowage experience (y-axis). See appendix for bigger resolution of single plots.

The findings in subgroup W (top left) on questions 15–28, suggest a smaller perceived mental workload by the participants when relating the answers to their previous experience. The findings in subgroup L (mid left), show a higher degree of 'ease of learning' when relating to an already established base of knowledge coming from previous experience. The most pronounced trend in the material is shown in subgroup F - familiarity. The findings in this group show the strongest pattern towards a higher perception of intuitive interaction when the participants relate to previous experience. This is expected as previous research, and literature describes familiarity as the most prominent basis for intuitive interaction (Blackler et al., 2011; Naumann et al., 2008; Raskin, 1994). The findings in subgroup E (bottom), show a prominent but slightly scattered pattern towards a lowered error rate among the participants based on previous experience. Overall, the emerging pattern is trending towards interaction with ORCA being perceived as more intuitive when the participants relate to previous experience, which is strongest in subgroup F. The response on the questions in subgroup G received a good response in both parts 3 (questions 1–14), and 4 (questions 15–28). However, it does not seem that previous experience, 'sans' education, might have a noticeable impact on the participants' perception of achievement of goals. A significant number of participants had a similar education: vocational/naval school, university college/naval university college, or university, which, by providing them with a common educational base might have had a smoothing impact on the responses related to vocational knowledge.

Answers from the survey indicate that the naval users perceive the ORCA system as *slightly* more intuitive when answering questions about how they perceive the interface linked to their previous maritime experience than when they answer questions about interaction in the user interface in general. Also, the user interface's use of metaphors taken from vocational real-life seems to be a factor that evens out or smooths the difference between the various levels of experience.

8.8 Learning and development

In the ORCA-project, the expansive cycle of development was also recognised both individually and organisationally in the development process. When the previous system at the company headquarters, Othello, was introduced, it was the first computer-based stowage system installed in the company. All operators had previously used pen and paper forms in the stowage planning, thus being novices when having to utilise computers in the stowage planning process. Engeström's concept of development, and expansive cycle of learning (Engeström, 1999a) shows a strong focus on internalisation, as the new software is being learned and 'memorised', and during disruptions and contradictions, becoming self-reflective, followed by an increased focus on externalisation possibilities, as the operator, armed with vocational experience, search for solutions in the user interface. During this learning process, internalisation and externalisation are dialectically balanced into a new, established, work pattern which in turn is internalised in the mind and subsequently externalised into intuitive interaction as operations in the user interface. Vygotsky's concept of the zone of proximal development that is describing the different levels of development, with and without guidance, i.e. we might reach a certain level of development on our own – but by being with someone more capable – in the proximity of a 'more knowledgeable other', we might reach a higher level of potential development that is characterised by working in collaboration with e.g. more experienced peers (Vygotsky, 1978). In the context of computer-mediated activity, as e.g. in the stowage process within the ORCA system, the 'more knowledgeable other' *could* be another, more experienced operator. However, in the context of the use of software, designed on the foundation of experience and best practices, 'the more knowledgeable other' could be the guides and methods that have been inscribed in the program.

As presented in ch. 5.4.1, there was a rather significant discontent among operators related to the instability and cumbersome use of the previous system, Othello. This lead in turn to the operators stopping using the software, and returning to the old way of pen and paper when stowing the vessels, showing what in activity theory is termed *contradictions,* representing a lack of coherence between the operators' desired and expected way of executing operations in the user interface, within the designated timeframe, and expected system feedback. An operator complained e.g. that the program did not show warnings if something went wrong or simply crashed, but just, as one operator said: 'crashed silently' (ch. 5.4.1), causing problems, glitches, delays and breakdowns. In activity theory, such contradictions are regarded as sources for change and improvement - development (Kuutti, 1995). In an organisational development perspective, therefore, and for the sake of the stowage process as an activity system, the company had to develop a new stowage planning system that could adequately support a work flow based on established best practices, i.e. stowage planning by testing a simulation, visually - in a 'learning by expanding' manner. By grounding it on the acquired deep knowledge among experienced users and experts in the organisation, and by inscribing the existing task-competence into the software, ORCA was intended to become the platform for learning and development.

In the Vision Scope document for the project, it was stated that, in addition to assisting the personnel in obtaining the highest possible vessel utilisation, and ensure high quality stowage, the program had to be user-friendly and intuitive to use, and assist users in their daily work, and further develop their skills and knowledge of stowing chemical tankers.

They generated a set of specifications relied entirely on previous practice - intention and action. On top of the user requirements list was «Good user experience/friendliness». From observations, the findings show that the system follows the Vision Scope document closely, and facilitates for users to adjust parts of the user interface, approximating a personalised vocational familiarity. K., an experienced operator, demonstrated how she could alter e.g. the tab order in the Operation list in the main user interface.

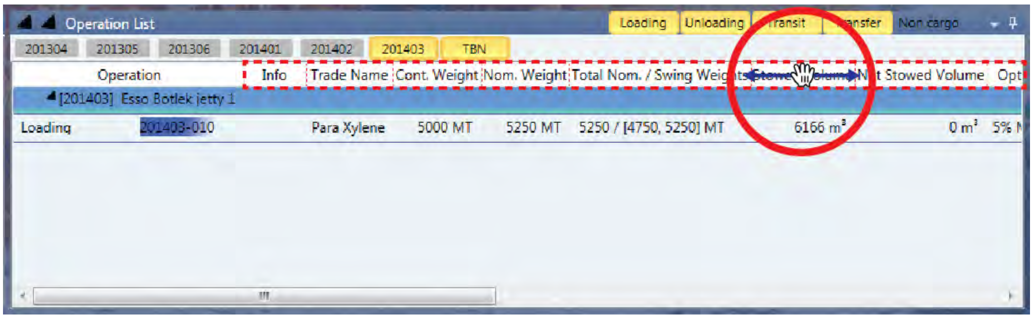


Figure 61. Customisable user interface - the various parts of the user interface, here, the tabs in the Operation List, can be moved.

K. explains how the GUI can be adjusted to the optimum workflow relative to the operator's skill and competence, and sets the user interface she prefers to show all the modules she requires, at a glance without having to scroll back and forth. This is parallel to what e.g. Trigg et al. calls 'adaptability through tailorability' (Trigg et al., 1987, pp. 725-726). The main operating procedure follows the standard modern drag-and-drop design pattern.

This feature of customising the user interface in ORCA, to fit the individual operator's skill-based workflow, skill and competence, was invoked by an established design pattern for accessing contextual menus in a user interface: the right-click/control-click, and would additionally aid the user in fitting the user interface to the externalisation of the mental probing of possibilities from being a mere thought, to operations and actions in the user interface. This customising functionality, of moving parts of the user interface would support the activity theoretical concept of internal and external activities transforming each other. The operator could change how the tool mediated the activity, and at the same time, being aware of this possibility, the tool would reciprocally influence how the operator would reach the objective.

8.8.1 Novices and experts and intuitive use

The current literature on intuitive interaction points at experience as the main prerequisite for familiarity, and discusses familiarity as the precondition for intuitive action. In ch. 6.2 on designing relations, and 7.1 on user interaction in ORCA, findings indicate that knowledge and experience being accumulated, and changes from being recently gained and new, to becoming previous knowledge and experience at various sections in the transition process, and by that becoming a part of the domain-specific knowledge (Popovic, 2000), and knowledge pattern repository (Simon, 1955). By pinpointing the knowledge sources, Mohs et al. (2006a) describes the sources in an expert-novice continuum going from: innate, sense experience, cultural, expertise, to the use of tools, recognising that as we approach the tool-plane the more domain-specific our knowledge becomes. The skilled worker context, described in the case description and the findings, relates to the two upper levels: expertise- and tool-level; that in Mohs et al. (2006a) would resemble the features of 'Domain-Specific Knowledge', and 'Task Experience and Expertise' in Popovic (Popovic, 2000, p. 935). However, in Popovic, 'Domain-Specific Knowledge', and 'Task Experience and Expertise' rest on general knowledge, that in Mohs et al. originates with innate knowledge and develops through sense experience and cultural influence. The number of computing devices in society has increased dramatically during the last couple of decades. There is nothing inherently intuitive in activities related to computers. Every aspect of computers is, at some point, an acquired skill. However, leaning on Svanæs, who states that we save and remember the knowledge of how computers work and operate, and add to this stored knowledge when new pieces of software are made or ways of operating the software are created within an existing structure (Svanæs, 2000), might explain how knowledge and experience in utilising computers as mediating tools, increasingly has become a part of the cultural set of experiences, competences and skills, in what Simon calls the societal knowledge pattern repository (Simon, 1955). What we perceive as intuitive would, then, in addition to the recollection processes, i.e. experiences, be biased by cultural and social shaping of activities, leading intuition, too, to be culturally dependent. As shown in figure 62, where Mohs et al. [2] situates culture as one of the contributors, and prerequisites for expert knowledge in the novice-expert continuum and subsequent intuitive interaction, but, as shown by Popovic (2000), not being a part of the domain-specific knowledge.

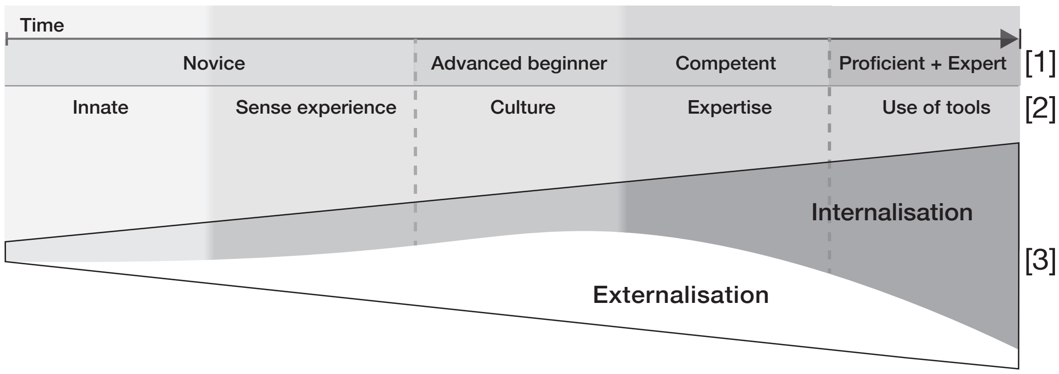


Figure 62. The transformation process of learning. The reciprocal process of expansive learning by the transformation of both internal and external process as the individual transforms and develops. Here, Engeström's model of expansive learning [3] is combined with Dreyfus and Dreyfus' skill acquisition matrix [1] and Mohs' knowledge category sources [2]. This illustration should only visualise that both internal and external aspects contributes to the transformation and development processes. The models are not placed in 'time-sync' in relation to each other.

In the ORCA case, this experience with culturally mediated tools and artefacts combined with educational similarities might explain why the responses shown in the answers to the questions about perceived intuitiveness in part 3 and part 4 in the survey, shows a slight difference in their perception of the user interface as intuitive. The responses in part 4, which relate to the connection of previous experience as a naval officer on board a chemical tanker to the use of the system, shows a *slightly* increased tendency to perceive the interaction as familiar, related to the mental workload, perceived achievement of goals, perceived effort of learning, familiarity and perceived error rate, compared to the received responses in part 3 that were independent of previous naval experience (see figure 59, p. 228 and 60, p. 231). However, the difference was small, as discussed in ch. 8.3.4.

8.9 Chapter summary

In this chapter I have discussed observations of the operators from a mediation perspective, indicating that the perception of interactions in a user interface as intuitive emerges through the transformations of actions into automated operations, and overcoming contradictions, and that in ORCA, by supporting domain-specific best practices, the user interface becomes transparent when mediating automated operations supporting the process of intuitive interaction as emerging from the transformation of actions into operations.

While the ORCA user interface becomes transparent in ordinary use, it becomes 'visible' when problems arise, as they constantly do during stowage planning of chemical tankers, similar to all activities in activity theory.

Through the operators' efforts to overcome the contradictions, which in ORCA consist of regular actions such as testing of stowage solutions, daily stowage planning through the ORCA user interface, will be in a continuous dialectical development process .

The ORCA user interface is designed to allow users to utilise errors or uncertainties to test and simulate solutions and to mediate these operations and actions in a sequential structure, which indicates that intuitive interaction does not just happen through momentary interactions, but may also emerge through sequences of actions. Thus, it is a tool aimed at operating in a skilled worker environment. The operators need to have a certain task-competence because ORCA is so closely linked to an established practice of doing things. This task-competence is inscribed into the user interface where the screen artefacts then represent familiar shapes, terms and concepts, thus performing operations in a familiar way, or become a part of a sequence of actions, extending human capabilities in a way that allows the human-ORCA functional organ to work as one unit.

CHAPTER 9.
Concluding remarks

What is intuitive interaction? The work conducted in this thesis has aimed to answer this question by exploring the foundation and concept of intuitiveness from the perspective of computer-mediated interaction in a skilled worker environment.

The newly developed ORCA software is a program for the stowage planning of chemical tankers that is designed to allow users to utilise errors or uncertainties as learning points, and to mediate these operations and actions in a sequential structure. Analysing patterns of use among ORCA operators indicates that intuitive interaction does not just happen through momentary interactions, but may also emerge through sequences of actions.

Intuitiveness, as related to a momentary, single operation that has often been the locus of HCI research related to a computer-mediated intuitive activity. Interaction with the ORCA user interface shows that intuitive interaction can be complex, i.e. both momentary and immediate interaction, and as a sequence of operations being perceived, over time, as one fluid action, knowing what the next step in that sequence of operations will be.

In ORCA, intuitive interaction does not primarily occur instinctively and quickly as a momentary doing, but is rooted in contradictions (as all activities in activity theory), and emerge as actions that consist of sequences of single operations that are transformed into one. ORCA is closely linked to practise and has inscribed the domain-specific competence into the user interface. This shows that there are no significant conflicts between a momentary approach to interaction and an approach to interaction as a sequence of actions.

Thus, it is a tool aimed at operating in a skilled worker environment. The operators need to have a certain task-competence because ORCA is so closely linked to an established practice of how things are done, which is inscribed into the user interface. The screen artefacts represent familiar shapes, terms and concepts; thus operations can be undertaken in a familiar way, or become a part of a sequence of actions, extending human capabilities in a way that allows the human-ORCA functional organ to work as one unit.

The second question was, 'How intuitive interaction can be designed?' This thesis reports from a software development project, where much attention was given to the users. The developers moved in and stayed with the users throughout the project, where the aim was to understand the possible impact this level of proximity might have had on the cooperation between the developers and the users. Co-locating the development team and the users not only in the same building, but in adjacent rooms on the same floor, created a proximity that gave the participants the experience of being accountable in the software development process, as well as making the domain-specific competence immediately accessible. For the ORCA project, this approach ended in a stowage planning system that has completely absorbed the task-competence originating from practice, which is all about testing whether something is possible or not, and in the correct order. Thus, the program must be able to absorb complexity, supporting the idea that not only momentary actions but also complex actions can be performed intuitively, as this is elaborately discussed in chapter 7: Use of ORCA, and 8: Mediation in ORCA.

9.1 Thesis contribution

The thesis contributes to the field by connecting the notion of intuitiveness as an emergent quality that occurs through a dialectical process of transforming actions as collections of sequential single 'operations' into one fluid operation, showing that a complex action can be performed intuitively as an operation, related to the extended capabilities of the human-computer functional organ. Kuutti & Bannon's paper (2014), which reviews as problematic that research on intuition within HCI, until now, has been focused almost entirely on momentary actions, where the character of operation resembles that of being instinctive. Thus, this thesis aims at contributing by suggesting that the perception of intuitive interaction in user interfaces may emerge as a result of development.

This thesis is contributing to practise, with implications for the process of systems design, by identifying aspects of intuitive interaction, and a suggestion on how intuitive interaction, in a human-computer interaction context, might be designed, by suggesting that software which facilitate the handling, or even embracement of contradictions or breakdowns as possibilities for progress could inform practitioners in designing systems that would facilitate development through 'dialectical' error handling, and that the extreme form of participatory design presented in this thesis might point in a direction from which modern systems design could benefit.

9.2 Future work and research

The case study is focused on the human-interface interaction in a narrow niche program in a particular vocational field, where the findings indicate that intuitive interaction might emerge from complex actions evolving into single operations. The emergence, and increase, of ordinary people's needs of computers in order to solve ordinary, everyday tasks – i.e. computer-mediated everyday activities, means that issues of human agency related to the use of computers are of general importance. Further studies, of the action-operation dynamics of use-activity in a broader context, including cases that would not necessarily be regarded as best practices, would seem like a natural direction for future work.

Also, the findings from the ORCA system, with its focus on embracing and overcoming contradictions as means for intuitive interaction through development might point in the direction of researching the character of mediation and prospective intuitive human-technology interaction.

9.3 Limitations

The case study has investigated the development and use of the specific niche software of stowage planning of chemical tankers – ORCA, and was undertaken in a skilled worker environment, and one limitation regarding the validity of the findings is the lack of applicability in general, and that it might prove as a challenge to find similar findings in other contexts and user groups. Additionally, although human activity towards a specific goal might have a similar character, the use activity might be interpreted differently in a different perspective, thus providing differing results.

The main method for the case study, both for the development and post-implementation use phases has been semi-structured interviews and non-participant observation in the field. Observation of use was restricted by the fact that stowage activity happened in a random pattern which made access to the observation field similarly random, thus difficult to plan. This was solved by setting up simulation sessions, with real cargo and real vessels, but with simulated stowage planning. This provided realistic stowage operations, actions, and activities, but within a timeline more compressed than would have been normal within a regular stowage session. Since the tasks undertaken in the user interface, and their relation to the each other in the hierarchical activity structure, have been the primary focus of the analysis have been sequences of operations and actions. Therefore they are, individually, not affected by this simulated compression, and it is unlikely that this affected the results, but it should be mentioned here as an element of the description of possible limitations. More observations would have been desirable, but realistic simulations greatly increased the number of observed interactions in the user interface, causing the amount of observation data to be somewhat greater than the number of observation sessions might indicate.

The collection of data was influenced by the fact that observations could not be planned in advance. Combined with observation premises being rather distant, and with somewhat limited access to the field of data collection, depending on e.g. frequent air transportation, led to the qualitative material having a slightly smaller size than would have been preferred. A survey was conducted in February 2016 among naval personnel as an effort to supplement the data collected from the interviews and observation sessions, especially the findings related to the discussion of interaction based on domain-specific competence. However, it is only the users' descriptions of their impression of how they experience the interaction with the user interface in ORCA, post factum. There is a significant degree of uncertainty, and possible heuristic bias connected to the answers from the survey, and to conclude from these findings would be questionable.

The empirical material in this thesis suggests that intuitive interaction in user interfaces emerges from the transformation of use-activity, evolving from actions to operations, that stems from the dialectical processes of internalisation and externalisation within a human-computer functional organ. However, there are still many studies to be done in order to reach a greater degree of understanding of what intuitive interaction is, and how it might be designed. This thesis shows just one possible approach among several.

References

BÆRENTSEN, K.B. (2000) Intuitive User Interfaces. *Scandinavian Journal of Information Systems*, **12(1)**, 29-60.

BÆRENTSEN, K.B. and TRETTVIK, J. (2002) *An activity theory approach to affordance*. ACM, Association for Computing Machinery ACM, New York, New York, USA.

BAKKE, S. and BRATTETEIG, T. (2015) The Closer the Better: Effects of Developer-User Proximity for Mutual Learning Human-Computer Interaction: Design and Evaluation. In *9169(Chapter 2),* Springer International Publishing, Cham, pp. 14-26.

BANNON, L.J. (1991) From Human Factors to Human Actors. In *Design at Work: Cooperative Design of Computer Systems,* Lawrence Erlbaum Associates, Hillsdale, pp. 25-44.

BANNON, L.J. and BØDKER, S. (1989) Beyond the Interface: Encountering Artifacts in Use. *DAIMI PB -299*, **18(288)**, 1-32.

BANSLER, J.P. and HAVN, E.C. (1994) Information systems development with generic systems. *Proceedings of ECIS '94,* 707-715.

BARNARD, P.J. (1987) Cognitive resources and the learning of human-computer dialogs. **Interfacing thought: Cognitive aspects of human-computer interaction** 112-158.

BASTICK, T. (1982) *Intuition: How We Think and Act*. Wiley & Sons Inc, Chicester.

BERTELSEN, O.W. and BØDKER, S. (2003) Activity theory. In *HCI models, theories, and frameworks: Toward a multidisciplinary science multidisciplinary science,* (Ed, Carroll, J.M.) San Francisco: Morgan Kaufmann, pp. 291-324.

BJERKNES, G. and BRATTETEIG, T. (1987) Florence in wonderland. In *Computers and Democracy - a Scandinavian Challenge,* (Eds, Bjerknes, G., Ehn, P. and Kyng, M.) Avebury, Aldershot, UK, pp. 279-295.

BJERKNES, G. and BRATTETEIG, T. (1995) User Participation and Democracy: A Discussion of Scandinavian Research on System Development. *Scandinavian Journal of Information Systems*, **7(1)**, 73-98.

BLACKLER, A. (2008) *Intuitive Interaction with Complex Artefacts - Empirically-Based Research (VDM Verlag)*. Ph.D. thesis,

BLACKLER, A. and POPOVIC, V. (2015) Editorial:Towards Intuitive Interaction Theory. *Interacting with Computers*, **Special issue on Intuition)**, 1-7.

BLACKLER, A.L., POPOVIC, V. and MAHAR, D.P. (2010) Investigating users' intuitive interaction with complex artefacts. *Applied Ergonomics*, **41(1)**, 72-92.

BLACKLER, A.L. and HURTIENNE, J. (2007) Towards a unified view of intuitive interaction : definitions, models and tools across the world., 36-57.

BLACKLER, A.L., POPOVIC, V. and MAHAR, D.P. (2002) Intuitive Use of Products. *Design Research Society, Faculty of Built Environment and Engineering; Faculty of Health, Queensland University of Technology*.

BLACKLER, A.L., POPOVIC, V. and MAHAR, D.P. (2003) Designing for Intuitive Use of Products: An Investigation. *Faculty of Built Environment and Engineering; Faculty of Health; School of Psychology & Counselling, Queensland University of Technology*.

BLACKLER, A.L., POPOVIC, V. and MAHAR, D.P. (2005) Intuitive Interaction Applied to Interface Design. *Faculty of Built Environment and Engineering, Queensland University of Technology*.

BLACKLER, A.L., POPOVIC, V. and MAHAR, D.P. (2014) Applying and testing design for intuitive interaction. *International Journal of Design*, **20(1)**, 7-26.

BLACKLER, A.L. (2006) *Intuitive interaction with complex artefacts*. Queensland University of Technology, Adelaide.

BLACKLER, A.L. et al. (2011) Researching Intuitive Interaction. *Diversity and unity: Proceedings of IASDR2011, Delft, The Netherlands,* Delft, The Netherlands,

BLOMBERG, J., SUCHMAN, L. and TRIGG, R.H. (1996) Reflections on a work-oriented design project. *Human-Computer Interaction*, **11(3)**, 237-265.

BØDKER, S. (1989) A Human Activity Approach to User Interfaces. *Human–Computer Interaction*, **4(3)**, 171-195.

BØDKER, S. (1991) *Through the Interface - a Human Activity Approach to User Interface Design*. Lawrence Erlbaum, Hillsdale, New Jersey.

BØDKER, S. et al. (1987) A UTOPIAN experience: On design of powerful computer-based tools for skilled graphic workers. In *Computers and Democracy: a Scandinavian Challenge,* (Eds, Bjerknes, G., Ehn, P. and Kyng, M.) Avebury, Aldershot, UK, pp. 251-278.

BØDKER, S. and KLOKMOSE, C.N. (2011) The Human–Artifact Model: An Activity Theoretical Approach to Artifact Ecologies. *Human–Computer Interaction*, **26(4)**, 315-371.

BØDKER, S. and KLOKMOSE, C.N. (2012) Preparing students for (inter-) action with activity theory. *International Journal of Design*, **6(3)**, 99-111.

BOWERS, K.S. et al. (1990) Intuition in the context of discovery. *Cognitive Psychology*, **22(1)**, 72-110.

BRATTETEIG, T. et al. (2013) Methods: Organizing Principles and General Guidelines for Participatory Design Projects. In Routledge International Handbook of Participatory Design, Routledge, New York, pp. 117-144.

BRATTETEIG, T. and WAGNER, I. (2014) *Disentangling Participation. Power and Decision-Making in Participatory Design*. Springer, Heidelberg.

BROOKS, F.P. (1975) *The mythical man-month*. Addison-Wesley, Boston, MA.

BUNGE, M.A. (1962) *Intuition and science*. Prentice-Hall, Englewood Cliffs, N.J.

CAPPON, D. (1994) A new approach to intuition: IQ2. *Omni*, **16(12)**, 34-42.

CARD, S.K., MORAN, T.P. and NEWELL, A. (1983) *The psychology of human-computer interaction*. Lawrence Erlbaum, Hillsdale, NJ.

COLE, M. (1988) Cross-Cultural Research in the Socio historical Tradition., 1-21.

COLE, M. (1990) Cultural psychology: A once and future discipline? In *Nebraska Symposium on Motivation 1989: Cross-cultural perspectives (vol. 37),* (Ed, Berman, J.J.) University of Nebraska Press, Lincoln, pp. 279-336.

COLE, M. (1996) *Cultural Psychology. a once and future discipline*. The Bellmap Press of Harvard University Press, Cambridge, Massachusetts, and London, England.

COLE, M. and GRIFFIN, P. (1980) Cultural Amplifiers Reconsidered., 1-13.

DANIELS, H., COLE, M. and WERTSCH, J.V. (2007) *The Cambridge Companion to Vygotsky*. Cambridge University Press, Cambridge.

DAVYDOV, V.V., ZINCHENKO, V.P. and TALYZINA, N.F. (1983) The Problem of Activity in the Works of A. N. Leont'ev. *Journal of Russian and East European Psychology*, **21(4)**, 31-42.

DAVYDOV, V.V. (1999a) A New Approach to the Interpretation of Activity Structure and Content. In *Activity Theory and Social Practice,* (Eds, Chaiklin, S., Hedegaard, M. and Jensen, U.J.) Aarhus University Press, Århus, pp. 39-50.

DAVYDOV, V.V. (1999b) The content and unsolved problems of activity theory. In *Perspectives on activity theory,* (Eds, Engeström, Y., Miettinen, R. and Punamäki, R.-L.) Cambridge University Press, pp. 39-52.

DE GROOT, A.D. (1978) *Thought and choice in chess. 2 ed.* Mouton Publishers, The Hague.

DOURISH, P. (2004) What we talk about when we talk about context. *Personal and Ubiquitous Computing*, **8(1)**, 19-30.

DREYFUS, H.L. and DREYFUS, S.E. (1986) *Mind over machine: The power of human intuition and expertise in the era of the computer*. Free Press, New York.

EHN, P. (1993) Scandinavian design: On participation and skill. *Participatory design: Principles and practices*, 41-77.

ENGELBART, D.C. (1962) *Augmenting Human Intellect: A Conceptual Framework.* Summary Report. Stanford Research Institute, Menlo Park, CA.

ENGESTRÖM, Y. (1987) *Learning by Expanding: An Activity-Theoretical Approach to Developmental Research*. Ph.D. thesis, Helsinki.

ENGESTRÖM, Y. (1996) Development as breaking away and opening up: A challenge to Vygotsky and Piaget. *Swiss Journal of Psychology*, **55**, 126-132.

ENGESTRÖM, Y. (1999a) Activity theory and individual and social transformation. In *Perspectives on Activity Theory,* (Eds, Engeström, Y., Miettinen, R. and Punamäki, R.-L.) Cambridge University Press, Cambridge, pp. 19-38.

ENGESTRÖM, Y. (1999b) Expansive visibilization of work: An activity-theoretical perspective. *Computer Supported Cooperative Work (CSCW)*, **8(1-2)**, 63-93.

ENGESTRÖM, Y. (1999c) Learning by expanding: ten years after., 1-10.

ERICKSON, T.D. (1990) Working with interface metaphors. In *The art of human-computer interface design,* (Ed, Laurel, B.) Addison-Wesley, pp. 65-73.

FISHBEIN, E. (1987) *Intuition in science and mathematics: An educational approach.* D. Reidel Publishing, Dordrecht.

FJELD, M. et al. (2002) Physical and Virtual Tools: Activity Theory Applied to the Design of Groupware. *Computer Supported Cooperative Work (CSCW)*, 1-28.

GAVER, W.W. (1991) Technology affordances. **Proceedings of the SIGCHI conference on Human factors in computing systems** 79-84.

GHAOUI, C. (2005) *Encyclopedia Of Human Computer Interaction*. IGI Global, London.

GIBSON, J.J. (1986) *The ecological approach to visual perception*. Psychology Press, Hove, East Sussex.

GIBSON, J.J. (1977) The concept of affordances. *Perceiving, acting, and knowing*, 67-82.

GIBSON, J.J. (1979) *The Theory of Affordances. The Ecological Approach to Visual Perception*. Lawrence Erlbaum, Hillsdale.

GLASER, R. (1987) Thoughts on expertise. In *Cognitive functioning and social structure over the life course,* (Eds, Schooler, C. and Schaie, K.W.) Ablex Publishing, Norwood, NJ, pp. 81-94.

GREEN, T.R.G., DAVIES, S.P. and GILMORE, D.J. (1996) Delivering cognitive psychology to HCI: the problems of common language and of knowledge transfer. *Interacting With Computers*, **8(1)**, 89-111.

GREENBAUM, J.M. and KYNG, M. (1991) *Design at work: Cooperative design of computer systems*. L. Erlbaum Associates Inc., Hillsdale, NJ.

GREGORY, R.L. and ZANGWILL, O.L. (1987) *The Oxford companion to the mind.* Oxford University Press, Oxford, New York.

HEIDEGGER, M. (1988) *The basic problems of phenomenology*. Indiana University Press, Bloomington & Indianapolis.

HERSTAD, J. (2007) *Circumspective use of equipment: The case of bicycle messengers*. Ph.D. thesis, University of Oslo.

HEWETT, T. (1992) *Acm Sigchi Curricula for Human Computer Interaction*. Assn for Computing Machinery, New York.

HOFFMAN, R.R. (1992) *The Psychology of Expertise*. Springer-Verlag, New York.

HOLTZBLATT, K. and JONES, S. (1993) Contextual Inquiry: A Participatory Technique for System Design. In *Participatory design: Principles and Practices,* (Eds, Schuler, D. and Namioka, A.) Lawrence Erlbaum, Hillsdale, NJ, pp. 177-210.

HORNECKER, E. (2012) Beyond affordance: Tangibles' Hybrid Nature. **Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction** 175-182.

HORNECKER, E. and BUUR, J. (2006) Getting a grip on tangible interaction: a framework on physical space and social interaction. **Proceedings of the SIGCHI conference on Human Factors in computing systems** 437-446.

HUMMELS, C., SMETS, G. and OVERBEEKE, K. (1997) An intuitive two-handed gestural interface for computer supported product design Gesture and Sign Language in Human-Computer Interaction. In *1371(Chapter 17),* Springer, Berlin, Heidelberg, Berlin, Heidelberg, pp. 197-208.

HURTIENNE, J. and NAUMANN, A. (2010) QUESI—A questionnaire for measuring the subjective consequences of intuitive use. In *Interdisciplinary College 2010*, (Eds, Porzel, R., Sebanz, N. and Spitzer, M.), **Focus Theme: Play, Act and Learn**, 536.

HURTIENNE, J., WEBER, K. and BLESSING, L. (2008) Prior Experience and Intuitive Use: Image Schemas in User Centred Design. In *Designing inclusive futures,* Springer, pp. 107-116.

HURTIENNE, J. (2009) *Image Schemas and Design for Intuitive Use Exploring New Guidance for User Interface Design*. Ph.D. thesis, Technischen Universität Berlin.

HURTIENNE, J. (2017) How Cognitive Linguistics Inspires HCI: Image Schemas and Image-Schematic Metaphors. *International Journal of Human-Computer Interaction*, **33(1)**, 1-20.

HURTIENNE, J. and BLESSING, L. (2007) Design for Intuitive Use-Testing image schema theory for user interface design. *Guidelines for a Decision Support Method Adapted to NPD Processes*,

HURTIENNE, J., DINSEL, C. and STURM, C. (2009) QUESI - Design and Evaluation of a Questionnaire for the Subjective Consequences of Intuitive Use. *Unpublished Project Documentation*, Technische Universität Berlin.

HURTIENNE, J. and ISRAEL, J.H. (2007) Image Schemas and Their Metaphorical Extensions – Intuitive Patterns for Tangible Interaction. *Proceedings of TEI '07,* Baton Rouge, Louisiana, USA, 127-134.

HUTCHINS, E.L., HOLLAN, J.D. and NORMAN, D.A. (1985) Direct Manipulation Interfaces. *Human-Computer Interaction*, **1**, 311-338.

ILYENKOV, E.V. (2009) *Dialectical Logic; Essays on its History and Theory*. marxist.org, (Republishing of version from 1977 by Progress Publishers, Moscow).

ISHII, H. (2008) The Tangible User Interface and its Evolution. *Communications of the ACM*, **51(6)**, 32-36.

ISO (1998) ISO9241-11 – Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability.,

ISRAEL, J.H. et al. (2009) On intuitive use, physicality and tangible user interfaces. *International Journal of Arts and Technology*, **2(4)**, 348-366.

JOHNSON, M. (1987) The Body in the Mind: The Bodily Basis of Meaning, Imagination, and Reason., 1-268.

KAHNEMAN, D. and KLEIN, G. (2009) Conditions for intuitive expertise: A failure to disagree. *American psychologist*, **64(6)**, 515-526.

KAHNEMAN, D. and TVERSKY, A. (1982) On the study of statistical intuitions. *Cognition*, **11(2)**, 123-141.

KAPTELININ, V. (1992) Integration of Computer Tools Into the Structure of Human Activity: Implications for Cognitive Ergonomics. *The Sixth European Conference on Cognitive Ergonomics (ECCE 6),* Balatonfüred, Hungary, 85-294.

KAPTELININ, V. (1995a) Activity theory: Implications for human-computer interaction. In *Context and consciousness: Activity theory and Human-Computer Interaction,* (Ed, Nardi, B.A.) MIT Press, Cambridge, pp. 103-116.

KAPTELININ, V. (1995b) Computer-Mediated Activity: Functional Organs in Social and Developmental Contexts. In *Context and Consciousness: Activity Theory and Human-Computer Interaction,* (Ed, Nardi, B.A.) The MIT Press, Cambridge, MA, pp. 45-68.

KAPTELININ, V. (1996a) Distribution of cognition between minds and artifacts: Augmentation of mediation. *AI & SOCIETY*, **10(1)**, 15-25.

KAPTELININ, V. (1996b) Activity Theory: Implications for Human-Computer Interaction. In *(5),* pp. 102-116.

KAPTELININ, V. (2005) The Object of Activity: Making Sense of the Sense-Maker. *Mind, Culture, and Activity*, **12(1)**, 4-18.

KAPTELININ, V. (2015) Designing mediation. *Proceedings of the European Conference on Cognitive Ergonomics, ECCE '15,* New York, New York, USA, 1-4.

KAPTELININ, V., KUUTTI, K. and BANNON, L.J. (1995) Activity Theory: Basic Concepts and Applications., 1-13.

KAPTELININ, V. and NARDI, B. (2003) Post-cognitivist HCI: second-wave theories. *CHI '03,* 692-693.

KAPTELININ, V. and NARDI, B. (2012a) Affordances in HCI: Toward a Mediated Action Perspective. *CHI '12,* 967-976.

KAPTELININ, V. and NARDI, B. (2012b) *Activity Theory in HCI: Fundamentals and reflections*. Morgan & Claypool Publishers, San Rafael.

KAPTELININ, V. and NARDI, B. (2018) Activity Theory as a Framework for Human-Technology Interaction Research. *Mind, Culture, and Activity*, **25(1)**, 3-5.

KAPTELININ, V. et al. (2003) Post-cognitivist HCI: Second-Wave Theories. *CHI '03,* Fort Lauderdale, Florida, 692-693.

KAPTELININ, V. and NARDI, B.A. (1997) Activity theory. *CHI '97 extended abstracts,* New York, New York, USA, 158-159.

KAPTELININ, V. and NARDI, B.A. (2006) *Acting with Technology: Activity Theory and Interaction Design*. The MIT Press, Cambridge.

KAY, A. and GOLDBERG, A. (1977) Personal dynamic media. *Computer*, **10(3)**, 31-41.

KAY, A. (2001) User Interface: A Personal View. In *Multimedia: From Wagner to Virtual Reality,* W.W. Norton & Company, New York, pp. 121-131.

KLEIN, G. (2003) *The Power of Intuition*. Currency-Doubleday, New York.

KLEIN, G.A. (1999) *Sources of power: How people make decisions*. MIT Press, Cambridge, MA.

KLEIN, H.K. and MYERS, M.D. (1999) A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Infomation Systems. *MIS Quarterly*, **23(1)**, 67-94.

KUUTTI, K. (1995) Activity Theory as a potential framework for human- computer interaction research. In *Context and Consciousness: Activity Theory and Human Computer Interaction,* (Ed, Nardi, B.A.) MIT Press, Cambridge, pp. 17-44.

KUUTTI, K. (1999) Activity theory, transformation of work, and information systems design. In *Perspectives on activity theory,* (Eds, Engeström, Y., Miettinen, R. and Punamäki, R.-L.) Cambridge University Press, pp. 360-373.

KUUTTI, K. and BANNON, L.J. (2014) The turn to practice in HCI. *the 32nd annual ACM conference CHI '14,* New York, New York, USA, 3543-3552.

LAKOFF, G. and JOHNSON, M. (1980) *Metaphors we live by*. The University of Chicago Press, Chicago.

LATOUR, B. (1999) *Pandoras Hope. Essays on the Reality of Science Studies*. Harvard University Press, Cambridge MA.

LEONT'EV, A.N. (1974) The Problem of Activity in Psychology. *Soviet Psychology*, **13(2)**, 4-33.

LEONTIEV, A.N. (1978) *Activity, Consciousness, and Personality*. Prentice-Hall, Englewood Cliffs.

LEONTYEV, A.N. (2009) *The Development of Mind*. Marxist Internet Archive, Pacifica, CA. (A reproduction of the English edition, *Problems in the Development of the Mind*, by Progress Publishers, Moscow, 1981 with the addition of "Activity and Consciousness," published by Progress Publishers in 1977.).

LEONTYEV, A.N. (1977) Activity and Consciousness. In *Philosophy in the USSR, Problems of Dialectical Materialism,* Progress Publishers, Moscow, pp. 180-202.

LEVY, S. (1995) *Insanely great: the life and times of Macintosh, the computer that changed everything*. Penguin Books, New York.

LICKLIDER, J.C.R. (1960) Man-Computer Symbiosis. *IRE Transactions on Human Factors in Electronics*, **HFE-1**, 4-11.

LÖWGREN, J. and STOLTERMAN, E. (2007) *Thoughtful I nteraction D esign: A Design Perspective on Information Technology*. The MIT Press, Cambridge, MA.

LURIA, A.R. and COLE, M. (1978) *Selected Writings of A.R. Luria*. M. E. Sharpe, White Plains, N.Y.

LURIA, A.R., COLE, M. and COLE, S. (1979) *The Making of Mind: A Personal Account of Soviet Psychology*. Harvard University Press, Cambridge.

MCGRENERE, J. and HO, W. (2000) Affordances: Clarifying and evolving a concept. **Graphics Interface** 179-186.

MEEHL, P.E. (1954) *Clinical versus statistical prediction: A theoretical analysis and a review of the evidence.* University of Minnesota Press, Minneapolis, MN.

MOHS, C. et al. (2006a) IUUI–intuitive use of user interfaces. In *Usability professionals 2006,* (Eds, Peissner, M. et al.) German Chapter der Usability Professionals' Association., Stuttgart, pp. 130-133 (in German).

MOHS, C., NAUMANN, A. and KINDSMÜLLER, M.C. (2007) Mensch-Technik-Interaktion: intuitiv, erwartungskonform oder vertraut? *MMI Interaktiv-User Experience: Vol. 1, No. 13*,

MOHS, C. et al. (2006b) IUUI – Intuitive Use of User Interfaces. In *Usability Professionals 06,* (Eds, Bosenick, T. et al.) German Chapter der Usability Professionals' Association, Stuttgart, pp. 130-133 (English translation).

NARDI, B.A. (1995a) Activity Theory and Human-Computer Interaction. In *Context and consciousness: Activity theory and human-computer interaction,* (Ed, Nardi, B.A.) The MIT Press, Cambridge, pp. 7-16.

NARDI, B.A. (1995b) *Context and Consciousness: Activity Theory and Human-Computer Interaction*. The MIT Press, Cambridge, MA.

NAUMANN, A. et al. (2007) Intuitive Use of User Interfaces: Defining a Vague Concept. 1-9.

NAUMANN, A.B. et al. (2008) *Design for intuitive use: beyond usability*. ACM, Association for Computing Machinery ACM, New York, New York, USA.

NEWELL, A. and CARD, S.K. (1985) The prospects for psychological science in human-computer interaction. *Human-computer interaction*, **1(3)**, 209-242.

NIELSEN, J. (1993) *Usability Engineering (Interactive Technologies)*. Morgan Kaufmann, NIELSEN, J. (1995) *10 Usability Heuristics for User Interface Design*. https://www.nngroup.com/articles/ten-usability-heuristics/. Access date: June 11, 2014.

NORMAN, D. (2013) *The Design of Everyday Things. Revised and expanded edition.* Basic Books, New York.

NORMAN, D.A. (1987) Cognitive engineering–cognitive science. **Interfacing thought: Cognitive aspects of human-computer interaction** 325-336.

NORMAN, D.A. (1998) *The Invisible Computer: Why Good Products Can Fail, the Personal Computer Is So Complex, and Information Appliances Are the Solution*. The MIT Press, Cambridge, MA.

NORMAN, D.A. (1988) *The Psychology Of Everyday Things*. Basic Books,

NORMAN, D., A (1999) Affordance, Conventions, and Design. *Interactions*, **may + june**,

NYGAARD, K. (1979) The iron and metal project: trade union participation. *Computers dividing man and work - Recent Scandinavian research on planning and computers from a trade union perspective*, **Demos project report 13**, 94-107.

PAYNE, S.J. (2003) Users' mental models: the very ideas. *HCI models, theories, and frameworks: Toward a multidisciplinary science*, 135-156.

PLATO. (1969) *Plato in Twelve Volumes. Vols. 5 & 6 (translated by Paul Shorey)*. Harvard University Press, Cambridge, MA, William Heinemann Ltd., London.

POPOVIC, V. (2000) Expert and Novice User Differences and Implications for Product Design and Useability. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, **44(38)**, 933-936.

POPOVIC, V. (2003) Expert and Novice Users Model and their Application to the Design Process. *Journal of the Asian Design International Conference*.

RASKIN, J. (1994) Viewpoint: Intuitive equals familiar. *Communications of the ACM*, **37(9)**, 17-18.

RASKIN, J. (2000) *The Humane Interface: New Directions for Designing Interactive Systems*. Addison-Wesley Professional,

RICHMAN, H.B. et al. (1996) Perceptual and memory processes in the acquisition of expert performance: The EPAM model. In *The road to excellence: The acquisition of expert perfonnance in the arts and sciences, sports and games,* (Ed, Ericsson, K.A.) Erlbaum Associates, NJ, pp. 167-187.

ROGERS, Y. (2012) *HCI theory: classical, modern, and contemporary*. Morgan & Claypool, San Rafael.

ROGERS, Y. (2008) 57 Varieties of Activity Theory. *Interacting with Computers*, **20(2)**, 247-250.

SCHOLZ, D. (2006) Intuitivität von Mensch-Maschine-Systemen aus Benutzersicht. Diplomarbeit. Technische Universität Berlin.

SCHÖN, D.A. (1983) *The r eflective pr actitioner: Ho w pr ofessionals th ink in ac tion*. Basic books,

SCHULER, D. and NAMIOKA, A. (1993) *Participatory Design: principles and practices*. L. Erlbaum Associates Inc., Hillsdale, New Jersey.

SHANTEAU, J. (1992) Competence in experts: The r ole o f t ask characteristics. *Organizational behavior and human decision processes*, **53**, 252-266.

SHNEIDERMAN, B. et al. (2009) *Designing the User Interface: Strategies for Effective Human-Computer Interaction (5th Edition)*. Pearson, Boston.

SHNEIDERMAN, B. et al. (2016) *Designing the User Interface: Strategies for Effective Human-Computer Interaction (6th Edition)*. Pearson, Boston.

SIMON, H.A. (1955) A Behavioral Model of Rational Choice. *The Q uarterly J ournal of Economics*, **69(1)**, 99-118.

SIMONSEN, J. and ROBERTSON, T. (Eds.) (2013) Routledge international handbook of participatory design Routledge, New York.

SUCHMAN, L. (2010) *Human-Machine Reconfigurations.Plans a nd s ituated a ctions. 2 ed.* Cambridge University Press, Cambridge.

SUTHERLAND, I. (1963) *Sketchpad-A Man-Machine Graphical Communication System.* Ph.D. thesis, Massachusetts Institute of Technology.

SVANÆS, D. (2000) *Understanding interactivity: Steps to a phenomenology of human-computer interaction*. Ph.D. thesis, Norges teknisk-naturvitenskapelige universitet (NTNU).

SWANN, L. et al. (2014) Airport security screeners expertise and implications for interface design. *Design Research Society Conference 2014,* Umeå, Sweden.

TRIGG, R.H., MORAN, T.P. and HALASZ, F.G. (1987) Adaptability and Tailorability in NoteCards. *Human–Computer Interaction–INTERACT '87,* 723-728.

TVERSKY, A. and KAHNEMAN, D. (1974) Judgment under uncertainty: Heuristics and biases. *Science*, **84**, 1124-1131.

VYGOTSKY, L.S. (1978) *Mind in society: The d evelopment o f h igher p sychological processes*. Harvard University Press, Cambridge, MA.

VYGOTSKY, L.S. (1987) *The Collected Works of L. S. Vygotsky vol. 1*. Plenum Press, New York/ London.

VYGOTSKY, L.S. (1993) *The C ollected W orks o f L . S . V ygotsky v ol. 2* . SPRINGER SCIENCE+BUSINESS MEDIA, New York.

WALSHAM, G. (1993) *Interpreting information systems in organizations*. John Wiley & Sons, Chichester.

WERTSCH, J.V. (1981) *The C oncept o f A ctivity i n S oviet P sychology*. M .E. S harpe Inc., Armonk, New York.

WERTSCH, J.V. (1991) *Voices of the Mind: A Sociocultural Approach to Mediated Action*. Harvard University Press, Cambridge.

WERTSCH, J.V. (1998) *Mind As Action*. Harvard University Press, Cambridge.

WERTSCH, J.V. and STONE, C.A. (1985) The concept of internalization in Vygotsky's account of the genesis of higher mental functions. In *Culture, Communication and Cognition: Vygotskian perspectives,* (Ed, Wertsch, J.V.) Cambridge University Press, Cambridge, pp. 162-179.

YIN, D.R.K. (2008) *Case Study Research: Design and Methods (Applied Social Research Methods)*. Sage Publications, Inc, Thousand Oaks.

ZINCHENKO, P.I. (1981) Involuntary memory and the goal-directed nature of activity. In *The concept of activity in Soviet psychology,* (Ed, Wertsch, J.V.) M. E. Sharpe, Inc., Armonk, New York, pp. 300-340.

ZINCHENKO, V.P. and GORDON, V.M. (1981) Methodological Problems in the Psychological Analysis of Activity. In *The Concept of Activity in Soviet Psychology,* (Ed, Wertsch, J.V.) M. E. Sharpe Inc., Armonk, New York, pp. 72-133.

ZINCHENKO, V.P. (1995) Developing Activity Theory: The Zone of Proximal Development and Beyond Context and Consciousness. In *Context and Consciousness: Activity Theory and Human-Computer Interaction,* (Ed, Nardi, B.A.) The MIT Press, Cambridge, MA., pp. 283-324.

Appendices

Appendix 1. US Coast Guard Compatibility Chart

CARGO COMPATIBILITY CHART (per USCG 46 CFR part 150)	REACTIVE GROUPS	Non-Oxidizing Min. Acids	Sulfuric Acid	Nitric Acid	Organic Acids	Caustics	Ammonia	Aliphatic Amines	Alkanolamines	Aromatic Amines	Amides	Organic Anhydrides	Isocynates	Vinyl Acetate	Acrylates	Substituted Allyls	Alkylene Oxides	Epichlorohydrin	Ketones	Aldehydes	Alcohols, Glycols	Phenols, Cresols	Caprolactum Solution
REACTIVE GROUPS		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Non-Oxidizing Mineral Acids	1		×			×	×	×	×	×	×	×	×	×			×	×		○	○		
Sulfuric Acid	2	×		×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
Nitric Acid	3		×			×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	
Organic Acids	4		×			×	×	×	×	○			×				×	×			○		
Caustics	5	×	×	×	×		○	○				×	×		○	○	×	×	○	×	⊗	×	×
Ammonia	6	×	×	×	×	○					×	×	×	×	○		×	×		×			
Aliphatic Amines	7	×	×	×	×	○						×	×	×	×	×	×	×	⊗	×	⊗	⊗	×
Alkanolamines	8	×	×	×	×							×	×	×	×	⊗	×	×	○	×			
Aromatic Amines	9	×	×	×	○							×	×							×			
Amides	10	×	×	×			×						×						○			×	
Organic Anhydrides	11	×	×	×		×	×	×	×	×											○		
Isocynates	12	×	×	×	×	×	×	×	×	×	×				○	○			○	○	×		×
Vinyl Acetate	13	×	×	×			×	×	×														
Acrylates	14		×	×		○	○	×	×				○										
Substituted Allyls	15		×	×		○		×	⊗				○										
Alkylene Oxides	16	×	×	×	×	×	×	×	×												○		
Epichlorohydrin	17	×	×	×	×	×	×	×	×												○		
Ketones	18		×	×		○		⊗	○		○		○										
Aldehydes	19	○	×	×		×	×	×	×	×			○								○		
Alcohols, Glycols	20	○	×	×	○	⊗		⊗				○	×				○	○		○		○	○
Phenols, Cresols	21		×	×		×		⊗			×										○		
Caprolactum Solution	22		×			×		×					×								○		
CARGO GROUPS		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Olefins	30	○	×	×		○	○	○	○														
Paraffins	31																						
Aromatic Hydrocarbons	32		○	×																			
Misc. Hydrocarbon Mixtures	33			×																			
Esters	34	○	⊗	×	○	○							○							○			
Vinyl Halides	35			×																			×
Halogenated Hydrocarbons	36	○	○	○	○	○		○					○										
Nitriles	37		×																				
Carbon Disulfide	38							×	×														
Sulfolane	39																						
Glycol Ethers	40		×										×										
Ethers	41	○	×	×	○																		
Nitrocompounds	42					×	×	×	×	×													
Misc. Water Solutions	43	○	×	○	○	○	○	○	○	○	○	○	×										

× : Incompatible Groups ⊗: Incompatible Groups with Exceptions * ○: Compatible Groups with Exceptions *

Appendix 2. Survey form, intuitive interaction ORCA 2016

Intuitive interaction in ORCA

Part 1 of 4

Dear respondent,
This questionnaire is designed to record your evaluation of the stowage system, ORCA. The questionnaire consists of 28 questions and answering them should take about 5 minutes.

The questionnaire is not designed to assess you or your abilities, but only your evaluation of this stowage planning tool, as an experienced and skilled worker. The information you provide will be kept anonymous and will not be able to be used to personally identify you.

(Questions marked with an * are mandatory)

Part 2 of 4

About you
Would you please tell us a little bit about your background. You will remain anonymous.

I have been using ORCA for *:
- I have just started
- half a year
- half a year - year
- more than a year

Age:

Highest Education:
- High school
- Vocational / naval school
- University college / Naval university college
- University
- Other

Years of stowage experience * :

Current position:

Part 1 and 2 of the ORCA survey

Part 3 of 4

Interaction with ORCA

These questions are related to how intuitive you perceive the ORCA user-interface to be in a general sense, ie. without using or utilizing your prior experience with stowage processes. How do you perceive the interaction with the program by relying entirely on the icons and other graphical screen elements in the user interface.

Note: There are no right or wrong answers. Please answer spontaneously and don't leave any questions out.

	Fully Disagree	Mainly Disagree	Neutral	Mainly Agree	Fully Agree
1. I can use the program without thinking about it.					
2. I achieve what I need by using the program.					
3. The way the program works is immediately clear to me.					
4. I can interact with the program in a way that seems familiar to me.					
5. I can easily fix problems if they should occur.					
6. The program is not complicated to use.					
7. I am able to achieve my goals in the way I expect.					
8. The program was easy to use from the start.					
9. It is always clear to me what I have to do to use the program.					
10. The process of using the program goes smoothly.					
11. I barely must concentrate on using the program.					
12. The program helped me recognise what I had to do to completely achieve my goals.					
13. It is clear to me straight away how the program is used.					
14. I automatically do the right thing to achieve my goals.					

Part 3 of the ORCA survey

Part 4 of 4

Perceived intuitive interaction in ORCA, based on your stowage experience.

These questions are related to how intuitive you perceive the ORCA user-interface to be, based on your knowledge of and experience with regular stowage processes, ie. how is your prior acquired experience and knowledge helping you in operating the program.

Note: There are no right or wrong answers. Please answer spontaneously and don't leave any questions out.

	Fully Disagree	Mainly Disagree	Neutral	Mainly Agree	Fully Agree
15. The ORCA interface uses the same 'images' as I am used to in the regular stowing process, therefore I can use the program without thinking about it.					
16. Because of my experience with regular stowage task flow I achieve what I need by using the program.					
17. ORCA works the way we usually do when stowing the ship, therefore it is immediately clear to me the way the program works.					
18. Since the program operation resembles regular stowage task flow, I can interact with the program in a way that seems familiar to me.					
19. If problems occur, I can easily fix them, in the same way I fix problems that occur during a regular stowing process.					
20. Since ORCA works the way we usually do when stowing the ship, the program is not complicated to use.					
21. The ORCA interface uses the same 'images' as I am used to in the regular stowing process, therefore I am able to achieve my goals in the way I expect.					
22. Since ORCA works the way we usually do when stowing the ship, the program was easy to use from the start.					
23. Since the program operation resembles regular stowage task flow, it is always clear to me what I have to do to use the program.					
24. Because of my experience with regular stowage task flow, the process of using ORCA goes smoothly.					
25. Since the program operation resembles regular stowage task flow, I must barely concentrate when using ORCA					
26. Because of my experience I could detect it when the program helped me recognise what I had to do to completely achieve my goals.					
27. Because of my experience with regular stowage task flow, it is clear to me straight away how the program is used.					
28. The ORCA interface uses much of the same 'images' as I am used to in the regular stowing process, therefore I automatically do the right thing to achieve my goals.					

Part 4 of the ORCA survey

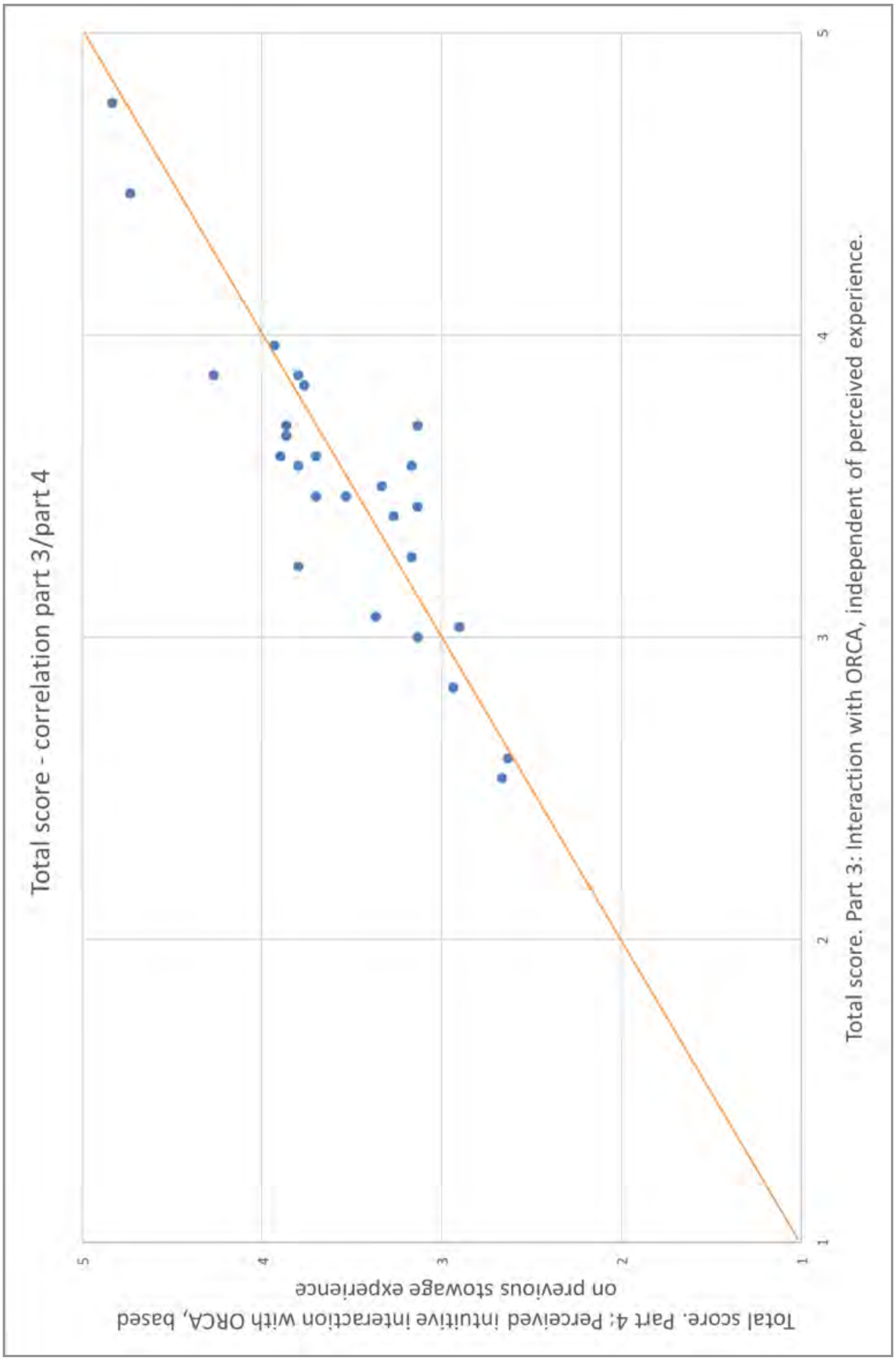
Appendix 3. Survey. Findings - questions 1-14

AVERAGE SUB SCALES - GENERAL					
Respondent	W	G	L	F	E
1	4,6666667	5	5	4,6666667	4,5
2	3,6666667	3,6666667	3,3333333	3	3,5
3	2,3333333	4	2,6666667	3,3333333	3
4	2,6666667	4	3,3333333	2,6666667	2,5
5	3,6666667	4	3,6666667	3	3
6	3,3333333	4	4,3333333	3,6666667	3
7	3,3333333	3,6666667	3,6666667	3,3333333	3
8	3	3,6666667	4	3,6666667	3
9	2,3333333	5	3,6666667	3,3333333	3,5
10	3,3333333	4	3	3	3
11	3,6666667	4	3	3,3333333	3,5
12	3,6666667	5	3,6666667	5	5
13	4	4,3333333	3,3333333	3,3333333	3,5
14	3,6666667	4,3333333	4	4,3333333	3,5
15	2,6666667	4	3,6666667	3,3333333	2,5
16	2,6666667	3,3333333	3,3333333	2,6666667	3
17	4,3333333	3,6666667	2,3333333	3	4,5
18	2	5	2,6666667	4,3333333	4
19	3,6666667	4	4	3,6666667	4
20	4,3333333	3,6666667	4	3,6666667	3,5
21	3,6666667	4	3,6666667	3,6666667	3,5
22	2,6666667	3,6666667	1,6666667	3	2
23	3	3,3333333	3	3,3333333	1,5
24	3,3333333	4	3,3333333	4,3333333	3
25	4,3333333	4	3	4	4
26	2,3333333	3	2,3333333	3	2

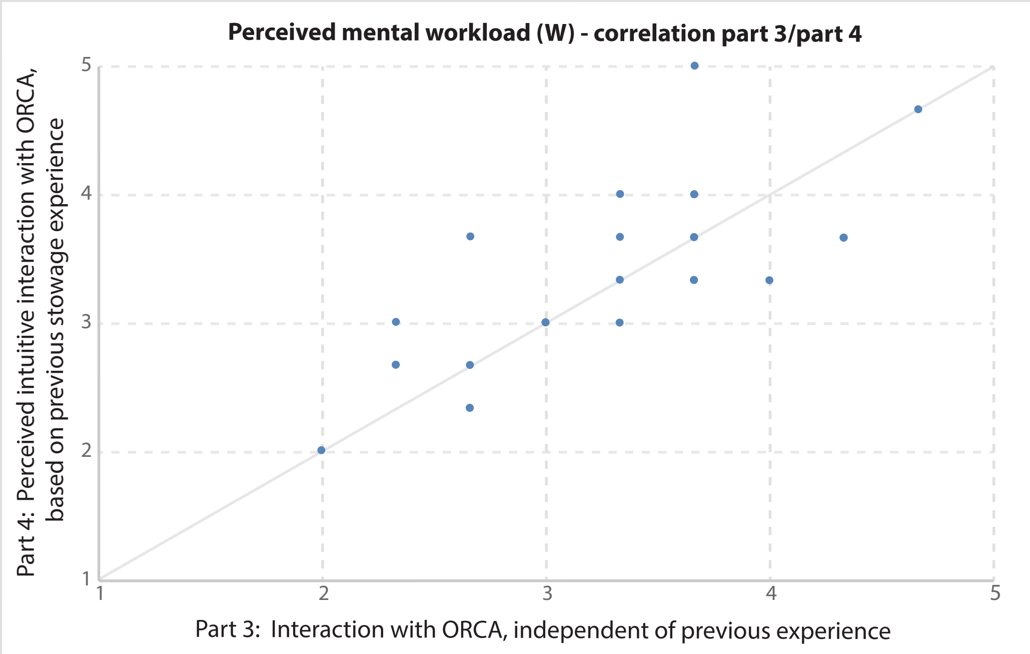
Appendix 4. Survey. Findings - questions 15-28

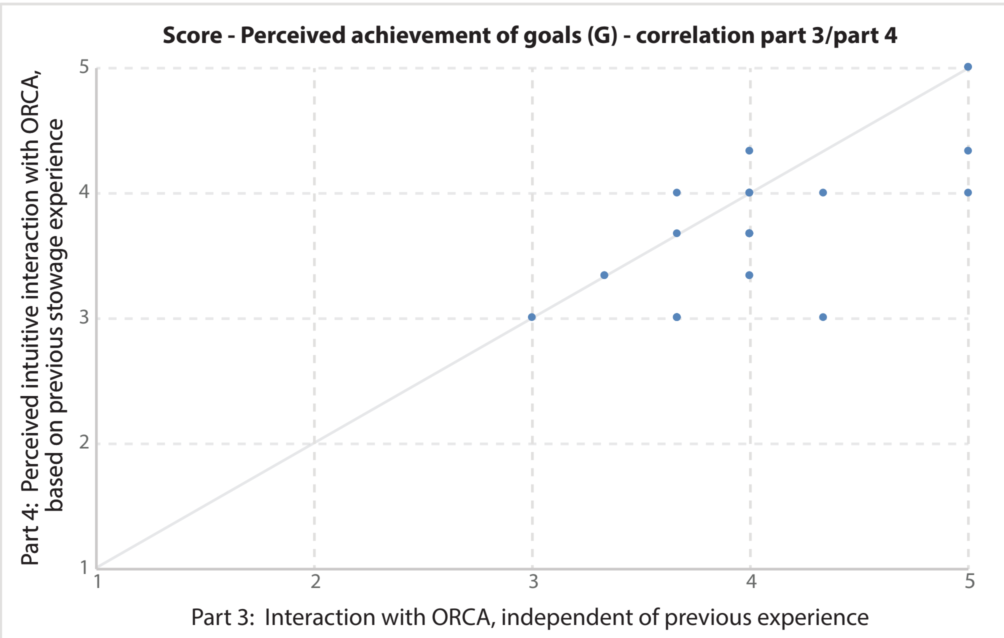
AVERAGE SUB SCALES - ORCA SPECIFIC					
Respondent	W	G	L	F	E
1	4,6666667	5	5	5	4,5
2	3,3333333	3	3,3333333	3	3
3	3	3,6666667	3,3333333	3,3333333	3,5
4	2,3333333	3,6666667	2,6666667	3,3333333	2,5
5	3,6666667	3,6666667	3,6666667	4	3,5
6	3,3333333	3,6666667	4,3333333	4	4
7	3,6666667	3	3,6666667	3	3
8	3	4	4	3,6666667	3
9	2,6666667	4	4	4,3333333	4
10	3	3,3333333	3	4	2,5
11	3,3333333	3,3333333	3,3333333	3,6666667	3
12	5	5	3,6666667	5	5
13	3,3333333	3	3,3333333	3	3
14	3,6666667	4	4	4	4
15	3,6666667	4	3,6666667	3,6666667	4
16	2,6666667	3,3333333	3,3333333	3,3333333	3
17	3,6666667	3	2,3333333	3,3333333	3,5
18	2	4,3333333	3,6666667	4	4,5
19	4	4,3333333	4,6666667	4,3333333	4
20	3,6666667	3,6666667	4	4	3,5
21	4	4	3,6666667	3,6666667	4
22	2,3333333	3	2,3333333	3	2,5
23	3	3,3333333	3,3333333	3	2
24	4	4	4	4	3,5
25	3,6666667	3,6666667	3,6666667	4	4
26	2,6666667	3	3	2,6666667	2

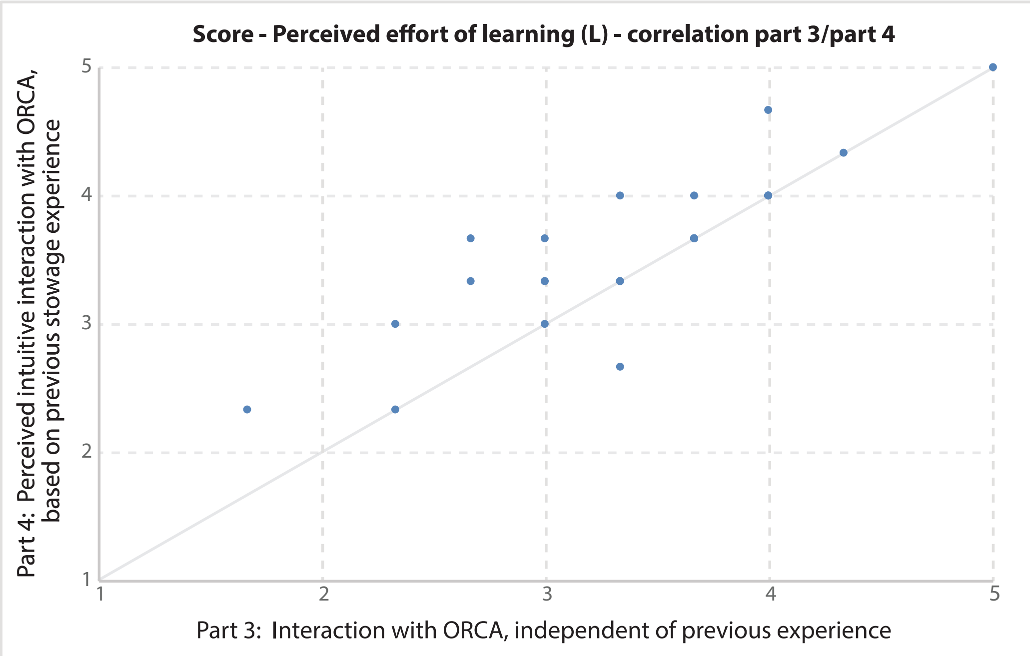
Appendix 5. Plot - correlation average values part3/part4

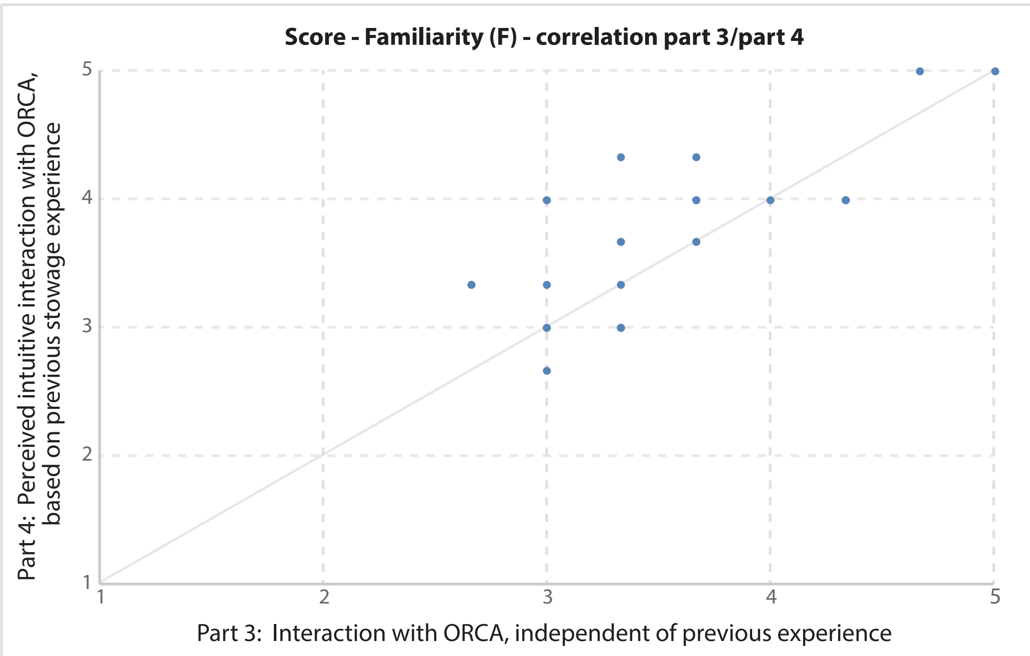


Appendix 6. Correlation patterns - Subgroups W-E.






Score - Perceived effort of learning (L) - correlation part 3/part 4
Part 4: Perceived intuitive interaction with ORCA, based on previous stowage experience
Part 3: Interaction with ORCA, independent of previous experience
1
2
3
4
5


Score - Familiarity (F) - correlation part 3/part 4
Part 4: Perceived intuitive interaction with ORCA, based on previous stowage experience
Part 3: Interaction with ORCA, independent of previous experience
1
2
3
4
5


Score - Perceived error rate (E) - correlation part 3/part 4
Part 4: Perceived intuitive interaction with ORCA, based on previous stowage experience
Part 3: Interaction with ORCA, independent of previous experience
1
2
3
4
5