Means of Coordination

PETER BØGH ANDERSEN[1], PETER H. CARSTENSEN[2] AND MORTEN NIELSEN[3]

[1] Center for Human Computer Interaction, Department of Computer Science, Aalborg University, Denmark, pba@cs.auc.dk, [2] The IT University of Copenhagen, Denmark, carstensen@it-c.dk, [3] Center for Human Computer Interaction, Department of Information and Media Science, Aarhus University, Denmark, mnielsen@imv.au.dk.

Abstract

Work is coordinated by means of many different media and codes: by means of oral or written language, by means of diagrams and maps, and by means of dedicated computer systems. The purpose of the paper is to set up a small set of dimensions that can be used to characterize and compare the different ways in which coordination can be effected.

The paper focuses on the relation between oral and artifact-based coordination. The former is often replaced by the latter when coordination complexity exceeds the limits of oral coordination. But the artifacts must still connect to the oral communication surrounding it. Therefore it is relevant to investigate the similarities and differences between the two distinct means of coordination, and to look for a general framework, encompassing both means of coordination, that enables us to understand how they may work together. We set off from two field studies: One of oral coordination as performed aboard the world's largest container carrier and one of a standardized form and a related protocol used in a software development project. We conclude by proposing four dimensions for classifying media and codes for cooperation. Two of them are pragmatic since the relate to the medium: *Persistence vs. Non-persistence* and *Passive vs. Active.* The other two are semantic, since they relate to the reference of the representations: *Process vs. State,* and *Field of Work vs. Work Arrangement.*

Keywords

Cooperation; Computer Supported Cooperative Work (CSCW); Coordination; Complexity; Speech; Coordinative artifacts.

1. Introduction

The coordination of complex cooperative work is an intricate matter that can impose a sever workload on the cooperating actors—so much so, in fact, that the magnitude of coordination work can become a general obstruction to work effectiveness, flexibility, or even safety. Therefore, systems for computer supported cooperative work (CSCW) are often engaged as a means of reducing the workload of the coordination activities. In order to arrive at a design that reduces the magnitude of coordination work, some of the coordination formerly performed manually by the cooperative actors will be made part of the CSCW system; that is, certain coordination functions will be supported or even fully automated by the computer system. Workflow systems could, for example, be considered a system that reduces the coordination workload by implementing a pre-specified protocol for the routing of information, and, by providing structured forms, telling the user what information he is expected to enter, etc. In order to establish a basis for designing such systems we need a coherent conceptual understanding of the coordination work conducted.

The CSCW literature is rich on studies illustrating the rich and seamless nature of coordination performed by co-located actors (Harper et al., 1991; Heath et al., 1993). All these studies show that the coordination of cooperative work can sometimes be achieved effectively and effortlessly simply by means of our everyday modes of interaction. Other studies have shown however, that oral interaction becomes increasingly labor intensive and sometimes even inadequate as a means of coordination in high complexity cooperative work (Carstensen and Sørensen, 1996; Schmidt and Simone, 1996). The latter studies highlight the need for artifacts that support the actors in coordinating the complexly interdependent cooperative tasks which, when carried out by means of oral interaction, tend to generate a very high coordination workload. When designing CSCW systems with the objective of reducing the magnitude of coordination work, it will therefore often be necessary to port coordination work from the modality of oral interaction to artifacts—this transformation is neither trivial nor well understood.

This paper discusses how oral and artifact-based interaction among collaborating actors is organized in order to reduce the coordination workload. We relate and compare means for simplifying the interaction applied for oral interaction to the structures and usage of artifact-based coordination mechanisms. From doing this we can identify several similarities that seem to indicate that it could be useful to approach both oral and artifact-based coordination interaction in a systematic and theory-supported way when aiming at understanding what goes on. Furthermore, we argue that it could be relevant and useful to aim at a general set for concepts and approaches for understanding coordination activities.

The process of moving coordination work from oral interaction to computer-based artifacts fosters complicated issues as to the relation between the two modalities of communication: How do cooperating actors coordinate by means of oral interaction and artifacts respectively? And how can we conceive of the transformation needed in order to arrive at a working system that does in fact reduce the coordination workload? The present paper seeks to contribute to the understanding of these issues by addressing the similarities and differences between dedicated coordinative constructs realized in artifacts and oral interaction, respectively.

We will start out by illustrating and analyzing a few fragments of oral coordination as performed in maritime operations on board the world's largest container carriers the M/S Sally Mærsk. We follow the work of the crew during passages of manual steering where the oral interaction between the actors serve to coordinate a host of complexly interdependent, time-critical, and highly cooperative tasks.

After this we dig into the organization and structuring of the testing and debugging of complex software. Here the coordination of the work of handling and correcting bugs is supported by a form and a set of associated procedures for the use of the forms. This is related to the conceptual framework of Coordination Mechanisms (Carstensen, 1996; Schmidt and Simone, 1996) founded on studies of coordination artifacts. The framework provides insight into characteristics and qualities of artifact designed specifically to reduce the workload of coordinating complex cooperative work.

In the last part of the paper we compare two types of coordination methods.

It is our hope and intention that the analyses of oral and artifact-supported coordination and the comparison between different media of coordination—in a wider perspective—will provide input for establishing a useful analytical approach to the analysis and modeling of coordination.

Before presenting the two field studies let us briefly introduce our basic understanding of cooperation and coordination: The field of CSCW has developed different understandings of cooperative work and coordination. In the following we will adhere to the notion that cooperative work is constituted of interdependent actors.

> Cooperative work is constituted by the interdependence of multiple actors who, in their individual activities, in changing the state of their individual field of work, also change the state of the field of work of others and who thus interact through changing the state of a common field of work (Schmidt and Simone, 1996, p. 158).

The concept of *interdependence* is at the heart of our understanding of cooperative work and coordination. Interdependence sets the limits for what we understand by the term cooperative work, and it shapes a basic explanation for why coordination is an inescapable aspect of cooperative work.

A group of people—a *work arrangement*—performing a set of tasks is only to be considered a *cooperative work arrangement* if the tasks performed are interdependent. Thus, the distinction between cooperative and individual actors cannot be made by focusing on the actors themselves but on the work they perform, cf., work related approaches like Strauss (1985), Carstensen, (1996), Schmidt and Simone (1996).

It is the structure of the work—the interdependencies between tasks—that binds the actors in a cooperative work arrangement together. The interdependent tasks and the world of objects and processes, within which they are performed, are referred to as the *common field of work* in an attempt to underline the difference between isolated work phenomena and work tasks that have a bearing on actors, and beyond the field of work of the individual actor.

2. Maritime Operations

The maritime operations with which we will be concerned in the following were observed during the most recent in a series of field studies concerned with time-critical cooperation and coordination as performed by the crew on board some of the world's largest container carriers. We have spent a total of four months on board three container carriers calling on 35 harbors on four continents.

The field work has generated a rich data material comprising a large number of video recordings, interviews, and technical documentation on the vessels in general and the instruments on the bridge in particular.

The data presented below stem from a European roundtrip performed on board the carrier M/S Sally Mærsk during spring 1999. The fragments of communication that will be analyzed in the paper have been transcribed from video recordings. The overall objective of the analysis has been to establish an understanding of the interdependencies between the cooperative actors, the need for coordination, and the nature of the actual coordination performed via oral interaction between the cooperative actors. In the following we shall focus in particular on the communicative structures engaged as a means of reducing the complexity of coordinating the complex task interdependencies of maritime operations.

2.1 M/S Sally Mærsk

The M/S Sally Mærsk is the largest container carrier ever built: She is 347 meters long, 42 meters wide, weighs about 85000 ton when fully loaded with a load counting some 7500 containers.

When traveling fully loaded and at maximum speed of 25 knots it takes about 16 minutes and a distance of 6.1 kilometers to bring the vessel to a complete stop. Even in ballast condition it will take 9 minutes and 3.2 kilometers to stop.

Fig. 1. Sally Mærsk approaching the harbor of Algeciras

Basically, the movements of a sea-going ship are determined by the interaction between controllable and uncontrollable physical forces. From this perspective, navigation is concerned with applying the controllable forces on the uncontrollable forces in a way that results in the desired movement of the ship; the objective being to transport the ship and its cargo between ports in a safe and efficient way.

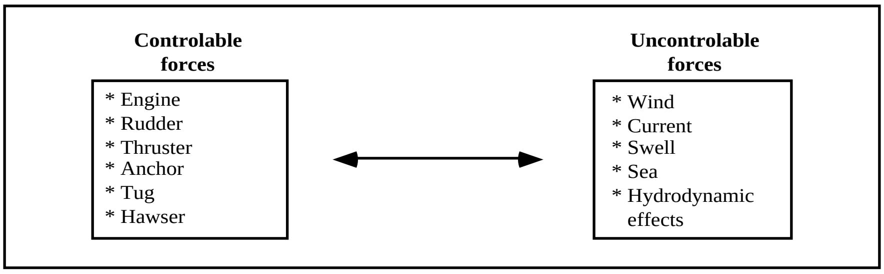


Fig.2. Controllable and uncontrollable forces determining the movements of the vessel.

Ships the size and like of Sally Mærsk have quite limited maneuverability. During the voyage it will often be the case that the uncontrollable forces work strongly against the desired navigation, and at times they might even be the most powerful. But even leaving aside the influence of forces like wind and current, the vessel still remains the victim of massive inertia—any change to speed or course takes time to build, and once the change is initiated it is hard to arrest or reverse.

2.2 Overall bridge layout

The bridge of the Sally Mærsk is fully housed by a glass and steel construction sheltering the delicate instrumentation and the actors from the elements. The center of the bridge is the main working area for the navigating crew; the bulk of instruments are located here. Most of the time, operations are performed from the center bridge, except when in and around the harbor basin where the vessel will be controlled via the instruments in the port- or starboard bridge wing.

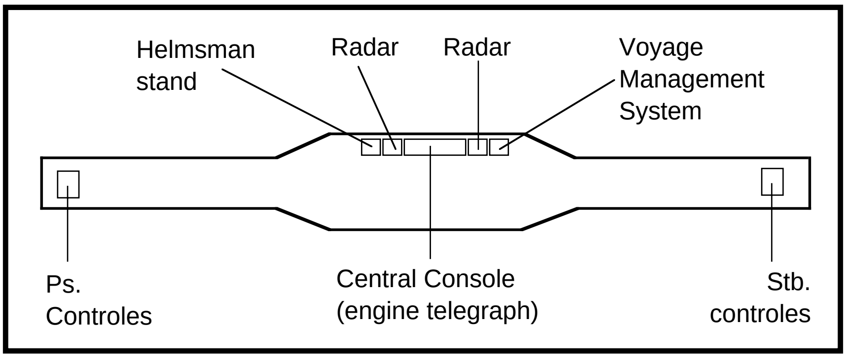


Fig.3. The overall layout of the bridge.

The bridge is fitted with four sets of manual rudder controls positioned on the central console, helm stand, and the starboard- and port consoles, respectively. Three engine telegraphs are available, one on the central console, and one in each bridge wing.

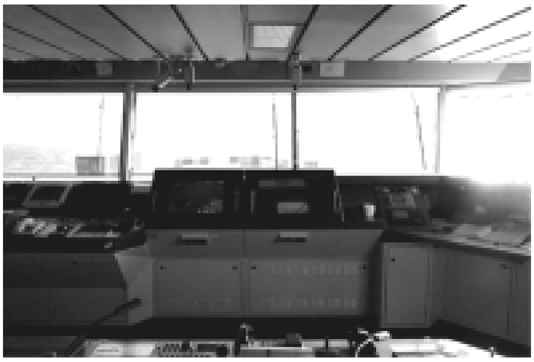


Fig. 4. The central console of the bridge.

During manual steering the maneuvering is manipulated hands-on by either the master (captain) or the helmsman. The master can choose to control both speed and course or he can engage a helmsman who will work the rudder from the helm stand.

2.3. Cooperative work and coordination in maritime operations

As mentioned in the introduction the concept of interdependence is at the heart of our understanding of cooperative work and coordination. It is the structure of the work—the interdependencies between tasks—that binds the actors in a cooperative work arrangement together.

Maritime operations involve many types of task interdependencies. Consider, for example, the general nature of the work-interdependency between the master of the Sally Mærsk and the pilot who assists operations when entering and leaving harbor areas.

The fragments presented below may, at first glance, give the impression that the pilot is in command of the vessel as it is he who issues the commands. The fact is, however, that the master is in supreme command of the vessel at all times—the pilot is granted the mandate of command but it can be reclaimed by the master at any time and without further ado.

The division of labor between the master and pilot is implemented in order to achieve the combination of two distinct specialties. The master knows his vessel inside out, while the pilot is the expert on the local waters. The pilot's job is to guide vessels safely in and out of the Rotterdam harbor, he does so with several vessels a day, and is backed up by a network of other pilots and the Vessel Traffic Management Service (VTMS) who are responsible for the overall logistics of in- and outbound traffic. The pilot carries a printed traffic plan and gets updated on changes via radio contact to the VTMS; the pilot knows who is coming in and who is going out, at what time, and what route the specific vessels are going to take. If direct communication between two or more vessels should be needed, the pilot on board the Sally Mærsk can contact his colleagues on the others vessel via radio to make arrangements beyond the general traffic plan. Just as important as his logistic knowledge is his experience in sailing the local waters of the harbor area. The harbor is narrow, water depth is limited, currents may run in unpredictable ways, and wind may change drastically between locations in the harbor—the pilot knows about these matters and is on board the Sally Mærsk in order to utilize his expertise for the safe handling of the vessel.

The pilot's weak spot is that he does not possess in-depth knowledge and familiarity with the individual vessel. The pilot works on many different vessels rather than one, while the master operates this one vessel across oceans and in and out of many different harbors. Safe and efficient operations during the approach to Europort involves the expertise of both the master and the pilot; the basic interdependence is a product of their work specialization and the fact that the pilot is giving the orders while the master carries ultimate responsibility for operations.

3. Communication as coordination

Maritime operations are characterized by significant shifts in the need for coordination. While operating in open waters the vessel is steered by one officer who will perform most tasks in solitude; he is alone on the bridge most of the time, and ship-to-ship and ship-to-shore communication is infrequent. However, when the vessel approaches a harbor and enters coastal waters, operations become highly cooperative. When operating in and around the harbor the master (captain), chief officer, helmsman, and pilot are working on the bridge; two groups of three and four actors—commanded by the first- and second officer, respectively—are positioned on deck; the pilot station, vessel traffic management service, and dockers are located ashore and communicate via VHF; tugboats and other vessels operate in the immediate surroundings of the Sally Mærsk and will at times become part of the cooperative work arrangement involved in maintaining safe operations.

Cooperation grows in line with rises in work complexity. When the vessel moves from open waters through coastal waters on to the waters in and around the harbor, work constraints become increasingly tight. Basically, maneuvers have to be performed with an increasing degree of precision, while there is less time to make the navigational decisions.

The tighter work constraints are reflected in the ways the actors communicate and coordinate. In and around the harbor area—where work complexity peeks—communication is about work only and proceeds according to well established patterns. It is this type of coordination—the oral coordination of complex cooperative maritime operations— that we shell be concerned with in the following. The theoretical basis for the analysis will be drawn from empirical and theoretical work on work communication (Andersen, 1997). We shall characterize instances of work related communication by means of the key concepts, *focus*, *background*, and *protocol*.

3.1 Focus and background

Consider the following situation. The Sally Mærsk is inbound for Rotterdam. There are four actors on the bridge, master, chief officer, pilot and helmsman. The auto pilot has been discharged and steering is performed manually. All are silent, then the pilot speaks: *starboard twenty*. The helmsman speaks: *starboard twenty*. And all are silent again.

1	Pilot	Starboard twenty
2	Master	Starboard twenty

Fragment 1: Rudder command

In any communicative situation, some features are taken for granted, whereas others are the topic of discussion. We shall say that the former constitute the *background*, the latter the *focus* of communication[1]. Background information is either not expressed at all, referred to via unstressed pronouns or definite nouns that recur identically during the conversation, or referred to via certain backgrounding constructions, such as adnominal adjectives and clauses.

In the communication between the pilot and the helmsman, the background information is completely left out. The pilot does not state who is the receiver of his command (the helmsman) nor which action is to be done (turning the wheel); he does, in fact, not state that it is a command at all.

Particular combinations of backgrounded and focused information types define specific communicative functions. The communicative functions can be defined in a very precise manner by considering a typical sentential schema used to interpret actual utterances[2].

Subject	Tense Aspect Modality	Verb	Object	Manner	Time, place
Someone	do	acting on	Something	in some way	sometime

Schema A. The basic sentential schema.

A sentential schema consists of a set of slots following one another in a more or less fixed sequence. Each slot can be filled with a particular kind of linguistic material, e.g. nouns, verbs or adverbials. Such sets have been called *paradigms* since Saussure's days. In the following diagrams we shall shorten "Tense, aspect, modality" to TAM.

Below a set of communicative functions are defined by different combinations of background and focus (boldface) paradigms (the functions are defined based on data from the Swedish Postal Giro, cf. Holmqvist (1989) Andersen (1997), pp. 379 ff.):

---

[1] Focus and background is well-known concepts in linguistics. They describe the information structure of the sentence, cf. the textual metafunctions in Halliday 1994.

2 The approach is mainly inspired by Halliday 1994. According to Halliday, utterances must simultaneously fulfill three main types of functions: *ideational* (how should we structure the topic?), *interpersonal* (how do we interact?), and *textual* (how do we make the text cohere?), cf. note 1. The schema tries to capture the combination of functional features that are relevant in the data. *Ideational* functions: the schema only covers *Doings*, i.e. material processes implying a change of state. Halliday's five other process types are not covered. The structure *Subject (Actor) + verb + Object (Goal)* derives from the process type of Doings. The adjuncts *Manner, Time* and *Place* are added since they turned out to be important in the data. *Interpersonal* functions: A slot for *tense, aspect* and *modality* is included since we are concerned with co-operation, i.e. regulation of interpersonal relations. *Textual* functions: the distinction between *background* and *focus* is a textual function. It is included in order to describe the reduction of complexity so important in complex work settings. Thus, the schema is an assemblage of features that are expected to be important in cooperative, physical work of the type we are dealing with. The schema may not be able to extract important features of other types of work, e.g. work mainly consisting of communicating – "sayings" in Halliday's terms.

Someone	should begin	acting on	Something	in some way	*sometime*

Schema 1. Work distribution: *who* should begin doing it *when*?

Someone	*is*	acting on	Something	in some way	now

Schema 2. Supervision: *is* she doing it?

Someone	*has*	acted on	Something	in some way	now

Schema 3. Worker Reporting: *has she* done it?

Someone	*has*	acted on	*Something*	in some way	now

Schema 4. Object Reporting: *has* it been done to *it*?

I	should	*act on*	*Something*	in some way	now

Schema 5. Work coordination: *what* should I do now?

She	should	act on	Something	*in some way*	sometime

Schema 6. Instruction: *how* should she do it?

Someone	*has*	acted on	Something	*in the correct way*	sometime

Schema 7. Control check: *has* it been done to *it in the correct way.*

If we consider the rudder command communicated between the pilot and the helmsman in terms of the basic sentential schema, only the *manner* slot is focused (Schema 8) the rest of the slots are backgrounded. Thus it belong to the instruction type.

Subject	TAM	Verb	Object	Manner	Time, place
Helmsman	should	turn	the wheel	*Starboard twenty*	Now, at the helm stand

Schema 8: Instruction. Helms man should turn the wheel *in some manner* now at the helmstand

The purpose of using background and focus paradigms is to reduce complexity. If the utterance is to specify an action that can be performed, then all paradigms must contain exactly one member that connects to work operations. One cannot make the ship sail “some” course. The course must be “001°, 002°, 003°, etc.”. So the purpose of the conversations is to fix all focus paradigms to exactly one of these. But if we can choose between, say 10 items in each slot, there are 10^6 possibilities to choose from, and we will never get around to do anything before the ship is grounded.

However, if we do neither have to bother with the subject (the helmsman does it), the time (it is now), the action (turn), the object (the wheel), complexity is reduced to the manner slot with a few degrees of freedom (the wheel can be moved 30 degrees to each side). Formally speaking, the technique of backgrounding reduces complexity exponentially.

Bridge work builds on a well established and predefined division of labor. One basic rule is that there is only one actor—the actor in command—who can issue commands. The other actors can suggest actions but they do not become commands before spoken by the actor in command. This division of labor can be defined as a backgrounding of the actor in command who is only focused watch shifts: “Will you take over now?”

When approaching a harbor, the command of the vessel is first passed on from the on-duty officer to the master (when initiating cooperative operations), and then from the master to the pilot during the final stages of the approach. The pilot receives command of the vessel shortly after embarkation and keeps it until the vessel is made fast at the quay. During this period the vessel is most often steered manually via one of four steering consoles. Only one console can be active at a time. The master has hands-on control of three consoles, while the helmsman operates the last console, the helm stand. If the helm stand is in operation and the pilot has command of the vessel, then the helmsman knows that he is the presumed addressee of a rudder command issued by the pilot. Conversely, if steering is performed from one of the three other steering consoles, the addressee is the master, etc.

Typically very few paradigms are focused. The most frequent are the *manner* (cf. Schema 8), the *time* and the *aspect/modality* paradigm. Many maritime conversations are about the *detailed parameters* for handling instruments, the *time* for doing an action, and whether an action is *imminent, executing* or *ended* (aspect). Fragment 2 shows an example with focus on time

1	C officer	… **half an hour before**
2	2. officer	**Half an hour before**; that is **a quarter to**
3	C officer	Yes, it will have to be, if the pilot is not boarding **until a quarter past**
4	2. officer	I guess so

Fragment 2. Focus on time.

Fragment 3 below contains utterances concerned with removing the hawsers and lines during the first stage of a departure from harbor. The fragment involves the following actors on the bridge: master, chief officer, and pilot. And the first- and second officer on the fore- and aft mooring decks.

The words in boldface signify modality (*can*), tense and time (*will have, before*), or aspect (imminent: *ready*, *let*; executing: ***are*** *wheel**ing***; ended: the present perfect, e.g. *all gone, (made) fast, singl**ed** up*, etc.). Note also time indications that follow a quasi *ready-steady-go* pattern also found in some sports.

1	C. officer	Yes, ok, thank you N – you **can** go fore and aft now [radio]
2	2. officer	We are **ready** aft [radio]
3	Master	You are **ready** aft [radio]
4	1. officer	We are **ready** on the Forecastle [fore mooring deck] [radio]
5	Master	You are **ready** on the Forecastle [radio]
6	Master	You **can** single up to one and one [radio to the officers on the fore and aft mooring decks.
7	2. officer	We single up to one and one [radio]
8	1. officer	One and one **fast** fore [radio]
9	Master	And that was one and one fore [radio]
10	Master	**All** singl**ed** up for and aft Sir
11	Pilot	Okay
12	1. officer	(...) fore hawser **gone** [radio]
13	Master	Fore hawser **gone** [radio]
14	2. officer	And the aft spring is **gone** [radio]
15	Master	Aft spring **gone** [radio]
16	Master	Yeah, I guess we **can** let him let him take the stern line (...), hanging in the forward spring
17	Pilot	Yeah, okay
18	Master	Just **let** the aft hawser go [radio]
19	2. officer	**Let** go the hawser aft [radio]
20	Master	You can wheel that in at the same time can't you [radio]
21	2. officer	yes [radio]
22	C. officer	He **will have** it wheeled in before, **before** they [the thrusters] get to it I guess
23	Master	Yes, just about so, it's not going **very fast**

Fragment 3. Orchestrating departure.

Schema 9 describes the sentential schema underlying Fragment 3. It is a combination of *reporting* and *supervision* and may be termed *Orchestrating*.

Subject	TAM	Verb	Object	Manner	Time, place
1. and 2. Officer +	**Preparing, starting,**	Letting go, pulling in.	Springs and hawsers		Fore and aft now

ship assis-tents	**executing, ending.**				

Schema 9. Orchestrating: will he do it, is he doing it, or has he done it?

In Schema 9, the hierarchy and the phase of the voyage has filled out all slots except the modality and time slots. For example, it is the right of the 1st officer to be located afore and not aft!

Such communication types, with many backgrounded paradigms, are normal in work communication (Falzon, 1983; Falzon, 1984) but it is always accompanied by another type, where suddenly the fixed background paradigms "thaw out" and become focus paradigms. What was previously taken for granted suddenly becomes the subject of discussion. The ensuing utterances increase in complexity since they contain more focus paradigms, and therefore open more choices (cf. the distinction between *language in* and *about a situation* (Halliday, 1978) and the *participant* versus *spectator* perspective (Holmqvist, 1989: 80).

Shifting from participant to spectator perspective is difficult but highly important, since it must happen when the assumptions hidden in the background paradigms are not valid any longer. The captain on the ship has this ability, as the following example shows.

Fragment 4 is from an approach to the Harbor of Rotterdam where the Sally Mærsk has been forced to perform a 360° turn in the deep-draft fairway some five miles of the harbor entrance. The maneuver is critical because no vessels can pass her while she is sideways in the canal. Traffic conditions are cleared prior to the maneuver. Yet, just after having initiated the turn, the Sally Mærsk is contacted on the VHF by the outbound Sealand Atlantic that will meet with Sally Mærsk in the narrow canal.

1	S.Atlantic	Sally Mærsk, the Sealand Atlantic [VHF]
2	Pilot	Sally Mærsk [VHF]
3	S.Atlantic	Yeah, good afternoon Captain, are you **turning to port** now, are you, over ? [VHF]
4	Pilot	Yes, I'm turning **slowly to port**, yes [VHF]
5	S.Atlantic	Okay, we, we are, we will be steering our course of about two nine zero, and we will stay to the north of you, if that is agreeable with you [VHF]
6	Pilot	yeah fine, I will be following the deep draft route outside [VHF]
7	S.Atlantic	Yeah, and can you give us **a red to red** passing, please, **port to port** [VHF]
8	Pilot	**port to port**, yeah fine okay [VHF, talking to the Sealand Atlantic]

Fragment 4. Focus on parameters.

The assumed background action is “passing head-on” and the focus paradigm contains two members, *red to red* and *red to green*, from which they have to choose. Sealand Atlantic chooses *red to red* because this is the normal choice. However, since Sally is turning port, the action will not be “passing head-on” but “overtaking” where *red to red* is a wrong choice.

Sealand Atlantic knows that Sally is turning port, but has not realized the consequences. The pilot does not realize the error either, but the captain does. He begins to focus the modality paradigm (how *can* we) and the previously backgrounded action slot (he will *overtake* us probably) and the Pilot quickly realizes his error *(not possible)* — although he does not admit to it!

1	S.Atlantic	Standing by zero two one six[VHF]
2	Pilot	okay [VHF]
3	Master	who was that?
4	Pilot	the outgoing ship, Sealand Atlantic, she want to pass **port to port**
7	Master	How, **how can** we do that? [pass the Sealand Atlantic port to port]
8	C. officer	(...)
9	1. officer	Yes
10	Pilot	We are steering around slowly, slowly
11	Pause	
12	1. officer	I presume he means that he will …
13	Master	Yeah, but **how can** he ... (...), he **will overtake** us probably ....
14	Pilot	Yeah, I don’t know
15	Master	...yeah, no, so um, **port to port**
16	Pilot	Sealand Atlantic, the Sally Mærsk [VHF]
17	S.Atlantic	Sally Mærsk, Sealand Atlantic [VHF]
18	Pilot	Um, you want to pass us on our **starboard** side, on the **north** side? [VHF]
19	S.Atlantic	Roger, Ill like to ...I, I heard you were **turning** to your **port** to go back in, I (...) in with the drencher, otherwise we are going **red to red**, over [VHF]
20	Pilot	I think **red to green** with us [VHF]
23	PilotSt	Sealand Atlantic ,Pilot Maas [VHF]
24	S.Atlantic	Pilot Maas [VHF]
25	PilotSt	The, the Sally Mærsk is **turning** to **port**, so you can proceed her north of her, over [VHF]
26	S.Atlantic	she is gonna **turn to port**, okay thank you, thank you, Sally Mærsk [VHF] [notice that the Sealand Atlantic thinks they are talking to the Sally Mærsk—when they are in fact talking to the Pilot Station Maas Approach].
27	Pilot	yeah (...) **port to port, red to red not possible**
28	Master	that is **not possible**
29	Pilot	his **red** side to our **green** side

Fragment 5. Shift of focus from parameters to modality.

Fragment 5 is a good example of how important it is that speakers are able to focus what was previously a background paradigm. On the one hand, it is necessary to reduce complexity by backgrounding information, but it is risky too if the backgrounded information is not valid.

The examples given above can be summarized into the following observations regarding oral communication and coordination:

- In oral communication, information is distributed in background and focus paradigms. This enables the actors to reduce complexity of their interaction.
- Coordination communication consists in closing the focus paradigms so that one executable member remains.
- Communicative functions can be defined by combinations of background and focus paradigms.
- Oral communication allows smooth changes between focus and background paradigms, enabling speakers to question tacit assumptions.

3.2 Protocols

As is well-known, communication is realized as a sequence of utterances governed by certain turn-taking rules of the form

(1) If someone is saying something to me, I should say something to someone.

An example from the maritime domain is:

(2) If I receive information (e.g. via VHF or intercom), I should repeat or summarize it in order to prevent misunderstandings.

However, other events than communicative ones are governed by conventions. In (3) it is a physical action that triggers speech, in (4, 5) a perception:

(3) If I have changed the controls on the bridge, I should tell it to my colleagues.
(4) If I can see something relevant which my colleagues cannot see, I should inform them of it.
(5) If I notice someone making an error, I should inform him of it.

Ronald Stamper and his collaborators (Chong & Liu 2000) claim that such rules — or *norms*, as they call it — are a vital ingredient in the proper functioning of work organizations. They set up two main types of norms, *behav-*

ioral norms and *intrinsic* norms, comprising *evaluative*, *perceptual*, and *cognitive* norms. (2)-(5) are examples of behavioral norms that have the form:

(6) If <condition> then <agent> is <obliged|permitted|forbidden> to perform <action|speech act>

An intrisic norm has the following form:

(7) If <condition> then <agent> adopts <an attitude> towards <some consequences or proposition>

Example:

(8) If someone is hindering me to do something then I *expect* him to notify me when the coast is clear.

Norms in our data are *heterogeneous* since general and specific norms are mixed; they are *semi-conscious* since some of them are written down and others are verbalized when they are broken; and they are not designed as a system, but have *evolved* through history.

The existence of norms can be verified in four ways: they can be abstracted from regularities in the data; they are verbalized when they are violated; they are taught explicitly during training; and some norms occur in the ship-owners manuals.

Norm (2) can be extracted from the data in Fragment 3 and norm (4) becomes explicit in Fragment 6:

1	Master	And call out when you are on the new course, K, right (…), when it's there you say one one five – then we know it's there
2	Helmsman	I did say so last time
3	Master	Well, I just did not hear it
4	Helmsman	(..)
5	Master	okay (...) that's fine – keep up the good work

Fragment 6: Violating rule (4).

Norm (8) surfaced when a ship was delayed in the berth which Sally Maersk was scheduled for; the ship left the berth without notifying Sally, which caused some grumbling aboard. (3) could be observed indirectly several times when the captain asked to pilot to keep his hands off the controls: this reason for this is that pilots forget to say what they are doing and therefore violate norm (3).

Finally, (5) occurs in the ship-owner's manual as the formulation: "subordinates MUST SPEAK UP when orders in their considered opinion are contrary to rules and regulation or will obviously give results contrary to the superior's intent". Thus, norms are real and can be empirically verified.

Norms work concurrently as well as sequentially. The former is the case where a single utterance or act is the result of more than one norm; the latter occurs where a norm adds a new action or utterance after the previous one. For example when the helmsman receives a course-order, (2) is responsible for his repetition of the order, (4) causes him to report when the new course has been reached, and (2) makes the master repeat the report. Sequential application of simple norms can generate long complex sequences.

Many of the maritime norms are rooted in the fact that all actors are obliged to take active part in monitoring the state of affairs in the field of work, the maneuvering of the ship. In this respect, the maritime field of work is common to all the cooperative actors—if a state change is performed by one actor, it is imperative that the other actors are informed of the change.

During work norms are only verbalized when they are endangered and therefore need maintenance. We have already seen this in cases of violation, but less dramatic examples also occur, e.g. when colleagues remind one another of their duties. In this case the crew often communicate about how to communicate: the norms are self-referential (Fragment 6,7 and 8).

1	C officer	You have to **call** the pilot on channel nine …[radio]
2	2. officer	Yes, I'm on it now [radio]

Fragment 7. Self-referential communication.

1	C officer	Have you **talked** to the pilot [radio]
2	2. officer	No, I'm about to **call** him[radio]
3	C officer	Then **ask** him what side [of the vessel] he wants the pilot-ladder on outside [the ladder that the pilot uses to get to the pilot boat when he disembarks]
4	2. officer	Yes, okay [radio]

Fragment 8. Self-referential communication

As has appeared from the preceding, norms are *modal*, i.e. they contain words like *is forbidden, allowed, possible, impossible, probable, expected,* etc. In some areas, norms are conventionalized: for example, the shape of a traffic sign indicates whether it prohibits, demands, or allows an action, or just gives information.

The norms for oral interaction have a number of characteristics:

- They are *semi-conscious*, i.e., normally not conscious, but can be verbalized and discussed in case of violations. Formulations can also be found in maritime education and in ship-owner's manuals.
- Norms do *not form a system*, but are a set of concurrently working, heterogeneous, and relatively autonomous guidelines that have evolved through time.
- Protocols can communicate about themselves (are *self-referential*).

A special subset of norms are those concerned with regulating cooperation between two or more persons. We shall use the term *protocol* to denote a coherent set of norms of this type. All the preceding examples of norms belong to a protocol.

4. Coordinating Artifacts

Let us now change the scene completely: We are at Foss Electric, a Danish manufacturing company developing, producing, and marketing equipment for, amongst others, measuring the compositional quality of milk, a highly specialized field. Development and production is localized in Denmark, sales, service and distributors are spread all over the world. The Foss Electric Corporation employs approximately 700 people.

Foss Electric has implemented concurrent engineering cf. e.g., Helander (1992) yielding integration between manufacturing functions throughout the development process and the organization is very much structured in terms of projects including specialists with competence in fine mechanics, chemistry, hardware and software design. The study to be discussed here concentrated on the software development in one large project: The S4000 project. It was a highly complex design: It included facilities earlier provided by several instruments, measurements of new parameters in the milk were included, speed was to be improved significantly, and it was the first product with an Intel-based 486 PC build-in.

4.1 Software testing and correction work

The software complex contained more than 200.000 lines of C-code and was organized in approximately 25 modules distributed in 15 different application. The S-4000 project involved more than 50 different people and lasted approximately 2 1/2 years. During the last 18 months the software design group had a stable size of approximately 8-10 designers, each having rather clear roles related to the design and implementation of one or more specified modules.

The most import roles related to the testing and correction activities included: (1) Software designers responsible for designing, implementing, maintaining, and correcting bugs in one or more of the software modules; (2) A group of three software designers called the *Spec-team* responsible for diagnosing reported bugs and deciding how to handle each of the bugs; (3) A Platform master responsible for managing and coordinating all the activities involved in integrating the outcome of one working period (called a "platform period"). He was, among other things, responsible for verifying the corrections of the software made by the designers, i.e., control that the reported bugs had been dealt with; (4) A Project plan manager responsible for maintaining a project plan spreadsheet; (5) Testers testing of the software embedded in the S4000 instrument; And (6) the central bugs file manager organizing and maintaining the central bug file, a ring binder containing copies of all reported bugs and organized according to their status. It is interesting to notice, that there was no software group manager during this one and a half year period.

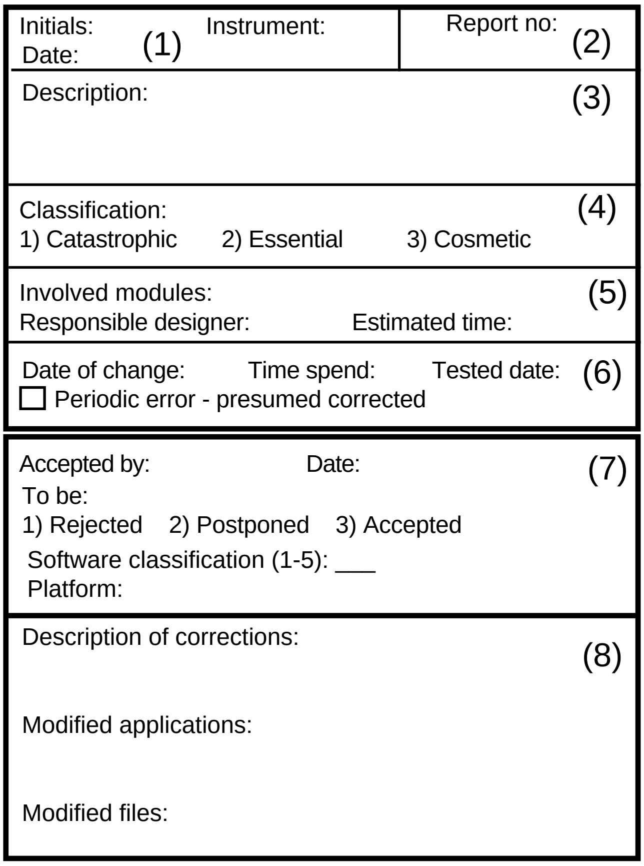
Early in the S4000-project the software designers realized problems in coordinating, controlling, monitoring, and handling the testing activities. They invented and used a standardized bug form that all testers had to fill in whenever they identified an error (a bug). To prescribe the use of the forms, a structured ring binder (being used as a central file) and a set of procedures and conventions for the use of the form were established. Some of the procedures were written down as organizational procedures, others were just conventions developed and refined during the project. It was also for the recognition of the mentioned problems that led to the establishment of the roles mentioned above. These were defined in order to establish the basis for running the procedures.

The aim of this section is to illustrate how the use of the form and concomitant work procedures were used in order to cope with the complexity of coordinating certain activities in relation to reporting, diagnosing, and correcting software bugs.

The development and correction work was organized in phases called "platform periods". A platform period was typically 3–6 weeks work followed by one week of integration. All the work and the plans were structured in relation to these periods. For each period a designer was appointed Platform Master responsible for collecting all information on updates and changes made to the software, and for ensuring that software was tested and corrected properly.

When a bug was identified, the tester filled in a form and sent it to the spec-team. The spec-team diagnosed the problem and decided which developer should fix the problem. The responsible designer were notified (by receiving a bug form), and estimated the correction time needed. When the problem was dealt with, the designer notified the Platform Master who could then verify the corrections.

At any state, the binder contained a copy of the form in its current status. The binder had seven entries reflecting the status of a specific bug: (1) Non-corrected catastrophic bugs (copies); (2) non-corrected essential bugs (copies); (3) non-corrected cosmetic bugs (copies); (4) postponed bugs (originals), (5) rejected bugs (originals); (6) corrected bugs not yet verified (copies); and (7) corrected bugs (originals). For each of the seven categories were the forms filed chronologically. The entries played a central role in stipulating the coordination by providing all involved designers and testers access to the state of affairs in the testing.


Initials: Instrument: (1)
Date:
Report no: (2)

Description: (3)

Classification: (4)
1) Catastrophic 2) Essential 3) Cosmetic

Involved modules: (5)
Responsible designer: Estimated time:

Date of change: Time spend: Tested date: (6)
☐ Periodic error - presumed corrected

Accepted by: Date: (7)
To be:
1) Rejected 2) Postponed 3) Accepted
Software classification (1-5): ___
Platform:

Description of corrections: (8)
Modified applications:
Modified files:

The actors fill (or add information) in:

The testers: (1), (2), (3), and (4)
The Spec-team: (3), (4), (5), and (7)
The designers: (6) and (8)

The procedure for handling bugs:

- A tester register and classifies a bug (field 1,2,3, and 4)
- The tester sends the form to the spec-team
- The spec-team diagnose and classify the bug (field 3, 4 and 7)
- The spec-team identifies the responsible designer (field 5)
- The spec-team estimates the correction time (field 5)
- The spec-team incorporates the correction work in the work plans
- The spec-team requests the designer to correct the problem
- The designer corrects the bug and fills in additional correction information (field 6 and 8)
- The designer sends the form to the central file
- The CFM sends the form to the PM and insert copy in central file
- The PM verifies the correction
- The PM returns the form to the central file

Fig. 5. A translated version of the bug form and the procedure followed when using the forms. CFM is central file manager and PM is platform master. The form is a sheet of A4 paper printed on both sides. The figure illustrates who fill in the information in the form.

The work plans were organized in a large spread-sheet containing information on: which tasks are to be accomplished and a reference to a detailed description of the task, the estimated amount of labor-time per module for each task, the responsibility-relations between modules and software designers, the relationships between the tasks and platform periods, and the total planned work hours per platform period for each software designer. The work plans were maintained by the Platform Master and a member of the spec-team.

The usage of the forms and the procedures can also be viewed as information flow between the involved roles and actors. The figure below illustrates the flow between the six different roles involved in coordinating the software testing and correction activities. In many situations the involved actors undertook more than one role.

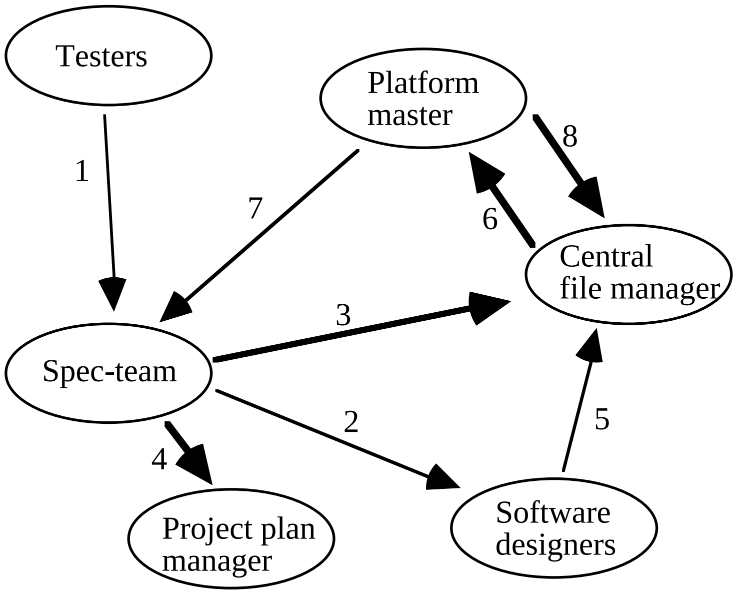


Fig. 6: A visualization of the roles involved in the software testing of the S4000 project, and the information flow between them.

The information flow described in Fig. 6 concerns only the stipulated (through organizational procedures) flow of the bug form mechanism. The thick arrows (3, 4, 6, and 8) indicate, that the flow is often a bunch of forms sent. The thin arrows indicate, that the forms typically are send one at the time. Other types of information are frequently exchanged between the actors and there might be situations were one of the actors choose not to follow the stipulated information flow.

4.2 Coordination of software testing and correction

Software testing and correction is a very complex and complicated task requiring a lot of coordination. In order to avoid redundant work, the testers need to be aware of each others work. The study showed, that it was very difficult for the testers to get an overview of reported errors, their diagnoses, correction status, etc. It was difficult for the actors involved in the S4000 project to determine the state of affairs in the software testing at a glance, and it was difficult for the testers and designers to communicate about the software complex and its status at a given point in time.

The coordination activities within software testing and correction were mainly based upon conceptualizations of structures in the field of work (e.g., the structure of the software complex) and structures reflecting the current implementation of the cooperative work arrangement (e.g., the involved actors, the working cycles, verification procedures, etc.). The conceptualizations ware, among other things, used to support the distributed bug registration activities, support the planning activities, monitor progress in correcting the software, monitor the state of affairs in general, and to simplify the needed bug classification and diagnosing activities.

Furthermore, aggregations of detailed information of the state of affairs (e.g., the total number of “not yet corrected category 2 bugs”) was used to support the coordination work, especially in order to simplify the required monitoring activities. Several structures for classification and categorization of bugs, corrections demands, and software modules was used. Concrete information from the software testing and development was also used when the activities were coordinated, e.g., the software code itself, the documentation, or the content of the bug registrations was used to decide the estimated correction time for a bug.

4.3 Coordination support provided by the bug report mechanism

It can, of course, be questioned whether the bug form mechanism genuinely has “eliminated” coordination complexity. The coordination to be conducted is as complex as always, but it can appear simpler to the actors through improving the representations of the work domain, by enforcing a specific behavior of the actors involved, by "automating" certain activities of the coordination, and by establishing a division of labor minimizing the need for coordination. The claim here is, that the bug form mechanism supported the coordination of software testing and correction by providing several of these things:

- The bug form provided a standardized information structure by which all bugs were described. By allowing information to be used for the diagnosis to be included in the form in a standardized pre-structured manner, the mechanism made it easier for the spec-team to find the relevant information. The classification of the bugs made it easier for the spec-team to deduce the testers’ perception of the problem reported. The form can thus be seen as improving the representation of the field

of work (the bugs) by establishing a "common standardized language". This makes it easier for the actors to interact. "Standardized languages" are, of course, problematic too. They constrain the actors, and it takes time for the actors to become familiar with them, and they need to be maintained.
- The standardized format of the form, furthermore, supported the work of reporting, both from tester to spec-team, from spec-team to designer, and from designer to platform master. This is because it forces a specific behavior on the actors. Through a specific surface representation the form supported the actors: No effort was required for considering which information to include.
- The bug form mechanism also supported the coordination activities by stipulating the work flow for handling the reporting, diagnosing, and correcting process. Although it was not completely automated, the pre-specified flow minimized the need for communication and interaction among the actors when handing over the form (and thus the obligations) from one actor to the next. The pre-specified flow (the embedded protocol) afforded support through constraining the actor: He could just apply the pre-specified routing without further considerations.
- Finally, the bug form mechanism was a central tool in the attempt to establish a well-understood and well-defined division of labor. By establishing different roles, and very clearly defining their responsibilities, the mechanism reduced the need for communication and interaction among the actors. All actors in the reporting, diagnosing, and correcting process knew exactly, what their obligations were. When they had dealt with their part of the treatment, they could just pass on the form, and others would take care of the rest. Their need for coordination was reduced.

Coordination activities, like monitoring the state of affairs, and negotiation of classifications, allocations, etc., were attempted to be supported by the bug form mechanism too. The establishment of the central file (the binder) including all registered bugs and their current status made it easier to get a coherent picture of the state of affairs. Although the testers and designers found it difficult to achieve an overview from the content of the binder, the benefit was that they only had to search in one place. Furthermore, the standardized information structure in the forms, and the standardized index of the binder, made it easier for the actors to find the relevant information, for example the classification and status of a bug or the number of not-yet-corrected category 2 bugs. Regarding negotiation of bug classification or resource allocation the bug form mechanism, and its related mechanisms, made classifications and resource allocations visible and accessible to the actors. The classifications and allocations became easier to discuss.

4.4 Artifact-based coordination

The example of the bug report illustrates that artifacts can be effective means of reducing the complexity of coordinating complex cooperative work. The issue of how artifacts can be instrumental in reducing coordination complexity has been thoroughly addressed in many insightful publications (Malone et al., 1987; Fitzpatrick et al., 1995). While inspired by the above mentioned, the analysis of the bug form mechanism has contributed to a conceptual framework called Coordination Mechanisms (Carstensen, 1996; Schmidt and Simone, 1996).

The aim has been to develop a set of concepts for analyzing, understanding, and modeling coordinative artifacts as observed in real world work settings. The framework addresses both analytical and constructive aspects of artifact-based coordination.

As briefly introduced in the introduction, our understanding—and the conceptual framework of Coordination Mechanisms—is based on an analytical distinction between the *cooperative work arrangement* and the *common field of work*, and a distinction between *work* and *articulation work* (Schmidt, 1994; Carstensen, 1996; Schmidt and Simone, 1996). The first analytical distinction is important when identifying *who* are collaborating (i.e., mutually interdependent) and with respect to *what* they are collaborating. The latter distinction is needed in order to distinguish *coordination* from the 'ordinary' work activities that will have to be performed no matter if the work is performed cooperatively or not. The concept of coordination mechanisms has been established in relation to these analytical distinctions. A coordination mechanism is a dedicated coordinative constructed—the coordination mechanism is aimed specifically at supporting the coordination needed in order to bind together the individual, yet interdependent, tasks performed by the cooperative work arrangement. Coordination mechanisms are objects having a physical form and they achieve a reduction of coordination workload by *stipulating and mediating* coordination. The coordination mechanism stipulates who is to do what, where, and when, while also mediating the information needed by the individual actors.

From the analysis of the bug form mechanism and the work on the conceptual framework a number of relevant characteristics of artifact-based coordination mechanisms have been identified:

- An artifact-based coordination mechanism is essentially a *protocol*. It is a set of explicit procedures and conventions that *stipulate* the coordination of the distributed activities. The distributed activities are coordinated by "executing the protocol".
- The stipulations of the protocol are (partly) conveyed by a *symbolic artifact,* i.e., they are persistent and publicly available in some kind of 'physical form'. Thus, a cognitive symbolic structure only existing in 'the head of the actors' is not considered a coordination mechanism.
- It is the symbolic artifact that *mediates* the coordination of the distributed activities. The artifact is an intermediary between the collaborating actors.
- An essential feature of an artifact-based coordination mechanism is that it has a *standardized format* that reflects pertinent features of the protocol. It thus provides affordances to—and impose constraints on—the required coordination.
- The state of the protocol embedded in the artifact-based coordination mechanism is *distinct* from the state of the field of work, i.e., it contains conceptual representations of aspects of the field work and the work arrangement. Changes to the state of the field of work are not automatically reflected.

5. Coordination mechanisms as reified oral communication

If we compare the communicative functions described in Section 3 to the functions realized in the bug-report from the previous section, we find most of them again.

Parts of the bug-report can be analyzed as a medium for conducting a sequence of coordinative communicative functions with different focus and background paradigms. In addition, the focus paradigms in one function becomes background in the following. As shown in Schema 10, the verb is always backgrounded to *correct;* (1) focuses on the module identification, (2) on modality (should/should not be corrected *(rejected/accepted)*, time *(now/postponed)* and manner (classification of bug: *how* should the correction be done?), and (3) focuses on the subject (*who* is going to do it). Finally, (4) is concerned with aspect (*has* it *been* done) and manner (has it been done *in the proper way*).

Subject	TAM	Verb	Object	Manner	Time, place
1. Someone	should	correct	**which module**	in some way	sometime
2. Someone	**should/**	correct	the module	**in which**	**when**

	shouldn't			**way**	
3. Who	should	correct	the module	in that way	at that time
4. Someone	**has**	correct**ed**	the module	**in the proper manner**	at some time

Schema 10. Coordinative functions in the bug-report in terms of foreground / background focus paradigms

If we compare the paradigms to the different combinations presented in Section 4.1 we can see that (3) is Work Distribution (Schema 1), (2) is a variant of Instruction (Schema 6) where modality is focused too (should/ shouldn't be done), and (4) is Control Check where tense (has it been done?) and manner (here "correctly") are focused.

The first function (1) is a new one (compared to those introduced in Section 4.1) where only the work-object is focused. Also the "seriousness-classification" (catastrophic, essential, cosmetic) lack analogs in the data of Section 4.1, although it resembles the Work Priority Ordering defined in Andersen (1997: 375).

We can conclude that communicative functions found in oral communication recur in coordination mechanisms. The bug-report collects a set of communicative functions that are normally applied after each other according to the oral protocol, and their sequence on the form to some degree stipulates their temporal sequence. For example, the Control Check in (4) will normally be performed after the Work Distribution in (3).

The bug report is thus a reified version of the oral coordination.

In our two examples, the main difference between oral and artifact based coordination is that the present state of the field of work, its history and possible futures are made persistent and thus publicly accessible. Whereas the software developers at any time can inspect the forms to ascertain how far work has progressed, what has already been done, and what might probably happen in the future, the situation awareness of the maritime officers must continually be reproduced and updated. Thus, the developers seem to be better off than the officers and one may wonder why persistent coordination mechanisms are not utilized more on the bridge: the only persistent mechanism we could find on the bridge were the checklists used before voyage start.

The reason is that there is another side of the coin. When coordination means become persistent they are more difficult to change and adapt to current conditions.

As stated in much of the work on coordination mechanisms (Carstensen, 1996; Schmidt and Simone, 1996), adaptability of coordination mechanisms is a desideratum for coordination artifacts too. Therefore the protocol of a coordination mechanism must be publicly available and malleable. Although the bug-report form obviously reduced part of the work-load of handling the testing and correction tasks, it was also clear from the study that situations occur where the protocol is not applicable, and that the flow-structure (and thereby the protocol) changed over time. The consequence—at least in the case of computer-based coordination mechanisms—is that much of the underlying protocol needs to be acquired and explicated in some form.

This particular facet of artifact-based coordination support has made the concept of protocol controversial within the field of CSCW research—and for good reasons. Schmidt and Simone discuss the notion of protocol from a sociological point of view and state that

> study after study have demonstrated, unambiguously and beyond any doubt, that the status of these formal organizational constructs in the actual course of work is problematic in that these constructs are impoverished idealizations when taken as representations of actually unfolding activities (Schmidt and Simone, 1996: 166).

One reason why persistent coordination mechanisms do not play any important role in maritime navigation was in fact already given in Fig. 2: the work consists in pitting the controllable forces against the uncontrollable forces of nature. Maneuvering is much more time-critical and less controllable than bug-correcting, and therefore the flexibility of the means of cooperation is more important.

This does not mean that maritime work could not benefit from coordination aids, but they will probably take the form of the “maps” that more loosely delimits the possible courses of actions. A simple example is bridge-layout: it is possible that coordination could be made much easier simply by designing the bridge as a common information space, implying that the individual instrument is not a one-person instrument but should be accessible to the whole crew. For example, one could introduce a large display to which the master could move that information that is crucial in the present phase of the voyage and which all ought to see clearly.

If we extend the shared field to work to include the sea, we can raise the question of cooperation between vessels. The simple rule which all vessels must obey is that no two vessels can occupy the same volume of water. A volume of water is a scarce resource which only one vessel can possess at a time. This necessitates cooperation in maneuvering. In order to help the captain guess the intentions of other vessels, selected instruments of foreign ships can be made accessible via transponder technology. For example, if Sealand Atlantic had had access to the rudder angle of Sally Maersk, the coordination problem in Fragment 5 might have been avoided. Similarly, access to a selection of the tug-boats instruments might be useful for the master during berthing, since the tugs are often so close that they cannot be seen from the bridge, and communication between tugs and pilot is often not conducted in English. Vessels in fact already try to make their intentions accessible to other ships: for example, changes of course should be so abrupt that other ships can perceive them as a course change.

A third possibility we saw at the VTS station at Bremerhafen. It is used by the pilots to prevent ships passing each other at dangerous bends of the river Elb.

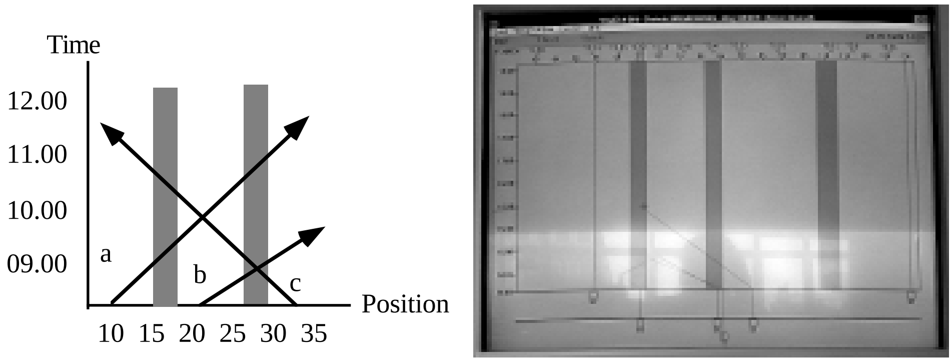


Fig. 7. Display used by the pilots at the Bremerhafen VTS station.

The y-axis represent time and the x-axis position along the river (in the actual systems, *tonnes* were used as units of measurements). The vectors represent the ship's future positions extrapolated by means of the present speed, and the shaded areas represent the sections of the river where encounters are prohibited. Vectors pointing left are ships sailing upstream, those to the right are downstream ships. The simple rule is that vectors crossing each other inside the shaded areas are prohibited, so *b* and *c* must be warned of each other, but no one else.

The Bremerhafen system is a good example of a map overlaid with a simple protocol. The map asserts true statements about the field of work by plotting vessels into a grid defined by *position* X *time*. The protocol is the modal element in the display, viz. the vertical columns denoting *forbidden* time-space areas.

6. Requirements for cooperative work

In this section we shall collect requirements for means of cooperation, irrespective of the medium in which they are realized. The requirements are extracted from our two field studies, and we shall indicate how they are be realized in the oral medium and in persistent artifacts.

There are four requirements:

(1) They must reduce complexity by dividing information into focus and background and by supporting standardized protocols.
(2) The protocols must be malleable.
(3) They must support focus-shifts.
(4) They must provide an overview of the state of the field of work and the cooperative arrangement

6.1. Reduction of complexity

The debugging process involved five different roles that would be assumed by different actors at specific points in time. The paper-based bug report form reduced the coordination complexity by stipulating the task responsibilities of each role, the classification of bugs, the intricate flow of the bug report form, etc.

Forms, checklists, and other types of artifact-based coordinative constructs enable a reduction of the coordination workload in that they represent a pre-computation of the central aspects of the coordination work. The bug report form can be described as a semi-structured form—a standardized format presenting the roles (actors) with captioned fields to be filled in, boxes to be ticked, etc.

In the oral maritime communication, the sentential schema constitute the structural basis for the elaborate backgrounding of coordination information reducing complexity exponential, the backgrounded slots being the oral versions of pre-computation. The schema represents a finite set of place holders without which one would have to cope with a much wider spectrum of information. Schema 11 is the sentential schema underlying the interpretation of a oral rudder command. In this case only the information in bold face has focus (is uttered), the rest of the paradigms remaining in the background.

Subject	TAM	Verb	Object	Manner	Time, place
Helmsman	should	turn	The wheel	**Port twenty**	now

Schema 11. Sentential schema and coordination mechanism

The bug-form also uses the technique of backgrounding since it too requires the designer to only address a few focus paradigms at each time step. However, the backgrounding is done differently. Whereas oral coordination often does not express the backgrounded information, coordination mechanisms display it explicitly. For example, in the bug-form the identity of the component to be corrected is not focused in the subsequent Work Distribution, but it is still explicitly represented in the form.

6.2. Malleable protocols

If protocols are not implemented in a form that is causally active, i.e. as a computer program, they must be continually maintained by the actors in order to stay in business. On the one hand, reproduction makes them vulnerable to errors, but, on the other hand, reproduction enables changes too, which means that protocols can evolve and be adapted to new circumstances. Oral coordination accomplishes this task be allowing self-reference, i.e. one can speak about the protocols by means of the protocols themselves (Fragment 7-8). The bug-form is not self-referential, since it contains no entries for changing the procedure of bug-correcting. In order to do this, the designers will have to resort to varieties of language that allow for discussing changes of cooperation. There are in fact examples of self-referential form-based protocols. In a study of the usage of a change request form in a large manufacturing company (analyzed as a coordination mechanism), Hans Andersen reports that the request form itself several times were used to change the protocol according to which the form should be used (Andersen, 1997). Inventing self-referential coordination mechanisms is therefore necessary if the mechanisms are to retain the malleability inherent in oral communication (see Stiemerling, & Cremers 1998: 303 for attempts in this direction).

6.3. Focus-shifts.

Since reduction of complexity entails that some information must be backgrounded, and since very few kinds of work — if any — are completely predictable, coordination mechanisms must support focus-shifts where backgrounded and presupposed information is re-focused. Focus-shifts are required in any kind of work (Bødker 1996) and coordination work is no exception.

Oral communication allows for smooth focus-shifts, as was evident in Fragment 5 that also provides the reason why focus-shifts are indispensable: cooperating people will misunderstand one another and therefore need methods for discussing and changing tacit assumptions. The bug-report did not support focus-shifts except in the sequence in which fields appear on the paper, and in that the actors can choose to use the fields for describing other aspects than stipulated.

6.4. Overview over the field of work

A major benefit of the bug-form is that it makes it easier for the developers to maintain an updated overview over the field of work: the information needed is in the same format and in the same place. In maritime operations, an updated overview of the state of affairs is just as important, but it is realized by people verbalizing what they do and see. Information which all need is routed via the intercom with loudspeakers in most locations: an example is announcement of the time when clocks should be reset because of a different time-zone; another the call of crew half an hour before their watch. Walkie-talkies are used to coordinate actions in bow and stern during berthing.

6.5. Dimensions of coordination methods

From the two cases-studies we can extract four dimensions relevant for coordination mechanisms. Two of them are pragmatic since they relate to the medium: we distinguish between *Persistent vs. Non-persistent* media, and *Passive vs. Active* media.

By a *persistent* medium we mean a medium that maintains its information over time. In non-persistent media, information is lost and must be recorded elsewhere. The bug-report is persistent whereas oral communication is non-persistent.

By *passive* media we mean media that cannot cause actions to happen by themselves, whereas *active* media can do this without human intervention. They are executable on some machine. Both oral language and the bug-report are passive media since both need human intervention, whereas the VTS system is active since the information about vessel positions changes without human intervention.

However, these two dimensions are not sufficient, since there is are differences between the VTS system and the computer-based version of the bug-report form outlined in Carstensen (1996). One difference is that the computerized bug-report contains a facility for automatically displaying the next step of the protocol with a default receiver filled in. Thus it explicitly refers to the next step in the work process, whereas the VTS system only indicates possible problematic situations, not the methods for handling them.

Another difference is that the state displayed by the pilot system is the state of the field of work — vessel positions in the Elb river — whereas the state displayed by the bug-report system is not the field of work, but the state of the work arrangement. The computerized bug-report does not provide access to the software system under construction, only to the allocation of tasks to actors and the timing of tasks.

In order to capture these differences we need to look at the semantics of the representations: do they primarily refer to a *State* or a *Process*, and do they primarily refer to the *Work Arrangement* or to the *Field of Work*?

The dimensions should be taken as continua, but sometimes one of the oppositions dominate. For example, the VTS system mainly denotes a state and mainly refers to field of work, with the stipulated coordinative tasks superimposed in the form of the shaded areas.

This tendency to assign a superordinate, structuring role to one of the terms can be re-found in conventional text genres. In *descriptive* genres, the state provides the structuring, and actions are only mentioned in connection with objects or locations. In narrative genres, things are opposite: here actions provide the general structure, and objects and locations are only entered when motivated by actions. In the former case we shall talk about *maps*, in the latter about *scripts* (cf. a similar notion in Schmidt 1999).

Let us now see how our examples can be characterized by means of these four dimensions:

	Pragmatics	*Semantics*
Paper-based Bug Report	+Persistent, + Passive	+ Work Arr, + State.
Computer-based Bug Report	+Persistent, + Active	+ Work Arr, + State, + Process.
VTS system	+Persistent, + Active	+ Fld of Work, + State
Object reporting	-Persistent, + Passive	+ Fld of Work, + State
Worker reporting	-Persistent, + Passive	+ Work Arr, + State
Work distribution	-Persistent, + Passive	+ Work Arr, + Process

Table 1. Classification of various means of coordination.

The paper-based Bug Report is classified as *+State*, since the protocol is only weakly mentioned in the form. In opposition to this, the computer-based version is [+ State, + Process], since the screen displays the current state of the cooperation as well as stipulations for future actions. The VTS system is a map of the Field of Work overlaid with information pertaining to cooperation [+ Fld of Work, + State].

If we look at the different communicative functions, they can also be classified according to the four dimensions. All share the features [-Persistent, + Passive], but whereas Object reporting *(has it been done to it?)* is [+ Fld of Work, + State] since it concerns the state of a work object, Worker Reporting *(has she done it?)* is [+ Work Arr, + State] since the topic is the status of the tasks of a worker. Finally, Work Distribution, *(who should begin doing it* *when?)*, is [+ Work Arr, + Process] since it concerns the assignment of a task to a worker.

Some of the concepts introduced above can be defined in these terms. For example, a *protocol* is defined by the reference of its representations: [+ Work Arrangement, + Process]. An *executable protocol* is a protocol with particular pragmatic properties: [+Persistent, + Active]. A *Map* is a representation in which the State is the structuring factor, and the Process must be inferred, whereas a *Script* is the opposite.

Finally, let us return to the concept of a Coordination Mechanism defined in Section 4.4. We can now define it as possessing the features [+Persistent, + Work Arr, +Process], so the computerized Bug Report is a coordination mechanism, whereas the VTS system is not.

We propose the term *Means of Coordination* for any representation that is used for coordinating actions in a common field of work. This is a pragmatic definition, referring to the use of the representation. Within this broad class of representations, the Coordination Mechanism is a special means of coordination that satisfy further pragmatic and semantic restrictions: it must be persistent and refer to features of the work arrangement.

7. Acknowledgements

This research documented in the paper could not have been conducted without the invaluable help of numerous people at Mærsk Sealand and Foss Electric. We would like to thank the hospitable people at A. P. Møller Copenhagen for hosting us, and we thank the crew of the Sally Mærsk for their considerable support during the field study. We thank Steven Verjans for useful comments. The research has partially been funded by the Center for Human-Machine Interaction (The Danish Basic Research Foundation) and the DIWA project (The Danish National Research Foundation). All errors naturally remain the responsibility of the authors.

8. References

Andersen, Hans H. K.: “Cooperative Documentation Production In Engineering Design. The ‘Mechanisms of Interactions’ Perspective”, Centre for Cognitive Informatics, Roskilde University, 1997.

Andersen, Peter Bøgh: *A theory of computer semiotics. Semiotic approaches to construction and assessment of computer systems*, (Paperback, 1997), Cambridge University Press, Cambridge, 1997.

Bardram, Jakob E.: “Plans as Situated Action: An Activity Theory Approach to Workflow Systems,” in J. A. Hughes at al. (eds.): *The fifth European Conference* *on Computer Supported Cooperative Work, Lancaster*, Kluwer Academic Publishers, 1997, pp. 17-32.

Bødker, S.: "Understanding Computer Applications in Use — a Human Activity Analysis". In *Signs of Work*, eds. B. Holmqvist, P. Bøgh Andersen, H. Klein & R. Posner. Gruyter: Berlin, 1996, 325 – 348.

Carstensen, Peter H: *Computer Supported Coordination*, Writings in Computer Science (No. 61), Department of Computer Science, Roskilde University, Roskilde, Denmark, 1996.

Carstensen, Peter H., and Morten Nielsen: "Guiding the Thrust! Ananlytical concepts in the service of coordination support systems," in R. Dieng at al. (eds.): *Designing Cooperative Systems. The Use of Theories and Models. The 5th International Conference on the Design of Cooperative Systems (COOP'2000), Sophia Antipolis, France*, IOS Press, 2000, pp. 131-146.

Carstensen, Peter H., and Carsten Sørensen: "From the social to the systematic. Mechanisms supporting coordination in design," *Computer Supported Cooperative Work. The Journal of Collaborative Computing*, 1996, vol. 5, no. 4, pp. 387-413.

Chong, S. & Kecheng Liu: "A semiotic approach to increase the design quality of agent-based information systems". In: Proc. of the 3rd Workshop of Organizational Semiotics, 2000, Univ. of Stafford.

Falzon, P. (). *Understanding a technical language.* Institut National de Recherche en Informatique et en Automatique., 1983.

Falzon, P.: "The analysis and understanding of an operative language," in B. Schackel (ed.): *INTERACT 1988*, North-Holland, 1984.

Fitzpatrick, Geraldine, Wiiliam J. Tolone, and Simon Kaplan: "Work, Locales and Distributed Social Worlds," in H. Marmolin, Y. Sundblad, and K. Schmidt (eds.): *Proceedings of the Fourth European Conference on Computer Supported Cooperative Work - ECSCW'95, 10-14 September, 1995, Stockholm, Sweden*, Kluwer Academic Publishers, 1995, pp. 1-17.

Halliday, M. A. K.: *Language as Social Semiotics. The Social Interpretation of Language and Meaning*, Edward Arnold, London, 1978.

Halliday, M. A. K. *An introduction to functional grammar*. London: Edward Arnold, 1994.

Harper, R. R., J. A. Hughes, and D. Z. Shapiro: "Harmonious Working and CSCW: Computer technology and air traffic control," in J. M. Bowers and S. D. Benford (eds.): *Studies in Computer Supported Cooperative Work. Theory, Practice and Design*, North-Holland, Amsterdam etc., 1991, pp. 225-234.

Heath, Christian, Marina Jirotka, Paul Luff, and Jon Hindmarsh: "Unpacking Collaboration: The Interactional Organisation of Trading in a City Dealing Room," in G. De Michelis, C. Simone, and K. Schmidt (eds.): *ECSCW '93. Proceedings of the Third European Conference on Computer-Supported Cooperative Work, 13-17 September 1993, Milan, Italy*, Kluwer Academic Publishers, Dordrecht, 1993, pp. 155-170.

Helander, Martin, and Mitsou Nagamachi (eds.): *Design for Manufacturability — A Systems Approach to Concurrent Engineering and Ergonomics*, Taylor & Francis, London, 1992.

Holmqvist, B.: “Work-language and perspective,” *Scandinavian Journal of Information Systems*, vol. 1, no. 1, 1989.

Malone, T. W., K. R. Grant, K. -Y. Lai, R. Rao, and D. Rosenblitt: “Semistructured messages are surprisingly useful for computer-supported coordination,” *TOIS*, vol. 5, no. 2, April 1987, pp. 115-131.

Schmidt, Kjeld: *Modes and Mechanisms of Interaction in Cooperative Work*, Risø National Laboratory, P.O. Box 49, DK-4000 Roskilde, Denmark, 1994. [Risø-R-666].

Schmidt, Kjeld: “Of maps and scripts: The status of formal constructs in cooperative work,” *Information and Software Technology*, vol. 41, 1999, pp. 319-329.

Schmidt, Kjeld, and Carla Simone: “Coordination Mechanisms: Towards a Conceptual Foundation of CSCW Systems Design,” *Computer Supported Cooperative Work. The Journal of Collaborative Computing*, vol. 5, no. 2-3, 1996, pp. 155-200.

Stiemerling, O. & A. B. Cremers. "Tailorable component architectures for CSCW-systems". In*: PDP ’98 (6th Euromicro workshop on parallel and distributed processing)*, 1998, Madrid, Spain. IEEE Press, 302-308.

Strauss, Anselm: “Work and the Division of Labor,” *The Sociological Quarterly*, vol. 26, no. 1, 1985, pp. 1-19.

Suchman, Lucy A.: *Plans and situated actions. The problem of human-machine communication*, Cambridge University Press, Cambridge, 1987.