

Elastic Systems

Peter Bøgh Andersen

Department of Computer Science/Center for Human Machine Interaction, University of Aalborg, Frederik Bajers vej 7, Aalborg, Denmark

pba@cs.auc.dk

Abstract: Maritime work is dynamical. As the ship sails, conditions change and so do the information needs. If instrumentation is to keep pace with these changing demands, it must be flexible and tailorable. Since, in addition, self-description and verbalization are important ingredients in maritime work, the system too must be able to present its inner state in a comprehensible manner.

Keywords: adaptation, maritime instrumentation, work analysis, anthropological research, multimedia

1 Introduction

This paper reports on ongoing research on maritime instrumentation at the *Danish Center for Human Machine Interaction*. The maritime domain is a very dynamic one, and this made us re-think basic assumptions in systems design. The paper starts by diagnosing the theoretical problem; Section 3 presents the empirical findings that made the problem visible. Section 4 draws practical consequences of the empirical studies, and Section 5 suggests a set of more radical consequences.

2 System and Usage: Computer Science

The system concept is both indispensable for designing computer systems and a perpetual nuisance. It refers to the idea that computer systems embody stable properties that constrain the user activities. They do it by defining the tools users apply to their objects of interest, and the properties that can be manipulated. Object-oriented programming captures the distinction very precisely: it consists of classes (the stable *system*) that work as templates for generating identical objects, i.e. objects with the same properties and actions. *Execution* of an object-oriented program consists in making the properties of the objects change state under the control of the actions of the objects (Madsen et al., 1993).

The distinction between system and usage coincides with a division of labor: systems are made by

software companies, whereas usage is the task of the customers.

A glance at contemporary commercial products reveals symptoms that something is wrong with this idea. Products are not stable but evolve via versions, often once a year.

The individual version is not that stable either: it offers settings or preferences that allow the user to adapt it to the current environment. Many systems allow new plugins to be added when needed and offer tools for tailoring the interface and for adding new functionality.

Finally, computer systems with a communicative function, such as EDI (Damsgaard & Truex 2000) and WWW (W3C, www.w3.org/ Consortium/), show a mode of development where local ideas for change have become a major factor in the development process vis á vis centralized attempts to standardization.

Thus, systems are more unstable than theory admits, and there are many more intermediate processes than developing systems and using them. This is thematized in life cycle research (e.g. Clarke 1996)

The situation is not unique to software but is also true of hardware systems. In our project I and my colleagues visited a number of ship bridges and were struck by the archeological layers that were detectable on older bridges. Outdated equipment (e.g. obsolete radars and manual machine telegraphs) co-existed with newer automatic and computerized technology.

Even the new technology reveals older sediments inside, as one descends through the layers of automa-

tion. For safety reasons, if a higher level technological layer breaks down on a ship, a lower level must be ready underneath with a handle the officer can use. Unlike a PC, ships cannot be stopped and sent to the repair-shop in a thunderstorm in the middle of the Atlantic Ocean.



Figure 1: Modern steering. The Interface of a modern VMS that controls the rudder via the Autopilot.

On one ship, maneuvering involved at least 5 layers: the Voyage Management System (VMS) (Fig. 1), the Autopilot, the Helm, the Non Follow Up Steering (NFU), a handle controlling the rudder machine in the engine room, and — if everything else failed — hawsers through the rudder pulled by capstans.

The latter method was the one used in the previous century. See Figure 2.



Figure 2: Old-fashioned steering. Rudder stem with tiller on frigate from 1860. The cordage is fastened to the tiller, runs through several cabins and is lead out through the ceiling to the deck where the helm is placed

Software systems too contain such layers: application interface, programming languages, assembler languages, and machine code. The higher levels are typically younger than the lower ones and build upon them.

We can conclude that reality exhibits much more complicated processes than users manipulating data

under the stable constraints of an application program. This motivates two questions:

- How should we understand these processes?
- Which consequences should it have for design of computer systems and their interfaces?

Before we answer these questions, let us consider how the same problem appears in another domain, namely linguistics.

3 System and Usage: Linguistics

Linguistics has been vexed by the problem of system and usage for the most part of a century. The dichotomy was launched a hundred years ago by the Swiss linguist Ferdinand de Saussure (Saussure 1993) under the headings of *langue* and *parole*. Saussure reacted to the historical tradition of the 19th Century that disregarded the systematicity of language. According to Saussure, the only stable scientific object of linguistics is the synchronic language system, whose parts mutually define each other. He considered language changes to be fortuitous and outside the reach of scientific methods. The consequence was a very peculiar way of thinking (structuralism) that spread to many sciences during the 20th century: we first divide the subject matter into *synchronic* and *diachronic* phenomena; then synchronic phenomena are divided into the collective system (*langue*) and the individual usage (*parole*). Thus, we end up with two kinds of unrelated times: the diachronic *language change* and the synchronic *language usage*. It is easy to see that computer science has not escaped this figure of thought either: language system = computer system, language change = system development, language usage = system usage.

In linguistics this created the following problem (Aitchison 1995):

- The causes of system change are found in usage, but since system change and usage belong to two categorically different time-scales, how can the latter influence the former?
- If the system defines the constraints under which usage must occur, how can usage influence the very system that determines its nature?

Thus, it appears that linguistics and computer science share the same problem, namely of being unable to conceptualize influences from usage to system changes. But whereas this problem has always been a nuisance in linguistics, it is only in recent years that it has begun to appear as a practical and theoretical problem in information technology.

Now, if Saussure and structuralism had not been around, how would we describe these facts in a simple way? I offer the following suggestion:

What we can observe is simply changes at different rates and involving different dimensions. What we used to call systems is slow changes, and usage is fast changes. Some parts of a system change faster than other parts. In language, the vocabulary changes every day, whereas syntax, morphology, and phonetics are slower. In the same way, the different versions of Word retain a large part of its basic functionality. The change only affects selected dimensions of the system, although the advertisements naturally want us to believe that everything is brand new. An important feature of object oriented programming, is in fact re-use of components (Madsen et al, 1993).

Just as "system-change" is really composed of more or less stable parts, so is usage. The pixels of the canvas of *Adobe Photoshop* change faster than the preferences or the collection of plugins.

The metaphor that emerges from this is a waveform with key note and overtones, the latter riding on top of the former. In complex systems not all changes are smooth. Like real waves, language and software may change continuously for long periods, and then suddenly enter "catastrophic" stages where accumulated changes are reorganized (Lightfoot 1979). For example, a new type of ship bridge has recently emerged that looks more like an airplane cockpit than a traditional bridge.

4 Maritime Work and Communication

In this section¹ I shall present an analysis of tasks and communication on the bridge. The analysis shows that the basic idea of work as a dynamic phenomenon consisting of many *concurrent* changes with different wave-lengths fits the maritime domain well. Maritime work falls into different *phases* that largely depend upon the waterways the ship passes through (open sea, traffic waters, harbor) and differ in information needs and cooperative arrangements. I shall add one more characteristic, namely the importance of *self-reference* (Luhmann 1984, 1990).

4.1 Self-reference

According to Luhmann (1984, 1990), self-reference is necessary for social systems to persist, and suc-

¹ The data are from three ferries, a freighter, a 3 weeks trip with a large container ship, and from two simulated voyages at the Danish Maritime Institute. It can be found on the center's web-site: <http://www.chmi.dk>.

cessful organizations are characterized by enhanced abilities to describe themselves.

This is very pronounced on ship-bridges. It is crucial that the crew maintains an updated overview of the current state of affairs (Carstensen 1998). For this reason, they are trained to verbalize their actions ("saying what they are doing") and this verbalization can in fact be observed:

- C (...) but it is just easier to, let [us] see, I can put it in remote here, and then give it (...)
- P Yeah
- C port ... port fifteen...
- P Oh, Yeah Yeah Yeah
- C and then it is easier to control and ...
- P Okay
- ...
- C I have put it on hand steering here

Text 1: Verbalizing own actions. C(aptain) informs P(pilot) that he will change from autopilot to hand steering. From container ship.

Another characteristic feature is the *ready-steady-go* structure of actions. They verbally prepare their collaborators, either colleagues aboard the ship or external actors, such as pilots, harbor authorities, or other ships, for impending actions.

- A And we are *ready* astern (radio)
- C You are *ready* astern (radio)
- ...
- D And we are *ready* at the bow (radio)
- C You are *ready* at the bow (radio)
- C Okay, we can *start* singling up
- P Yes Sir
- C You single up to one and one

Text 2. Ready-steady-go. Container ship leaving the quay.

Finally, communication in general is self-referential in the sense that one can talk *about* a language *in* the language. The reason is that communication, as well as actions, needs regulation, and regulation is effected by communication. Therefore, communication must regulate itself. In Text 3 we have communication about communication about communication!

- C I heard on the walkie that you *talked* about your having *talked* to pilot.

Text 3. Communication about communication about communication.

4.2 Phases

There are three main phases: open waters, trafficked waters and harbor. The phases are clearly marked by different types of cooperation and information needs. For example, in open waters one officer can handle the navigation, whereas in large ships two to five may be needed in the two other cases.

In one ship we visited the following sequence reflected a decreasing curve of tension: (1) When things are difficult inside the harbor, the captain handles the steering advised by the pilot and helped by the chief officer. (2) Then the helmsman is called in and receives commands from the captain. (3) Outside the trafficked area, the pilot leaves, and the watch officer replaces the chief officer. (4) The helmsman leaves, and (5) in open waters (in the daytime and in clear weather) only the watch officer is on the bridge. The sequence 1 5 is used when leaving the harbor, whereas the reverse 5 1 is used during entry.

Not only the work organization, but also the place of work changes with the phases. During berthing, navigation is moved to the bridge wing so that the navigator can see the ship's side. Each bridge wing must therefore contain a copy of the relevant displays and controls.

Information needs change as well. Outside the harbor, the radar is used to plot bearings and distances to other ships, but inside there may be too many vessels too close for this to be useful. In open waters, the GPS information of latitude and longitude is useful for determining the position of the ship, whereas bearings to landmarks or buoys are used in coastal waters. Inside a harbor, the importance of visual assessments of distances increases; some instruments increase their relevance as well, while others change their function. For example, drift and set become important during berthing. The radar is now used for measuring the distance to the quay, and the VMS for assessing the position of the ship in the basin.

The level of automation changes as well. The highest level of automation, the VMS, is only allowed in safe open waters, whereas manual steering by means of the wheel is required in trafficked and restricted waters and harbors.

Finally, the crew reconfigures the system as a part of normal routine, e.g. changes from 10 to 3 cm radar when approaching the harbor.

4.3 Concurrency

The phases consist in changes of a *set* of concurrent processes running at different paces, not in a change of a single process. The instrumentation provides

information about a subset of the concurrent physical processes. For example, the radar gives information about the relatively slow processes of change of position, speed, and heading of own and foreign vessels; the rudder angle indicator displays the movements of the rudder, and the R(evolutions) P(er) M(inute)-indicator follows the fast process of propeller revolutions.

On top of the physical processes we have the tasks and communication of the officers. Tasks can span from whole months (voyage planning and execution), over a couple of hours (approaching or leaving the harbor) to a few minutes (nudging the ship sideward to the pier by means of thrusters).

The individual actor can be involved in several communications at the same time. For example, the pilot will typically intertwine communications with the captain of the ship, the VTS station, and his pilot colleagues.

5 Immediate consequences for design

After having completed the field studies, we are entering the next phase of our project, namely suggesting useful ideas for maritime instrumentation. Our observations of the dynamic nature of maritime work makes the theoretical problem of relating system and use-features highly relevant: is it possible to design interfaces which officers can smoothly adapt to their changing information needs? Which guidelines are available for this task? Is it possible to replace the century old system-use dichotomy by something else?

In this section I shall present some non-controversial ideas for building user interfaces.

From the field studies, we know that processes such as the following routinely already happen to a limited degree:

- Changing automation level (VMS, autopilot, hand-steering)
- Exchanging sensors (radars, GPS, echo-sounders)
- "Moving" instruments (what really happens is not that instruments are moved, but that the crew moves to copies in the bridge wings).

In addition to this, we observed various types of re-design. For example, the *medium* was changed in one case where the auditory alarms were too difficult to distinguish and a supplementary visual display had therefore been mounted on the bridge.

In another case, the *display type* was changed: the captain wanted to save fuel by fitting the power of the engine to the speed actually acquired. To his dis-

posal he had various displays consisting of changing numbers. However, they were located far apart, and it was difficult for him to relate four or five changing numbers to one another. However, he found out that the engine information system offered trend curves instead, and that he could have curves of speed and RPM on the same display. In order to assess whether he got value for his money, he could just compare the slopes of the two curves. If both were rising, all was well. If the RPM rose more than the speed curve, he was wasting fuel. Thus, to our requirements we can add:

- Modifying individual instruments by changing medium or display type.

There were also cases where the available instrumentation caused problems that could not be alleviated; however, it did not stop the officers from suggesting solutions. The following conversation occurred on the tapes:

- C Because it has happened to me, where, where an officer phones down and says, but the ship can't steer, it can't maintain the course, and it takes a while before you realize what is wrong, right, and then it is the rudder limit that is set to five degrees.
- A Yes
- C Then it can't maintain steering ... the course, if it, if it begins to storm or something, or there are high waves.
- A It has happened to me, this there, you, you turn, and you don't feel that no shit happens.
- C No, nothing happens.
- A Well, then you have to go over to take a look, so...
- C Yes, yes, and it can also be the turn rate that is set incredibly low.
- A Yes
- C Or it can be the rudder

Text 4: Problems caused by wrong locations of rate of turn indicator, rudder angle indicator, turn limit and rudder limit.

The problem here is the placement of Rate Of Turn, Rudder Angle, Rate Of Turn Limitation, and Rudder Angle Limitation. Both rudder angle and rate of turn can be limited to not exceed e.g. 5 degrees as in Text 4. However, the two former displays were placed apart, and the limiters were knobs located on the autopilot that were difficult to see. A 1st officer solved the problem on a napkin (Figure 3: the bullets are the limitations).

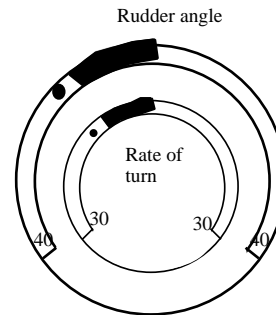


Figure 3: “Stand-up redesign on a napkin.

Therefore we can add the following requirement:

- Modifying instrument combinations, e.g. merging separate instruments into one.

To these tailoring facilities we can add the requirement that the behavior of the instruments should be verbalizable and therefore understandable. Otherwise introduction of computerized instruments would jeopardize the requirement for *self-reference*, i.e. maintaining a public overview of the current situation. We did in fact hear worries about the interpretability of the integrated bridges (“How do we know what is left when a component falls out”), and, according to Woods (1996), these worries are in fact typical for users of advanced automation. If automatic systems are “strong, silent, clumsy, and difficult to direct” (Woods 1996: 6) they generate confusion and errors.



Figure 4: Prototype enabling “Stand-up” redesign. Courtesy Henrik Garde.

In our center we have begun building simple prototypes that fulfill some of the requirements recorded above. Figure 4 shows one that would allow the 1st officer to realize his “stand-up” redesign, had it been

available on the bridge. The prototype allows the user to “merge” two or more instruments by clicking and dragging.

6 Radical Consequences for Design

We can sum up the requirements as follows:

- *The relevance of information changes.* There is a need to *resize* the displays, change their *prominence*, and adapt the *medium* and *style* of presentation.
- *The location of the work changes.* This creates a need to *move* instruments to other locations.
- *The level of risk changes.* This creates a need for moving safely between different *levels* of automation.
- *The work arrangements change.* We need instrument setups that *cooperative* work and others that support *single-person* work.

Besides these changes on the individual voyage, other needs for change arise when the ship is given a new function. For example, if a vessel is used as a ferry, the GPS is no longer as important as it were in open sea voyages. In fact, it is tempting to view a series of ships built on the same shipyard as a part of a very slow process of change, where the new ship is seen as a redesign of its predecessors.

6.1 Principles

In this last section I shall generalize these ideas to a more radical and systematic set of principles. Most of the principles have been suggested by other authors or experienced officers, or they can actually be found in an embryonic shape aboard real ships. Thus, what I do here is to bring order into a preexisting set of ideas pointing in the same direction. The basic idea was already introduced in Section 2, namely to replace the system-use dichotomy by the idea of a waveform of concurrent changes, some very slow, others very fast. The system-use dichotomy classifies one portion of the slow ones as system changes, and another faster portion as changes occurring during use. I want to view the two sections as a continuum.

If instruments are to be adaptable during the individual voyage, the following principles seem to be useful:

- *Interpretation:* What goes on between two internal objects of the system is similar to what goes on between the user and the interface-objects.
- *Peeking:* the user should have access to the way higher level components act upon lower level components.
- *Action:* Changing any object of the system is like changing the work objects of the system.

The *Principle of Interpretation* is motivated by the fact that modifying a system requires the user to understand what goes on inside it. The relationship between understandability and tailorability is nicely formulated by Stiemerling & Cremers (1998: 303):

On a very abstract level one could say, an adaptable application has to include a representation of aspects of itself. This self-representation needs to be manipulable and causally connected to the represented aspects, i.e. if the representation changes, the application changes as well.

In addition, the principle is motivated by the requirement of self-reference documented in Section 3.1.

The principle essentially requires the user to be able to conceptualize the internal interaction between system components in terms of concepts evolved in the use situation. This principle has been suggested by other researchers in tailorability (Malone, T. W., K-Y. Lai & C. Fry. 1995: 178):

More specifically, by tailorable we mean that end users (...) can progressively modify a working systems (...) without ever having to leave the application domain to work in a separate underlying “programming” domain.

The reason is that users do not want to burden themselves by learning a new domain on top of the normal task (Wasserschaff & Bentley 1997: 307).

I do not believe that exactly the same concepts that are relevant for using the system will be able to elucidate the whole inner logic of the system, e.g. that all aspects of maritime automation can be formulated in terms of maneuvering and navigation. A more realistic formulation may be:

- *Learning:* There must exist a continuous path, starting in the interface and ending somewhere inside the system, where each stage uses a slight modification of the concepts of the previous stage.

The *Principle of Peeking* was suggested by an instructor of training courses, and our data supports his idea: on a ship we visited, the VMS turned out to use too much fuel, and the reason was that it did not use the autopilot in the same way as an experienced navigator would do. It “kept to the line” and therefore used the rudder superfluously, whereas a human navigator would sail inside a corridor around “the line”. This discovery was made easier because the automation partially conformed to the Principles of

Interpretation and Peeking. The VMS sent course commands to the autopilot in the same way as the officer could do manually, and it was possible for the officer to see the course commands on the same display as he himself used.

If the steering automation is to be designed completely according to the Principles of Interpretation and Peeking, it would look like Figure 5.

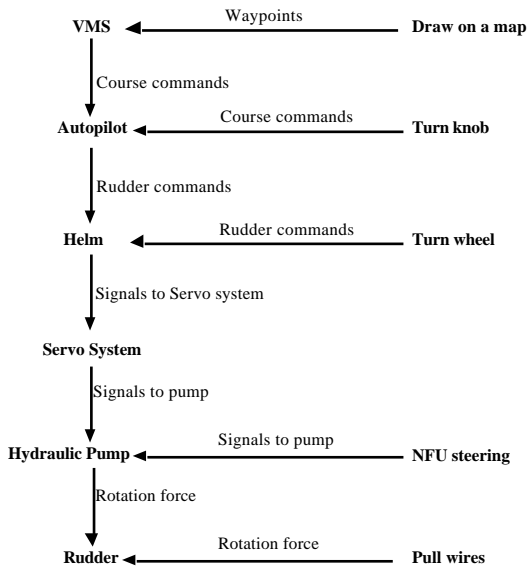


Figure 5: The principle of transparency applied to the steering system. *NFU* means “non-follow up”: the rudder keeps turning as long as the officer presses the *NFU*-button, whereas the signals from the helm go to the servo-system that moves the rudder to the desired angle.

Finally, the *Principle of Action* claims that manipulation of internal objects of the system should involve the same operations as does manipulation of its “normal” interface. A good example of this is found in *Microsoft Word 98* where the tool-bar is edited in the same way as the work-object, the text: by copying and pasting the item from a list onto the toolbar.

6.2 Architecture

The basic architecture of a system conforming to these principles can be captured by the following simple recursive rules:

- Panel {Object}*
- Object Interior + Interface
- Interior, Interface {Object}*

The opposition between an *Interior* and an *Interface* is repeated throughout the system in a uniform manner, and is no longer merely a top-level distinction.

To get a feeling for the consequences of these rules, consider the course-display in Figure 6.



Figure 6. Closed course-display object.

This object only shows its interface, but both its hidden interior and its interface can be clicked open and modified. If its interior is opened, a picture like Figure 7 displaying its inputs from the Gyro-sensors may appear.

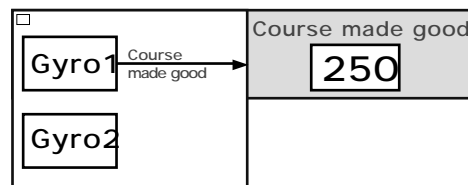


Figure 7. The interior is opened and the Gyro-sensor can be changed.

This facility is in fact already in use now, since the officer needs to be able to exchange gyro. What is new is that, according to the recursive rules, both the Gyros and the Interface can be opened too, inspected and possibly modified in some way. For example, if there is currently a heavy workload on the eyes, the Interface may be opened, inspected, and made to communicate the course made good by means of sound too (Figure 8).

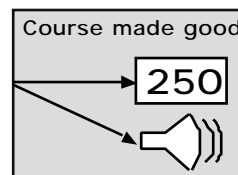


Figure 8. The interface is opened and sound modality is added.

6.3 Elastic interfaces

We are currently researching how such a structure could be practically designed. One immediate problem is that the possibility of “opening” the system for inspection to an arbitrary depth may be more confusing than helpful if it is based on overlapping windows, as we did in the prototype in Figure 4. The requirement to the display system is that it should maintain part-whole relations between the compo-

nents and clearly show the user his position in the hierarchy. Traditional overlapping windows do not do this, they destroy part-whole relationships. An attractive possibility is to use the notion of *elasticity* (Kandogan & Shneiderman 1996) or *fluidity* (Zellweger, Chang, & Mackinlay 1998; Chang, Mackinlay, & Zellweger 2000). Both concepts are based on the idea that when something is opened it requires more space; however, this space is not acquired by obscuring objects but by deforming them as far as it is possible.

The difference between these authors and our work is that “elasticity” is not seen as merely a good way of presenting hidden information, but is a property of the whole system. In addition, its ability to simultaneously maintain overview and present details must be highly valued in the context of maritime work where verbalization and collective situation awareness are mandatory for safety reasons.

7 References

- Aitchison, J. (1995), *Language Change: Progress or Decay?* Cambridge, Cambridge University Press.
- Andersen, P. B. (1999), Elastic interfaces. Maritime instrumentation as an example. *Proceedings of the CSAPC'99*, Université de Valenciennes, Valenciennes France, 35-41.
- Carstensen, P.H. & M. Nielsen (1998), *SAFECO WP II.1.2: Task analysis of navigator performance*. DMI/Risø National Laboratory.
- Chang, B-W., J. Mackinlay, P. T. Zellweger (2000), Fluidly revealing information in Fluid Documents. *Proceedings of Smart Graphics 2000 AAI Spring Symposium, Stanford University*.
- Clarke, R. (1996), The persistence of systems in organisations: Genre analysis of systems decommissioning in Holmqvist, B., P. Bøgh Andersen, H. Klein & R. Posner (eds.), *Signs of Work*, Berlin, de Gruyter, 59-107.
- Damsgaard, J. & D. Truex (2000), Binary Trading Relations and the Limits of EDI Standards: The Procrustean Bed of Standards. *European Journal of Information Systems*, 9(3)173-188.
- Hutchins, E. (1996), *Cognition in the Wild*. Cambridge, The MIT Press.
- Kandogan, E., B. Shneiderman (1996), Elastic Windows: Improved Spatial Layout and Rapid Multiple Window Operations, *ACM AVI'96 Advanced Visual Interfaces*, Gubbio, Italy, 29-38.
- Lightfoot, D.W. (1979), *Principles of Diachronic Syntax*. Cambridge, Cambridge University Press.
- Luhmann, N. (1984), *Soziale Systeme*. Frankfurt am Main, Suhrkamp.
- Luhmann, N. (1990), *Essays on Self-Reference*. New York, Columbia University Press.
- Madsen, O. L., B. Møller-Pedersen, K. Nygaard (1993), *Object-Oriented Programming in the BETA Language*. Addison-Wesley.
- Malone, T. W. K-Y. Lai & C. Fry (1995), Experiments with Oval: A Radically Tailorable Tool for Cooperative Work. *ACM Transactions on Information Systems*, 13/2, 177-205.
- Saussure, F. de (1993), *Course in General Linguistics*. London, Dockworth.
- Stiemerling, O. & A. B. Cremers (1998), Tailorable component architectures for CSCW-systems in *PDP '98 (6th Euromicro workshop on parallel and distributed processing)*, Madrid, Spain. IEEE Press, 302-308.
- Wasserschaff, M. & R. Bentley (1997), Supporting Cooperation through Customisation: The Tviews Approach. *Computer Supported Cooperative Work* 6, 305-325.
- Woods, D. D. (1996), Decomposing automation: apparent simplicity, real complexity in R. Paradruramen & M. Mouloua (eds.), *Automation and human performance: theory and applications*, Mahwah, New Jersey, Erlbaum, 1-17.
- Zellweger, P. T. B.-W. Chang, J. Mackinlay (1998), Fluid links for informed and incremental link transitions. *Proceedings of Hypertext'98*, 50-57.