Objectives, strategies and resources as design drivers

Bob Fields, Peter Wright and Michael Harrison

Human-Computer Interaction Group
Department of Computer Science
University of York, York, YO1 5DD, UK
{bob, pcw, mdh}@cs.york.ac.uk

ABSTRACT This paper builds on an existing model of human-computer interaction that focuses on the information structures employed as *distributed information resources* by users in deciding how to act. Such resources are not limited to internal mental constructs, but may be distributed between the single user and their environments, as representations in artefacts or the heads of other human actors. The concept of *interaction strategy* is introduced and used as a theoretical construct that generalises over a number of modelling perspectives in HCI. They are used to form a link between users' high-level objectives and resources and the way they're implemented. The usefulness in design of the general idea of distributed information resources and the concepts of strategies and objectives is illustrated by an example.

KEYWORDS Interaction design, interaction strategy, distributed cognition

1. INTRODUCTION

This paper uses a framework in which alternative theoretical perspectives on HCI modelling problems are integrated, and illustrates its applicability to designing interactive systems. An example is given from a particular domain: the construction of hypertext documents. The paper expands on the framework referred to as the *resources model* of human-computer interaction (Wright et al., 1996; Fields et al., 1996) by adding notions of "interaction strategy" and "objective", and showing their application in design.

The paper begins with the observation that there are a number of rather diverse ways of supporting apparently identical or similar tasks. In order to understand some of the similarities and differences between the various ways of supporting a task, the model presented here steps outside of the traditional task analytic explanations of work (for example, Kirwan and Ainsworth, 1992) and reflects the way designs support different ways of interacting or *interaction strategies*. The general architecture of strategy-based distributed cognition can be explained in terms of *information resources* and *objectives*. Resources include constructs traditionally considered as part of a human's internal cognitive processing (such as plans and goals); the difference is that resources may also be externally represented. Objectives correspond to the high-level purposes of users' work (for example, acquiring information about some subject could be an objective). In general, an abstract information resource can be represented or implemented by any of the (human or machine) agents in a cognitive system. Resources are treated as objects in a distributed cognitive system, rather than the controlling mechanisms of cognition.

Paradigms for modelling human-computer interaction such as situated action (Suchman, 1987), display-based interaction (Kitajima and Polson, 1995), and planning (John and Kieras, 1994) emerge as strategies in the resources view. Strategies and resources help the designer generate design concepts, thereby forming a bridge in the development process between a high-level view of users'

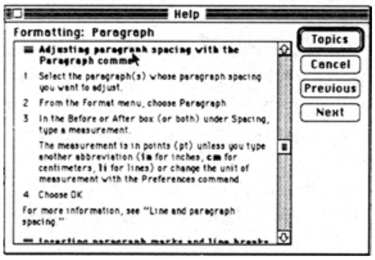
Human–Computer Interaction: INTERACT'97 S. Howard, J. Hammond & G. Lindgaard (editors)
Published by Chapman & Hall ©IFIP 1997

objectives and the lower level goals and other resources that may be considered as design objects. The ideas are illustrated by an example using the relationship between objectives, strategies and resources: the design of a hypertext information system.

A perspective that will emerge in the paper is that designing interactive artefacts is a process of deciding how *information resources* are to be *implemented*; this involves understanding the *objectives* underlying a user's activity and the *interaction strategies* that are appropriate to those objectives.

2. DESIGN CHOICES

It is clear that numerous design possibilities exist for giving users access to a system's underlying functionality. Consider as examples, two tasks: following a procedure described in the online help system of a word processor (Figure 1), and producing a chart using a computer spreadsheet package such as Excel (Figure 2).



Help

Formatting: Paragraph

Adjusting paragraph spacing with the Paragraph comme

1 Select the paragraph(s) whose paragraph spacing you want to adjust.

2 From the Format menu, choose Paragraph

3 In the Before or After box (or both) under Spacing, type a measurement.

The measurement is in points (pt) unless you type another abbreviation (**in** for inches, **cm** for centimeters, **li** for lines) or change the unit of measurement with the Preferences command.

4 Choose OK

For more information, see "Line and paragraph spacing."

Inserting paragraph marks and line breaks

Figure 1: Word help screen

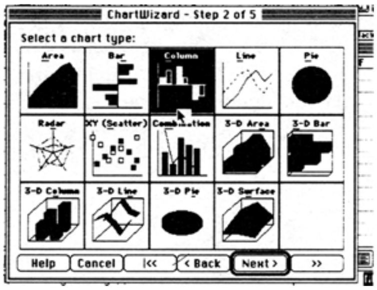


Figure 2: Chart type selection

Besides the obvious superficial differences between these two interfaces and the tasks they support, what are the differences in terms of how the user's interaction is constructed? In the first case, the actions that the user should perform are very clearly defined and are displayed as a procedure by the online help. In addition, the steps of the procedure may be memorised by the user and recalled as necessary. This is an example of what Hutchins, 1995, Chapter 7) refers to as a "pre-computed plan". In the second case, however, the user need not have a very clearly defined goal about the kind of chart to be produced nor a clear plan for how to go about producing it. The interface presents possible goals (types of chart) and ways the interaction may be moved forward (buttons and so on), obviating the need for the user to plan the course of the interaction in advance. This paper will argue that the difference between these two situations, and many others, can be captured by saying that each of the interfaces aims to provide support for a different *interaction strategy*. Different strategies require the presence of different information, and in the next section, we look at the information resources that allow users to follow strategies. The concept of information resources can be seen as laying out a space of possible design choices; objectives refer to the things a user will try to achieve using a design, and strategies, the means by which they go about doing so.

3. RESOURCES FOR ACTION

3.1 Resources

Instead of focusing on task knowledge the user may have, the resources model looks at the information, distributed *throughout* a system, that is needed to decide which actions to perform. A user may, in some situations, rely on knowing the current goal to select appropriate actions from those that are possible, while in others rely on knowing a pre-determined plan. The effects of actions in the current context or a record of past events can be similarly employed. Furthermore, there is no single place where all this information resides – some items are explicitly represented in external artefacts (the system interfaces, pieces of paper, and so on), some are external but implicit in the constraints of the dialogue, while others must necessarily be stored internally in the user's memory. A number of usability issues can be understood in terms of this assignment of *abstract information resources* to *concrete representational media*. A set of resources that play a role in shaping interaction is shown in Figure 3 and contains: *plans*, specifying actions to be performed; *goals* and sub-goals to be achieved; the *current state* of the world or interactive system; *historical information* about previous actions and what properties held of the state in the past; a model of the *effect* that actions have on the system; and the set of *actions* or "affordances" that the system currently supports (including constraints on the interaction arising from when it is legitimate to perform actions; for example, see (Zhang and Norman, 1994)).

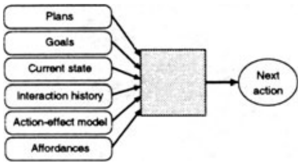


Figure 3: Information resources employed in interaction

Interactions consist of a number of steps, each being characterised by the set of resources that are available. In making a step, two processes are carried out by the joint human-machine system: determining a next action and doing it, and updating the required resources ready for the next step. Both are important from the perspective of design because the allocation of resources to representational media will affect how the processing is done. For instance, if a goal or a plan is implemented in the system's interface, then updating it can be performed by the system rather than the human.

In the early stages of understanding a task and designing to support it, no commitment need be made about where in the human-machine system the information resources and processing will reside. This is deliberate and allows us to talk about user activities early in the development process prior to the precise allocation of task information between user and system.

3.2 Implementation and distribution of resources

The resources described above are abstract specifications of the information required for a particular strategy of interacting. In any real situation, the resources must be represented or "implemented" in some physical medium. The expression of resources in media has a number of dimensions of variability: *distribution* (is the resource in the user's head or in the system interface?); *explicitness* (how is the resource expressed?); and *accessibility* (how much work is needed to "acquire" the resource in a form that can be used?).

Note that the relationship between these aspects of resources and the strategy that is employed is a complex one. The set of resources that are provided externally doesn't determine, in a simple way, the strategy that is adopted. The nature of the implementation is one crucial factor in this relationship.

A central part of the work of the designer is therefore to decide how the abstract information structures needed to support a strategy of interaction are to be implemented as concrete resources. The designer must decide whether a resource is to be represented externally (i.e., in the designed artefact), or whether the user must rely on their memory or that of another human agent. If the resource representation is to be external, the designer must decide whether it should be explicit (in that the user will be aware of it) or more implicitly (as is the case in plan-type structures embedded in Macintosh-style dialogues).

4. SUPPORTING STRATEGIES WITH INFORMATION RESOURCES

The concept of resources can go further than simply allowing us to describe particular interaction scenarios. The framework of resources allows a number of paradigms that have emerged in the HCI literature to emerge as what we refer to as *strategies*.

Plan following The "plan following" strategy requires the following two resources from the set identified in the previous section: a *plan* (with information about what actions are to be carried out, as well as an indication of which is the current item in the plan), and the *current state* of the work or the interactive system (which may be required in carrying out conditional parts of the plan). This is the kind of interaction described by the GOMS model (John and Kieras, 1994): plans correspond roughly to methods, and conditionals requiring a knowledge of system state are selection rules.

An example of support for this type in strategy is when a written procedure is followed, as in Figure 1, for example. This kind of implementation supplies the user with an external representation of the sequence of actions in a plan, though the "marker" showing the current item must be represented by the user. Halverson, 1994), however, describes a situation where the "marker" is separate from the actions of the plan and is externally represented.

Planning In contrast to plan following, the result of planning is to produce a plan (which can then be used in plan following). The input resources required are a *goal*, *system state* information, an *action-effect map* and *action affordances* for each stage of the interaction.

This strategy is often used in interactions with command line-based interfaces such as those found on unix systems: the goal resides only in the user's head; by default, very little resides in the interface, though actions' effects and status information (and some history) can be accessed by issuing commands.

Semantic matching In many situations, it is not plausible to claim that humans act by recalling and following previously formulated plans. Many of the studies of users of graphical user interfaces conclude that this is not how users act, and that they often have rather poor recall of the tasks being performed and the interfaces used (see, for example (O'Malley and Draper, 1992)). In the *semantic matching* strategy, the user decides what to do in a much more localised way by matching the effects of an action with the current goal, and checking to see if the resulting system state satisfies the goal. When the effects and availability of actions is represented externally in the interface, the strategy may be called *display-based interaction* (Kitajima and Polson, 1995). The information used to determine the next action can be a *goal*, the current *system state* information, an *action-effect map* and *action affordances*.

Many Macintosh style direct manipulation interfaces (like the one shown in Figure 2) provide support for this strategy by providing, in the interface, representations of the commands that are available and the effects that they have. Planning, however, is not so well supported because of the complexity of interactions and the contingency of action effects of the current state.

Goal-directed exploration In some situations the user may not be able to follow a plan or perform semantic matching of actions to goals. In such situations, users may, either through lack of information (about plans or actions), or in response to excessive cognitive demands, revert to an alternative strategy. A very simple sounding strategy is to act in an exploratory way by "trial and error". This involves selecting actions, possibly even at random, from those that are available, and checking to see if the resulting system state satisfies a goal. If not, then another action is tried.

Maximally, a trial and error strategy for selecting actions can require a goal and system state (to determine when the end-point of the interaction has been reached), the set of actions available at each point, a history of actions performed (to avoid going around in circles), and possibly a partial model of actions' effects (so as to rule out some actions as being unhelpful or dangerous).

Learning by exploring The above strategy, although it might seem rather inefficient, is one of the ways in which users can learn about their tasks and systems (Carroll and Mack, 1986). In addition to selecting next actions, the outputs of this strategy can be new plans. By reflecting on the history of a successful interaction, a plan can be developed and used in a future plan following interaction. Alternatively, exploration can help to enrich the user's model of actions' effects, which can be used in future semantic matching interactions.

This kind of exploration can be captured as a strategy in its own right, requiring a system state, actions' availability, and a history of which actions have already been performed. The product of such a strategy is an enriched model of actions and effects, as well as actions that are performed.

An example of learning or achieving goals by exploration is in complex, but highly structured, interactions such as with a mathematical theorem proving assistant system (e.g., the Jape system: Bornat and Sufrin, 1996)). The user can often learn, or even make progress by a trial and error exploration, which can move from an apparently hopeless state to one where another strategy will bear more fruit.

5. FROM OBJECTIVES TO DESIGNS

The previous sections have sketched out a framework for constructing "broad brush" models of human computer interaction. This section looks at one way these ideas may be used in design and the next one shows how the ideas are being used in the design of a hypertext document.

5.1 Objectives

A common starting point for interactive systems design is to understand the high-level objectives of the work being carried out. After deciding which aspects of this work are to be computerised, a process of task decomposition is used, eventually reaching a level where the sub-tasks can be supported by features of the interface.

A suggestion of this paper is that different high-level objectives of users may be best supported by different interaction strategies, and that design can progress from high-level objectives to lower level designs by way of a consideration of strategies and resources (see Figure 4).

To illustrate the importance of thinking about interface design in terms of user's objectives, strategies, and resources and their distribution through the cognitive system, consider the following three activities for which a user may make use of a hypertext system: getting an overall idea of the subject matter; secondly, answering a specific question; and learning about a highly structured collection of information. Each can be achieved most effectively by a different strategy: browsing and querying can both be thought of as goal directed, so appropriate strategies may be variants of semantic matching. Learning can be supported by a tutorial-style "tour" based on a plan embedded in the interface. Section 6 follows these ideas through in a little more detail.

5.2 Designing the external environment

As mentioned earlier, when viewed from the perspective of the resources model, part of the process of designing interactive systems involves making decisions about which strategies are to be supported, which resources are to be represented, and precisely how they will be represented. The process of doing design is therefore one of deciding which resources are to be embodied in system agents, and how they are to be implemented.

As we begin to move the ideas of the resources model towards a set of concepts that can be used in design, it is important to emphasise what the scope of design activity is in this context. In deciding how to provide resources to support an interaction strategy, the designer is engaged in designing the *external environment*; the designer cannot mandate what is represented internally by a user of the system. This fact might seem to limit the usefulness of the resources model (or indeed any similar activity) as a tool for thinking about design. However, two implications of the design of external artefacts rather than internal structures suggest ways in which the model can deliver benefits to the design process.

Firstly, although one cannot design the structure of users' knowledge, a resources model captures assumptions about what the user must know for a particular strategy to work. This has the effect of recording requirements that are placed upon a human agent in order to be able to carry out a particular strategy.

Secondly, we want to design not just an interface, but also an "ecology" for the development and evolution of strategies. In other words, a designer may anticipate the ways in which the strategies used may change over time, as a result of resources becoming internalised by the user. A simple example of this is where a resource represented in the interface (such as the "plan" in Figure 1) becomes, over a period of time, internalised by the user. The abstract information resources used to carry out a strategy may remain the same, but their physical location changes, with a consequent impact on performance (cf. Hutchins, 1995).

A more interesting example occurs when a plan is being followed. If the user interface provides sufficient information about the system state as it changes during the plan's execution. Over time, the user may internalise a model of the effects that actions have. On it's own, that may not be particularly useful, but if combined with a representation of which actions are available at any point in time, the user may be able to make a transition to a semantic matching strategy.

6. A DESIGN EXAMPLE

The ideas of strategy and information resource have been applied to the design (and implementation using the SuperCard authoring system) of a hypertext document describing certain aspects of the process of safety analysis for critical interactive systems. The target user population is composed of engineers who design and evaluate such systems. The hypertext is intended to provide support for a range of activities and objectives the users engage in, including:

1. *browsing* to find general information;

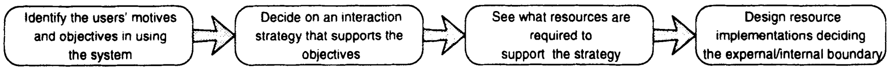

Identify the users' motives and objectives in using the system
Decide on an interaction strategy that supports the objectives
See what resources are required to support the strategy
Design resource implementations deciding the expernal/internal boundary

Figure 4: From high level objectives to designs via strategies and resources

2. *learning* in detail how the analysis process works;

3. *querying* to find particular information.

The view taken in this paper is that a single cognitive perspective provides a rather narrow basis for designing interactive technology and that the concept of strategy allows the modeller to generalise over a number of paradigms. In this section we aim to show how such a wide-ranging view can yield some benefits in design.

In HCI in general and hypertext design in particular, many authors have sought to provide advice to the designer in the form of principles and guidelines (e.g., Hardman and Sharratt, 1990). This often leads to principles that are general but rather abstract and hard to apply, or guidelines that tend to be numerous and address only low-level aspects of the design. The aim of this section is to show how the resources concepts might help to occupy some of the middle ground by being more constructive without focusing on the detail of interaction.

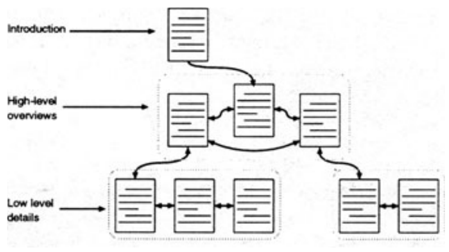
The need to support a number of cognitive styles is discussed by Hammond and Allinson, 1989), who suggest that a range of strategic options can be supported in an interface without undue increase in its complexity.

We shall see that for the three classes of objective, there are three corresponding interaction strategies, or cognitive styles, that are appropriate. This presents the interface designer with a number of choices for how to provide resources to support the strategies.

6.1 Supporting browsing

A user interacting with the hypertext may have an objective to acquire fairly general information about the process being described ("get an overview of the process", or to "look at the information on task analysis"). Since the objective in a specific situation can be articulated, a goal-oriented exploration strategy seems most appropriate. A plan-based strategy is likely to be less effective because the user with a browsing objective is unlikely to have a sufficiently good knowledge of the hypertext's structure, and the designer cannot be expected to anticipate all the browsing styles and possible user preferences a priori. The implications of this for design are twofold: the user should be able to see how their work-level objective can be re-expressed in terms of interaction-level goals, and that sufficient resources should be provided in the interface to support matching of actions to goals formed in this way.

These two issues together can be addressed by the design of the overall structure of the hypertext and the linking structure within it. The text can be organised so that there is a rich linking structure within a "domain" or logical group of information (such as the information on task analysis); the link labels can make it clear when selecting a link would leave one domain and enter another, thus providing support for browsing objectives like "see what's available on topic X".

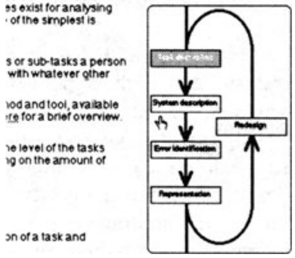


Feedback may be provided on where the user has been in the text, to help the user determine when a goal like "look at *all* the information on task analysis" has been accomplished.

6.2 Taking the user on a tour

Allowing the user to learn about the process in detail may seem very similar to browsing. There is, however, a difference: the process itself embodies a sequencing of sub-processes, and this may be reflected in a plan resource for the order in which the text is navigated. Learning can be supported by a tutorial-style presentation where the user is taken on a "guided tour" of the process. The design will accommodate a plan following strategy, and the plan will be located externally in the interface of the system, encouraging the user to see *all* the relevant material and to look at it in the *right order*. This still leaves a number of questions open, not least of which is the issue of how explicitly the plan should be implemented.

A fairly explicit way of representing a plan is to show a "map", either in a separate window, or integrated with the rest of the text. Boxes in the map represent sub-processes in the safety analysis method, and clicking on a box follows a link to an appropriate page. The current "marker" item is shown and items which have been seen previously can be indicated.



es exist for analysing
of the simplest is

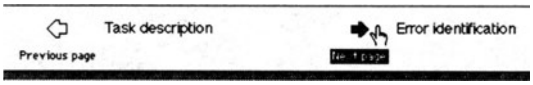
s or sub-tasks a person
with whatever other

nod and tool, available
re for a brief overview.

ne level of the tasks
ng on the amount of

on of a task and

More implicit implementations of the plan structure are also possible by, for example, providing hypertext links or navigation buttons to move around the structure of the text (and therefore the structure of the process being described).



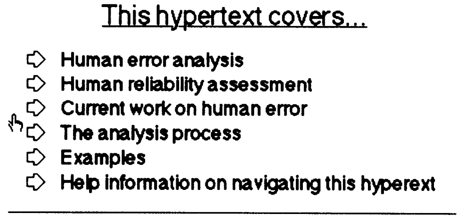
This implicit implementation shows the user enough of the plan for the next action to be taken, but gives no overall visibility of the structure of the plan itself or the progress that is being made through it.

6.3 Goal-directed information retrieval

In common with the objective of browsing the text, the retrieval of answers to specific queries can best be served by a goal-oriented strategy such as semantic matching. A difference between this case and browsing is that the goal can probably be more easily specified, and is related primarily to the information content of the text, rather than to the overall structure.

In interface terms, the appropriate strategy can be supported in a number of ways. A simplistic (in user interface terms, thought not necessarily in terms of the implementation!) design is to provide a searching mechanism. This allows a user to phrase quite arbitrary "commands" that are closely related to goals. One disadvantage of search interfaces is that they often provide no external representation of what "actions" are available and what their effects will be. In other words, the user is not aware of what keywords have been indexed for searching, so interpreting the results of a search may be hard.

A alternative is to provide navigational help, by externalising affordance and action-effect resources, either as navigation links at appropriate points in the document. Of course, this requires the designer to have anticipated possible goals, so that the relevant resources can be implemented in the navigational structure. A simple use of this approach is the provision of indices and contents pages that give the user some higher-level visibility of how goals might be achieved.



This hypertext covers...

- Human error analysis
- Human reliability assessment
- Current work on human error
- The analysis process
- Examples
- Help information on navigating this hyperext

7. CONCLUSIONS

The paper has described a model for human-computer interaction based on the notions of *information resources* and the *interaction strategies* in which the resources may be deployed. The central message of the paper is that a single cognitive perspective isn't an adequate basis for designing interactive technology; the concept of strategy allows the model to generalise over a number of paradigms. Furthermore, the process of designing interactive systems can be seen as a progression from an understanding of the objectives of the system, through interaction strategies appropriate for the objectives, to decisions about how and where resources required by the strategies are to be implemented.

The kinds of domains used as examples in this paper could be classified as consisting of "user driven" interactions: the interactive artefact is a fairly passive component, it's behaviour being largely determined by user's inputs. In more dynamic domains such as aircraft flight decks or process control systems, however, the picture is much more complex. Here, the external world impinges on users' activity, generating events at unpredictable times and imposing temporal deadlines to satisfy higher level requirements (such as safety). Extending the concepts of resource and strategy into domains with such complex constraints is one of the main challenges for the future development of this work. One way of meeting the challenge is to consider not only the resources that might be employed in support of a strategy, but also the cognitive demands associated with using a resource, and the effect that this has on "metacognitive" issues like strategy switching.

A longer term aim of the work is to inform models of failure in interactive systems. Many existing approaches to understanding failure concentrate of errors at the subtask or action level. We aim to consider more of the contextual factors surrounding failure by looking at the ways that available resources can fail to support a strategy, and how this is affected by the demands of the work.

REFERENCES

Bornat, R. and Sufrin, B. (1996). Jape's Quiet Interface. In Merriam, N. A., editor, *International Workshop on User Interfaces for Theorem Provers*. Department of Computer Science, University of York. URL: *http://www.cs.york.ac.uk/~nam/uitp/proceedings.html.*

Carroll, J. and Mack, R. (1986). Learning to use a word processor: By doing, by thinking, and by knowing. In Thomas, J. and Schneider, M., editors, *Human Factors in Computer Systems*, chapter 2, pages 13–51. Ablex Publishing Corporation.

Fields, B., Wright, P., and Harrison, M. (1996). Designing human-system interaction using the resource model. In *APCHI'96: First Asia Pacific Conference on Human Computer Interaction*, pages 181–191. URL: *http://www.cs.york.ac.uk/~bob/papers.html.*

Green, T., Cañas, J., and Warren, C., editors (1996). *Proceedings of ECCE8: European Conference on Cognitive Ergonomics*. EACE.

Halverson, C. (1994). Distributed cognition as a theoretical framework for HCI: Don't throw the baby out with the bathwater – the importance of the cursor in air traffic control. Report 9403, Department of Cognitive Science, University of California, San Diego.

Hammond, N. and Allinson, L. (1989). Extending hypertext for learning: An investigation of access and guidance tools. In Sutcliffe, A. and Macaulay, L., editors, *Proceedings of the Fifth Conference of the BCS HCI Special Interest Group*, pages 294–304. Cambridge University Press.

Hardman, L. and Sharratt, B. (1990). User-centred hypertext design: the application of HCI design principles and guidelines. In McAleese, R. and Green, C., editors, *Hypertext: State of the art*, chapter 28, pages 252–259. Intellect Ltd.

Hutchins, E. (1995). *Cognition in the Wild*. MIT Press.

John, B. and Kieras, D. (1994). The GOMS family of analysis techniques: Tools for design and evaluation. Technical Report CMU-CS-94-181, School of Computer Science, Carnegie Mellon University.

Kirwan, B. and Ainsworth, L. (1992). *A Guide to Task Analysis*. Taylor and Francis, London.

Kitajima, M. and Polson, P. (1995). A comprehension-based model of correct performance and errors in skilled, display-based human-computer interaction. *International Journal of Human-Computer Studies*, 43(1):65–100.

O'Malley, C. and Draper, S. (1992). Representation and interaction: Are mental models all in the mind? In Rogers, Y., Rutherford, A., and Bibby, P., editors, *Models in the Mind: Theory, Perspective & Application*, chapter 6, pages 73–91. Academic Press.

Plowman, L. (1996). What's the story? narrative and the comprehension of educational interactive media. In (Green et al., 1996).

Suchman, L. (1987). *Plans and Situated Actions – the problem of human-machine communication*. Cambridge University Press.

Wright, P., Fields, B., and Harrison, M. (1996). Distributed information resources: A new approach to interaction modelling. In (Green et al., 1996), pages 5–10. URL: *http://www.cs.york.ac.uk/~bob/papers.html.*

Zhang, J. and Norman, D. (1994). Representations in distributed cognitive tasks. *Cognitive Science*, 18:87–122.