ISSN 0105-8517

On Models and their Rôle in the Use of Computers

John Kammersgaard

DAIMI PB - 202
December 1985

This paper is also part of the preceedings of the Working Conference on Development and Use of Computer-based Systems and Tools, Århus, August 1985.

PB - 202

TRYK: RECAU (06) 12 83 55

PB - 202

J. Kammersgaard: On Models and their Rôle . . .

ISSN 0105-8517

On Models and their Rôle in the Use of Computers

John Kammersgaard

DAIMI PB - 202
December 1985

DATALOGISK AFDELING
Bygning 540 - Ny Munkegade - 8000 Aarhus C
tlf. (06) 12 83 55, telex 64767 aausci dk
Matematisk Institut Aarhus Universitet

1 Introduction

When studying systems development projects a lack of consideration for training and education often is observed. A typical reason is that management is unwilling to offer the required money; all they will offer is a short introduction to "how to push the buttons". But this explanation is insufficient. Even in situations where there is a demand for training and education a typical offer from the vendors is an introduction to "how to push the buttons".

Why is it so? The hypothesis brought forward in this paper is that there is a lack of understanding for the necessity for to deal with training and education already during the design process. Some vendors are aware of this situation. Their solution to the problem seems to be to develop computer applications which are *easy to learn.*

In my opinion this is a wrong solution. How can it be easy to learn a computer application which deals with very complex tasks? The "easy-to-learn syndrom" leads to a tendency towards development of computer applications which possibly are easy to learn, but which are not sufficently powerful to accomplish the tasks they have to deal with. Learnability instead of efficiency becomes the main goal. Instead I will propose this solution to the problem:

"Develop computer applications, which are easy to teach"

How do we make a computer application easy to teach? A promissing path to follow seems to be to base the development of a computer application on consistent conceptual models, which are based on concepts from the area of application. Such models must be developed during the systems development process to be used both for further design, during teaching and training, and during interaction with the computer application. A model for this purpose is here called a **user model**[1].

Literature on user models primarily reflects the designer's viewpoint[2]. It deals with how user models can be valuable during design. An understanding of user models also covering the users' point of view is needed — we must understand how user models can be valuable to the users. Design considerations do not necessarily raise the same demands to the contents and form of a user model as is raised by considerations concerning training and education, or by considerations concerning man-machine interaction. My purpose with this paper is to try to look at user models and their use mainly from the users' point of view, and from that perspective point at some paths, which leads towards a deeper understanding of the subject.

To present the background for my interest in these topics, I will give a brief description of some essential experiences from the UTOPIA-project[3], a project on the use of computers for page- and image processing in newspaper production, which was initiated by the Nordic Graphic Trade Union in 1980, and carried out jointly by graphic workers and computer and social scientists in Sweden and Denmark[4]. The aims of the project were to develop powerful tools for skilled graphic workers.

[1] In literature often called "users conceptual model". [Norman83]: "A conceptual model is invented to provide an appropriate representation of the target system".

[2] See for instance [Newman79], [Moran81] or [Foley82].

[3] In the Scandinavian languages, UTOPIA is an acronym for "Education, technology and product in a quality of work perspectice".

[4] The original ideas behind the project is presented in [UTOPIA80]. The results from the project is documented in twenty reports written in Swedish and Danish. An english summary of most of these reports can be found in [UTOPIA85].

During the project, in which I have worked for approximately two years, we have developed user models for computer-based page make-up and image processing[5]. We chose this strategy mainly because we realized that some kind of general models based on profession oriented terms (i.e. terms used within the area of application) were neccessary to explain to the graphic workers, which possibilities the new technology could offer, and how these possibilities related to possibilities with the existing technology. Studies of literature on the topic made us realize that the field of research was rather superficially explored. Very little theoretical work had been done. We could only in very vague terms get an idea about what was meant by the concept user model. We knew a lot about what we were looking for within our specific area, and we knew examples of models within other areas which seemed to have a lot in common with what we thought of as user models[6]. But there was no theory trying to define the concept in an application-independent way.

Furthermore there were no guidelines for the development of user models. We developed our own techniques and tools that worked very well for our purposes[7] — but can they be used in general?

Finally, we could only find very vague indications about how the models we developed could be made useful to the users. Some literature claims that user models are valuable to the user but does not deal with how to make them operative[8], and some literature views user models as nothing but a tool for the designer[9]. Obviously the models have been valuable to the graphic workers participating in the development process, but in which way can they be made valuable to others, for instance in relation to design and implementation; in relation to training and education; and in relation to the interaction with computer-based page make-up and image processing systems.

A lot of questions about user models obviously need further clarification:

1. What is a user model?
2. How can user models be used for training and education and as basis for interaction with computer applications?
3. What are the relations between user interfaces and user models?
4. How can user models be used/developed within systems development projects?
5. Which techniques and tools are needed to develop user models?
6. How do we describe and present user models to the users?

This paper mainly deals with the first four questions. The purpose of the paper is not to introduce a new concept, but rather to try to point at some paths towards an understanding of the concept of user models which includes a view on the subject as seen from the users' perspective. To do that I have chosen the following overall structure of the paper:

- First I give my motivation for the choice of topic (above).

---

[5]Readers who are unfamiliar with the user models developed within the UTOPIA-project are recommended to take a look at the appendix in [Ehn84].

[6]As for instance the desk-top model for computer-based office work. See [Star82] and [Lisa83].

[7]See [Ehn84] and [Bødker85].

[8]See for instance [Norman83].

[9]See for instance [Newman79] or [Foley82].

- Secondly I present some key concepts which are used in the presentation and discussion. The presentation of concepts includes a very general "definition" of the concept **user model**.

- Thirdly I present and discuss a view on user models known from literature. I have chosen the view presented by Newman and Sproull[10] to illustrate some of my critique.

- Fourthly I present ideas which lead towards a broader view. The role of user models when interacting with computers is proposed to be seen as analogous to the role of models within science in order to try to learn something about the nature of models. Furthermore the role of user models in relation to learning is discussed. To be able to do that I have brought in knowledge from psychology and pedagogy. Finally the role of user models in relation to design of user interfaces is briefly discussed.

2 Key concepts

Before going on to the presentation and discussion of a well known view on user models I will be a little bit more specific about what I am talking about.

All this discussion takes place within the field of **man-machine interaction**, or for short **interaction**. By using the word interaction we get associations to processes in which tasks are performed by common efforts from equal participants. Looking at a process involving human beings and computers this is a bad association — humans and machines are not equal — but in need of a better word the term will be used to denote characteristics of the processes taking place when humans are using computers.

According to Moran[11] a **user interface** consists of those parts of a system, which the user comes into contact with — physically, perceptually or conceptually. Although there is much truth in this definition it seems inappropriate to me. For instance it indicates that the user interface is dependent on the user; for a given application the user interface is defined differently by each user. Instead I would like to set focus on two characteristics of a computer application — **what and how**.

What can be done with a computer application? Which products can be produced? When trying to answer such questions we focus on the **functionality**.

How do we obtain a wanted effect? Which possibilities do we have to control the computer application? When trying to answer such questions we focus on the **user interface**.

A specific part of software might contribute to the definition of both functionality and user interface; so software cannot be split into two disjoint parts by means of these concepts. Instead the concepts can be seen as defining two different perspectives[12] on a computer application.

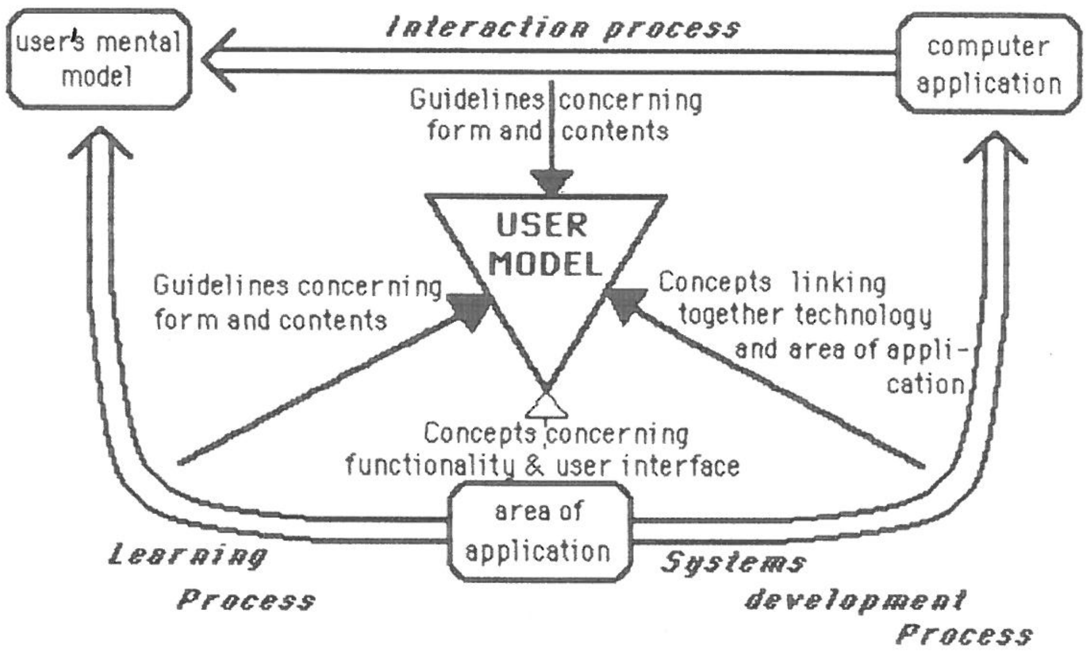
Each application is developed to fulfil some purpose within its **area of application**. Functionality relates to tasks performed within the area of application. **User models** relate to both functionality, user interface, and area of application. A user model is a user oriented conceptual model linking user interface and functionality together and relating them to the area of application. Or to put it another way: it is a user oriented conceptual model linking concepts from a specific area of application to their technical realization. A

---

[10]See [Newman79] chapter 28.

[11]See [Moran81] p. 4.

[12]For a definition of the concept "perspective" see [Sørgaard85].



user model explains what a system can be used for and how it is used. This is done partly by means of concepts anchored in the area of application, and partly by means of concepts linking together the technology and the area of application. A user model is more than a model of a computer application; it is a model of how a computer application can be used within a labour process.

Why do we need user models? As mentioned some psychologists[13] claims that humans interact with computers on the basis of mental models — the **user's mental model**. Each user understands a computer application in his own way (i.e. has his own mental model); a user model is developed to support the creation of suitable mental models.

Let us now try to look at how the various concepts defined relate to user models. The figure above is an attempt to summarize from which sources the development of user models might be influenced.

The area of application is the primary source on the conceptual level. Most concepts concerning functionality as well as user interface should be anchored within the area of application, although it is necessary to bring in other concepts to be able to link together the area of application and the technology in use. These latter concepts, which should be concerned mainly with the user interface, are developed during the systems development process in order to make it possible to take advantage of new possibilities offered by computer technology.

As it will be discussed more intensively later, other concerns have to be taken when user models are developed. User models are intended to be used in learning processes and during interaction processes to support the development of operational mental models. An understanding of the role of user models in relation to these processes is therefore essential when user models are developed.

---

[13]See for instance [Gentner83].

3 A traditional view on user models

I will now present and discuss the view on user models presented by W. M. Newman and R. F. Sproull in their book "Principles of Interactive Computer Graphics". The view is in many aspects identical to views presented by for instance Moran[14] and Foley[15].

In being among the first persons at all to write about user models Newman and Sproull have given a very important contribution to the discussion. My critique is not an attempt to be wise after the event by for instance stating that their view is misleading, but to point at an area where a complement is needed.

In their book Newman and Sproull discuss user models in a chapter on user interface design. They define the user interface as consisting of four components: *user's model, command language, feedback and information display.* On the user's model and the relation to the other three components they say:

> "One of these underlies the other three: this is the user's model, the conceptual model formed by the user of the information he manipulates and of the processes he applies to this information. Without this model the user can do little more than blindly follow instructions, like an inexperienced cook following a recipe. The model enables him to develop, even with little or no knowledge of computer technology, a broad understanding of what the program is doing. With the model's help he can anticipate the effect of his actions and can devise his own strategies for operating the program. Sometimes the design of the user's model is simply a matter of simulating as closely as possible a real world system, so that the user need not develop any model of his own. This is what we would do, for example, in designing an aircraft pilot training system. This approach to the user's model does not always work, however, because simulation of the real world often proves inappropriate or difficult."

What they are talking about here is what I call the **user's mental model**. Newman and Sproull see this model as part of the user interface. What has this got to do with **user models**? Well, Newman and Sproull are discussing design of user interfaces as design of the four components, so they also talk about "the design of the user's model":

> "The use of a familiar set of concepts provides us with a good starting point for the user's model design, and we can then proceed to extend and refine the model, keeping it as simple and consistent as possible in order to help the user assimilate it."

So without explicitly mentioning the term, Newman and Sproull are dealing with user models — namely as "the design of the user's model".

Newman and Sproull give some guidelines on how to describe a user model[16] as a set of **objects** together with a set of **actions**:

> "Each object is an item of information over which the user has some control: he may be able only to display it and to ask questions about it, or he may be able to modify it, destroy it, and create other objects in its place.
>
> "Actions are the operations the user can apply to objects. The complete set of actions thus defines the functional capability of the program."

[14]See [Moran81].

[15]See [Foley82].

[16]They use the terms:"to represent a user's model".

Furthermore they make a distinction between two kinds of objects:

> "The user's model generally contains two kind of objects: those which are intrinsic to the application and those whose purpose is to assist in the control of the program. The former kind of objects we call intrinsic objects, the latter control objects."

Different points of critique can be raised against this view on user models. One point is primarily philosophical. The phrase:"to design the user's model" indicates that it is possible to design something and just put it into the mind of another person. I don't think that this interpretation is what was intended by Newman and Sproull, but they have not been aware of the danger although they also use phrases like:"...to extend and refine the model keeping it as simple and consistent as possible in order to help the user assimilate it."

In my point of view it is neccessary to talk about training and education in this context. A user model is an educational tool intended to support the creation of suitable mental models — not just the design of a mental model. The point of critique raised here is further justified by the fact that Newman and Sproull make a very vague distinction between user models and the user's mental model; they talk about "user's mental model" and "the design of the user's mental model".

My second point of critique has to do with the overall perspective. From the quotations above it is obvious that Newman and Sproull take the designer's perspective. They deal with how to design user models and how to connect that to the design of user interfaces. They deal with how designers and implementors can make use of user models. Obviously this perspective is necessary, but it is not sufficient. We also need to view the subject from the users' perspective. We must try to view the questions raised during my motivation in a way, which reflects the users' point of view.

A very essential experience from the UTOPIA-project is that active user involvement has been important during design of user models. A lot of activities have been carried out to make this kind of user involvement possible (for instance mutual exchange of experience between designers and users), and new techniques and tools have been developed to support the design process[17]. One serious consequence of the designer perspective described by Newman and Sproull is that design of user models might be reduced to a task carried out solely by designers.

My third point of critique is strongly connected to the previous point. The object/action representation of a user model might be the way the designer would like to view the subject — but it is not necessarily the way the user sees it. The user structures his view on user models according to the labour process in which the application is intended to take part; i.e. according to the area of application. Furthermore concepts which link together the technology in use and the area of application play a crucial role.

Let us as an example take the UTOPIA-model for electronic page make-up[18]. Seen from a user's point of view this user model is defined by means of make-up material and make-up tools; some concepts (like "table" and "lenses"), which explain how material and tools relate to a graphical workstation; and an explanation about how make-up tools are realized through a combination of general interaction devices and graphic operations. The model relies heavily on an understanding of what a high quality product is, and how the labour process traditionally has been carried out; it is for instance impossible to explain which (lense)scales are needed without knowing a lot about the labour process and about high quality products. Although objects and actions seem to be good concepts to use

[17] For instance the mock-up simulation technique. See [Ehn84] and [Bødker85].

[18] See the appendix in [Ehn84].

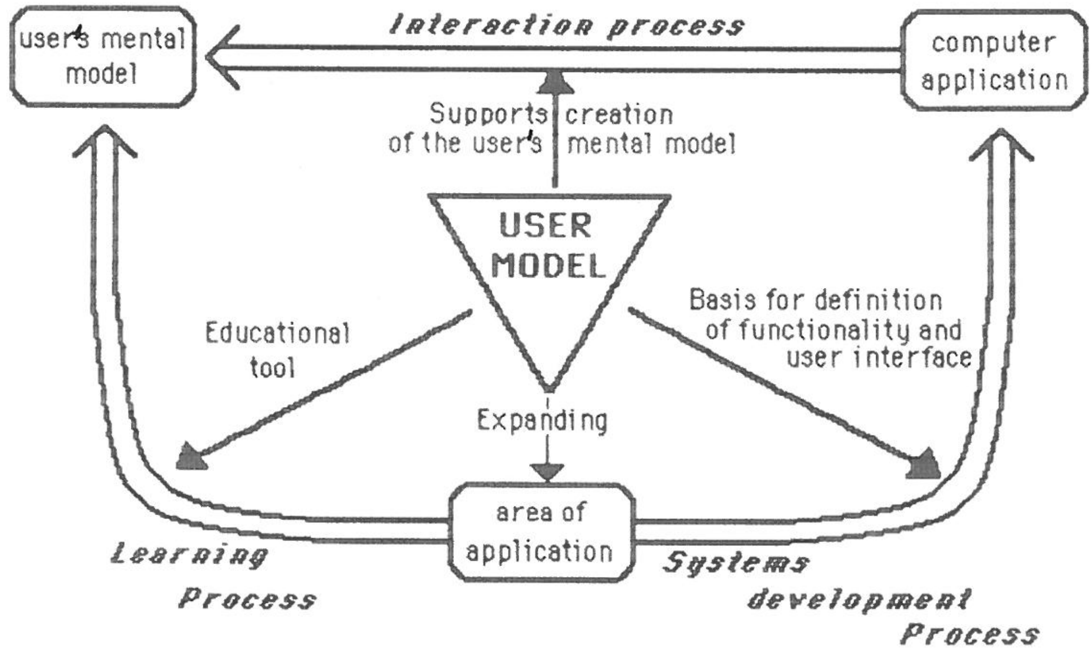
(better than for instance flow analysis concepts) to help computer professionals span the gap between user models and implementation languages, I claim that user models like the UTOPIA-model for electronic page make-up cannot be explained or understood solely by means of objects and actions without loosing essential features.

My final point of critique has to do with the relation between user models and computer applications. From the description of Newman and Sproull it appears that there is only one user model for each computer application. Seeing user models as models for the use of computer applications in labour processes I cannot agree with that. First of all a user might benefit from using models at different levels of detail. Secondly different groups of users might need different user models. They might need to view the computer application from different points of view, i.e. focusing on different aspects of the application. So there might be a point in developing user models which are focusing on different aspects. This standpoint will be further elaborated in section 4.

To sum up, my conclusion is that the view presented by Newman and Sproull gives a lot of information about the topic, especially if our primary interest is design. I have pointed out some weak points and stated that a user perspective on user models is needed as a complement.

4 Towards a broader view on user models

As pointed out earlier I find the traditional view on user models too restricted. In this section I will try to broaden the perspective by considering the relations to learning and interaction to be as relevant as the relations to systems development. The figure below illustrates the use of user models in relation to some of the key concepts defined in section 2. In the following three subsections I will consider the relations to the three processes separately, but before going on to that I will make a short comment on the relation to the area of application.



As mentioned earlier the area of application plays an essential role during the development of a user model. Knowledge about and experience with the area of application is a precondition when we want to develop a model which makes it possible to relate the use of a new computer application to the existing work processes. But this is not the only relation between a user model and the area of application. The introduction of a computer application based on a user model often leads to an expansion of the area of application according to traditions within the profession. New concepts are introduced to make the computer application "fit into" the area of application (i.e. the language of the profession is extended) and new skills are developed based on knowledge about the user model.

4.1 User models and man-machine interaction

Users create mental models of the computer applications they deal with. The creation of mental models might be supported by user models. This situation has an analogy within science, where theories often are developed, explained, and tested in close connection to models. Explanation of theories is in many aspects analogous to explanation of computer applications and testing of theories is in many aspects analogous to interaction with computer applications. In both situations models play a crucial role — so it seems to be a good idea to try to learn something about models from philosophy of science[19].

Within philosophy of science the term model is used in three different ways:

- a **black-box model** is a model, which allows us to predict the reactions of a system when it is affected in certain ways,
- a **grey-box model** is a model, which allows us to explain general characteristics of a system in terms of observational concepts.
- a **theoretical model** is a model, which allows us to give a formal explanation of general characteristics of a system in theoretical terms.

Relating this to my former very general definition of the term user model it turns out that a user model might be either a grey-box model or a theoretical model. Both these types of models allow us to explain what a system can be used for and how it is used. So my use of the term model seems to be consistent with the way it is used within philosophy of science.

Now let us look in deeper detail at the role of models within science. Hesse lets in the first chapter of her book "Models and Analogies in Science" two persons discuss the role of models within science. One of the persons, taking the standpoint of Duhem[20], claims that scientific theories can be formulated in pure theoretical terms (i.e. without reference to a model) whereas the other person, taking the standpoint of Campbell[21], claims that models are necessary preconditions for the formulation of theories. The Campbellian defines a model in this way[22]:

> "A model, for me, is any system, whether buildable, picturable, imaginable, or none of these, which has the characteristics of making a theory predictive."

[19]See for instance [Hesse70], [Freudenthal61], [Ortony79], [Black66] and [Bunge73].

[20]French philosopher.

[21]British philosopher.

[22]See [Hesse70] p. 19.

Typewriter	Word processor
alfanumeric buttons	alfanumeric buttons
paper	screen
wheel	font
positioning	positioning

The essential property of a model is predictiveness; by reference to a model it must be possible to predict the behaviour of a phenomenon under consideration. Referring to the behaviour of a well-known phenomenon is a nice short-cut when we have to make such predictions. This is why analogical models are used so often.

Within the field of man-machine interaction some arguments has been raised against the use of analogies[23]. By examining the concept of analogy within a philosophical framework I will try to analyse these arguments.

Analogies are drawn between two phenomena. We try to predict or explain the behaviour of **the primary phenomenon** (i.e. the phenomenon under consideration) on the basis of what we know about **the secondary phenomenon.**

Two sorts of relations between primary and secondary phenomenon are established when drawing an analogy. The most obvious is **the similarity relation**, which is concerned with identity and difference between properties of the phenomena. If we, for instance, draw an analogy between a word processor (the primary phenomenon) and a typewriter (the secondary phenomenon), we might consider the properties shown in the table above.

The similarity relation says that the alfanumeric buttons are for the word processor as they are for the typewriter; the screen is for the wordprocessor as the paper is for the typewriter, etc.

Besides the similarity relation there is another important relation — **the causal relation.** The causal relation between the properties of the typewriter defines the interconnection between these properties. For instance: When an alfanumeric button is touched the corresponding character on the wheel is printed on the paper in the current possition. To draw an analogy requires that a causal relation similar to one between the properties of the secondary phenomenon exists between corresponding properties of the primary phenomenon. So for our example to be an analogy the following must be true: When an alfanumeric button is touched the corresponding character of the font is printed on the screen in the current position.

The discussion above only deals with **the positive analogy** (i.e. relations between properties we consider to belong to both primary and secondary phenomenon), but analogies are more than that. **The negative analogy** consists of properties of the secondary phenomenon, which we know do not belong to the primary phenomenon. The ribbon is for instance a member of the negative analogy in the typewriter-example. Furthermore, because of our limited knowledge about the phenomena, there is **a neutral analogy** consisting of those properties of the secondary phenomenon, which we do not know whether belong to the primary phenomenon or not.

Now let us look at the opponents against the use of analogies in relation to computers. What are their arguments? They claim that:

[23]See [Halasz82].

1. We prevent ourselves from developing substantially new products.
2. We extend the risk that users will make wrong predictions about the behaviour of a computer application.

Both arguments are, as far as I can see, based on a lack of understanding of the nature of analogies and their relation to models. Let us look at the first argument. Of course it is correct if we assume that analogies always can be used as models. Then we must be able to predict the behaviour of a computer application from our knowledge about an existing phenomenon, which means that no properties existing solely because of the characteristics of the computer can be included. But this assumption is wrong. Situations exist where the neutral analogy is very limited, and where the negative analogy only contains inessential properties. In such cases analogies might fulfil the requirement of being predictive. But in many situations we need more than an analogy to fulfil this requirement — we need a model. So analogies are not models, but models might be based on analogies[24].

A similar discussion about the relation between analogies and models has taken place within philosophy of science. Some philosophers have identified models with analogies. Philosophy of science as it appears in the dialectical-materialistic tradition gives some general arguments for the necessity to distinguish between analogies and models, and for the value of analogies as basis for the development of models[25]:

> "Reducing models to analogies means that we cannot reach an independent comprehension of the characteristics and connections in reality."

> "Analogies are only one way out of many in which we construct models."

> "The use of analogies is a powerful instrument when we want to expand the scientific comprehension to new areas."

Juul Jensen, a Danish philosopher, who belongs to the dialectical-materialistic school of thought, also discusses why so many people make the mistake to consider analogies as models[26]. In his view the reason is that they have mixed up the question about how models are created and under which conditions they can be used, with the question about the ontological contents of the model (i.e. which part of reality the model can be used to reason about). As an example he mentions that theater often is used as a metaphor for social behaviour and that some sociological theories (models) are based on this analogy. To say that the analogy is a model is to say that reasoning about social behaviour can be carried out on the basis of knowledge about theater, and that inexplicable social behavior can be made explicable through analysis of theater (the model). Of course this cannot be true in general. How, for instance, to explain that humans sometimes behave as if they were playing theater?

Then back to the second argument. Does the use of analogies extend the risk that the users will make wrong predictions about the behavior of a computer application? If analogies are used without taking into account that a model based on the analogy must be present this argument might be true in many situations. But then it is no more an argument

[24] As an example see the model for computer-based make-up in [UTOPIA84].

[25] See [Juul Jensen80] p. 120, authors translation.

[26] See [Juul Jensen80] p. 121.

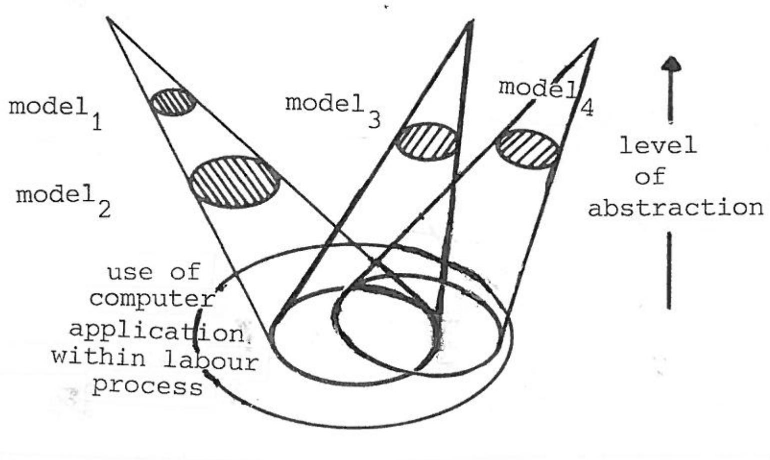
against the use of analogies in general, but rather an argument against the use of analogies which have no connections to a model. But why do people then use the argument against the use of analogies in general? In my opinion the reason is the existence of computer applications which are claimed to be based on pure analogical models although this is not the case. This situation might be arrived at either because of a lack of awareness of the difference between models and analogies, or because the designer has failed when drawing the analogy. The latter can easily happen if there is not enough awareness of the causal relation. Even obvious similarity relations do not necessarily mean that a causal relation similar to that of the secondary phenomenon holds for the primary phenomenon.

The above analysis leads to a rejection of the arguments against the use of analogies. Analogies are useful during development of user models, but only in rare situations can a model be equal to an analogy.

History of science contains a lot of examples of analogies used in the construction of models. The planetary system has, for instance, been used in many analogies during construction of atomic models. Also very incomplete analogies (i.e. analogies with a substantial negative analogy) can be extremely useful. Models of sound propagation (explaining among other things reflection and diffraction) are, for instance, developed on the basis of analogies with water-waves, although sound propagates with longitudinal amplitudes whereas water-waves propagates with horizontal amplitudes.

In some situations it might be valuable to have alternative models depending on what is to be explained. Models of the same phenomenon might even be based on at least partially contradictory analogies. Quantuum mechanics is the key-example of that within philosophy of science. Depending on what we want to explain it is operational to consider propagation of light as analogous to either wave propagation or propagation of particles; two views, which are obviously contradictory.

Also everyday-situations show the value of multiple models of the same phenomenon. Let us as an example look at a local bus system. Which model does a citizen need to get around town by bus? Of course it is possible to manage the situation without models; but not without loosing both time and energy. The only way is to walk around town until suddenly a bus shows up, board it, and hope it will arrive at the intended destination. If this seems to fail, nothing else can be done than to have another try. By using and combining two models this situation can become much more comfortable. A town plan showing bus routes and bus stops makes it possible to predict where to stand to catch a bus to a specific destination. So this model creates a connection between busses and places, but it is impossible from the model to predict when and how often a bus will arrive at a specific bus stop. To make that possible the model must be combined with a timetable; a model connecting to each bus stop hour and minute of bus arrivals/departures. Most bus passengers will be able to use the bus system in a satisfactory way using the combination of these two models (eventually further combined with models regarding fares, etc.), but is the same model operational for the busdriver? To him one bus is particularly important; namely the one he is driving. He can make use of the town-plan/timetable model both to find his way along the route, and to provide service to customers not knowing how to catch a bus to a specific destination. But normally more than one bus are driving along a route. If it for instance takes 30 minutes to drive from one terminus to the other and a bus has to arrive at each bus stop each 10 minutes then at least 6 busses (3 in each direction) has to be driving along the route at the same time. The busdriver needs to know which of these tours his specific bus has to make. To be able to know that, an alternative timetable (irrelevant to ordinary passengers), showing when his bus must leave the terminus and when it has to be at each bus stop, is a usable model. Notice that an essential difference between the models



required by the two categories of "users" is that busdrivers view the system as it looks from one specific bus driving one specific set of tours on one specific route, whereas passengers do not care about specific tours or specific busdrivers, but only about routes, bus stops, etc.

I could go on expanding this example by discussing models for controllers, routeplaners, etc., but I think enough is said to show that multiple models is a common requirement. It is my assertion that this also is the case when we turn to the use of computer applications. Different user models might be valuable for the same person for different purposes. Furthermore different user models might be valuable to different groups of users according to, for instance, their experience with the area of application and their experience with the computer application, or according to the task they have to perform.

The figure[27] above illustrates the mentioned relations between different models. The figure illustrates that different models can select different parts of the domain (model3 and model4 have different selections); that it is possible to interprete the same selection in different ways — i.e. see the selection from different points of view (model1 and model3 interprete the same selection in different ways); and that the same interpretation can be made on different levels of abstraction (model1 and model2 give the same interpretation of a selection at different levels of abstraction).

4.2 User models and learning

I have stated earlier that user models are relevant in relation to learning. To be able to elaborate on this statement I will present and discuss psychological and pedagogical theories of learning.

How do we learn? How is new knowledge established? Such questions have been dealt with for decades within psychology. Unfortunately no agreement about these questions has been reached. So to find a psychological foundation, which seems suitable as basis for pedagogical-didactical reflections about user models is not straightforward.

A theory which seems attractive has been developed by Piaget[28]. This theory, which is basically a theory of cognitive development, has, primarily because of its uniformity (i.e. its uniform treatment of development of skills, theoretical knowledge, feelings, attitudes, etc.),

[27] Strongly inspired by a figure by Oberquelle. See [Oberquelle84] p. 28.

[28] A survey of Piaget's theories can be found in [Flavell63] and in [Bjerg74].

been used in attempts to form a psychology of learning with a much broader perspective than what can be found in the previously very dominant stimuli-response tradition.

Piaget views human cognition as a specific form of biological **adaption** of a complex organism to a complex environment. Cognition always exhibits two simultaneous and complementary aspects which Piaget refers to as **assimilation** and **accomodation**. Although it is often convenient to talk about them as if they were distinct and separate cognitive activities, it must be kept in mind that Piaget conceives of them as two indissociable aspects of the same adaptional process.

Assimilation is the process in which we try to adopt external stimuli to our existing cognitive structures. We assimilate when we try to understand observations from what we already know.

Accomodation is the process in which we try to adopt our cognitive structures to external stimuli. We accomodate when we change our set of cognitive structures to be able to understand a previous unknown situation.

Piaget uses the term **schema** to denote a cognitive structure. The adaption process transforms our experiences to schemes, which might again be part of other schemes. When new knowledge is established the schemes are changed.

Assimilation can be seen as a process where relations between schemes are preserved, whereas accomodation leads to a change of these relations. Assimilation takes place within existing cognitive structures whereas accomodation although restrained by existing cognitive structures intends to change these structures. So there is a dialectical relation between assimilation and accomodation; accomodation is a superior process and assimilation is an inferior process with respect to the development of cognitive structures.

Inspired by Piaget's theory a group of Danish researchers on pedagogy have tried to formulate a theory of learning-processes[29]. This theory deals with learning processes which are mainly assimilative, and with learning processes which are mainly accomodative. Furthermore the theory deals with a third type of learning called cumulative learning. So according to this theory there are three different types of learning:

- **Cumulative learning**: Takes place in learning processes where new knowledge is transformed to schemes tied very close to the learning situation. Knowledge established through cumulative learning processes can only be made operational in situations similar to the learning situation. Schemes developed through cumulative learning processes are isolated items in the total cognitive structure, so the total competence established by cumulation is characterized by the sum of partial competences (i.e. no interference).

- **Assimilative learning**: Takes place in learning processes where new knowledge is transformed to fit into existing schemes. Knowledge established through assimilative learning processes can be made operational in situations which have some properties in common with the learning situation, and it might be valuable for further learning when trying to cope with new situations. Schemes developed are results of extension and differentiation of existing schemes. The competence established by assimilation spans wider than the learning situation. General applicable knowledge is established together with a characterization of situations in which the knowledge might be used. The established competence is tied to specific purposes.

- **Accomodative learning**: Takes place in learning processes where new knowledge is established by splitting existing schemes, which then are free to fit into new structures.

[29]See for instance [Nissen70].

Knowledge established makes it possible to use existing knowledge disconnected from the environment it was part of during an assimilative learning process. Schemes developed are results of reorganization and dissociation of existing schemes. The competence established by accomodation is not tied to specific purposes, but are closely related to abilities as flexibility, creativity, sensitivity, etc.

Now let us try to use this theory to discuss the types of learning relevant in relation to the use of computers.

During my introduction I stated that training and education often are reduced to an introduction to "how to push the buttons". The users learn how to use the computer application for a few purposes specific to their own labour process, and the learning situation is as similar as possible to the situation in which the computer application is to be used. This kind of learning is a typical example of cumulative learning. What is learned cannot be related to what is previously known, so it has no value in situations different from the learning situation. If something unpredicted happens the user has learned nothing, which can be used to cope with the situation.

Now what if someone wants to improve this situation? What can be done to provide users with more general applicable knowledge about a computer application? According to the described theory an assimilative learning process must be planned and carried out. During such a process general applicable knowledge tied to a specific purpose (i.e. the purpose of the computer application within the area of application) can be established. So, remembering the requirement raised in the introductory section the question is: How do we make it easy to carry out an assimilative learning process in relation to a specific computer application. Basing the development of a computer application on user models is one way, which makes it easier to carry out assimilative learning processes in relation to that computer application. User models are based on concepts from the area of application, and they seek to place the computer application in the area of application by using concepts linking these phenomena together. So, user models support assimilative learning by creating a connection to existing schemes. Using Piaget's terms we can say that the use of application-oriented terms supports assimilation (i.e. makes it possible to assimilate knowledge about the computer application in existing schemes), and introduction of concepts, which link together technology and area of application, supports accomodation (i.e. makes it easier to accomodate existing schemes to the new phenomena).

Assimilative learning processes might of course be possible to carry out even if the computer application is developed without connection to a user model; at least if the application is consistently designed. But there will not necessarily be any help to get from the computer application — the teacher has to do the job required to create a connection to existing schemes.

If no consistent model in terms of the area of application can be developed assimilative learning will be more or less impossible. Instead a model using computing terms can be developed, but it can only be used for assimilative learning if the users, at least to some degree, are computer professionals. So making assimilative learning possible in such a situation will often require a lot of learning to make the users familiar with computers in an application-independent way. And this is certainly not a situation where a computer application is easy to teach!

4.3 User models and user interfaces

According to the russian psychologist A. N. Leontjev two interfaces are involved when humans perform machine-mediated actions on an object (i.e. interacts with a machine to perform a specific task): an **operational-oriented interface** and an **action-oriented interface**[30]. The operation-oriented interface covers the relation between human and machine (computer) and the action-oriented interface covers the relation between machine and object. A similar distinction is made by Polanyi[31], who distinguishes between two types of awareness when we use machines (tools): **the focal point of awareness** is on the interface between the machine and the object and **the subsidiary point of awareness** is on the interface between ourselves and the machine.

Both the above mentioned authors point out that what we are interested in when using machines is to be able to focus on the focal point of awareness (the action-oriented interface). This indicates that a good user interface is an interface which "disappears" when we use it (i.e. which we don't have to deal with intellectually during our use of the computer). How can this be obtained?

A popular answer seems to be that we must develop user interfaces, which have a strong correspondence to a user model. But how do we create such correspondence? Believing that user interfaces can be put on top of a finished system design is wrong. Development of user models and user interface must go on simultaneously and take place at an early stage in the development process. But even that is not enough. Experiences from the UTOPIA-project indicate that development of user models and user interfaces, which correspond to each other in a reasonable way, besides being an integrated part of the systems development process, requires heavy user participation. You simple cannot tell whether or not the user interface will "disappear" without bringing in skills and experiences of professionals. The UTOPIA-project has shown some possible techniques and tools for this purpose[32], but more general guidelines and/or guidelines for other areas of application are needed.

5 Outlook

Our understanding of the role of models in relation to the use of computers is still very limited. In this paper I have suggested some new paths, which seems interesting to follow to get a better theoretical understanding:

- To learn from the role of the model within science.
- To learn from psychology and pedagogy to get a better understanding of requirements raised from educational considerations.

Ideas are only sketched in this paper; further research on the topic will hopefully show that the proposed paths were no blind alleys.

Besides this theoretically oriented research a lot of practical oriented research is needed. We must try to develop user models and we must put great effort into understanding why they are successful or why they are failures. As mentioned before we need guidelines to do this. We need techniques and tools to help us develop and present user models; we need techniques to help us utilize user models in the design of user interfaces; and we need to understand better how to utilize user models during training and education.

An interesting question is whether it is at all possible to develop computer applications, which can be used without knowing at least some of the basics of computer science. Application programs are developed on top of many layers of programs. If we want to

[30]The relevant parts of Leontjev's theories are described in [Kvorning84].

[31]See [Polanyi58].

[32]See [Ehn84] and [Bødker85].

prevent these layers from being able to destroy the correspondence between our user models and what is presented to the user (i.e. preventing lower layers from breaking the illusion that they don't exist), some sort of error-interpreter which is able to interpret all lower level events (for instance errors caught by the run-time system or by the operating system) in terms of the model, is needed. How such interpreters are to be designed, is still an open question, and an interresting issue for further research.

Bibliography

[Bjerg74] Bjerg, J., "Tænkning og udviklingsforløb" 3. udg., Munksgaard, Copenhagen 1974 (in Danish).

[Black66] Black, M., "Models and Metaphors", New York 1966.

[Bunge63] Bunge, M., "Model, Matter and Method", Dordrecht 1973.

[Bødker85] Bødker, S., "UTOPIA and the Design of User Interfaces", Comp. Science Dept., University of Aarhus 1985.

[Carroll82] Carroll, J. M. and Thomas, J. C., "Metaphor and the Cognitive Representation of Computer Systems", IEEE Transactions on Systems, Man and Cybernetics, vol. SMC-12, nr. 2, 1982.

[Ehn84] Ehn, P. and Kyng, M., "A Tool Perspective on Design of Interactive Computer Support for Skilled Workers", Proceedings of the 7th Scandinavian Seminar on Systemering, Helsinki 1984.

[Flavell63] Flavell, J,, "The Developmental Psychology of Jean Piaget", Van Nostrand Publ. Comp., New York 1963.

[Foley82] Foley, J. D. and van Dam, A., "Fundamentals of Interactive Computer Graphics", Addison-Wesley, 1982.

[Freudenthal61] Freudenthal, H., "The Concept and the Role of the Model in Mathematics and Natural and Social Sciences", Reidel Publ. Comp., Dordrect 1961.

[Gentner83] Gentner, D. and Stevens, A. L., "Mental Models", Erlbaum Publishers, Hillsdale 1983.

[Halasz82] Halasz, F. and Moran, T. P., "Analogy Considered Harmful", in Moran, T. P.: Eight Short Papers in User Psychology, Xerox PARC 1982.

[Hesse70] Hesse, M. B., "Models and Analogies in Science", University of Notre Dame Press, Indiana 1970.

[Juul Jensen80] Juul Jensen, U., "Videnskabsteori", Berlinske Forlag, Copenhagen 1980 (in Danish).

[Kvorning84] Kvorning, J., "Forsøg på generalisation af interface- og kompatibilitets-begreberne ved analyse af menneske-maskin interaktionen" (preliminary version), Institute of Psychology, University of Aarhus 1984 (in Danish).

[Lisa83] Williams, G., "The Lisa Computer System", BYTE vol. 8, nr. 2, 1983, p. 33-50.

[Moran81] Moran, T. P., "The Command Language Grammar: A Representation for the User Interface of Interactive Computer Systems", Int. J. Man-

Machine Studies vol. 15, 1981, p. 3-50.

[Newman79] Newman, W. M. and Sproull, R. F., "Principles of Interactive Computer Graphics", 2. ed., McGraw-Hill 1979.

[Nissen70] Nissen, T., "Indlæring og pædagogik", Munksgaard, Copenhagen 1970 (in Danish).

[Norman83] Norman, D. A., "Some Observations on Mental Models", in [Gentner83].

[Oberquelle84] Oberquelle, H., "On Models and Modelling in Human-Computer Cooperation", in van der Veer, G. C. et al. (ed.): "Readings on Cognitive Ergonomics - Mind and Computer", Springer Lecture Notes No. 178, 1984.

[Ortony79] Ortony, A., "Methaphor and Thought", Cambridge 1979.

[Polanyi58] Polanyi, M., "Personal Knowledge", Routledge and Kegan Paul, London 1958.

[Star82] Smith, D. C. et al., "Designing the Star User Interface", BYTE, April 1982, p. 210-223.

[Sørgaard85] Sørgaard, P. and Nygaard, K., "Perspective — A Key Concept in Informatics", Comp. Science Dept., University of Aarhus, 1985.

[Utopia80] Ehn, P., Kyng, M. and Sundblad, Y., "The UTOPIA Project — On Training, Technology and Products Viewed from the Quality of Work Perspective", Comp. Science Dept., University of Aarhus, 1980.

[Utopia84] Ehn, P., Eriksson, B., Eriksson, M., Frenckner, K., and Sundblad, Y. "Utformning av datorstödd ombrytning för dagstidninger" Swedish Center for Working Life, Stockholm 1984 (in Swedish).

[Utopia85] The UTOPIA project group, "Graffiti 7 — Alternatives in Text and Picture", Swedish Center for Working Life, Stockholm 1985.