

Improving the User-Interface to Increase Patient Throughput

A. Gupta, J. Masthoff, Philips Research Laboratories, Redhill, UK

P. Zwart, Philips Medical Systems Ned. B.V., Best, The Netherlands

Keywords: Medical, Error, Guidance, Customisation, Learning, Task Analysis, User Interface, Help

Abstract

Philips Medical Systems is a leading supplier of equipment to the international health care industry. We have been exploring the potential for applying advances in user interface technologies (particularly customisation) in future medical X-ray systems. In this paper, we outline the rationale for this by looking at the trends in the health care industry, and the types and sources of error, which we have observed first hand in various hospital departments. We distinguish and discuss three ways to reduce errors: improved design of the user-interface, customised default parameter values, and active support for learning during system usage i.e. during examinations. We have based our work firmly on the principles of user-centred design: observations, task modelling, user involvement and prototyping have been undertaken.

Introduction

The research presented in this paper has been conducted in a collaboration between Philips Research Laboratories and Philips Medical Systems, with input from Philips Design. It is a three-year project that has been going on for a year.

As part of a team skilled in user-interface design, the authors of this paper from Philips Research were contracted by Philips Medical Systems to suggest improvements in the examination and control rooms. Our prior experience with the medical domain was rather limited. We needed to build up our knowledge of the clinical X-ray environment, so that we could identify real opportunities for improvement. Our customer did not see this as a burden – we were needed for our knowledge in the user-interface field and particularly in the methods underlying user-centred design, the customer provided the knowledge about the medical field. We therefore initiated a series of hospital

observational studies, interviews, and training courses. We also needed to then conceptualise solutions that were technologically feasible. We organised a multi-disciplinary Design Workshop at which we held many brainstorming sessions.

Structure of this paper

The structure of this paper is as follows. We begin by discussing what we mean by human error in the clinical domain. We then describe the observational studies undertaken. The process followed, methods used, and our findings, are described. The user-interface technologies that were judged to be applicable are then described. Summaries of all of the above were presented to participants at a Design Workshop as the raw material on which to base concept generation. The process we used is described. Additionally, the notation used to document the task model is also described. We then discuss a number of ways to reduce human error, namely improved design of the user-interface, default parameter values customised to the situation at hand, and active support for on-line learning. Customisation is the essential aspect and is described. We end with some conclusions.

Definition of human error in the clinical domain

Research on human error in safety-critical systems is generally concerned with the prevention and analysis of major accidents and near misses, concerning aircraft, trains, and oil and chemical plants. We are, however, not focusing on the extreme examples, but on more subtle, latent types of errors, where each occurrence in itself does not have catastrophic effects, but the cumulative impact on relatively less tangible aspects is considerable.

Leplat (as quoted in reference 1) states that "a human error is produced when a human behaviour or its effect on a system exceeds a

limit of acceptability". Another definition of human error is "all the occasions in which a planned sequence of mental or physical activities fail to achieve its intended outcome, and when these failures cannot be attributed to the intervention of some chance agency" (ref. 2). Both of these definitions associate human error with not reaching a set goal ("exceed a limit of acceptability" and "fail to achieve its intended outcome"). So, in order to define human error in the clinical domain, we have to first assess what the goals in the clinical domain are.

Some important goals in the healthcare industry are to:

- maximise patient *safety and comfort*, for instance, by minimising dose and contrast fluid used, and by accurate diagnosis,
- maximise *patient throughput*, by reducing the time needed per examination,
- minimise the *resources* used per patient, such as film, contrast fluid, and staff time.

We take a broad view of error and define it as *human behaviour resulting in sub-optimal achievement of the goals listed above*. These errors are extremely likely to occur, but individually do mostly not have severe and obvious consequences. However, they pose a high risk even if the severity of their individual consequences tends to be low, as risk is defined as the probability or frequency of occurrence multiplied by the severity of the consequence.

According to our definition, the following are examples of errors:

- Applying a higher dose than strictly necessary (even when not violating safety limits!)
- Taking more time for an examination than necessary
- Using more contrast fluid than necessary

- Producing sub-optimal images and thereby endangering good diagnosis.

It should be noted that it is very difficult to assess whether one of these errors has occurred, because of the necessary balancing between different aspects. For instance, taking less time for an examination may lead to less patient comfort and less accurate diagnosis. Applying a lower dose may lead to a lower quality of images.

There are many reasons why an error may occur:

- Sub-optimal operation of the system, i.e., not using all functions of a system or incorrectly using them, or not operating the system in the most efficient way
- Sub-optimal system design, i.e., the system does not support users' tasks sufficiently, the system is difficult or slow to operate, or the system is not designed for the particular examinations for which it is being used,
- Sub-optimal workflow management.
- Non-optimal diagnosis, for instance, not optimally using results from medical research. This issue is addressed in evidence-based medicine (ref. 3), and is outside the scope of this paper.

Often a combination of these occurs: the clinical user does not use a certain function of the system because the design of the system does not make the purpose of the function clear.

The basis of good user-interface design lies in applying knowledge of users, tasks and the context of use. This is the essence of the user-centred design approach, which we have practised from the outset. The analytical activities undertaken are now described.

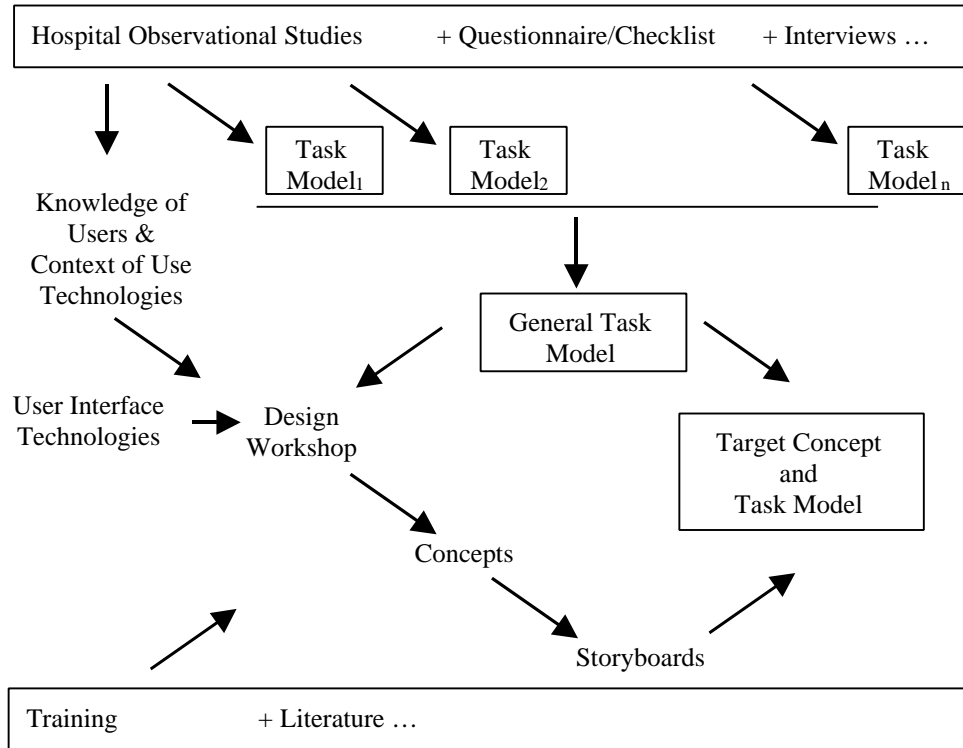


Figure 1 – User-centred Design Activities Conducted

Analysis

The range of activities we undertook is illustrated in the process map shown in Fig. 1. We have built up domain knowledge via training sessions at Philips Medical Systems (PMS), observational studies in hospitals, interviews with PMS' application specialists, and studies of manuals of some current Philips systems (the MD4 general radiology system, the BV300 surgery system, and the BH5000 cardiology system).

Organisation of observational studies: Visits were organised to five hospitals with which PMS has a special relationship. An application specialist of PMS who was very well acquainted with the hospital would accompany us. Application specialists are attached to hospitals and maintain frequent contact with clinical staff at all levels; they also deliver on-site training and support. The whole process was guided by a checklist (see Appendix).

Typically we would start and end a visit with a conversation with the application specialist. We were interested in the type of hospital, the

specialisms of the department, the kinds of examinations conducted, the number and types of people working there, the types of systems used, the process used for introducing new systems into the hospital, trends in the hospital, and problems experienced and anticipated.

Next, we would be guided through the department and begin our observations; each observer typically doing about eight. During the examinations, we would stand in the control room, or the examination room wearing protective tunics, and observe what was going on. We would draw plans of the rooms, and record the actions of and communications between the clinical staff and patient, noting information present on walls, and ambient conditions, for example, lighting and noise. We would not only look at what happened during an examination, but also between examinations.

We questioned the staff present, namely radiographers, radiologists, cardiologists, and nurses. We were fortunate to be able to talk to very experienced staff and also to trainee radiographers and doctors. We would ask about the nature of their work, their roles, the problems

they experienced, the changes they would like, the trends in their work, their training, and what they liked in their job.

Observational studies conducted: We have performed observations in hospitals in the Netherlands, Belgium, and the UK (ranging from film-based to a digital hospital). We have seen both diagnostic examinations and interventions, varying from general radiology, to MR, CT, and cardiology. We have also seen some other parts of the workflow, namely reporting and archiving. The observational studies are an ongoing activity, starting off as a way to get to know the domain, the users, their tasks, and the environment, and now becoming more specific relating to the prototype system we are developing.

Samples of information obtained: We have recorded each hospital visit in a report (on average 25 pages, typical sections: users, systems, environment, tasks, information, training, UI issues, trends). Some fragments are included below (see also Figures 2 and 3):

“There is considerable ambient noise in the examination room: an intercom system is in use for broadcasting announcements, motors and jets may sound from the X-ray equipment, the people present in the working area talk amongst themselves, music may be played on a hi-fi system. The noise is intermittent.

There is no daylight, only artificial light - incandescent or fluorescent is available, usually lighting is soft, it may sometimes be dim, lighting levels may be changed during an examination.”

A small bowel enema examination:

- The patient is selected from the RIS list.
- The examination-table is put in position. In the meantime, the patient is brought in.
- Local anaesthetic is applied (using a spray).
- A tube is prepared and inserted in the patient's nose.
- The patient is put on the table, on his back.
- The C-arm is moved.
- The monitors are moved.
- The lights are dimmed.
- The table is moved up, the C-arm down.
- An image is taken.
- The tube is inserted by a radiographer in the patient's nose and down to the stomach while looking at fluoroscopy images on the ceiling mounted monitor. The fluoroscopy is operated by the other radiographer at the table operator's control (TOC).
- The one radiographer tells that the tube is in position. The other radiographer releases the fluoroscopy footpedal.
- The intercom is used to inform the radiologist that room 7 is ready. In the mean time, the infusion with Barium is prepared.
- The radiologist enters the room, puts on a lead tunic. - He looks at the paper in the control room and speaks to the patient.
[Various announcements are heard on the intercom.]
- The infusion is attached to the nose tube.
- They wait for the contrast to reach the right spot, some fluoroscopy is done.

In the mean time, the patient is covered with a white blanket.

- Two images are made.
- They wait again. Fluoroscopy is used, images are made. The C-arm is moved, using the TOC. This is repeated a couple of times.
- The patient repeatedly tries to look at the monitor, but has difficulty doing this as the monitor is behind his head.
- The patient is put on his stomach. The tube in his nose is put in the right position, using a kind of scissors or tang.
- An alarm sounds (a collision?).
- The patient is put on his back again. The C-arm is moved. Fluoroscopy is used, an image is taken.
- The patient is instructed on the position. The table and C-arm are moved. The radiologist looks at the monitors, then walks towards the patient to change something: he is pressing the Barium along. This is repeated.
- A lot of changes in geometry take place, and images are being taken. At some point the radiologist says it is okay.
- The table is moved down and the tube from the nose is removed.
- The patient's face is cleaned.
- The patient is helped of the table.
- The monitors are moved to get them out of the way.
- Forms are being filled in, with number of images taken, exposure duration, etc.
- The room is cleaned.

Figure 2 – Sample of an Examination Observation

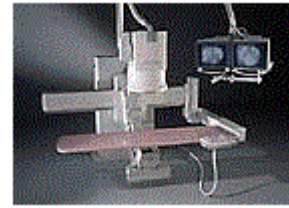
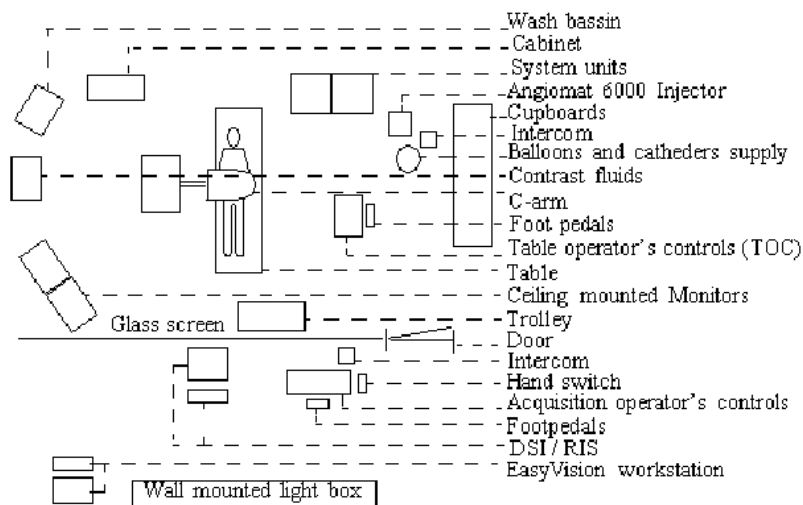


Figure 3 – Sample Sketch of an Examination Environment and Pictures of an X-ray Setup.

Main observations: We have made the following observations during hospital visits:

- [Obs1] Radiologists and radiographers often only know part of the available system functionality, and not always how to accomplish their task efficiently.
- [Obs2] People with different backgrounds may have to operate the same system. For instance, surgery systems are often operated by cardiac radiologists as well as by nurses, for different types of examinations. The growing and complex functionality, for instance, for cardiac applications, further raises the threshold for the nurses to operate the systems efficiently.
- [Obs3] Hand-written reminders, tips, and other information, for instance on allergic reactions to contrast fluid, are stuck to the walls or the lead glass screen in the control room.
- [Obs4] *Training* takes place during a number of days on installation of a new system. A small number of clinical users are trained: initially by giving them a system overview, and then by assisting them during real examinations. These clinical users then go on to train their colleagues. So, a substantial part of training is conducted on the job. Colleagues exchange the odd tip about, for example, which command sequence gives a better image, and stick to this. In our hospital studies, user manuals were almost never seen in control rooms.
- [Obs5] *Device controls are spread around* – by the examination table, on a trolley-mounted console, a remote control for a ceiling mounted monitor, etc. We have observed physicians occasionally actually running from one to the other. Radiologists in the examination room have been observed shouting commands to the radiographer operating the system from the control room. Additionally, the latter may shout instructions to the patient: “hold your breath!”, “don’t move!”, etc.
- [Obs6] Due to the range of systems from different manufacturers, and the lack of industry standards, controls vary from system to system causing confusion amongst clinical users who must repeatedly move from using one to another.
- [Obs7] When inserting a catheter, the physician must be sterile, which can make system control harder.
- [Obs8] We have observed instances where staff and patient do not have a common language making it difficult for correct

instructions to be given to the patient. We will not address this issue in this paper.

Trends in the clinical domain: The following trends can be observed in the hospitals:

- [Trend1] An increasing amount of functionality is being offered in systems. For instance, the manual for Philips' Cardiology X-ray system has almost doubled in size, increasing from 120 pages for the BH3000 to 245 pages for the following system, the BH5000.
- [Trend2] Boundaries between domains are becoming fuzzy. For instance, the boundaries between diagnostic and interventional radiology, and between cardiology and surgery are diminishing. Increasingly, for example, cardiologists prefer when performing a minimal invasive by-pass operation to also place an abdominal stent (i.e., a small intervention in a leg normally performed by a radiologist) when needed.
- [Trend3] There is a shift from stand-alone to networked systems where the medical equipment is effectively a peripheral. Increasingly, there is a requirement for equipment to be based on open standards, and for inter-operability with systems from other manufacturers.

- [Trend4] Information is increasingly stored digitally, giving rise to film-less departments.

The above observations and trends are not exhaustive, but are illustrative of some of the material on which we based subsequent work.

After each hospital study, we developed task models. This activity is now described.

Task Analysis and Modelling

A Task Model is a drawing that represents the tasks undertaken to achieve a goal. It is a hierarchical drawing as tasks may be decomposed into sub-tasks. It is read from left to right and captures the process used by the subject in going about his or her job. It is also independent of any technology, user-interface or other, as the level of abstraction is much higher than mouse and keyboard interactions. Task Models are of two types: those representing currently used systems, and those representing the target system to be designed. An example-fragment of the former kind is shown in Figure 4. We have based our task modelling work in part on the MUSE method described by Lim & Long (ref. 4). Note: The relation between a Task Model and a system is a subtle one: though a task model is rooted in one or more systems it is only a description of tasks, and is entirely independent of system-level details.

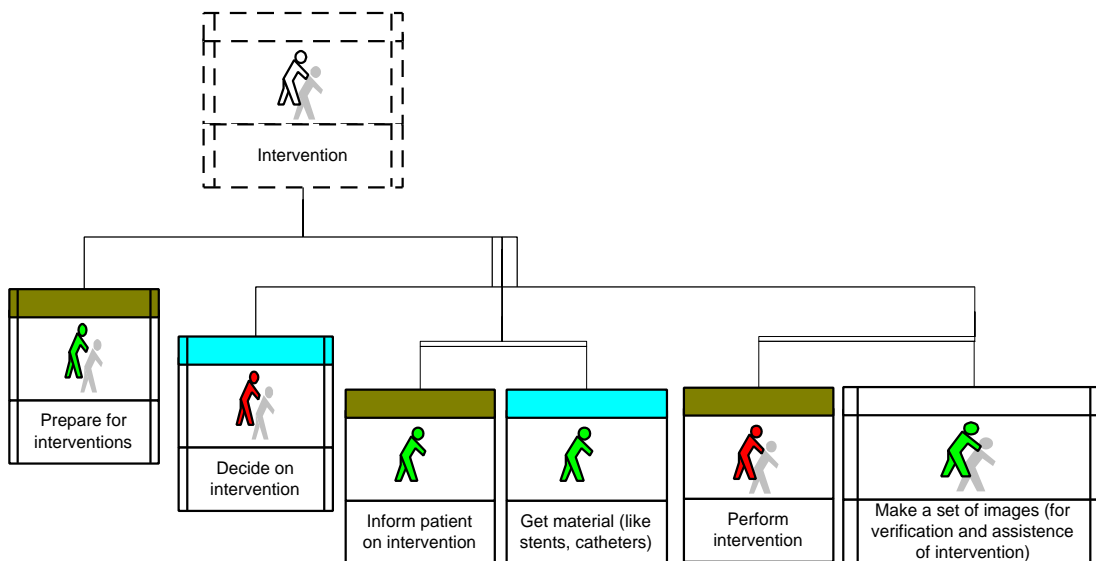


Figure 4 - Fragment of a generalised task model of existing systems

Benefits of task modelling:

1. Capturing important characteristics: In the observations we conducted it soon became clear that task characteristics such as location, communication, co-operation and duration would be useful to record. These could constrain design solutions (e.g. co-operation between users), or point to where improvements would be most effective (e.g. duration). These can easily be recorded as annotations on the Task Model.
2. Communication: Having a common frame of reference is useful both within the project team, and when communicating with partners. The task model proved to be very useful when collaborating with others during idea generation and interaction design.
3. As a basis for design: Comparing task models is a good starting point to find commonalities and differences which impact user-interface design. Once agreement on the desired task structure has been obtained, candidate design solutions can be brainstormed. Optimal functional allocation between system and user, can then be followed by interaction and graphic design.
4. Validation: Task Models can be shown to application experts to check that descriptions are correct. The notation (explained below) is simple enough to understand.

Notation:

Task Ordering: Two types of ordering are distinguished in the task model diagrams: Sequential Ordering (indicated by a single horizontal line, tasks ordered from left to right) and Free Ordering (indicated by a double horizontal line).

Suppose a task A consists of subtasks B, C, D, and E. Figure 5 indicates that the subtasks should be done in sequential order: B followed by C followed by D followed by E.

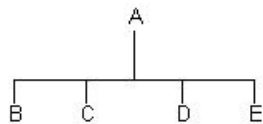


Figure 5 – Sequential Ordering

Figure 6 indicates that the ordering of the subtasks is free, i.e., unconstrained: B, C, D, and E can be done in any order.

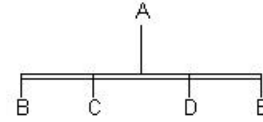


Figure 6 – Free Ordering

Figure 7 shows a mixture of sequential and free ordering: it indicates that B should be done before C and D, which should be done before E. The order in which C and D are done is not important.

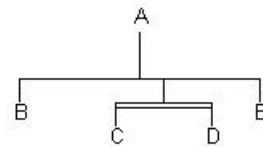


Figure 7 – Mixed Ordering

Repeatable tasks: These are indicated by multiple vertical bars connecting a task to its parent task.

Optional Tasks: These are represented by boxes drawn with dashed lines.

Location of tasks: As we are only interested in two locations (examination or control room) we have used simple colour coding in the top of the box to represent this property. Brown (which prints as a dark shade in grey-scale) represents the examination room; blue (a light scale) represents the control room. An absence of colour implies no restriction on location.

Person: This indicates any demarcation on tasks amongst radiologists and radiographers. Red (which prints as a dark shade in grey-scale) represents tasks done by the former; green (a light shade) the latter.

UI-technology: Customisation

Reflecting on our observations, it occurred to us that there was a need for more flexible systems – unconstrained by requirements to be used at a fixed location, or by certain kinds of users, or by fixed interactions. Additionally, there was potential for a system which could be pro-active in enhancing the proficiency of the user.

Customisation technology is the approach we followed. The Microsoft Office Assistant is an example of customised assistance (in this case of context-sensitive help). We can expect professional users to expect such features in the systems they use, after exposure to them in Word processing systems. We now discuss customisation in the user interface.

Customisation is the process of adapting a system's behaviour (by the designers i.e. system developers, by the system itself, or by the user) to suit the individual or group using it. In a sense, customisation takes place in each system design. The important variables are to *whom* the system is customised, *what* is customised, and *how* the customisation is done. These are discussed in turn.

To whom is it customised: Customisation is normally aimed at "the customer". In the medical domain, this may be the radiology department of a certain hospital. Each department may have different wishes with respect to the kind of functionality required, may want to use the system for different tasks, may have its own procedures, its own terminology, etc. Each customer can also have a different environment in which the system will be used, for instance, different lighting and noise conditions, different cultural background, etc. Usually a system is designed with certain prototypical customers in mind, having different versions of the system for different categories of customers, and providing options to configure a system to the wishes of the customer. In the medical domain, for instance, different versions may be made for the different countries (or regions), or for low-end and high-end applications, while there may be options for incorporating extended functionality. As every customer tends to be different (and no rough categorisation will be perfect) more customisation may be needed than can be achieved through these means.

Even for one customer (say a certain radiology department), many people may be involved with the system (like radiographers, technicians, administrators, radiologists, surgeons, nurses, etc.). The satisfaction of the customer then depends upon the satisfaction of a group of people, who may have different (sometimes even conflicting) requirements and preferences. Depending on their rôle, they will want to use the same system for different tasks, they may have a different level of education and

experience, etc. To a certain extent, it may be possible to use stereotypical roles, and customise the system to these roles. However, these roles and their corresponding characteristics (like level of education) may differ per customer group (for instance, the education level of radiographers and the tasks they are permitted to perform differ according to national regulations).

Differences may also exist between individuals having the same role. There is a lot of literature on individual differences between people that may affect their operation of systems. Straightforward differences are, for instance, the level of experience, and the level of domain knowledge. Other differences mentioned are differences in psycho-motor skills, learning ability, understanding, expectations, motives, cognitive strategies, cognitive abilities, preferences, etc. These individual differences may impact the best way to present information for a certain user on the screen, the amount of information needed, the input device preferred, etc.

Over time, the system will be used to perform different tasks under different conditions. For instance, in the medical domain, different patients are examined, and different types of examinations take place. Customisation can also take place to these different situations over time.

In summary, a system can be customised to the environment in which it is used, to the (group of) users that operates it, to the tasks that these users perform in general and at a certain moment in time, to the situation at hand.

What is customised: Many aspects of a system can be customised. For instance, in the Appeal tutoring system (ref. 5), various aspects were adapted at run-time to the individual student. Examples are the navigation path, timing and amount of instruction, form of information (e.g., using formulae or pictures), (amount, tone and timing of) feedback to the student, and the content of exercises. In an adaptive component of a fighter aircraft (ref. 6), the system adapted task allocation on the basis of the pilot's state, system state and world state.

In general, a system gives the user the opportunity to give input, i.e., to make choices, and presents output depending on the choices made. For instance, an X-ray system gives the

user the opportunity to set several parameters (like dose, geometry position, shutters), and the system produces X-ray images accordingly. As such, a system can be said to consist of:

- *Parameters* that the user can set, such as Brightness and Contrast, each with an associated default and range.
- *Functions* that the user can apply to change the value of parameters.
- A *Navigation Structure* determining which functions are directly available to the user at a certain moment in the interaction and how the functions are grouped (this includes the menu and dialogue structure).
- *Ways of Interaction* determining how the user can operate a function and how output or information is presented.
- *Information* regarding which functions the user can apply and how to apply them, and which tasks to perform without the system (like what to tell/ask the patient).
- Features determining the *appearance* of the system such as colours, shapes, icons, sounds, labels, and fonts.

All of these aspects can be customised. For instance, the preferred value of a parameter (such as Brightness) can depend on the task (namely the particular examination type), individual preferences of the user, the environment (such as the lighting condition in the room) and the values of other parameters (like Image Polarity). The default offered to the user could reflect this. Similarly, the preferred way of interaction may depend on the availability of hardware (like microphones), the examination type, and the experience and preferences of the individual user. For instance, during an intervention the radiologist has to be sterile and the use of speech input for non-critical tasks may be preferred. On the other hand, users who have operated a certain function in a certain way for a long time will not always like to change.

The preferred navigation structure may depend on the user's rôle (for instance, a nurse normally does not need advanced functions), the current task, and the experience of the individual user. How is it customised: There are several ways in which customisation can take place (see also reference 7):

- System designers can search for a general design that is customised to most of the users' needs
- the interface can be designed to incorporate the means for users to customise the interface themselves
- mechanisms can be devised and built in that are able to determine a user's particular needs and preferences and automatically customise the interface accordingly
- it can be attempted to "customise the users to the interface" by providing support for training users.

Mostly, a combination of these options is preferable.

In every *run-time* customisation process four phases can be distinguished (ref. 8): in the first phase a customisation is suggested, in the second phase a number of alternatives are offered for the customisation, in the third phase a choice is made from the alternatives, and in the fourth phase this choice is carried out. Each phase of the customisation can be performed by the system or the user.

If the system performs all phases, the process is called Self-Customisation (e.g., ref. 9). In case the user performs all phases, the process is called User-Controlled Customisation (e.g., ref. 10). We are most interested in hybrid situations, in which depending on the situation, the phases can be performed by either the system or the user. A primary reason for intermediate situations is that the user may have more information than the system in some areas, especially concerning his or her own preferences, while the system may have more information in other areas, especially concerning its options. The system may, for instance, suggest alternative available methods for increasing Brightness from which the user can make a choice. A second reason for intermediate situations is that the user may feel more at ease knowing that he or she is in control (ref. 5).

Constraints: There are several issues to be considered when designing self-customising systems (see amongst others ref. 11).

One potential problem is that it might become more difficult for the user to learn to operate a system, if it changes over time. The acceptability

of the system may also decrease, as the user might not understand the system's behaviour. To compensate for this it has been suggested that the user be given an explicit model of the customisations being undertaken by the system. For instance, the user may be notified that the position of items on a list may change according to their most frequent choices. In this way the user can anticipate, or at least rationalise the actions being taken by the system.

Norcio & Stanley point out that users may feel a lack of control if the system starts to make choices on their behalf (ref. 12). This may be prevented by giving them at all times the chance to override the system's decisions, or by only giving them suggestions. An alternative is to give the users the opportunity to turn off automatic customisation without prior user confirmation. In this way they have ultimate control, the knowledge of which may be sufficient for them to accept customisations. Also, a system should know when and when not to consult the user about a certain customisation.

Related to the previous issue is the issue of safety and legal responsibilities. Especially in the medical domain, these issues are very important. The operator should have the final responsibility for the settings chosen, if these settings have an impact on the safety of a patient.

In order to self-customise, a system has to gather and store data about the user, his tasks and preferences. There are, of course, privacy issues related to this. For instance, not every radiographer might appreciate it if a supervisor could inspect the way they normally use the system. So, there have to be rules about who can access which data.

Users may also not like to be categorised. Great care has to be taken when using stereotypes.

Concept generation

After the analysis phase (in which we had already identified some options for improvement), we started to generate concepts for new systems. For this, we needed individuals skilled in ergonomics, industrial design, interaction design, and graphic design. We were able to call on colleagues in Philips Design, for this skills set.

First we organised a workshop involving four designers (three with experience in consumer systems like audio systems, one with experience with medical systems), three researchers, an application specialist of PMS, two people from PMS Predevelopment, and a very experienced radiographer. The purpose of the workshop was to come up with far-out ideas. This is typical for a research project. In a more development-like project, you would also come up with concepts, using focus groups etc, but these concepts would (intentionally) have shorter time horizons for realisation.

Next, five concepts were worked out in more detail, into storyboards, and made more realistic. The designers involved (both without previous experience in the medical domain) were brought up to speed by reading the hospital visit reports and studying the draft generalised task model. The task model was used as a basis for this work: we identified which sub-tasks were closely related to a concept and visualised how the concept could impact these tasks. The storyboards were presented to PMS and we used some criteria (like technical feasibility and marketing potential) to decide which concepts to proceed with.

Figure 8 shows some fragments of storyboards.

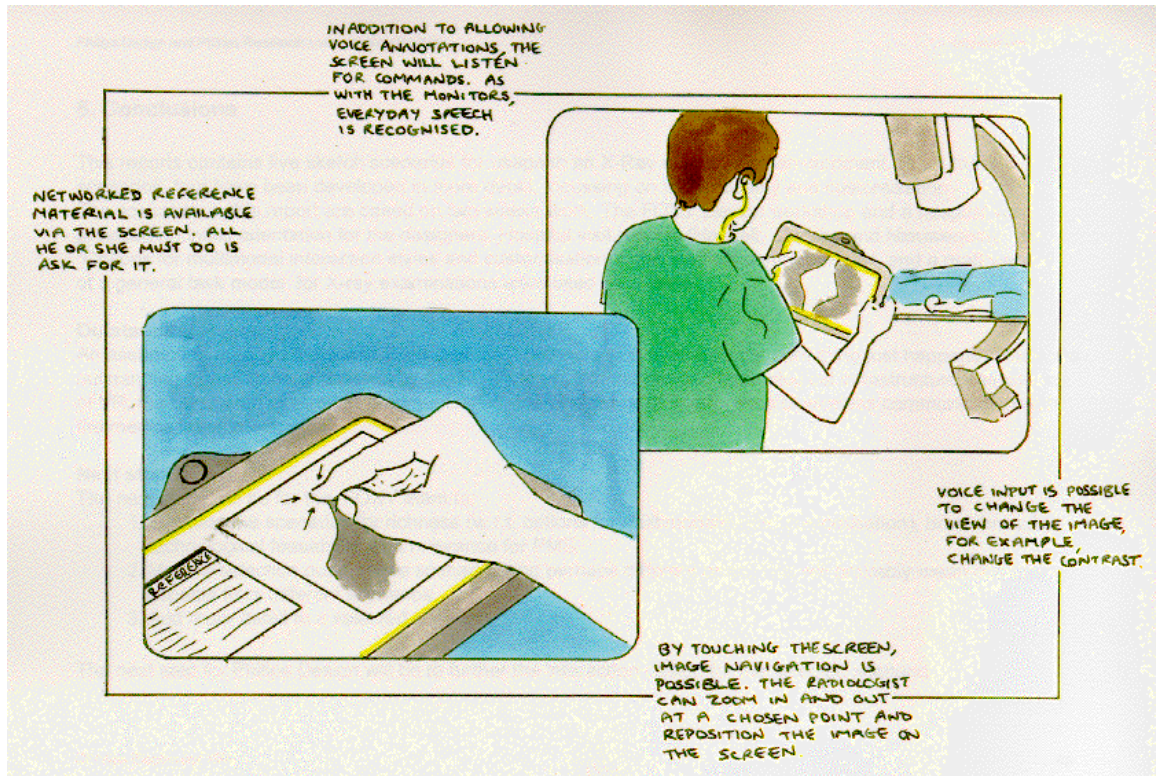


Figure 8 - Fragments of two Story Boards

Ways to reduce human error

From our hospital observations, it can be concluded that current systems are being used sub-optimally. This is a cause for concern, as trends in the clinical domain indicate that sub-optimal use will continue to increase in line with system functionality and complexity. According to our definition of human error in the clinical domain, sub-optimal use is a main source of errors. In order to ensure optimal use, the clinical users need to know:

- *Which functions can be useful in a certain situation.* For instance, when performing a Subtraction task (i.e. subtracting a reference mask image from another image), motion artefacts can occur due to patient movement. The user needs to know that the 'Pixelshift' function can correct this (by shifting the relative positions of the mask and source images). This relates to observations Obs1, Obs2, and Obs3.
- *When to apply a certain function.* For instance, Brightness can be set whenever viewing an image, but it is advisable to set Polarity, electronic shutters, and optionally perform subtraction, beforehand.
- *How to operate a certain function in a certain situation.* This concerns actions, their sequencing, and timing. For instance, in a future system, one could envisage that a Zoom function could be activated by a voice command and then, touching the area that is to be the centre of the zoomed image. There may be alternative ways of operating the same function. During some interventions, hands-free or even eyes-free operation may be needed (cf. Obs7 and Obs5). If the background noise is too high, speech recognition is impaired and another way of operating the function may be more appropriate. Additionally, the experience and preferences of users is a factor (cf. Obs2 and Obs3).
- *What the scope of a function is.* For instance, when changing Brightness while viewing an image, the effect can be a change in Brightness of all images in that set, all subtracted images, or only that single image.
- *How to undo the effect of applying a certain function.*

The system needs to support the various users in performing their tasks and obtaining the knowledge relevant in a certain situation. We try to achieve this by a combination of:

- (1) a well designed user-interface (including consistent operation, cf. Obs6),
- (2) customised default values minimising the amount of effort required from the user, and
- (3) active support by the system for the learning process (needed as training takes place on the job, cf. Obs4).

These are now discussed in turn.

Improved design of the user-interface:

Designing a user-interface requires us to answer the following questions: (1) which user tasks to support, (2) which functions to provide in order to support these tasks, (3) what navigational structure and screen layout to use to make these functions available, (4) how to enable the user to operate these functions, and (5) what information to present to the user about the functions.

The first question leads to the construction of a task model of the target application (as discussed in the task modelling section). This task model is based on (improvements of) the generalised task model of existing systems and the concept generation activities. A fragment of a task model for a target application supporting Viewing and Post-processing of images is presented in Figure 9. The notation is similar to the notation above, except that location and person indications have been omitted, as all the tasks are intended to be performed in the same location and by any user type. We are currently in the process of validating the task model with application specialists and clinical users.

The second and fourth question lead to the determination of which functions to provide to a user to assist with the leaf-tasks of the task model and which interaction to provide for these functions. To make this decision, we have analysed the way in which these tasks have been implemented in existing systems (such as existing X-ray, surgery and cardiology systems, but also completely different systems like drawing packages). Such an analysis leads to a choice inspired by best practise across many domains, as much as the need to balance consistency with existing practise in the medical domain.

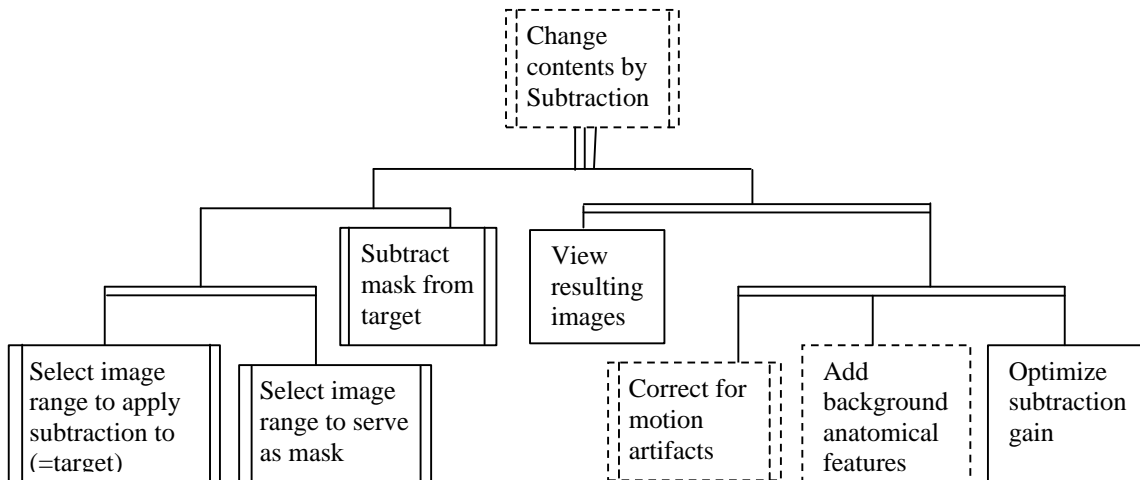


Figure 9 - Fragment of a Task Model for a Target Application

The third question (about navigation structure and screen layout) can be answered on the basis of the task model, the functions and interactions chosen, and the available space on the interface. The ordering in the task model influences the navigation structure and screen layout of the system as follows. A task that has to be performed before other tasks, might be better represented on an earlier screen. Tasks that can be performed in any order should preferably be present on the same screen. In western systems, tasks on the same screen will normally be ordered from top to bottom and from left to right. For example, in the case of a sequential ordering of tasks A and B, the functions to enable task A should either be on a screen that precedes the screens with the functions of task B, or they should be to the top (and left). It should be noted that even in the case of free ordering, the layout of the task model drawing may lead to conclusions from, say, the screen-layout designers: there is a high probability that in a free ordering of a number of tasks, the order from left to right will be used as a normal (though not required) ordering. Therefore, we have ordered the tasks in the task model within a free ordering from left to right according to a likely sequence. Notes have been added to the task model to indicate more than one likely sequence, or to state conditions under which a certain sequence is likely to occur.

The grouping of subtasks in the task model, reflects the fact that certain tasks are conceptually more related to each other than to other tasks. In the user interface this could result in visually grouping functions related to those tasks (for instance by proximity).

The fifth question (what information to present about the functions) is addressed below in the section on active support. In principle, all relevant information, which the user does not know has to be presented. The different types of information have been discussed above.

Customised default values: The main input of a user to a system can be seen as the setting of parameter values, such as, for instance, choosing brightness and contrast values, determining the current image by image navigation, etc. Often, a system will provide the user with default values (these may also be the last values used), which the user can modify to his own choice. The closer the default values are to the preference of the user, the less effort is required from the user to change them. Ideally, the user does not have to do anything, as the values offered by the system already reflect his preference. This is very difficult to achieve, as the situation in which the parameters have to be chosen changes continuously, for instance, as the user is viewing different types of images. However, as a step in this direction, we have developed a so-called Customisation Engine, which determines the default values (and ranges between which the parameters can be modified)

on the basis of the current situation, and the preferences of the (individual) user.

The Customisation Engine uses (see Figure 10):

- *Rules* specifying the range of parameter values regarded as good in a certain situation. We distinguish two types of rules: constraints and suggestions. Constraints specify the allowed values of a parameter in a given situation, while suggestions give good default values.

The situation in which a rule is applicable can consist of a number (concatenation) of situational conditions. Each situational condition specifies a range for a certain parameter. For instance, the following can be situational conditions: “Examination type in {Barium, ERCP, Skillet, Myelo}”, “ $12 \leq \text{Brightness} \leq 25$ ”, “Subtraction=Off”, and “Physician =Dr Spock”.

It is possible that more than one rule is applicable in a given situation. In that case, the intersection of the ranges of suggested or allowable values for the parameter is used. If this intersection is empty, the rule that uses the most situational conditions is applied.

- *The History of the Interaction* reflecting how the user has reacted to suggested default values in the past. For instance, when an image is presented to the user with a default

Brightness of, say, 5, then it is stored whether the user has kept the Brightness at that level (i.e., implicitly agreed to the offered value; this is called a “match”), or to which value he has changed it (i.e., did not agree to the offered value; this is called a “mismatch”). The values of other relevant parameters at the time when the user made his decision (like the Image Polarity and the type of examination) are also stored.

The history is used to learn the preferences of the user at run-time. For instance, when the user repeatedly increases the Brightness value of images of a certain type then it is concluded that the user may have a preference for a higher value in that situation. The speed at which the system learns is directly related to the number of history instances it takes into account (the more instances the lower the speed). If the learning speed is high then customisation takes place rapidly but is prone to misunderstand the user. If the learning speed is low then customisation is slow but more accurate. The learning speed should be customisable by the user (or the system could learn it itself based on observation of correct and incorrect customisations in the past).

It should be noted that no deep modelling of the user takes place, but that decisions are almost directly based on the data stored in

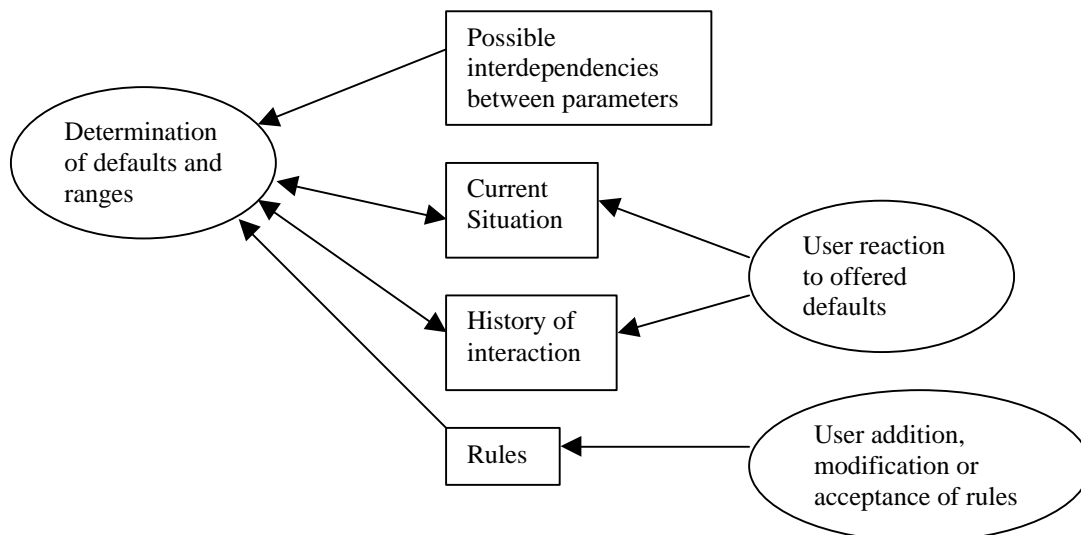


Figure 10 – Schematic Representation of the Customisation Engine

the history.

- The *Current Situation* specifying the current values of all the parameters. The current situation determines whether a rule is applicable or whether a certain piece of history information is relevant.
- *Possible Interdependencies between parameters*. In order to make the use of the history information for learning easier to realise, information is available on which parameters are likely to depend on which other parameters. This makes it possible for the designer to exclude highly improbable dependencies from being considered, like the preference for an input modality depending on the Brightness of the image.

The use of rules described above is not only applicable when determining the default values and ranges of the parameters. It can also be used to customise other types of parameters. For example, interaction style parameters (e.g., does the user prefer touch or speech), implementation parameters (e.g., what is the maximum time between two touches for the system to regard them as part of one action), and navigation structure parameters (e.g., “if Subtraction is off then Availability of Pixelshift function is false”). Learning the preferences of the user through observation is, however, more difficult and requires different mechanisms, because preferences of the user can only be determined as above if the user is presented with a choice.

Interaction for customisation: As mentioned above (see sub section “Constraints”), it is very important that the user keeps feeling in control, understands the adaptation taking place, and is not hindered in his learning of the interface by continuous changes. Therefore the interaction style used for the customisation – i.e. the way customisation is presented to the user and the way the user is involved- is crucial.

In the description of the Customisation Engine above, various moments can be seen at which such an interaction might occur:

- Parameters are presented to the user, with certain defaults. After some observation of the user’s actions in the past, the system may conclude that the preference of the user is different to what is programmed in its

rules. The following options are now possible:

- (1) The user is presented with a new default, with an indication that the default has changed by making it very conspicuous
- (2) The user is notified that a deviation in his behaviour from the standard settings has been observed, and asked whether he would like to change the standard settings to a newly suggested setting reflecting his recent behaviour,
- (3) The user is presented with the old default, but it is made very clear that the system has observed a new default, and the user can get this default if he wants it (without being explicitly asked).
- (4) The user is presented with the old default, but at a moment when the user is not busy, like between patients or at the beginning or end of day, he is notified about the observed deviation and given the opportunity to add a new rule, or modify an existing one.

A problem we have found (during informal evaluations) is that it is difficult for users to realise that changing a rule during a certain examination may have implications on other similar examinations in future (even though this was clearly stated in the question asked to the user). Therefore the option to change rules when the user is not busy with a specific examination seems preferable. We still have to conduct user-evaluations to see which option or combination of options is the best.

- When the user changes the value of a parameter, then the best default for another parameter may change because of interdependencies. The following options are now possible:
 - (1) The defaults (and perhaps even the ranges) of the dependent parameters are changed automatically, with a clear indication that they have changed
 - (2) The user is notified about the interdependencies and given the opportunity to accept or reject them (this holds only for defaults, ranges have to be accepted).
 - (3) Nothing is changed automatically, but the user is informed that the system sees an opportunity for optimisation and can

obtain this optimisation if he wants it (without being explicitly asked).

The acceptability of automatic interdependency changes has to be investigated, especially when it concerns automatic changes of a parameter that the user has already explicitly set. (For instance, the user sets the Brightness value, then determines to change the Polarity, which has a possible implication on the Brightness value.)

We have the hypothesis that the third option is the best, but we have to evaluate this in practice.

In addition to these two moments of interaction, there is also the explicit changing of rules by the user. We have explored a way to enter knowledge that may be more familiar to clinical users than the use of rules: the user is presented with prototypical images and can change the appearance of these images, or chose between alternative appearances of images. Rules are deducted on the basis of the results. This work is still in a preliminary phase.

Active support for the learning process: It is impossible to give a user all the knowledge mentioned above during a short training period. However, more training time is unlikely: users generally have to learn on the job. Therefore, we are working on active support by the system of the learning process. The support has to be highly customised, depending on the hospital, the experience and preferences of clinical users, the examination being conducted, and user roles.

There has been quite some research into the design of intelligent help (see, e.g., reference 13), focusing on providing the users with good ways to ask questions and on volunteering answers when the user is observed to have a problem. In that research, the user-interface is seen as a constant factor. We have taken a different approach, and are focussing on changing the user-interface to increase learnability. We see both approaches as being complimentary.

Learnability is mainly related to efficiency and effectiveness over time. Preferably, the system should be highly intuitive to use the first time, i.e., have an immediate high effectiveness without much effort on the side of the user, and efficiency should increase over time, leading to a highly efficient system for experienced users.

Moreover, systems should support the rapid progress of users from initial to experienced level.

In the system we consider (and this holds for most systems), only a limited amount of space is available for the presentation of functions and information. Nevertheless, we know for the X-ray viewing and post-processing domain that the users want many functions to be available at the same time, without having to take extra navigational actions (so, in manner of speaking “within one button click”). By using a good user-interface design, and making only those functions available which are appropriate in a certain situation (see section above and Customisation section below) it is to a certain extent possible to meet this user wish. If we include the use of speech commands to operate functions, it is even possible to have functions available that are not immediately visible on the system. However, to present all allowable functions on the system or hide them by using speech commands, has a rather negative impact on learnability. There is insufficient room to explain what all the functions do, and how to operate them, and this is particularly acute when the user is presented with a wide choice. So, our concern for novice users conflicts with our desire to provide highly efficient systems to experienced users.

One option may be to provide separate interfaces for novice and experienced users. However, this has only limited advantages: a user who is familiar with the use of certain functions (such as setting Brightness and Contrast) may never have heard of other functions (like Pixelshift) which can be very useful for his current situation (say, performing a Subtraction with motion artefacts). This is something we have actually observed in our hospital visits. The option of two separate interfaces also poses the danger that the gap between novice use and experienced use becomes too large.

What we would like, is that, for instance, when the user is performing a Subtraction, and has not used Pixelshift before, the system indicates that this function can be used to remove motion artefacts, and how to use it. This means that we will not use two separate interfaces, but one interface which gradually changes to reflect the experience of the user (i.e., the navigation structure and information is customised to the experience of the user).

The (simplified) example illustrates how this could work (see Figure 11, note that it uses a

touch sensitive screen). If the user is not familiar with the concept of subtraction then this concept is introduced (image 1). Next, the user is guided

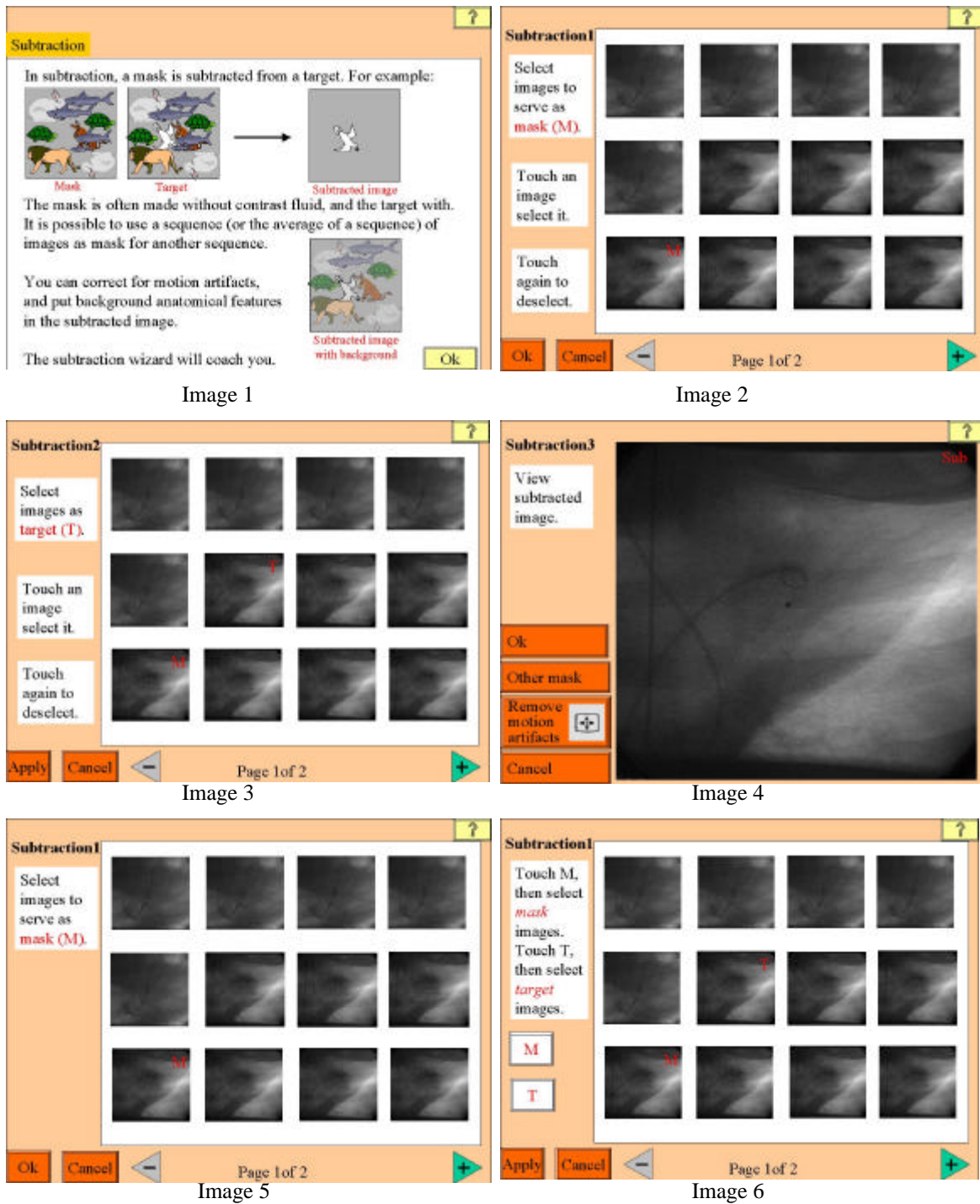


Figure 11 - Example of Active Support for the Learning Process

through the subtraction task. At each step, information is given on what he should do, how he should do it, and how he can correct mistakes (images 2-4).

Note that the initial sequence presented (images 2-4) is directly based on the task model (see fragment of the task model in Figure 10). Initially the user is forced to do the tasks in the task model from left to right, even though the model allows free ordering. When the user gets more experienced, the free ordering is introduced in the user interface (compare images 2,3 and 6). Note also that the information presented reflects the types of information mentioned above. What information is presented to the user depends on his experience. If the user has never used a certain function before (and not seen the information related to that function recently), then information regarding the purpose of the function is given, and the user is instructed how to operate it. More detailed information regarding the steps to take is given only when needed. For instance, if the user already has experience with selecting images, as part of another task, then no information is given anymore on how to (de)select an image (compare images 2 and 5). The user can get this next level of information if he wants it after all, by explicitly asking for it with the help function (available from the button marked "?"). We envisage that users will also be able to explicitly select the level of information they like for certain functions. This enables experienced users to still use more structured interfaces if they prefer that.

Conclusions

Our main interest is in exploring how new user-interface technologies can be applied to the design of medical systems, so as to reduce sub-optimal system usage in the X-ray examination and control rooms. This has a direct impact on patient throughput, which is expected to continue to be an important measure in the healthcare industry. We have described how we have applied user-centred design methods basing our work on first-hand observations conducted in hospitals, and on Task Analysis and Modelling.

Various problems have been observed during field studies. In this paper, we have suggested a number of ways these could be addressed:

- (1) Improved user-interface design based on task analysis and customisation can make it easier for people with different backgrounds to operate the same system. The increasing amount of functionality can be shielded. By basing the design on the task model, it can be prevented that users have to run from one place to the other in order to operate the system. By relating functionality and interaction styles to tasks, consistency over different systems can be better achieved. Customisation is also an option to allow the operation of systems to be customised to what the user is accustomed to.
- (2) customised default values minimising the amount of effort required from the user. The customisation uses a combination of pre-programmed rules, and on-line system learning. A prototype system has been implemented. We have thought about the interaction styles needed for customisation, but evaluations have to be conducted to test our hypotheses.
- (3) active support by the system for the learning process (needed as training takes place on the job). We have sketched a way in which the learning process of the user is supported by gradually changing the interface to reflect the user's growing experience.

Our work is ongoing; we are presently in the midst of building an integrated research prototype, which will be tested in usability evaluations.

Appendix: Checklist for hospital visits

1. Background

Obtain or develop a global organisation model.

2. Workflow in X-ray examination

- Who are the people carrying out tasks? (Position, level of experience)
- What objects are used when performing the tasks?
- How are different systems used together?
- What are the actions - what is actually happening?
- Which information is needed at each stage of any process?
- How long does each stage of the process take? (Both own observation and actors opinion)

3. Examination environment

- Make a plan of the examination room and control room.
- Where are the domain objects located?
- Where are the actors located at each stage of the process?
- In what direction are the actors looking at each stage of the process?
- Where is the patient at each stage of the process?
- What can the patient perceive at each stage of the process?
- What kind of background noise is there (causes, level)?

4. Information

- How do the actors have access to relevant information (in particular about the patient, but also about the process)?
- Is sound used to convey information? If yes, for what kind of information?
- How is patient or other information exchanged between different systems (like RIS, HIS, etc)?
- What information is recorded during each stage of the process and how is it recorded? (Particularly interesting would be information on the procedure used and parameter settings.)
- Try to get samples of forms and annotate the meaning of terms on the forms!
- Are computer logs of the various systems available (in particular reflecting the use of functionality)? (Question to ask the application specialist)
- Do such logs also have an actor code on them? Or is there another way to find out who the actor was?
- What kind of information can we get from the help desk/service desk? (Question to ask the application specialist)

5. X-ray acquisition procedures

- For which X-ray acquisition procedures is the equipment used? (Both now, in general in that particular room, and in the hospital.)
- Do all the actors perform all the procedures, or do actors specialise in certain procedures?
- How do actors use the possibility to adjust parameters and sequences manually?
- Are there fixed scenarios (parameter settings) in this hospital for specific circumstances, meaning that default values are changed on a formal basis?

- How much freedom do actors have (within the hospital policy) in adjusting parameters to their own taste or the taste of the image reviewer?
- Do the actors believe there are patterns in the changes of parameters, and what kind of patterns are they thinking of?

6. The way the system can be operated

- What are good features of the way the systems can be operated?
- What would be good things to change in a future product (future meaning in a couple of years)
 - with respect to the way systems can be operated?
 - with respect to the way information is provided by the system at each stage of the process?
- How easy was it to learn to use the systems? What can still be improved?
- How easy is it to use the systems? What can still be improved?
- How efficient is it to use the systems? What can still be improved?
- Are there functions/buttons on the systems that the actors never use? Which? Why don't they use them?
- Would the actors prefer to obtain some information/messages from the systems in audio which are currently on a display? Which messages? Why?
- Would the actors prefer to enter some commands in speech instead of by pressing buttons? Which commands? Why?
- How do the actors think about customisation? Can they see a use for it? What would they like to see customised?
- General comments?

7. Patients

- How do patients experience the various X-ray procedures?
- What information do the actors give them before or during the process?
- What information would the actors like the patient to have, but have difficulty providing?
- What information would the patient like to have but is not getting?
- How regular does it happen that the same patient is examined again (e.g., after a year)?
- Would the parameter settings of the previous X-ray acquisition be useful in such a case?

8. Introduction of new equipment

- What kind of process is used for introducing new equipment in the hospital?
- Who are the main persons involved in an early stage?
- Who are the first people using it?
- Who are influential in a hospital in making new equipment successful?
- How long does it take before new equipment is in full operation?
- What would be good things to change in the process of introducing a future product in the hospital?
- What is the role of Service in the introduction process?
- How does customisation by Service of pre-set system parameters take place?

9. Training

- What kind of training do the radiographers/radiologists receive?
- How long does it take?
- What can be improved in the training process?

10. General

- What are the trends in this hospital (e.g., with respect to digitalisation and use of radiology by other departments)?
- What is the opinion of the radiologists/radiographers on these trends? What potential problems do they see? What opportunities?
- Look for any opportunity for introducing existing/new UI techniques, plus opportunities for customisation and multi-modal interaction!

11. List systems in the hospital (full names and versions) and the software running on these.

(When talking to an operator, always ask what system is referred to.)

Additional questions related to Image viewing and post-processing

During which parts of the examination are images viewed?

When do they use the monitors in the Examination room, when do they use the monitors in the Control room?

What image processing and viewing functionality is used, why, where, when, and by whom? Ask about:

- Zooming (on center of image?)
- Changing image contrast

- Changing image brightness
- Changing polarity
- Masking image edges
- Changing edge enhancement
- Subtraction
- Correction for motion artefacts (pixelshift)
- Adding background anatomical features
- Quantitative analysis
- Annotation of images: left/right, examination data, observations

Are images stored during the viewing process?

Hardcopies versus digital images

- Are hardcopies still used in this digital hospital? If yes, for what?
- What are the main advantages of having digital images compared to hard copies with respect to possibilities for viewing and post-processing?
- What are the main disadvantages of having digital images compared to hard copies?

Ideas for improvement

Mention/request any other ideas.

Acknowledgements

We would like to thank the application specialists who accompanied us in the hospital visits: Erik Vossen, Theo Loohuis, Jacques Lukassen, and Peter Sutton. Thanks are due to the following designers: Horland Hudson for organising the workshop, Job Rutgers and Jude Rattle for their work on the scenarios, and Sue Coles for her work on interaction styles for customisation. Boris de Ruyter was involved in some discussions on the notation for the task model. Paul Gough gave comments on an earlier version of this paper. This project was supported in PMS by Eric von Reth and Franklin Schuling.

References

1. Bés, M-O., Analysis of a human error in a dynamic environment: The case of air traffic control. Proceedings of the workshop on human error and systems development, Glasgow, 1997.
2. Reason, J. Human error. Cambridge: Cambridge University Press, 1990.
3. Sackett, D.L., Rosenberg, W., Gray J.A., Haynes, R.B., & Richardson, W.S. Evidence-based medicine: what it is and

what it isn't. Centre for Evidence-Based Medicine,
<http://cebm.jr2.ox.ac.uk/ebmisisnt.html>,
1996.

4. Lim, K.Y., & Long, J. The MUSE method for usability engineering. Cambridge series on human-computer interaction, 8, Cambridge: Cambridge University Press, 1994.
5. Masthoff, J. An agent based interactive instruction system. PHD thesis. Eindhoven University of Technology, 1997.
6. Rouse, W.B., Geddes, N.D., & Curry, R.E. An architecture for intelligent interfaces: outline of an approach to supporting operators of complex systems. Human-Computer Interaction, 3, 87-122, 1987.
7. Totterdell, P., & Rautenbach, P. Adaptation as a problem of design. In D. Browne, P. Totterdell, & M. Norman (Eds.) Adaptive user interfaces. London: Academic press, 1990.
8. Dietrich, H., Malinowski, U., Kuhme, T., & Schneider-Hufschmidt, M. State of the art in adaptive user interfaces. In M. Schneider-Hufschmidt, T. Kuhme, & U. Malinowski (Eds.), Adaptive user interfaces. Amsterdam, 1993.
9. Dang, W. (1988). Intelligence in a user interface management system. Proceedings of ERGO-IA 1988. CNRS, Université de Paris-Sud, Orsay.
10. Sherman, E.H. A user-adaptable interface to predict user's needs. In M. Schneider-Hufschmidt, T. Kuhme, & U. Malinowski (Eds.), Adaptive user interfaces. Amsterdam, 1993.
11. Browne, D. Conclusions. In D. Browne, P. Totterdell, & M. Norman (Eds.) Adaptive user interfaces. London: Academic press, 1990.
12. Norcio, A.F., & Stanley, J. Adaptive human-computer interfaces. Naval Research Laboratory Report 9148, 1988.
13. Mallen, C. Designing intelligent help for information processing systems.

International Journal of Human-Computer Studies, 45, 349-377, 1996.

Biography

Ashok Gupta, Philips Research Laboratories, Cross Oak Lane, Redhill, RH15HA, UK. Telephone: +44 (1293) 815647, Fax: +44 (1293) 815500. Email: gupta@prl.research.philips.com.

A. Gupta is a Principal Scientist at Philips Research Laboratories where he has been working since 1988. His primary research interests are in user-centred design for professional applications, user-interface process assessment, requirements specifications, and software process improvement. In addition to medical, he has worked in the consumer electronics, and semiconductor domains.

Judith Masthoff, Philips Research Laboratories, Cross Oak Lane, Redhill, RH15HA, UK. Telephone: +44 (1293) 815383, Fax: +44 (1293) 815500. Email: masthoff@prl.research.philips.com.

J. Masthoff has experience in artificial intelligence, situated cognition, interactive instruction, user-centred design and experimental evaluation.

Both A. Gupta and J. Masthoff are working in the Software Engineering and Applications Group, focusing on research into mobile and personal user-interfaces.

Paul Zwart, Philips Medical Systems Nederland B.V., Veenpluis 4-6, 5680 DA Best. Tel: ++31 (0)40-2763024, Fax: ++31 (0)40-2765657. E-mail: P.Zwart@best.ms.philips.com

P.Zwart has been with Philips Medical Systems since 1985. His interests are in image compression, medical image processing and user interfaces to medical systems.