

AN INTRODUCTION TO SCREEN DESIGN AND USER INTERFACE MODELLING

Written by: Robin Beaumont e-mail: robin@organplayers.co.uk

Date last updated: Sunday, 29 January 2012

Version: 4

How this document should be used:

This document has been designed to be suitable for both web based and face-to-face teaching. The text has been made to be as interactive as possible with exercises, Multiple Choice Questions (MCQs) and web based exercises.

If you are using this document as part of a web-based course you are urged to use the online discussion board to discuss the issues raised in this document and share your solutions with other students.

Who this document is aimed at:

This document is aimed at three types of people:

- Those who wish to become involved in systems development but are not interested in the nuts and bolts of programming, such people are commonly called domain experts and act a bridges between a professional group (e.g. medics, Solicitors etc) to which they belong and IT experts.
- For those just beginning professional computer science courses.
- For those who wish to become involved in some form of analysis activity. This might be Process re-engineering or Data flow analysis etc.

I hope you enjoy working through this document.

Robin Beaumont

Acknowledgements

Many people have been involved in the development of this document over the years, including numerous MSc and DMI students over the last 5 years. Many thanks for their astute observations, feedback and generous gifts of additional material which have undoubtedly improved this document significantly. Furthermore the staff at the Informatics Faculty at the RCSed (Tracy Noden and more recently Grainne Comerford) have played a significant roll in it's development.

Contents

1. Before you start	3
1.1 Prerequisites	3
1.2 Required Resources	3
2. Learning Outcomes	4
3. Introduction	5
4. Why Consider the Screen Separately?	6
4.1 Screen Development	7
5. Task Analysis	9
5.1 Sources of Information	10
5.1.1 Value of Flow Diagrams for Communicating With Users	10
6. Why Worry About Defining Dialogues?	12
6.1 Example 1 - Making soup	12
6.2 Example 2 - An electronic Calendar	13
6.3 Designing PEFs	15
6.3.1 Designing Screen Objects From PEFs	18
7. Windows and Controls	18
7.1 Mapping PEFs to Controls	19
7.2 Windows and Data Structures	20
8. Print versus Screen	22
9. Screen Layout Standards	22
9.1 Fonts	24
9.2 Standards for Luminosity/contrast on web pages	26
9.3 Standardisation of Controls	26
10. Low Level Dialogue Design	27
10.1 Escapes and Help	27
11. Interactive Iterative Prototyping - Collecting Information from the User to Improve the Design	28
12. Usability	28
13. Revision Exercise	29
14. Summary	29
15. Further reading	29
16. Multiple Choice Questions	30
17. References	33
18. Links	35
19. Exercise answers	35

1. Before you start

1.1 Prerequisites

This document assumes that you have the following knowledge and skills:

Skills:

- That you have used the following features of a Database Management System (DBMS) such as Access or Paradox:
- Created Tables
- Created Relationships (and therefore known about the relationship window)
- Created simple Queries in the query design window
- Created a simple form

Knowledge:

- You should also be able to describe what the following concepts mean:
- Tables, indexes and Fields
- Relationships (not a detailed description)
- Forms
- Queries

If you have done the ECDL (European Computer Driving Licence) you will have covered these topics. If not I recommend that you do so now. You can take the ECDL either at a local college (in the UK) or as a distance Learning course. There are also some very good books guiding you through the ECDL. Alternatively a basic Access course can be found at <http://www.robin-beaumont.co.uk/virtualclassroom/contents.htm>

1.2 Required Resources

You need the following resources to work through this document:

The “Scenarios for practicing modelling techniques” handout available

from <http://www.robin-beaumont.co.uk/virtualclassroom/contents.html> and follow the links in section 11.

2. Learning Outcomes

This subsection aims to provide you with the following skills and information. After you have completed it you should come back to these points ticking off those you feel happy with.

Learning outcome	Tick box
Be aware of the range of topics covered in the HCI discipline	<input type="checkbox"/>
Be aware that the presentation and application can be two distinct software modules	<input type="checkbox"/>
Be aware of the four main stages of screen specification as outlined by Hutt & Flower 1990	<input type="checkbox"/>
Be aware of the STUDIO method for specifying screen design	<input type="checkbox"/>
Be aware of the recommendation for flat rather than deep hierarchies in navigation paths and menus	<input type="checkbox"/>
Be aware of screen prototyping as a method of developing screen designs	<input type="checkbox"/>
Be aware of the use of informal 'flow charting' techniques to obtain initial information about tasks from users	<input type="checkbox"/>
Be aware of the need to formalise informal flow charts at the end of the task analysis	<input type="checkbox"/>
Be able to draw simplified state diagrams indicating the navigation route through a set of windows	<input type="checkbox"/>
Be aware of the problems with using simplified state diagrams to fully describe the navigation paths	<input type="checkbox"/>
Be aware of the importance of the optional display of unimportant information as well as consistency in providing a friendly interface	<input type="checkbox"/>
Be aware of the clinical advantages and disadvantages of using PEFs	<input type="checkbox"/>
Be aware of the importance of mimicking the natural consultation flow when developing PEFs (Patient Encounter Forms)	<input type="checkbox"/>
Be aware of the subtlety of positioning and formatting enhancements to aid use of PEFs	<input type="checkbox"/>
Be aware of the layout differences between PEFs and screen implementations	<input type="checkbox"/>
Be aware of some of the factors that come into play when designing screens for PEFs	<input type="checkbox"/>
Be able to describe the difference between various windows controls	<input type="checkbox"/>
Be able to recognise modal and modeless windows	<input type="checkbox"/>
Be able to choose the appropriate windows control for an item within a PEF	<input type="checkbox"/>
Be aware that the successful screen presentation of databases usually requires the use of queries and filters	<input type="checkbox"/>
Be aware that the appropriate layout of fields in a database depends upon the intended use of the screen (ie data entry, viewing, analysing etc)	<input type="checkbox"/>
Be aware of some of the fundamental differences between screen based and printed media	<input type="checkbox"/>
Be able to name an international standard concerned with the presentation of visual information on a computer screen	<input type="checkbox"/>
Be able to list two de facto standards concerned with screen design	<input type="checkbox"/>
Be aware of the experimental research that takes place regarding screen preferences	<input type="checkbox"/>
Be aware that de facto standards provide much information concerning detailed screen layout guidelines	<input type="checkbox"/>
Be aware of the online screen colour contrast testing tools	<input type="checkbox"/>
Be able to interpret state diagrams describing low level user dialogues	<input type="checkbox"/>
Be able to draw state diagrams describing low level user dialogues	<input type="checkbox"/>
Be aware of the problems associated with using state diagrams to specify low level user dialogues	<input type="checkbox"/>
Be aware of some alternative approaches to specifying the low level user dialogue	<input type="checkbox"/>
Be aware of the term usability	<input type="checkbox"/>

3. Introduction

This document looks at the methods used to specify what is displayed on computer screens along with the navigation paths provided to users. This document requires you to carry out a number of exercises some with pen and paper others requiring access to the Internet

While the bulk of this document provides you with a description of some of the common methods used to facilitate the development of screen layouts for computer programmes, We will first consider the broad area in which this topic sits before diving into this fairly low level topic.

The process of designing screens is just one aspect of an area of computing science called HCI (Human Computer Interaction). "Put simply HCI is the study of people, computer technology and the ways these influence each other" (Dix & Finlay et al 1993, p xiii). The term became popular in the 1980s along with 'Industrial engineering', 'human factors' and 'ergonomics' all of which have a similar area of concern. While we will not be looking at all of the aspects of HCI in this module, it is useful to be aware of the topics covered in the HCI discipline. To achieve this, please carry out the task below.

We will investigate screens at two levels:

Specifying possible navigation paths

Specifying individual screens

Why, you may be thinking, should you consider the screen design separately from the data aspects? Surely if we have a good ERD or object model -- in other words, the data requirements -- that should be enough to derive what the screen should look like?

Exercise 1. Topic areas of HCI

You should take no longer than 45 minutes for this exercise

1. Write down a list of no more than ten headings which you think would cover the various aspects of HCI (10 minutes).

2. General introduction to HCI

Read: <http://hamilton.bell.ac.uk/btech/hci/hciintro.pdf>

Saul Greenberg, professor in Human-Computer Interaction & Computer Supported Cooperative Work at the University of Calgary has produced a good set of resources for teaching HCI topics. Specifically the concept of Affordance, Norman, D.A. (1988), see http://pages.cpsc.ucalgary.ca/~saul/hci_topics/pdf_files/psychology-academic.pdf (warning this takes time to download). Saul's site also has some nice examples of bad designs, see http://pages.cpsc.ucalgary.ca/~saul/hci_topics/bad_interfaces/bad_interfaces.html

For another explanation of affordance along with that of how he defines visibility see John McSweeneys computing page at: <http://www.jmcsweeney.freeserve.co.uk/computing/m206/hci1.htm>.

In a previous version of this document I suggested the following http://www.dcs.napier.ac.uk/~michael/hci_ohps/hci41n.html. This was a lecture by Michael Smyth at Napier University in Edinburgh (<http://www.dcs.napier.ac.uk/~michael/home.html>). Unfortunately it has now been removed, shame it was an excellent introduction.

From the above sites compare what they believe are the components of HCI; with your list.

3. Visit <http://www.hcibook.com/e3/chapters/intro/> This is the list of contents for the Dix & Finlay book. Compare your list with the chapter contents on this website; how does it match up? Incidentally we will be covering some of the material in chapters 7 and 8, task analysis and dialog notations and design, in this subsection.

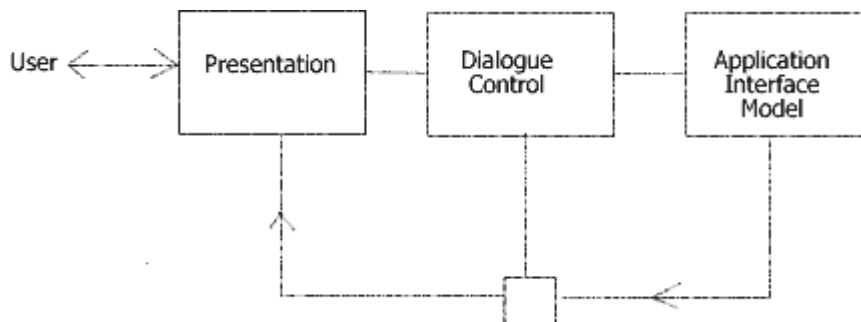
4. Visit <http://hfes.org/Publications/TOC/hcicontents.html>, a book listing the 79 best papers presented at HFES (Human Factors and Ergonomics Society) meetings in the last fifteen years from a selection of more than 3500 papers.

4. Why Consider the Screen Separately?

You may feel that by now you have spent long enough on design issues, and surely there is no need to spin it out any further. If anything, the design of the screens and the possible navigation routes are dictated by the functions and data that you have already specified?

This is not the case. At a very superficial level, think how very different a Dos and Windows interface are. For example you can perform exactly the same function by typing "copy A:mydoc.txt C:/somefolder/copy.txt" and pointing and clicking in Windows, yet the method and appearance are very different.

Once users began to interact directly with computers, rather than via punch card operators (notice Micheal Symth's picture of a sketch of a computer circa 1960 in the task on the previous page), it became necessary to think about developing a method of communicating between them. Some of you may remember the old **Teletext terminals** where you both typed in the instructions and received feedback on a continuous paper roll. Clearly such a method could not allow the Windows style interface that we are all now so familiar with. Such paper based systems demanded a '**command based**' interaction in contrast to the **iconic** method now in vogue.



The idea that the interface to the user and the nuts and bolts of the system (functions and data), in terms of software design, should be separated was formulated in the 1980s. Because the idea was developed during a series of workshops in Seeheim, Germany in 1983 (Edmonds 1990 p59), the model has subsequently been called the Seeheim model.

Here is a brief description of the three logical components to the software architecture.

Presentation is the component responsible for the appearance of the interface, including what output and input are available to the user.

Dialogue control is the component which regulates the communication between the presentation and the application.

Application interface is the component which translates the dialogue semantics (ie meaning) into that of the application (eg in a database application clicking on the next button [presentation/dialogue] moves the pointer on one record [application]).

It must be remembered here that the context is one of software development, and the main issue is how to divide it up.

Basically you can think of the application as the back end of the software. For example, the back end of a database would be the tables and queries. The presentation aspect in this instance would be the forms, reports and menu structures that you could design to allow various users to view and manipulation the data. The dialogue control component (modified by Dance et al, referenced in Edmonds 1990 p60) is concerned with providing semantic support between the application and the presentation components. For example, consider a screen that is displaying three things: a window, an icon for a file and an icon for a disc drive. This is the 'presentation'. Now consider the differences in the two following situations:

A user drags the window over the disk icon; **result:** disk icon obscured

A user drags the file icon over the disk icon; **result:** icon highlighted

The interface therefore needs to have a certain degree of intelligence about the various objects in the system, hence the development of 'intelligent graphical interfaces'.

There is a large amount written about this important topic for programmers (now called software engineers). I advise you to try Edmonds 1981, 1990 for starters.

4.1 Screen Development

Hutt & Flower 1990 describe a four stage process for developing user interfaces (for the present, assume this to mean screen specification):

Consider the 'overall appearance' required.

Consider the 'dialogue control' structure.

Design visual images.

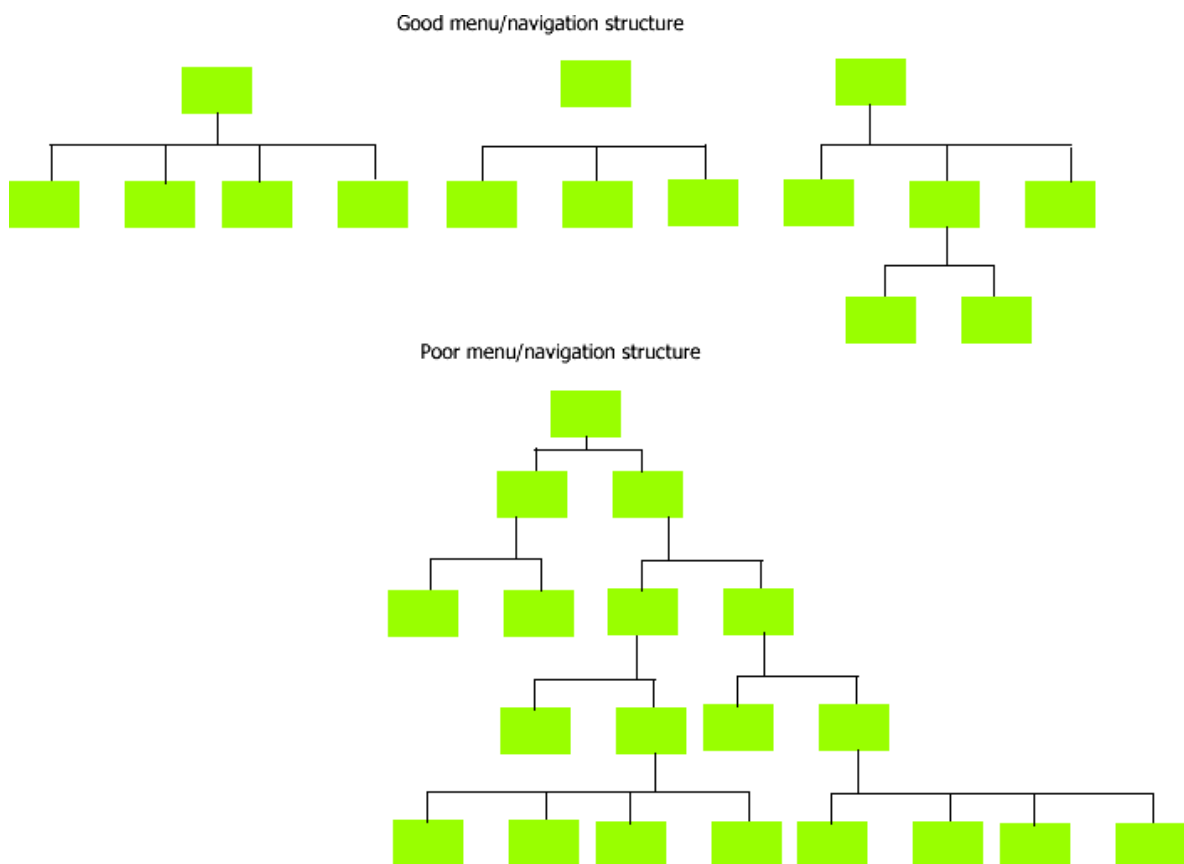
Design low level dialogue.

Overall appearance involves selecting a design metaphor which the users can relate to (eg the Windows desktop or an old command driven interface such as Dos). Alongside this, the feel of the screen should be considered. For example, is it to be formal or friendly? What about the use of colour schemes? Input and output methods such as the provision of keyboard shortcuts and deciding if textual alternatives to graphics for the visually impaired will be provided is important. Nowadays the universal adoption of Windows, the Web and cascading stylesheets makes this frequently an academic exercise. For an interesting example of a different metaphor to Microsoft Windows, see the WordPerfect (correct spelling) website: http://www.e-2.org/word_perfect.html.

Dialogue control takes into account the range of prospective users (novice to expert), help facilities, flow of control and navigation techniques (menus, etc) and paths (tree/linear/user defined etc). Within the Windows style interface, we are basically thinking of menu options and window sequencing/positioning etc.

For example, think of the difference between using Microsoft Access to develop a query in design view, basically a user defined dialogue structure, to that of using a Wizard which has a clearly defined path. Much of the information for this stage is derived from a **task analysis** (more about this on the next page).

I personally feel that the navigational freedom of the Web is frequently seen as a bad thing by novice users who prefer 'portals' that offer a clearly defined navigational structure. The ability to allow users to create a favourites tree structure does something to alleviate this problem, but it does nothing to provide a method of allowing users to specify a navigation path. Most writers recommend that navigation paths should be flat and broad rather than highly nested:

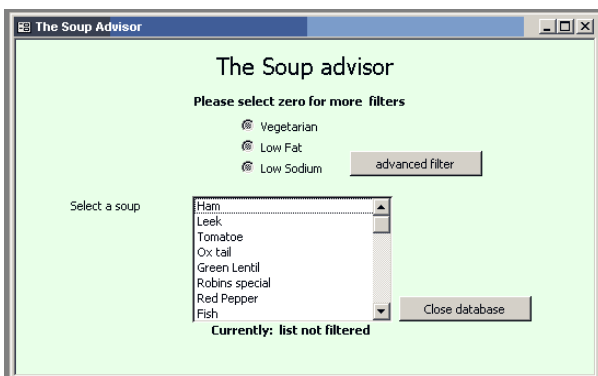


The above diagrams can also be represented as a textual tree indicating the actual options. For example, taking the familiar Word for Windows menu (which has become a de facto standard) we can create the following tree:

File	Edit	View	Insert	Format	Tools	Window	Help
- New	- etc	- etc	- etc	- etc	- etc	- etc	- etc
- Open							
- Close							
- Save							
- etc							
- etc							
- Exit							

Screen design documents often contain a table similar to the one presented above specifying a proposed menu structure.

Design visual images In terms of the windows environment this is the design of the individual 'windows' for the application. Nowadays often Access or PowerPoint are used to create a set of mock windows (a '**screen prototype**'). The following was produced in approximately 5 minutes in Access 2000. This provides users with a real feel for the system, even if no actual data structure exists behind these mock windows. They are also very useful to provide immediate feedback concerning the level of disruption that may occur in everyday tasks like introducing a computer into the medical consultation:



Design low level dialogue This is concerned with specifying the interaction at the individual window level, such as tabbing sequences and options that are dependent upon other values set within a window.

The above method is obviously not the only one available. Nearly every book on user-interface development proposes one, and they vary widely in perspective (see Dix, Finlay & Abowd et al 1993; Hix & Hartson 1993). Nielson 1993 p224 provides a list of various techniques that can be used to assess the **usability** of interfaces, including his own Heuristic evaluation method which does not involve users! (You can find details of this on his website listed at the end

of this subsection.)

As would be expected from a KPMG management consultant, Dermot Browne (Browne 1994) offers a method that stresses management aspects. The details of his STUDIO (STRUCTURED User-interface Design for Interaction Optimisation) method (Browne 1994) are given below. While I find his method attractive, I find it saddening that he fails to provide references for the various ideas he has obviously borrowed from other writers.

STUDIO consists of five stages:

Project Proposal and Planning The most important part of this stage is to consider if user interface design expenditure is appropriate for the project.

User Requirements Analysis This bears much in common with the traditional systems analysis stage of software development. Importantly, within it a 'task analysis' is carried out.

Task Synthesis "This stage takes the results of the user requirements analysis and delivers an initial user interface design. There are five steps in this stage, some of which can be performed in parallel. The step entitled 'User support' generates the necessary documentation such as user manuals that will form a part of the user interface. STUDIO offers much guidance with respect to providing effective documentation. The 'style guide' step delivers a document stipulating rules that must be adhered to by the user interface design. In essence, this step ensures that a consistent user interface is developed. 'Task synthesis' is the step during which designers begin to generate user interface designs. Interface specification notations are introduced at the 'Design specification' step. These notations permit designers to perform 'Formative evaluations' before prototyping is performed. In this way, potentially expensive design inadequacies can be identified and rectified in a cost effective manner. User involvement during this stage is largely in the role of reviewer." (Browne 1994, p9)

Usability Engineering This stage "combines prototyping and impact analysis to provide a manageable approach to iterative prototyping. This stage takes the design(s) produced in Stage 3 and submits them to evaluation through prototype testing. This begins with a step for 'usability engineering planning', which includes the time-tabling of user involvement and production schedules for evaluation materials. 'Prototype build' is then undertaken. Prior to user testing of any prototype a 'Design Audit' is undertaken. This audit identifies flaws in the design through the application of simple diagnostic tests. In parallel with building the prototype, the evaluators 'Prepare evaluation materials'. Following the successful completion of these steps 'Prototype evaluation' takes place. The results of this evaluation are submitted to 'Impact Analysis', which prioritises the desirable modifications to the prototype allowing valuable development resources to be deployed to best effect. Prototype evaluation and impact analysis may be repeated. Having concluded all evaluations and prototype modifications this stage is completed by 'Update documentation'. Usability Engineering (stage 4) delivers a usable prototype and supporting specifications." (Browne 1994, p9)

User Interface Development This stage consists of four steps. The 'hand over specification' step is concerned with ensuring that the deliverables from the previous stage when handed over to the system developers are properly understood. Once a dialogue with the system developers starts inevitably 'integration/interfacing' issues are brought up which forms the next stage. The third step is 'acceptance testing' where the user interface designers have a role in validating and verifying the final user interface. The final step is 'termination reporting' which provides a chance to reflect upon the experience including the planned and actual effort expended on the project.

We will now look in more depth at the Hutt & Flower method, stage 2 dialogue control specifically, by looking at task analysis.

Exercise 2. Menu Structures

You should take no longer than 30 minutes for this exercise.

For the DopeHead scenario consider:

What might be the main menu options available to a technician? Start by looking at the menu options of some Windows programmes.

5. Task Analysis

"Task analysis is the process of analysing the way people perform their jobs: the things they do, the things they act on and the things they need to know." (Dix & Finlay et al 1993 p221).

We have actually already investigated a method of task analysis in the dynamic modelling subsection. In that subsection we took for granted the actual sources of information we might use to produce the model. But it is now time to consider how you might actually obtain the information in more detail.

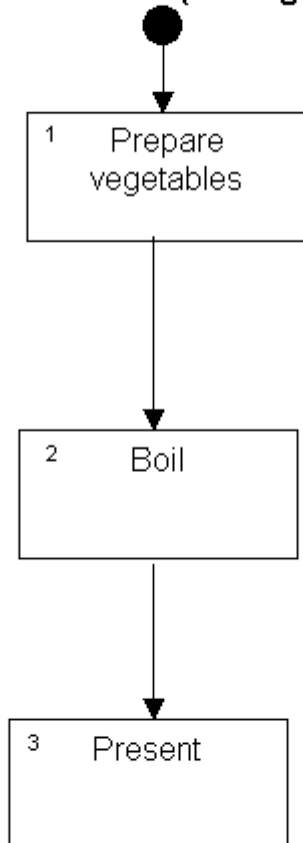
5.1 Sources of Information

One of the most common methods of gleaning information about tasks is to inspect existing documentation. This can be supplemented with observation and interviews as well as using CCTV footage that is coincidentally collected. This last technique is becoming increasingly popular.

If the task involves paper documents, one particular method which is usually very effective is to pin each of the documents on a wall and then stick small notes beside each one. You can use such things as different coloured tape to indicate importance etc.

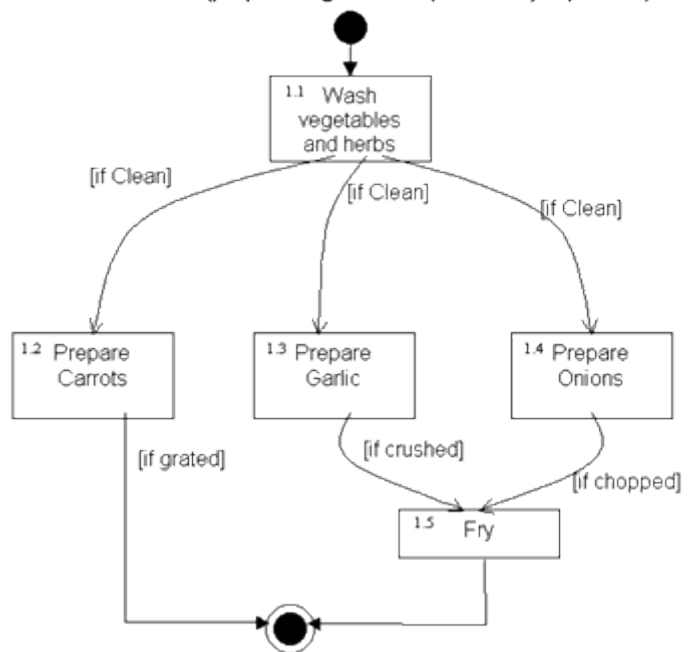
5.1.1 Value of Flow Diagrams for Communicating With Users

Top level flow chart (Making soup)



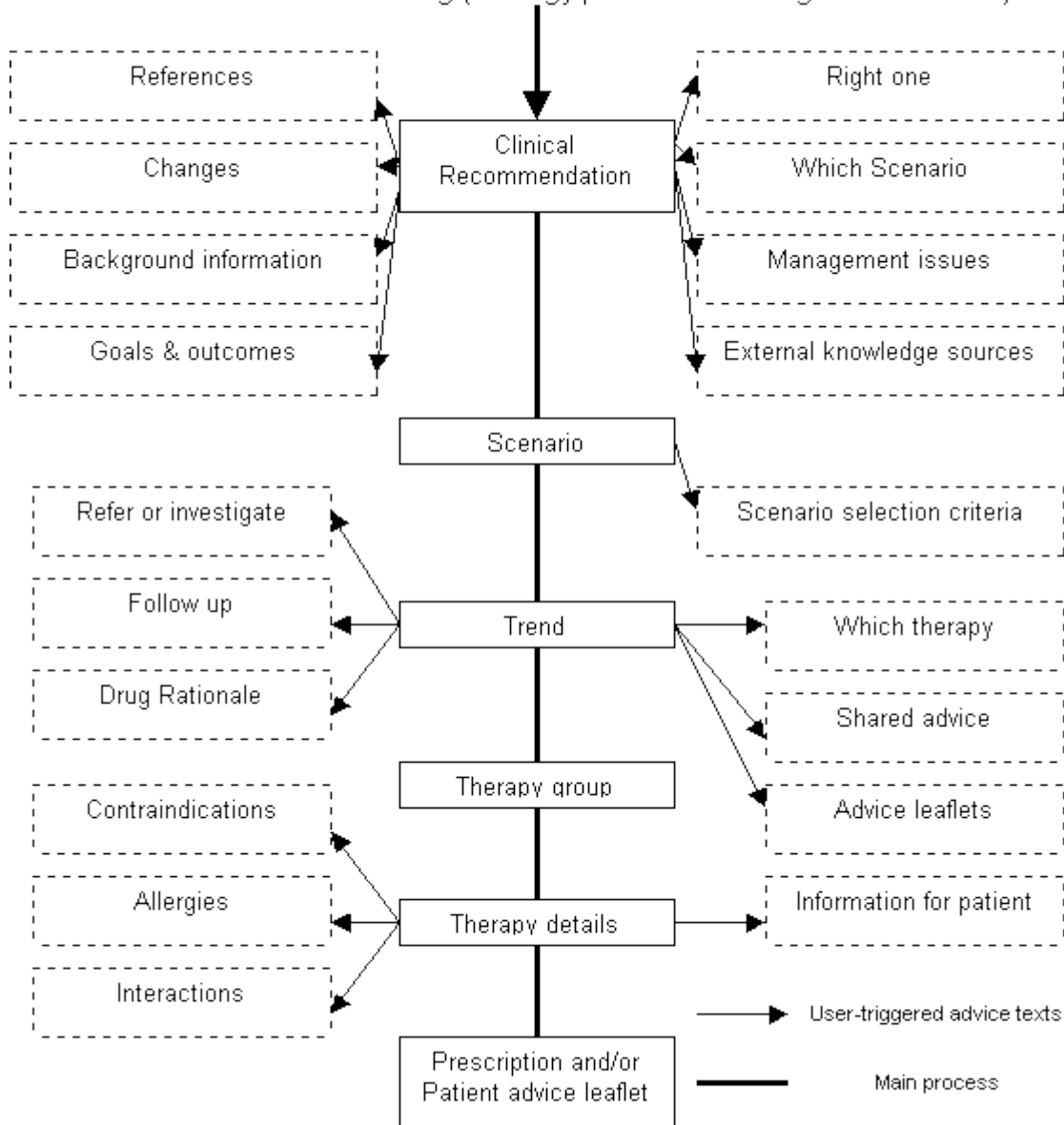
Extracting information from users about their tasks can be difficult. One simple technique is to get them to draw a flow chart of some type to clarify the process. For example the diagrams below describe the process of making soup. Notice the use of expanding the process 'prepare vegetables' in the top level flow chart into a separate chart. Note also that each process is given a number which also relates to its level within the hierarchy.

Flow chart (prepare vegetables – process 1) expanded



When working initially with users to gain insight, the formalities of diagramming semantics such as UML can be forgotten to a large extent. For example, below is a process diagram explaining how a prescribing module ('Prodigy') within a GP computer system might be used in a typical scenario. Do not worry about understanding the various terms such as therapy group. The important thing to notice is the deliberate lack of rigor at this stage:

Process model for Prescribing (Prodigy phase 3 Bob Sugden et al 1998)



By the end of a task analysis, the 'informal' diagrams should be re-designed using the rigor of a standard diagramming technique such as that offered by UML.

Following on from a task analysis, we are now ready to develop a dialogue control structure for the software.

Exercise 3. Navigation paths

You should not spend more than 60 minutes on this exercise.

For the DopeHead scenario consider:

What might be a typical navigation route through the application for a technician in terms of performing the following tasks:

Answering a request from an athletics association

Entering results for a test

It may help to think in terms of screens or windows within a screen. Do not worry about all the possible navigation routes, only a typical scenario.

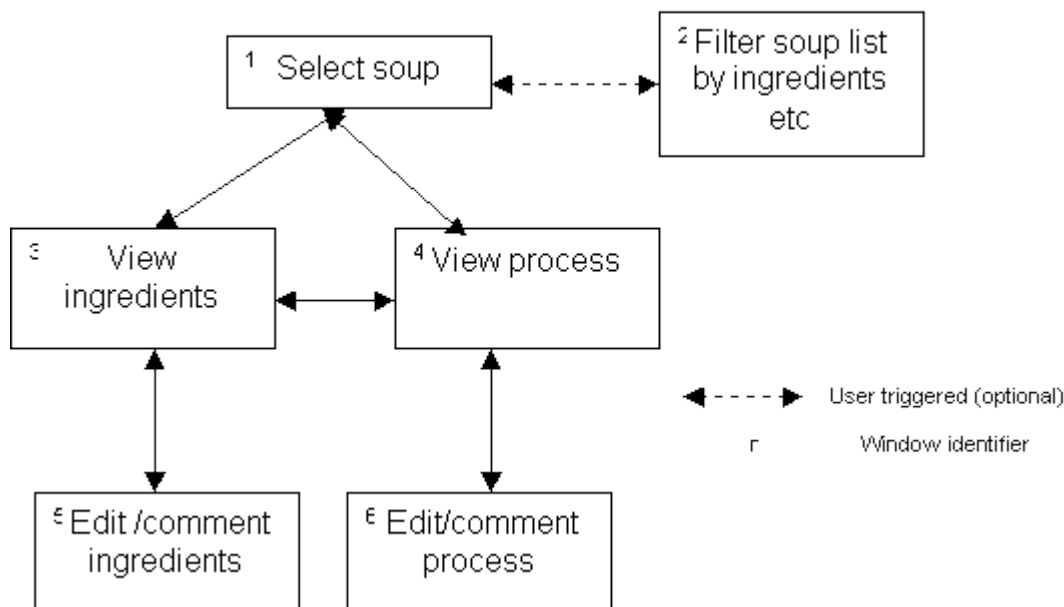
6. Why Worry About Defining Dialogues?

The Prodigy Project was concerned with developing a simple prescribing module for several GP computer systems in the UK. One of our findings was that GPs' satisfaction with the system corresponded to the relative length of time it took them to carry out a task manually compared to obtaining help from the computer. Therefore if you produce a dialogue which makes the task longer or more difficult than it took previously, it is unlikely to succeed.

Dix et al 1993 mentions the fact that dialogue definition is also concerned with presentation besides just the navigation. The main reason for this is related to cognitive processing. A user might actually be carrying out a task more quickly than previously without the aid of a computer, but if the amount of information presented during it is far greater than before then she/he may feel as if the task took longer. The **Optional display of unimportant information** is extremely important as is **consistency**. I wonder how many people like the Windows 2000 adaptive interface which only presents you with the options you most recently have chosen?

6.1 Example 1 - Making soup

Dialogue in Soup making software



The above diagram illustrates some of the advantages and disadvantages of using this type of diagram (basically simplified STD):

Each window is given a unique identifier (a number in this case).

Each box presents a window.

There is not an easy way to indicate that one or more windows may be displayed. For example it may be desirable to indicate that both windows 3 and 4 may be viewed side by side. The above diagram would imply that a single window is viewed at a time.

One solution to the multiple window problem is to add another window identifier (both graphical and textual) which groups a number of windows. For example a dotted line around the windows to be grouped along with a grouping identifier might help.

6.2 Example 2 - An electronic Calendar

The following example is taken from Hix & Hartson 1993 concerning the development of an electronic calendar. The first picture shows one of several possible views of the calendar (notice the multiple windows) while the second shows the possible navigation between them:

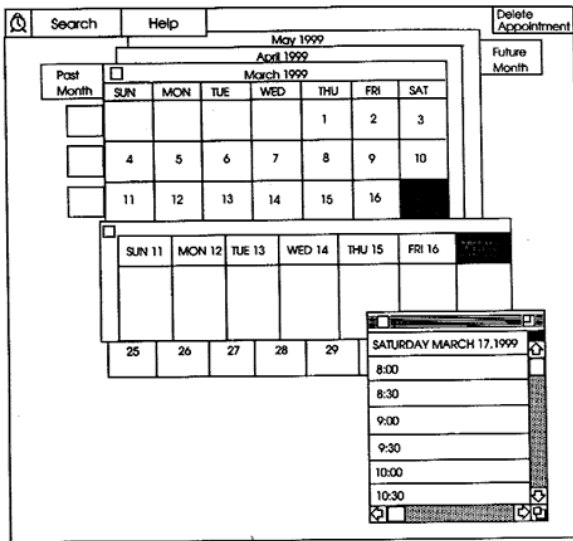


Figure 5.3. A sample screen picture for the revised design to the Calendar Management System. From Hartson & Gray, *Human-Computer Interaction*, pp. 1-45. © 1992. Reprinted by permission.

VIEW LEVEL NAVIGATION

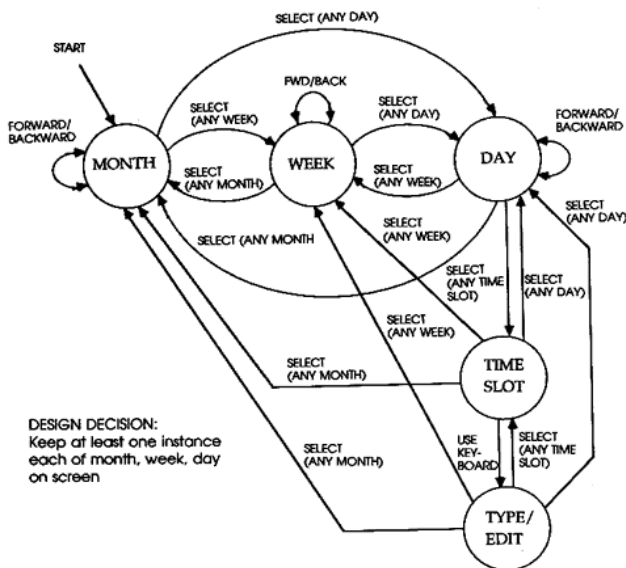
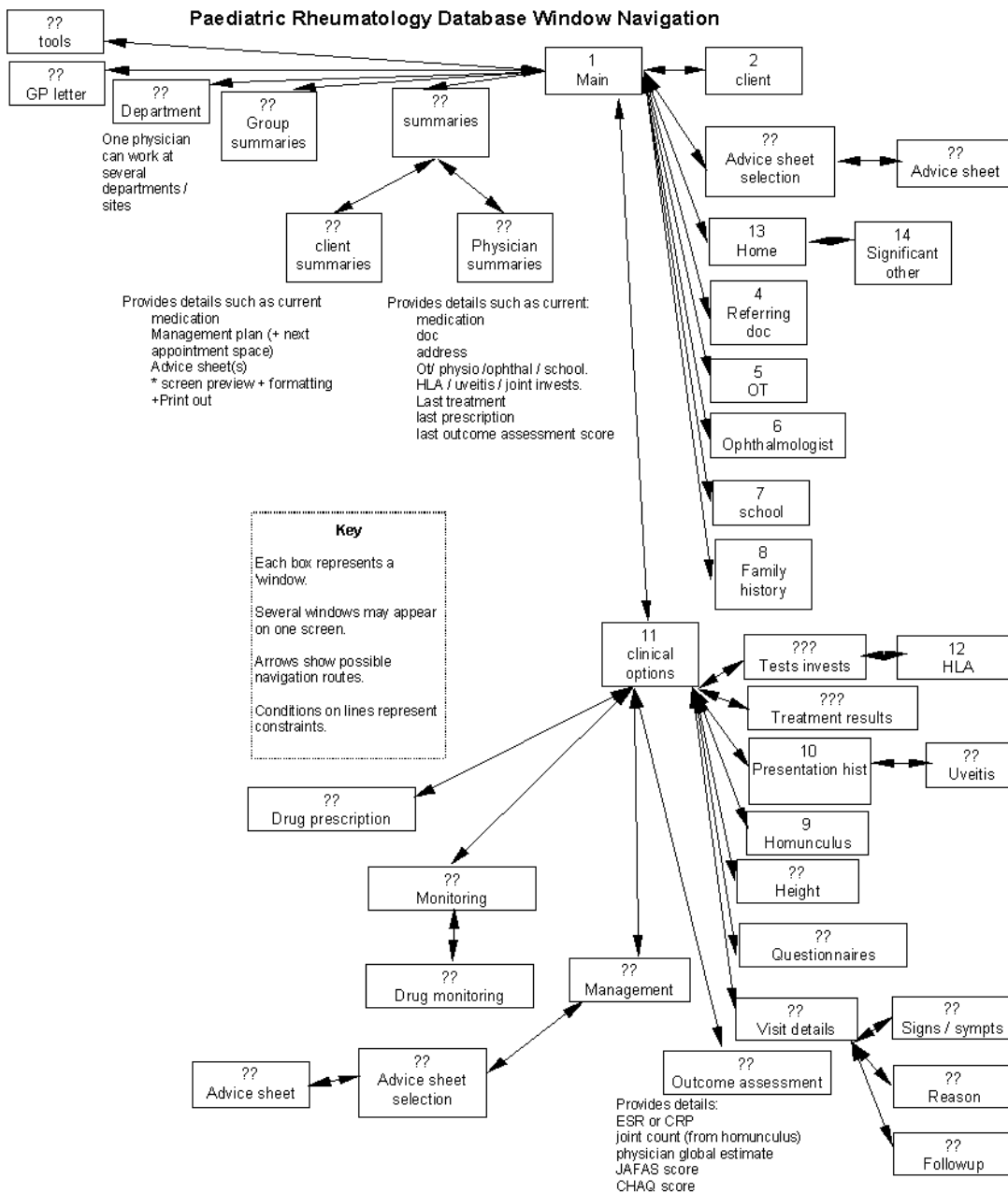


Figure 7.3. State transitions for navigating among views of the Calendar Management System. Adapted from Hartson & Gray, *Human-Computer Interaction*, pp. 1-45. © 1992. Reprinted by permission.

The picture on the following page presents a first attempt at defining a navigation path through a proposed paediatric rheumatology database. Take note of all the comments and queries; obviously the final version will be very different from this initial design.



As with most databases the person who originally requested a 'database' began by presenting a few line drawings of possible 'screens' they felt to be appropriate. As is usually the case, they presented a summary report like screens rather than some type of data collection screens. To those of you who know all about databases, this may seem rather strange, but in actual fact it is a very sensible way of approaching systems design. By describing the required output you can then begin to think of what effort is required to achieve it! As you can imagine, the proposed user had rather a shock when she realised what effort was required to collect the data needed for her reports.

The lack of awareness concerning data collection is a common problem. A common work around (I wouldn't call it a solution) is to provide paper data collection forms (frequently called Patient Encounter Forms **PEFs**) which the person fills in at the time and then passes on to an administrator. In the past, and not so infrequently nowadays, these PEFs form a basis for a computerised (back office) system.

We have now just about finished talking about stage two of the four specified by Hutt & Flower 1990 for developing user interfaces (for the present think of screen specification). At the end of this stage you should have clearly documented navigation paths and menu structures.

Hutt & Flower 1990's four stages are given below to jog your memory:

- Consider the 'overall appearance' required
- Consider the 'dialogue control' structure
- Design the visual images
- Design a low level dialogue

By way of introducing you to stage three of the process, we will look at the relationship between paper based Patient Encounter Forms (PEFs) and the computer screen equivalent.

Exercise 4. Dialogue specification

You should spend no more than 60 minutes on this exercise

Go to: <http://www.hiraeth.com/alan/tutorials/prototyping/prototyping.pdf> and read through to the pages concerned with dialogue specification.

Carry out any necessary updating to the "Navigation route" task you did on the previous page.

6.3 Designing PEFs

Patient Encounter Forms are frequently used to enforce some degree of standardisation in practice. Personally I have found that physicians who introduce them also find them useful for speeding up the consultation process. While this is possibly a mixed blessing, there are some clear disadvantages to using PEFs rather than entering the data at the **point of service**. Bemmel & Musen 1997 (p104) provides the following reasons why computer data entry is better than traditional paper based records:

- The paper record can be only at one place at a time; it may be unavailable or even missing.
- The paper record contents are in free text; hence they are:
- Variable in order
- Possibly illegible
- Possibly incomplete
- Possibly ambiguous
- For scientific analysis, the paper record contents need to be transcribed, with potential errors.
- Paper based notes cannot give rise to active reminders, warnings or advice.

I feel that PEFs fall somewhere between the point of service approach and the traditional medical record in terms of advantages and disadvantages. They certainly improve the content by structuring the data collection, but they increase administrative overhead. The desire to reduce this is often seen as a good reason for introducing computers. For example this has been of particular concern in the probation service in the UK where the introduction of an electronic case management system (CRAMS) was seen primarily as a way of reducing these costs, without taking into account the effects upon those who would be entering the data. One result was that unions presented data that demonstrated that workers would be increasingly stressed due to the particular implementation, hence the project has been 'realigned'.

The real question is: are electronic patient record systems good enough yet to be used for point of service data collection in the consultation? I leave it up to you to decide. Now let's get back to the more mundane aspect of looking at PEFs.

Please look at the following two PEFs (see next two pages), both designed for collecting information from a clinic for pregnant diabetics.

Notice the following:

Use of sub-headings

Multiple entry boxes for some items

Tips for collecting data

Positioning of data on the page:

General: You may not realise it, but the presentation of the information on the form mimics the way most of the physicians naturally collect the data during the consultation within the particular department where the forms were developed.

Outer edge for test results for the review form. The test results are placed on the outer edge so that clinicians can easily browse through the forms and compare results.

Exercise 5. Encounter forms

You should not spend more than 60 minutes on the exercise.

Consider the DopeHead scenario. Develop a form for toxicologists to use to record samples taken within the context of the scenario given.

You may use the drawing tools in Word to create boxes, tables and lines to divide up the page.

Booking form for RVI Pregnancy Diabetes Database

To be filled in after entering data into computer. Important: Always check first to see if patient is on database.

Computer id (=clientid)

Names in brackets '()' refer usually to database field names.

Pregnancy id (=pregid)

First visit

First name	<input type="text"/>	Surname	<input type="text"/>
Occupation	<input type="text"/>	Hospital number (ID)	<input type="text"/>
DOB	<input type="text"/>	Source (ring one)	NDC=1 Other DC=2 IUT=3 Other =4 Not known=5
Pre-pregnancy Diabetes Consultant	<input type="text"/>		
Diabetes Consultant (during pregnancy)	<input type="text"/>		
Consultant Obstetrician	<input type="text"/>	Date seen	<input type="text"/>

Obstetric Data

Previous pregs (number)	<input type="text"/>	LMP (date)	<input type="text"/>
Previous live births (number)	<input type="text"/>	Gestation at booking (weeks)	<input type="text"/>
Stillborns post 24 weeks (number)	<input type="text"/>	Estimated Deliv. Date	<input type="text"/>
Mis carriages (number)	<input type="text"/>	Estimated Deliv. Date by scan	<input type="text"/>

Infertility treatment (ring one) Yes No

Fetus:	Gestation at Fetal echo <input type="text"/>	Gestation at anomaly Scan (weeks) <input type="text"/>
1	Fetal heart at fetal echo (ring one) Normal=1 Abnormal=2	Structure (ring one) Normal=1 Abnormal=2 Type of abnormality (ring one) Cardiac=1 GI=2 CNS=3 Skeletal=4 Other=5
2	Fetal heart at fetal echo (ring one) Normal=1 Abnormal=2	Structure (ring one) Normal=1 Abnormal=2 Type of abnormality (ring one) Cardiac=1 GI=2 CNS=3 Skeletal=4 Other=5
3	Fetal heart at fetal echo (ring one) Normal=1 Abnormal=2	Structure (ring one) Normal=1 Abnormal=2 Type of abnormality (ring one) Cardiac=1 GI=2 CNS=3 Skeletal=4 Other=5

Diabetes Data

DATE of onset of Diabetes (Year if exact date not known)

Type of Diabetes Mellitus (ring one) Type 1 pre-pregnancy=1; Type 2 pre-pregnancy=2; Diabetes diagnosed during pregnancy=3

Eye complications (ring one) None=1; Early NPDR =2; Mod NPDR=3; Severe NPDR=4; Proliferative=5; Maculopathy=6; Other=7

Renal problems (ring one) None=1; Microalbuminuria=2; Proteinuria=3; Abnormal creatinine=4; Other=5

Other medical problems (ring one) None=1; Cardiac=2; Respiratory=3; CNS=4; GI=5; Multiple=6; Other=7

Contraception (ring one) None=1 COCP =2 POP=3 IUCD=4 Barrier=5 Depo=6 Other=7 **Cigs per day (number)**

Pre-Pregnancy Data

Data set completed for pregnancy (completed) To be filled in at end of visits following delivery

Yes No

Other (optional): Date of observation: <input type="text"/> <hr/> Date of observation: <input type="text"/> <hr/> Date of observation: <input type="text"/>	<p style="text-align: center;">HbA1c <input type="text"/></p> <p>Date of observation: <input type="text"/></p> <p>Pre-preg. advice given Yes No</p> <hr/> Systolic BP <input type="text"/> Diastolic BP <input type="text"/> Date of observation: <input type="text"/>	<p style="text-align: center;">Insulin</p> <p>name: <input type="text"/></p> <p>(Units) short Intermediate (imt)</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">am</td> <td style="width: 33%;"><input type="text"/></td> <td style="width: 33%;"><input type="text"/></td> </tr> <tr> <td>noon</td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> <tr> <td>pm</td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> <tr> <td>bed</td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </table> <p>Date of observation: <input type="text"/></p>	am	<input type="text"/>	<input type="text"/>	noon	<input type="text"/>	<input type="text"/>	pm	<input type="text"/>	<input type="text"/>	bed	<input type="text"/>	<input type="text"/>
am	<input type="text"/>	<input type="text"/>												
noon	<input type="text"/>	<input type="text"/>												
pm	<input type="text"/>	<input type="text"/>												
bed	<input type="text"/>	<input type="text"/>												

Comments

Diabetes Pregnancy Database - Regular Review Form

To be filled in after entering data into computer

First name
Surname
Date
Gestation (Week)

Computer patient id (clientID)

pregnancy id

Visit id

Obstetric Data

.....

.....

.....

.....

.....

Total Insulin (paper records only)

Seen by

Weight (Kg)	<input style="width: 80px;" type="text"/>
Systolic BP	<input style="width: 80px;" type="text"/>
Diastolic BP	<input style="width: 80px;" type="text"/>
Urine Glucose	<input style="width: 80px;" type="text"/>
Urine Protein	<input style="width: 80px;" type="text"/>
Creatinine clearance	<input style="width: 80px;" type="text"/>
24hr Protein (24hrp)	<input style="width: 80px;" type="text"/>
Hb	<input style="width: 80px;" type="text"/>
Triglyceride	<input style="width: 80px;" type="text"/>

Fetal measurements At 28, 32, 34, 36 and 37 weeks

Fetus:	BPD	HC	AC	FL	EFW	AFI	PI
1	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
2	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
3	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>

Insulin id

Diabetes Data

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Seen by

Home Blood Glucose Monitoring (HBGM)

	Low	High
Before:		
Breakfast (BB)		
Lunch (BL)		
Dinner (BD)		
Supper (BS)		

Hypos. (number in last month)

	am	ash	eve	bed-3	3-bit
3rdparty	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>
other	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>	<input style="width: 20px;" type="text"/>

Current Insulin for computer record

name:

	short	Intermediate (irr)
am	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
noon	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
pm	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
bed	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>

Change to for paper record only

	short	Intermediate (irr)
am	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
noon	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
pm	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>
bed	<input style="width: 60px;" type="text"/>	<input style="width: 60px;" type="text"/>

Eye complications (ring one)

Booking +20wks +as indicated

None=1; Early NPDR =2; Other=7
 Mod NPDR=3; Severe NPDR=4;
 Proliferative=5; Maculopathy=6;

HbA1c

6.3.1 Designing Screen Objects From PEFs

On the previous pages we looked at several PEFs. The problem is how to convert them to some variety of data input form in a computer database. The obvious way is to create a screen equivalent to the form, and this is often done. However, there are several disadvantages to this. For example, the paper version frequently necessitates a single form for each patient, but on the screen different layouts can be achieved. Considering the booking form PEF shown on the previous page, each form is for one pregnancy for a single patient. However, a possible screen equivalent might be something more like the window displayed below, where a user can select a particular pregnancy for a given patient.

Patient details: id: Hospital no: NHS no: DOB:

First name: curSname: Status:

Pre preg. Diabetes: Consultant: oldSname1: Date of death:

oldSname2: Code of death:

oldS3: Death text:

List of Booking visits (pregnancies) for patient:
Select the correct booking record by clicking on the far left of the record:
Ignore first two fields - for development purposes

Pregnancy id	client id	dateseen	Gestation at visit (week)	Source	Diabetes consultant during pregnancy:	Obs_Gynae consultant:	Occupation
42	19	03/08/2000	4	NDC	Marshall	Loughney	
▶ 20	19	29/04/1997	6	Other	Senior	Davison	
* jtoNumb	19	22/05/2001					

Once you have chosen the correct pregnancy record please click on the appropriate button opposite to add visit or delivery details.

Visits

View / Edit: Add:

Pregnancy

This provides immediate visual feedback to the person entering the data as to the possibility of the data having already been entered, whereas a direct implementation of the PEF would not have provided this. Also because the data is being entered retrospectively it is likely that more than one pregnancy record may be entered in a single session. Given this requirement, ease of navigation between pregnancies and patients is more important than within a single pregnancy for the person entering the data. In contrast, if the database were to be used during the consultation the opposite would probably apply.

Exercise 6. Comparing PEF with screen equivalent

You should spend no more than 30 minutes on this exercise

Consider the booking form PEF shown on the previous page as well as the possible screen equivalent on this page and compare the two.

What bad features does the screen version possess?

Do you think it is necessary to show such things as the pregnancy and client id fields, which are basically fields to make the database work (foreign keys)?

If you were told that this database was designed for research purposes where the user wished to be able to extract raw data from the various tables for analysis, would it change the answer you gave to the above question?

7. Windows and Controls

Hutt & Flower 1990's stage three is 'Design visual images' which within the Windows environment basically means menus, toolbars, windows and icons. (I assume you recognise each of these concepts.)

Windows can be of various types: modal and modeless. Modal windows require the user to close the window before moving to another one, as is the case with most. In contrast, modeless windows allow you to move off the window thus keeping it in the background, such as with help windows. Windows can also have other optional features such as allowing you to resize or minimize windows, whereas others do not.

Each window, in turn, can contain one or more controls, each of which works in a particular way. For example a static control simply displays text on the window, whereas an edit control allows a user to input data. Carry out the task below to refresh your memory of how each of these controls works.

Exercise 7. Windows controls

You should spend no more than 30 minutes on this task

Go to the “Official Guidelines for User Interface Developers and Designers” guide at and skim read through the page do not read it thoroughly <http://msdn2.microsoft.com/en-us/library/ms997492.aspx> the contents pages can be found at: <http://msdn2.microsoft.com/en-us/library/ms997612.aspx>.

When you have finished it complete the table below:

Control	Use
Static control	displays text
Edit control	allows user to input text
List box	
Combo box	
Push button	
Check box	
Radio button (option button)	
Group box	

7.1 Mapping PEFs to Controls

Now that you clearly understand the functions of the various controls that a window might possess, you should be able to map the various items in a PEF to the appropriate control. For example, the diabetes booking form PEF contained the following question:

Renal problems (ring one) None=1;
 Microalbuminuria=2; Proteinuria=3;
 Abnormal creatinine=4; Other=5

This question would probably become a set of option buttons as each of the responses is mutually exclusive.

Exercise 8. Converting PEFs to windows controls

The following is a picture of the window to collect data concerning the diabetic mother's condition during delivery. Design a PEF based upon the information. You will need to make some guesses, but please don't worry about them being clinically correct. If you wish you can design the PEF using the drawing tools in Word for Windows or just draw it on a piece of paper using a good old pencil.

delivery

Id: 26 Pregid: 21 date: 12 September 2000 Source: [dropdown]

Mother

No of admissions during pregnancy: 0 Reasons for admission during preg: [Data not collected] Bishop 1: [] 2: [] 3: []

Admitted after delivery: [data not coll] Reasons for admission after delivery: [Data not collected]

Days gestation at deliver: 269 Labo: [] Induced: [] Total PG dose: 1 Time first PG (24hr clock): 18:50 ARM: [x]

Entonox Opiates Epidural Spinal GA other analgesia Stage 1 min: 2 Stage 2 min: 3 Syntocinon: [x]

Maternal Insulin Details Reason Em CS: [data not collected] Reason for assisted delivery: [data not collected]

Maternal Insulin changes during labour (leave blank if data not available): [] Was maternal IV glucose needed?

Was maternal hypoglycaemia corrected?

Last 8 Blood Glucose Pre-Delivery -1: 6.4 -2: 5.7 -3: 5.9 -4: 5.6 -5: 6.4 -6: 5.7 -7: 6.6 -8: 4.6

Childrens details [Open Form] [Save Record]

Record: [] of 1 (Filtered)

7.2 Windows and Data Structures

The window presented to the user is usually a result of a database query rather than simply the presentation of a table on a screen. This was clearly the case in the example of the pregnancy details shown again below.

Patient and pregnancy details

Patient details: id: 19 Hospital no: 86486168 NHS no: [] DOB: 22 November 1975

First name: Claire curSName: sooy Status: [Data not collected]

Pre preg. Diabetes: [] oldSName1: [] Date of death: []

Consultant: [] oldSName2: [] Code of death: []

oldS3: [] Death text: []

List of Booking visits (pregnancies) for patient:

Select the correct booking record by clicking on the far left of the record:
Ignore first two fields - for development purposes

Pregnancy id	client id	dateseen	Gestation at visit (week)	Diabetes consultant during pregnancy:	Occupation
				Source	Obs_Gynae consultant:
42	19	03/08/2000	4	NDC [] Marshall []	Loughney []
▶ 20	19	29/04/1997	6	Other [] Senior []	Davison []
* utoNumb	19	22/05/2001		[] []	[] []

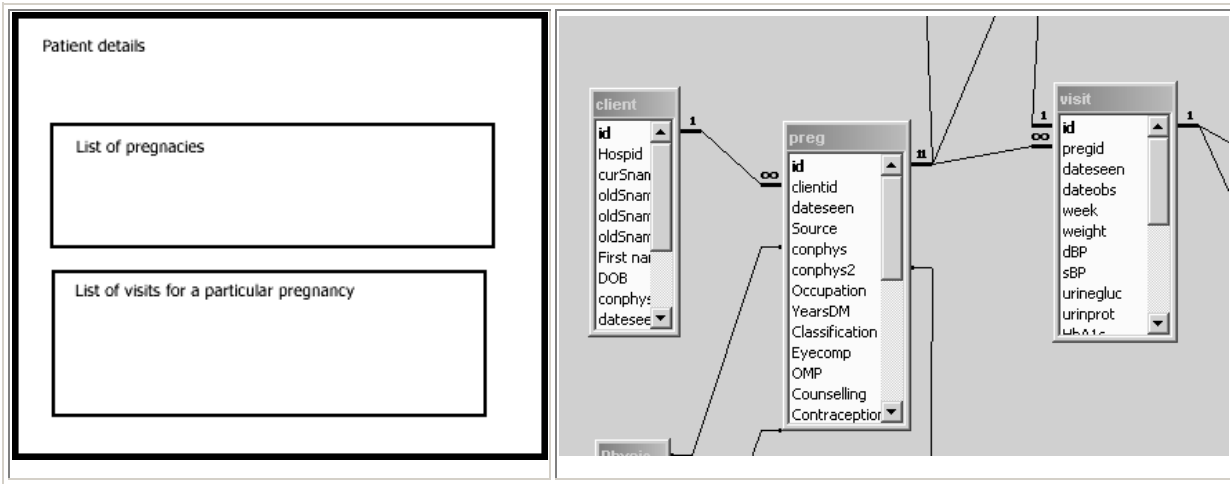
Once you have chosen the correct pregnancy record please click on the appropriate button opposite to add visit or delivery details.

Visits

View / Edit: [All VISITS] Add: [pre-] [during] [After] [Delivery]

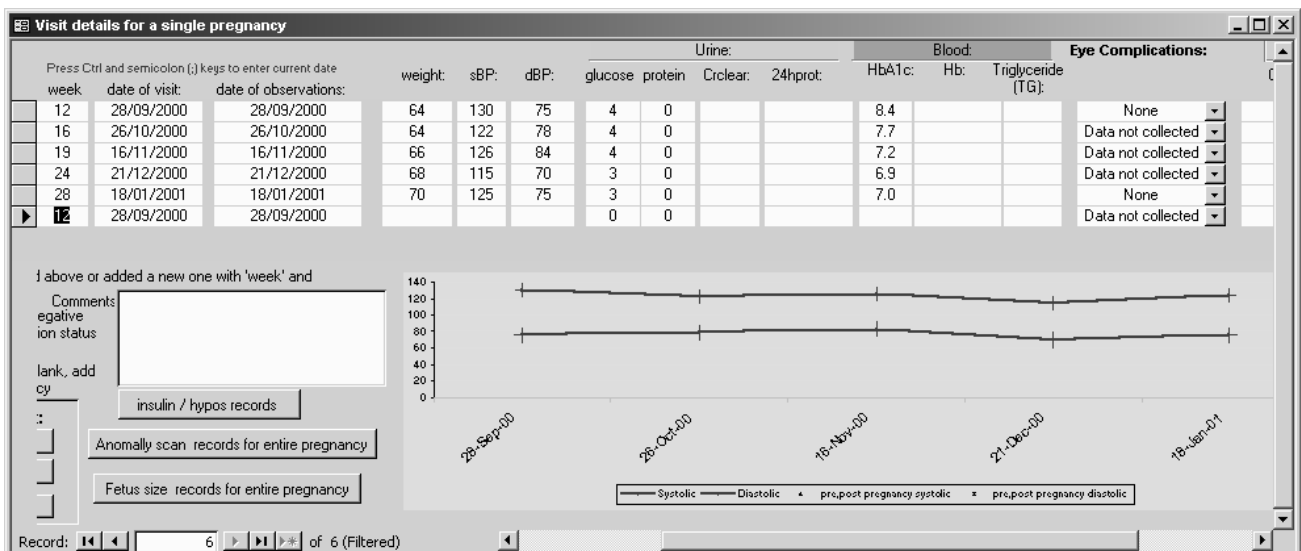
Pregnancy

Each patient has zero or more pregnancy records, and obviously the 'List of Booking visits' changes for each patient selected. It should also be noted that there are a number of buttons to see details of the individual visits for each of the pregnancies. In contrast to the above layout, you could have set the window up like shown below on the left. On the left is a diagrammatic portrayal of the window while beside it the relevant part of the database schema is shown:



Even more rarely is an unfiltered table displayed on the screen. In the above examples each table has in effect a filter, the client table displaying only one record, while the pregnancy table is filtered to show only the relevant pregnancies as would be the visits table.

Clearly the amount and type of information on the screen is related to how the user intends to use the screen. Data input screens and viewing screens may appear very different but show the same data. For example the screen shot below shows how records of visits are displayed to help the user notice trends developing as the data is entered by showing various data items graphically:



Exercise 9. suggesting improvements to a screen design

You should spend no more than 15 minutes on this task

The above examples are from a prototype database. Look carefully at them and then write beside the pictures some improvements that could be made.

8. Print versus Screen

Because the use of the Web is ubiquitous, with the palmtop catching up, much material is now being specifically produced for viewing on a screen rather than on the printed page. Conversion of this screen output to a printed version is often considered to be rather trivial, basically because both are seen as text based media. For example, Acrobat 5 and the reader allow you to 'reflow' text to suit any viewing width. However, this conversion is really far from trivial. There are fundamental differences between the media including:

"The resolution of a screen font is less than 1/4 of its printed counterpart. Because of that, it's harder to read text on the screen than it is in print, since the eyes spend more time trying to blend the shapes together into something legible on the screen" (Hunter 2000, p124).

"One consequence of this is that font sizes need to be larger to be more legible. Unfortunately, what looks just fine on the screen can look huge on the printed page and visa versa, so it's a good idea to reset the size of fonts when you're outputting to the printer." (Hunter 2000, p124) When working with Web-based material this can easily be achieved by using the @media directive in a cascading stylesheet.

The screen size/shape is different to that of the standard printed page.

The method of reading a computer screen and the Web ('browsing') is different from that of reading a traditional paper based text. For example Krug 2000 says that people use the web by scanning, satisficing (choose the first reasonable option) and muddling through (click the mouse rather than think what its going to do etc). Therefore one of his recommendations is that web designers design pages for scanning not reading - a very different approach to that of the traditional author.

This is a huge topic, and we will not investigate it any further, I haven't even mentioned the topic of e-learning environments!. I only wanted to shatter any beliefs you might of had concerning the similarity between the printed page and the web other than at a very superficial level.

9. Screen Layout Standards

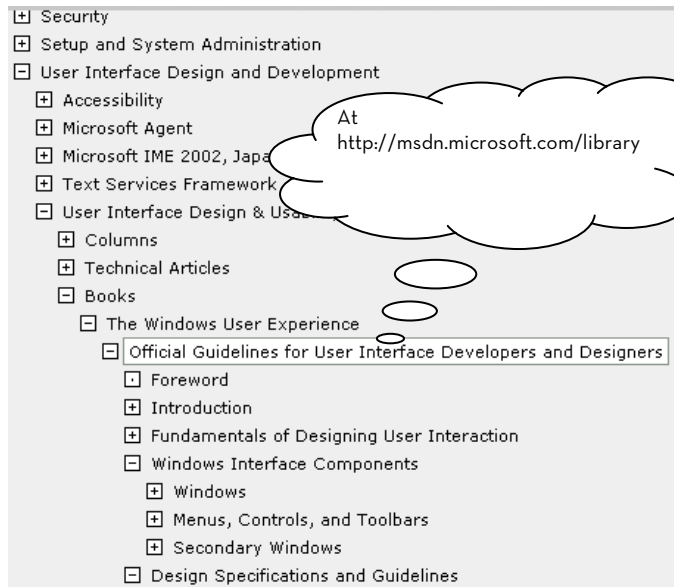
In a following section we will discuss standards concerning luminosity and contrast, the Web Content Accessibility Guidelines (WCAG) 2.0 W3C guidelines. in fact the same document does discuss the screen real estate as well however there are also ISO standards such as ISO 9241-10/12 See http://www.hostserver150.com/usabilit/tools/r_international.htm if you want to buy the documentation go to: <http://www.iso.org/>. <http://www.system-concepts.com/> which provides comments about the latest draft, there is also a bluffers guide to it at <http://www.userfocus.co.uk/articles/ISO9241.html>. The first link also provides details of other relevant standards.

De facto standards include:

Apple at <http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHIGuidelines/index.html> which is also available in PDF format (warning: it is about 14 Mb - April 2007). This guide is excellent in many respects for anyone thinking about developing a computer application or as a list of the best references concerned with HCI issues. Click here to see the [local copy](#).

Warning: many of the urls on this page seem to have disappeared - any help on finding their replacements is welcome. Robin Beaumont April 2007

The Microsoft Official Guidelines for designers and developers at <http://msdn.microsoft.com/library/books/winguide/ch00a.htm> unfortunately this is frequently moving so I have included exact details of where it is on the site in the screenshot opposite. If you click the sync toc button on the top left frame you can see the tree as show opposite. I have also come across a pdf version (3.5 Mbytes and 420 pages at <http://www.gpa.etsmtl.ca/cours/gpa789/pdf/Official GUI.pdf>) but am not sure how old it is.



The DoD TRM (DISA 1999) is a composite model derived from the TAFIM (see below) and the SAE GOA see <http://trm.disa.mil/>. The latest version seems to have adopted the Sun Systems user interface style guidelines rather than developing there own in detail, see below. <http://trm.disa.mil/trmv2.pdf>

This also relates to the older JASA [Joint Airborne SIGINT Architecture] standards handbook 1997 chapter 5 HCI <http://www.fas.org/irp/agency/daro/product/jasa/chapter5.html>

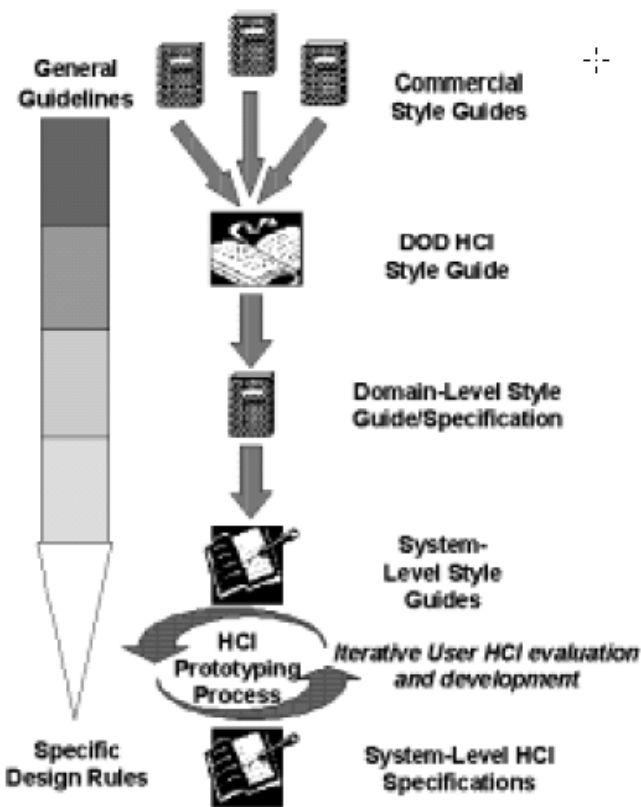


Figure 1. The hierarchy of style guides that are to be followed to maintain consistency of the human-computer interface design (From DA, 1998).

TAFIM (Technical Architecture Framework for Information Management) was a project funded by the US Centre for Standards (CFS) which is itself part of the Defense Information Systems Agency (DISA). For an overview of the TAFIM documents see <http://sunset.usc.edu/GSAW/GSAW98/bowser.pdf> or <http://www.sei.cmu.edu/str/descriptions/tafim.html>. Unfortunately much of the details seem now to have been removed from the web. The old, now defunct, link was (http://www-library.itsi.disa.mil/tafim/tafim3.0/pages/tafim_8.htm working early 2003) to all the source documents. However the minutes from various subgroups are still available which makes fascinating reading on a cold winters night, for example, http://hfetag.dtic.mil/docs-hfs/hfs_minutes_11-99.doc You can also find other US military standards documents at http://hfetag.dtic.mil/hfs_docs.html Much of what was once freely downloadable now seems to be chargeable - one wonders if the whole US military is becoming a commercial organisation?

From the above one is bewildered by the complexity of the situation. Fortunately the various User interface guidelines for various military applications is succinctly described along with some nice graphics (one shown above) in the summer 1999 issue of the US military Manprint magazine.

http://www.manprint.army.mil/manprint/docs/quarterly/summer_fall%2099%20qtrly.pdf

My advice is not to try and understand the detail of the above situation but just be aware of the general flavour of it! If you wish to follow up this topic area I would suggest you begin by reading chapters 8 and 9 (p227 - 253) of Nielsen 1993.

9.1 Fonts

The question as to which fonts to use for web pages has been investigated in detail over the last few years. Consider the abstract below.

****start of abstract from http://www.webusability.com/article_more_about_fonts_3_2002.htm *****

***** full article including font characteristic descriptions at <http://www.humanfactors.com/downloads/feb02.asp#bobbailey> *****

More About Fonts

by Dr. Bob Bailey *March, 2002*

Tom Tullis and his colleagues (1995) used a proofreading task to evaluate the differences in reading rate between type styles and sizes. They had subjects use Arial, MS Sans Serif, MS Serif and Small Font type styles all at 6, 7, 8, 9 and 10 points. The reading material in this and all of the following studies used a black font on a white background. There was no difference between serif and sans serif fonts; however, the 9-point and 10-point fonts elicited reliably faster performance than the smaller sizes. Subjects preferred the 10-point Arial and MS Sans Serif fonts rather than the MS Serif fonts.

Boyarski, et.al. (1998) at Carnegie Mellon University evaluated the reading speed of people using Georgia (serif), Times New Roman (serif) and Verdana (sans serif) fonts. Georgia and Verdana were specifically designed for reading from a computer monitor. All text was set at 10 points. They used a comprehension task (Tinker Reading Speed Test) rather than a proofreading task. Participants included faculty, staff and graduate students who ranged in age from 20 to 53. The subjects read the text on a 17-inch screen with a resolution of 640x480 pixels. They reported no reliable performance differences in reading speed.

Over the past couple of years, Michael Bernard, Melissa Mills and their colleagues at Wichita State University have conducted a series of studies on font sizes and styles. In their first study (Bernard and Mills, 2000) they evaluated 10-point and 12-point Arial and Times New Roman fonts. They had participants read Encarta passages.

The words were presented on 17-inch monitors with a resolution of 1024x768 pixels. The test subjects were asked to read each passage "as accurately and as quickly as possible." As they read, they were to find some randomly placed "substitution words" in each passage (e.g., "fun" replaced "sun"). The researchers reported no reliable differences among the passages in reading speed or in the detection of word errors. However, the 12-point fonts were reliably preferred over the 10-point fonts.

In a second study (Bernard, et.al., 2001a) used a similar procedure to evaluate three different font sizes (10, 12 and 14-points) used with eight different fonts types:

- Serif fonts
- Century Schoolbook
- Courier New
- Georgia
- Times New Roman
- Sans serif fonts
- Arial
- Comic Sans
- Tahoma
- Verdana

They found that Arial and Times New Roman were read reliably faster than Courier, Schoolbook and Georgia, and that the 12-point fonts were read reliably faster than the 10-point fonts. All of the fonts except Century Schoolbook were reliably preferred over Times New Roman.

In a more recent study (Bernard, et.al., 2002) they used only 12-point fonts, but extended the number of font styles by adding Goudy Old Style (a serif font) and Agency (a sans serif font) to their original eight font styles. Like in previous studies, participants read 2 to 3 page passages located at a fixed distance from their screens. They found no reliable differences between the major fonts in reading efficiency; however, Arial, Verdana and Comic Sans were reliably preferred.

The studies just discussed used participants that were young or middle-aged. Bernard and his colleagues (2001b) examined font characteristics that could assist older adults when reading from the Web. The older users had an average age of 70, with a range of 62 to 83 years. The study used 15-inch monitors that had a resolution of 800x600 pixels. Again, all of the readers were required to remain 22 inches (56 cm) from the screen. They evaluated 12-point and 14-point versions of Times New Roman, Georgia, Arial and Verdana. All the 14-point fonts produced better reading efficiency, and both of the 14-point sans serif fonts were reliably preferred over all four 12-point fonts.

What can we conclude from these studies?

No Web page fonts should be less than 10-points,

Optimal reading speed for most adults will be elicited with 12-point fonts (size=3)

There is probably no reliable difference in reading speed for most adults when viewing common font styles (Arial, Verdana, Georgia, Times New Roman),

Most users tend to prefer sans serif fonts (Arial, Verdana), and

Older users will benefit from type sizes that are at least 14-points

References

Bernard, M. and Mills, M. (2000), So what size and type of font should I use on my Web site? *Usability News*, July, 2(2). <http://psychology.wichita.edu/surl/usabilitynews/2S/font.htm>

Bernard, M., Mills, M., Peterson, M. and Storrer, K. (2001a), A comparison of popular online fonts: Which is best and when? *Usability News*, July, 3(2). <http://psychology.wichita.edu/surl/usabilitynews/3S/font.htm>

Bernard, M., Liao, C. and Mills, M. (2001b), Determining the best online font for older adults, *Usability News*, January, 3(1). <http://psychology.wichita.edu/surl/usabilitynews/3W/fontSR.htm>

Bernard, M., Lida, B., Riley, S., Hackler, T. and Janzen, K. (2002), A comparison of popular online fonts: Which size and type is best? *Usability News*, January, 4(1).

<http://psychology.wichita.edu/surl/usabilitynews/41/onlinetext.htm>

Boyarski, D., Neuwirth, C., Forlizzi, J., and Regli, S.H. (1998), A study of fonts designed for screen display, *CHI'98 Conference Proceedings*, 87-94. <http://www.humanfactors.com/downloads/feb99.asp#screenfonts>

Tullis, T.S., Boynton, J.L. and Hersh, H. (1995), Readability of fonts in the windows environment, *CHI'95 Conference Proceedings - Extended Abstracts*, 127-128.

http://sigchi.org/chi95/Electronic/documnts/intpost/tst_bdy.htm

***** End of Abstract *****

In passing you might have noticed the Interesting link to guidelines for writing patient information leaflets at one of the sites above.

<http://www.med.utah.edu/pated/authors/literacy.html>

Exercise 10. Screen font, size and type

You should spend no more than 30 minutes on this task

After considering the above information, what would you consider to be:

The preferred font size? And is there a problem with this?

Previous research had identified an advantage of serif fonts. What was it?

Exercise 11. Screen and font, colour

You should spend no more than 10 minutes on this task

Colours on the screen are also important. See <http://hubel.sfasu.edu/research/reslvs.html>

What do you think of the colour combination used?

9.2 Standards for Luminosity/contrast on web pages

In the last few years various guidelines for web accessibility have been developed the most pertinent to our present discussion is the Web Content Accessibility Guidelines (WCAG) 2.0 W3C Recommendation 11 December 2008 available from <http://www.w3.org/TR/2008/REC-WCAG20-20081211/>

This is what it has to say about luminosity and contrast on web pages:

contrast ratio

$(L1 + 0.05) / (L2 + 0.05)$, where

- L1 is the **relative luminance** of the lighter of the colors, and
- L2 is the **relative luminance** of the darker of the colors.

Note 1: Contrast ratios can range from 1 to 21 (commonly written 1:1 to 21:1).

Note 2: Because authors do not have control over user settings as to how text is rendered (for example font smoothing or anti-aliasing), the contrast ratio for text can be evaluated with anti-aliasing turned off.

Note 3: For the purpose of Success Criteria 1.4.3 and 1.4.6, contrast is measured with respect to the specified background over which the text is rendered in normal usage. If no background color is specified, then white is assumed.

Note 4: Background color is the specified color of content over which the text is to be rendered in normal usage. It is a failure if no background color is specified when the text color is specified, because the user's default background color is unknown and cannot be evaluated for sufficient contrast. For the same reason, it is a failure if no text color is specified when a background color is specified.

Note 5: When there is a border around the letter, the border can add contrast and would be used in calculating the contrast between the letter and its background. A narrow border around the letter would be used as the letter. A wide border around the letter that fills in the inner details of the letters acts as a halo and would be considered background.

Note 6: WCAG conformance should be evaluated for color pairs specified in the content that an author would expect to appear adjacent in typical presentation. Authors need not consider unusual presentations, such as color changes made by the user agent, except where caused by authors' code.

Exercise 12. Webpages Luminosity/contrast

Also investigate the colour contrast checker at http://www.snook.ca/technical/colour_contrast/colour.html

There is also available a online Luminosity Colour Contrast Ratio Analyser at

<http://juicystudio.com/services/luminositycontrastratio.php>
this provides the luminosity index mentioned above.

Colour Contrast Check
Date created: January 11, 2005
Date last modified: December 6, 2009

Results	
This is example text. Some of it bolded. Some of it italicized.	
Brightness Difference: (>= 125)	220.60
Colour Difference: (>= 500)	504
Are colours compliant?	YES
Contrast Ratio	18.08
WCAG 2 AA Compliant	YES
WCAG 2 AA Compliant (18pt+)	YES
WCAG 2 AAA Compliant	YES
WCAG 2 AAA Compliant (18pt+)	YES

9.3 Standardisation of Controls

The previous page referenced several de facto standards. These frequently provide guidelines concerning every small detail of the display such as the relative position of 'OK' and 'Cancel' buttons in a window. Such standardisation means that a user can move more quickly between various applications as well as learn how to use new ones more rapidly than was previously possible.

Exercise 13. Positioning of buttons

You should spend no more than 10 minutes on this task

In a Microsoft Windows application (e.g. Word) where is the 'Cancel' button in relation to any other button in a window?

10. Low Level Dialogue Design

We are now ready to consider the last stage in Hutt & Flower's method: the design of the low level dialogue. We have already touched on this issue discussing windows controls. What we will now consider is the problem of actually specifying on paper the behaviour for a particular window.

Our old friend the state diagram has been mentioned several times since its introduction in the subsection on dynamic modelling. You may remember when it was first introduced that a single state diagram was used for each object, by considering each window as a separate object. We can once again use the same technique.

The following example is taken from Dix, Finlay, Abowd & Beale 1993.

The example describes a fictitious drawing tool. The user can choose between the options of drawing a circle or a line from the 'menu'. The term 'rubber band' is used in the diagram which is simply is a temporary representation of the object on the screen which the user can manipulate. If the user selects the circle option all the user can do is click on the circumference whereas if the user chooses the line option he/she can add any number of points. To complete the line they must double click.

256 Dialogue Notations and Design

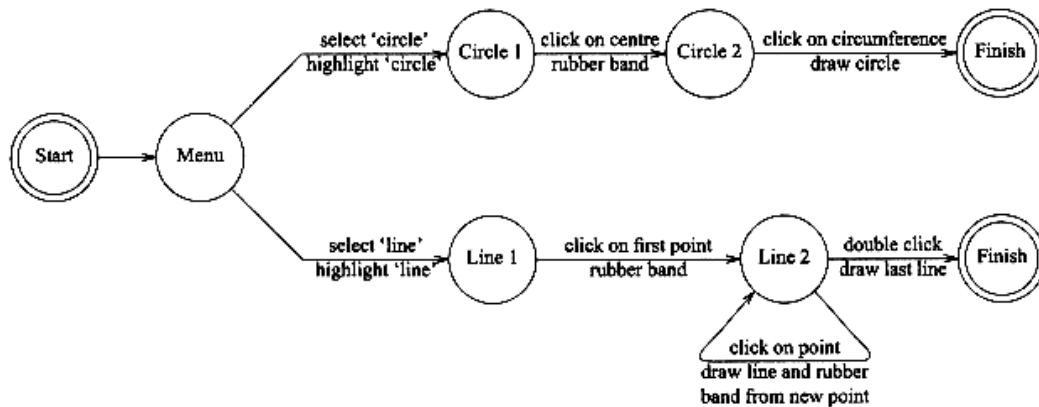
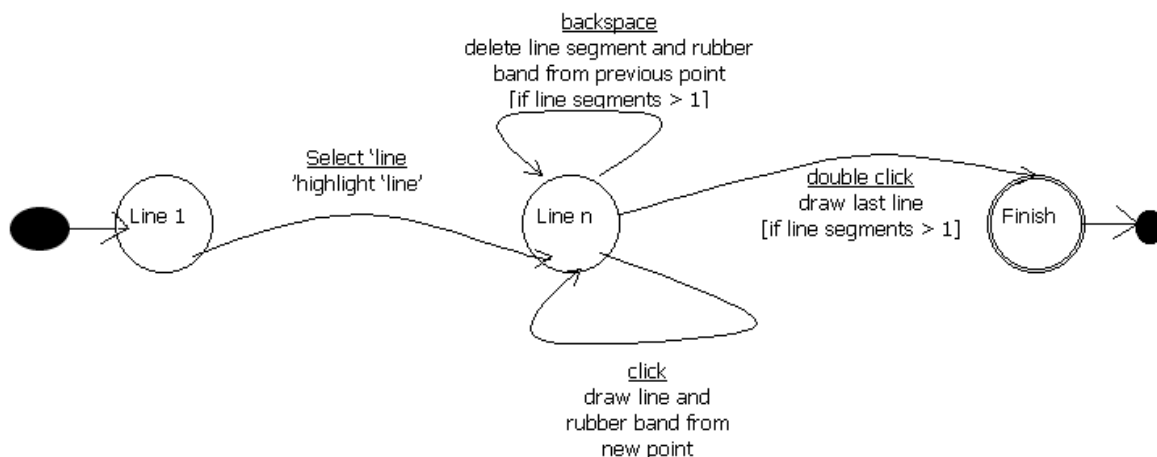


Figure 8.1 State transition network for menu driven drawing tool

Using the above design to develop the drawing package would mean that a user would only be able to draw a single circle before once again selecting the appropriate menu option. Adding another transition line would solve the problem.

Suppose that we would like to model another enhancement: the user has requested that they can press the **Backspace** key to delete the last line segment added except for the first one. (This enhancement has been taken from Green 1986.) Adding condition statements to the above diagram allows us to model this situation. (Technically this type of state diagram is very similar to what is called an enhanced STD; see Green 1986 for details.) Each transition line has on it: **User action / Affect / Condition:**

Draw Line - state diagram 001



10.1 Escapes and Help

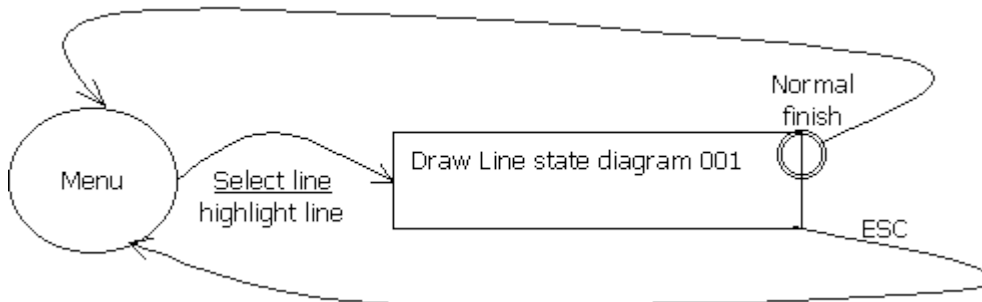
The state diagram is now beginning to look quite complex, but there are still features missing:

At any point we would like the user to be able to press the 'Esc' key to save and leave the drawing as it is.

At any point we would like the user to be able to call up the 'help' system.

If we showed these features in the current state diagram we would need an additional line from each state, but this would make the diagram extremely cluttered. Let's consider the 'Esc' key problem first.

A way around this problem is to use the 'nesting' concept described earlier along with some additional notation. By giving the state diagram we wish to nest within another (representing what now become substates) an identifier and using notation to indicate the transition is 'normal' or generic from each substate we can model the above. Notice the double circle to indicate a normal exit.



The problem with diagrammatically showing help is more complex as the 'help system' returns the user to the point from which they called it. Virtually every modeller has offered their own solution to this problem, and I'm sure you are capable of providing your own. The important thing is to be consistent throughout the modelling documents and provide a clear description of your method.

It cannot be denied that there are problems with modelling low level dialogues within the windows environment. Each window (and dialogue box) can have numerous states ('**combinatorial explosion** of states' if the window possesses several groups of options). I personally feel that the above state diagrams are best used at the window to window navigation description level. It is probably as quick nowadays to produce a screen prototype using Access or a specialist tool as it is to specify on paper the behaviour of individual windows. Clearly the situation is slightly different for the actual individual window layout where paper versions may be useful in elucidating requirements from users.

11. Interactive Iterative Prototyping - Collecting Information from the User to Improve the Design

It must not be forgotten that the best way to design a system is to adopt the iterative prototyping approach. If such an approach is adopted it is possible to collect details of how the user interacts with the screen for the purpose of evaluating the system to provide suggestions for further enhancements. Two examples of this approach are given below.

Payne & Howes 1992 developed a tool which allowed them to trace the activities of users of a new computer system to see how they 'learnt' the system, hence analysing the **learnability** aspect of it. Sears (1993) developed a mathematical technique based upon retrospectively analysing movements between windows components (he called them Widgets) to derive an optimal layout. In his article he presents a nice state chart annotated to show the relative proportion of interactions users made between controls in a dialogue box. You can then use this information to place them in the most ergonomic position. If you are interested in using a mathematical method for deriving the appropriate layout of controls in a window this article is the one for you!

12. Usability

Much of what we have discussed in this subsection would be covered under the topic of usability. To give you a feel for what usability is and how it is assessed in the real world, have a look at the following link which is of a report co-authored by myself evaluating an evaluation report of a computer based case management system for the UK probation service.

Read sections 4.3 to 4.5, which provide a description of what usability is and how it is related to such things as Shekel's four factors and user satisfaction.

The report: <http://www.robin-beaumont.co.uk/probation/assess1.pdf>

13. Revision Exercise

You should spend no more than 120 minutes on this task

Answer two of the questions below:

1. Select a paper form you regularly use at work and sketch out a suitable computer user interface to:

Collect data from it

View aggregated information from a set of forms

2. Consider a computer system with which you work and attempt to draw a navigation route through it (only consider a typical scenario).
3. Take one of the screens ('windows') from a computer system with which you work and attempt to produce its low level dialogue in the form of a state diagram.
4. Consider a computer system with which you work. Suggest where graphical descriptions of the data might be useful to display alongside the standard numerical format. Give your reasons.

14. Summary

This document has looked at various aspects of what is presented on computer screens. We specifically looked at specifying navigation between, and behaviour within, windows. Two possible development methods were presented, STUDIO and Hutt & Flower 1990's method. We worked through the Hutt & Flower method specifically looking at various techniques that can be used to help collect the necessary information, including task analysis and the conversion of paper based data collection forms (Patient Encounter Forms PEFs).

The idea of the separation between a back end application and a front end with which the user interacts was presented in the Seeheim model. This was emphasised with several database examples where queries formed the basis for the front end.

Both professional (ISO) and de facto standards were considered in specifying screen navigation and content, the former being at a much more general level than the very specific recommendations made by companies such as Microsoft and Apple.

The relationship between paper based and screen output was discussed, highlighting the differences and problems with converting material between the two in a haphazard way.

Several tasks were introduced to highlight specific issues such as menu structures and the scope of HCI (Human Computer Interaction) science. We also looked briefly at usability and how this is evaluated in the real world.

15. Further reading

This depends very much on which aspect of HCI you are interested in. The web is a good source for general HCI information- see the following links section.

Concerning books Preece et al 1994, Dix 1993, and Schneiderman 1997 are good introductions. For web design start with Nielsen 1993, Veen 2001 and Krug 2000

For information about those who develop type faces a very interesting place to start is Carter 1995. For examples of more trendy /web implementations see Bellantoni & Woolman 1999.

16. Multiple Choice Questions

You should spend no more than 30 minutes on this task

Below are a few MCQs (Multiple Choice Questions) to help you revise the material presented in this document.

1. Donald Norman (1988) in his book *The Psychology of Everyday Things* identified which two principles that he considered would help towards good interface design (choose two)?

- a) Appropriate colour
- b) Visibility
- c) Size
- d) Affordance
- e) Positioning on the screen

2. What did the Seeheim model recommend?

- a) That software be designed in modular components
- b) That software be designed in which a way that there are as few modules as possible
- c) That software be designed in several modules separating the interface from the application
- d) That software be developed by properly trained professionals only
- e) That software be tested using a rigorous methodology

3. Hutt & Flower 1990 describe a four stage process for developing user interfaces. Which of the following is the correct sequence?

- a) Overall appearance, Dialogue control, Visual Images, Data description
- b) Overall task analysis, Dialogue control, Visual Images, Low level dialogue
- c) Overall appearance, Dialogue control, Visual Images, Low level dialogue
- d) Overall task analysis, Dialogue control, Low level dialogue, Visual Images
- e) Overall appearance, Dialogue control, Visual navigation, Low level dialogue

4. Which of the following is said to be the good characteristics of a menu/navigation system:

- a) Flat and narrow number of options
- b) Highly nested
- c) Flat and broad
- d) Deep and broad
- e) Deep with a narrow number of options

5. The STUDIO (STructured User-interface Design for Interaction Optimisation) method has five stages. Which of the following lists correctly represents them?

- a) Project proposal and planning, User requirements analysis, Task analysis, Usability engineering, Evaluation
- b) Project proposal and planning, User requirements analysis, Data analysis, Usability engineering, User interface development
- c) Project proposal and planning, User requirements analysis, Data analysis, Usability engineering, Testing
- d) Project planning, User requirements analysis, Task analysis, Usability engineering, User interface development
- e) Project proposal and planning, User requirements analysis, Task analysis, Usability engineering, Testing

6. Which of the following most accurately describes task analysis:

- a) Task analysis is the process of analysing the way people interact with machines and the effect machines may have upon the worker.
- b) Task analysis is the process of analysing the way people perform their jobs and developing ways of optimizing these tasks.
- c) Task analysis is the process of analysing the way people interact with machines in their jobs: the things they do, the things they act on and the things they need to know.
- d) Task analysis is the process of analysing the way people interact with machines in their jobs: the things they do, the things they act on and the things they need to know to make the machines perform effectively.
- e) Task analysis is the process of analysing the way people perform their jobs: the things they do, the things they act on and the things they need to know.

7. When developing dialogues with computers, which of the following will inevitably cause problems:

- a) The speed at which the task is performed is changed.
- b) The complexity of carrying out the task is reduced.
- c) The speed at which the task is performed is lengthened by the introduction of a computer.
- d) The task becomes more standardised.
- e) The task is provided with helpful hints.

8. Which of the following could be seen as advantages of PEFs over traditional unstructured medical records?

- a) They provide active reminders.
- b) They reduce the need for professionals.
- c) They encourage standardisation of data collection.
- d) They provide the opportunity for analysis.
- e) They can be seen simultaneously by several people.

9. What is a modal window?

- a) One that opens at the start of an application
- b) One that requires the user to close the window before moving to another one
- c) A window that cannot be resized by the user
- d) One that allows the user to move off it, thus keeping it in the background
- e) One where the position is fixed on the screen

10. Which of the following is not a windows control?

- a) Edit
- b) Combo box
- c) Check box
- d) Chime
- e) Static

11. Which of the following statements about the relationship between screen and paper formats is **NOT** correct:

- a) The resolution of a screen font is less than 1/4 of its printed counterpart.
- b) Screen font sizes need to be larger to be comparably legible to the printed page version.
- c) You can read more quickly on the screen than from a printed version.
- d) The screen size/shape is different to that of the standard printed page.
- e) The method of reading a computer screen and the Web ('browsing') is different from that of reading a traditional paper based text.

12. Bob Bailey 2002 discussed various research findings which indicated that the minimum font type for web pages should be one of the following sizes?

- a) 10-point
- b) 12-point
- c) 14-point
- d) 13-point bold
- e) 12-point bold

13. Bob Bailey 2002 discussing the various findings suggests which of the following is sensible (choose one option)?

- a) Serif type fonts (eg Times New Roman) which be used more often as user prefer them over San serif fonts.
- b) There is no difference in the font requirements for younger or older web users.
- c) Comic Sans was evaluated as being the best font for all age groups.
- d) For the elderly the minimum size font should be 14
- e) There was no practical difference between serif and san serif fonts from the users perspective.

14. What is the principle difference between standards produced by recognised 'standards' bodies and those proposed by manufacturers (ie de facto)?

- a) Recognised standards are concerned purely with implementation.
- b) De facto standards usually provide less detail than those produced by 'standards' bodies.
- c) De facto standards usually require some type of registration fee before you can use them.
- d) De facto standards tend to be at a more detailed level.
- e) Standards produced by standards bodies can usually be enforced, in contrast to de facto standards which are purely recommendations.

15. Which of the following types of diagram is usually used to model the low level dialogue?

- a) State diagram
- b) ERD
- c) Object model
- d) Flow chart
- e) Use case diagram

16. If a state diagram is used, what can be shown along the state lines?

- a) Computer prompt/user response/condition
- b) Computer prompt/affect/condition
- c) User action/screen changes/condition
- d) User action/affect/condition
- e) User action/affect/next state

17. Shekel 1990 suggests that usability consists of the following four factors:

- a) Learnability, Error rate, Flexibility, Attitude
- b) Learnability, Throughput, Flexibility, Attitude
- c) Learnability, Error rate, Support, Anxiety level
- d) Learnability, Error rate, Flexibility, Attitude
- e) Learnability, Error rate, Flexibility, Anxiety level

18. What is generally meant by the term adaptive user interface?

- a) A computer interface that can cope with the ways different people use it
- b) A computer interface which changes depending upon how the user interacts with it
- c) A computer interface which will be developed over time (another name for screen prototyping)
- d) A computer interface that adapts to different environmental conditions (eg lighting etc)
- e) A computer interface that allows users to choose from a set of options to personalise it

17. References

One of the most important list of references for literature concerning HCI and screen design in particular is the Apple user interface Guide available online at

http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHIGuidelines/XHIGIntro/chapter_1_section_1.html

Bellantoni jeff, Woolman Matt, 1999 Type in motion: innovations in digital graphics. Thames & Hudson. London

Borenstein, Nathaniel 1991 Programming as if People Mattered.. Friendly Programs, Princeton University Press Software

Brown, Judith R. 1989 Programming the User Interface: Principles and Examples, Wiley.

Browne D p 1994 STUDIO: Structured User-interface Design for Interaction Optimisation. Prentice Hall

Carroll, John M. 1991 Designing Interaction: Psychology at the Human-Computer Interface, Cambridge University Press.

Carter Sebastian 1995 (2nd ed) Twentieth century type designers. Lund Humphries publishers London

Cramer M 1990 Structure and mnemonics in computer and command languages. Int. j. Man-Machine Studies 32 707 - 722 [This article provides experimental evidence that mnemonics are stronger than structural aspects in using command languages. This may seem rather irrelevant but many power users still use short cut keys.]

Dix A, Finlay J, Abowd G, Beale R 1993 Human-Computer Interaction. Prentice Hall [book approximately 560 pages]

Dixon, Robert L. 1991 Winning with CASE, Tab Books

Edmonds E 1981 The man-computer interface: a note on concepts and design. int. j. Man-machine studies 16 231- 236

Edmonds E (ed) 1990 The separable user interface. Academic press. ISBN: 0-21-232150-2

Edmonds E 1990 The emergence of the separable user interface. ICL Technical Journal [now called the systems journal] May

- Fowler, Susan, Victor Stanwick 1994 *The GUI Style Guide*, Academic Press Professional, (contact FAST Writing Corp., 62 Alan Loop, Staten Island, NY 10304)
- Galitz, Wilbert O. 1994 *It's Time to Clean Your Windows: Designing GUIs That Work*, John Wiley & Sons, New York
- Galitz, Wilbert O. 1993 *User-interface Screen Design*, QED Publishing Group, Wellesley, MA,
- Green M 1986 A survey of three dialogue models. *ACM Transactions on Graphics* 5 (3) 244 - 275
- Hix D, Hartson R 1993 *Developing User Interfaces: Ensuring Usability through product and process*. Wiley [book approximately 380 pages]
- Hutt A T F, Flower F 1990 (May) *The User System Interface - a challenge for application users and application developers*. *ICL Technical Journal* 43 - 53
- Jacok R J K 1985 (August) *A State Transition Diagram Language for Visual Programming*. *IEEE Computer* 18 51 - 59
- Krug, Steve 2000 *Don't make me think*. New riders publishers
- IBM 1991 *Systems Application Architecture Common User Access Guide to User Interface Guide*, IBM Corp., Publication number SC34?4289
- Laurd, Brenda 1990 *The Art of Human-Computer Interface Design*, Addison Wesley Publishing Company
- Laurd, Brenda 1991 *Computers as Theatre*, Addison Wesley Publishing Company.
- Lewis, L. 1991 (August) "Cluster Analysis as a Technique to Guide Interface Design", *International Journal of Man-Made Studies*, 35:251?65
- Lim K Y 1996 *Structured task analysis: an instantiation of the MUSE method for usability engineering*. *Interacting with Computers* 8 (1) 31 - 50
- Luff P, Heath C, Greatbatch D 1992 *Tasks-in-Interaction: paper and screen based documentation in collaborative activity*. *Proc of CSCW 92 ACM* 163 - 170
- Marcus, Aaron 1992 *Graphic Design for Electric Documents and User Interfaces*, ACM Press/Addison-Wesley
- Microsoft 1992 *The Windows Interface: An Application Design Guide*, Microsoft Press
- Molich, R, Nielsen, J. 1990 (March) *Improving a Human-Computer Dialogue*", *Communications of the ACM*, 33:338?48
- Nickerson, R.S, Pew, R.W. 1990 (July) *Toward More Compatible Human Computer Interfaces* *WEE Spectrum*, 27:40-3
- Nielsen J, Tahir M 2001 *Homepage usability 50 websites deconstructed*. New riders Publishing
- Nielsen J 1999 *Designing Web Usability*. New Riders Publishing
- Nielsen J 1993 *Usability Engineering*. AP professional [by far the best of the Nielsen book listed here]
- Payne S J, Howes A 1992 *A task-action trace for exploratory learners*. *Behaviour & Information Technology* 2 (2) 63 - 70 [This article provides details of a method used to track how users interacted with a system which was then converted to task-action grammar (basically a narrative equivalent to state transition diagrams) to allow the system to be re-designed]
- Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S. & Carey, T. *Human-Computer Interaction*. Wokingham, UK: Addison-Wesley, 1994. (Excellent book very readable. 700 pages approx.)
- Rettig, M. 1992 *Interface Design When You Don't Know How*, *Communications of the ACM*, 35:29-34, January
- Schneiderman, Ben 1997 *Designing the User Interface*, 3rd Ed., Addison-Wesley Publishing Co., Reading, MA [This is the classic book on the subject and still used in many universities as the core text to an introductory course, unfortunately it is expensive. There is now also an accompanying web site: <http://www.awlonline.com/cseng/titles/0-201-69497-2/website/>]
- Sears A 1993 (July) *Layout Appropriateness: A metric for Evaluating User Interface Widget Layout*. *IEEE Transactions on Software Engineering* 19 (7) 707 - 719
- Smith H T, Green T R G 1980 *Human Interaction with computers*. Academic Press. [Book approximately 360 pages. Despite the age of this book it should not be dismissed much is still relevant]
- Smyth M, Clarke A 1990 (May) *Human-Human Co-operation and the Design of Co-operative Machines*. *ICL Technical Journal* 110 - 126
- Spencer, Charles D. 1990 *Digital Design for Computers Date Acquisition*, Cambridge University Press
- Thomas C G 1993 *Design, implementation and evaluation of an adaptive user interface*. *Knowledge-based Systems* 6 (4) 230 - 238 [An example of an adaptive user interface long before Windows 2000!]

Veen Jeffrey 2001 The art and science of web design. New riders

18. Links

Several of the links below are no longer active, however I have kept them in in the hope that this might be a temporary problem.

[useit.com: Jakob Nielsen's Website](#)

[Usable Web \(100s of links\)](#)

[HCI Resources Index](#)

[Human-Computer Interaction Resources on the Net](#)

[Donald Norman - HCI writer excellent](#)

[System Concepts Usability HCI](#)

[Systems concepts - Display Screen \(VDU\) Regulations and Ergonomics Standards For Procurement And Design](#)

[AskTog First Principles](#)

[Web-Based User Interface Evaluation with Questionnaires](#)

[eClass - example of a online community developing e-learning resources](#)

[Janet Finlay's Home Page](#)

[Article by Janet Finlay about genre and HCI](#)

[Fix, Finlay et al book - chapter 8 - overview](#)

[Alan Dix new home page \(Lancaster University\)](#)

[Micheal Smith - Napier University Edinburgh HCI course Introduction](#)

[Jennifer J. Preece home page](#)

[Visual demonstrations + Interesting Links Psychology](#)

[Useful sites Preece](#)

19. Exercise answers

Exercise 1

1. Donald Norman (1988) identified two principles that he considered would help toward good interface design

VISIBILITY (Controls need to be visible)

AFFORDANCE (Their design should suggest their functionality)

Norman, D.A. (1988) The Psychology of Everyday Things, Basic Books

Document details: c:\hicourseweb new\chap11\s10\hci_1.doc