Ekspsys, S2: Eksamen

Lasse Letager Hansen (201508114)

June 16, 2018

1 Mål / Eksamen

Deltagerne skal ved afslutning af kurset kunne:

- Beskrive og anvende teknikker til feltarbejde, konstruktion af scenarier, konstruktion og evaluering af prototype mm. i projektet med høj grad af brugersamarbejde
- Formulere og analysere et projekt med brugerinddragelse
- Planlægge, udføre og evaluere de tidligere dele at et systemudviklingsprojekt med brugerinddragelse
- Analysere (dele af) en organisation med henblik på udvikling af IT-støtte
- Diskutere og perspektivere de valgte løsninger, de problemer der opstår undervejs, og den valgte strategi som helhed.

Contents

2 Needs to know

- PACT
- Rich picture
- The Y model (Four processes of design)
- Scenario-based design method
- Scenarios
  - User stories
  - Conceptual Scenarios
  - Concrete Scenarios
  - Use cases
- Requirements and problems
- Scenario corpus
- Conceptual model
- Design language
- Future workshop
- Requirements
- Interviews
- Questionnaires
- Likert scales
- Probes
- Card sorting
- Focus groups
- Mock-up
- Sketch
- Snapshots
- Storyboards
- Mood boards
- Navigation maps
- Wireframes
- Metaphors
- Blends
- Contextual Design
- Contextual inquiry
- Contextual interviews
- Work modelling
  - Flow models
  - Sequence models
  - Artifact models
  - Cultural models
  - Physical models
- Affinity diagram
- User environment design (UED)
- Future laboratories.

Part I
Benyon et al 2014 (3. udg.)

3 Chapter 1, Essentials of designing interactive systems

- Understanding the concepts underlying design of interactive systems
- Understanding why being human-centered is important to design
- Understanding the historical background to the subject
- Understanding the skills and knowledge that the designer of interactive systems needs to draw upon.

3.1 The variety of interactive systems

The iPhone introduced pitching for zooming, and rotation by tilting the phone. The app store was an important step for the interaction. The newest model introduced a voice system called Siri.

Nintendo introduced the Wii, which used infra-red sensors to track a wand. The system could track gestures such as bowling, tennis or a host of other movements. When Wii Fit was introduced it appealed to a new audiance. In 2011 microsoft introduced the Kinnect system, that combined infra-red detection and cameras so the users could interactw with software using gestures without a wand.

Second life is an online world, consisting of thousands of simulated buildings, parks, seasides, factories, universities and everything else. People craeted avatars for the world. They can determine their size, shape, gender and what they want to wear. Other examples of highly populated virtual worlds are World of Warcraft and Sony Home environment.

The i Robo-Q domestic toy robot is an example of children's toys that are increasingly available. Toys are using all manner of new technologies to enchance the experiences of children at play. They use robotics, voice input and output, and a verity of sensors to provide novel and engaging interactions.

Facebook is a highly polular website that allows people to keep in contact with their friends. People can store and share digital photos, write notes to each other and get regular updates about what thier fiends are doing.

These examples capture many of the features that interactive systems designerns have to work with. The designer of interative systems need to understand the possiblities that exists for new forms of interaction.

3.2 The concerns of interactive systems design

The key concerns of the designer of interactive systems are:

- **Design**. What is design and how should you do it?
- **Technologies**. The interactive systems, products, devices and components.
- **People**. Who will use the systems and whose lives would we like to make better through our designs?
- **Activities** and **contexts**. What do peolpe want? What are the contexts within which those activities take place.

3.2.1 Design

Design is where you stand with a foot in two worlds, the world of technology and the world of peolpe and human purposes, and you try to bring it together – Mitch Kapor.

The term 'design' referes both to the creative process of specifying something new and to the representations that are produced during the process. One thing that is useful is to distinguish the amount of formality associated with a design:

- At one end of a spectrum is engineering design where scientific principles and technical specifications are employed to produce a model before cunstruction starts.
- At the other end of this spectrum is creative or artistic design where innovation, imagination and conceptual ideas are the key ingredients.
- Somewhere in the middle lies 'design as craft' which draws upon both engineering and creative approaches.

Most design involves asspects of all of these. Design works with and shapes, a medium; in our case this medium consists of interactive systems. Others emphasize that design is a conscious, social activity and that much design is often undertaken in a design team.

3.2.2 People and technologies

Interactive systems is intended to cover components, devices, products and software systems that are primarily concerned with processing information. Interactive systems are things that deal with the transmission, display and storage or transformation of information that people can perceive. They are devices and systems that respond dynamically to people's actions.

A fundamental challenge for interactive systems designers is to deal with the fact that people and interactive systems are different. Of course we take the people-centered view, but many designers still take the machine-centered view because it is quicker and easier for them, though not for the person who finishes up using the product. Another difference is that machines and people speak different languages. People express their desires and feelings in terms of what they want to do or how they would like things to be. Machines need to be given strict instructions.

3.2.3 The interface

The interface to an interactive system, the user interface (UI), is all those parts of the system with which people come into contact pysically, perceptually and conceptually. The interface needs to provide some mechanisms so that people can provide instructions and enter data into the system (Input). It also needs to provide some mechanisms for the system to tell people what is happening by providing feedback and mechanisms for displaying the content (Output). Designing interactive systems is not just a question of designing interfaces. The whole human-computer interaction needs to be considered, as does the human-human interaction that is often enabled through the systems. Interactive systems designers need to consider the whole environment they are creating.

3.2.4 Being human-centered

Interactive systems design is ultimately about creating interactive experiences for people. Being human-centered is about putting people first, and:

- Thinking about what people want to do rather than what the technology can do
- Designing new ways to connect people with people
- Involving people in the design process
- Designing for diversity

3.3 Being digital

We live in a digital age, when all manner of devices represents things using binary digits (bits). The significance of being digital is that bits are transformable, transmittable and storable using digital technologies. The huge range of interactions that we engage in and the interfaces that we use offer an exciting if daunting challenge. Moreover, increasingly disigners are having to deal with the issues of people engaged in multiple interactions with different devices in parallel.

3.3.1 How we got here

The first digital computers were huge machines housed in specially built, air-conditioned rooms. They were operated by scientists and specialist computer programmers and operators, who physically pressed switches and altered circuts. During the 1960s computer technology was still dominated by scientific and accounting applications. Data was stored on paper tape or cards with holes punched in them, on magnetic tapes and large magnetic disks, and there was little direct interaction with the computers.

The first screens and cathode ray tubes (CRTs) were being used as interactive devices and the first vision of a computer network – an internet – was formulated by Licklider. During the 1970s computing technology spread into buisnesses and screens linked to a central computer began to emerge. Computers were becoming networked together and indeed the first e-mail was sent over the ARPANET in 1972. As the decade ended so keyboards and screens became more common, but it was not until 1982 that the first real graphically based interfaces appered in the form of the Xerox star, Apple Lisa and Apple Macintosh computers. These used a bit-mapped display, allowing a graphical user interface (GUI) and interaction through pointing at icons and with commands

grouped into menues. This style became ubiquitous when, in 1985, the Windows operating system appered on personal computers.

The 1980s was the decade ofthe microcomputers, with the BBC Micro home computer selling over 1 million units and a whole plethora of home computers being adopted worldwide. It was during the 1980s that human-computer interaction (HCI) came of age as a subject. In both the USA and Europe the first big conferences on HCI were held.

In the 1990s color and multimedia arrived on the PC, which had begun to dominated the computer market. In 1993 a new interface was produced that took advantage of a simple mark-up or specification 'language' (HTML). Thus the 'World Wide Web' came about and revolutionized the whole process of transmitting and sharing files. Many parts of the world were not connected, but in the twenty-first century connections to the Web are global.

By turn of the century the convergance of communications and computing technologies was just about complete. Since all the data was digital, it could be transmitted over the airwaves or over wired networks, and it could easily be transformed from one form into another. Computing devices are now pervasive amongst people and across the world, providing all manner of services and experiences. Computing power continues to double every 18 months or so (Moore's law), producing mobile devices that are more powerful now that the largest computers where even just a few years ago. In the twnty-first century computing is truely ubiquitous and interaction is increasingly through touch and gestures rather than the keyboard. We now have Weiser's pads, tabls and boards in the form of phones and tablets in various sizes, large public screens and wearable computing.

3.3.2 Where are we heading

Making predictions about technology is hard. It is never just a technology that wins, but technology linked with a good business model linked with timing. Information appliances have the following set of characteristics:

- Appliances should be everyday things requiring only everyday skills to use
- Appliances have a clear focused function that can be used in a verity of circumstances
- Peer-to-peer interaction. A key idea of appliances is that they work together without the need for central control or uploading and downloading.
- Direct user interface. Appliances need to be simple and intuitive to use.
- Successful appliances are those which support the notion of the switch and simple completion of tasks.
- Appliances represent the ability to do something on impulse without haveing to think hard about how to do it.
- Appliances are personal and portable

Whether information appliances are just one of many directions that the future takes, we will have to see.

3.4 The skills of the interactive systems designer

Designers of interactive systems need a variety of skills, that allow them to be able to:

- Study and understand the activities and aspirations of people and the contexts within which some technology might prove useful and hence generate requirements for technologies.
- Know the possiblities offered by technologies.
- Research and design technological solutions that fit in with people, the activities they want to undertake and the contexts in which those activities occur.
- Evaluate alternative designs and iterate until a solution is arrived at.

It is often the case that no single person possesses all the skills needed for some design activity, which is why the design of interactive systems is often an affair for a design team. Designers of interactive systems cannot be experts in all these fields, but they must be aware enough to be able to take techniques from different areas, or access research in different disciplines when appropriate. We group subjects that contribute to the design of interactive systems under the headings of knowledge of people, technologies, activities and contexts, and design.

3.4.1 People

Sociology is the study of the relationships between people in society, the social, political and other groups that they participate in, and the settings in which such relationships take place. Anthropology is similar but focuses also on the study of culture, biology and language and on how these have evolved and changed over time. Both use techniques such as interviews and observation to arrive at their conclusions.

Psychology is the study of how people think, feel and act. In particular, congnitive psycology seeks to understand and describe how the brain functions, how language works and how we solve problems.

Ergonomics is the study of the fit between people and machines. In designing interactive systems, the designer will borrow much from each of these disciplines, including methods to help understand and design for people.

3.4.2 Technologies

The technologies that interactive systems designers need to know about include both software and hardware. Designers need to be aware of hardware for sensing different types of data and for bringing about some change. There are many different components available that produce many different effects and here designers will draw upon engineering knowledge, principles and methods. Communication between devices uses various communication 'protocols'. Designers need to know how different devices communicate.

3.4.3 Activities and contexts

Interaction will usually take place in the context of some 'community of practice'. This term is used to denote groups of people who have the shared interests and values and engage in similar activities. Social and organizational psychology are needed to look at the effects of technological change on organizations, and recently knowledge management and social computing have become important areas. Finally, new technologies offer new opportunities as buisness and interactive systems designers find that they are sometimes creating whole new wats of working with their designs.

3.4.4 Design

Ideas and philosophy from architechture, garden design, interior design, fashion and jewellery design all crop up in various ways and different forms. It is not easy to simply pick up ideas from design disciplines, as much design knowledge is specific to a genre. Designers need to know the materials they work with. Product design is an important contributing descipline to the skills of the designer of interactive systems. Graphic design and information design are particularly important for issues of information layout and the understandability and aesthetic experience of products. Human-computer interaction has itself evolved many techniques to ensure that designs are people-focused.

3.5 Why being human-centered is important

Being human-centered in design is expensive. It involves observing people, talking to people and trying ideas out with people, and all this takes time. But taking the human-centered approach to design of interactive systems is advantageous for a number of reasons.

3.5.1 Return of investment

Williams et al. provide details of a number of case studies looking at the cost of taking a human-centered approach to interactive systems design and at the benefits that araise. Paying attention to the needs of people, to the usability of the product, results in reduced calls to customer helplines, fewer training materials, increased throughput, increased sales and so on.

3.5.2 Safety

In the early 1980s there was an accident at a nuclear power plant at Three Mile Island in the USA that almost resulted in a 'meltdown'. Reportedly one of the problems was that the control panel indicated that a valve was closed when it was not, and another indicater was obscured by a tag attached to another control: two fundamental design errors that human-centered design techniques would help to avoid.

3.5.3 Ethics

Being human-centered also ensures that designers are truthful and open in their design practice. Now that it is so easy to collect data surreptitiously and to use that data for purposes other than what it was intended for, designers need to be ever more vigilant. People need to trust systems and be in a position to make choices about privacy and how they are represented. The issue of intellectual property is another important aspect of ethical design; it is very easy to take an image

from a website and use it without giving proper acknowledgement for its source. Privacy, security, control and honesty are all significant features of the interactive systems designer's life. Equality and attention to access are two of the 'political' issues that designers must address. As technology changes so do traditional views and approaches to big moral and ethical questions.

3.5.4 Sustainability

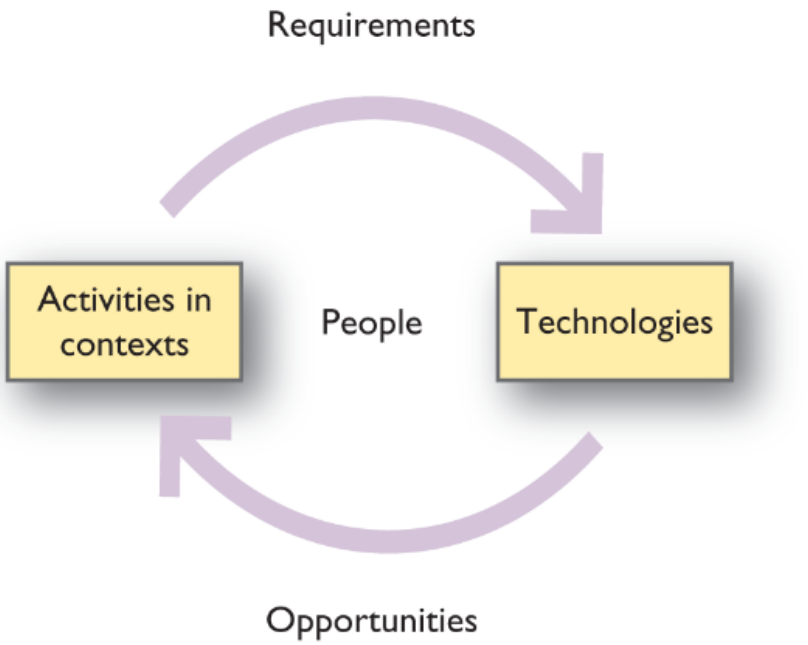
Interactive systems have a big impact on the whole world, and designers should approach interaction design from the perspective of what is sustainable. Millions of mobile phones and other devices are thrown away each year and they contain metals that are potentially dangerous to the environment. Large displays gobble up power. Cultures get swamped by the views and values of the dominant suppliers of hardware and software and local languages die out when all information is in English, Chinese or Hindi. Human-centered design needs to recognize diversity and design to enhance human values.

4 Chapter 2, PACT: a framework for designing interactive systems

- Understand the relationship between activities and technologies
- Understand the PACT framework
- Understand the main characteristics of people that are relevant to designing interactive systems
- Understand the main issues of activities and the contexts in which they occur
- Understand the key features of interactive technologies

4.1 Introduction

Technologies are there to support a wide range of people undertaking various activities in different contexts. If the technology is changed then the nature of the activities will also change. This issue is nicely summed up by:



Designers need to keep this cycle in mind as they attempt to understand and design for some domain.

4.2 People

There can be few less controversial observations than that people differ from one another in a variety of ways.

4.2.1 Physical differences

People differ in physical characteristics such as height and weight. Variability in the five senses – sight, hearing, touch, smell and taste – has a huge effect on how accessible, usable and enjoyable using a technology will be for people in different contexts. In Europe there are 2.8 million wheelchair users, so designers must consider where technologies are placed; and many people have dexterity impairment involving the use of their fingers.

4.2.2 Ergonomics

The term 'ergonomics' was coined in 1948 to describe the study of the relationships between people and their environment. The environment includes the ambient environment (temperature, humidity, atmospheric pressure, light levels, noise and so on). and the working environment too (the design of machines, health and safety issues – e.g hygine, toxicology, exposure to ionizing radiation, microwaves, etc.). Ergonomics is multidisciplinary, drawing on anatomy and physiology, various aspects of psycology, physics, engineering and work studies among others.

Ergonomics can put numbers on what constitutes samll and usable, and what is too small and

unusable. The best-know example of ergonomic knowledge being applied to HCI issues is Fitt's law.

4.2.3 Psychological differences

Psychologically, people differ in a variety of ways. For example, people with good spatial ability will find it much easier to find their way around and remember a website than those with poor ability. Designers should design for people with poor ability by providing good signage and clear directions. Languages differences are of course crucial to understanding and cultural differences affect how people interpret things. People also have different needs and abilities when it comes to attention and memory and these can change depending on factors such as stress and tiredness. All people are better at recognizing things than they are at remembering things. Some people can quickly grasp how somethings works, wheres for others it can take much longer. People have had different experiences and so will have different conceptual 'models' of things.

4.2.4 Mental models

The understanding and knowledge that we posses of something is often referred to as a 'mental model'. A key design principle is to design things so that people will form correct useful mental models of how they work and what they do. People develop mental models through interaction with systems, observing the relationship between their actions and the behaviors of the system and reading any manuals or other forms of explanation that come with a system. So it is important that designers provide sufficient information in the interface for people to form an accurate mental model. In systems of any large size, no single designer will know everything that the system does. Designers design a system's image that they hope will reveal the designers' conception. Normal has made the following general observations about the nature of mental models of interactive systems:

- Mental models are incomplete. People will understand some parts of a system better than others.
- People can 'run' their models when required, but often with limited accuracy.
- Mental models are unstable – people forget details.
- Mental models do not have firm boundaries: similar devices and operations get confused with one another.
- Mental models are unscientific exhibiting 'superstitious' behavior.
- Mental models are parsimonious. People are willing to undertake additional physical operations to minimize mental effort, e.g. people will switch off the device and start again rather than trying to recover from an error.

Mental models can support reasoning about devices, or the physical world in general, by running simulations in the mind's eye.

4.2.5 Social differences

Novice and expert users of a technology will typically have very different levels of knowledge and hence requirements for design features. Experts use a system regularly and learn all sorts of details, whereas a begninner will need to be guided through an interaction. Therre are also people who do not have to use a system, but who the designer would like to use the system. Designing for homogeneous groups of people is quite different from designing for heterogeneous groups.

4.3 Activities

First and foremost, the designer should focus on the overall purpose of the activity. After that the main features are:

- Temporal aspects
- Cooperation
- Complexity
- Safety-critical
- The nature of the content

Temporal aspects
Temporal aspects cover how regular or infrequently activities are. People will soon learn how to make calls using a mobile phone, but may have great difficulties when it comes to changing the battery. Designingers should ensure that frequent tasks are easy to do, and infrequient tasks are easy to learn (or remember) how to do.

Other important features of activities include time pressures, peaks and throughs of working.

Some activities will take place as a single, continuous set of actions whereas others are more likely to be interrupted. If people are interrupted, the design needs to ensure that they can 'find their place' again and pick up.

The response time needed from the system must be considered. As a gerenal rule people expect a response time of about 100 miliseconds for hand-eye coordination activities and one second for a cause-effect relationship. Anything more than five seconds and they will feel frustrated and confused.

Cooperation
Another imporatnt feature of activities is whether they can be carried out alone or whether they are essentially concerned with working with others. Issues of awareness of others and communication and coordination then become important.

Complexity
If a task or activity is well defined it can be accomplished with a simple step-by-step design. A vague activity means that people have to be able to browse around, see different types of information and so on.

Safety-critical
Some activites are 'safety-critical', in which case any mistake could result in an injury or a serious accident. Others are less so. Clearly, where safety is involved designers must pay attention to ensuring that mistakes do not have a serious effect.

In general it is vital for designers to think about what happens when people make mistakes and errors and to design for such circumstances.

The nature of the content
If large amounts of alphabetic data have to be input as part of the activity then a keyboard is almost certainly needed. In other activities there may be a need to display video or large quality color graphic displays.

Just as important as data is the media that an activity requires. A simple two-tone display of numeric data demands a very different design from a full-motion multi-media display.

4.4 Contexts

Activities always happen in a context, so there is a need to analyse the two together. Three useful types of context are distinguishable: the organizational context, the social context and the physical circumstances under which the activity take place. Sometimes it is useful to see context as surrounding an activity. At other times it can be seen as the features that glue some activities together into a coherent whole.

4.4.1 Physical environment

The environment may be very noisy, cold, wet or dirty. The same activity may be carried out in geographically remote environments.

4.4.2 Social context

A supportive environment will offer plenty of help for the activity. There may be training manuals available, tuition or experts to hand if people get into truble. There may be privacy issues to consider, and an interaction can be very different if the person is alone compared to being with others. Social norms may dictate the acceptability of certain designs.

4.4.3 Organized context

Changes in technology often alter communication and power structures and may have effects on jobs such as deskilling. The circumstances under which activities happen (time, place and so on) also vary widely and need to be taken into consideration.

4.5 Technologies

Interactive systems typically consists of hardware and software components that communicate with one another and transform some input data into some output data. People using such systems engage in interactions and physically devices have various degrees of style and aesthetics. Designers

of insteractive systems need to understand the materials they work with. Of course, interactive technologies change at a fantastic rate. It is also very difficult to classify technologies as they are continueally being packaged in new ways and different combinations facilitate quite different types of interactions.

4.5.1 Input

Input devices are concerned with how people enter data and instructions into a system securely and safely. Switches and buttons facilitate a simple and direct method of issueing instructions but they take up space. On small mobile devices there is not enough room to have many buttons, so designers have to be careful which functions have their own buttons. Alphanumerical data is usually input to an interactive device through a 'QWERTY' keyboard. The design of the 'QWERTY' keyboard is still with us today, despite some devices using an alphabetic keyboard where the letters are arranged in alphabetic order. Touchscreens are sensitive to the touch of a finger. They function through either infrared sensitivity or electrical capacitance. Because of their lack of moving or detachable parts, they are suitable for applications intended for public places, and provided the interface is well designed they present an apparance of simplicity and ease of use. Touchscreens make use if the person's finger as the input device, which has the obvious benefit that people always have their finger with them. Light pens was arguably the original pointing device. When it is pointed at the screen it returns information about the screen location to a computer. Other forms of pointing device include the stylus which is used on very small displays. Being more precise than a finger, a stylus can be used for handwriting recognition. Writing with a stylus directly onto a computer's screen or tablet is a natural way of working. However, it is quite slow and can be inaccurate. It requires people to 'train' the device to recognize their handwriting, which improves the recognition accuracy of the software. Many people can type faster than they can write by hand.

One of the most ubiquitous of input devices is the mouse. The mouse consists of a palm-sized device that is moved over a flat surface. One or two buttons sit on top of the mouse and are operated with the person's fingers. The mouse has become the default pointing device. More contemporary mouse design include a thumbwheel for scrolling through documents or Web pages. A mouse may be cordless. In 2009 Apple introduced the 'magic mouse' that combined traditional mouse functions with multi-touch capability allowing a range of new touch gestures for interaction. A trackball is another pointing device, which is best described as a mouse lying on its back. To move the pointer the user moves the ball. Again, like all other pointing devices, there are one or more buttons which can be used to select on-screen items. A joystick is a handle which pivots from a central point. Viewing the joystick from above, it may be moved north, south, east and west to control an on-screen pointer. Joysticks are used mostly for computer games, but thay are also found in conjuction with CAD/CAM (computer-aided design / manufacture) systems and VR applications.

With the introduction of the Nintendo Wii a whole ne wgeneration of input became possible. The Wii uses infra-red to register the movement of a wand. This allows gestures to be recognized. Other systems, notably the Microsoft Kinect, recognize recognize gestures through tracking limb and body movements by attachning sensors to the limb or by tracking using cameras.

Air pressure sensors, acoustic sensors vibration detectors, infra-red motion detectors and accelerom-

eters are all readily available for designers to detect specific aspects of an interaction. Speech input is becomming increasingly accurate, particularly if people are willing to spend a few minutes training a system to recognize their voice. Even without training, the Siri system on the iPhone can be quite impressive. Other forms of input include quick response (QR) codes and augmented-reality (AR) fiducial markers. Fiducial markers are used to recognize an object and hence to tailor some interaction towards. Markerless AR uses a photo of an object to register a connection allowing graphics, video and other content to be overlaid onto the scene.

4.5.2 Output

Technologies for displaying content to people rely primarily on the three perceptual abilities of vision, hearing and touch. The most fundamental output device is the screen. Flexible organic light-emitting diode (OLED) displays for screens are just coming onto the market that will enable displays of any shape and size that can bend and hence can be used in clothing. The physical dimensions of display devices are, however, only one of the factors involved in the resulting output. One way past the problems with restrictive display 'real estate' is to use a data projector. While the resolution is usually less than that of a monitor, the resulting projected image can be huge. Besides the visual display of content, sound is an important method of output. Sound is an output medium that is significantly under-used. Speech output is also an increasingly popular option. With effective text-to-speech (TTS) systems, simply sending a text message to the system results in clear spoken output.

A printer is a device that prints text or illustration on paper, while plotter draws pictures or shapes. Plotters differ from printers in that they draw lines using a pen. As a result they can produce continuoues lines, whereas printers can only simulate lines by printing a closely spaced series of dots. Three-dimensional printers work by placing layers of a powdery material on top of each other to create a real-life model of a digital image. It is thought that with the use of hundreds and perhaps thousands of layers everything from 'coffee cups to car parts' could be created. These printers allow for rapid prototyping of physical designs for new products.

'Haptics' refers to the sense of touch. Parhaps the most widespread haptic devices are those games controllers that incorporate so-called force-feedback. The benefits of force-feedback devices are:

- Sensations can be associated with interactions, suchas feeling driving surfaces or feeling footsteps
- Sensations can also be used to provide feedback as to the location of other players, objects and so forth.
- Force-feedback can allow the player to feel what it would be like towield a sword, drive a high-speed car, fly a 'speeder' or engage the Empire with a light-sabre.

Other examples include the 'silent alert' vibration of a mobile phone and even the feel of a key when pressed.

4.5.3 Communication

Communication can take place through wired connections such as a telephone line or an Ethernet newtork. Ethernet is the fasest form of communication, but the device has to be plugged into a network to make use of it. Wireless communication is becomming much more common and often a wireless 'hub' is attached to an Ethernet network. Wireless communication can take place over the wireless telephone network used for mobile phones or over a Wi-Fi connection. Wi-Fi is quite limited in range. The newer 4G technologies promise to deliver much faster connectivity over mobile devices. Other forms of wireless communications continue to be developed. Near-field communication (NFC) is used to connect devices simply by bringing the close to each other. All new mobile phones will soon have NFC capability.

4.5.4 Content

Content concerns the data in the system and the form it takes. Good content is accurate, up to date, relevant and well presented. Therer is little point to having sophisticated information retrieval system if the information, once retrieved, is out of date or irrelevant. Most technologies have a mixture of function and content. Content can be retrieved when required or it can be pushed from a server to a device. RSS feeds on websites provide automatic updates when a website's content is changed. The characteristics of the data is imporatnt for choosing input methods. Barcodes, for example, are only sensible if the data does not change often. Touchscreens are useful if there are only a few options to choose from. Speech input is possible if there is no noise or background interference, if there are only a few commands that need to be entered or if the domain is quite constrained. 'Streamy' outputs such as video, music and speech have different characteristics from 'chunky' media such as icons, text or still photographs. Most imporatnt, perhaps, is that streamy media do not stay arround for long.

4.6 Scoping a problem with PACT

To do a PACT analysis that designer simply scopes out the variety of Ps, As, Cs and Ts that are possible, or likely, in a domain. This can be done by brainstorming and other envisionment techniques and by working with people through observations, interviews and workshops. A PACT analysis is also useful for developing personas and scenarios. The designer should look for trade-offs between combinations of PACT and think about how these might affect design.

5 Chapter 3, The process of human-centered interactive systems design

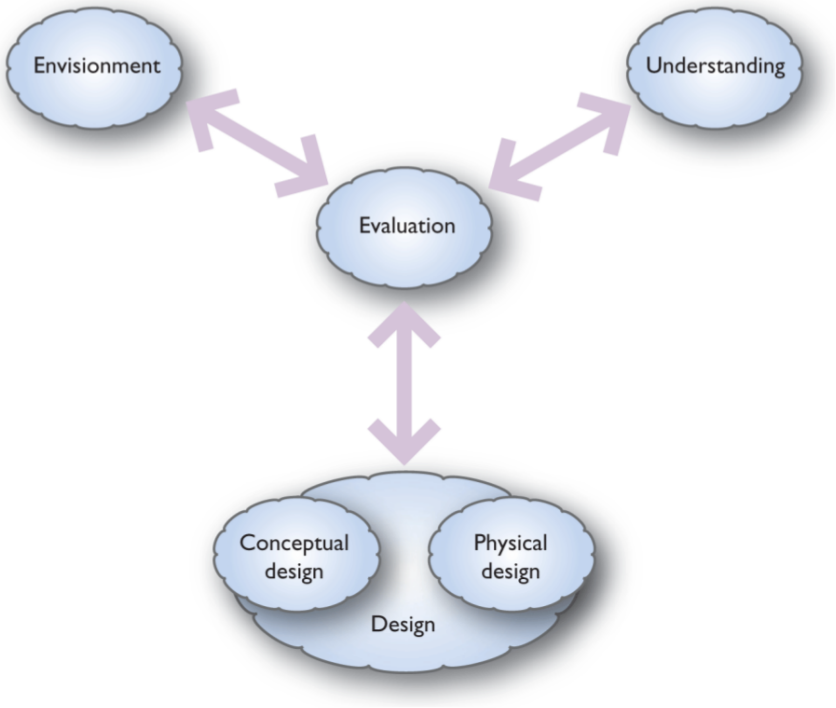
- Understand the nature of interactive systems design
- Understand the four processes involved in design: understanding design, envisionment, evaluation
- Understand the centrality of evaluation in human-centered design
- Understand the scenario-based design approach
- Develop scenarios and personas

- Understand the scenario-based design method

5.1 Introduction

Design is messy; designers try to understand the mess. Design is about users and use. The key features of the four activities representation of the overall design process are as follows:

- Evaluation is central to designing interactive systems. Everything gets evaluated at every step of the process
- The process can start at any point – sometimes there is a conceptual design in place, sometimes we start with a prototype, sometimes we start with understanding.
- The activities can happen in any order, for example understanding might be evaluated and a prototype built and evaluated and some aspect of a physical design might then be identified.



5.1.1 Understanding

Therer are both functional and non-functional requirements to consider.Functional requirements are concerned with what the system should be able to do and with the functional constraints of the system. There are also logical and organization constraints that may make particular designs infeasible.

Requirements are generated through discussions and interaction with people, and also through observations of existing systems. The aim is to collect and analyse the stories people have to tell. Requirements are essentially about understanding.

5.1.2 Design

Design activites concern both conceptual design and physical design. Conceptual design is about designing a system in the abstract, physical design is concerned with making things concrete.

Conceptual design
Conceptual desing is about considdering what information and functions are needed for the system to archieve its purpose. There are a number of techniques to help with conceptual design. Software engineers prefer Modelling possible solutions with objects, relationships and 'use cases'. Entity-relationship models are another popular model. Flow can be represented using dataflow diagrams and structure can be shown with structure charts.

One way to conceptualize the main features of a system is to use a 'rich picture'. A rich picture captures the main conceptual relationships between the main conceptual entities in a system. Most imporatntly, the rich picture identifies the issues or concerns of the stakeholders, thus helping to focus attention on problems or potential design solutions. The key feature of conceptual design is to keep things abstract – focus on 'what' rather than 'how' – and to avoid making assumptions about how functions and information will be distributed.

Physical design
Physical design is concerned with how things are going to work and with detailing the look and feel of the product. Physical design is about structuring interactions into logical sequences and about clarifying and presenting the allocation of functions and knowledge between people and devices. There are three components to physical design: operational design, representational design and design of interactions.

Operational design is concerned with specifying how everything works and how content is structured and stored. Taking a functional view of an activity means focusing on processes and on the movement, or flow, of things through the system. Events are occurrences that cause, or trigger, some other functions to be undertaken.

Representational design is concerned with fixing on colors, shapes, sizes and information layout. It is concerned with the style and aesthetics and is particularly important for issues such as the attitudes and feelings of people, but also for the efficient retrieval of information.

Interaction design, in this context, is concerned with the allocation of functions to human agency or to technology and with the structuring and sequencing of the interactions.

5.1.3 Envisionment

Designs need to be visualized both to help designers clearify their own ideas and to enable people to evaluate them. Envisonment is concerned with finding appropriate media in which to render design ideas. The medium needs to be appropriate for the stage of the process, the audience, the resources available and the questions that the designer is trying to answer. Some methods are: Sketches, fully functioning prototypes, cardboard mock-ups, scenarios, storyboards.

5.1.4 Evaluation

Evaluation is tightly coupled with envisionment because the nature of the representation used will affect what can be evaluated. Techniques for evaluation are many and various, depending once again on the circumstances. The important thing to keep in mind is that the technique used must be appropriate for the nature of the representation, the questions being asked and the people involved in the evaluation.

5.1.5 Implementation

If interactive systems designers were architects they would have well-understood methods and conventions for specifying the results of the design process. They would produce various blueprints from different elevations and engineering specifications for particular aspects of the design. In interactive systems design there are a varity of formal semi-formal and informal methods of specification. The best know formal methods is Unified Modeling Language (UML).

5.2 Developing personas and scenarios

The people who will use the system are represented by personas: profiles of different types, or archetypes, of people that designers is designing for. Activities and the contexts in which they will occur are envisioned through scenarios of use. Personas and scenarios are developed through the understanding process, using any of a wide range of methods, and through undertacking a PACT analysis. Almost inevitable, personas and scenarios evolve together as thinking about people involves thinking about what they want to do, and thinking about activities involves thinking about who will be undertaking them.

5.2.1 Personas

Personas are concrete representation of the different types of people that the system is being designed for. Personas should have a name, some background and importantly, some goals and aspirations. Designers need to recognize that they are not designing for themselves. Designers create personas so that they can envisage whom they are designing for. They create personas so that they can put themselves in other people's shoes.

5.2.2 Scenarios

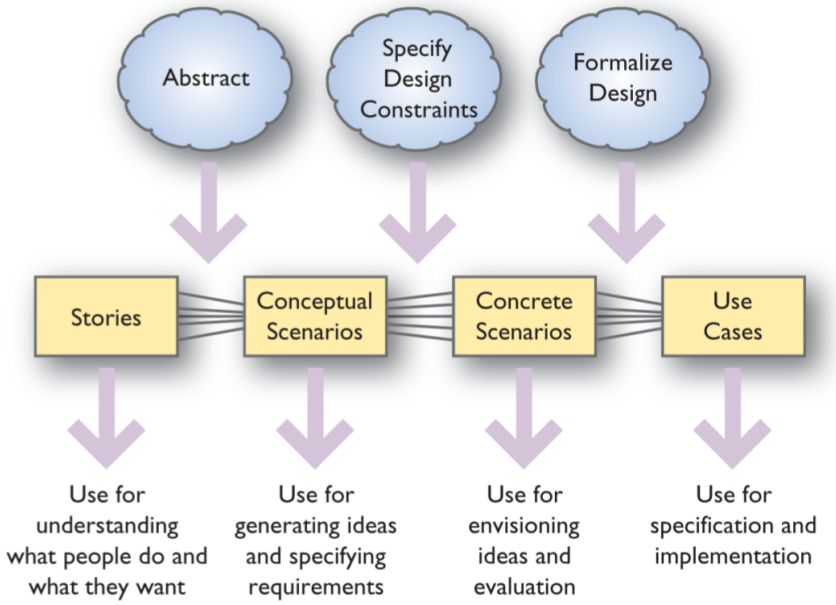
Scenarios are stories about people undertacking activities in contexts using technologies. They appear in a varity of forms throughout interactive systems design and are a key component of many approaches to design. Scenarios are effective at dealing with five key problems of design:

- The external factors that constrain desing such as time constraints, lack of resources, having to fit in with existing design and so on.
- Design moves have many effects and create many possibilities, that is a single design decision can have an impact in many different areas and these need to be explored and evaluated.
- Scientific knowledge lags design application. In interactive systems design the technology changes as soon as, or even before general solutions have been discovered.

- Action vs Reflection. The importance of reflection and action in design.
- The slippery nature of design problems.

5.3 Using scenarios throughout design

Scenarios (and their associated personas) are a core technique for interactive systems design. They are useful in understanding, envisioning, evaluation and both conceptual and physical design: the four key stages of interactive systems design. There are four different types of scenario: stories, conceptual scenarios, concrete scenarios and use cases. Stories are real-world experiences of people. Conceptual scenarios are generated from abstract scenarios by adding specific design decisions and technologies. Once completed they can be represented as use cases. Use cases are formal descriptions that can be given to programmers. Scenarios are helpful in understanding current practice and any problems or difficulties that people may be having in generating and testing ideas, in documenting and communicating ideas to others and in evaluating design.



5.3.1 Stories

Stories are real-world experiences, ideas, anecdotes and knowledge of people.This could include videos of people engaged in activity, diary entries, photographs, documents, the results of observations and interviews and so on.Stories also capture many seemingly trivial details that are usually left out if people are asked.

5.3.2 Conceptual scenarios

Conceptual scenarios are more abstract than stories. Much of the context is stripped away, and similar stories are combined. Conceptual scenarios are particularly useful for generating design ideas and for understanding the requirements of the system. At this stage, there is little or no specification of precise technologies or how the functions will be provided.

5.3.3 Concrete scenarios

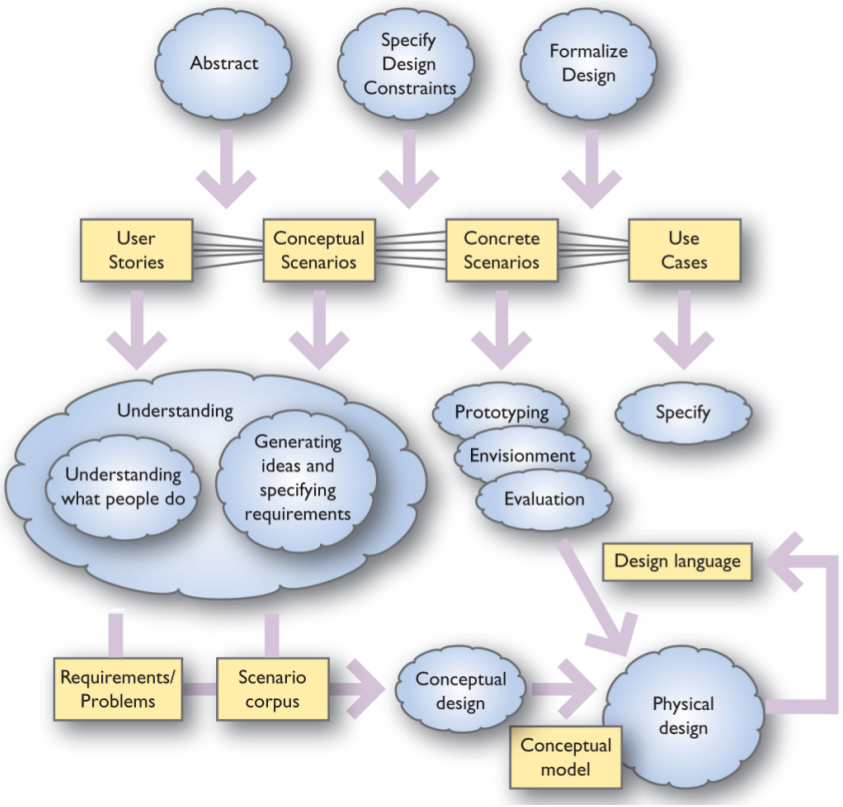
Each conceptual scenario may generate lots of concrete scenarios. When designers are working on a particular problem or issue they will often identify some feature that applies only under certain circumstances. At this point they may develop a more specific elaboration of the scenario and link it to the original. Concrete scenarios also begin to dictate a particular interface design and a particular allocation of functions between people and devices. Concrete scenarios are particularly useful for prototyping and envisioning design ideas and for evaluation because they are more prescriptive about some aspects of the technology.

5.3.4 Use cases

A use case describes the interaction between people and devices. It is a case of how the system is used and hence needs to describe what people do and what the system does. Before use cases can be specified, tasks and functions have to be allocated to humans or to the device. The specification of use cases both informs and is informed by the task/function allocation process. A set of use cases can be produced which specifies the complete functionality of the system and the interactions that will occur. There are a number of different ways of representing use cases – from very abstract diagrams to detailed 'pseudo code'.

5.4 A scenario-based design method

The use of the different types of scenario throughtout design can be formalized into a scenario-based design method:



Products of the design process shown as boxes and processes shown as clouds. Besides the four different types of scenario, four other artifacts are produced during the design process: requirements / problems, scenario corpus, object model and design language. The specification of a system is the combination of all the different products produced during the development process.

Requirements and problems
In the gathering of people's stories and during the analysis and abstraction process, various issues and difficulties will come to light. These help the analyst / designer to establish a list of requirements – qualities or functions that any new product or system should have. The requirements and problems product is a prioritized list of issues that the system to be designed needs to accomodate.

Scenario corpus
We seek to develop a representative and carefully thought-through set, or corpus, of scenarios. Having undertaken some analysis activities designers will have gathered a wide range of user stories. Some of these will be very general and some will be quite specific. The relation for the development of a corpus of scenarios is to uncover the 'dimensions' of the design situation and to demonstrate different asspects of those dimensions. Dimensions include characteristics of various domains within which the product will operate, the various media and data types that need to be accomodated and the characteristics of the people who will be using the system. The corpus of scenarios need to cover all the main functions of the system and the events that trigger the functions. The dimensions include different types of content and how that can be structured, issues of style and aesthetics.

The aim is to specify the scenarios at a level of abstraction that captures an appropriate level of generality that will be useful across the range of characteristics that is demonstrated within a domain.

Conceptual model
An object or data model results from the process of cenceptual modelling, including developing the scenarios and understanding the objects and actions that are evident fromthe analysis of the scenario corpus. The conceptual model shows the main objects in the system, their attributes and the relationships that exist between them. The conceptual model will also form the basis of the information architecture of a system and of any metaphor that is used in the design.

5.4.1 Design language

The design language produced consists of a set of standard patterns of interaction and all the physical attributes of a design – the colors, shapes, icons and so on. These are brought together with the conceptual actions and objects, and the 'look and feel' of the design is completed. A consistent design language means that people need learn only a limited number of design elements and then they can cope with a large variety of different situations.

Documenting scenarios
We use the PACT framework to critique scenarios and to encourage designers to get a good description of the scenario. Each scenario should be given an introduction. The history and authorship can be recorded, along with a description of how the scenario generalizes and the rationale for the

scenario.

Another key feature of writing scenarios is to think hard about the assumptions that are being made. The use of personas can help to focus on specific issues. Finally, with scenarios it is important to provide a very rich context. The guiding principles for scenario writing are people, activities, contexts and technologies.

6 Chapter 6, The home information Centre (HIC): a case study in designing interactive systems

- Understand how to undertake an interaction design project
- Discuss how design decisions are made in an interactive design project
- Understand the trade-offs that are inherent in such a project
- Appreciate the central role of evaluation in design

6.1 Introduction

The Home Information Centre, HIC, is proposed. This should be a device where people can get at and provide information while they are occupied with other household activities. The industrial partner characterized this as 'a move-around situation for infotainment'.

The abstract concept of an HIC as a device to deal with 'a move-around situation for infotainment' was initially translated into a number of high-level features and functions. Two key features were required: an intuitiv navigation support system and a flexible query system. The software should provide the following:

- An abstract representation of the content of information sources that should be extracted and maintained semi-automatically
- Speech, pen, touch and keyboard as input
- Sound, images, text and animation as output
- Speech recognition (SR)
- Natural language (NL) queries
- An intuitive user interface

The industrial partners on the project imposed a key constraint on the project: the HIC should not look or behave like a PC.

6.2 Scenarios for the HIC

Conceptual scenarios are abstract descriptions of the PACT elements in some domain. For the HIC design, there were three general, highly abstract uses for the HIC – informational, communicational and entertainment. The communication centered on how the HIC meight be used as a sorf of 'Post-it' system. The HIC would also be used as a video phone and e-mail system, and for social networking. The informational scenarios concerned activities such as finding a recipe. The HIC would be able to calculate the quantities in a recipe for one, two or three people. Other scenarios concerned activities such as going sailing, hence the need to get information on the tides, going skiing, going out, etc. The entertainment scenario led to ideas that users might like to play a game, find some sport, watch TV on demand or automatically record preferred programmes, and so on. Other scenarios include integration with other devices, such as heating control, home security and so on. Children doing homework was another, investigating, finding out, doing crosswords and quizzes, ordering videos, photographs and surfing the Web. Often the issue came back – how to accommodate large amounts of data, large visualizations, and the multitude of media that would be needed.

In developing the various, more concrete versions of the abstract activities, the awareness of the need from the PACT analysis to cover the different people – children, people with disabilities, as well as the young and wealthy who would be the early adopters of such technology. Eleven scenarios constituted the final scenario corpus that was used in the project:

- What shall we have for dinner?
- What shall we do now?
- News and e-books
- Entertain me
- Message board
- Traffic
- New users
- Housekeeping
- Payment
- I cannot get my phone to work because ...
- Planning ahead.

The aim was to explore the concept of the HIC first before settling on the physical design. The scenario corpus constitued the first prototype, P0.

6.2.1 A future workshop

One important thing about developing prototypes is using them. The scenario corpus was used internally within the project team to generate ideas and discuss the details of functionality. The scenarios were also taken outside the team and used as the basis of 'future workshops'. Future workshops are one of the design methods advocated in the participating design approach to the design of interactive systems. A future workshop comprises three stages:

- **Critique**. This is a group brainstorming session that tries different approaches to the activities/problems being addressed by the proposed system. A set of themes arising out of this stage is used in stage 2.
- **Fantasy**. The emphasis here is on creative solutions/ideas, no matter how unrealistic. From these a number of fantasy themes are generated and these are then used to drive stage 3.
- **Implementation**. The group work out what would be necessary in order to produce a real solution to some of the fantasy themes generated (eg. a mock-up or storyboard)

This concrete future workshop resulted in the following. The groups focused on the need for the HIC to be personalized. They wanted mobility in the system and they emphasized the communication function over the informational function of the HIC.

6.2.2 A more concrete scenario

At this stage it was important to pick one of these scenarios to orient various project teams around. The scenario choosen was the 'What shall we do now?' scenario, since it would have data for most teams to do work. The remaining scenarios would enable the investigation of other data types, use contexts, activity types and input/output devices not covered in the first scenario.

6.2.3 Developing the scenario

At one of the workshops, one group took the 'What shall we do now?' scenario and discussed it in more detail. The discussion at the workshop switched rapidly between high-level concerns and the details of a concrete interaction. This also resulted in the development of more detailed personas.

6.3 Evaluating early interface prototypes

In order to evaluate some interface concepts, three prototype design solutions, each taking one of the scenarios as a starting point, were produced.

6.3.1 Usability principles

- **Leanability**. Can people guess easily what the system will do, based upon previous experiences? This covers the usability principles of visibility, consistency, affordance and familiarity.
- **Effectiveness**. Can people correct or change previous descisions, and complete their desired task easily, effectively and safely? This covers the usability principles of navigation, control, feedback, recovery and constraints.

- **Accommodation**. Is the system designed to afford a multiplicity of ways in which people can accomplish their goals? Is it pleasent to use? This covers the usability principles of flexibility, style and conviviality.

The prototype defines multiple levels of information available through the use of a color-coded history top bar. It also provides a solution for the many actions and operations to be presented on screen only when required by means of hidden docks. It deals with the issues involved with utilizing the HIC from a distance in connection with speech input or remote control with the implemenation of 'Near' and 'Far' buttons which magnify the contents on the screen.

6.3.2 Conclusions

Although not perfect, the design gave a good starting point for providing an effective solution to the many problems that the HIC poses. In particular a history bar was an effective idea, but the color coordination, representation of the bar with no history or huge amounts of history and representation of the bar when people are moving backward through the information all needed to be looked at further.

6.4 ...

6.5 Summary of key points

- How scenarios were developed to explore the design space of the HIC device
- How the scenarios weree analysed in order to understand the main functionality the device required
- How the key design concepts were developed through prototyping ideas, evaluating them and redesigning them
- How key design concepts were realized physically and how the physical design affected the conceptual design and vica versa
- How a physical design was evaluated focusing on some key aspects of the design.

7 Chapter 7, Understanding

- Understand what requirements are
- Understand the range of requirements generation techniques
- Use techniques for understanding people and their activities in context
- Domcument the results as requirements on interactive technologies and services

7.1 Understanding requirements

A requirement is 'something the product must do or a quality that the product must have'. The task now is to turn data studies of activities and gatherings of stories of use, into requirements for new a products, system or service. Just to further complicate matters, additional requirements will emerge as the design process continues.

Requirements specifications increasingly include prototypes, screenshots and other media. When written they should be expressed in clear, unambiguous language, and worded so that it will be possible to test whether the requirements has been met in the final system. Conventionally, requirements are divided into two types, functional and non-functional. Functional and non-functional. Functional requirements are what the system must do. Non-functional requirements are a quality the system must have. Non-functional requirements cover a number of aspects of design, including image and aesthetics, usability, performance, maintainability, security, cultural acceptability and legal restrictions. For both types of requirements, note that how the technology will meet the requirements is not specified.

7.1.1 Prioritizing requirements

Requirements should be reviewed with customers and clients and modified as necessary. One way of doing this is by using the '**MoSCoW rules**'. These classify requirements into:

- **M** ust have – fundamental requirements without which the system will be unworkable and useless, effectively the minimum usable subset.
- **S** hould have – would be essential if more time were available, but the system will be useful and usable without them.
- **C** ould have – of lesser importance, therefore can more easily be left out of the current development
- **W** ant to have but **W** on't have this time round – can wait till a later development

The MoSCoW method is part of the Atern development method. The method takes a very business-focused view of prioritizing requirements, tying in the specification of priorities with the overall business costs of developing a system.

7.2 Participative design

Research work involves using a variety of techniques to understand and analyse someone else's needs, goals and aspirations. Designers need to understand the requirements of other people. This is not easy, but talking to people using interviews, observing people and recording their activities on video, organizing focus groups, workshops, etc. will all help the designer to understand both the requirements for the new design and the problems people are having with existing ways of doing things.

First, it is important that humans characteristics and activities are taken into accout. But beyond this, wherever possible, it is right that people who will use new interactive technologies have

an input to the design process itself. The situation is very different from custom-made systems for a small group of people, where it is genuinely feasible for the people concerned to act as co-designers and so acquire ownership of the technology to be introduced.

7.3 Interviews

One of the most effective ways of finding out what people want and what problems they have at the moment is to talk to them! The structured interview uses questions that are developed beforehand. The interview follows the wording exactly. Structured interviews are reasonably easy to carry out, simply because of the degree of pre-structing. However, people are limited to very restricted replies, and it is difficult for the interviewer to follow up the unexpected response. Designers very frequently use semi-structured interviews. Sometimes, the interviewer is armed with pre-prepared questions, but can reword these as appropriate and explore new topics as they arise. Often, the interviewer simply prepares a checklist, sometimes with suitable prompts such as 'Tell me about the first thing you do when you get into the office in the morning'. Clearly, this free-form approach is more demanding for the interviewer, but the data obtained does generally repay the effort. Completely unstructured interviews are sometimes used where it is particularly important to minimize designers' preconceptions, or where very little background information is available beforehand. As the term suggests, there are no preset questions or topic beyond the general subject of the project in question.

7.3.1 Contextual Inquiry

Contextual Inquiry (CI) brings together a number of techniques including artifact collection and observation under one unifying theme or philosophy. There are four guiding principles of contextual Inquiry:

- **Context**. Here the advice is to go to the customer's workplace and observe how work is actually carried out. This allows the analyst to experience the rich everyday detail of work. It is best to focus on concrete data and tasks (i.e. user stories) rather than generalized abstraction.
- **Partnership**. One of the core premises of CI – and of its Scandinavian ancestors – is that analyst and customer are expert in their different fields. Thus customers can genuinely influence the analyst's interpretations of the work and design ideas based upon it.
- **Interpretation**. It is not enougth simply to observe and document: the analyst must interpret workpace data. The analyst abstracts from the stories, producing a more conceptual interpretation. The analyst should reflect her interpretation back to the customer and listen to the response. Be prepared to be wrong!
- **Focus**. Each site visit and interview needs a focus, though concentrating on one part of the work helps to see detail, but at the expense of other aspects. Be clear about the focus and try to make neither too broad nor too narrow.

7.3.2 Stories, scenarios and early prototying in interviewing

Once there is a rought idea of what the new technology might do, discussing a scenario will highlight many issues, from the naming of individual functions to the impact of changes in work practice.

Prototypes are very often used to embody scenarios in possible technology. Whether or not a prototype is used, the analyst and the customer 'walk through' the scenario, while the analyst probes for comments, problems, possible alternatives and suggestions in general. Where many new issues emerge, it may be the case that the early concepts underlying the scenario or prototype are misconceived, and should be radically rethought.

7.3.3 Thnik-aloud commentaries

When it is necessary to know a good deal of low-level detail about current technology users can be asked to talk through the operations concerned – including their internal cognitive processes – as they use the technology in question. It is important to remember, however, that by imposing the requirement to generate a commentary you are interfering with the very process you are attempting to study.

7.3.4 Practical considerations in interviewing

Some practical 'hints and tips' for interviewing

Preparation 'Idiot questions' can uncover unspokent assumptions but use them deliberately, not by accident.

Keeping track of the interview Interviewing is hard work and more effective if carried out by a pair of interviewers. One person can take the lead while the other makes notes. A full transcription is rarely needed, but if it is, an audio-typist can save hours of your time. The typist will need briefing about any technical terms.

Telling stories Just because telling stories and listening to them is such a natural thing to do, they can be misleading. As listeners, designers are looking for current problems, scope for improvements or endorsements of early design ideas. As storytellers, people may respond by giving such things disproportionate emphasis. You need to be aware of this in analysing interview data.

Reflection and exploration Reflecting back during the interview helps confirm that you have understood what has been said. It is often a good idea to have the interviewee review a summary of the interview.

General-purpose exploratory questions

- Tell me about your typical day
- Tell me three good things about ...
- and three bad things?
- What if you had three wishes to make the application better?
- What has gone wrong with the application recently? How did you cope?
- What else should we have asked about?

When to stop Deciding when to stop interviewing means balancing practical constraints against the comprehensiveness. In many cases, client resources limit the process. With unlimited resources, the general rule is to stop once no new insights are being obtained.

7.4 Questionnaires

Questionnaires are one way of streamlining the understanding process if a large number of people are to surveyed and resources are not available to interview them individually. Questionnaires need to be designed, prototyped and evaluated in the same way as any other form of interaction design. Questionaires are ideally suited to gathering a large amount of quantifiable data, or to capture responses from people who cannot be involved more directly. Another technique for gathering data is 'crowd sourcing'. Here, small specific tasks are put on the Web and volunteers sign up to take the tasks. A good questionnaire is time-consuming to construct so that all the items:

- are understandable
- are unambiguous
- collect data which actually answers evaluation questions
- can be analysed easily

Perception of system design are often collected through rating scales known as Likert scales. The Likert scale is the most common of a number of methods for elicting opinion. People are asked to indicate their agreement with a statement using a five point scale. Another approach is to device 'bipolar' rating scales, often called semantic differentials. Perhaps the most important piece of advice is to pilot the questionnaire in draft form with a few people who are similar to the target group. It is always surprising how an apparently simple question can be misunderstood.

7.5 Probes

Probes are collections of artifacts designed to elicit requirements, ideas or opinions in specific contexts. Gaver argues that probes are meant to confront, they are intended to provide inspiration for designers rather than elicit specific requirements. Technology probes are another form of probe that were used to gather requirements for home technologies and the area has now evolved into a whole area of 'probology'. Probes inspire and provoke designers to engage with the lives of others. Another analysis of probes (cultural, mobile, domestic, urban) concludes that probes represent the 'turn to the personal' in direct ference to the 'turn to social' that happend in HCI. Probes are an amalgam of social science methods that enable designers to focus upon the individual's everyday life, going beyond the general.

7.6 Card sorting techniques

Card sorting refers to a number of techniques concerned with understanding how people classify and categorize things. How people organize things is a very personal matter. Card sorting is particular relevant in website design as the structure of the content is critical. As a method of understanding, card sorting can be used in a number of ways. At its most basic card sorting involves writing concepts onto cards then grouping them in different ways. A group of people work

with a facilitator to structure data, concepts, objects or other artifacts, trying to understand what categories are most appropriate to group them together. This results in taxonomy and a set of high-level concepts known as an ontology. There are two types of card sort:

- An open card sort starts with blank cards and participants are asked to write down the objects or actions they think are important in some domain. These are then gathered together into categories
- A closed card sort starts with predefined categories and asks participants to place objects into categories.

Analysts really need to practice card sorting to understand the type of insight it can provide and when best to use the technique.

7.7 Working with groups

The most common example is the focus group. Here a group of people are posed questions by facilitators and encouraged to react to each other's comments. Members of the group can stimulate each other's memories, and discussion may flow more naturally than in the single-person interview. Focus groups can be enhanced by the use of scenarios, prototypes and other stimuli. However group discussion may also inhibit comment about sensitive issues, and can have the effect of highlighting unusual incidents disproportionately.

7.7.1 Brainstorm

Another important group activity is brainstorming. Brainstorming sessions should be fun to participate in, but to achieve this they requre an experienced facilitator. They also require some stimuli, whether as pictures, text or video, to get the ideas flowing. Participants will need some way of recording their thoughts and ideas. An important point about brainstorming is not to dismiss ideas too soon. The sessions should begin with an 'anything goes' approach. Generate plenty of ideas. These can then be filtered in a part of the session that tries to look at the feasability of the ideas and their practical impact. A good technique for helping brainstorming sessions is to get different members of the group to adopt different roles – the ideas generator, the critic, the sceptic, the pragmatic, the documenter, and so on.

7.8 Fieldwork: observing activities in situ

Interviews and questionnaires provide one side of the story, but it is difficult for people to describe all the details of the relevant aspect of everyday life or work. Sometimes this is because the activities is intrinsically difficult to describe in words or because it requires complex subtle cooperation with other people or events. Data from observation helps to get round these problems. In it simplest form the designer can simply ask 'Can you show me how you do that?' during an interview. Ideally you need to see a range of variations on the normal activity and situations where things go wrong, but this is not possible in many situations. Here the important point is to identify what you have not observed, so you do not over-generalize from your data. Being unobtrusive is a skill of its own, and your very presence will naturally tend to make people self-conscious and may alter their behaviour. There are also ethical issues associated with observing people, permissions need to be obtained and anonymity of who said and did what should be ensured.

7.8.1 Workplace studies

By studying work as it actually happens in its real-world setting, reasearchers and practitioners aim to overcome many of the difficulties inherent in Computer Supported Cooperative Working (CSCW). The aim of these studies is to describe in fine detail the day-to-day work.

7.8.2 A guide to practising workplace ethnography

Resource limitations in many projects can support only a limited amount of ethnogrphic work. There are also 'political' issues that raised by the very nature of an ethnographic intervention. Four preconditions for the use of ethnography in commercial projects are:

- Both analysts and the user organization must have positive attitude to investing significant resources
- Stakeholders must be content with the overall purpose of the new system. Ethnographic approaches, with their emphasis on close work with people are unlikely to succeed if there are aims to deskill or replace them.
- Analysts must be prepared to handle the 'political' issues which can araise from such intervention
- Areas of focus must be identified

A review of workplace studies and suggest the following:

- Most can be gained in the early stages when the main design issues are unclear; later work can be focused by data from interviews and other techniques.
- Most information is obtained where people collaborate in some observable way, and share information artifacts in real time.
- Multiple analysts can be valuable, both in observing different activities and in combining perspectives on the same activity.
- Video and audio recording is valuable in capturing data, but field notes remain a vital resource.

Time is required not just to acquire the data but to analyse it. Communicating ethnographic results can be challenging. One approach is to encapsulate the findings in 'vignettes' – short descriptions of typical scenes. A vignette is very similar to a scenario but less structed.

7.9 Artifacts collection and 'desk work'

Data from interviews, questionnaires and observations will have identified a range of artifacts in the form of things that support an activity. It is often possible to supplement this by collecting artifacts – such as documents, forms or computer printouts, in office settings – or to video or photograph items that cannot be removed.

8 Chapter 8, Envisionment

- Use a varity of techniques for envisioning design problems and possible solutions
- Understand the role of concrete scenarios in envisioning design
- Select and use appropriate prototyping techniques
- Understand the main factors in communicating design effectively

8.1 Finding suitable representations

Different representations of design ideas are useful at different stages for different people. They help with generation, communication and evaluation of ideas. In an ideal world, developers would use a wide variety of representations, but this may not be possible. Representations work by suppressing unnecessary detail, thus ensuring that the significant features of some artifact or activity stand out. A good representation is accurate enough to reflect the features of the system being modelled, but simple enough to avoid confusion.

8.1.1 An outline envisionment process

A suggested series of steps for the envisionment process:

- Review requirements and conceptual scenarios
- Develop representations of your design ideas. At a minimum these should include concrete scenarios, storyboards developing the main interaction sequence and snapshot sketches of key screens or other aspects of the product.
- If your product is a new one, experiment with different metaphors and design concepts through yuor representation
- Explore design ideas with people who will be using the system wherever possible
- Develop wireframes to provide more detail on proposed structure and navigation
- Iterate and gradually formalize the design through prototypes and further evaluation.

8.2 Basic techniques

Envisionment is about bringing abstract ideas to life.

8.2.1 Sketches and snapshots

Ideas and thoughts can be quickley visualized and explored through sketching. Designers do well to carry a sketchbook with them so that inspiration can be quickly captured and preserved. Individual snapshots of a design can be provided to show key moments in an interaction and are particularly useful for exploring the impact of certain style or design. Snapshots can be single sketches, or frames, from a storyboard or they can be produced using software.

8.2.2 Storyboards

Storyboarding is a technique taken from filmmaking – using a simple cartoon-like structure, key moments from the interactive experience are represented. The advantage of storyboarding is that it allows you to get a feel for the 'flow' of the experience. It is also a very economical way of representing the design – a single page can hold 6-8 'scenes'. It is often helpful to sktech out a storyboard based around a concrete scenario. Three main types of storyboarding are commonly found in interactive media design are:

- **Traditional storyboarding**. Notes below each sketch usually contain the relevant steps from a scenario and the skteches themself are annotated to indicate interactive behaviour. This is the most usual form of storyboard if there is not a strongly multimedia flavour to the application.
- **Scored storyboard**. If the application has a lot of motion graphics the storyboard can be annoted – a sketch is annoted with appropriate notation and notes about, for example, type, colors, images, sound and other issues are attached underneath.
- **Text-only storyboards**. These are useful if the application has a lot of very complex sequences. You can specify what images appear, what text accompanies them, any accompanying media, general notes about tone, flow, etc.

8.2.3 Mood boards

You gather visual stimuli that capture something of how you feel about the design – photographs and other images, colors, textures, shapes, headlines from newspapers or magazines, quotations from people, pieces of fabric and so on. Attach the stimuli to a pinboard. Even thinking about their arrangement can stimulate ideas. The rule with mood boards is that 'anything goes'. The point of the board is not to formally represent some aspect of the design, simply to act as inspiration. One technique is to get the client to create a mood board. This can give you an insight into the kinds of aesthetics that are likely to appeal to them.

8.2.4 Navigation maps

Navigation is a key feature for many systems. Navigation maps focus on how people move through the site or application. The aim is to focus on how poeple will experience the site. Each page in the site, or location in the application, is represented with a box or heading and every page that can be accessed from that page should flow from it. The maps can be used with scenarios to 'walk through' particular activities and are a very good way of spotting poor aspects of design such as 'orphan pages' or dead ends.

8.2.5 Wireframes

Wireframes are outlines of structure of a software system. Just as navigation maps focus on how pages are structured and linked together, so wireframes focus on the structure of particular types of pages. Use the two together and you have the basics of an app or website design. Wireframes work because they focus on the general elements of a design without worrying about the final detail. Wireframes sit in between the informal sketching and the development of prototypes.

8.2.6 Summary

These techniques are filtering mechanisms for the designer, effectively screening out parts of the design space that the designer does not want to explore in order to focus on the parts that are of interest. A key feature of design and of the techniques discribed here is not to sit staring at a blank piece of paper. Getting inspiration from magazines, websites, software systems, other people, similar systems or products and so, and externalizing ideas through envisionment techniques, are the first steps in design.

8.3 Prototypes

A prototype is a concrete but partial representation or implementation of a system design. Designers communicate the rationals of their design decisions through prototypes. Prototypes stimulate reflections, and designers use them to frame, refine, and discover possibilities in a design space. Prototypes may be used to demonstrate a concept in early design, to test details of that concept at a later stage and sometimes as a specification for the final product. A prototype may be made of something as simple as paper, cardboard or other suitable material, or it may be developed using a sophisticaed software package. The prototyp emight seek to highlight just the interface, or some crucial aspect of the functionality. Prototypes are first and foremost a way of involving people and clients in evaluating your design ideas. There are two main types of prototyping low-fidelity (lo-fi) and high-fidelity (hi-fi).

8.3.1 Hi-fi prototyeps

Hi-fi prototypes are similar in look and feel, to the anticipated final product. Hi-fi prototyping has the following features:

- It is useful for detailed evaluation of the main design elements (content, visuals, interactivity, functionality and media).
- It often constitues a crucial stage in client acceptance.
- It is generally developed fairly well into the project when ideas are beginning to firm up, unless there is some crucial issue that needs to be resolved before any other work can proceed.

A problem with developing hi-fi prototypes is that people belive them. This is dangerous if the designer has not checked details and thought through ideas clearly beforehand. For hi-fi prototyping, accurate details is vital. Another problem with hi-fi prototyping is that it suggests such a system can be implemented. Inevitably a degree of effort and time is consumed in producing the prototype. If this is in the eventual development environment, developers can be understandably reluctant to discard work on features rejected in exploring the prototype.

8.3.2 Lo-fi prototypes

Lo-fi prototypes – often termed paper prototypes, since that is what they are usually made from – on the other hand, have the following features:

- They are more focused on the broad underlying design ideas – such as content, form and structure, the 'tone' of the design, key functionality requirements and navigation structure.

- They are designed to be produced quickly, and thrown away as quickly
- They capture very early design thinking and should aid, not hinder, the process of generating and evaluating many possible design solutions.

Very flexible prototypes can be produced simply and quickly using screen-sized pieces of stiff paper and index cards or Post-its in different colors. Permanent features of each screen are drawn on the card; dynamic items such as dialogue boxes or menues use the cards or Post-its, cut to size as necessary. But it is really important not to spend too much time doing this – the whole point is the low investment in the prototype build. The main practical issues with designing paper prototypes are aas follows:

- **Robustness**. If a paper prototype is to be handled by lots of people it needs to be touth enough to survive.
- **Scope**. Focus on broad issues and key elements; if you are trying to tell too detailed a stroy it can be hard for users to understand
- **Instructions**. There is a trade-off between adding enough detail for someone to be able to use the prototype without the designers helping and adding so much detail that it needs someone to talk the person through it
- **Flexibility**. Have parts of the paper prototype adjustable so that people viewing it can 'redesign it' on the fly, e.g. by using stick notes to represent parts of the screen where the user can move elements around or add new items.

8.3.3 Video prototypes

'Video brainstorming' and 'video prototyping' is a method involving the creationof a physical mock-up model of the product; a video is then shot with an actor interacting with the model as though it were fully functional. A second method is what is sometimes referred to as the 'weartherman' technique, where a video image is superimposed onto computer graphics. Actions are captured against a blue screen / green screen, allowing the removal of the background and the superimposition of the video image onto a pre-modelled 3D environment. The bottleneck is not with the video production hardware or software now, but rather with the skill of the production team.

8.3.4 Different approaches to functionality in prototypes

A horizontal prototype aims to go across the whole system, but deals only with top-level functions, so much of the detail is omitted. In contrast, a vertical prototype implements the full range of features from top to bottom, but is applied to only a small number of functions of the overall system. Combination of these are common. Evolutionary and incremental prototypes eventually develop into the full system.

8.4 Envisionment in practice

People find it difficult to react to a prototype if it is just placed in front of them devoid of any context. The most common strategy is to have people step through a scenario using the new

application or to try carrying out one of their current tasks if the application is to replace an earlier system.

8.4.1 Prototypes and participatory design

Lo-fi prototypes are an essential part of participatory design because people cannot always understand formal models, but they can explore and evaluate ideas through engaging with prototyped systems.

8.4.2 Trade-offs in prototyping

When reflecting on how and what to prototype, the designer should think in terms of the PACT elements. Some trade-offs for prototyping is:

- High-quality graphics and animation can be used to create convincing and exciting prototypes but may also lead to premature commitment to some design decisions.
- Detailed special-purpose prototypes help to answer specific questions about a design, but building a meaningful prototype for each issue is expensive.
- Realistic prototypes increase the validity of user test data, but may postpone testing, or require construction of throw-away prototypes
- Iterative refinement of an implementation enables continual testing and feedback, but may discourage consideration of radical transformations

Prototyping is used throughout the design process. Used at an early stage, a quick prototype can be developed and shown to the client / user for comment on the general design. Rapid prototyping is common in user interface design where software such as is used to illustate concepts. The prototype will be 'thrown away' because implementation will be in a different language. Use case prototyping is when a 'polished' video is produced to disseminate to a wider audience and also to the software and hardware development team whose job it is to bring the product into existence.

8.4.3 Prototyping tools

A good prototyping tool should:

- Allow easy, rapid modification of interface details or functionality
- For designers who are not programmers, allow direct manipulation of prototype components.
- For incremental and evolutionary prototypes, facilitate reuse of code
- Not constrain the designer to default styles for interface objects.

Prototyping functionality in software has its own pitfalls. For exmaple, if the interface prototype diverges from the functional prototype it may not be possible for them to be brought together.

8.4.4 Presenting designs

Presenting design ideas clearly and appropriately is a key skill of the designer. The design process is a long one, with many different stages, there are many different people involved and there are many different reasons for giving a presentation. The combination of these will affect what sort of presentation and what sort of representation are suitable. If the prototype or design is still at the concept stage, broad images of the system are appropriate, with little functionality except in key areas. An early design will emphazise the design principles and the basis of the design language. Finally it is important to be clear about what is being highlighted by the presentation. Is it the functionality and events, or is it the interactions and usability with the focus on look and feel, or ease of use?

9 Chapter 9, Design

- Understand the nature of conceptual and physical design
- Understand how metaphor works in design
- Undertake an object-action analysis to inform design and to produce a conceptual model of a new system
- Describe how the system will look and behave through specifying the design language and interaction patterns
- Specify a design in a form that can be implemented by programmers using use cases.

9.1 Introduction

The minimum system specification is a conceptual model, a set of use cases and a design language. Conceptual design is concerned with arriving at an abstract description of the syste – its logic, functions, structure and content – but not with how the structure and functions are to be physically realized. Physical design is concerned with who does what, how the artifacts will look and how they behave. The distinction between conceptual and physical design does not dictate that conceptual design should be finished before physical design starts. Analysts and designers will iterate between these two levels of design description and will fix on some physical design decisions in order to understand the conceptual level better.

9.2 Conceptual design

Clear conceptual design is central to developing systems that are understandable. Designers need to ensure that their conception of the system is easily learnt by people and fits with their expectations and preferences. This is so that people can develop a clear 'mental model' of the system. A good conceptual model will come from considering the underlying metaphor that is being used to provide the structure for the design, and by considering how things are classified and organized.

9.2.1 Exploring design concepts

How do you do? is concerned with the ways in which we affect the world. Do you poke it, manipulate it, sit on it?

How do you feel? concerns how we make sense of the world and the sensory qualities that shape media. Focusing on 'how do you feel' takes us into the areas of satisfaction, affect, enjoyment, involvement and engagement.

How do you know? concerns the ways that people learn and plan; how designers want people to think about their system.

9.2.2 Exploring the design space

A design space constrains a design in some dimensions whilst allowing exploration of alternatives in others. Designers always work within constraints, whether these are financial or functional, but they need to take care not to impose too many constraints too early in the process. Brainstorming is a good way of expanding the design space. One way of highlighting design issues is to undertake a walkthrough using the concrete scenarios to drive the thinking of the designer. Scenarios are very effective at forcing issues into the open, so that claims about designs can be articulated, documented and evaluated.

9.3 Metaphors in design

Metaphor is generally seen as taking concepts from one domain and applying them to another. In the development of interactive systems we are constantly trying to describe a new domain to people. So we have to use metaphor to describe this new domain in terms of something familiar. After a while the metaphorical use of a term becomes entrenched in the language to such an extent that people forget it ever was a metaphor. Paths and maps may be thought of as metaphors for the design of interactions. Different metaphors will lead to different conceptions and designs. Considering one metaphor can stimulate creative leaps into other ways of thinking. Metaphor is not just a literary thing, it is fundamental to the way we think. An important contribution from this view is that a metaphor is much more than a simple mapping from one domain to another. It is much more complex affait. What we call 'metaphors' in design are really blends. A blend takes input from at least two spaces, the characteristics of the domain described by the source and the characteristics of the target that we are applying it to. The blend that results from bringing two domains together in this way will have some features that were not in the original domains. Blends have an emergent structure that results from bringing two sets of concepts together. It is exactly these emergent properties that make one design better than another. For metaphors and blends to work, there must be some correspondence between the domain that come from a more generic, or abstract, space. Of course, the generic space is itself a domain and hence may itself be using metaphorical concepts. This process works its way back until we reach the fundamental image schemas that are core to our thinking. These include the container, path, link and other such as colors and those bodily schemas that come from experience and perception (up, down, in, out, central, peripheral, etc.). Metaphor design works as follows:

- The source domain has some features (concepts and functions).
- The target domain has some concepts and features.
- So it is important to analyse the relationship between these
- Too many features in the basic domain results in 'conceptual baggage' of the metaphor.

- Too few features, or too many inappropriate features, may lead to confusion.
- Aim for people deriving appropriate expectations

The important thing about metaphor is to get a good conceptual correspondence. As with any aspect of interactive systems design, evaluation of metaphor is essential. There are, however, a few principles for good metaphor design:

- **Integration**. This is to do with coherence and not mixing metaphors. The aim here is to manipulate the whole blend, maintaining the web of relationships. The blend has its own structure and it is this that needs to have consistency maintained.
- **Unpacking**. People should be able to unpack the blend and understand where the inputs have come from and why they work. With consideration, reflection and evaluation the designer can archieve this. Designers should only have things in the blend for a good reason.
- **Topology**. The different spaces should have a similar topology. Topology is about how the concepts are organized and structured.
- **Analysis**. When undertaking an analysis the designer should concentrate on getting the appropriate functionality and concepts, exploring the ramifications of the metaphor, evaluating how people will interpret it.
- **Design**. At the design level designers should consider how to represent objects and actions. They do not have to be realistic visual representations.

9.4 Conceptual design using scenarios

Stories aaid understanding, conceptual scenarios abstract from stories to provide generic activities. Fixing certain design constraints leads to concrete scenarios that mey finish up as functional specifications expressed as use cases. A scenario corpus is developed that should be discussed and evaluated at design team sessions and with participation of stakeholders. The most concrete scenarios are used to envision or evaluate specific interactions. While they are superficially easy to construct, there are a number of ways in which scenarios can be made more effective:

- Complement the scenarios with some of the more visual envisioning techniques.
- In a large design team, include real data and materials so that people not directly involved can appriciate, concrete details.
- Think hard about underlying assumptions.
- Include good characterization and develop a number of personas. If this is done well, members of the team start talking about the characters.
- Provide a rich contextual background – this grounds design decisions in real life, forcing designers to think about practicality and acceptability.
- Team members can write their own concrete version of a conceptual scenario that reflects their particular concerns.

The aim is to come up with a collection of scenarios that covers all the major uses and functionality of the product. It would be impossible to writescenarios for all possible variations in use, but those produced should cover:

- Interactions that are typical of a number of similar use situations
- Design issues that are particularly important for the focus of the project
- Areas where requirements are unclear
- Any apsects that are safety-critical.

A good way of doing conceptual design is to undertake an object-action analysis of the scenario corpus. For each of the scenarios in the corpus the analyst works through the scenario descriptions, identifying the various objects that are mentioned and the various actions that are performed.

Working with a corpus of scenarios in this way requires four stages:

- Analyse the individual scenarios, distinguishing between specific actions and more general, higher-level activities.
- Summarize objects and actions from each scenario, merging similar or identical actions where necessary.
- Bring together the analyses from the individual scenarios, collating them into summarized objects, actions and more generic activities.
- Merge actions and objects where they are identical and give them a single name.

9.4.1 Diagrammatic techniques

The result of the object analysis could be represented as an object model, or entity-relation. By developing a conceptual model, we are able to raise and discuss design issues. You may disagree with some of the assertions, this is fine, this is exactly why making the conceptual model explicit is useful. Object models, or entity-relationship models, represent the main objects of interest in a domain and the relationship between them. There is an important distinction in conceptual models between the specific instances of an object and the class or type of the object. What makes us group things together as an object type is that they have certain characteristics in common. Relationships between objects are expressed in terms of how many instances of an object can be related to how many instances of another object. Typically we are not interested in exactly how many instances, but rather whether a relationship exists between one or many instances. Attributes can be shown on the diagram as ovals if requred. It is not the intention to explore all the details of conceptual modelling techniques here, but rather to make designers aware that they exist.

9.5 Physical design

Physical design is about structuring interactions into logical sequences and about clarifying and presenting the allocation of functions and knowledge between people and devices. Physical design is concerned with taking this abstract representation and translating it into concrete designs. Therer are three components to physical design:

- Operational design is concerned with specifying how everything works and how content is structured and stored
- Representational design is concerned with fixing on colors, shapes, sizes and information layout. It is concerned with style and aesthetics.
- Interaction design in this context is concerned with the allocation of functions to humans or to technology and with the structuring and sequencing of the interactions.

9.5.1 Design language

A design language consists of the following:

- A set of design elements such as the use of color, style and types of buttons, sliders and other widgets
- Some principles of composition, the rules for putting them together
- Collections of qualifying situations – context and how they affect the rules

A design language is how designers build meaning into objects, enabling people to understand what things do and to make distinctions between different types of object. Any language provides a way of expressing things, and design languages are ways of expressing design concepts. Design languages have most impact when they become deeply embedded and when people use them and exploit them unconsciously. A method for design, develops a design languge through:

- **Characterization**. The process of describing existing assumptions and any pre-existing design languages
- **Re-registration**. The creation of a new assumption set through exploring trends and needs through field research
- **Development and demontration**. Using storyboards, prototypes and other envisioning techniques
- **Evaluation** of the reactions to the design
- **Evolution** of the language over time. No matter how good a design, it will only last so long – until circumstances force it to be revisited.

Design languages help to ensure transparency, helping people to understand what is going on inside a device. Finally, people will identify with a style, which helps to define their identity; they act through the design language.

9.6 Designing interactions

The conceptual design should be as independent of an implementation as possible. The move from conceptual to physical design requires designers to allocate functions and knowledge to persons or to devices and hence to create interactions. In designing interactions designers need to consider the capabilities of people and the constraints on what they can do. People will forget things over time. They are not good at following long lists of instructions, at carrying out borring stasks repeatedly and so on. On the other hand, people are good at improvising and at dealing with ambiguity and incomplete information. The interactions should be engaging, enjoyable and fulfilling.

9.6.1 Interaction patterns

The idea of 'patterns' has been adopted by designers of interactive systems and appears as interaction patterns. As with architectural patterns, interaction patterns can be identified at many different levels of abstraction.Patterns build up into complex interactions of menus and mice that we are familiar with: patterns of layout of menues, of the highlighting when the mouse rolls over an item, flashing when an item is selected and so on. More recently, people have been developing gesture patterns for interacting with multi-touch displays. Each pattern is described in a standard way, given a name and description, an indication of the problems it addresses, the design rationale or focus that act on the design decision and the solution that has been adopted. Patterns will typically refer to other patterns and be referenced by other patterns.

9.6.2 Diagrammatic techniques

The diagrammatic technique are concerned with representing the structure of a system. By contrast, the design of interactions is concerned with processes. In many design situations there will be a need to provide a structured 'dialogue' between teh system and the person using it. The system may need to elicit particular pieces of data in a particular sequence, or to guide people through a series of actions, or to present a number of related options together. Dataflow diagrams are a good way of showing the logical steps required to complete some interaction. With a logical flow diagram, designers can debate where the human-computer interface should be and where certain functions should take place. Several other methods are available for representing interactions. Sequence models are one; task structure diagrams are another. Use cases can be used to describe the interactions, as can simple tabular layouts that show people actions on one side and system responses on the other. Another common diagrammatic technique is the state transition network. This shows how a system moves from one state to another depending on the actions by the user.

Part II
Benyon et al 2014 (2. udg.)

10 Chapter 12

Ideally, applying Contextual Design (CD) is a team activity as it requres a wide range of skills. Contextural Design supports the designer in finding out about the context and translating these factors into the design of products that meet customers' needs.

Contextual Inquiry (CI) is the name given to the first stage of Contextual Design. It is essentially a combination of a focused interview and observation. CI brings together a number of techniques including interviews, artifact collection and observation under one unifying theme or philosophy. There are four guiding principles of contextual inquiry: context, partnership, interpretation and focus.

Contextual interview having a length of 2-3 hours of mixed conversation and observation is not unusual. You should try to observe as many different types of everyday tasks as possible. Use the

last 15 minuttes of the interview to review what you have learned.

A flow model in CD represents how work is broken up across people and how it is coordinated to get the whole job done. A flow model is drawn from a particular point of view. Cinoibebts of a flow model:

- **Individuals**. Who is involved. The individuals not the roles.
- **Responsibilities** belonging to each individual.
- **Groups**. A group is more than one person with the same responsibilities.
- **Flow**. How people communicate to get work done.
- **Artifacts**. The 'things' that are processed or support the work.
- The **Communication topic** or **action** such as a request for a room.
- **Places**, such as meeting rooms, are shown if they are central to coordinating the work.
- **Breakdowns**. Problems in communication and coordination.

A sequence model represents work tasks. They can be shown as a sequence of steps or actions. Again drawn from a particular point of view. Components of a sequence model:

- The **intent** (or purpose) that the sequence is intended to archieve.
- The **trigger** that causes the sequence of actions. The intent is the reason why a sequence action is taken, the trigger is the event that sets it off. There is only one trigger for each sequnce.
- A series of **steps** that archieve the intent. There should be enough detail to allow a colleague to implement software to support the activity without having to ask you for help.
- Any **breakdowns** or problems.

Sequence models are not always simply linear. Sometimes people need to loop back and carry out a different task more than once, or there may also be decision points where different steps are taken.

Artifacts are the things people use or create in their work. Interpreting artifacts is rather like detective work. Relevant artifacts are of many different types. the artifact models consists of the artifact itself or a sketch, photo, photocopy or scanned version of it, labelled to indicate apsects of interest. It can be used to tease out of how work is done currently, preferably with customers. Later in the design process it provides basic information as to what data or other material is currently held/processed/communicated, its current organization, and problems occuring. There is one model per stakeholder for each artifact. Components of the artifact model:

- The **information** content itself, e.g. the rooms booked and guest details in the booking diary.

- The **structure** of the object into different parts
- Informal **annotation** of the artifact
- The **presentation** of the artifact – general style and the use of color and fonts
- Note any aspects that change over **time**
- Note **when** it was created, **what** it is used for, and **by whom**
- Finally, any **breakdowns** in its use.

Often cultural assumptions are unspoken and invisible, even though there may be cultural divides between different parts of a customer's organization. The cultural model makes these issues explicit. In some ways it is quite similar to a rich picture. Components of the cultural model:

- **Influencers**. People or bodies which affect the way work is done.
- The **extent** to which the work is affected by the influencers.
- The **direction** of the influence.
- **Breakdowns** arising from cultural issues.

Typical cultural influences include national or international standards and policy, organizational policy.

The physical model is a representation of where the work takes place. It is not an exact floor plan, but rather shows the key features of the workplace. The physical model helps to show why work is carried out in a particular way. It also indicates physical features that may need an equivalent in a computer system. Components of the physical model:

- The **physical structures** of the workplace.
- **Movement** within the workplace.
- **Communications** and **computer technologies** and **networks**
- The location of **key artifacts** created or processed.
- The **layout** of the workplace.
- Finally, **breakdowns**.

11 Chapter 13

Affinity diagrams are brilliantly simple and effective. Construction of an affinity diagram:

- Write each seperate requirement, wish, need, ideally on a Post-it note. Make sure it is fairly succint – a word or two, a sentance at most.
- Repeat until you have perhaps several hundred.
- The affinity diagram is built bottom-up identifying common themes and structure. So, rather than providing the affinity diagram with predefined headings, they should define their own.
- The Post-its should then be affixed to a wall in groups, as the groupings emerge. During this process it is not unusual to discard duplicate or near duplicate requirements.
- Remember to record the groups and headings.

The creation of an affinity diagram is the first step in the design process, as we can begin to see already the features, properties and expected behaivor of parts of the new system.

Contextual design requires that we bring together (consolidate) all the different kinds of models. During the consolidation process, we are looking for ways in which the system can be redefined.

The user environment design (UED) is a customer-centered, high-level design for interactive application that we are intent on implementing. It is customer centered in that it does not use obscure notation but plain English with a few boxes connected by arrows. The UED is used to guide the detailed design of the system, the design of the user interface and the management of the development process. Creating and using a user environemnt design helps avoid undesirable state of affairs. The UED is a diagram that represents the structure of the new design. It is:

- Based on the underlying structure and sequence of the work itself
- A shared resource for designers, developers and project managers
- Constructed before user interface design take place
- Similar in some ways to a website navigation map or to the floor plan of a building.

At its simplest a UED comprises a number of focus areas.

Part III
Artikler

12 The Anatomy of Prototypes

Prototypes as filters, and prototypes as manifestations. Prototypes are not just tools for evaluating or proving successes or failures of design outcomes. Two fundamental aspects of prototypes form the basis of a framework:

- Prototypes are for traversing a design space, leading to meaningful knowledge about the final design as envisioned in the process of design
- Prototypes are purposefully formed manifestations of design ideas.

It is critical to choose carefully the fidelity and automation level of the evaluated prototypes. The most efficient prototype is the most incomplete one that still filters the qualities the designer wants to examine and explore. Externalization of thought gives rise to new perceptual and cognitive operations that allow for reflection, critique, and iteration. The material chosen for a prototype has direct implications on users' perceptions when it is used for evaluating a design concept. The fundamental prototyping principle is that the purpose of designing a prototype is to find the manifestation that, in its simplest form, will filter the qualities in which the designer is interested without distorting the understanding of the whole. In defining the set of filtering dimensions, we include appearance, data, functionality, interactivity, and spatial structure. We define the three core aspects of the manifested forms of prototypes as materials, resolution, and scope. Adding more filtering dimensions creates a more complex prototype that is more difficult to interpret. This added complexity means that the designer has to decide what to filter and carefully craft the prototype in relation to her chosen filtering dimensions. To create a prototype is to find the manifestation that, in its most economic form, will filter the qualities in which the designer is interested, without distorting the understanding of the whole. The purposes for which prototypes are used can be broadly categorized into the following areas: (1) evaluation and testing; (2) the understanding of user experience, needs, and values; (3) idea generation; and (4) communication among designers.

13 Making the future palpable (p. 145-163)

The stress and urgency of emergency situations make invention and adoption of new technologies and new practices hard to realise. Two issues in particular hamper innovation:

- The 'unknowability' of future practice
- 'Invisibility'. While this protects people from having to make many complex choices, it also takes the human 'out of the loop', making it hard for people to know what technologies are doing and to trust them.

The main design goal is to provide infrastructural support for making ubiquitous computing 'palpable', that is, 'noticeable, manifest, obvious, clear'.

Future laboratories introduce functional prototypes (developed to varying degrees) into as realistic as possible enactments of real work. The practitioners do their work with or 'through' future technologies, inventing new ways of working along the way. In future laboratories a prerequisite for deriving maximum profit is that the involved practitioners are experienced. Future laboratories are not just about technical and practical creativity and experimentation: reflection is a very important part.

New technologies should not only be meant to be used in major incidents situations – they also have to be used and be useful in everyday situations.

By enabling and 'forcing' hands-on, as realistic as possible, experience of one such socio-material-technical integration, future laboratories provide insight and knowledge that could not have been obtained through discussions or 'play' around a table. Future laboratories are a step towards more integrated socio-technical innovation, where researchers and professionals share expertise, ideas, and responsibilities for change, making safe experimentation a part of iterative, collaborative innovation in a way that makes it easier to realise the great potential of pervasive computing technologies.

14 The new methodology

Most software development is a chaotic activity, often characterized by the phrase "code and fix".

Engineering methodologies vs agile methodologies. These new methods attempt a useful compromise between no process and too much process, providing just enough process to gain a reasonable payoff.

Agile methods are adaptive rather than predictive. Agile methods are people-oriented rather than process-oriented. Design which is difficult to predict and requires expensive and creative people, and construction which is easier to predict.

- In software: construction is so cheap as to be free
- In software all the effort is design, and thus requires creative and talented people
- Creative processes are not easily planned, and so predictability may well be an impossible target.
- We should be very wary of the traditional engineering metaphor for building software. It's a different kind of activity and requires a different process

The problem with predictive processes is that project quality is measured by conformance to plan. A project that's on-time and on-cost is considered to be a success. This measurement is nonsense to an agile environment. There's also an interesting synergy: not just does adaptivity require a strong team, most good developers prefer an adaptive process. People are the most important factor in software development. Each team should not just choose their own process, but should also actively tune their process as they proceed with the project. Styles of agile development:

- XP
- Scrum
- Crystal
- Context driven development
- Lean development
- Rational unified process (RUP)

15 User-Centered agile methods

TODO

16 Back to thinking mode

Writing the diary supported the analysts' reflections on what happened. Writing the diary provided the analysts with an opertunity to evaluate and develop their working practices in a systematic way. The diary served as a shared memory. The diary serves to trigger the observation of important, latent apects of working practices. The use of diaries introduce both discipline and care into each group's working practice. The diary is written in close connection with the actual activity, and descriptions and reflections in the diary adress and affect future situations and actions. Advice for writing diary:

- Make the intentions clear
- Be desciplined and careful
- Make a checklist of issues
- Be selective and thorough
- Decide on when
- Reflect on how
- Consider using other related techniques