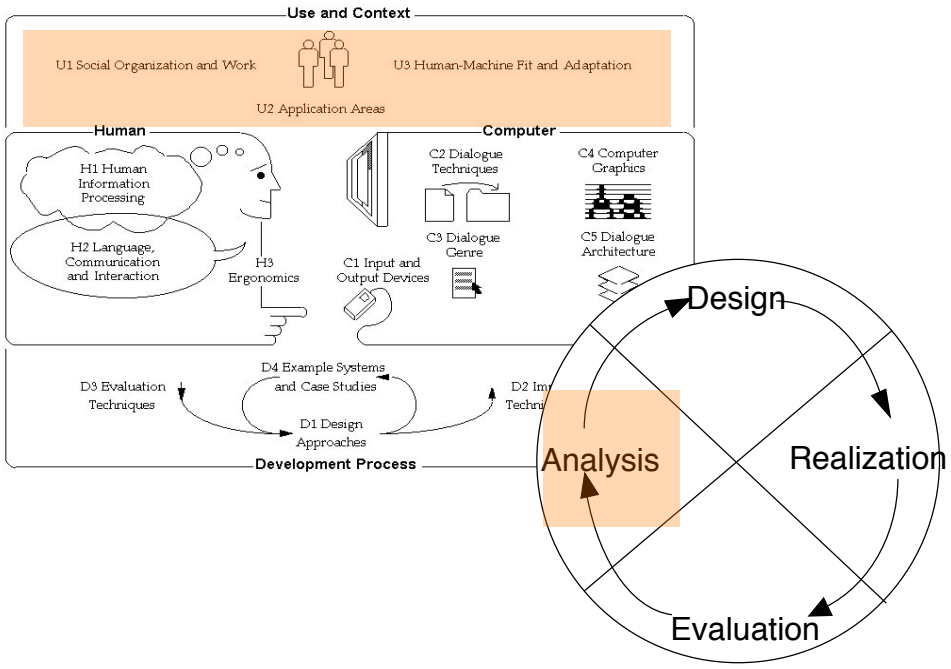
5 Analyzing the Requirements



Analyzing Existing Systems

- In most cases, some kind of system is already in use
  - Automated system
  - Incoherent combination of software tools
- Purpose of analysis:
  - Understanding the work processes
  - Finding opportunities for improvement
  - Baseline data for the new system
- Analysis mainly through user studies
- Possible manual analysis steps:
  - Observation of workflow
  - Creation of realistic example scenarios with real data
- Possible automatic analysis steps:
  - Statistics about actual usage of various features
  - Statistics about data usage, data volume, …

Automated Analysis of Existing Systems

- Use functions/mechanism included in products, e.g.
  - Log files for using web applications
- Use additional software to monitor usage
  - Key logger
  - Proxy server
  - Screen capture tool
- Extend the software that is used to track/analyze usage
- Typical questions
  - What applications are used in the work process
  - How often is application X or function Y used
  - What files are accessed during the work process
- Tools, e.g.
  - analog - Website usage analysis software
    http://www.analog.cx
  - Filemon – logging file usage
    http://www.sysinternals.com/Utilities/Filemon.html

5 Analyzing the Requirements

Requirements Definition Process (Cooper)

From A. Cooper, About Face 2.0

- Defining the requirements
  - Step 1: Creating problem and vision statements
  - Step 2: Brainstorming
  - Step 3: Identifying ***persona*** expectations
  - Step 4: Constructing context ***scenarios***
  - Step 5: Identifying needs
    - » Data needs
    - » Functional needs
    - » Contextual needs
- Scenarios
  - Are extremely helpful to understand the real needs of users
  - are an excellent starting point for design activities

Scenario Development

- Important methods
  - **General scenario**
    - » Fictional story featuring the product to be developed and explaining implications on users experience
    - » Similar to describing conceptual models, may be concept video
  - **"Day in the life" scenario**
    - » Creating a fictional user
    - » Describing a day in her life augmented with the product to be developed
  - **Situation scenarios**
    - » Fictional story concentrating on a specific situation, e.g. an emergency case)
- Forms of presentation
  - Writing
  - Video
  - Acting/playing it – connected to paper prototypes

Who to Design for? – Personas

- Don't design for the average user!!!
- Differentiate and create a set of typical users
  - "Persona" = concrete representative of one kind of typical users
- Use background information about the user group
  - Literature
  - Interviews
  - Statistics
  - Analysis and observations
- *Invent* a set of specific persons
  - Age, place of birth, current location where she lives
  - Education, profession, job profile, background, hobbies
  - Social environment, family, work relationships
  - Goals and abilities
- Personas are representative for the target audience, but they are NOT average!
- Personas often do not fully correspond to market segments!

Scenario Development
Why Persona?

- Avoiding the “elastic user”
  - If you do not specify the user you can change their abilities to support a design decision made = “elastic user”
- Avoiding self-referential design
  - The designer or developer often assumes (implicitly) that users have his goals and his skills and abilities.
- Avoiding design edge cases
  - Focusing on the design issues which are on the edge of the anticipated audience can consume a lot of effort. By use of typical users the focus on edge cases can be reduced.
- Generally, make requirements concrete
  - Seemingly unnecessary detail helps in making the requirements accessible and understandable for a large audience (users, managers, developers)

Persona Examples (1)

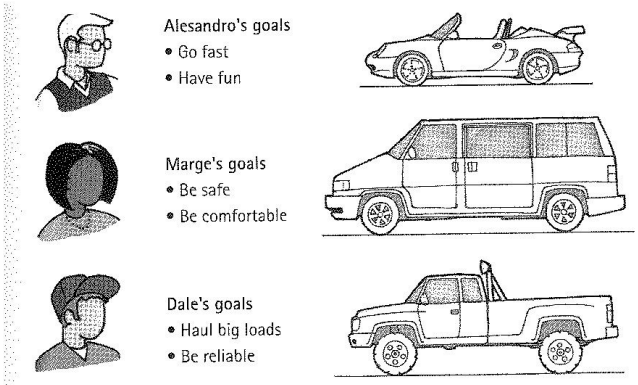


Figure 5-2: A simplified example of how personas are useful. By designing different cars for different people with different specific goals, we are able to create designs that other people with similar needs to our target drivers also find satisfying. The same holds true for the design of digital products and software.

A. Cooper

Persona Examples (2)

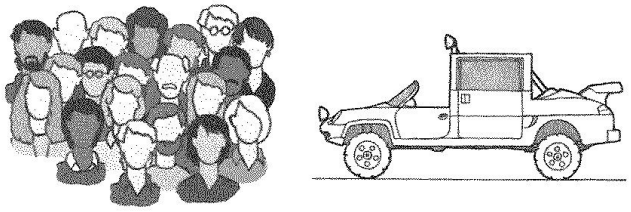


Figure 5-1: A simplified example of how personas are useful. If you try to design an automobile that pleases every possible driver, you end up with a car with every possible feature, but which pleases nobody. Software today is too often designed to please too many users, resulting in low user satisfaction. Figure 5-2 provides an alternative approach.

A. Cooper

"Day in the Life" Scenario

- Describe the usage of a product in the context of a day
  - In particular for products that are used more than once a day, e.g. mobile services, helps to identify practicalities
- Based on the information gathered invent a day
  - Working day or holiday
  - Make a plan what the persons is going to do on this day
  - Make it a normal day but include real life tension and trade-off (e.g. getting kids to school and having a meeting shortly after that)
  - Don't let the day to be perfect (e.g. you may forget a document at home)
  - Don't make the day a nightmare (e.g. do not anticipate the user's airplane is going to crash)
- Describe a day of the fictional user in detail
  - Concentrate on the relation between the users actions and tasks and the product introduced.
  - Basically asking: "How does the product change the life?"

“Day in the Life” Scenario (1)

Example from the European Project TEA: general approach

- Project Vision: Creating a mobile phone/PDA that is aware of the user’s action and the environment (e.g. user is driving, user is holding the device, user is in a meeting, it is raining, user is at a particular location etc.)
- Technology driven – but what are the applications?
- “Day in the life” scenario for 6 *fictitious* users to explore possible uses (assumed users are already mobile phone “power” users)
  - Franz, 34, journalist, Munich
  - Meredith, 38, Vice President, Marketing, Chicago
  - Mike, age 14, lives in Bath in the UK, ordinary school
  - Patricia, 35, Architect & building designer, Bologna
  - Jochen, 24, geo-physics student, Salzburg
  - Janni, 43, field engineer for a power company, Finland

“Day in the Life” Scenario (2)

Example from the European Project TEA: A day in Meredith’s life

- Complete scenario is about 6 pages, excerpts form the main sections
- User and Situation Summary
  - Professional, Female Doctor, Vice President, Marketing
  - Meredith, 38 in Chicago/USA
  - Married to Tom 37 (IT-professional), having a daughter Sheila (7 years).
  - The day: traveling, Medical Conference, A lot of meetings before the Conference duties, in conference Hotels and conference boot
- User

  “*Meredith Miller is a 38 year old Marketing specialist in the pharmaceutical industry. She was born in the U.K. but now she is based in Chicago, USA. She works for a medium company dealing with pharmaceutical products marketing and distribution, which acts as a strategy consultant for large pharmaceutical and medicinal preparations companies worldwide. She has a degree in medicine, and a master's degree in business administration for pharmaceutical and medical industry….*”
- Situation

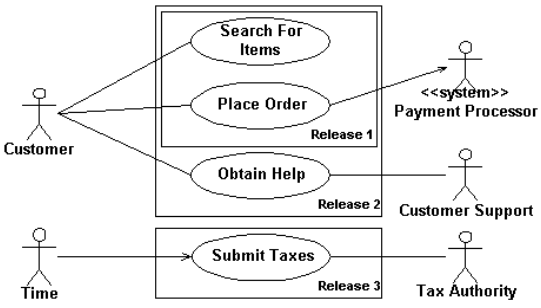
  “*This week, Meredith is traveling across Europe for her monthly visit to European key customers. It is also a special week because two important events, a scientific convention in Copenhagen and an industry fair in Hannover are being held…*”

Situation Scenarios

- Concentrating on a very specific situation
- Investigate the requirements and the impact in a specific situation
- May be rather short
- Situation were the product and potentially a particular function is situated into a context
  - e.g. scanning a document in a work context (interrupting work, going to the scanner, operating the device, getting the data, ..)
- Unlikely situations that are of major importance
  - E.g. emergency procedures such as a fire or building evacuation (not applicable to a word processor but relevant for a power plant control room)
- Methods
  - Writing a fictional story
  - Playing/acting the scene with anticipated functionality

Scenarios, Use Cases, UML

- Unified Modeling Language (UML)
  - Standard graphical modeling language for software systems
  - Includes requirements modeling
- UML "Use Case":
  - A specific way of using the system by performing some part of its functionality
  - Usually depicted graphically showing the involved stakeholders
- "Scenario":
  - In UML-based environments: Example for a use case, giving a detailed sequence of events
  - In HCI: General term for a story about how the system is used, may be used to derive use cases afterwards

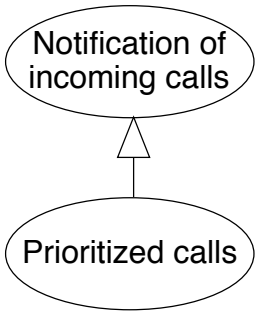


www.agilemodeling.com

Scanarios and Use Cases: Integrated View (1)

Meredith is in a panel discussion at the conference. The kindergarten of her daughter wants to inform her that Sheila has suddenly developed high fever.

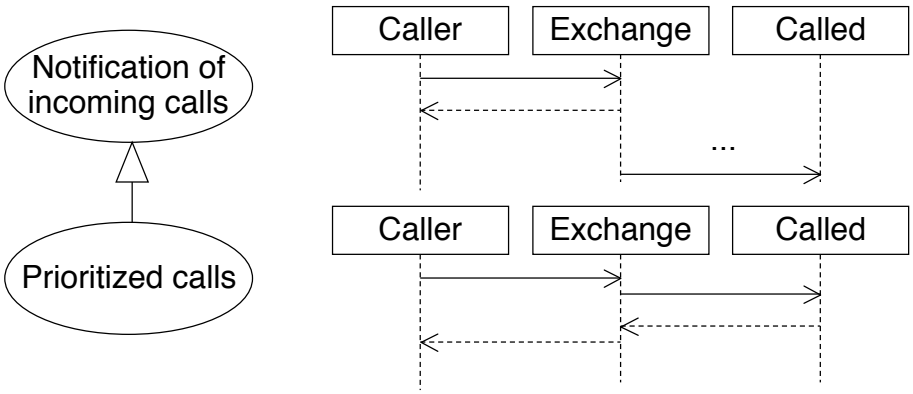
High-level scenario



Use-case diagram

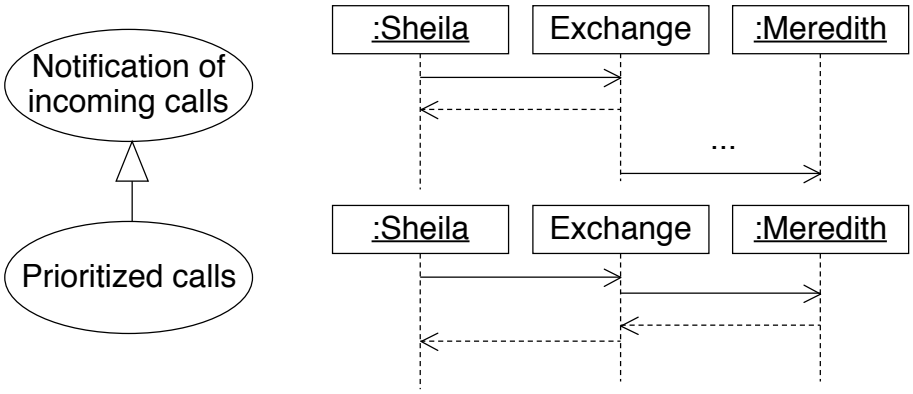
Scanarios and Use Cases: Integrated View (2)

In Software Engineering (specifically using UML), use cases are described e.g. by Sequence Diagrams

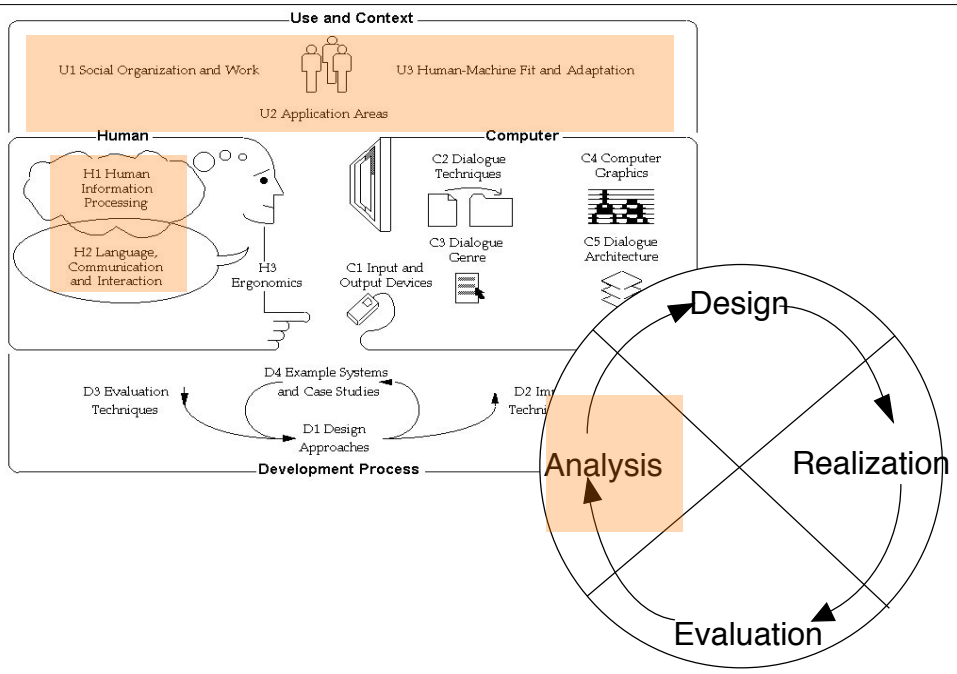


Scanarios and Use Cases: Integrated View (3)

Using concrete examples, often the difference between SE/UML-scenarios and high-level scenarios disappears.



5 Analyzing the Requirements



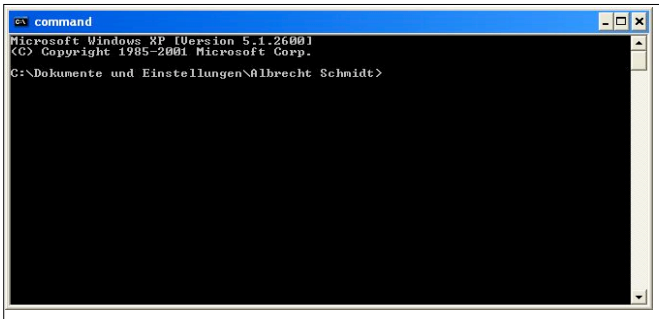
Motivation
Conceptual Models

- How do you figure out that those objects are not usable?
- How do you do it for software?

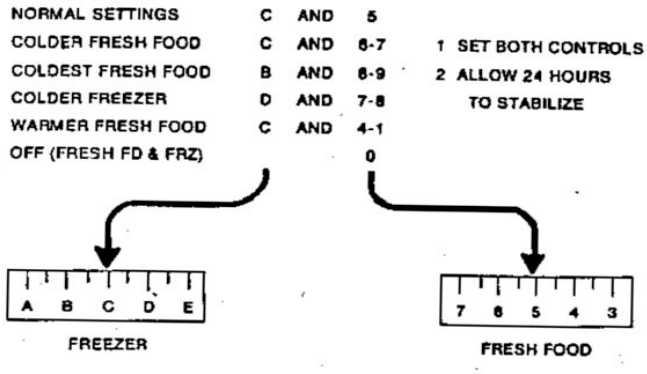
Images from: D. Norman, The Design of everyday things.

Background: The Psychology of Everyday Things

- Norman 2002
- Not primarily aimed at computer science problems but:
  With technologies (web, interactive media, embedded computers) moving into everyday life of most people it becomes highly relevant!
- Terms: Perceived and Real Affordances
  - Affordances determine the range of possible - usually physical - actions by a user on an system/object.
  - Perceived Affordances are the actions perceived by a user that appear to be possible.
  - Example: certain materials afford/support certain forms of vandalism (e.g. glass is smashed, wood is carved, graffiti appears on stone)
- This is also applicable to digital materials and designs.



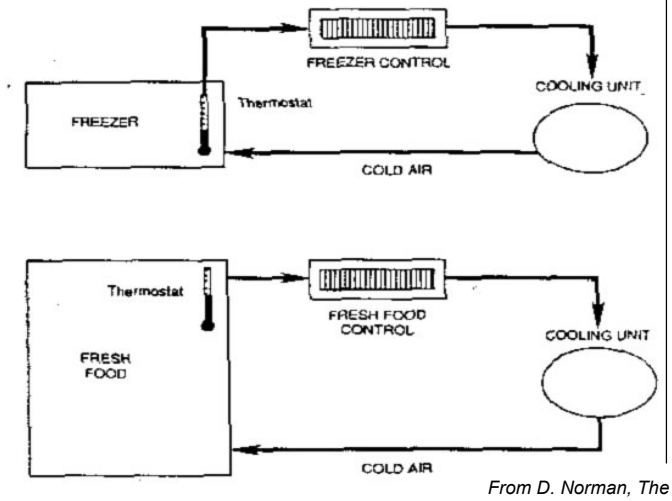
Conceptual Models
Example: Refrigerator

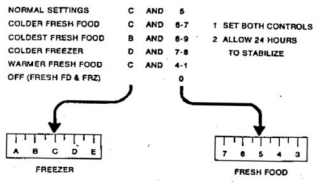


- 2 controls
- Freezer
- Fridge

From D. Norman, The Psychology of Everyday Things.

Example Refrigerator
Conceptual Model 1

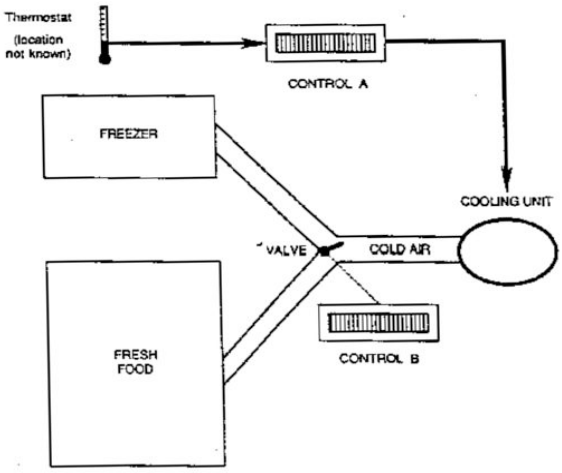


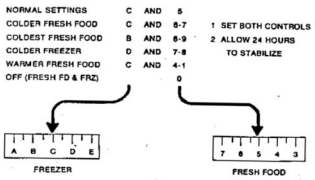


- Idea 1:
  2 cooling units
- One control each

From D. Norman, The Psychology of Everyday Things.

Example Refrigerator Conceptual Model 2





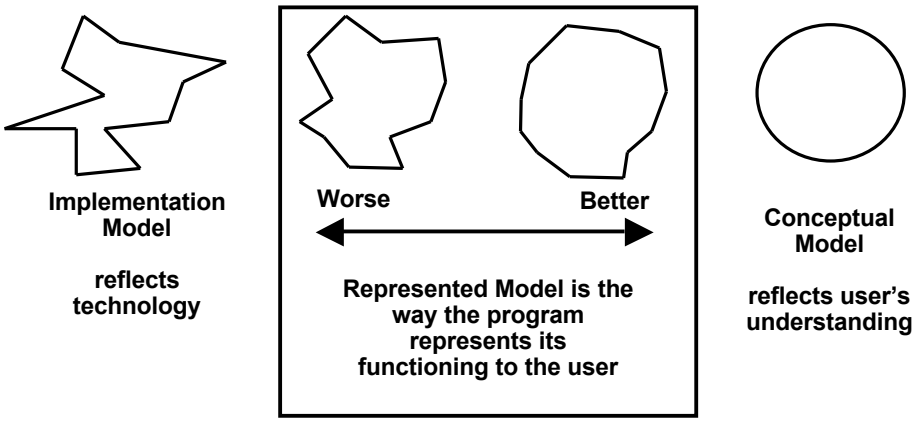
- Actual design – one cooling unit
- Controls have different functions

From D. Norman, The Psychology of Everyday Things.

Example: Heating Control

- You come home and it is very cold. Heating is off.
- Your heating system is thermostat controlled.
- To which setting do you turn the thermostat?
  - 1, 2, 3, 4, 5, 6

Implementation, Represented, Conceptual Model



From A. Cooper, About Face 2.0

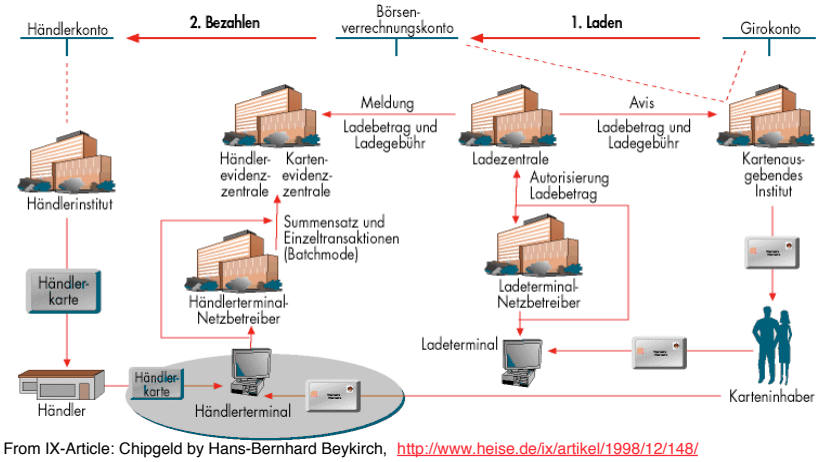
Example: 'Geldkarte' (1)

- Store cash on the card
- Pay with the card

Conceptual Model – by the user

Example: ‘Geldkarte’ (2)

Some aspects of the implementation model



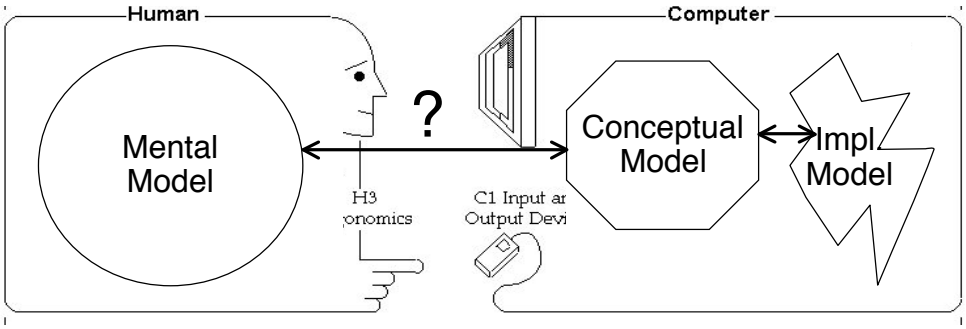
From IX-Article: Chipgeld by Hans-Bernhard Beykirch, http://www.heise.de/ix/artikel/1998/12/148/

Models – Human and Computer

- Applications work on an Implementation Model
- They were designed after a Conceptual Model
- Users operate on their Mental Model
- The user interface translates between models

- Provocative Statement from A. Cooper
  “Computer literacy is nothing more than a euphemism for making the user stretch to understand an alien logic rather than having software-enabled products stretch to meet the user’s way of thinking”

Mental Model and Implementation Model



Conceptual Model

- A conceptual model is “the proposed system in terms of a set of integrated ideas and concepts about what it should do, behave and look like, that will be understandable by the users in the manner intended” (Preece, Rogers & Sharp, 2002, Interaction Design, Wiley, p 40)

- “The most important thing to design is the user’s conceptual model. Everything else should be subordinated to making that model clear, obvious and substantial. That is almost exactly the opposite of how most software is designed.” (David Liddle, 1996, Design of the conceptual model. In T. Winograd, (editor), Bringing Design to Software. Reading, MA: Addison-Wesley, p17)

Why is This a Big Issue with Digital Products?

- For simple mechanical systems/processes, the conceptual model and implementation model are very similar, e.g.
  - Hammer
  - Power drill
- For digital systems the implementation model is often very complex
  - Many components, often distributed
  - The service provided is a result of contributions from different parts
  - The digital components are not visible – even when you open the device
- Users still have a simple conceptual models to operate digital products
  - Based on what they see and their experience gained in use
  - By the control options they are given
  - By the behaviour and reactions they observe
  - By what they have learned about the system

References

- A. Cooper: About Face 2.0, Wiley 2003
- D. Norman. The Psychology of Everyday Things, B&T 1989
- K. Bittner, I. Spence: Use Case Modeling, Addison-Wesley 2002