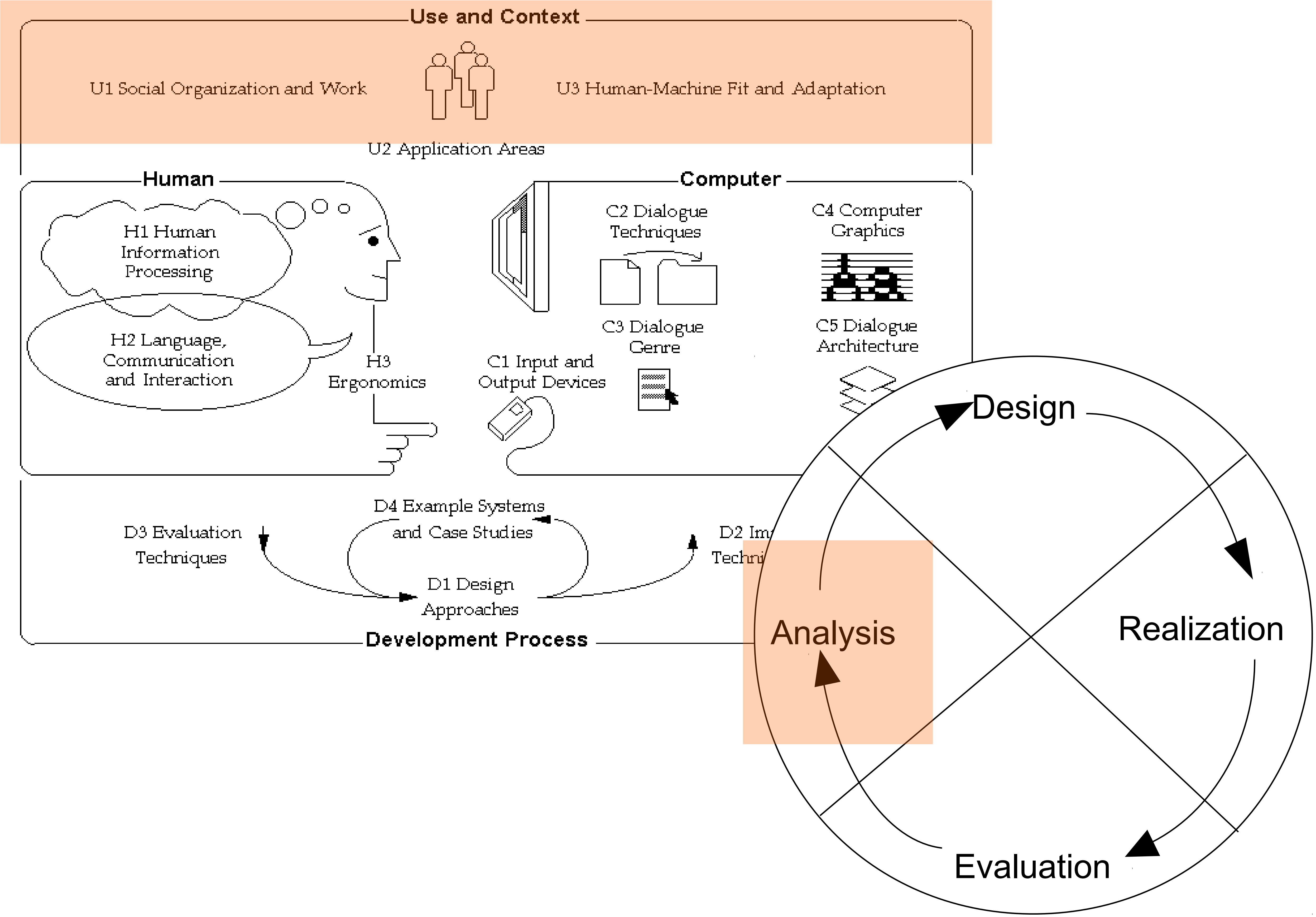
Mensch-Maschine-Interaktion 1

Chapter 4 (May 9th, 2012, 9am-12pm):

User Requirements

Overview

- Introduction
- Basic HCI Principles (1)
- Basic HCI Principles (2)
- ***User Research & Requirements***
- Designing Interactive Systems
- Capabilities of Humans and Machines
- Implementing Interactive Systems
- User Study Design & Statistics
- Basic HCI Models
- User-Centered Development Process



Use and Context

U1 Social Organization and Work

U3 Human-Machine Fit and Adaptation

U2 Application Areas

Human

H1 Human Information Processing

H2 Language, Communication and Interaction

H3 Ergonomics

Computer

C1 Input and Output Devices

C2 Dialogue Techniques

C3 Dialogue Genre

C4 Computer Graphics

C5 Dialogue Architecture

D3 Evaluation Techniques

D4 Example Systems and Case Studies

D1 Design Approaches

D2 Im

Techni

Development Process

Design

Realization

Evaluation

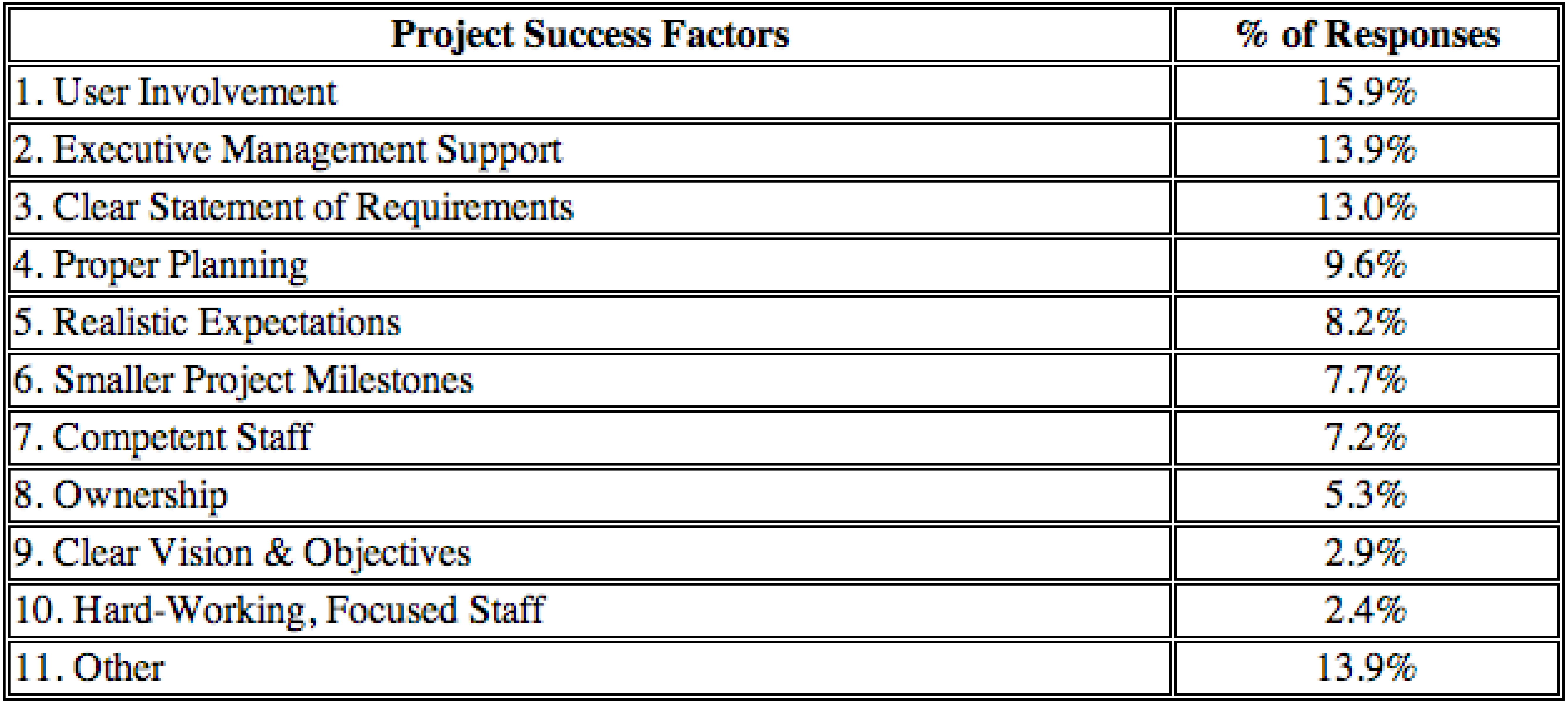
Analysis

Analyzing Requirements

- **Context of Requirements Analysis**
- Analysis of Existing Systems
- Analyzing Ideas and Concepts
- Work Processes, Bottom-Up
- Work Processes, Top-Down
- Scenarios and Use Cases
- Conceptual Models

What Can Keep Projects From Failing?

- Study by Standish Group, 1995
- Interviews with IT executive managers
- What causes projects to succeed?



Project Success Factors	% of Responses
1. User Involvement	15.9%
2. Executive Management Support	13.9%
3. Clear Statement of Requirements	13.0%
4. Proper Planning	9.6%
5. Realistic Expectations	8.2%
6. Smaller Project Milestones	7.7%
7. Competent Staff	7.2%
8. Ownership	5.3%
9. Clear Vision & Objectives	2.9%
10. Hard-Working, Focused Staff	2.4%
11. Other	13.9%

What Do We Need to Analyze?

- Analysis Phase
  - Access and investigate everything that has a potential impact on the solution
- Most important aspects
  - Requirements imposed by the tasks to be supported
    - goals of the project
  - Users, their strengths and limitations
    - people involved in the operation of the system that is to be built
  - Available options for the implementation of a system
    - e.g., technologies
  - Border conditions for development and deployment
    - processes that are improved, changed, or replaced
    - economic constraints
    - organizational constraints and company/customer policies

1. Identifying the Goals

- Why is a new software or system created? What is the main purpose?
  - Replace or improve on an existing system
  - Streamline operation and optimize work processes
  - Introduce a new process or a new option for a process
- In what context is this developed?
  - During continued operation
  - In a restructuring phase
  - In a start-up phase of a company or operation
- What is the role of the software/system?
  - Driver for restructuring
  - Only one issue within a set of changes made in the organization
- How important is the system to the customer?
  - Mission critical, essential for sustaining business?
  - Just a nice additional piece to have?

2. Understanding the People Involved

- Who are the people involved?
  - Who are the decision makers?
  - Who are the users?
  - What relationships exist between users?
  - What relationships exist between users and decision makers?
  - What roles do users have (customer, administrator, controller, supervisor, …)?
  - Which tasks (in the real world and in the system) are performed by the user?
  - Why do people use a system and what is their motivation?

- Shneiderman’s 1st principle: “Recognize User Diversity”

3. Identifying the Effect of Processes

- By introducing or changing software we affect processes in the real world, e.g.,
  - People will be able to do certain tasks they could not do before
  - Certain tasks will be automatically done without user involvement
  - Specific tasks will be speeded up and others may be slowed down
  - The quality of tasks and operations will be improved
  - Processes become traceable and people can be made accountable
  - Some operations will be made easier - others will be more complicated

- Often related to rationalization of the workflow
- Change is not always welcome by everyone

Analyzing Requirements

- Context of Requirements Analysis
- **Analysis of Existing Systems**
- Analyzing Ideas and Concepts
- Work Processes, Bottom-Up
- Work Processes, Top-Down
- Scenarios and Use Cases
- Conceptual Models

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
  - Process Monitor – logging file and process usage etc.
    http://technet.microsoft.com/en-us/sysinternals/bb896645.aspx

Analyzing Requirements

- Context of Requirements Analysis
- Analysis of Existing Systems
- **Analyzing Ideas and Concepts**
- Work Processes, Bottom-Up
- Work Processes, Top-Down
- Scenarios and Use Cases
- Conceptual Models

How to Perform a Requirements Analysis?

- (From a user-centred point of view...)

- General methods, before knowing user community in detail
  - Surveys, opinion polls, questionnaires
  - E.g. Internet polls
- Methods applicable when user groups are roughly known
  - Focus groups
  - Interviews
  - Diary studies
- Methods targeting very specific user groups
  - Ethnographic observation
  - Task analysis

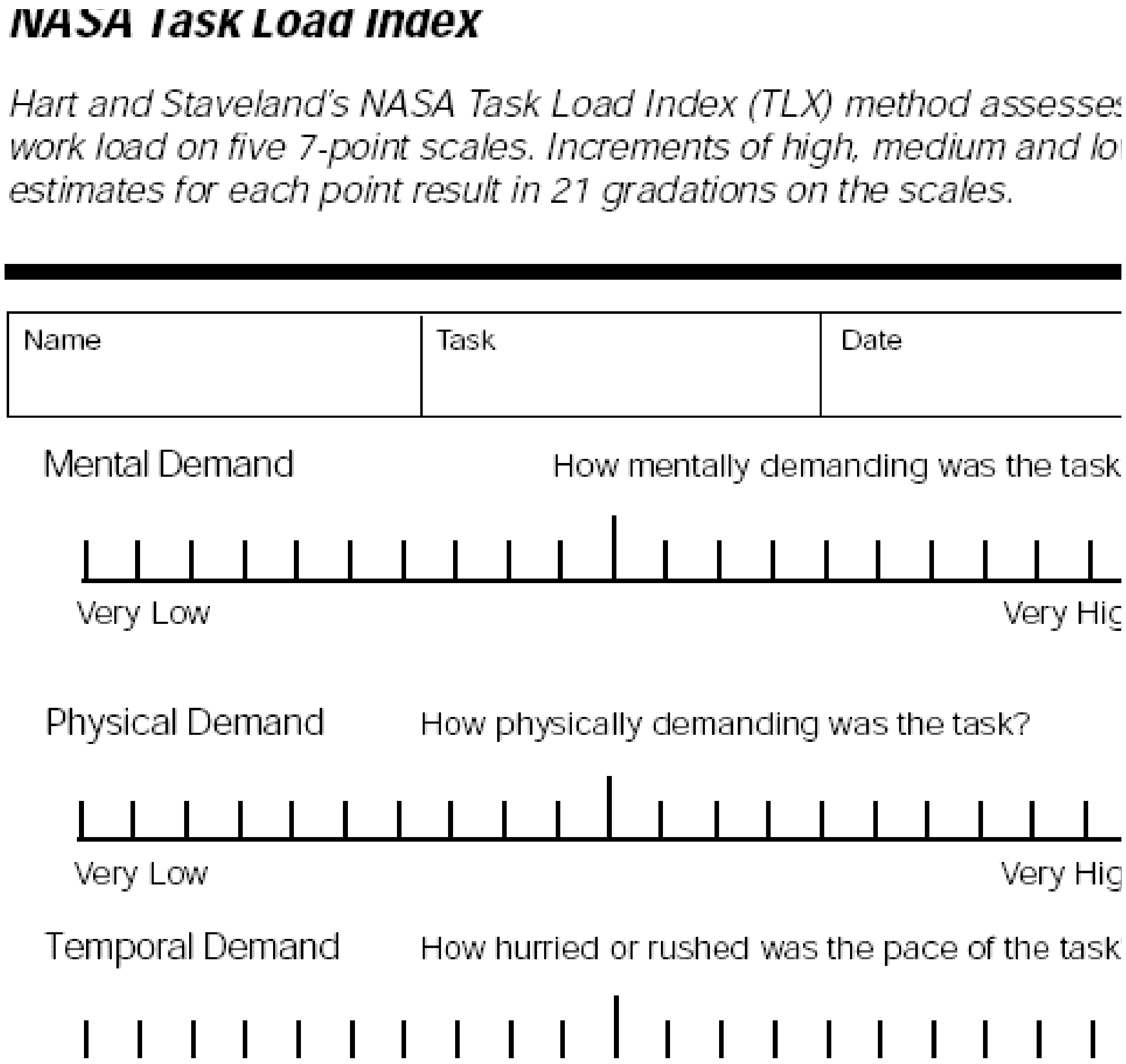
Surveys and Questionnaires

- Find out about
  - Potentially interesting / interested user groups
  - General acceptance / desire for a certain idea or concept
- Gather details about users
  - Demographics
  - Previous knowledge
  - Actual usage of an existing system
  - Opinions on new ideas / concepts / applications

Possible structure

- Focus on subjective opinions
  - Data from a users' point of view
  - E.g., how is a process perceived
  - E.g., how much time users think they spend

Standardized Example: NASA Task Load Index



NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesse work load on five 7-point scales. Increments of high, medium and lo estimates for each point result in 21 gradations on the scales.

Name	Task	Date

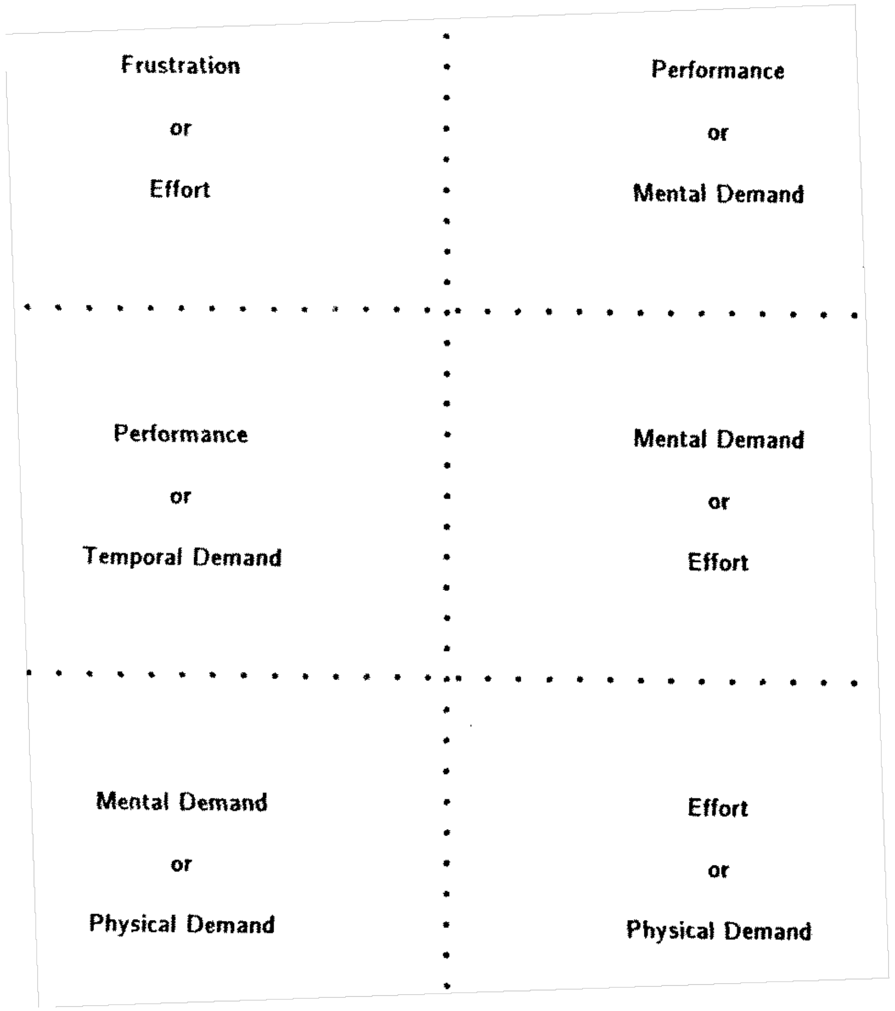
Mental Demand How mentally demanding was the task

Very Low Very Hig

Physical Demand How physically demanding was the task?

Very Low Very Hig

Temporal Demand How hurried or rushed was the pace of the task



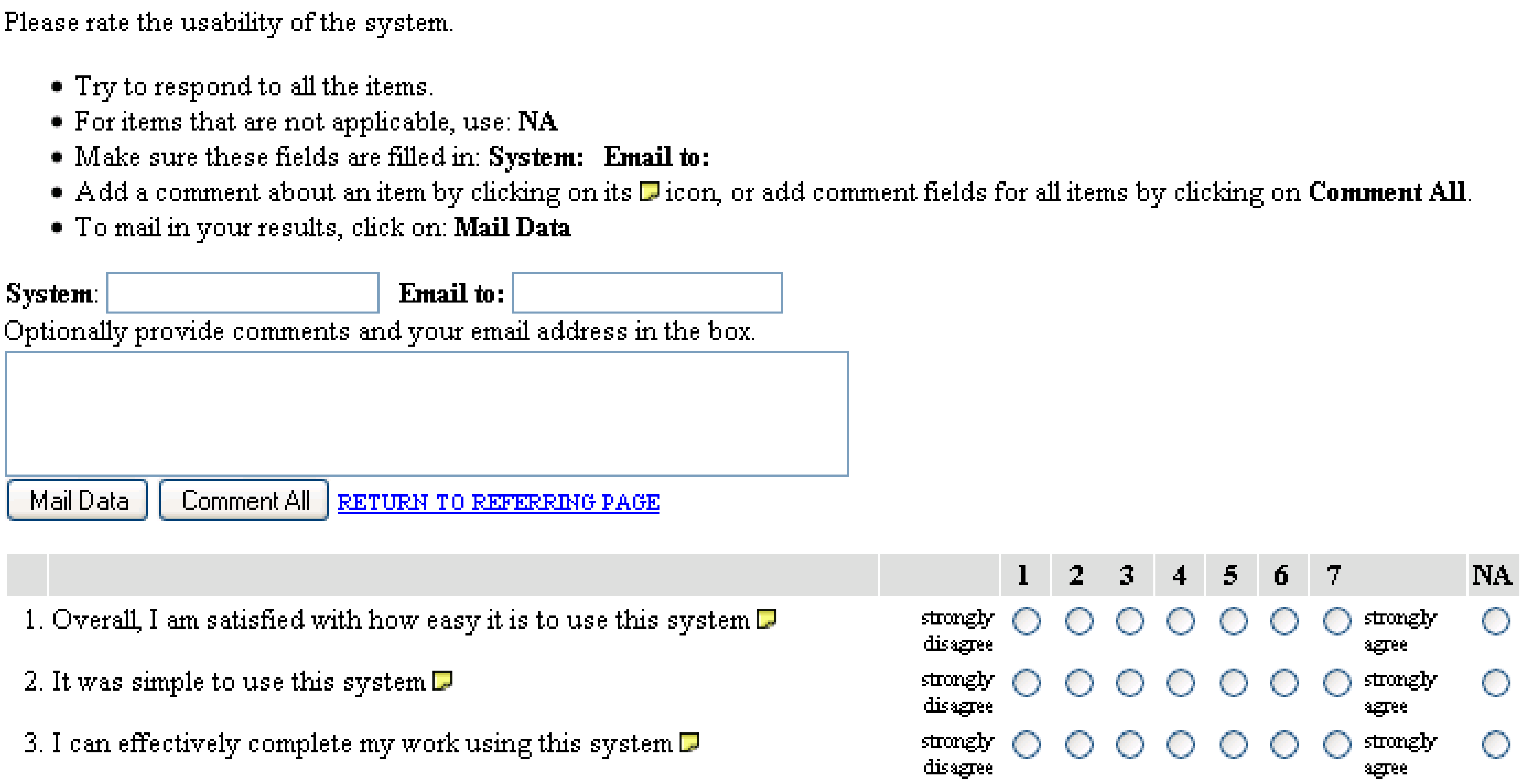
Frustration or Effort	Performance or Mental Demand
Performance or Temporal Demand	Mental Demand or Effort
Mental Demand or Physical Demand	Effort or Physical Demand

...

http://humansystems.arc.nasa.gov/groups/TLX/index.html

Hart, S. G., Staveland, L. E. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In Human Mental Workload, 239-250, 1998.

Std. Example 2: IBM Usability Satisfaction Questionnaire



Please rate the usability of the system.

- Try to respond to all the items.
- For items that are not applicable, use: **NA**
- Make sure these fields are filled in: **System:** **Email to:**
- Add a comment about an item by clicking on its 🗒 icon, or add comment fields for all items by clicking on **Comment All**.
- To mail in your results, click on: **Mail Data**

System: **Email to:**

Optionally provide comments and your email address in the box.

			1	2	3	4	5	6	7		NA
	1. Overall, I am satisfied with how easy it is to use this system 🗒	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
	2. It was simple to use this system 🗒	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
	3. I can effectively complete my work using this system 🗒	strongly disagree	○	○	○	○	○	○	○	strongly agree	○

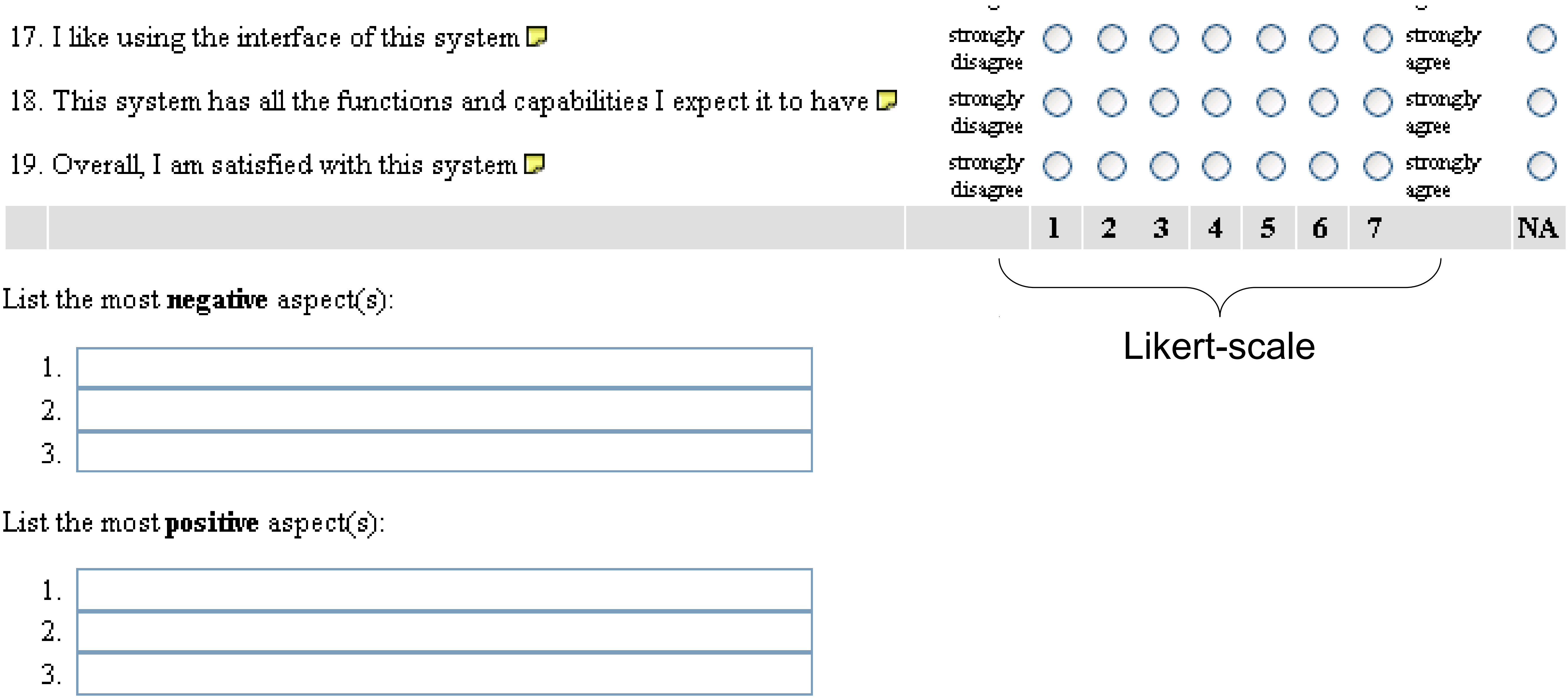
http://hcibib.org/perlman/question.cgi

Lewis, J. R. IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. In International Journal of Human-Computer Interaction 7 (1), 57-78, 1995.

Std. Example 2: IBM Usability Satisfaction Questionnaire

- Overall, I am satisfied with how easy it is to use this system
- It was simple to use this system
- I can effectively complete my work using this system
- I am able to complete my work quickly using this system
- I am able to efficiently complete my work using this system
- I feel comfortable using this system
- It was easy to learn to use this system
- I believe I became productive quickly using this system
- The system gives error messages that clearly tell me how to fix problems
- Whenever I make a mistake using the system, I recover easily and quickly
- The information (such as online help, on-screen messages, and other documentation) provided with this system is clear
- It is easy to find the information I needed
- The information provided for the system is easy to understand
- The information is effective in helping me complete the tasks and scenarios
- The organization of information on the system screens is clear
- The interface of this system is pleasant
- I like using the interface of this system
- This system has all the functions and capabilities I expect it to have
- Overall, I am satisfied with this system

Std. Example 2: IBM Usability Satisfaction Questionnaire



17. I like using the interface of this system	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
18. This system has all the functions and capabilities I expect it to have	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
19. Overall, I am satisfied with this system	strongly disagree	○	○	○	○	○	○	○	strongly agree	○
		1	2	3	4	5	6	7		NA

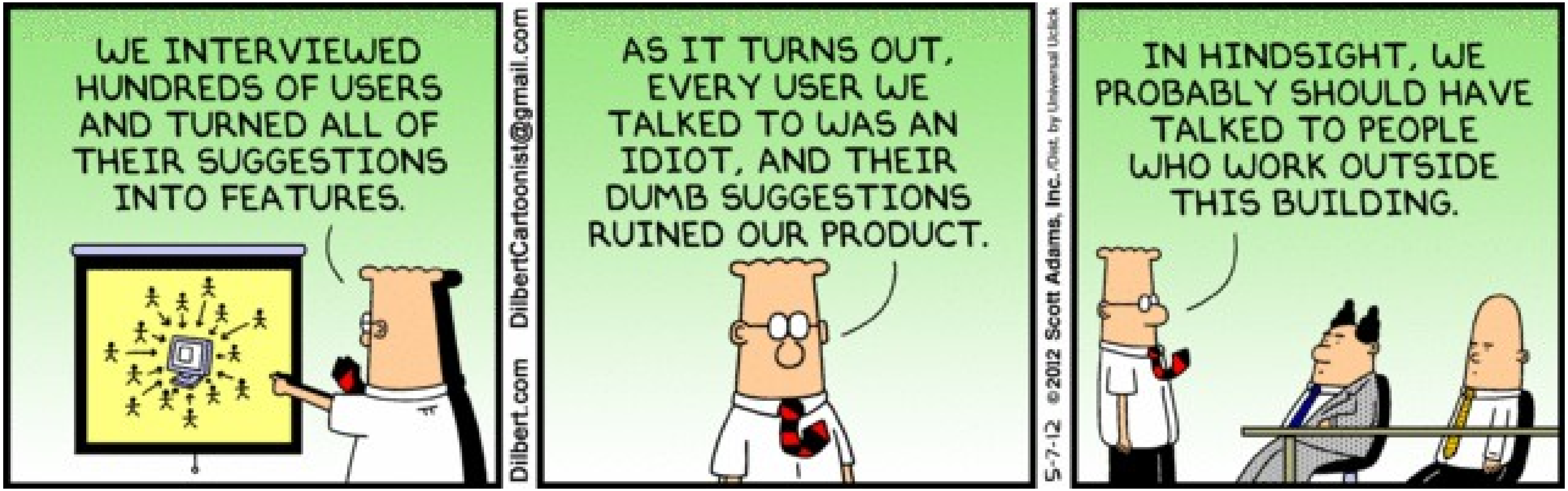
List the most **negative** aspect(s):

1.
2.
3.

List the most **positive** aspect(s):

1.
2.
3.

Likert, R. (1932). "A Technique for the Measurement of Attitudes". Archives of Psychology 140: 1–55.


WE INTERVIEWED HUNDREDS OF USERS AND TURNED ALL OF THEIR SUGGESTIONS INTO FEATURES.
Dilbert.com DilbertCartoonist@gmail.com
AS IT TURNS OUT, EVERY USER WE TALKED TO WAS AN IDIOT, AND THEIR DUMB SUGGESTIONS RUINED OUR PRODUCT.
5-7-12 © 2012 Scott Adams, Inc. /Dist. by Universal Uclick
IN HINDSIGHT, WE PROBABLY SHOULD HAVE TALKED TO PEOPLE WHO WORK OUTSIDE THIS BUILDING.

Analyzing Requirements

- Context of Requirements Analysis
- Analysis of Existing Systems
- Analyzing Ideas and Concepts
- **Work Processes, Bottom-Up**
- Work Processes, Top-Down
- Scenarios and Use Cases
- Conceptual Models

Diary Study

- A study that asks people to keep a diary, or journal, of their interactions with a computer system, any significant events or problems during their use of a system, or other aspects of their working life.
- A diary typically asks a user to record the date and time of an event, where they are, information about the event of significance, and ratings about how they feel, etc.
- An interesting alternative for making diary entries is to give users a tape recorder (or a mobile phone...) and a list of questions, so that users don't need to write things down as they encounter them.

(Usability glossary from www.usabilityfirst.com)

Examples for
real-world
work environments

Contextual Enquiry

- Investigating and understanding the users and their environment, tasks, issues, and preferences
  - Analyzing users’ needs
  - Related to task analysis

- Observing and interviewing users in their environment while they do their work
  - Done by visits in context

http://www.infodesign.com.au/usabilityresources/contextualenquiry
http://www.sitepoint.com/article/contextual-enquiry-primer

Ethnographic Observation in HCI - Interviews

- Prepare a set of questions beforehand
  - What do you want to know from the user?
- Tell people what you are doing
- Use capture (audio/video) if your communication partners agree
- If applicable, capture (take photos/video) material they use in their work (e.g., a manual, a checklist, the post-its around the screen)
- If possible summarize what your interview partner told you (to minimize misunderstandings)

Collecting Ideas from People

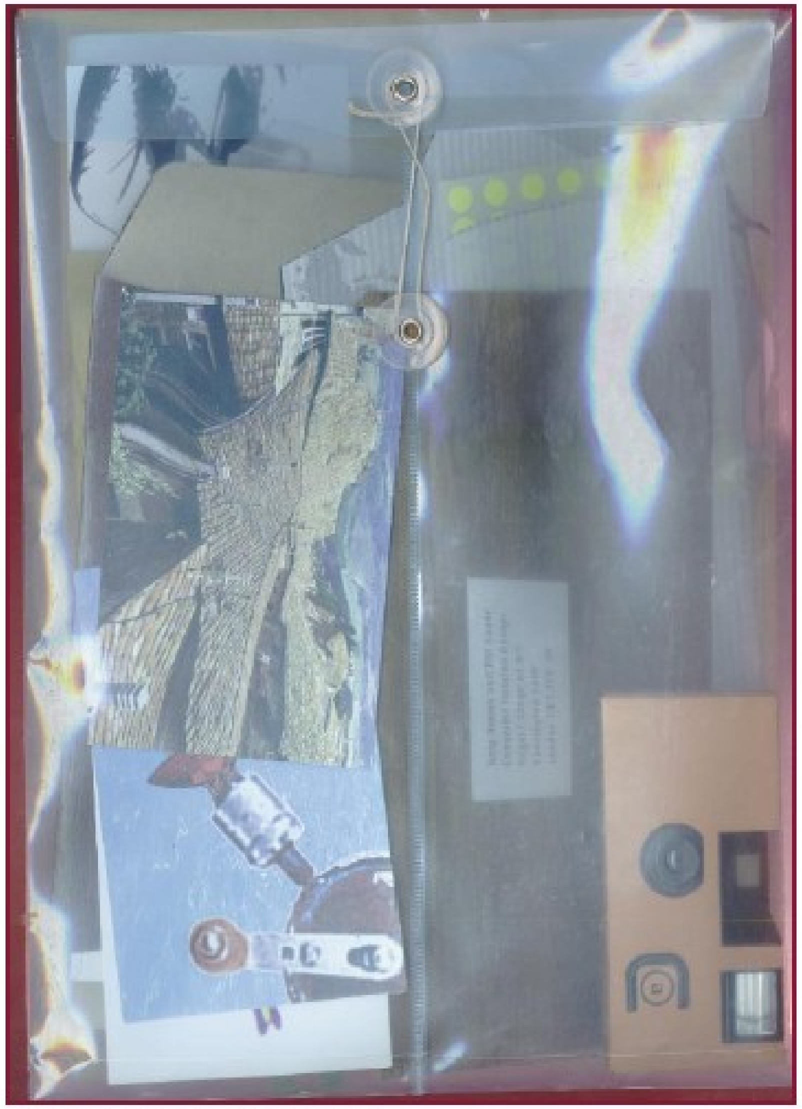


Figure 1. A cultural probe package.

- Cultural Probes
- Package of materials, e.g.,
  - Postcards
  - Disposable camera
  - Maps
  - Photo Album
  - Media diary
- Instructions for actions to be taken
- To provoke (contextual) inspirational responses from the users
- Over a period of time
- User centered inspiration

Gaver, W., Dunne, T., Pacenti, E.: Design: Cultural probes, ACM interactions 6(1), 1999

Cultural Probes (cont.)

- Be careful with trying to get concrete results
  - Summarizing collected data creates a non-existent average user
  - Summarizing removes unusual results that can be most inspiring
  - Open questions and tasks (even absurd ones) help getting surprising results
  - Analyses blur the connection between designer and user
  - Important aspects of cultural probes are imaginative engagement and story-telling which can be most useful for design

Gaver, W. W., Boucher, A., Pennington, S., and Walker, B. Cultural Probes and the Value of Uncertainty. Interactions 11, 5 (Sep. 2004), 53-56. 2004

Frameworks to Guide Observation

- The person: Who?
- The place: Where?
- The thing: What?

- The Goetz and LeCompte (1984) framework (“5W+H”):
  - Who is present?
    - What is their role?
  - What is happening?
  - When does the activity occur?
  - Where is it happening?
  - Why is it happening?
  - How is the activity organized?

Observations & Protocols

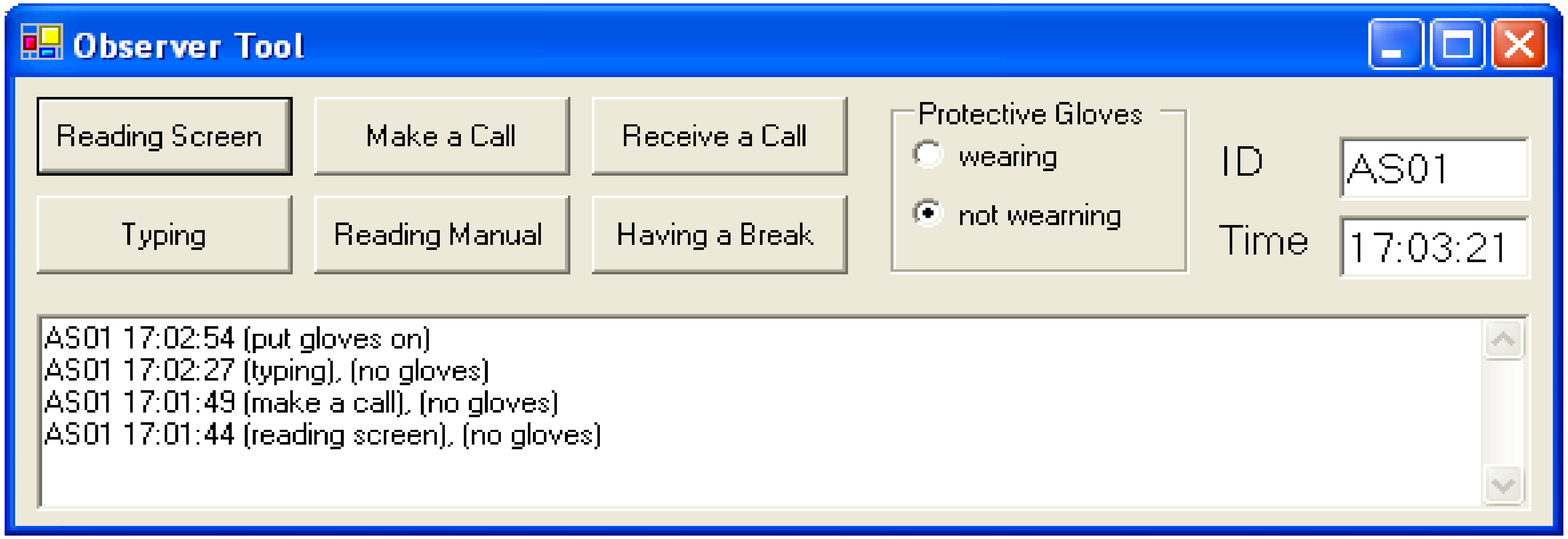
- Paper and pencil
  - Cheap and easy but unreliable
  - Make structured observations sheets / tool
- Audio/video recording
  - Including audio & still picture
  - Cheap and easy
  - Creates lots of data, potentially expensive to analyze
  - Good for review/discussion with the user
- Computer logging
  - Reliable and accurate
  - Limited to actions on the computer
  - Include functionality in the prototype / product
- User notebook/diary
  - Request to user to keep a diary style protocol

Structured Observations

- Observation sheet

time	typing	reading screen	consulting manual	phoning	…
14:00		X		X	
14:01	X		X		
14:02	X				
14:03	X				
14:04				X	
…					

Electronic version



Protective Gloves
○ wearing
● not wearning

ID: AS01
Time: 17:03:21

```
AS01 17:02:54 (put gloves on)
AS01 17:02:27 (typing), (no gloves)
AS01 17:01:49 (make a call), (no gloves)
AS01 17:01:44 (reading screen), (no gloves)
```

Wearable camera from
http://www.taserpromo.com/evidence-com/

Video Observation

- Observation is done with one or more cameras
- Cameras provide pictures of regions important to the task
- Camera attached to the user may be useful
  - Camera embedded into glasses
  - Allow the observer to see “through the eyes” of the user
- Different view points simultaneously
  - Camera overlooking the workplace
  - Camera looking from the screen to the user
  - Camera capturing what the user sees
- Analysis of raw material is very time consuming!
  - 3h to 20h for 1h recording
  - Automatically annotate video recordings (E.g., time stamps, possibly triggered by events)

Using Further Information Sources

- Sensors (e.g. motion, touch, RFID, …)
  - When did the person leave the room?
  - When did the person get something out of the shelf?
  - When did the person meet another person?
  - Where did the person go?

- Logfile of the interactive devices (e.g. key-logger, application logger)
- Log all the data (video, sensors, key input) with time stamps

- Use sensor information to find the video scenes that are of interest, e.g.,
  - Get me all video scenes that show what the user is doing before she/he switches to application X
  - Show me all sequences where users have to input a password

Data Analysis for Observations

- Qualitative data - interpreted
  - Used to tell the ‘story’ about what was observed
  - Key events, patterns of behavior
  - Include quotes, pictures, anecdotes in report
- Qualitative data - categorized
  - Using techniques such as content analysis
  - “Interpolation” between different data sources
- Quantitative data
  - Collected from interaction & video logs.
  - Presented as values, tables, charts, graphs and treated statistically
  - To be used with care! (Is the information basis representative?)

Analyzing Requirements

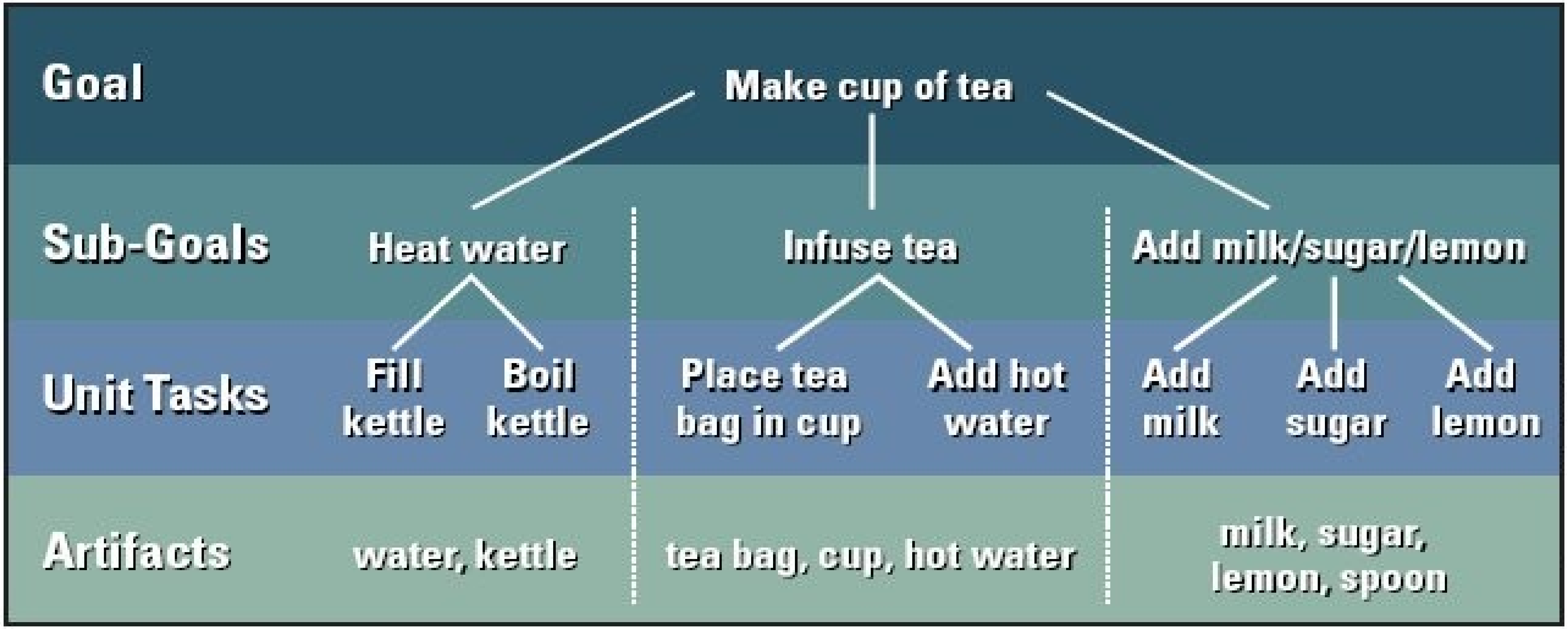
- Context of Requirements Analysis
- Analysis of Existing Systems
- Analyzing Ideas and Concepts
- Work Processes, Bottom-Up
- **Work Processes, Top-Down**
- Scenarios and Use Cases
- Conceptual Models

Task Analysis - Motivation

- Activities in daily life are driven by goals
  - E.g. “I want to show the pictures on my computer screen to the whole audience”
- Sequences of actions can be quite detailed
  - E.g. for setting up a video projector
    - unpacking the projector and placing it on the table
    - connecting the power cable to the projector and the socket
    - connecting a data cable between projector and computer
    - switching on the projector
    - waiting for the projector to be ready
    - switching the computer to dual screen mode
- Pure observation may miss key points
  - Equivalent sequences of actions, variants in order of actions, granularity ...

Task Analysis – Example

- William Hudson. HCI and the Web: a Tale of Two Tutorials: a Cognitive Approach to Interactive System Design and Interaction Design Meets Agility. ACM interactions 12(1), 2005, 49-51



Goal	Make cup of tea		
Sub-Goals	Heat water	Infuse tea	Add milk/sugar/lemon
Unit Tasks	Fill kettle Boil kettle	Place tea bag in cup Add hot water	Add milk Add sugar Add lemon
Artifacts	water, kettle	tea bag, cup, hot water	milk, sugar, lemon, spoon

Task Analysis – High level Questions

- How do users know their goal is attainable?
- How do users know what to do?
  - Analyze what the user has (or users have) to do in order to get a job done
    - What (physical) actions are done?
    - What cognitive processes are required?
    - What information is used?
    - What information is created?

Gulf of Execution!

- How will users know they have done the right thing?
- How will users know they have attained their goal?

Gulf of Evaluation!

- Task analysis is usually in the context of an existing system or for a established procedure
- The analysis is most often hierarchical
  - Task, sub task, sub sub task …
  - Understand how a task is composed of sub tasks

Task Analysis – How To?

- Task decomposition is at the center of the method
  - Identify high level tasks
  - Break them down into the subtasks and operations
- Task flows and alternatives
  - Identify for elementary subtasks their order (task flow)
  - Identify alternative subtasks
  - Understand and document decision processes (how are alternative subtasks chosen?)
- Present the result of the task analysis as chart
  - Charts may have different levels (overview and detailed subtasks)
  - Show sequences, alternatives, ordering in the diagram
- Questions that help in decomposition of tasks
  - How is the task done?
  - Why is the user doing this task?

http://www.usabilitynet.org/tools/taskanalysis.htm

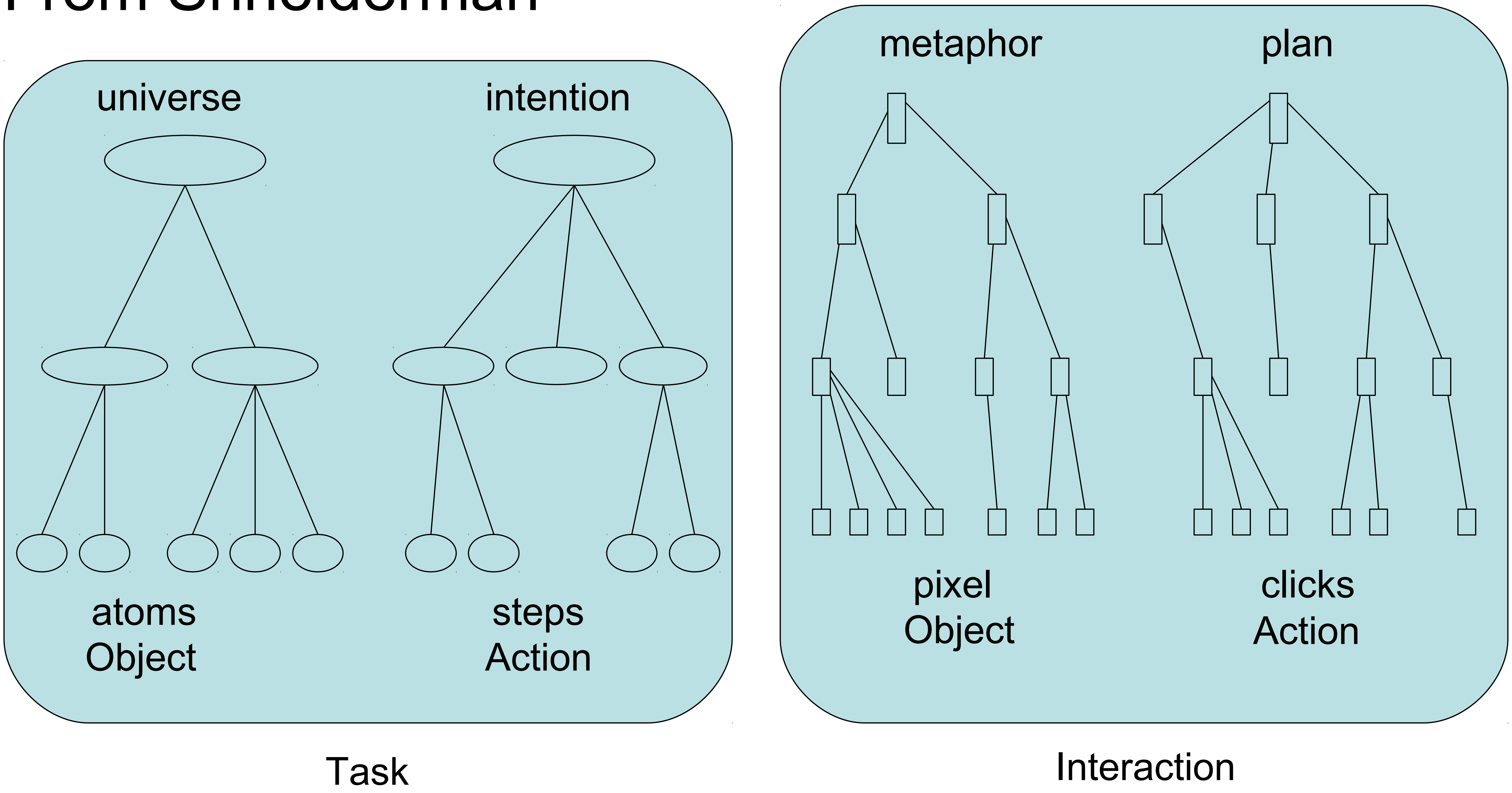
Action-Object vs. Object-Action

- Universal duality between Object & Action
  - Shall we name the object first and look for an adequate action?
  - Shall we name the action first and look for an adequate object?
  - Two different ways to structure the world ...
- For “task analysis”:
  - Implicit assumption of action-first approach?
  - More “object-oriented” alternative?
- Advantages of an object-based approach:
  - Easier to adapt to new tasks
  - Tasks are in general more easily changed/removed/added than objects we are working with
  - Better fit with human techniques for structuring complex situations
    - Generalization/specialization, Part-of hierarchies

A. Khella: Objects-Actions Interface Model
http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/oai.html

Mapping Human Tasks to Man-Computer Interaction

- From Shneiderman



Analyzing Requirements

- Context of Requirements Analysis
- Analysis of Existing Systems
- Analyzing Ideas and Concepts
- Work Processes, Bottom-Up
- Work Processes, Top-Down
- **Scenarios and Use Cases**
- Conceptual Models

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

“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 be too 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?”

Example: Day in a Life Scenario

A day with IYOUIT…

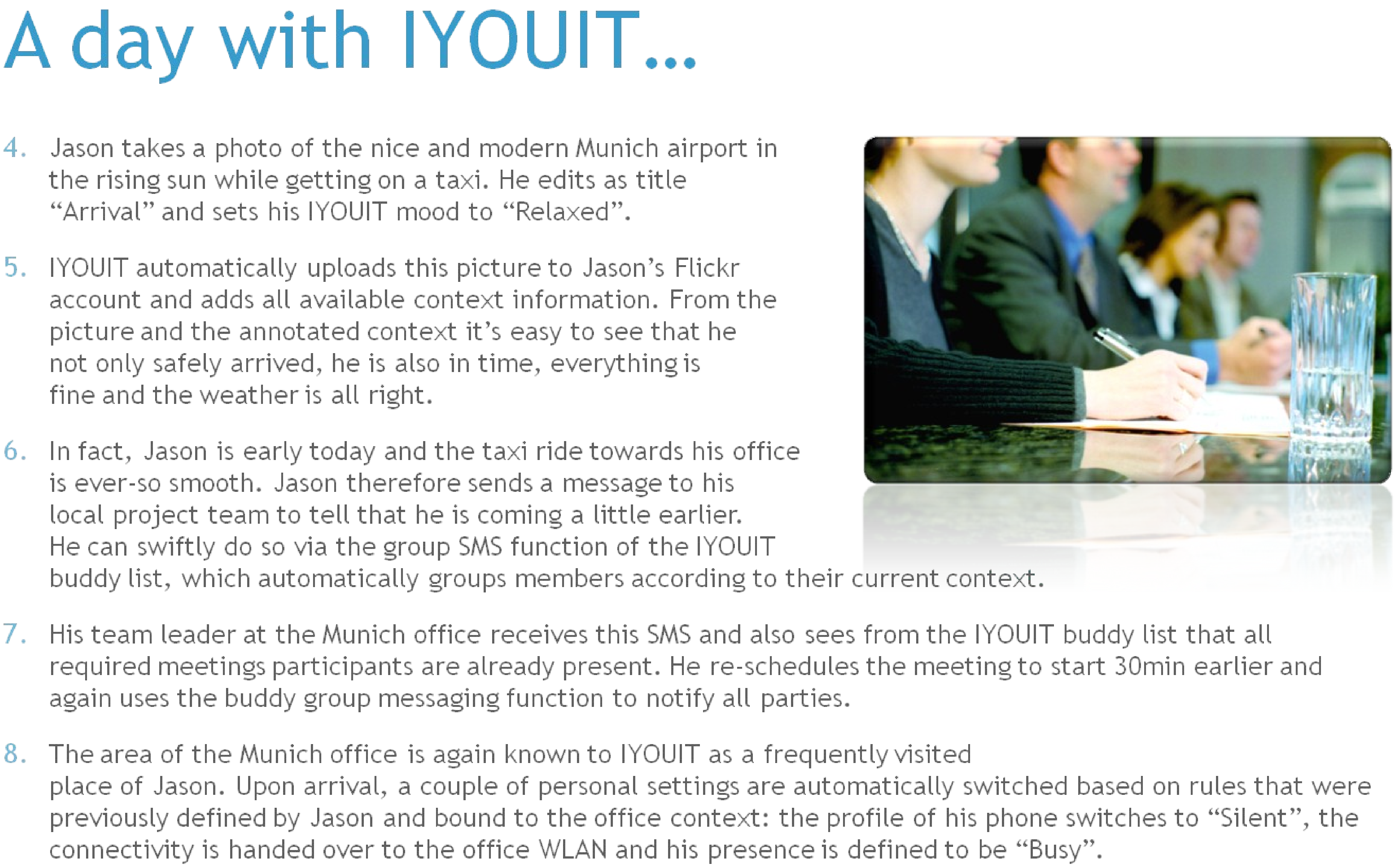
Read this story to follow Jason and his family to see how IYOUIT simplifies their life.

For a while now Jason is working for this Web2.0 start-up with offices in Paris and Munich. He lives in Paris together with his wife Kelly and his son Bob. For project and business appointments he goes to Munich on a regular basis.

Today is one of those days.

1. After an early flight he arrives at Munich airport, which he visited many times in the past. As he frequently comes here, the airport area had been automatically detected as a frequent place of stay by IYOUIT.
2. In IYOUIT, Places can be assigned a name and meaning. Jason is referring to this place as an airport named “MUC”. This piece of information automatically becomes part of his personal context as he goes through MUC together with other IYOUIT context items that can be acquired (Locations, Places, Experiences, Photos, Sounds, Observations, Products, Weather and Tags).
3. Through the IYOUIT Buddy list personal context items are shared with others as you go. This way Kelly, with a particular buddy view on family members, can observe Jason’s arrival in Munich on her mobile phone. It’s good to know that he safely arrived!

Example: Day in a Life Scenario



A day with IYOUIT...

4. Jason takes a photo of the nice and modern Munich airport in the rising sun while getting on a taxi. He edits as title "Arrival" and sets his IYOUIT mood to "Relaxed".
5. IYOUIT automatically uploads this picture to Jason's Flickr account and adds all available context information. From the picture and the annotated context it's easy to see that he not only safely arrived, he is also in time, everything is fine and the weather is all right.
6. In fact, Jason is early today and the taxi ride towards his office is ever-so smooth. Jason therefore sends a message to his local project team to tell that he is coming a little earlier. He can swiftly do so via the group SMS function of the IYOUIT buddy list, which automatically groups members according to their current context.
7. His team leader at the Munich office receives this SMS and also sees from the IYOUIT buddy list that all required meetings participants are already present. He re-schedules the meeting to start 30min earlier and again uses the buddy group messaging function to notify all parties.
8. The area of the Munich office is again known to IYOUIT as a frequently visited place of Jason. Upon arrival, a couple of personal settings are automatically switched based on rules that were previously defined by Jason and bound to the office context: the profile of his phone switches to "Silent", the connectivity is handed over to the office WLAN and his presence is defined to be "Busy".

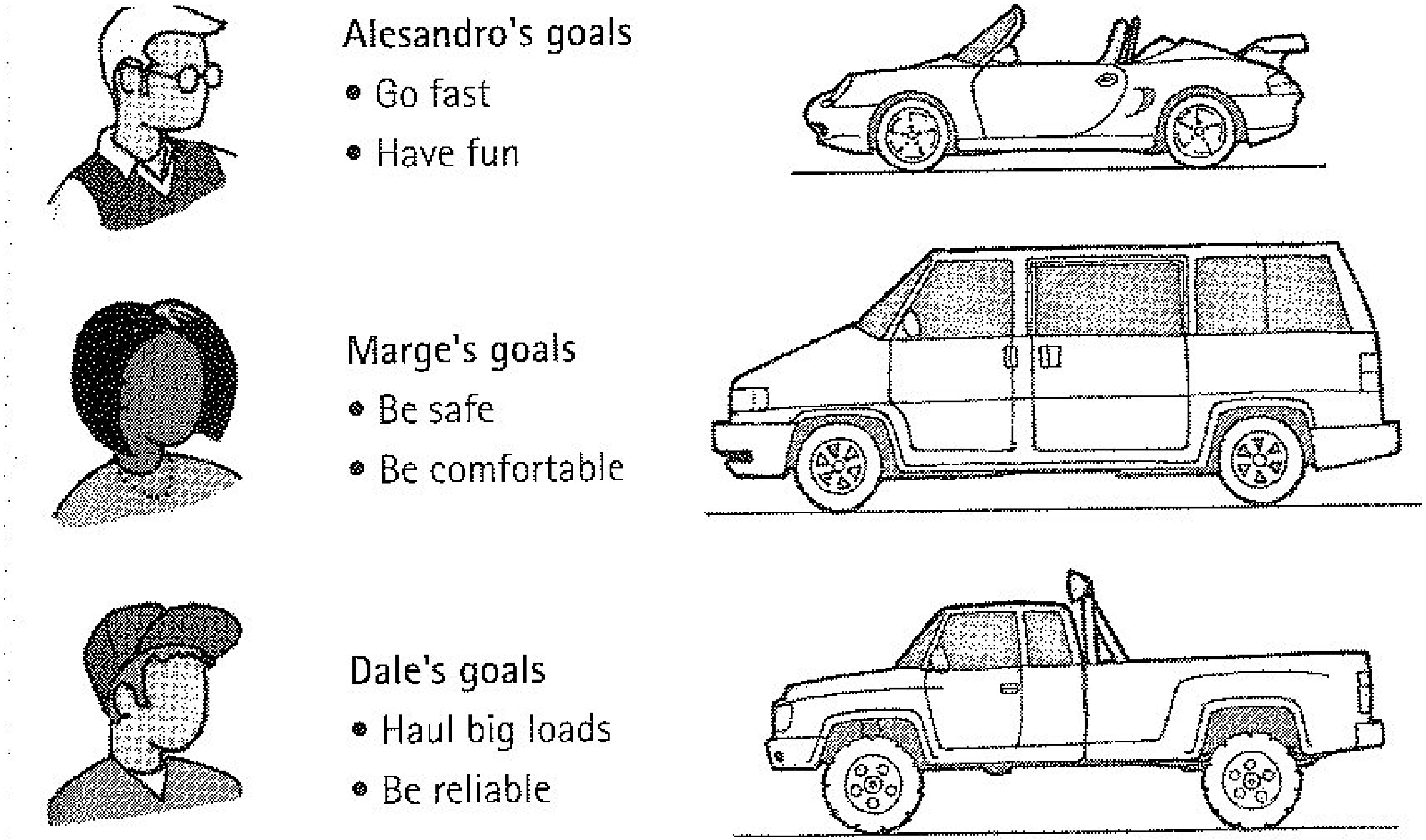
Situation Scenarios

- 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

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!

Persona Examples (1)



Alesandro's goals

- Go fast
- Have fun

Marge's goals

- Be safe
- Be comfortable

Dale's goals

- Haul big loads
- Be reliable

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)

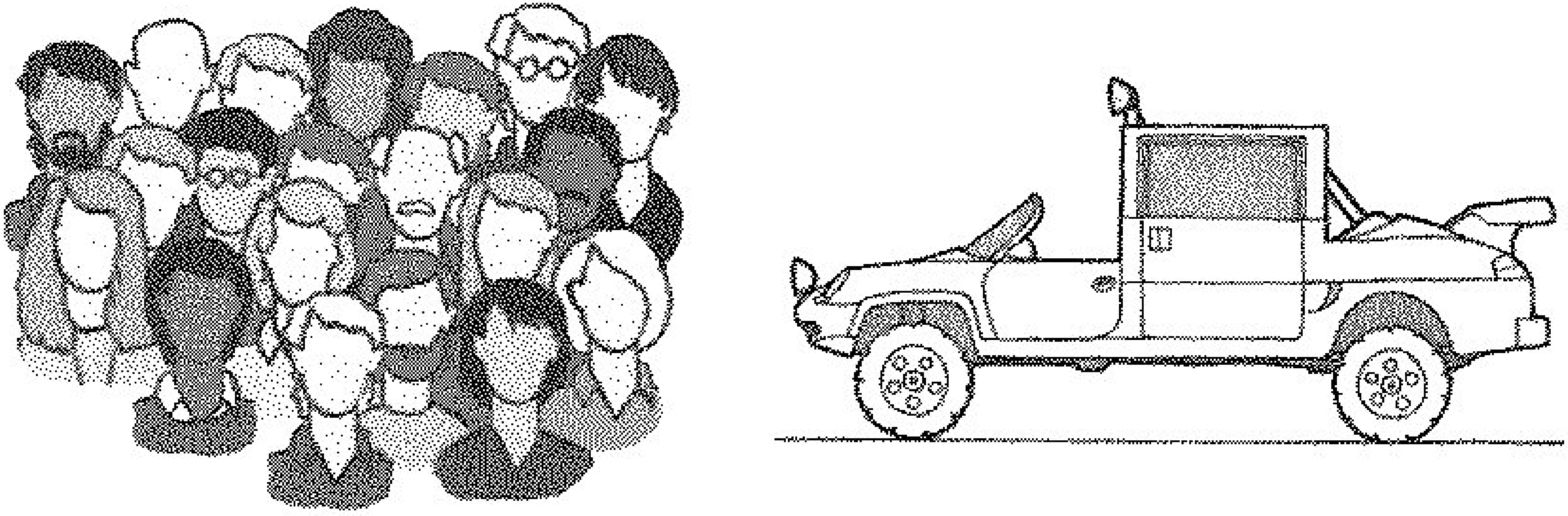


Figure 5-1: A simplified example of how personas are useful. If you try to design an automobile that pleases every possible driver, you end up with a car with every possible feature, but which pleases nobody. Software today is too often designed to please too many users, resulting in low user satisfaction.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

A. Cooper

Scenario Development: Why Persona?

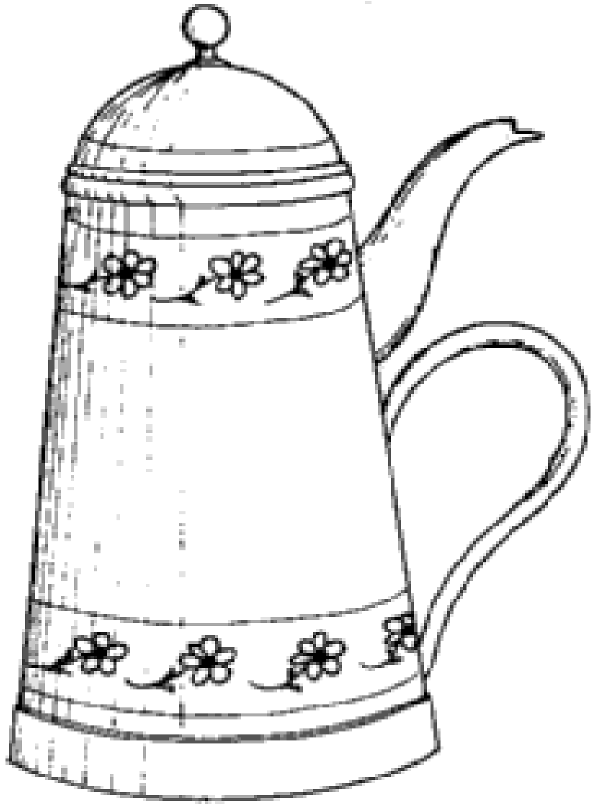
- Avoiding the “elastic user”
  - If you do not specify the user you can change their abilities to support a design decision made = “elastic user”
- Avoiding self-referential design
  - The designer or developer often assumes (implicitly) that users have his goals and his skills and abilities.
- Avoiding 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)

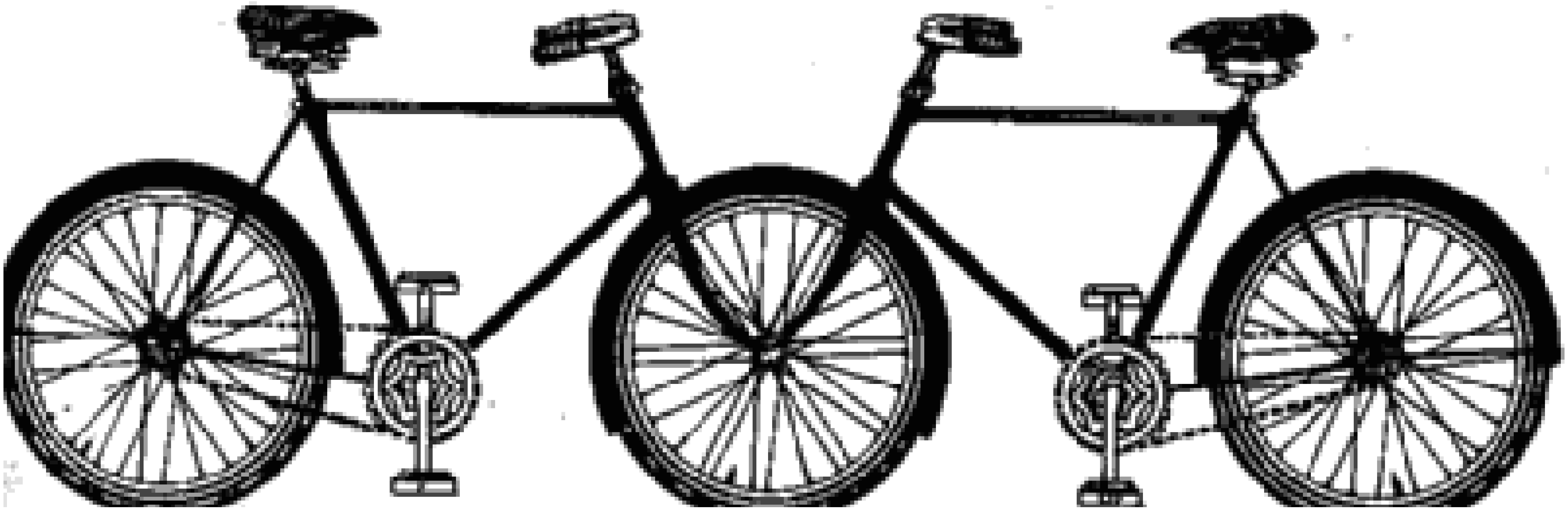
Analyzing Requirements

- Context of Requirements Analysis
- Analysis of Existing Systems
- Analyzing Ideas and Concepts
- Work Processes, Bottom-Up
- Work Processes, Top-Down
- Scenarios and Use Cases
- **Conceptual Models**

Motivation: Conceptual Models

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





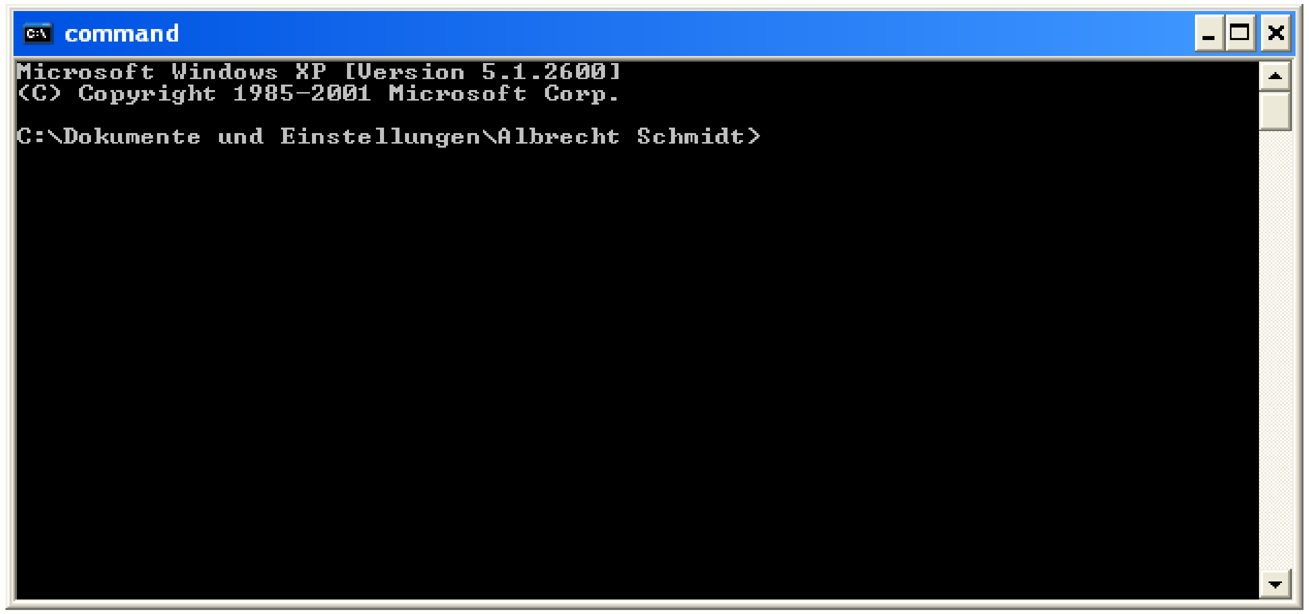
Images from: D. Norman, The Design of everyday things.

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.

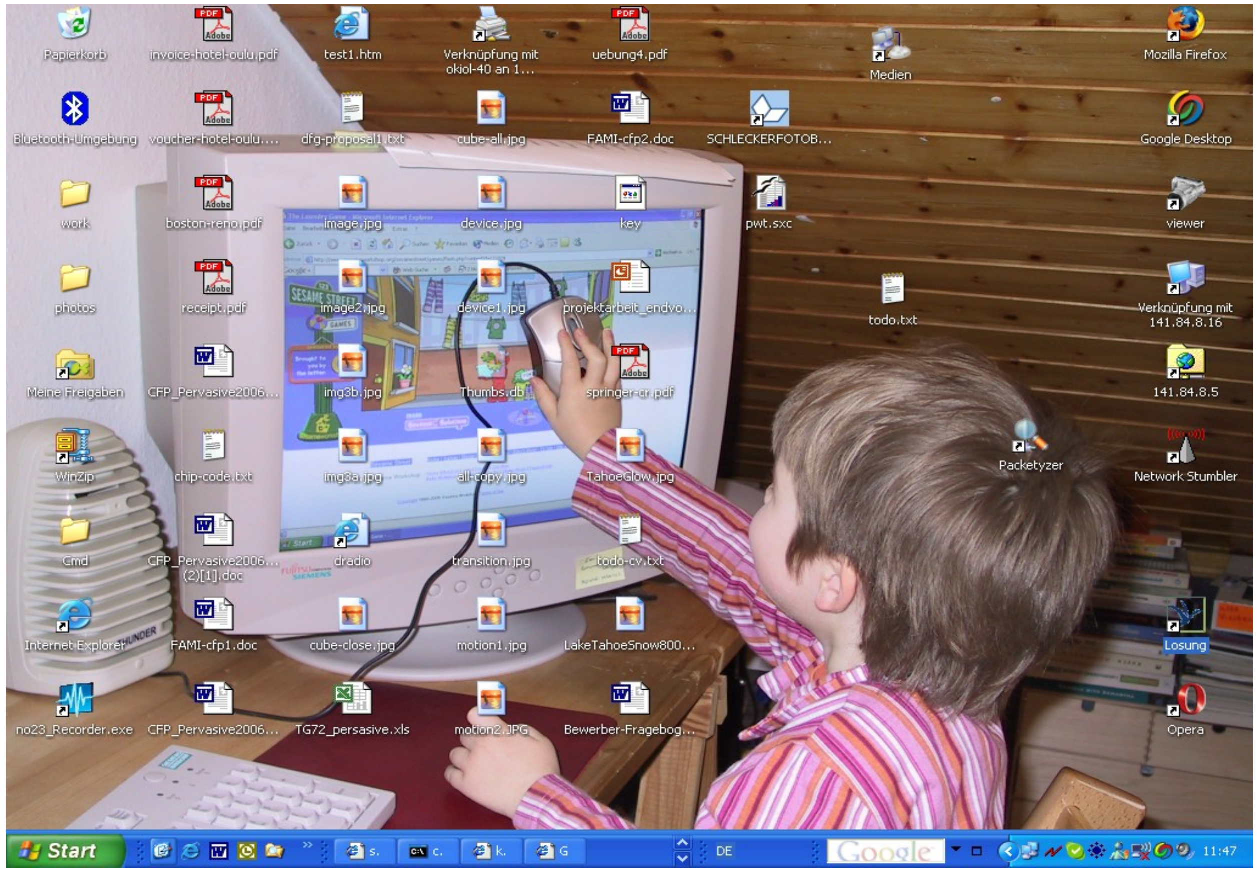
Low affordance



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Dokumente und Einstellungen\Albrecht Schmidt>
```

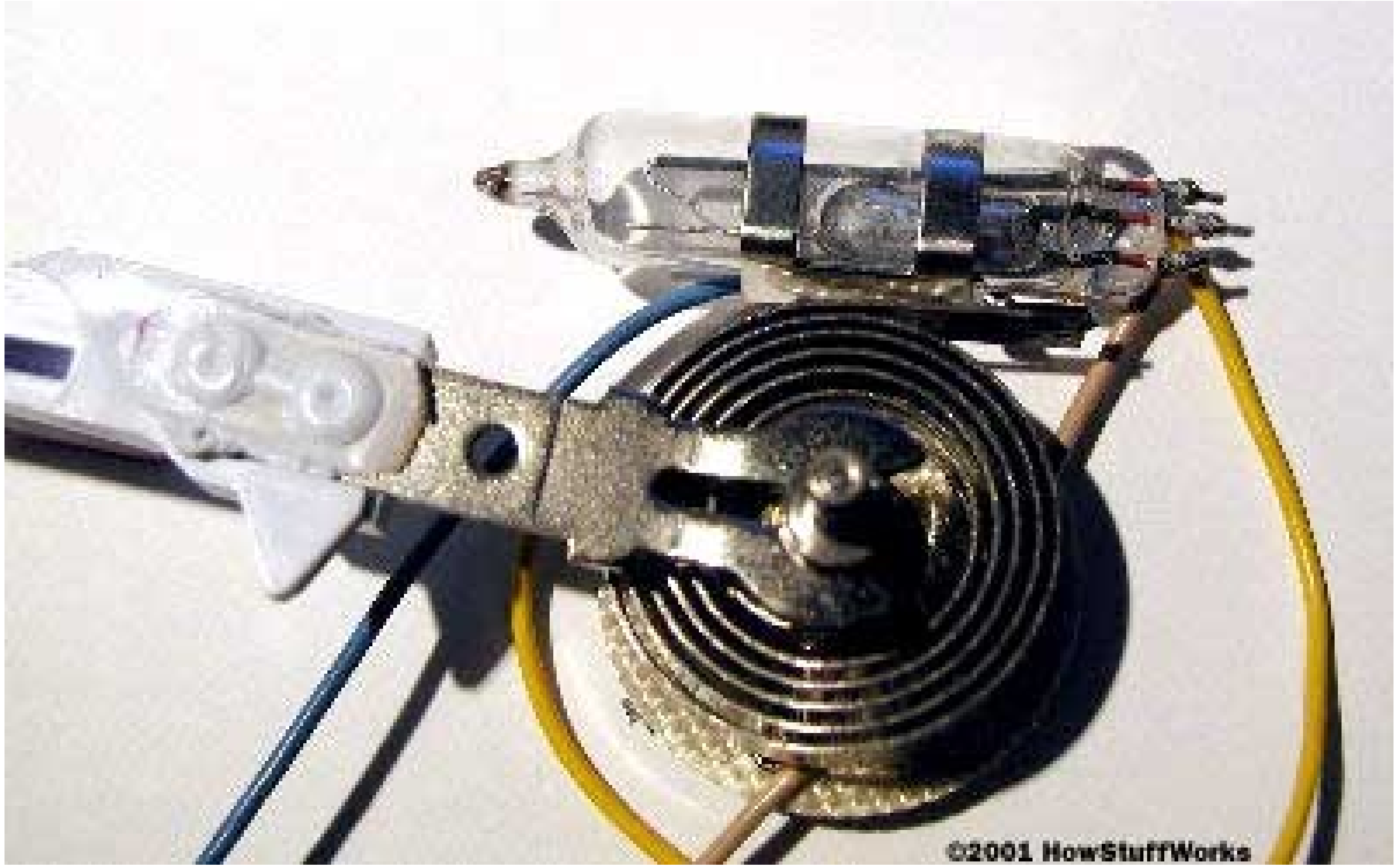
High affordance



Wrong affordance

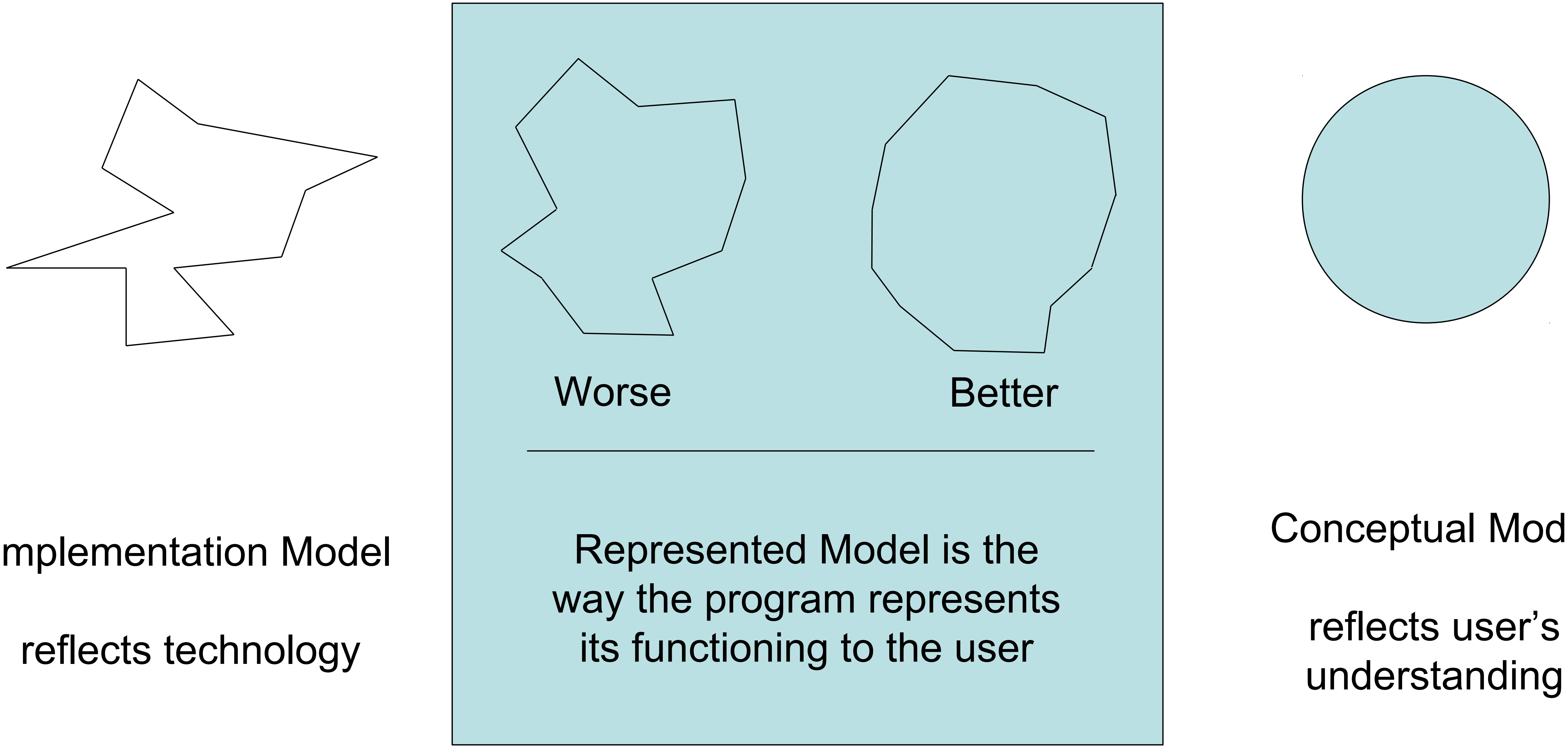
Example: Heating Control

- 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



©2001 HowStuffWorks

Implementation, Represented, Conceptual Model



Worse

Better

Implementation Model

reflects technology

Represented Model is the way the program represents its functioning to the user

Conceptual Model

reflects user's understanding

From A. Cooper, About Face 2.0

Example: ‘Geldkarte’ (1)

Store cash on the card

Geldautomat
Deutsche Bank
GeldKarte
Konto-Nr.
Karten-Nr.
Gültig bis Ende

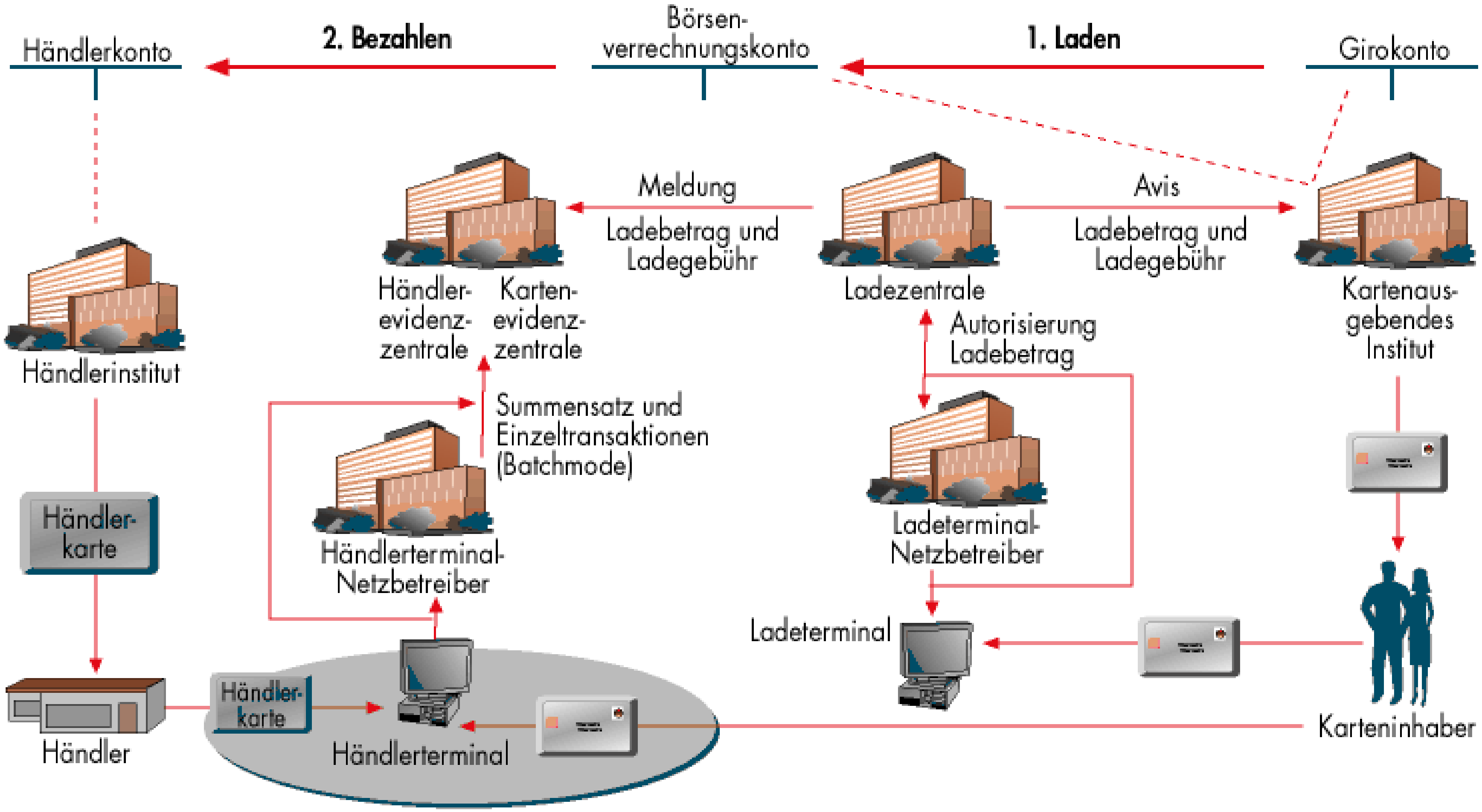
Pay with the card

Deutsche Bank
GeldKarte
Konto-Nr.
Karten-Nr.
Gültig bis Ende

Conceptual Model – by the user

Example: ‘Geldkarte’ (2)

- Some aspects of the implementation model



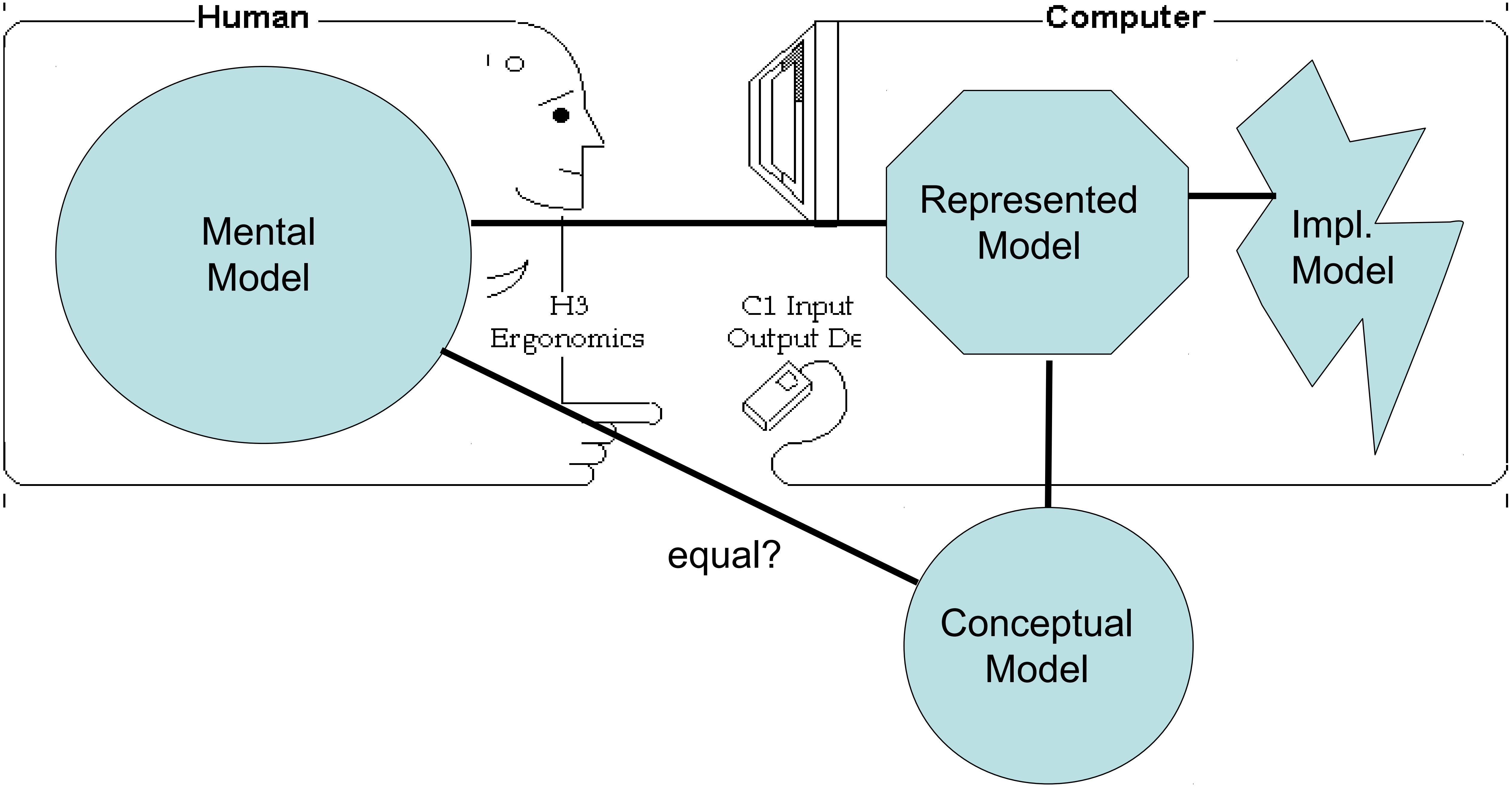
From IX-Article: Chipgeld by Hans-Bernhard Beykirch, http://www.heise.de/ix/artikel/1998/12/148/

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

- Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale: Human Computer Interaction (3rd edition), Prentice Hall 2003
- Jennifer Preece, Yvonne Rogers, Helen Sharp: Interaction Design, Wiley 2002
- M. D. LeCompte, J. Preissle: Ethnography and qualitative design in educational research (2nd ed.), Academic Press 1993
- Gary Gaffney: Contextual enquiry - a primer, www.sitepoint.com
- William Hudson. HCI and the web: A tale of two tutorials: a cognitive approach to interactive system design and interaction design meets agility, ACM interactions 12(1):2005, 49-51
- A. Khella: Objects-Actions Interface Model http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/oai.html
- Alan Cooper, Robert M. Reimann. (2007) About Face 3.0: The Essentials of Interaction Design; ISBN 978-0470084113
- D. Norman. The Psychology of Everyday Things, B&T 1989
- K. Bittner, I. Spence: Use Case Modeling, Addison-Wesley 2002