Vorlesung
Mensch-Maschine-Interaktion

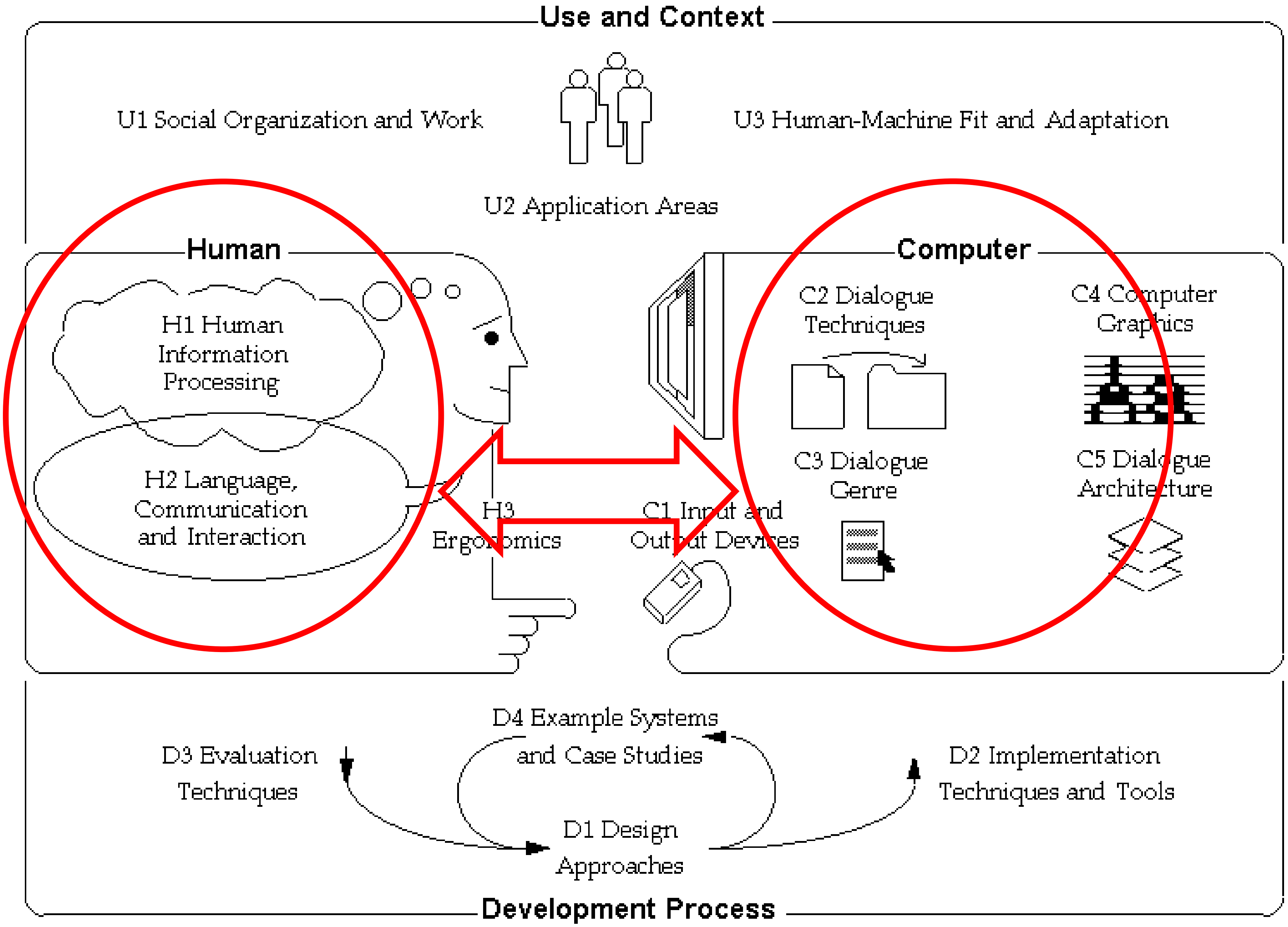
Ludwig-Maximilians-Universität München

LFE Medieninformatik

Andreas Butz & Albrecht Schmidt

WS2004/2005

http://www.medien.ifi.lmu.de/



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

Development Process

D1 Design Approaches

D2 Implementation Techniques and Tools

D3 Evaluation Techniques

D4 Example Systems and Case Studies

from ACM SIGCHI Curricula for HCI

What the User Sees

- Users see only what is visible!

What the Developer Knows

- Users see only what is visible!
- users have little idea about:
  - architecture,
  - state transitions,
  - dependencies
  - application context
  - system restrictions
  - …

Guidance for the designer
(Shneidermann, 97)

- Systematic approach is needed

- High-level theories and model
- Middle-level principles
- Specific and practical guidelines

Models & Theories

- What are models and theories used for?
  - explanatory
  - predictive
  - descriptive/taxonomy

- Models on different levels
  - keystroke
  - dialog
  - …
  - concept
  - human action

- What is modelled?
  - user
  - task
  - dialogs
  - transitions
  - software
  - input/output
  - system
  - interaction
  - behaviour
  - …
  - combination of these

Models and Theories

- There are plenty! We will cover some of them…
  - Seven Stages of Action
  - Seeheim Model
  - Conceptual, semantic, syntactic and lexical
  - GOMS and Keystroke
  - Object-Action Interface Model
  - PAC Model
  - Arch Model
  - MVC Concept
  - …
- Looking at a selection of them to understand
  - What models are used for
  - How models are applied
  - How they help to improve the design/development process

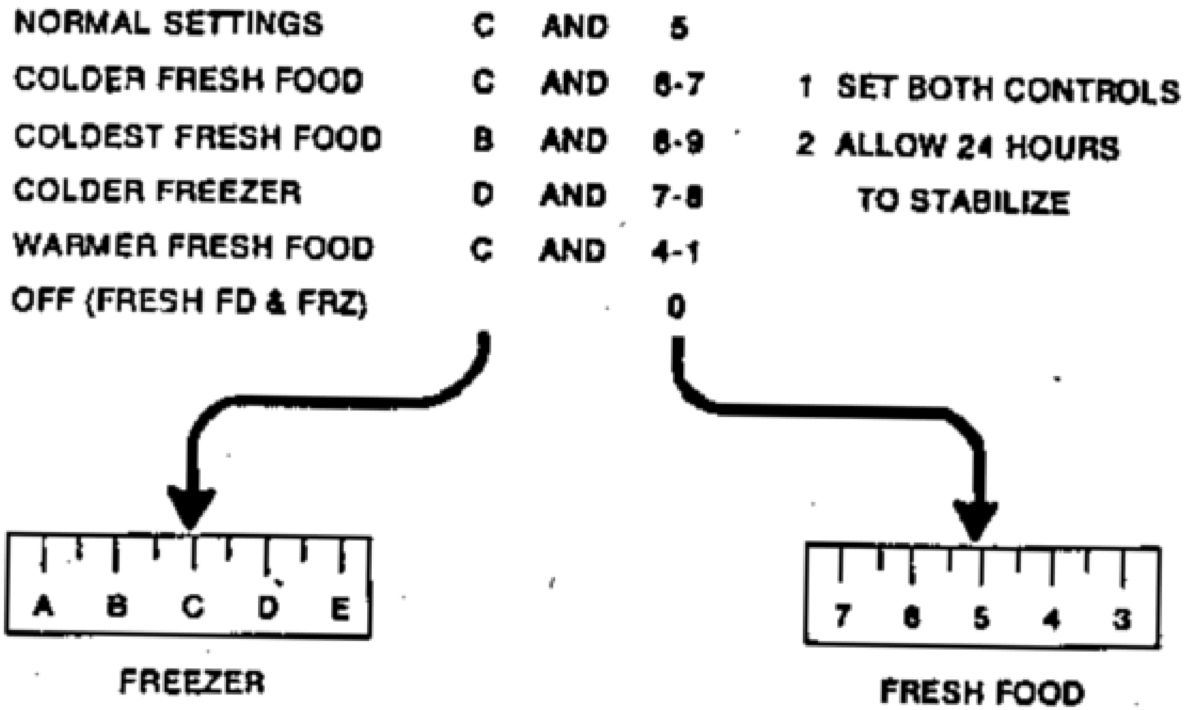
Background: The Psychology of Everyday Things (Norman 2002, Chapter 1)

- 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.

Explaining Conceptual Models Example –Refrigerator



NORMAL SETTINGS	C	AND	5
COLDER FRESH FOOD	C	AND	6-7
COLDEST FRESH FOOD	B	AND	8-9
COLDER FREEZER	D	AND	7-8
WARMER FRESH FOOD	C	AND	4-1
OFF (FRESH FD & FRZ)			0

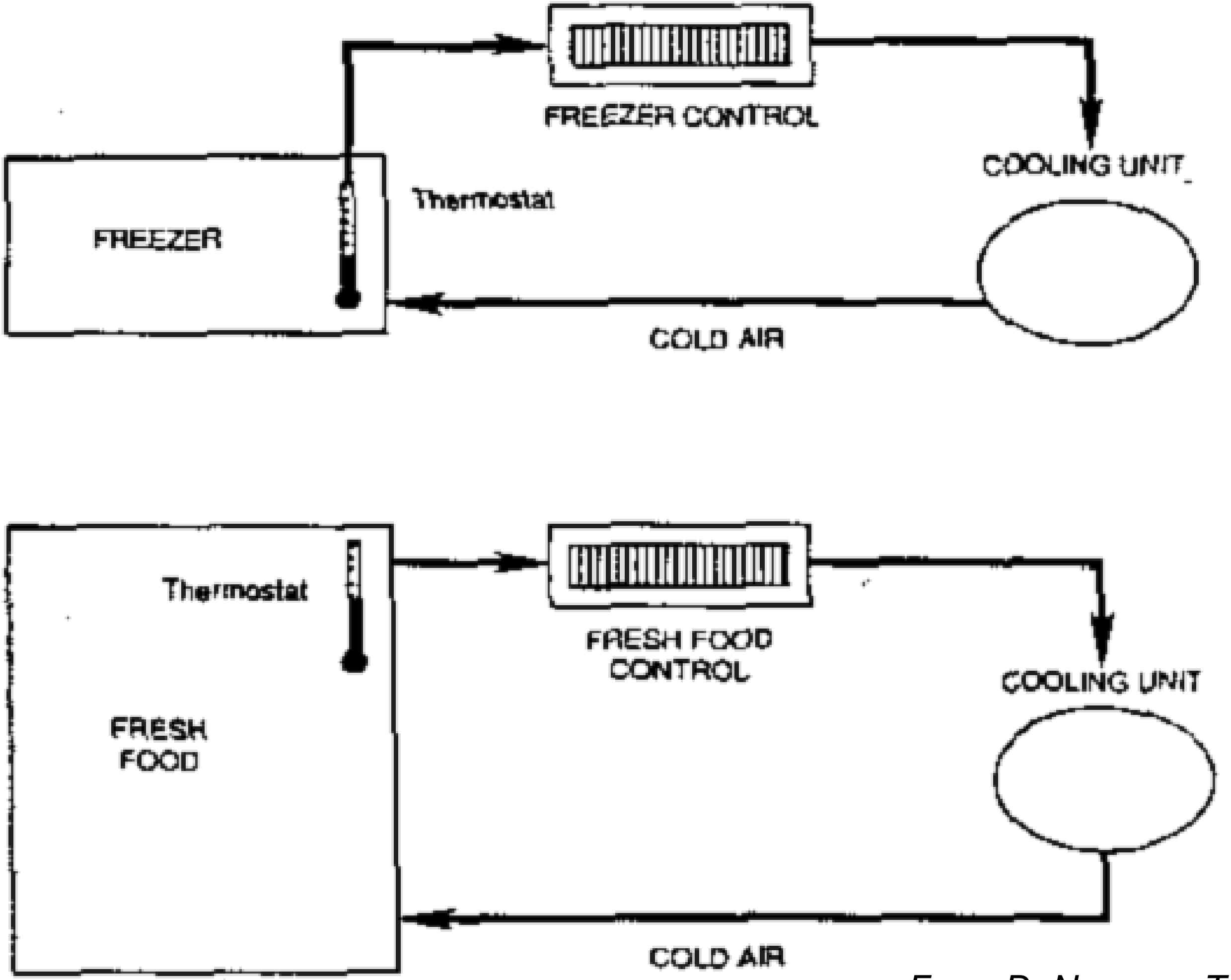
1 SET BOTH CONTROLS

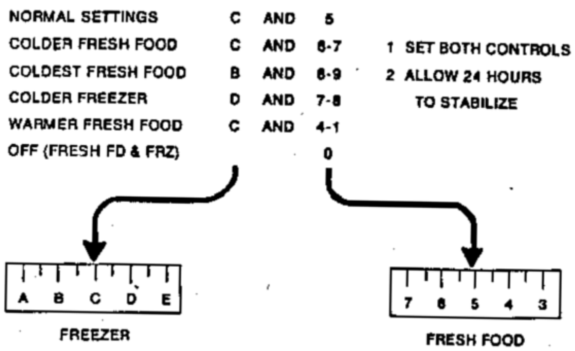
2 ALLOW 24 HOURS
TO STABILIZE

- 2 controls
- Freezer
- Fridge

From D. Norman, The Psychology of Everyday Things.

Example – Refrigerator Conceptual Model 1



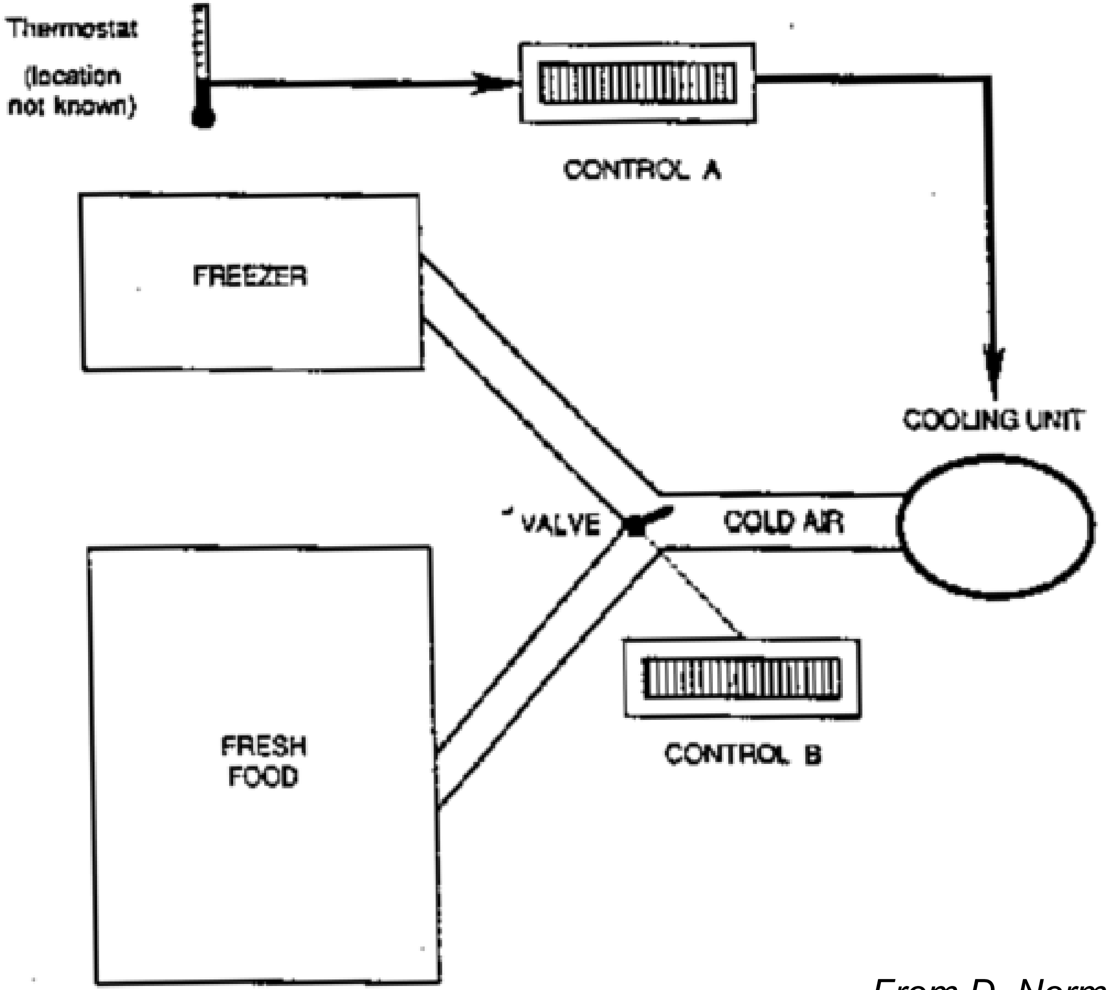


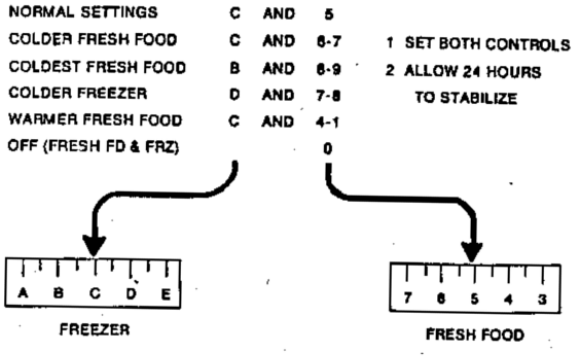
NORMAL SETTINGS	C	AND	5	
COLDER FRESH FOOD	C	AND	6-7	1 SET BOTH CONTROLS
COLDEST FRESH FOOD	B	AND	8-9	2 ALLOW 24 HOURS TO STABILIZE
COLDER FREEZER	D	AND	7-8	
WARMER FRESH FOOD	C	AND	4-1	
OFF (FRESH FD & FRZ)			0	

- Idea 1: 2 cooling units
- One control each

From D. Norman, The Psychology of Everyday Things.

Example – Refrigerator Conceptual Model 2





NORMAL SETTINGS	C	AND	5
COLDER FRESH FOOD	C	AND	6-7
COLDEST FRESH FOOD	B	AND	8-9
COLDER FREEZER	D	AND	7-8
WARMER FRESH FOOD	C	AND	4-1
OFF (FRESH FD & FRZ)			0

1 SET BOTH CONTROLS
2 ALLOW 24 HOURS TO STABILIZE

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

From D. Norman, The Psychology of Everyday Things.

Informal Exercise: Understand Conceptual Models

- Talk to “non-technical” people and try to understand their conceptual model for the following systems
  - Ordering a book from an online bookshop
  - Finding and reading information on the WWW on a particular topic using a search engine
  - Sending an email to someone who is traveling

- Hints to the conceptual model are often provided by
  - Observing what constraints on usage people apply (e.g. you have to do step x before step y)
  - How people explain errors (e.g. assuming the mental model does no include DNS – it is interesting to find out how people explain errors cause by failure of this component)

Understandability and Usability

- Principles of Design (Norman, 2002)
  1. Provide a good conceptual model
  2. Make things visible

- A conceptual model is used to predict the effect of actions performed. The conceptual model is based on:
  - Affordances - basic properties of the device/system
  - Constraints - possible actions that can be performed
  - Mapping - relationship between controls and outcome
  - Experience - knowledge acquired that is related to the domain

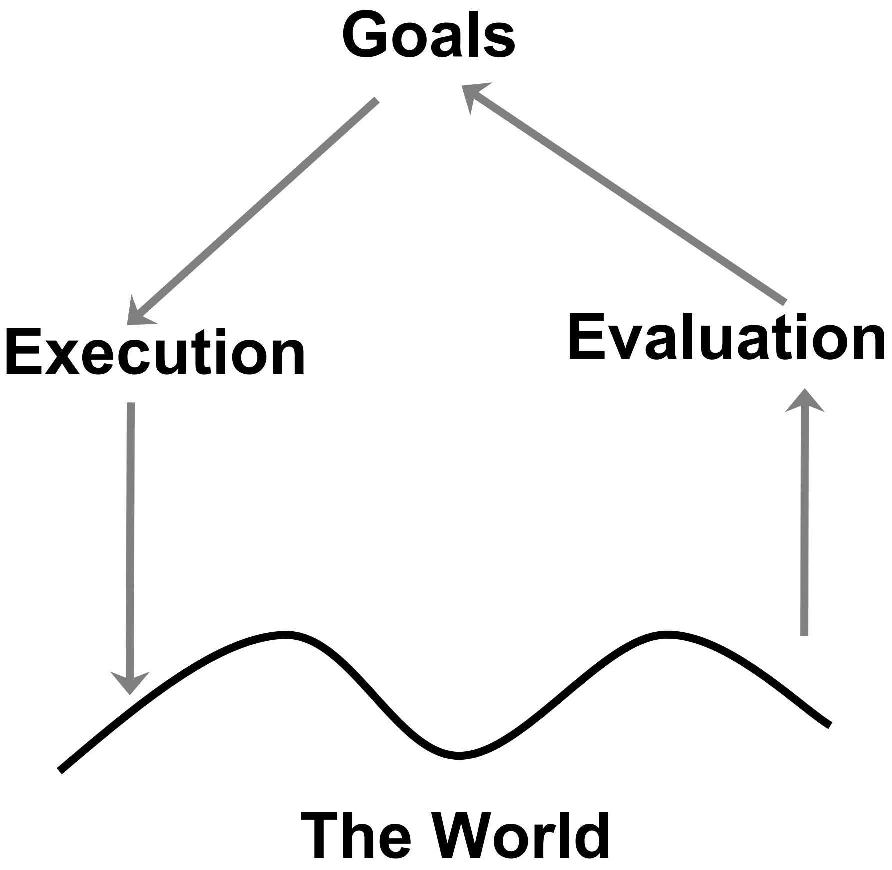
- Visibility relates also to mappings and feedback
  - Provide a control for each function (direct mapping)
  - Make actions and reactions visible (feedback)

Background: The Psychology of Everyday Action (Norman 2002, Chapter 2)

- People are blaming themselves for problems caused by design
  - If the system crashes and the user did everything as he is supposed to do the developer/system is blamed
  - If the system crashes and the user operated the system wrongly the user is blamed
- People have misconceptions about their actions
  - The model must not be fully correct – it must explain the phenomenon
- People try to explain actions and results
  - Random coincidence may lead to assumptions about causality

Action Cycle

- The action is goal directed
  - What do we want to happen?
  - What is the desired state?
- Human action has two major aspects
  - Execution: what we do to the world
  - Evaluation: compare if what happens is what we want



Action Cycle
Stages of Execution

- Goal

  translated into

- An intention to act as to achieve the goal

  translated into

- The actual sequence of actions that we plan to do

  translated into

- The physical execution of the action sequence

Action Cycle
Stages of Evaluation

- Perceiving the state of the worlds

  followed by

- Interpreting the perception according to our expectations

  followed by

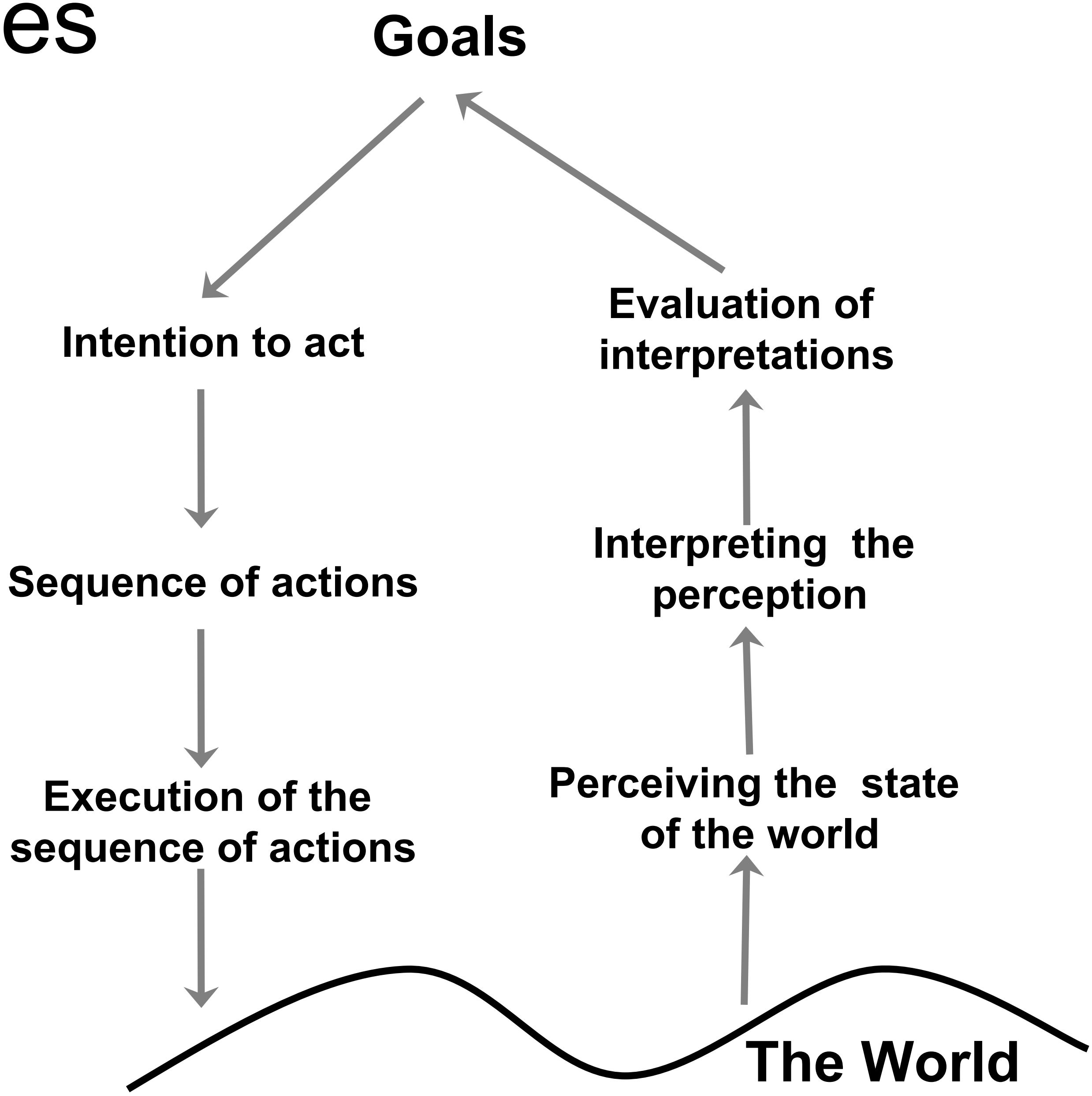
- Evaluation of the interpretations with what we expected to happen (original intentions)

  followed by

- Goal

Seven Stages of Action

1. Forming a goal
2. Forming an intention
3. Specifying an action
4. Executing the action
5. Perceiving the system state
6. Interpreting the system state
7. Evaluating the outcome



Goals

Intention to act

Sequence of actions

Execution of the sequence of actions

Evaluation of interpretations

Interpreting the perception

Perceiving the state of the world

The World

Gulf of Execution

- The difference between the intentions and the allowable actions is the Gulf of Execution
  - How directly can the actions be accomplished?
  - Do the actions that can be taken in the system match the actions indented by the person?
- Example in GUI
  - The user wants a document written on the system in paper (the goal)
  - What actions are permitted by the system to achieve this goal?
- Good design minimizes the Gulf of Execution

Gulf of Evaluation

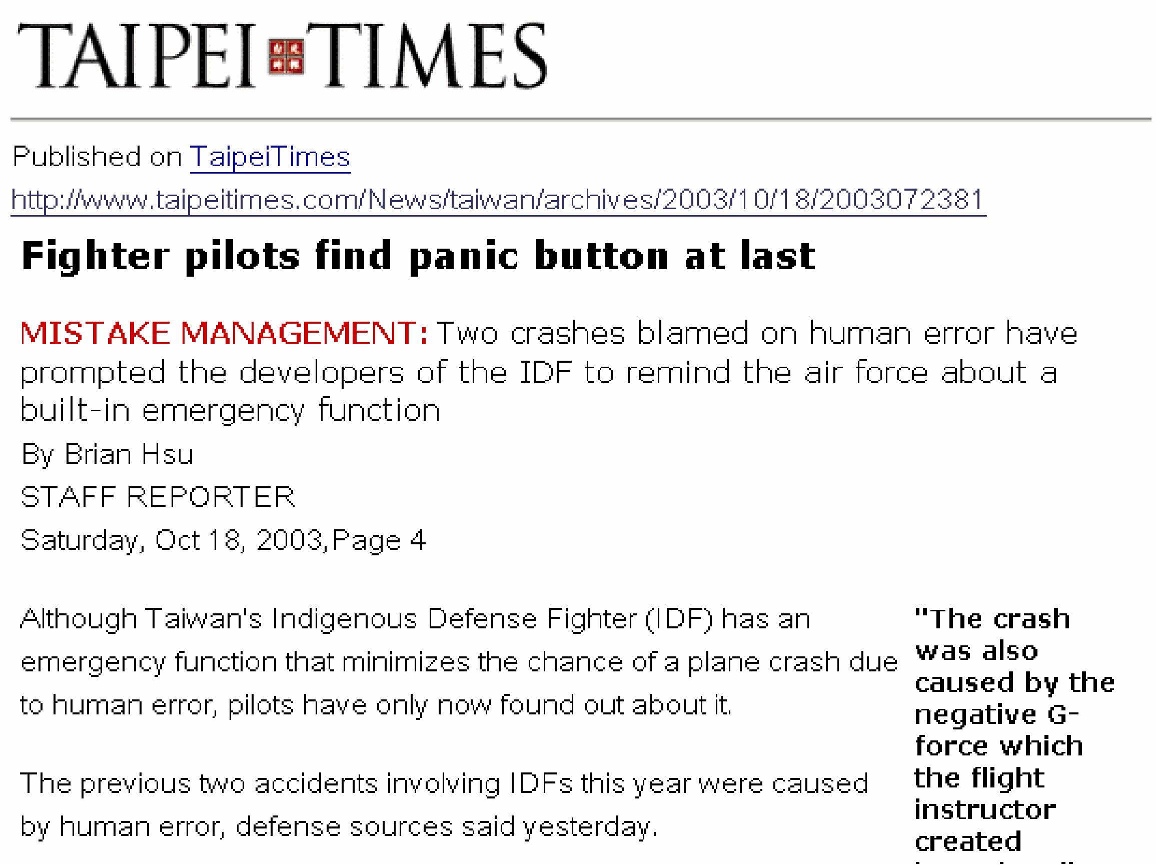
- The Gulf of Evaluation reflects the amount of effort needed to interpret the state of the system how well this can be compared to the intentions
  - Is the information about state of the system easily accessible?
  - Is it represented to ease matching with intensions?
- Example in GUI
  - The user wants a document written on the system in paper (the goal)
  - Is process observable? Are intermediate steps visible?
- Good design minimizes the Gulf of Evaluation

Implications on Design

- Principles of good design (Norman)
  - Stage and action alternatives should be always visible
  - Good conceptual model with a consistent system image
  - Interface should include good mappings that show the relationship between stages
  - Continuous feedback to the user

- Critical points/failures
  - Inadequate goal formed by the user
  - User does not find the correct interface / interaction object
  - User many not be able to specify / execute the desired action
  - Inappropriate / mismatching feedback

About (Human) Errors…



TAIPEI TIMES

Published on TaipeiTimes
http://www.taipeitimes.com/News/taiwan/archives/2003/10/18/2003072381

Fighter pilots find panic button at last

MISTAKE MANAGEMENT: Two crashes blamed on human error have prompted the developers of the IDF to remind the air force about a built-in emergency function

By Brian Hsu
STAFF REPORTER
Saturday, Oct 18, 2003,Page 4

Although Taiwan's Indigenous Defense Fighter (IDF) has an emergency function that minimizes the chance of a plane crash due to human error, pilots have only now found out about it.

The previous two accidents involving IDFs this year were caused by human error, defense sources said yesterday.

"The crash was also caused by the negative G-force which the flight instructor created

…In an attempt to prevent similar accidents in future, the air force has asked the AIDC to help teach pilots how to use the fighter's emergency function.

About (Human) Errors…

- "If an error is possible someone will make it" (Norman)
- Human Error may also be a starting point to look for design problems.
- Design implications
  - Assume all possible errors will be made
  - Minimize the chance to make errors (constraints)
  - Minimize the effect that errors have (that is difficult!)
  - Include mechanism to detect errors
  - Attempt to make actions reversible

Understanding Errors (Norman)

- Errors are routinely made
  - Communication and language is used between people to clarify – more often than one imagines
  - Common understanding of goals and intentions between people helps to overcome errors

- Two fundamental categories
  - Mistakes
    - overgeneralization
    - wrong conclusions
    - wrong goal
  - Slips
    - Result of "automatic" behaviour
    - Appropriate goal but performance/action is wrong

Understanding the types of Slips Users Make (Norman)

- Capture errors - Two actions with common start point, the more familiar one captures the unusual (driving to work on Saturday instead to the supermarket)
- Description errors - Performing an action that is close to the action that one wanted to perform (putting the cutlery in the bin instead of the sink)
- Data driven errors - Using data that is visible in a particular moment instead of the data that is well known (calling the room number you see instead of the phone number you know by heart)
- Associate action errors - You think of something and that influences your action. (e.g. saying come in after picking up the phone)
- Loss-of-Activation error ~ forgetting - In a given environment you decided to do something but when leaving then you forgot what you wanted to do. Going back to the start place you remember.
- Mode error - You forget that you are in a mode that does not allow a certain action or where a action has a different effect

Confirmation is unlikely to prevent Errors (Norman)

- Example
  - User: “remove the file ‘most-important-work.txt’”
  - computer: “are you sure that you want to remove the file ‘most-important-work.txt’?”
  - User: “yes”
  - Computer: “are you certain?”
  - User: “yes of course”
  - Computer: “the file ‘most-important-work.txt’ has been removed”
  - User: Oops, damn!

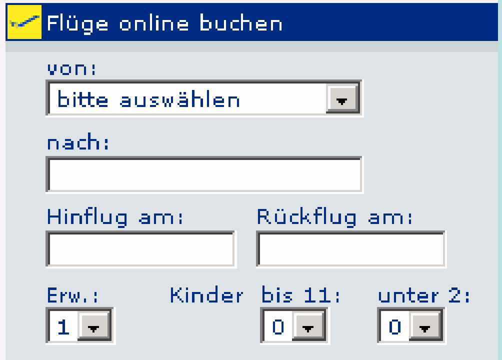
- A solution is to make the action reversible

Constraints

- Physical constraints
  - basic physical limitations
- Semantic constraints
  - Assumption that create something meaningful
- Cultural constraints
  - Borders provided by cultural conventions
- Logical constraints
  - Restrictions due to reasoning
- Applying constraints is a design decision!

GUI Example

Date unconstrained



Flüge online buchen

von: bitte auswählen

nach:

Hinflug am:

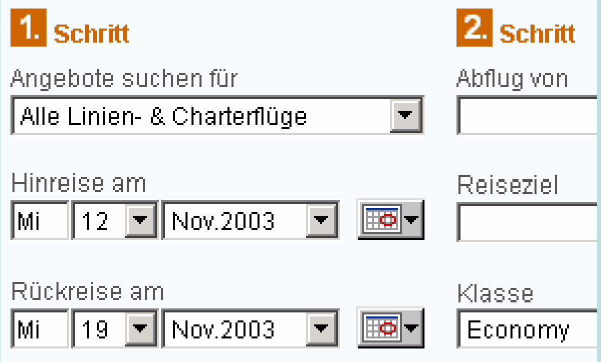
Rückflug am:

Erw.: 1

Kinder bis 11: 0

unter 2: 0

Date constrained



1. Schritt

Angebote suchen für: Alle Linien- & Charterflüge

Hinreise am: Mi 12 Nov.2003

Rückreise am: Mi 19 Nov.2003

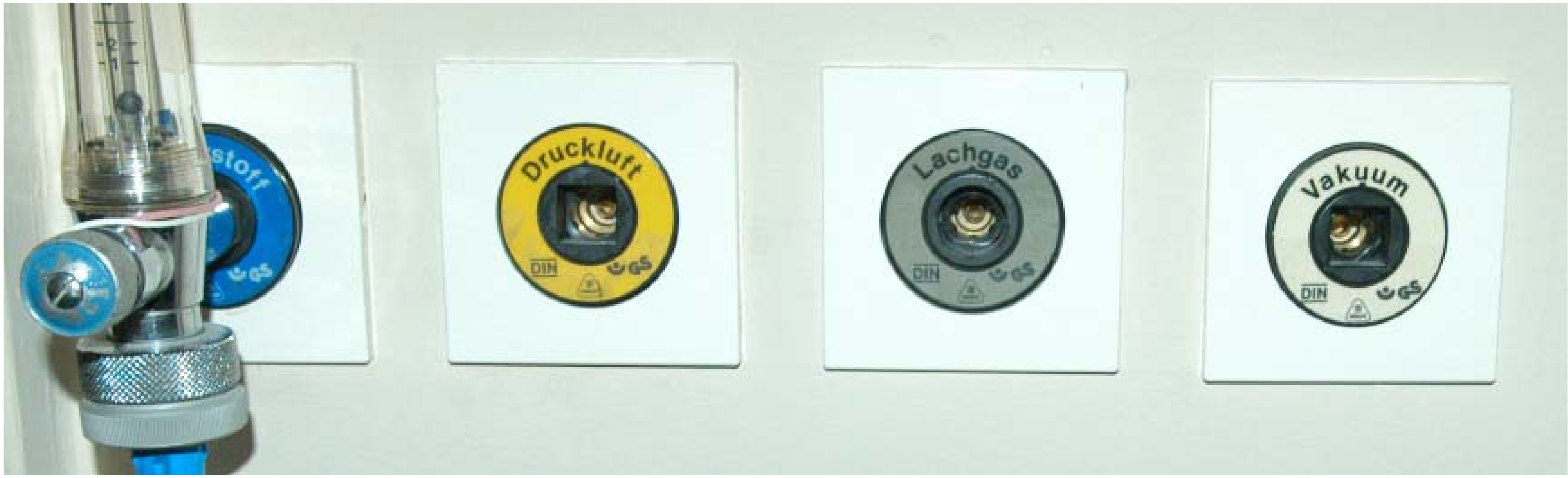
2. Schritt

Abflug von:

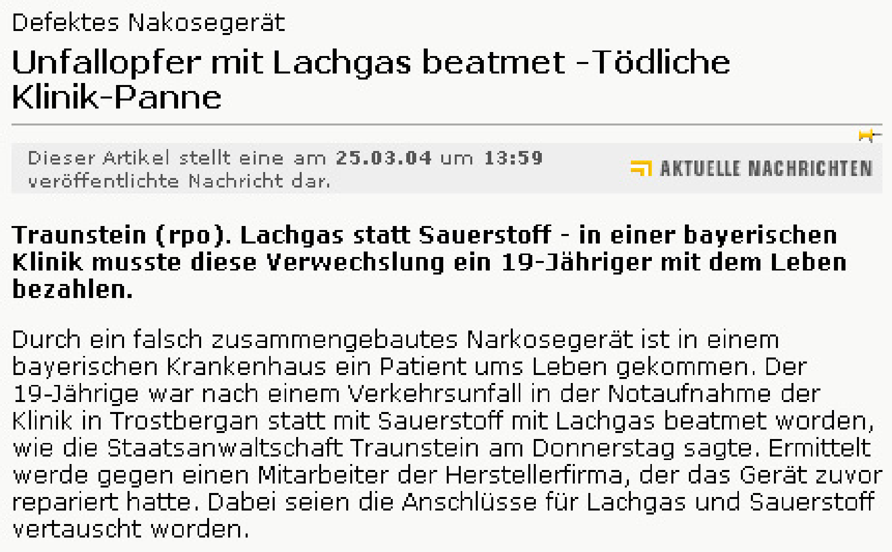
Reiseziel:

Klasse: Economy

Constraints & Redundancy



- Redundancy is safe!
- Constraints can only work at their own level
- But: things can go wrong elsewhere



Defektes Nakosegerät

Unfallopfer mit Lachgas beatmet -Tödliche Klinik-Panne

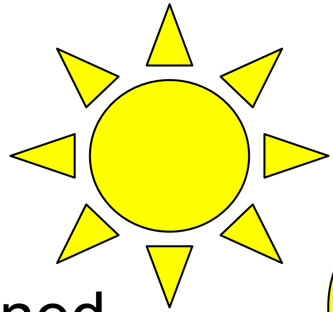
Dieser Artikel stellt eine am **25.03.04** um **13:59** veröffentlichte Nachricht dar.

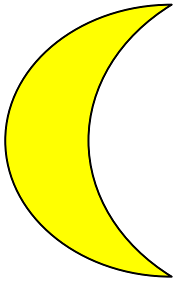
AKTUELLE NACHRICHTEN

Traunstein (rpo). Lachgas statt Sauerstoff - in einer bayerischen Klinik musste diese Verwechslung ein 19-Jähriger mit dem Leben bezahlen.

Durch ein falsch zusammengebautes Narkosegerät ist in einem bayerischen Krankenhaus ein Patient ums Leben gekommen. Der 19-Jährige war nach einem Verkehrsunfall in der Notaufnahme der Klinik in Trostbergan statt mit Sauerstoff mit Lachgas beatmet worden, wie die Staatsanwaltschaft Traunstein am Donnerstag sagte. Ermittelt werde gegen einen Mitarbeiter der Herstellerfirma, der das Gerät zuvor repariert hatte. Dabei seien die Anschlüsse für Lachgas und Sauerstoff vertauscht worden.

Cultural Constraints





- Universal or culturally specific
- Arbitrary conventions that have been learned
- Users' expectations build on cultural constraints

:-)

:-(

;-)

:D

:o

8-)

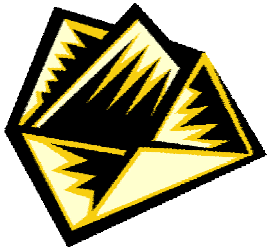
:`-(

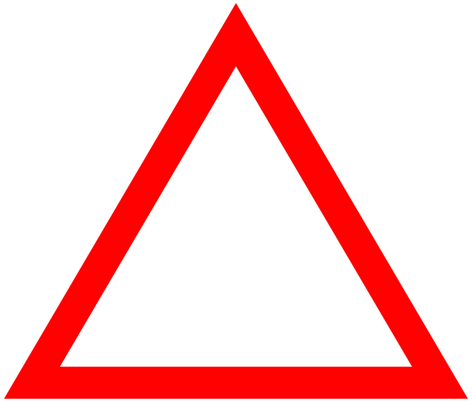
?

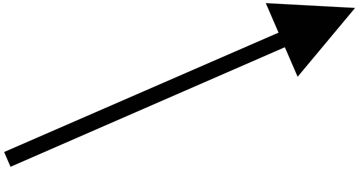
عربي

大阪大学

"Hi there!"

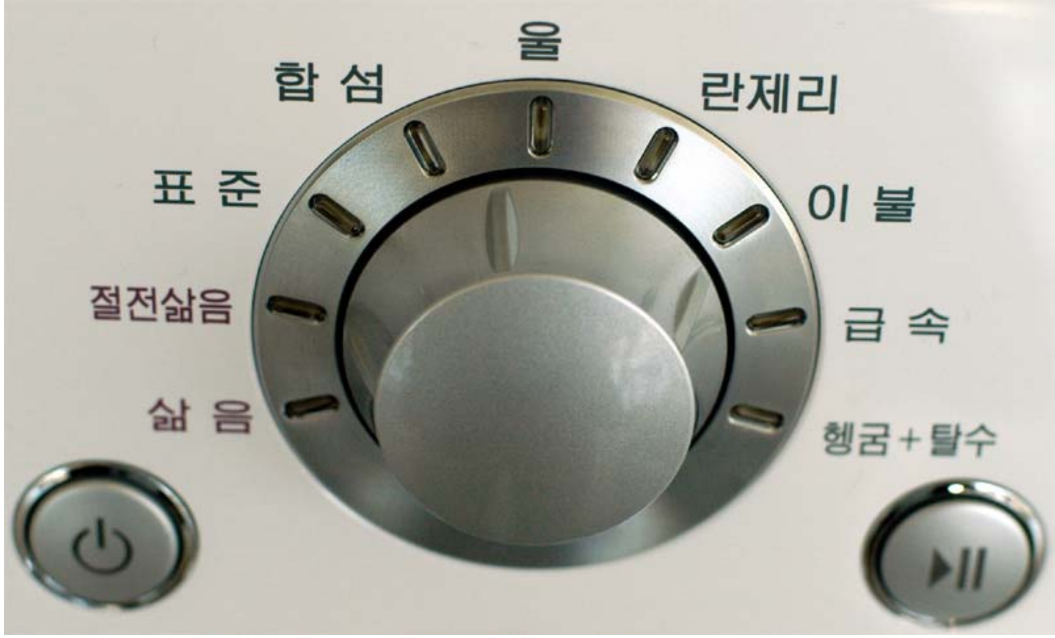


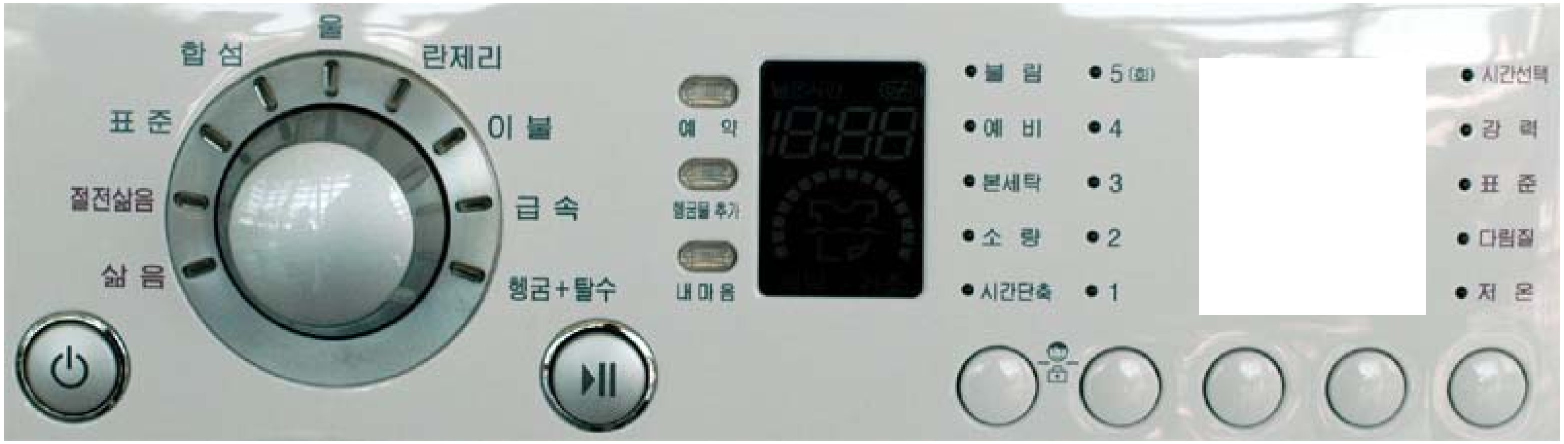




!

Foreign Cultures: Example

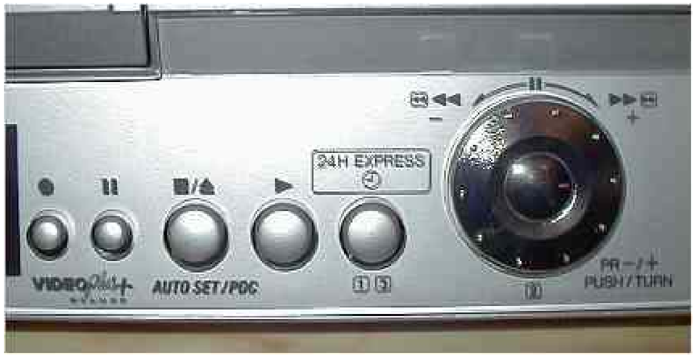




Physical Constraints & Affordances Examples

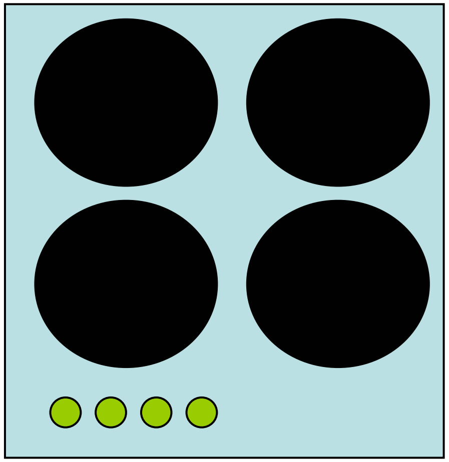
- USB Memory Stick vs. DVD vs. money
  - If there is more than one option (physically) cater these cases

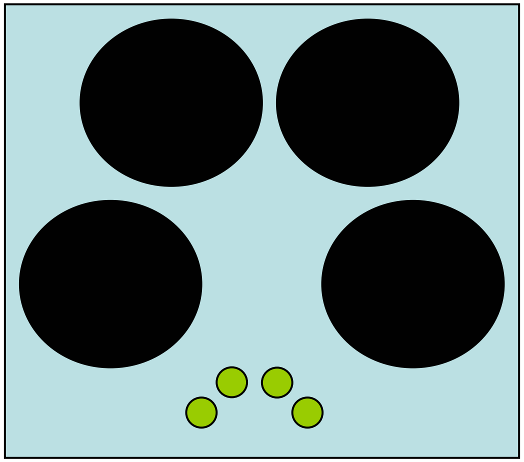
- Dials vs. Buttons vs. Sliders
  - Dials are turned
  - Buttons are pressed
  - Sliders are pushed



Mapping

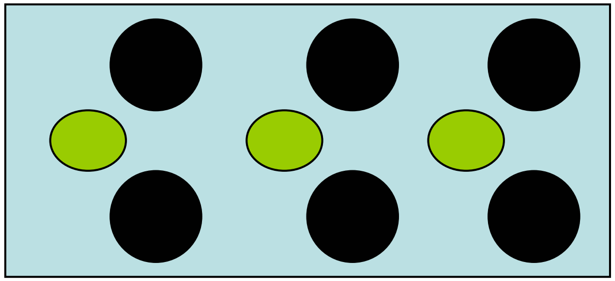
- Relationship between controls and action
- Mappings should be
  - Understandable (e.g. moving the mouse up move the slider up)
  - Consistent
  - Recognizable or at least quickly learnable and easy to recall
  - Natural, meaning to be consistent with knowledge the user already has
- Example: cooker
  (for these issues see also Gestalt theory)

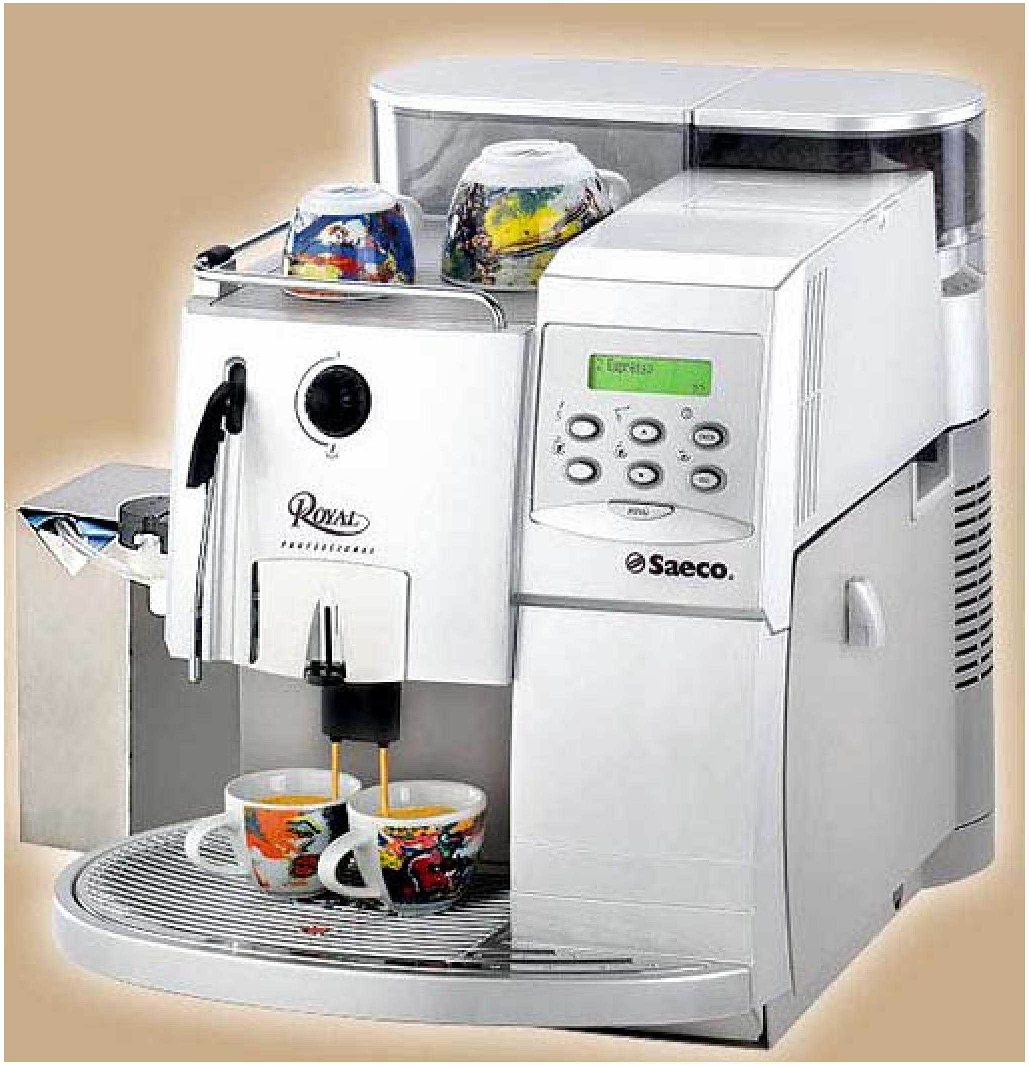




Mapping & Human Error

- Labels are correct
- However full context is needed
- Built-in source for potential frustration
- Missing context

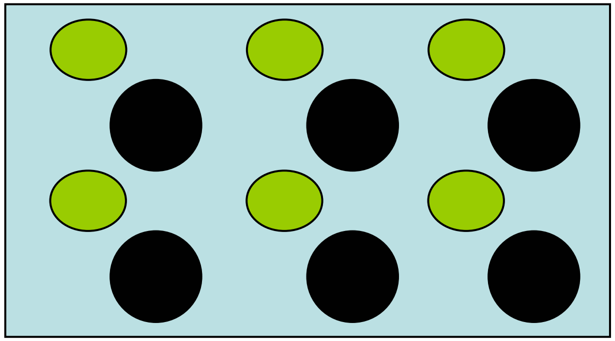




Mapping & Human Error

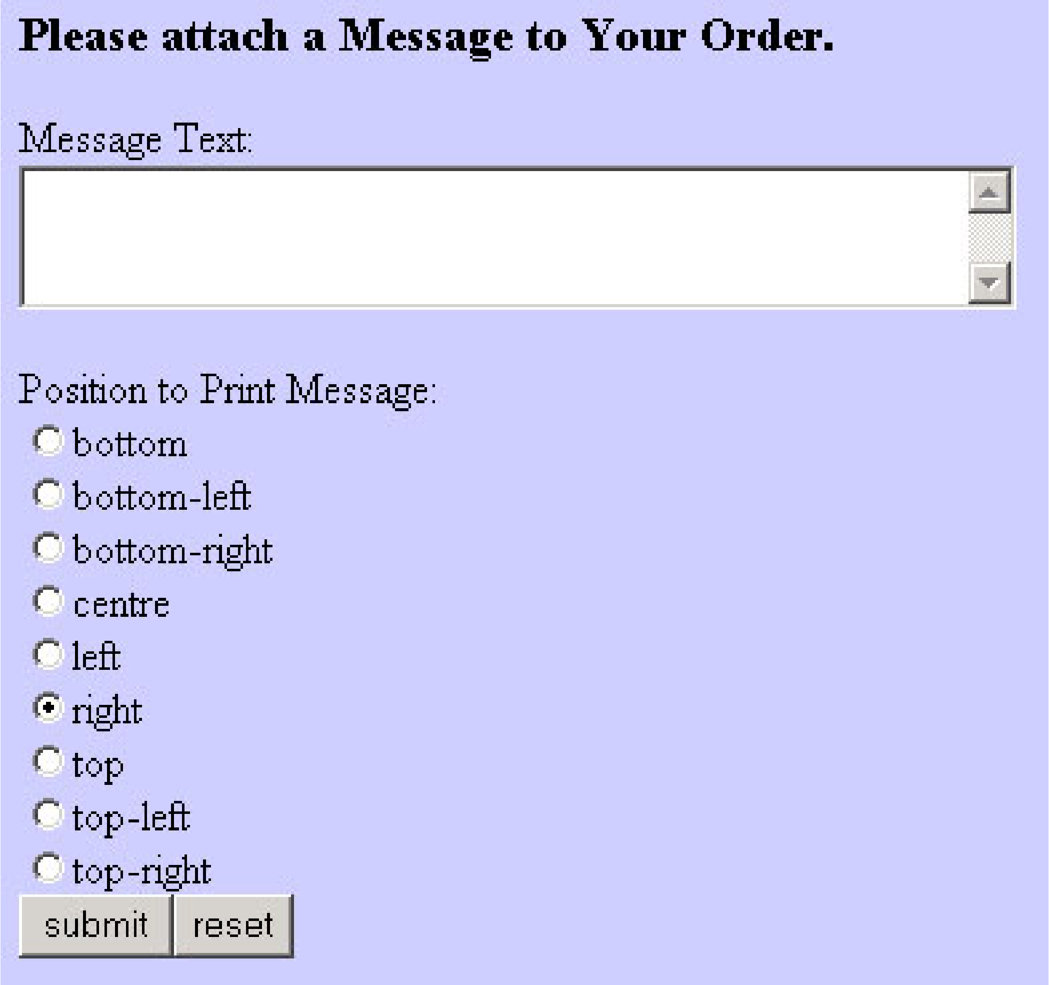
- Labels are correct
- However full context is needed
- Built-in source for potential frustration

- Missing context



Mapping – Examples (1)

- Relationship between controls and action



Please attach a Message to Your Order.

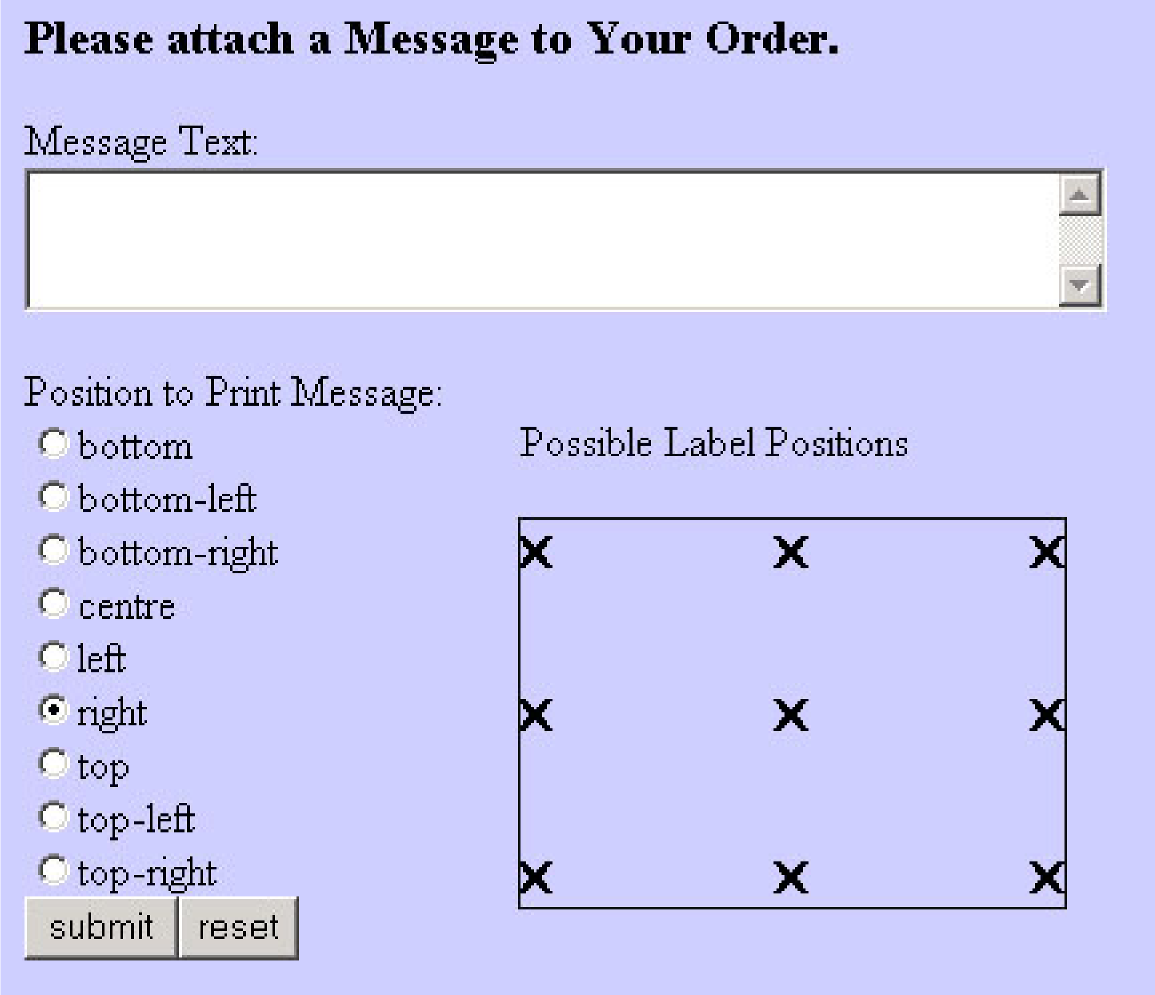
Message Text:

Position to Print Message:

- ○ bottom
- ○ bottom-left
- ○ bottom-right
- ○ centre
- ○ left
- ● right
- ○ top
- ○ top-left
- ○ top-right

Mapping – Examples (2)

- Relationship between controls and action



Please attach a Message to Your Order.

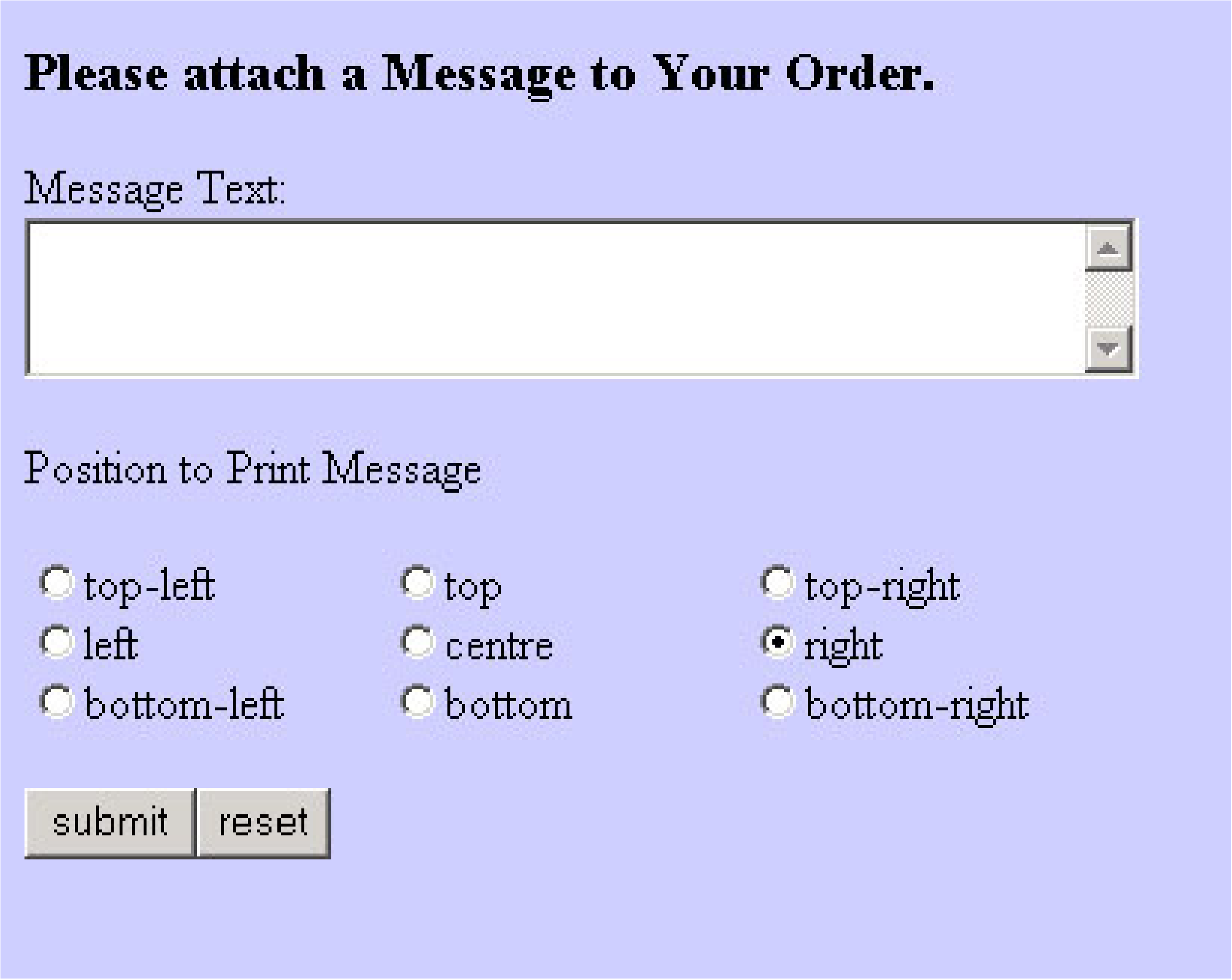
Message Text:

Position to Print Message:

○ bottom
○ bottom-left
○ bottom-right
○ centre
○ left
● right
○ top
○ top-left
○ top-right

Mapping – Examples (3)

- Relationship between controls and action



Please attach a Message to Your Order.

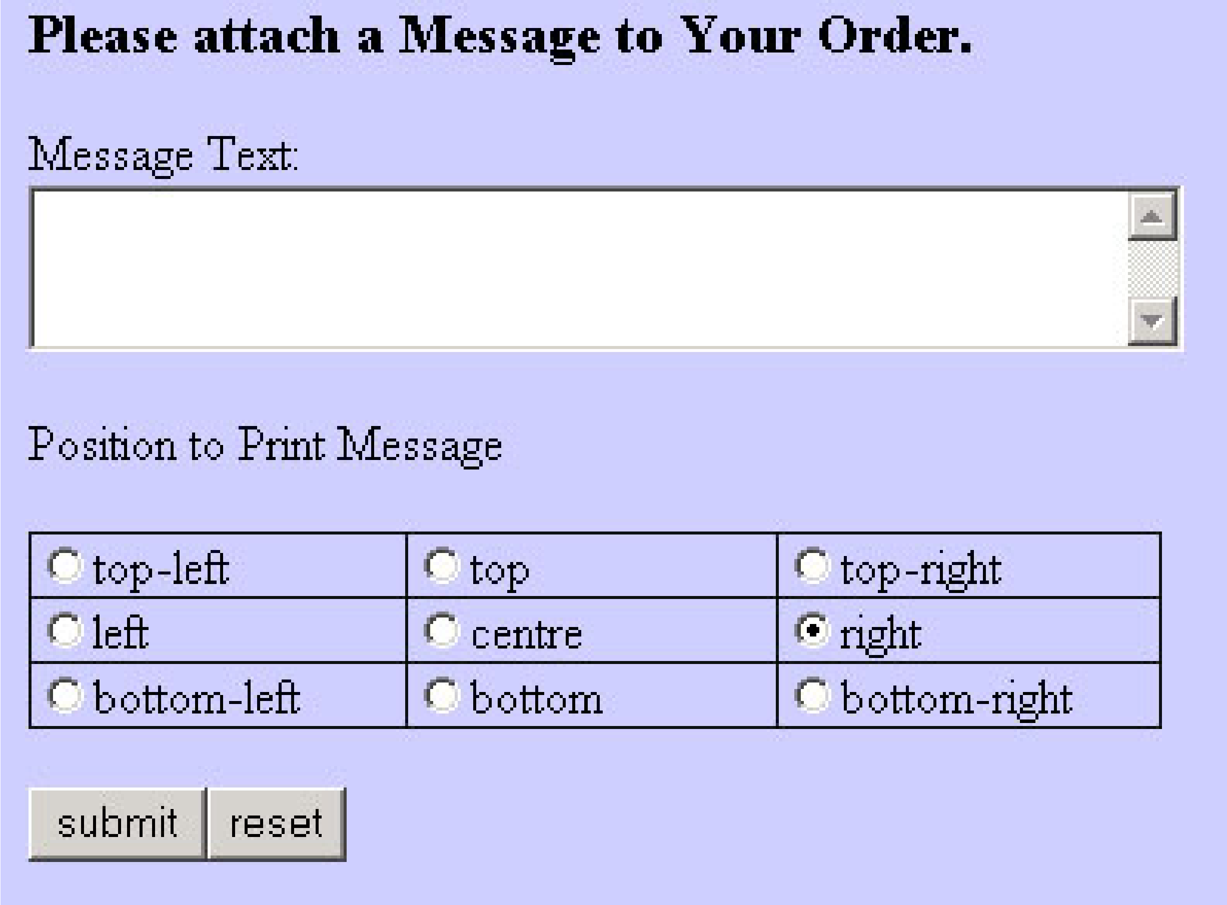
Message Text:

Position to Print Message

○ top-left	○ top	○ top-right
○ left	○ centre	● right
○ bottom-left	○ bottom	○ bottom-right

Mapping – Examples (4)

- Relationship between controls and action



Please attach a Message to Your Order.

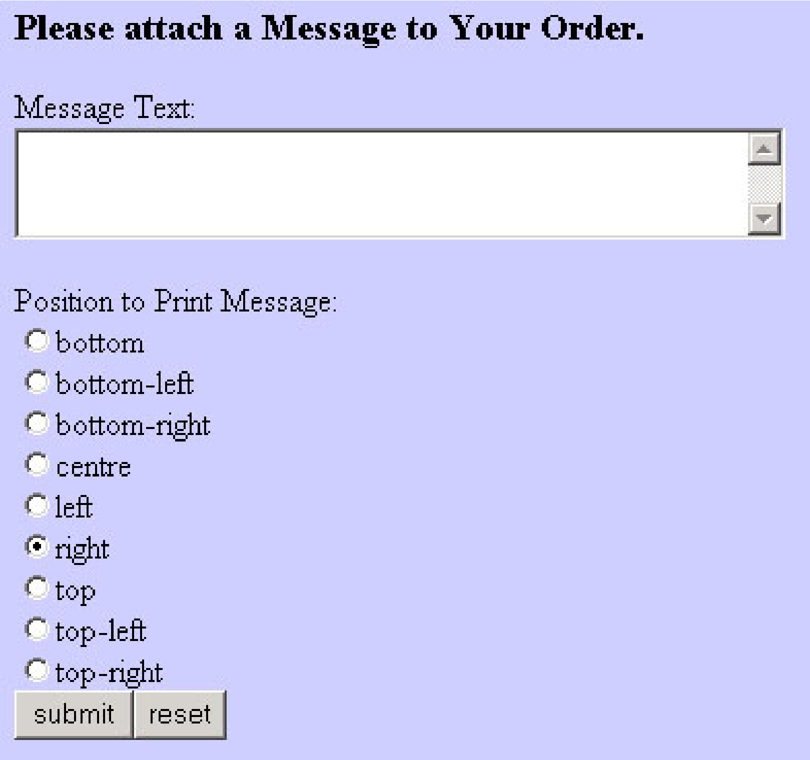
Message Text:

Position to Print Message

○ top-left	○ top	○ top-right
○ left	○ centre	● right
○ bottom-left	○ bottom	○ bottom-right

Mapping – Examples (6)

- Relationship between controls and action

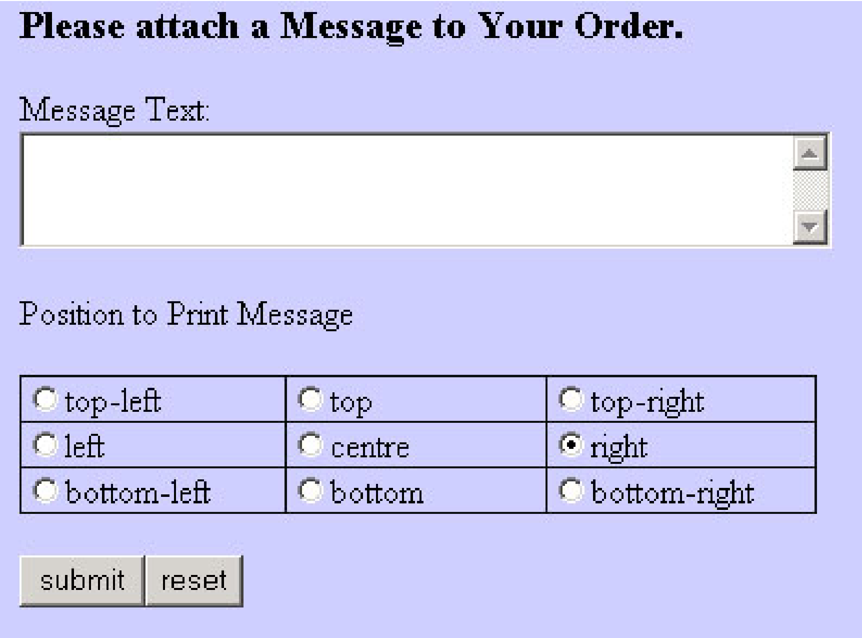


Please attach a Message to Your Order.

Message Text:

Position to Print Message:

○ bottom
○ bottom-left
○ bottom-right
○ centre
○ left
● right
○ top
○ top-left
○ top-right



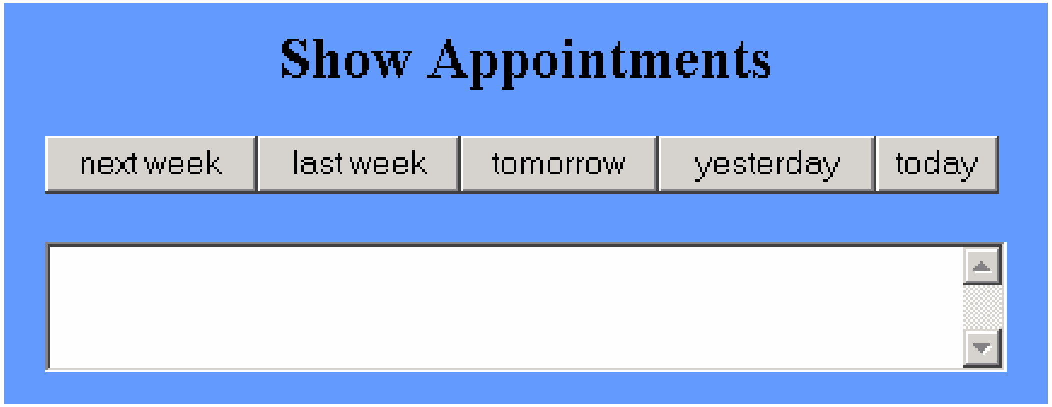
Please attach a Message to Your Order.

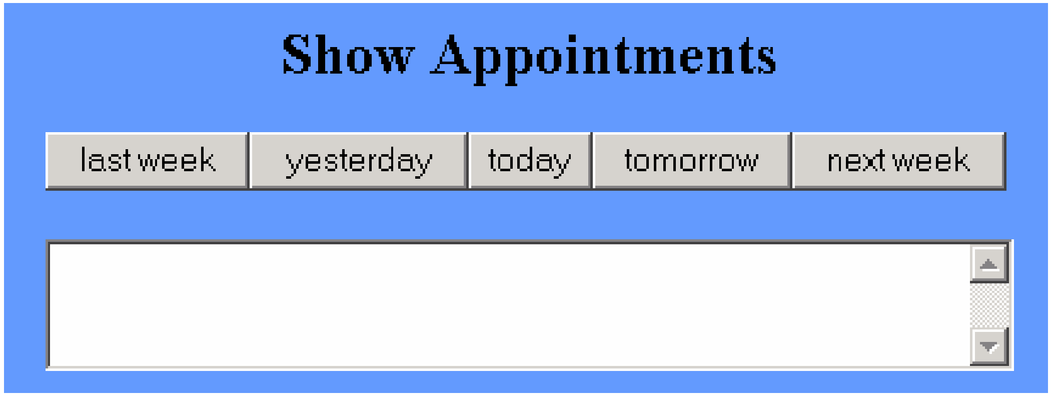
Message Text:

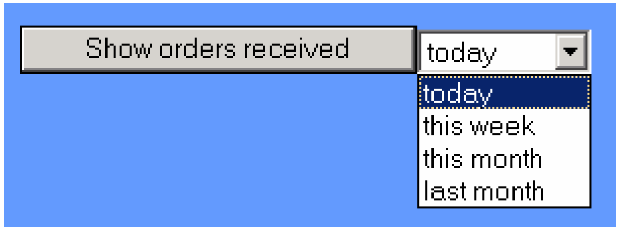
Position to Print Message

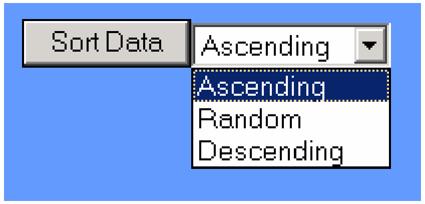
○ top-left	○ top	○ top-right
○ left	○ centre	● right
○ bottom-left	○ bottom	○ bottom-right

Mapping – Examples (5)









- “natural” mappings can be found in many areas
- It is not always obvious what the “natural” mapping is
- Correlation with cultural constraints

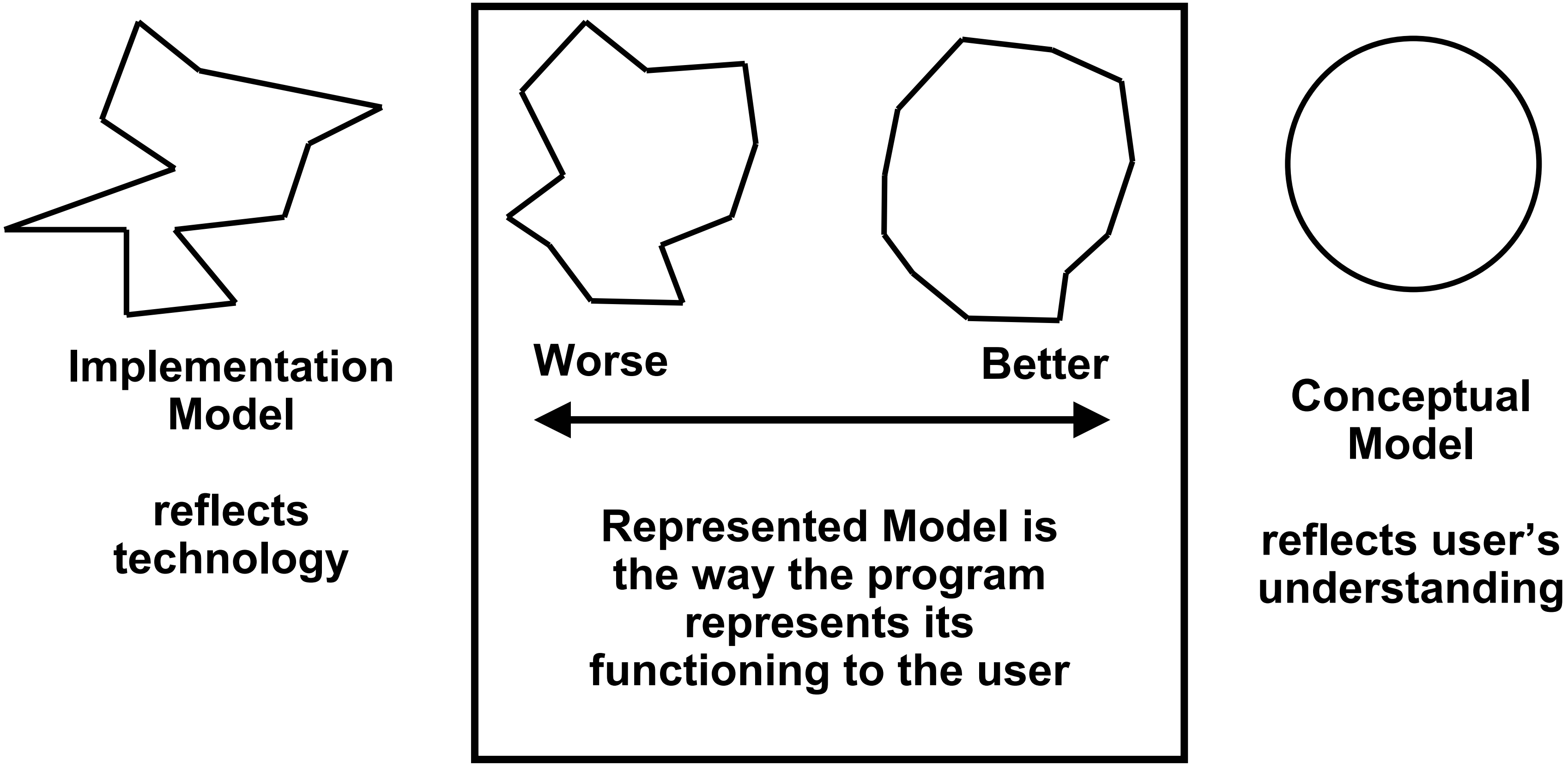
References

- D. A. Norman. The Design of Everyday Things. Basic Books 2002. ISBN: 0465067107
- B. Shneiderman. Designing the User Interface: Strategies for Effective Human-Computer Interaction , Third Edition. 1997. ISBN: 0201694972
- A. Cooper. About Face 2.0: Chapter 1 - Goal-Directed Design http://media.wiley.com/product_data/excerpt/13/07645264/0764526413.pdf

Models and Users (2)

- Conceptual Model in Detail
- Interface Metaphors
- Interaction Paradigms

Implementation, Represented, Conceptual Model



Implementation Model

reflects technology

Worse

Better

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

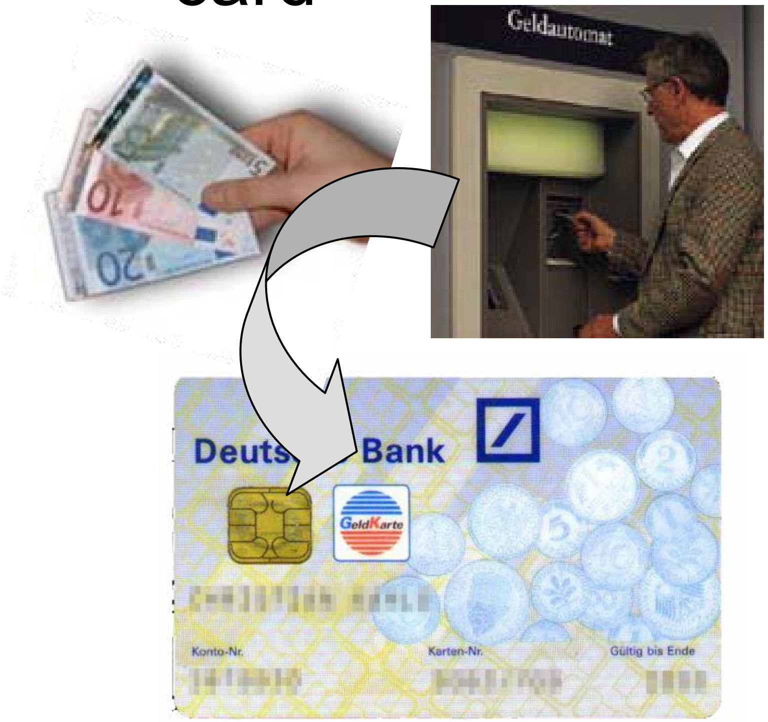
Conceptual Model

reflects user's understanding

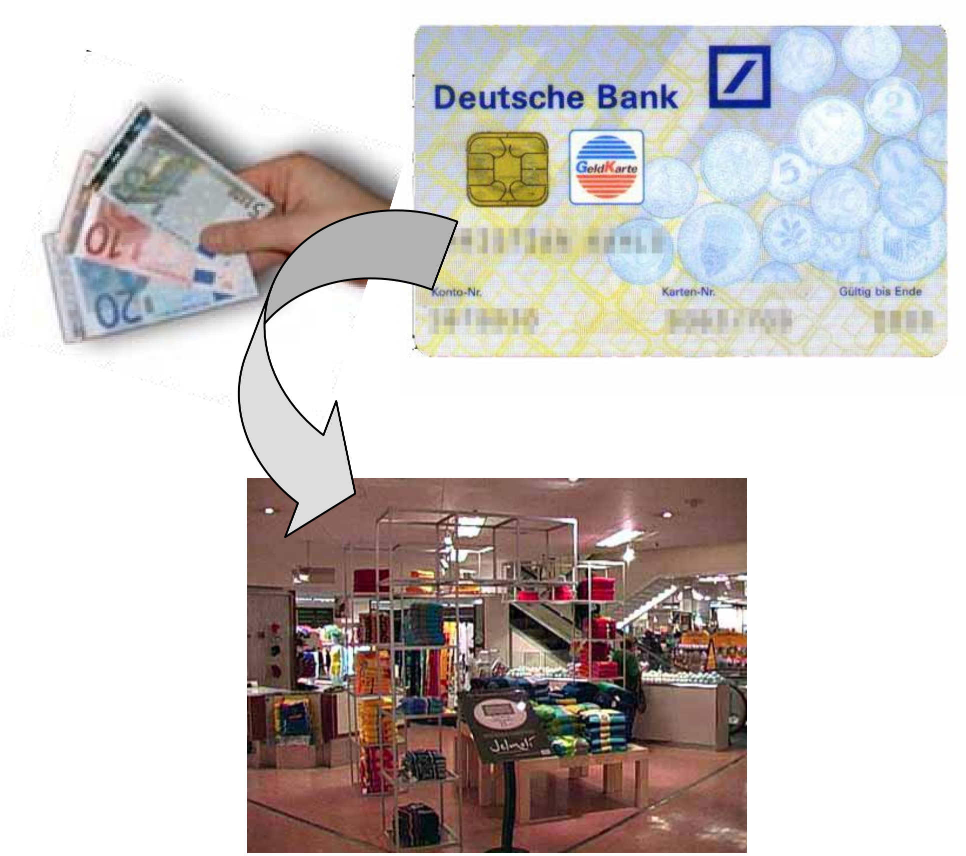
From A. Cooper, About Face 2.0

Example: ‘Geldkarte’ - Difference between the Conceptual Model and Implementation Model

- Store cash on the card



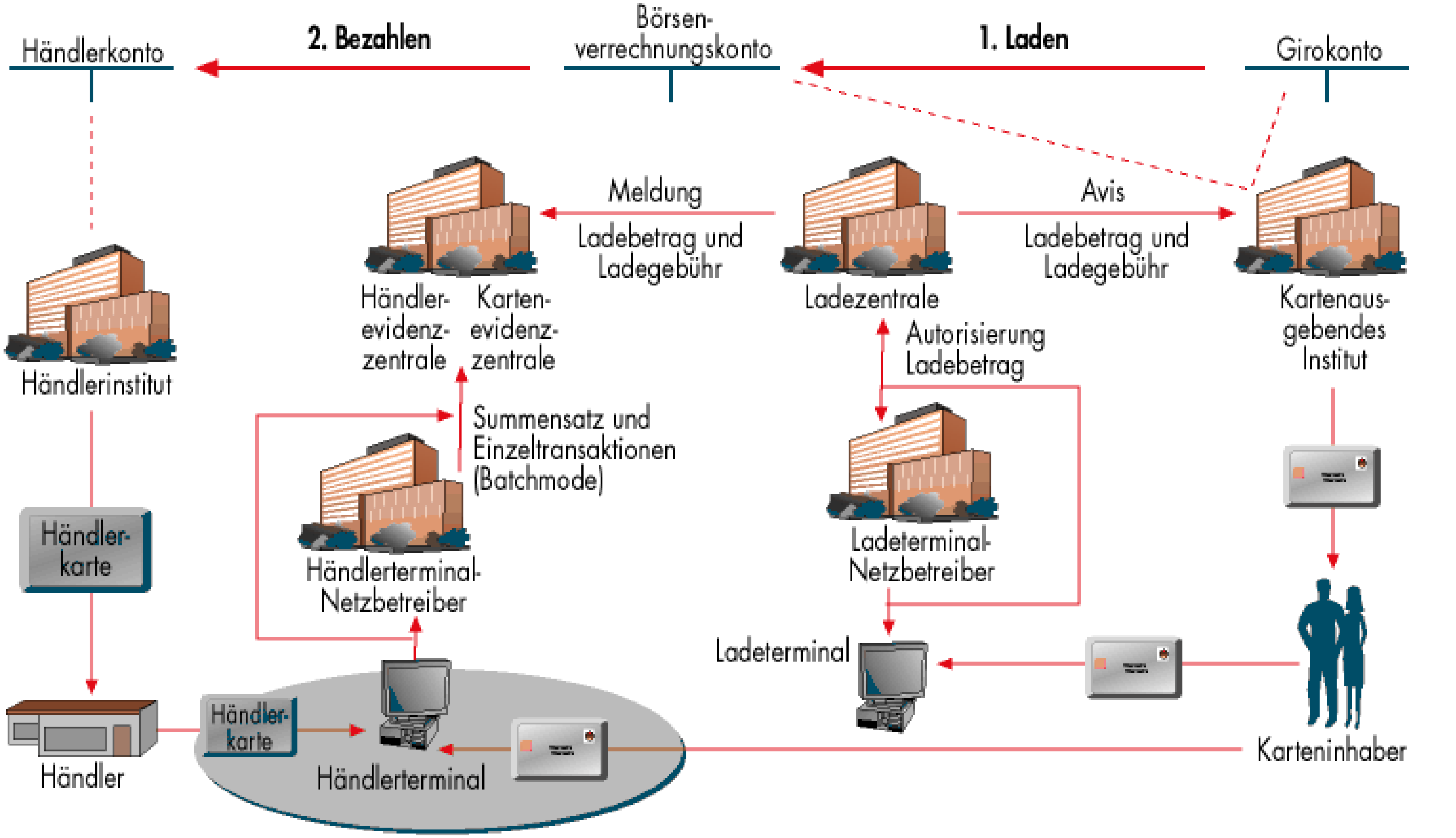
- Pay with the card



Conceptual Model – by the user

Example: ‘Geldkarte’ - Difference between the Implementation Model and Conceptual Model

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"

Implementation Model

- Model how a product is implemented
- Implementation details
  - data structures
  - control flow
  - functional components
- Constraints for the implementation, e.g.
  - remote data access vs. local data access
  - different ways to access records in a database depend on the existents of an index
- Terminology
  - terms/wording used reflect on technology
  - example – see error messages on various systems

Mental & Conceptual Model

- From the user’s point of view
  - the explanation how something works
  - describing the basic properties and possible behaviour
  - the basis on which assumptions and predictions about the system and its behaviour are made
- Technically this is
  - in most cases a simplification of the underlying technology and
  - will most likely not reflect the correct mechanism or the actual implementation
- From the developers/designer point of view
  - how will the system appear to the user
  - how will the user understand the process
  - a conceptual description of the system at high level
- For the user the conceptual model is a psychological shorthand to understand how they can interact with a system

Conceptual Model
A Definition and its Significance

- 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 new 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

How to get a Conceptual Model? 1st Analyse Problem Space

- Understand and analyse the problem space
  - Make problems of existing solution explicit (e.g. list of issue)
  - Why did you characterize them as problem? (because of intuition, reports, user studies, experiments?)
  - How does the envisioned concept solve the problem better? (is it faster, easier to use, easier to deploy, more fun?)
  - How would you see people using it with their current way of doing things?
  - How will it support people in their activities?
  - Will it really help them?
  - Would the envisioned solution introduce new problems? Which?
- Understanding the problem space leads to ideas about
  - What type of device/technology may be appropriate
  - What functionality is required under what conditions
  - What interaction metaphors can be used

How to get a Conceptual Model? 2nd Understand the User’s Goals

- What is the user (or are the users) trying to achieve
  - What is the final goal?
  - Are there intermediate goals?
  - Are there conflicting goals and trade-offs?
  - If multiple users - how are their goals related?
- Understand the tasks involved
  - What tasks and subtasks are carried out?
  - Why is the user doing these tasks?
  - How is this related to a potential solution?
  - Will the solution eliminate task and still reach the goals?
- Relate the user’s goals and tasks to the business model of the envisioned solution
  - Especially for service oriented digital products
  - Are there conflicts of interest between provider and consumer
    (e.g. quick answers and hence short connection time may conflict with a business model based on connection time, see WAP pages)

How to get a Conceptual Model? 3rd Make an Explicit Model

- Based on the analyses of the problem space and goals, identify
  - appropriate interface
  - Interaction methods and metaphors
  - Interaction paradigms
- Make the conceptual model explicit
  - Describe scenarios in detail and the use of the products
  - Storyboarding and videos
  - Sketching out ideas, design sketches
  - Put the solution into the wider context (e.g. an application on the mobile phone in the context of phone usage in general, what happens if a call comes in while you use the application?)
  - Create prototypes
    - low fidelity, e.g. paper prototypes, digital mock-ups (e.g. Flash examples, HTML-Forms with no Backend)

Example of Explicit Conceptual Model and Concept Studies

http://www.cavi.dk/vvpindex-playful.php