Designing Interactive Systems I

Prof. Dr. Jan Borchers

RWTH Aachen

WS 2006/07

Summary of

"*The design of everyday things*",

Donald A. Norman (2002 Edition)

by Alexander Schiffel

I wrote this document during the preparation for my midterm exam in DIS I.
Neither did the chair review this document nor do I take responsibility for any errors in this summary.
Feel free to contact me whenever you got any suggestions or find typos: alexander.schiffel@rwth-aachen.de

Alexander Schiffel, December 2006

chapter one
THE PSYCHOPATHOLOGY OF EVERYDAY THINGS

- ***visibility*** is one of the most important principles of design
- need of natural signals
- affordance:
  - refers to the perceived and actual properties of the thing
    -> fundamental properties that determine just how the thing could be used (stimulus-response compatibility)
- how things work:
  - forming of a conceptual model of the device and mentally simulate its operation
  - clues from visible structure:
    - ***affordances***
    - ***constraints***
    - ***mappings***
- good conceptual model allows to predict the effects of our actions
- good design: give feedback of the actions performed
- whenever the number of possible actions exceed the number of controls, there is apt to be difficulty
- controls with more than one function are harder to remember
- if user forgets the function, the controls serve as reminders
- ***natural mapping*** (i.e. close, natural relationship between the control and its function) leads to immediate understanding
- when things are visible, they tend to be easier than when they are not
- paradox of technology:
  - the same technology that simplifies life by providing more functions in each device also complicates life by making the device harder to learn, harder to use
  - anyway with clever design, they can be minimized

chapter two
THE PSYCHOLOGY OF EVERYDAY ACTIONS

- if an error is possible, someone will make it
- ***errors*** should be:
  - easy to detect
  - have minimal consequences
  - effects should be reversible
- mental models result from tendency to form explanation of things
- models are essential in:
  - helping to understand our experiences
  - predict the outcome of actions
  - handling unexpected occurrences
- models are based on real, imaginary, naive or sophisticated knowledge

- often a construct from fragmentary evidence with poor understanding of what´s happening

- people have tendency to blame themselves for difficulties with technology
- they tend to assign a causal relation whenever two things occur in succession

- ***learned helplessness***: way to explain self-blame
  - failing numerous times => they stop trying

- ***taught helplessness***:
  - few instances of failure generalize to every technical object
  - cause from badly designed objects: faulty mental models, poor feedback etc.

- once having an explanation – correct or incorrect – for otherwise discrepant or puzzling events, people tend to be complacent

- ***the seven stages of action****:*
  - two major aspects: doing something (*execution*) and checking (*evaluation*)
  - ***forming the goal -> forming the intention -> specify an action -> executing the action -> perceiving the state of the world -> interpreting the state of the world -> evaluating the outcome***
  - *goals* do not state precisely what to do
  - goals must be transformed into specific statements of what is to be done (*intentions*)
  - goal: something to be achieved; intentions: specific action to get to the goal
  - after specifying the actions, we must actually do them -> *stage of execution*
  - evaluation side of things: perceiving what happened in the world, trying to make sense of it, comparing what happened with what was wanted
  - the seven stages form an approximate model, not a complete psychological theory
  - almost certainly not discrete entities
    - most actions do not require going through all stages
    - most activities will not be satisfied by single actions
    - there are activities in which goals are forgotten, discarded, reformulated
    - often goals and intentions are opportunistic rather than planned
  - the seven-stages process can be started at any point
  - ***gulf of execution****:*
    - difference between the intentions and the allowable actions
  - ***gulf of evaluation****:*
    - reflects the amount of effort that the person must exert to interpret the physical state of the system and to determine how well the expectations and intentions have been met
    - small gulf when system provides information about its state in a form that is easy to get and interpret
  - on the whole, a valuable design aid
  - questions for each stage boil down to the principles of good design:
    - ***visibility***: by looking, the user can tell the state and the alternatives for action
    - ***good conceptual model***: consistency in the presentation of operations and results and a coherent, consistent system image

  - ***good mappings***: possible to determine the relationship between action and results, between the controls and their effects, between system state and what is visible
  - ***feedback***: full and continuous feedback about the results of actions

chapter three
KNOWLEDGE IN THE HEAD AND IN THE WORLD

- because not all the knowledge required for precise behavior has to be in the head
  => apparent discrepancy between the precision of behavior and the imprecision of knowledge
- precise behavior can emerge from imprecise knowledge:
  1. ***information is in the world***. behavior is determined by combining the information in the head with that in the world
  2. ***great precision is not required***. perfect behavior will result if the knowledge describes the information or behavior sufficiently to distinguish the correct choice from all others
  3. ***natural constraints are present***.
  4. ***cultural constraints are present***. have to be learned, but once learned they apply to a wide variety of circumstances

  1. *information is in the world*
     - whenever information needed to do a task is readily available in the world, the need to learn it diminishes
     - tradeoff between speed and quality of performance and mental effort
     - people function through their use of two kinds of knowledge:
       - **knowledge *of***
         - *"declarative knowledge"*
         - includes knowledge of facts and rules
         - easy to write down and teach
       - **knowledge *how***
         - *"procedural knowledge"*
         - how to perform certain actions
         - difficult or impossible to write down and difficult to teach
         - largely subconscious
     - people structure the environment to provide a considerable amount of the information required for something to be remembered

  2. *great precision is not required*
     - we store only partial descriptions of the things to be remembered, descriptions that are sufficiently precise to work at the same time something is learned
     - may not work later on, when new experiences have also been encountered and entered into memory

  3. +4. *natural and cultural constraints are present*
     - constraints reduce the amount that must be learned to reasonable quantity

memory in the head

- structure of memory:
  - ***short-term memory*** (STM)
    - memory of the just present
    - information is retained in it automatically and retrieved without effort
    - amount severely limited (5-7 items)
    - acts as a working or temporary memory
  - ***long-term memory*** (LTM)
    - memory for the past
    - takes time to put stuff away in LTM and effort to get it out again
    - does not contain exact recordings, but as interpreted through our understanding of them
    - how well experiences can be recovered highly depends upon how the material was interpreted in the first place
    - does not impose any practical limit
    - storage and retrieval are easier when the material makes sense, when it fits into what is already known
  - different categories how people use their memories:
    1. ***memory for arbitrary things.***
    2. ***memory for meaningful relationships***
    3. ***memory through explanation***

    1. *arbitrary things*
       - simple remembering of what is to be done, without reliance on an understanding
       - *"rote learning"*
       - problems caused:
         - learning is difficult
         - when a problem arises, the memorized sequence of actions gives no hint of what has gone wrong, no suggestion of what might be done to fix the problem

    2. *meaningful relationships*
       - when things make sense, they correspond to knowledge that we already have
       - we can use rules and constraints to help understand what things go together

    3. *explanation*
       - material does not have to be remembered, but rather can be derived from some explanatory mechanism
       - use of mental models to remember (here: derive) behavior is not ideal for tasks that must be done rapidly and smoothly
       - derivation takes time, requires mental resources
       - great benefit:
         - let you figure out what would happen in novel situations
         - whenever a problem occurs, it lets you figure out what is happening

memory is also knowledge in the world

- on one of the most interesting aspects of external memory: reminding (interplay between knowledge in the head and in the world)

- two different aspects to the reminder: the signal and the message
- ideal reminder has both components: the signal that something is to be remembered and the message of what it is

- proper natural mapping requires no diagrams, no labels and no instructions
- simple design principle:
  - if a design depends upon labels, it may be faulty
  - appropriate use of natural mapping can minimize the need of labels
  - wherever labels are necessary, consider another design

- knowledge in the world and in the head are both essential
- to some extent we can choose to lean more heavily on one or the other
- requires a tradeoff:

PROPERTY	KNOWLEDGE IN THE WORLD	KNOWLEDGE IN THE HEAD
Retrievability	Retrievable whenever visible or audible	Not readily retrievable. Requires memory search or reminding.
Learning	Learning not required. Interpretation substitutes for learning. How easy it is to interpret information in the world depends upon how well it exploits natural mappings and constraints.	Requires learning, which can be considerable. Learning is made easier if there is meaning of structure to the material (or if there is a good mental model).
Efficiency of use	Tends to be slowed up by the need to find and interpret the external information.	Can be very efficient.
Ease of use at first encounter	High.	Low.
Aesthetics	Can be unaesthetic and inelegant, especially if there is a need to maintain a lot of information. This can lead to clutter. In the end, aesthetic appeal depends upon the skill of the designer	Nothing needs to be visible, which gives more freedom to the designer, which in turn can lead to better aesthetics.

- knowledge in the world is self-reminding
- knowledge in the mind is ephemeral

chapter four
KNOWING WHAT TO DO

- affordances suggest the range of possibilities
- constraint limit the number of alternatives

- ***physical constraints***:
  - constrain possible actions
  - rely upon properties of the physical world for their operation
  - there should be only a limited number of possible actions or desired actions can be made obvious

- ***semantic constraints***:
  - rely upon the meaning of the situation to control the set of possible actions
  - rely upon our knowledge of the situation and of the world

- ***cultural constraints***:
  - rely upon accepted cultural conventions
  - guidelines for cultural behavior are represented in the mind by means of schemas, knowledge structures that contain the general rules and information necessary for interpreting situations and for guiding behavior

- ***logical constraints***:
  - there is a logical relationship between the spatial or functional layout of components and the things they affect or are affected by

- other relevant principles:
  - ***visibility***: make relevant parts visible
  - ***feedback***: give each action an immediate and obvious effect

  - using ***sound*** for visibility:
    - use is valuable and serves important function, but it is very limited in power
    - should be generated so as to give information about the source
    - should convey something about the actions that are taking place, actions that matter to the user but that would otherwise not be visible
    - sound tells us about things we can´t see, and it does so even while our eyes are occupied elsewhere
    - can annoy and distract as easily as it can help
    - the absence of sound can lead to the same kinds of difficulties encountered from a lack of feedback if feedback from an action is expected to come from sound

chapter five
TO ERR IS HUMAN

- errors come in several forms
- two fundamental categories: ***slips*** and ***mistakes***
  - *slips* result from automatic behavior, when subconscious actions get waylaid en route
  - *mistakes* result from conscious deliberations
  - connection to the seven stages of action:
    - *slips*:
      - form an appropriate goal but mess up in the performance, and you´ve made a slip
      - slips are almost always small things: a misplaced action, the wrong thing moved etc.
      - relatively easy to discover by simple observation and monitoring
    - *mistakes*:
      - form the wrong goal, and you´ve mad a mistake
      - mistakes can be major events
      - difficult or even impossible to detect
  - slips show up most frequently in skilled behavior
  - slips happen especially when the operator is experienced and well practiced and therefore not paying full attention, and if there are more important things to do

  - **types of slips**
    - ***capture errors***:
      - a frequently done activity suddenly takes charge instead of the one intended
      - appears whenever two different action sequences have their initial stages in common, with one sequence being unfamiliar and the other being well practiced
    - ***description errors***:
      - intended action has much in common with others that are possible
      - internal description of the intention was not sufficiently precise
      - usually result in performing the correct action on the wrong object
      - occur most frequently when the wrong and the right objects are physically near each other
    - ***data driven errors***:
      - automatic actions are data driven (triggered by the arrival of sensory data)
      - sometimes data-driven activities can intrude into an ongoing action sequence, causing behavior that was not intended
    - ***associative activation errors***:
      - also internal thoughts and associations can sometimes trigger actions
      - associations among thoughts and ideas
    - ***loss-of-activation errors***:
      - simply forgetting to do something
      - occur because the presumed mechanism ("activation") has decayed
    - ***mode errors***:
      - occur when devices have different modes of operation, and the action appropriate for one mode has different meanings in other modes
      - especially likely where the equipment does not make the mode visible

  - detection of slips can only take place if there is feedback
  - often possible to detect that the result of an action is not as planned, but then not to know at which level of specification the error has taken place
  - error correction mechanism tends to start at the lowest possible level and slowly works its way higher

- much of our knowledge is hidden beneath the surface of our minds, inaccessible to conscious inspection
- we discover our own knowledge primarily through our actions

- everyday structures are either *shallow* or *narrow*
  - in ***shallow structures***, there is no problem of planning or depth of analysis
  - a ***narrow structure*** arises when there are only a small number of alternatives and each possibility might lead to one or two further choices
    - any task that involves a sequence of activities where the action to be done at any point is determined by its place in the sequence is an example of a *narrow structure*
- everyday activities must usually be done relatively quickly, often simultaneously with other activities
- neither time nor mental resources may be available
- activities structure themselves to minimize planning and mental computation

- ***subconscious thought*** matches patterns
  - proceeds rapidly and automatically
  - good at:
    - detecting general trends
    - recognizing the relationship between what we now experience and what has happened in the past
    - good at generalizing
  - might find matches that are
    - are inappropriate
    - wrong
    - and it may not distinguish the common from the rare
- ***conscious thought***:
  - low and labored
  - slowly pondering decisions, thinking through alternatives, comparing different choices
  - tends to be slow and serial
  - seems to involve short-term memory and is thereby limited in the amount that can be readily available

- *designing for error*
  1. understand the causes of error and design to minimize those causes
  2. make it possible to reverse actions – to "undo" them – or make it harder to do what cannot be reversed
  3. make it easier to discover the errors that do occur, and make them easier to correct
  4. change the attitude towards errors. think of an object´s user as attempting to do a task, getting there by imperfect approximations. don´t think of the user as making errors; think of the

action as approximations of what is desired.
- design so that errors are easy to discover and corrections are possible

- <u>how to deal with errors</u>:
  - ***forcing functions***
    - form of physical constraint: situations in which the actions are constrained so that failure at one stage prevents the next step from happening
    - make sure it works right, is reliable, and distinguishes legitimate violations from illegitimate ones
    - *field of safety engineering*:
      - *interlock* forces operations to take place in a proper sequence
      - *lockin* keeps an operation active, preventing someone from prematurely stopping it
      - *lockout* device is one that prevents someone from entering a place that is dangerous, or prevents events from occurring

- *conclusions*:
  - Put the required knowledge in the world. don´t require all the knowledge to be in the head. Yet do allow for more efficient operation when the user has learned the operations, has gotten the knowledge in the head.
  - Use the power of natural and artificial constrains: physical, logical, semantic and cultural. Use forcing functions and natural mappings.
  - Narrow the gulf of execution and evaluation. Make things visible, both for execution and evaluation. On the execution side, make the options readily available. On the evaluation side, make the results of each action apparent. Make it possible to determine the system state readily, easily, and accurately, and in a form consistent with the person´s goals, intentions, and expectations.

chapter six
THE DESIGN CHALLENGE

- designers often think of themselves as typical users, but in fact they are not
- when operating their devices they almost do it entirely from knowledge in the head
- all of us develop an everyday psychology ("folk psychology") that can be erroneous and misleading as we have access to our conscious thoughts and beliefs but not to our subconscious ones

- clients of the designers are often not the end users
- purchaser is probably interested primarily in price, perhaps in size or appearance, almost certainly not in usability
- manufacturer is primarily concerned about these decision makers

- there is no such thing as the average person
- fixed solutions will invariably fail with some people
- flexible solutions at least offer a chance for those with special needs

- ability of conscious attention is limited: focus on one thing and you reduce your attention to other ("selective attention")
- when there is a error, people are apt to focus on it to the exclusion of others
- designer must design for the problem case, making other factors more salient, or easier to get to, or perhaps less necessary

- if you can´t put the knowledge on the device, then develop a cultural constraint: standardize what has to be kept in the head

- creeping featurism is the tendency to add to the number of features that a device can do, often extending the number beyond all reason
- each new set of features adds immeasurably to the size and complexity of the system
- more and more things have to be made invisible, in violation of all the principles of design
- two paths to treating featurism:
  - ***avoidance***, or at the least, restraint
  - ***organization***: organize, modularize, use the strategy of divide and conquer -> *modularization*
    - create separate functional modules, each with a limited set of controls, each specialized for some different aspects of the task

- *computers as chameleon:* its shape, form, and appearance are not fixed
  - *explorable systems: inviting experimentation*
    - important method of making things easier to learn and to use is to make them explorable
    - three requirements for a system to be explorable:
      1. in each state of the system, the user must readily see and be able to do the allowable actions
      2. effect of each action must be both visible and easy to interpret
         => let user develop a good mental model of the system, learn the causal relationship between actions and outcomes
      3. actions should be without cost, they should be easily reversible
         most actions should be cost-free, explorable, discoverable
  - *two modes of computer usage*
    1. "***command mode***" or "***third-person***" interaction
       - well suited for situations in which the job is laborious or repetitive
       - as well as those in which you can trust the system to do the job for you properly
    2. "***direct manipulation mode***" or "***first-person***" interaction
       - needed if the job is critical, novel or ill-specified, or if you not yet know exactly what is to be done
       - often easy to use, fun and entertaining
       - drawback: often difficult to do a really good job with them
         - require a user to do the task directly, and the user may not be very good at it

chapter seven
USER-CENTERED DESIGN

- design should
  - Make it easy to determine what actions are possible at any moment (make use of constraints),
  - Make things visible, including the conceptual model of the system, the alternative actions, and the result of the actions.
  - Make it easy to evaluate the current state of the system.
  - Follow natural mappings between intentions and the required actions; between actions and the resulting effect; and between the information that is visible and the interpretation of the system state

- performing difficult tasks into simple ones

  1. *Use both knowledge in the world and knowledge in the head.*
  2. *Simplify the structure of tasks.*
  3. *Make things visible: bridge the gulfs of Execution and Evaluation.*
  4. *Get the mappings right.*
  5. *Exploit the power of constraints, both natural and artificial.*
  6. *Design for error.*
  7. *When all else fails, standardize.*

  1. *use both knowledge in the world and knowledge in the head*
     - design should not impede action, especially for those well-practiced, experienced users who have internalized the knowledge
     - should be easy to go back and forth, to combine the knowledge in the head with that in the world
     - allow mutual support of both kinds of knowledge

     - designer must develop a conceptual model that is appropriate for the user, that captures the important parts of the operation of the device, and that is understandable by the user
     - three different aspects of mental models:
       - ***i. design model***
         - conceptualization that the designer has in mind
       - ***ii. user´s model***
         - what the user develops to explain the operation of the system
         - ideally, the user´s model and the design model are equivalent
         - user and designer communicate only through the system itself
       - ***iii. system image***
         - designer must ensure that everything is consistent with and exemplifies the operation of the proper conceptual model

     - the user acquires all knowledge of the system from that system image

     - system image includes instruction manuals and documentation

- the manuals should be written first, then the design would follow the manual

2. *simplify the structure of tasks*
   - tasks should be simple in structure, minimizing the amount of planning or problem solving they require
   - unnecessarily complex tasks can be restructured through technology
     - four major technological approaches:
       - Keep the task much the same, but provide mental aids.
       - Use technology to make visible what would otherwise be invisible, thus improving feedback and the ability to keep control.
       - Automate, but keep the task much the same.
       - Change the nature of the task.
         - change which type of skill is required

   - first two approaches to mentalaids keep the main tasks unchanged
   - they act as reminders
   - they reduce memory load by providing external memory devices

3. *make things visible: bridge the gulfs of execution and evaluation*
   - make things visible on the execution side of an action so that people know what is possible and how actions should be done
   - make things visible on the evaluation side so that people can tell the effects of their actions

4. *get the mappings right*
   - exploit natural mappings
   - make sure the use can determine the relationships:
     - between intentions and possible actions
     - between actions and their effects on the system
     - between actual system state and what is perceivable by sight, sound, or feel
     - between the perceived system state and the needs, intentions, and expectations of the user
   - basic requirement: spatial relationship between the positioning of controls and the system or objects upon which they operate should be as direct as possible, with the controls either on the objects themselves or arranged to have an analogical relationship to them
   - movement of the controls should be similar or analogous to the expected operation of the system

5. *exploit the power of constrains, both natural and artificial*
   - use constraints so that the user feels as if there is only one possible thing to do

6. *design for errors*
   - assume that any error that can be made will be made and plan for it
   - allow the user to recover from errors, to know what was done andwhat happened, and to reverse any unwanted outcome

   - make it hard to do irreversible actions

7. *when all else fails, **standardize***
   - when something can´t be designed without arbitrary mappings and difficulties -> standardize
   - standardization is essential only when all the necessary information cannot be placed in the world or when natural mappings cannot be exploited
   - standardization is simply another aspect of *cultural constraints*

- some things are deliberately difficult to use – and ought to be
- but even deliberately difficult designs should not be entirely difficult
- usually there is one difficult part (designed to keep unauthorized people from using the device); the rest should be following the normal good principles of design

- increasing the number of controls can both enhance and detract from ease of use
- the more controls, the more complex things look and the more the user must learn about
- on the other hand: as number of controls increases up to the number of functions, there can be a better match between controls and functions

=> tradeoff between two opposing factors