HCI Lecture 1: Principles

Barbara Webb

Key points:

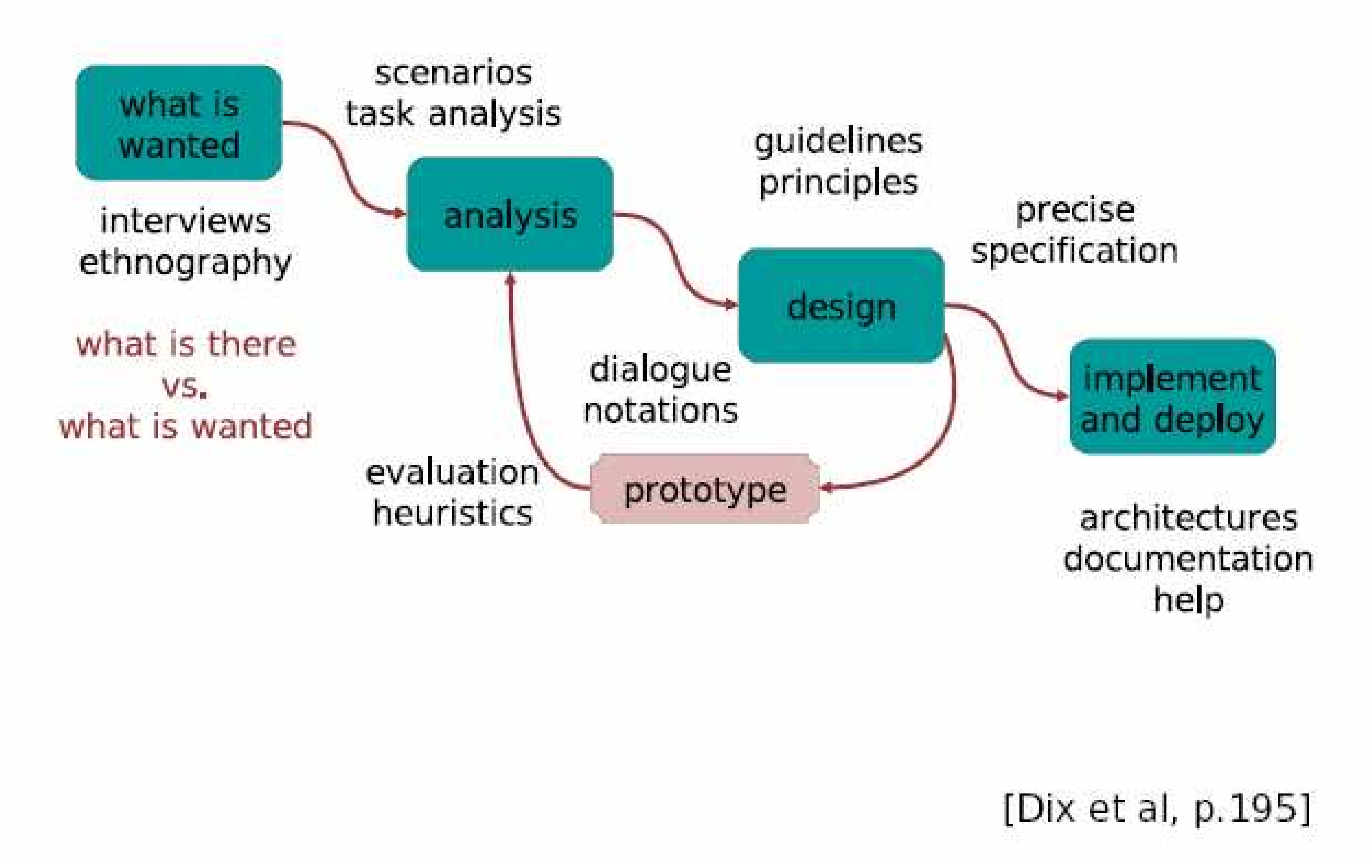
- Definition and related fields
- HCI in the design process
- General framework
- The action cycle and causes of error
- Standards, rules and principles

Definition and Related fields

- *"Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them."* Association for Computing Machinery
  - Equivalent terms are CHI and MMI
  - Usability Engineering focuses on design and implementation process
- Earlier fields of Human Factors & Ergonomics
  - More stress on physical issues; and on optimising (work) processes; concerns interaction with all kinds of human artifacts
- User Interface Design
  - Focus on interface, i.e. tends to assume deeper function is fixed
- User/Human Centred Design
  - Approach to software engineering with user focus at all stages
  - Participatory Design explicitly includes users in design process
- Interaction Design
  - Wider scope than computer, and more emphasis on cognitive/experiential factors than traditional HF.

HCI in the design process

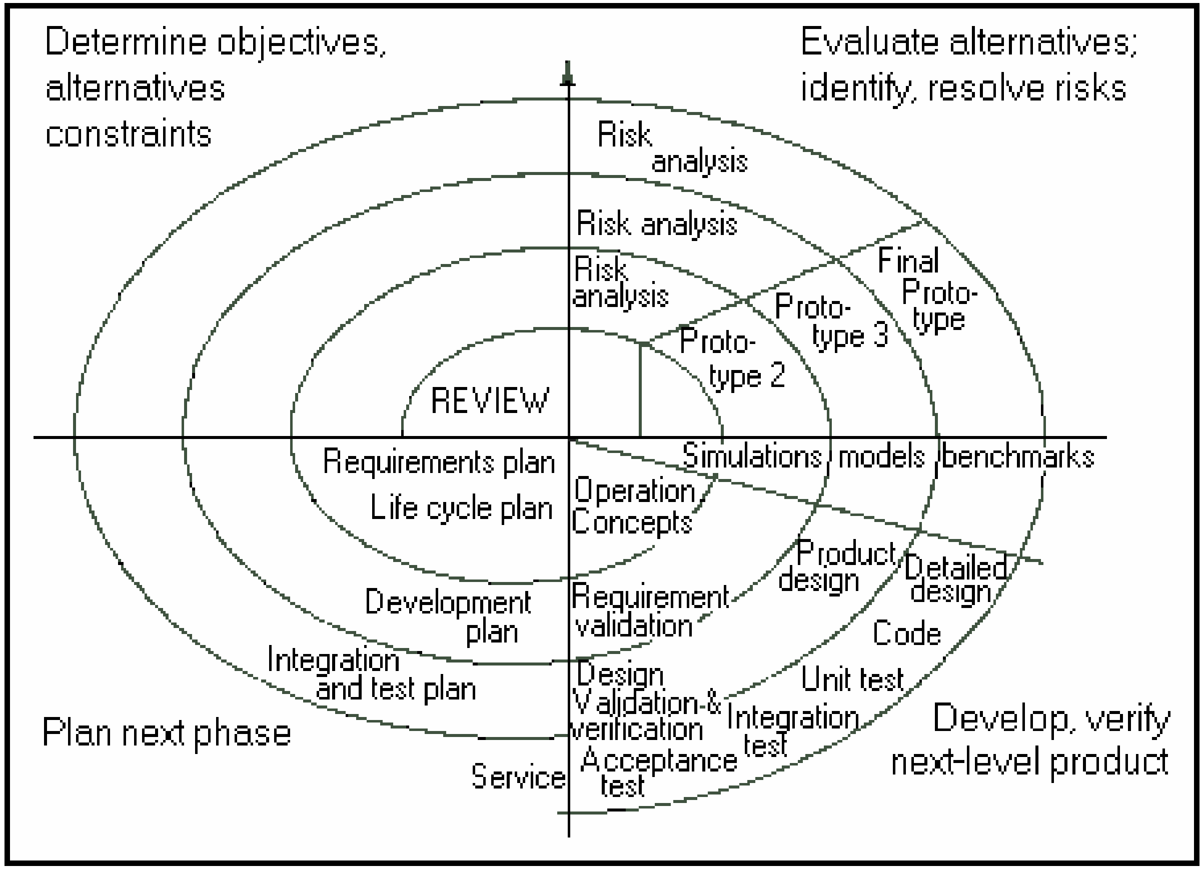
- Waterfall model



[Dix et al, p.195]

HCI in the design process

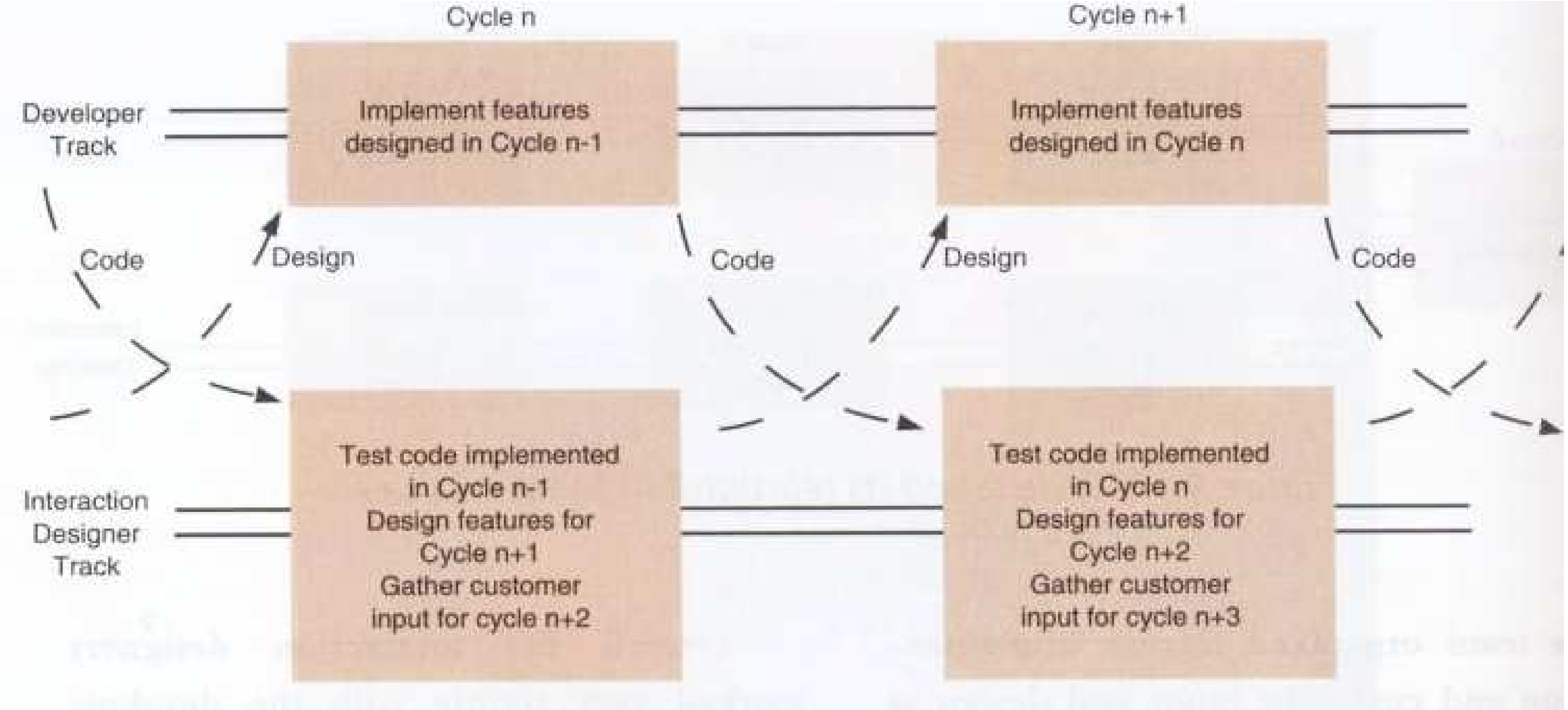
- The spiral lifecycle model (Boehm, 1988): importance of iteration in good design



Sharp, Preece & Rogers, 2007, p.451

HCI in the design process

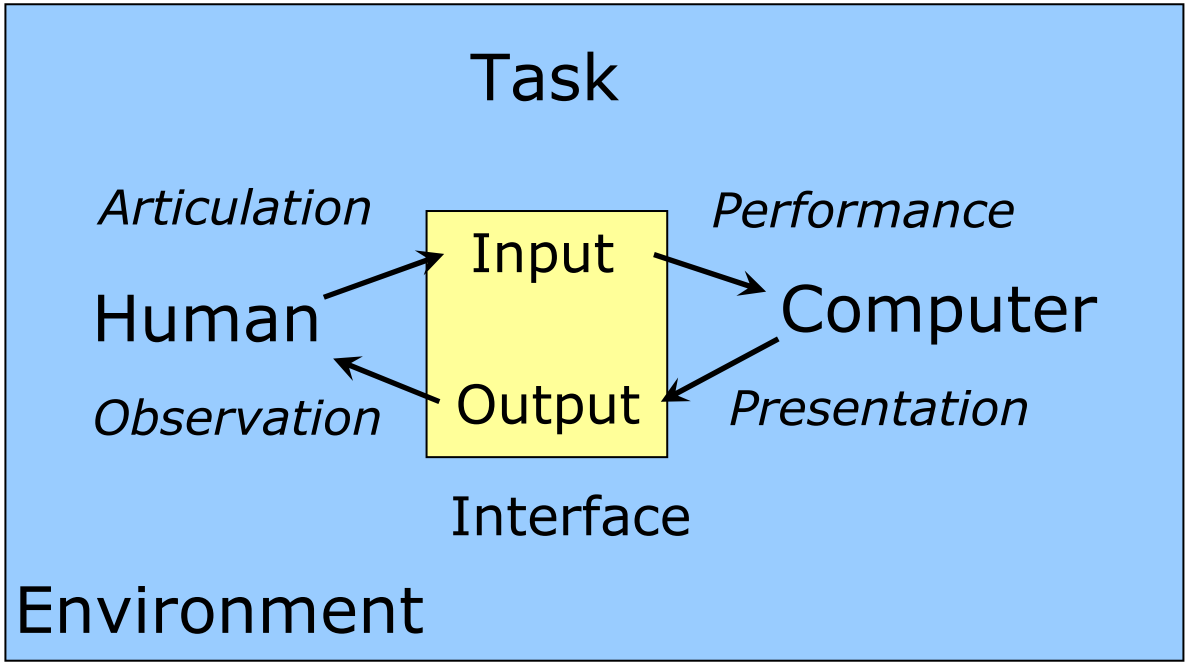
- Agile development e.g. eXtreme Programming (agilemanifesto.org): emphasises tight iteration in short timescales, close collaboration with customer



Cycle n
Cycle n+1
Developer Track
Implement features designed in Cycle n-1
Implement features designed in Cycle n
Code
Design
Code
Design
Code
Interaction Designer Track
Test code implemented in Cycle n-1
Design features for Cycle n+1
Gather customer input for cycle n+2
Test code implemented in Cycle n
Design features for Cycle n+2
Gather customer input for cycle n+3

Sharp, Preece & Rogers, 2007, p.458

General Framework



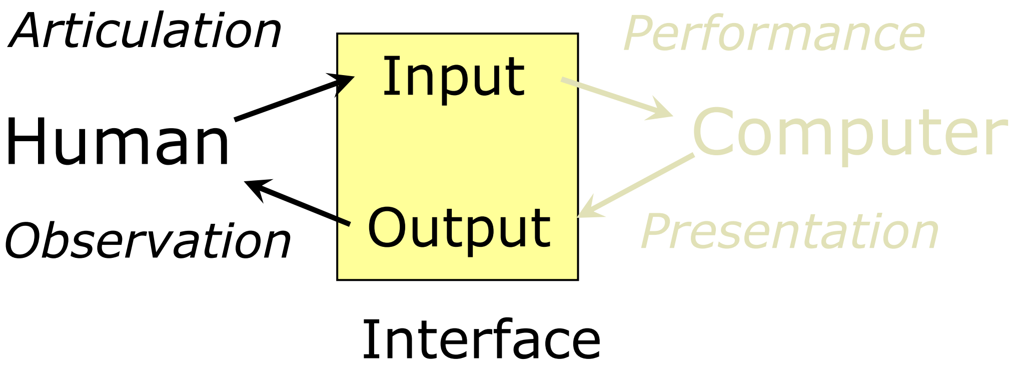
- Complex problem: we don’t have a full theory of the parts, or their interactions
- Note that a particular desired outcome may be best achieved by changing any one of these factors or interactions.

Why is interaction difficult?

- Examples of errors

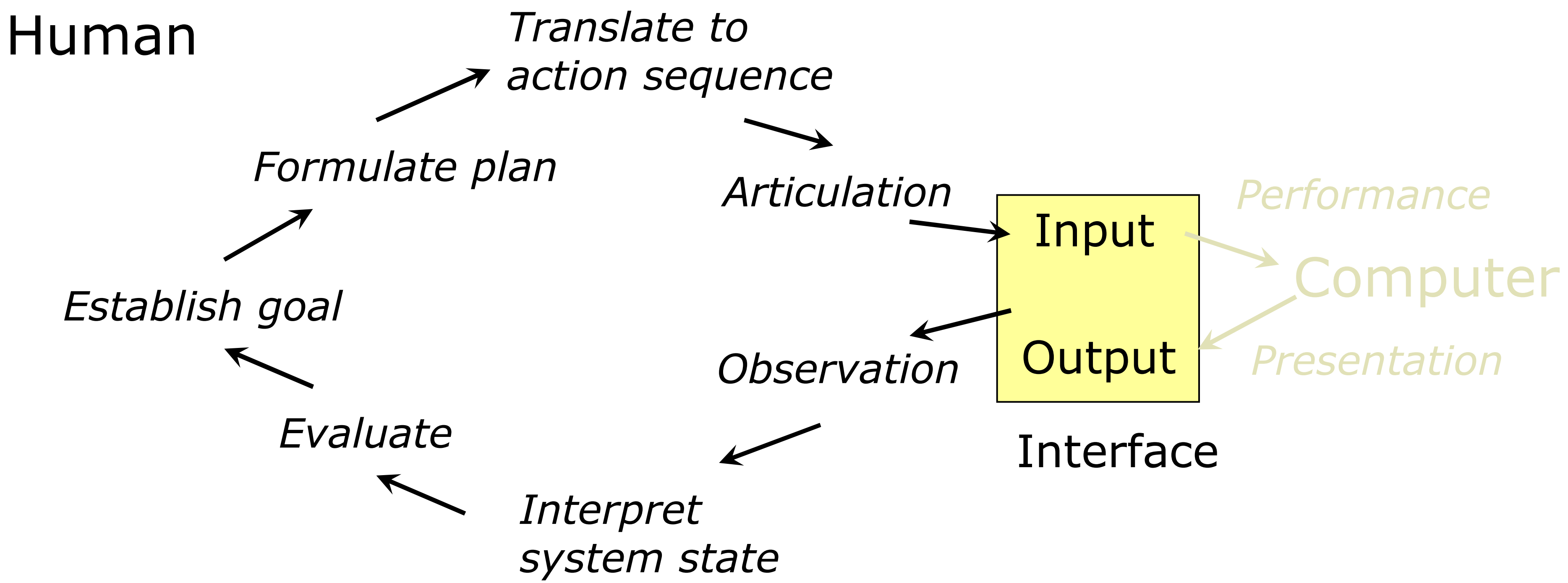
Why is interaction difficult?

- The problem from the human perspective



Why is interaction difficult?

- The problem from the human perspective

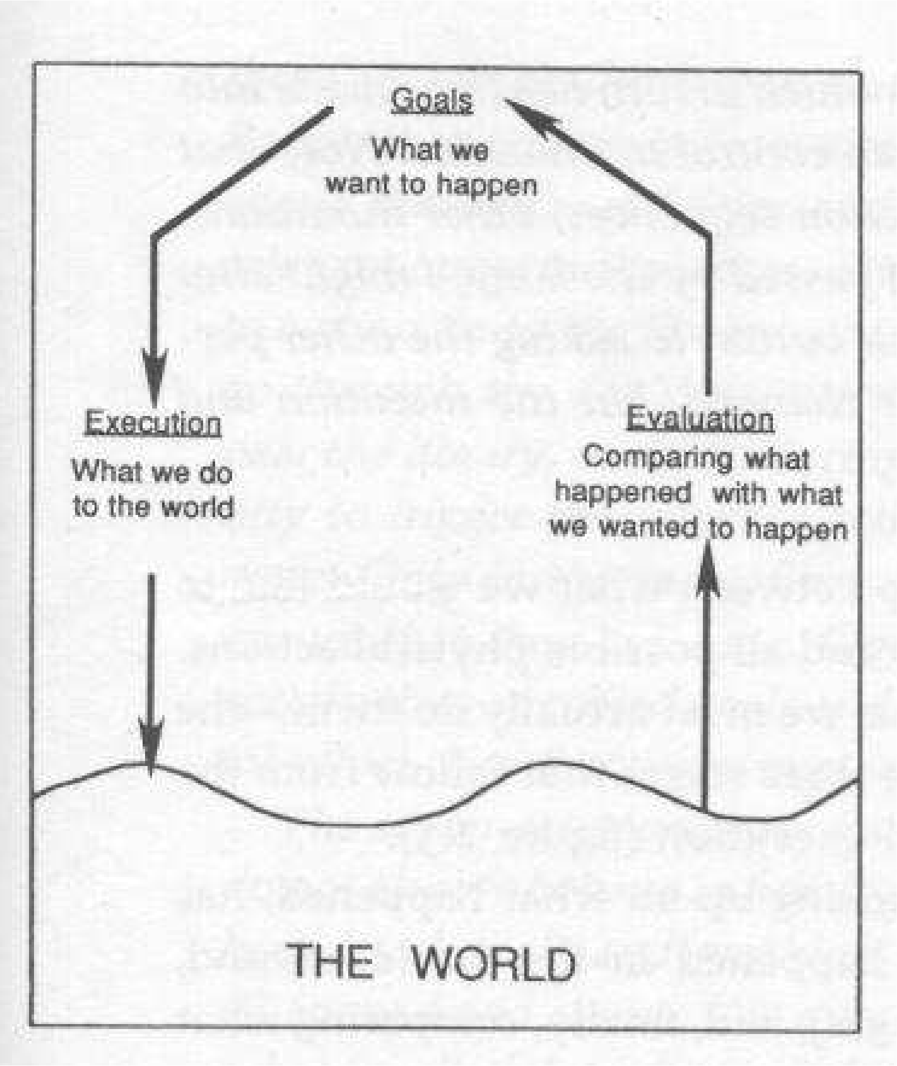


- This is called the ‘Action Cycle’ following Norman (1988)

Norman's Action Cycle

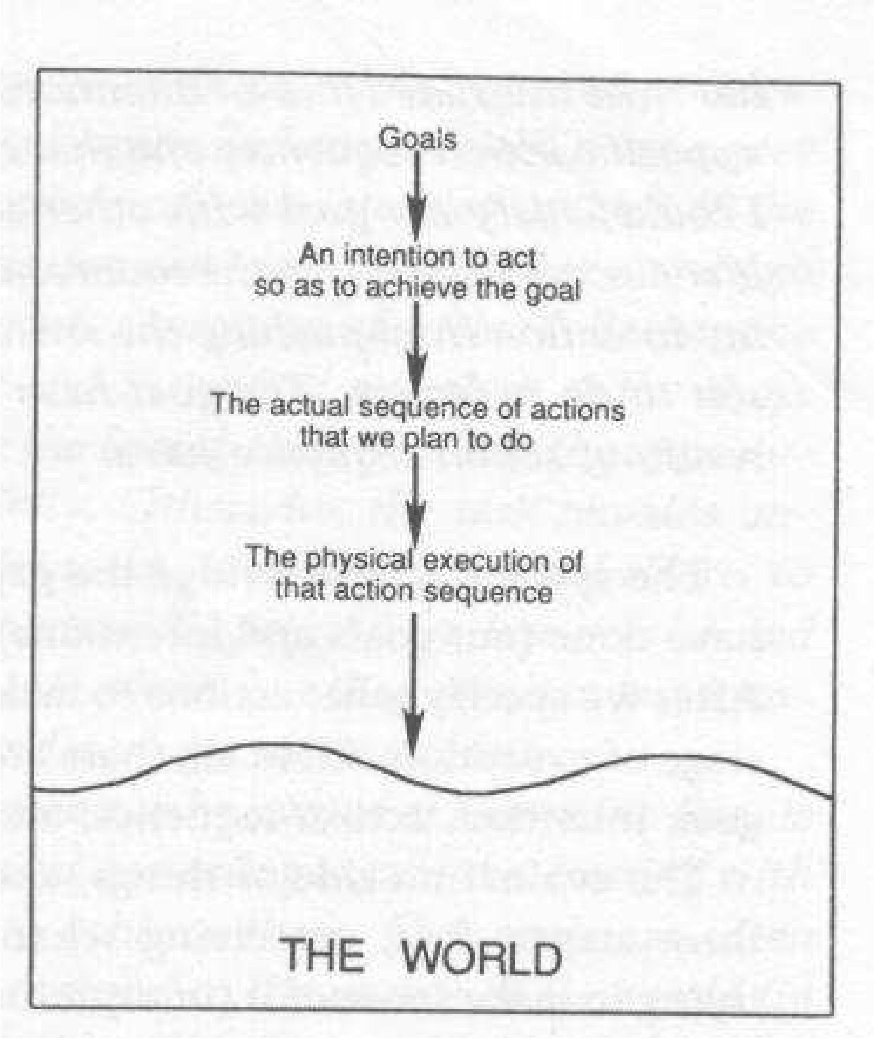
- Norman proposed that actions are performed in cycles:

1. Establish a goal
2. Execute the action
3. Evaluate the action



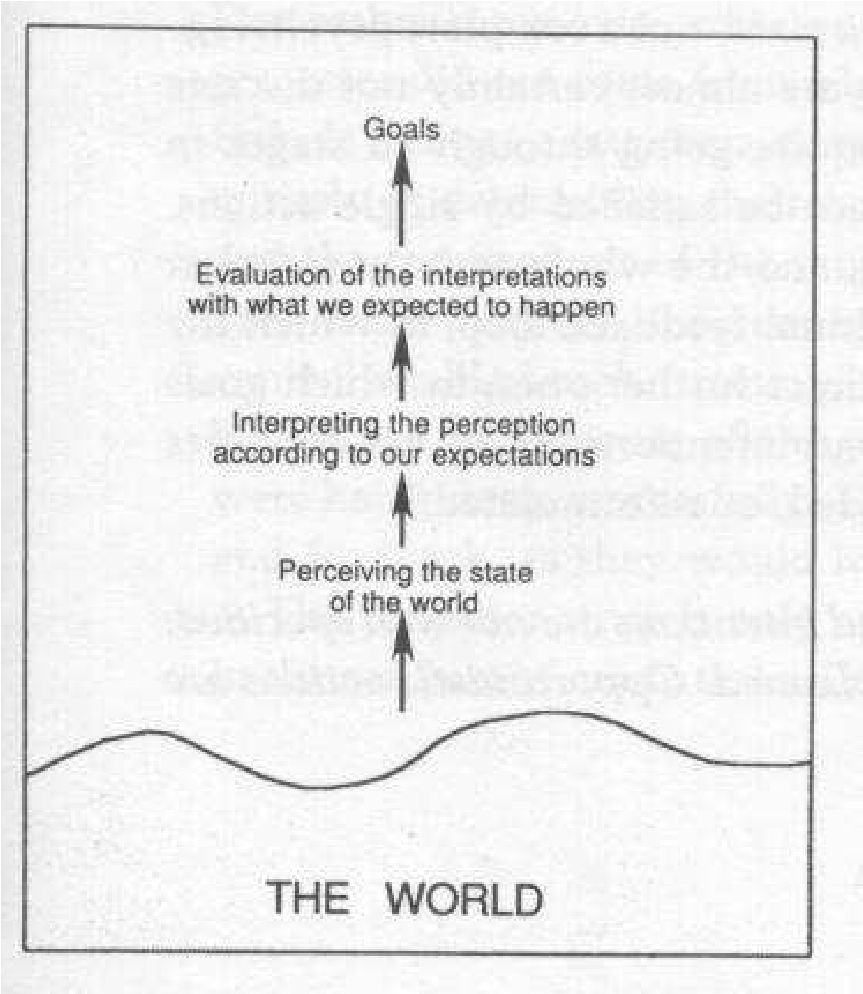
Norman’s Action Cycle

2. Stages of execution
   A. Form a plan for the task, a sequence of system operations to be performed on system entities
   B. Translate the plan into an action specification consistent with the interface ‘input’ language
   C. Output the action specification as a sequence of lexical tokens using the system input devices


Goals
An intention to act so as to achieve the goal
The actual sequence of actions that we plan to do
The physical execution of that action sequence
THE WORLD

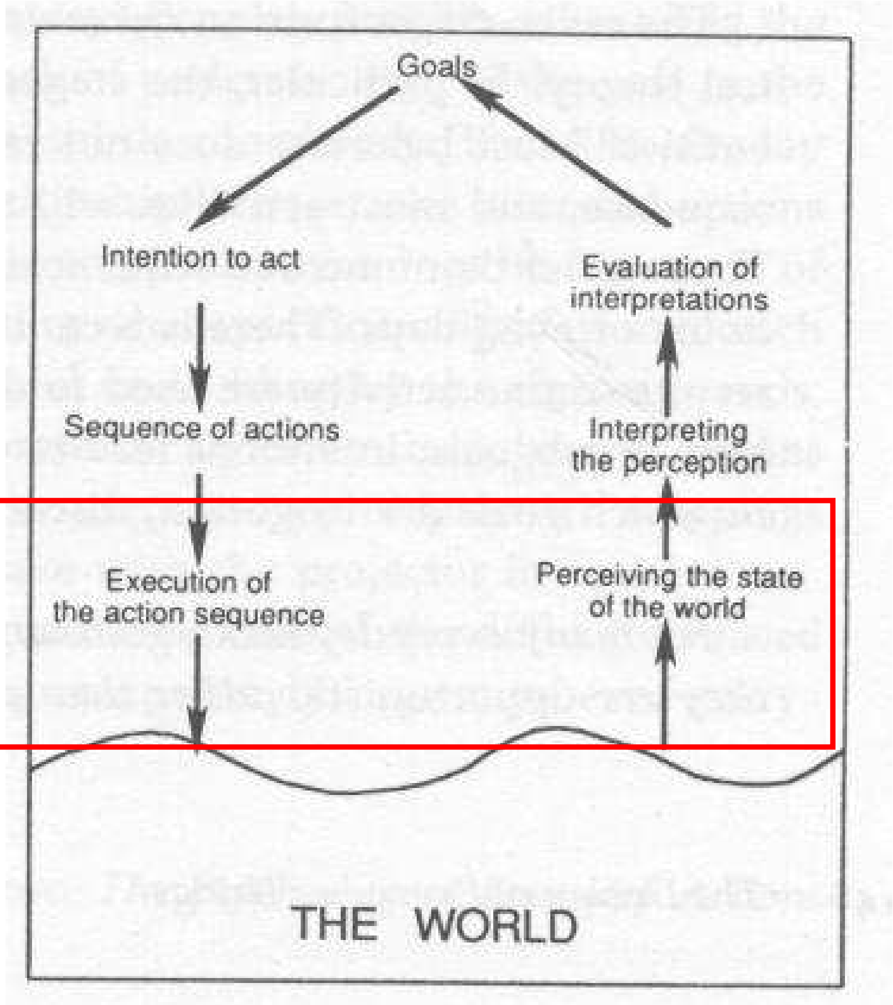
Norman's Action Cycle

3. Stages of evaluation
   - D. Observe the system response as e.g., visual, auditory tokens generated by output devices
   - E. Interpret the output in terms of changes in the system state, e.g., entity properties
   - F. Determine whether the new system state is consistent with what was intended


Goals
Evaluation of the interpretations with what we expected to happen
Interpreting the perception according to our expectations
Perceiving the state of the world
THE WORLD

Errors

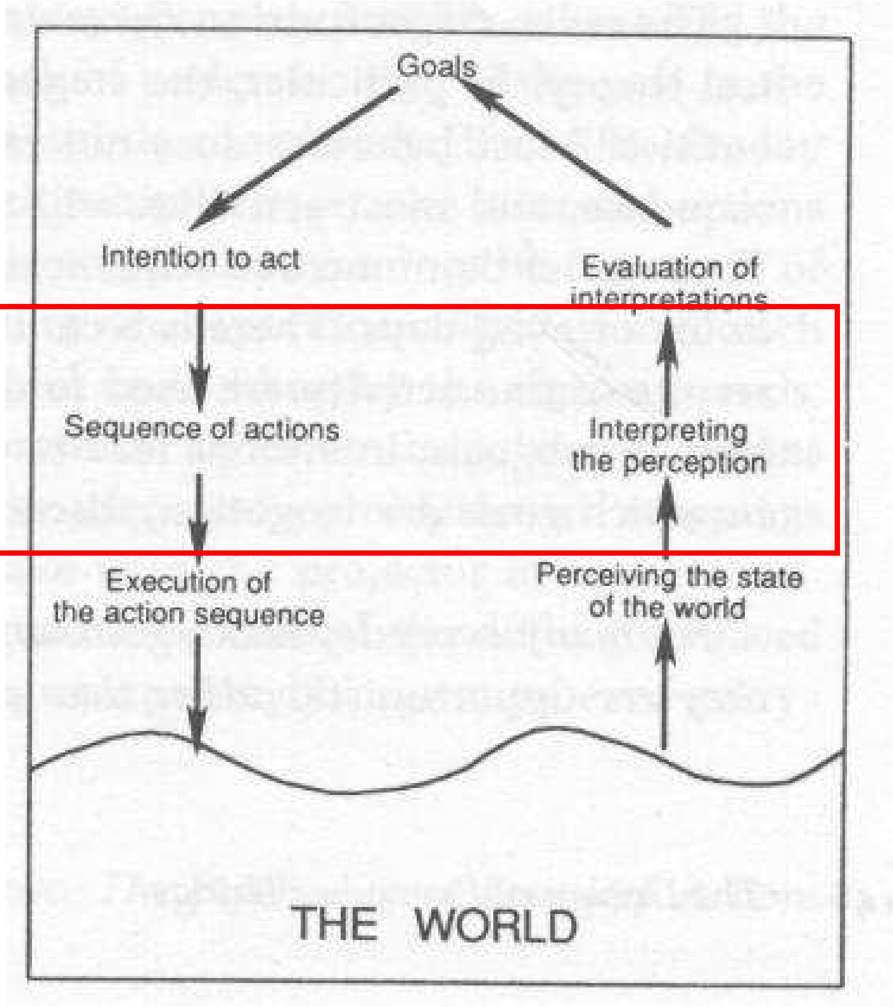
Low-level: slips of execution, misperceptions


Goals
Intention to act
Evaluation of interpretations
Sequence of actions
Interpreting the perception
Execution of the action sequence
Perceiving the state of the world
THE WORLD

Errors

Mid-level: problems in translation to machine I/O

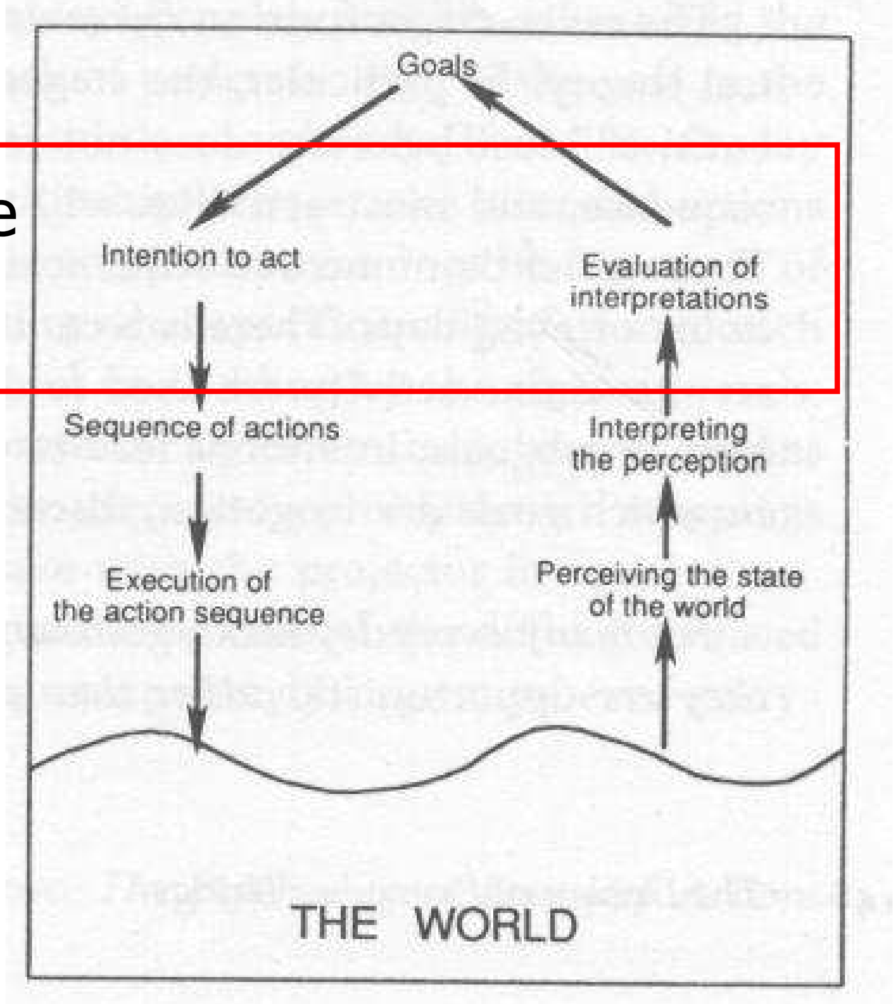
- Cannot produce input that the machine understands or produce input that it ‘misinterprets’ e.g due to mode
- Cannot understand or misinterpret the output from the machine e.g. due to mindset


Goals
Intention to act
Evaluation of interpretations
Sequence of actions
Interpreting the perception
Execution of the action sequence
Perceiving the state of the world
THE WORLD

Errors

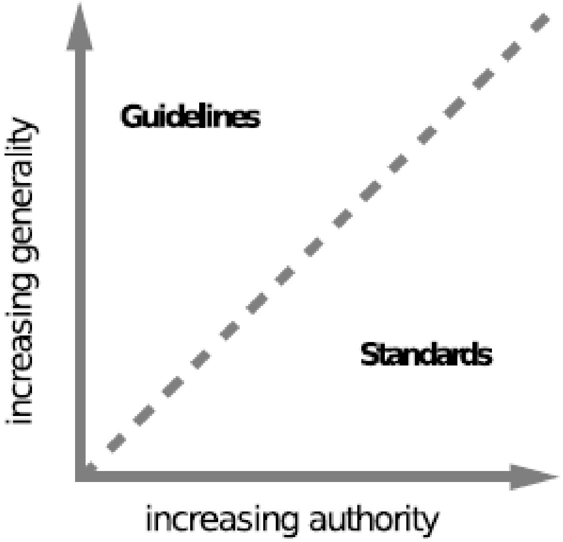
High-level: unable to conceive or recognise goal satisfaction

- Cannot determine correct thing to do, or make wrong choice
- Uncertainty about outcome or mistaken belief that have made progress towards the goal


Goals
Intention to act
Sequence of actions
Execution of the action sequence
Evaluation of interpretations
Interpreting the perception
Perceiving the state of the world
THE WORLD

Design Rules for HCI

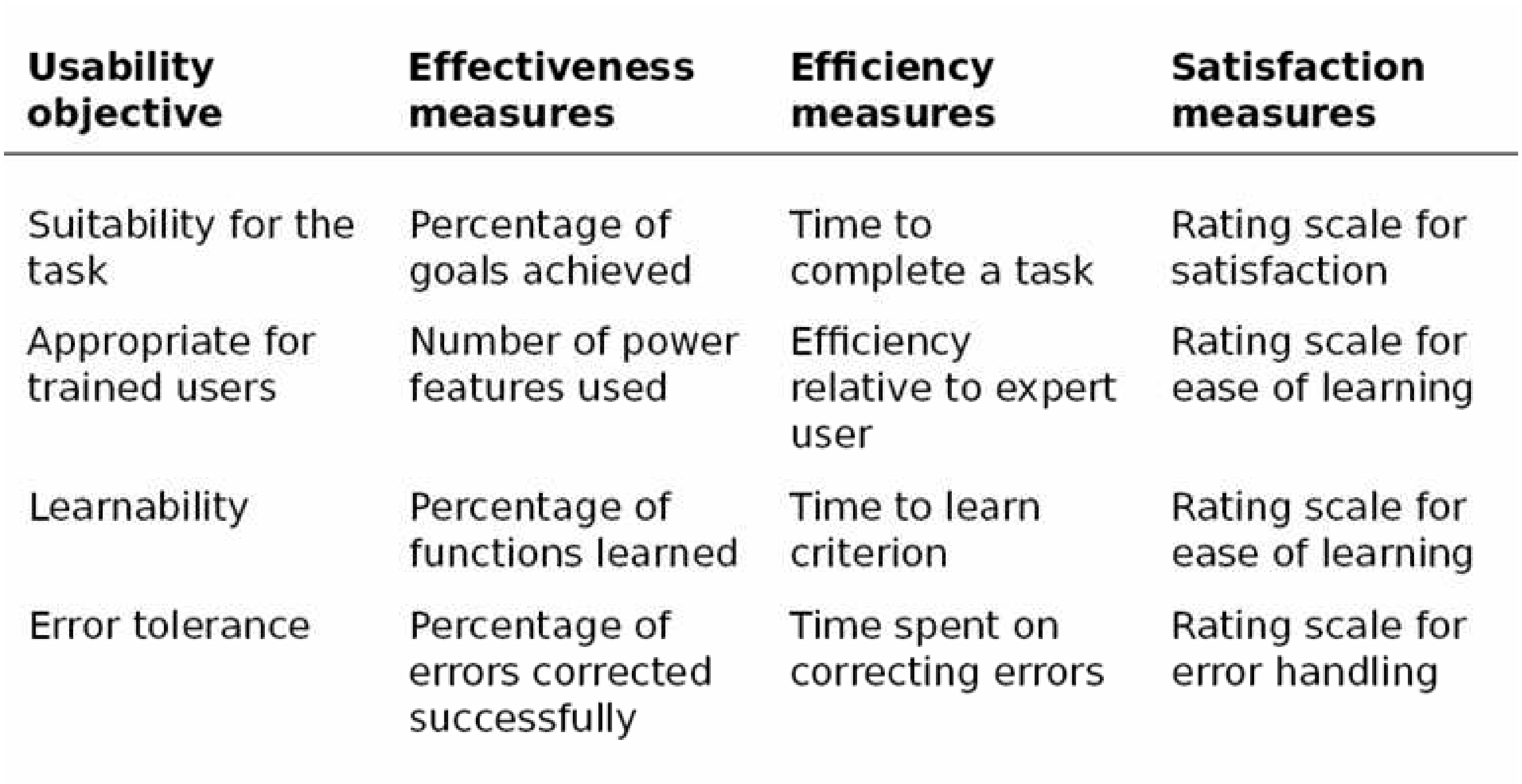
- Many sets of rules have been proposed to encapsulate understanding and best practice
  - Operate at various levels



- principles
  - abstract design rules
  - “an interface should be easy to navigate”
- guidelines
  - advice on how to achieve principle
  - may conflict; understanding theory helps resolve
  - “use colour to highlight links”
- standards
  - specific rules, measurable
  - “MondoDesktop links are RGB #1010D0”

Design Rules for HCI

- **ISO 9241**, *Ergonomics of Human System Interaction*, adopts traditional usability categories with specific measures, e.g.:



Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Suitability for the task	Percentage of goals achieved	Time to complete a task	Rating scale for satisfaction
Appropriate for trained users	Number of power features used	Efficiency relative to expert user	Rating scale for ease of learning
Learnability	Percentage of functions learned	Time to learn criterion	Rating scale for ease of learning
Error tolerance	Percentage of errors corrected successfully	Time spent on correcting errors	Rating scale for error handling

Shneiderman’s 8 Golden Rules (1987):

1. Strive for consistency
2. Enable frequent users to use shortcuts
3. Offer informative feedback
4. Design dialogs to yield closure
5. Offer error prevention and simple error handling
6. Permit easy reversal of actions
7. Support internal locus of control
8. Reduce short-term memory load

Norman’s 7 Principles (1988):

1. Use both knowledge in the world and knowledge in the head.
2. Simplify the structure of tasks.
3. Make things visible.
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.

Nielsen’s 10 Usability Heuristics (1994):

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Help users recognize, diagnose and recover from errors
6. Error prevention
7. Recognition rather than recall
8. Flexibility and efficiency of use
9. Aesthetic and minimalist design
10. Help and documentation

Consolidate the three lists here. Mark any you don’t understand with *

E.g. Consistency (Nor 7,Shn 1, Nie 4)

Usability Principles from Dix Chapter 7

Dix groups these and related principles as follows:

- **Learnability**
  - the ease with which new users can begin effective interaction and achieve maximal performance (e.g. familiarity, generalisability, predictability)
- **Flexibility**
  - the multiplicity of ways the user and system exchange information (e.g. customisability, substitutability, user control)
- **Robustness**
  - the level of support provided to the user in determining successful achievement and assessment of goal-directed behaviour (e.g. observability, recoverability)

Where do these rules come from?

- Many seem like common sense - but often violated
  - Home exercise: pick one everyday object and one piece of software and assess with respect to these rules
- Some are grounded in our understanding of how humans perceive, think and learn (c.f. next lectures)
- Some are the result of empirical study (e.g. Nielsen's heuristics are based on factor analysis of 249 usability problems)
- Some are derived from particular characterisations of the nature of human action (e.g. Norman's principles are closely related to his theory of action)
- Some are collections of experience (e.g. Shneiderman's rules)
- Some can be directly related to computational complexity
- In this course we will study the background and justification of these rules and elaborate on how they can be applied in specific contexts to design and assess human computer interaction.

References/Further Reading

- Dix et al., 3rd ed: chapter 7

- H. Sharp, Y. Rogers & J. Preece (2007) Interaction Design. John Wiley & Sons, Chichester. (See also www.id-book.com)
- Shneiderman (1987) Designing the user interface: strategies for effective human computer interaction. Addison-Wesley, Reading MA.
- Nielsen (1994) Enhancing the explanatory power of usability heuristics. Proceedings of the ACM CHI’94 Conference (see also www.useit.com)
- D.A. Norman (1988) The Design of Everyday Things. Doubleday, New York.