Reflection

Fundamentals of Human-Centered Computing

Reflection

Classical theories

Norman's theory, cognitive modeling

Modern theories

Distributed Cognition, Situated Action, Activity theory

Contemporary theory

A turn to design, culture, the wild, and embodiment

And any of your questions!

Classical theories

Norman's theory, cognitive modeling

Classical theories

Resolve the differences between cognitive scientists and engineers

- By building a bridge between the disciplines
- By translating the findings of cognitive science into applications

Improvement: move from studying the mind, to studying the mind **as it uses technology**

Norman's Theory

The action cycle and gulfs of execution/evaluation

> Explains how people use interfaces, and why they sometimes fail

Designer image, system image, use image

> Explains what causes some systems to be less usable than others

Constraints, signifiers, and feedback

> Explains how you can increase the usability of interfaces

The action cycle

1. **Plan** to turn my goal into an intention to act

2. **Specify** an action sequence

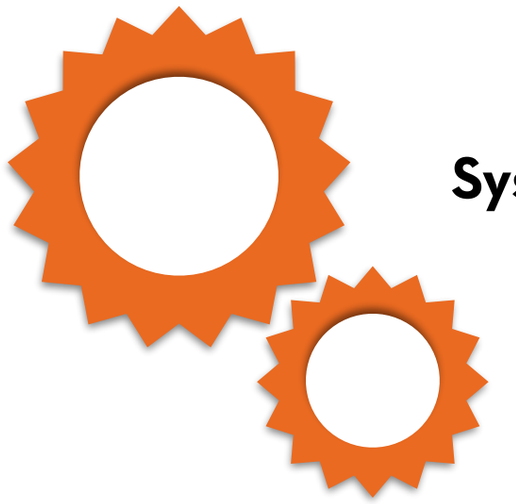
3. **Perform** this sequence

4. **Perceive** the change

5. **Interpret** the dialog

6. **Evaluate** the outcome

Program

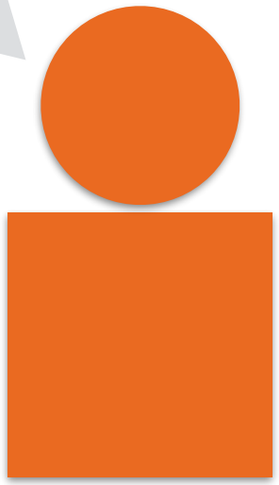


System image

Designer's model

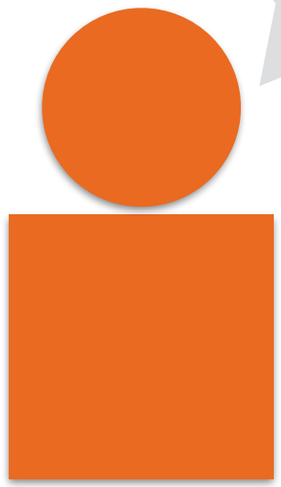
operate

User's model



design

interpret



Designer

Interface

User

Philosophy

The shared philosophy between DCog, SA and AT is that they all study systems in the **context** in which they are used

- DCog: the context is the system
- SA: the context is the situation
- AT: the context is the plan (anticipatory reflection)

User interface

Certain aspects of a user interface can help align the user's model and the system image:

- Constraints
- Signifiers
- Feedback

Cognitive modeling

Cognitive architecture: a specification of the structure of the brain at a level of abstraction that explains how it achieves the function of the mind

GOMS models: quantitative prediction of expert users' interaction performance

Cognitive Walkthrough: find out how novice users work with an interface

ACT-R

Visual: see the problem

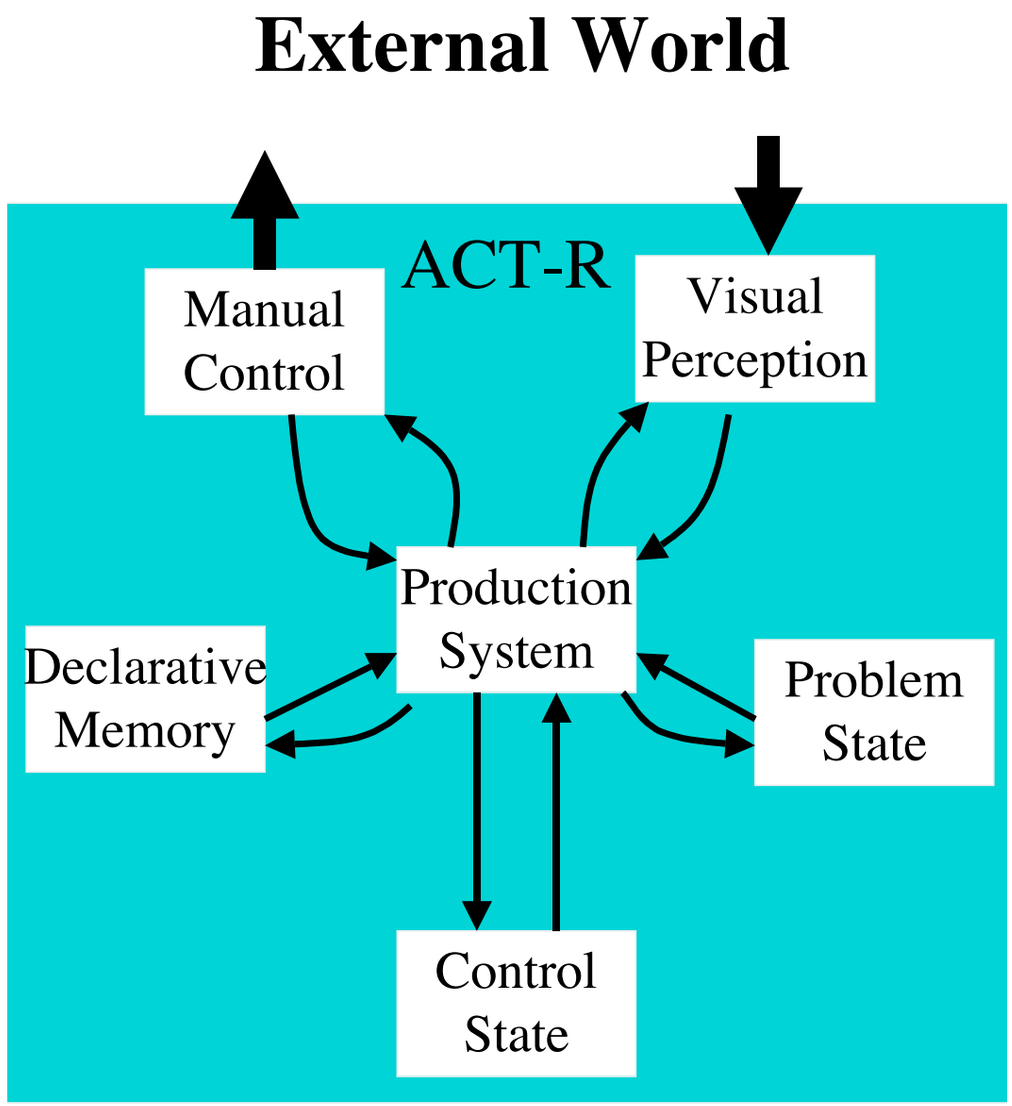
Problem state: STM-ish

Control state: objective

Declarative: LTM

Manual: create output

Production system: move things between modules



Keystroke modeling

List the overt actions necessary to do the task

- Keystrokes and button actions (K), mouse movements (P), hand movements from keyboard to mouse (H)

Also system response time (if user has to wait)

- Insert mental operators (M)

Assign execution times from previous research

- K, M, H are straightforward; P requires Fitts's Law

Add up the execution times

Cognitive walkthrough

Four principles of fulfilling tasks:

Q1. Will the user try to achieve the right effect?

Q2. Will the user notice that the action is available?

Q3. Will the user associate the action with the effect?

Q4. Will the user see progression?

Modern theories

Distributed Cognition, Situated Action, Activity theory

Modern theories

Distributed Cognition

Cognition as external, distributed, constrained by the environment

Situated Action

Cognition as improvised, situated

Activity Theory

Cognition is influenced by historical social context through anticipatory reflection

Modern theories

Structuration Theory

Technology is interpretively flexible; adoption is key

Boundary Objects

Technology can facilitate collaboration at the boundaries between different (types of) people

Distributed cognition

Combination of people, systems, and artifacts is a **cognitive system**

Paper and computer hard-drives are part of our distributed memory

Artifacts play an active role in cognition

Communication (especially when mediated) results in reinterpretation and loss of context

Flexible artifacts

Artifacts take an active role

They are not just stimuli or work output

Most successful examples of DCog show how people exploit the flexibility of the digital world

Look for secondary usage patterns

Consider representations as both abstract forms as well as the thing that is being represented

E.g. a form on my desk can be both a tool and a reminder

Memory as a process

Organizational memory resides in several individuals, objects and systems within an organization

- Both explicit and implicit

Memory can be viewed as both an entity and a process

- Memory processes are the transition of knowledge between humans and artifacts
- E.g. teaching a method, having a project meeting, assigning a task, writing down rules

Context

Knowledge transition happens through (mediated or direct) communication

Communication (especially when mediated by technology) results in reinterpretation and loss of context

- For efficiency reasons, the sender decontextualizes the information
- The receiver then has to recontextualize the information
- This process is not infallible, since contexts may be different for sender and receiver

Breakdowns

Result of this de- and re-contextualization? Breakdowns!

This makes it difficult to reuse knowledge

As a result, reuse is often limited to simple, familiar and frequently used pieces of information

Goal of a good information system: maintain context!

Situated Action

Situation Action assumes that actions are constrained and supported by social and physical circumstances

People use these circumstances to achieve their goals

Goals are retrospective reconstructions of what happened

The situation is the driving factor

Situation is an essential resource that makes knowledge and action possible

Situation enables and constrains knowledge and action

Abstractions

Distributed coordination

How are tasks divided? Does this happen ad hoc or by plan?

Plans and procedures

Compare against real actions: do they allow procedures to take hold? If not, why not?

Awareness of work

How actions are communicated or made visible to others?
One person's action is another person's context

Design implications

Allow systems to understand and support the actions and circumstances of the users

- If impossible, compensate for the lack of context

Demonstrate the limitations of the machine to the user

Allow ad-hoc coordination and signaling between users

Support rather than enforce adherence to procedures

- Allow people to do the work in whatever way they want

Practical result

Make technology fit the work practice

Rather than the other way around

If you assume that work is conducted according to procedures, your system becomes a mere repository for outcomes

It cannot assist the actual actions taken to do the work

Situated action approach: embrace the inherent ambiguity of work, thereby creating a tool for doing the work

Supporting situated rather than modeled interactions

Activity Theory

Activity Theory studies how people perform activities by interacting on several **different levels**:

- Activity, action, operation

Plans are **anticipatory reflections** of recurring activity

- This gives activity theory a partially-situated approach
- But the locus of control is still with users themselves

Plans

Because plans and activities start out as external and collective, **culture** and **society** transform all our activities, and in turn, our minds

> Hence, plans and activities are socially constructed, and may evolve in the course of action (short term) and over time (long term)

Artifacts

Good systems support **full activities** rather than just actions or operations

How? By implementing (rather than ignoring) the plans

But it has to be done flexibly!

Structuration

Structure is both a product and a constraint of actions

An **exogenous factor** (or a strategic change) may trigger a new social dynamic

- Technology could be one such exogenous factor

This may in turn **change the structure**

- This can be intentional (the technology was supposed to change the structure) or unintentional

Implications

It is important to support or at least acknowledge the **preexisting constitutional constraints** as a baseline

Adoption may occur **in stages**

The effects of technology may be **delayed**

The same technology and the same structuring process may still results in **different** social dynamics

Focus

Focus on the **consequences** of a change in structure:

Signification (distribution of knowledge)

Domination (distribution of power)

Legitimation (distribution of norms and values)

Flexibility and context are important!

Boundaries

In collaboration/coordination across heterogeneous entities, there exist **knowledge boundaries**

Boundary objects are shared and shareable across different problem solving contexts

- Repositories/tools (syntax, represent)
- Standardized forms and methods (semantics, learn)
- Maps, models and objects (pragmatic, transform)

Pragmatic boundaries have consequences!

Implications

A good system fulfills all three functions:

- It provides a repository for representing
- It provides forms/methods for learning
- It provides maps/models/objects for transforming

As a boundary object, a system is both practical and political

It must facilitate a process of transforming knowledge that is localized, embedded, and invested

This process is called **interessement**

Contemporary theory

Fundamentals of Human-Centered Computing

Contemporary theory

From **needs** to **values**

Health, fairness, activism

From **cause and effect** to **accountability**

i.e. from empiricism to philosophy

Focus on **social responsibility** and **moral narratives**

Contemporary theory

A turn to design: Technology as experience

Design to provoke thinking

A turn to culture: Critical Theory

Being skeptical about HCI and design artifacts

A turn to the wild: Ecological rationality

The use of heuristics

A turn to embodiment: Embodied interaction

Integrate tools into our body image

Questions

Your turn!