Wisdom is not the product of schooling but the lifelong attempt to acquire it.
- Albert Einstein

AI and Creativity

Gerhard Fischer

AI Course, Fall 1996, Lecture, Nov 25th, 1996

Overview

- **Creativity**
  - in general
  - in humans
  - in machines
  - AI and Creativity
- **Concepts for Creativity**
  - breakdowns and learning from mistakes
  - symmetry of ignorance
  - bring tasks to the foreground
- **Creativity and Computational Environments**
  - requirements
  - SER Model
  - Learning on Demand and End-User Modifiability
  - Examples: Evolution towards Design Environments
- **Assessment**
- **Conclusions**

Creativity

- valuable consequences
- novel or surprising
- high motivation and persistence
- vague and ill-defined problems
- **claim**: “there is no procedure which will guarantee that a person will invent something important or initiate a new artistic movement”
- **claim**: “design is a good domain to study creativity, because design problems are intrinsically open-ended, situation specific and controversial”

Historical versus Psychological Creativity

- **our goal:** creativity is often associated with art — we (i.e., L^3D) are concerned with creativity that is required in everyday work practice by emphasizing the importance of lifelong learning during these activities
- **claim:** the analysis of everyday design practices has shown that knowledge workers and designers have to engage in creative activities in coping with the unforeseen complexities of everyday, real-world tasks
- **historical creativity** = the activity or the product is novel or original to society as a whole
- **psychological creativity** = the activity or the product is personally novel and meaningful to the individual who produced it
- **claim:** while analyzing outstanding creative people contributes towards establishing a framework for creativity — understanding creativity in the context of everyday activities is equally important for letting people become more productive and create better work products

Different Research Directions Related to Creativity

- descriptive accounts of creativity
- make computers creative
- build computational environments which augment and empower the creative potential of individuals and groups (implying: augment the skill of designers -- do not “de-skill” them)

Creativity = f{Doing, Working, and Learning}

- breakdowns as opportunities
  - learning from mistakes
  - we can make mistakes only when we *do* something — not when we engage in passive observation

Norman — *"Real Learning: The way we learn is trying something, doing it and getting stuck. In order to learn, we really have to be stuck, and when we're stuck we are ready for the critical piece of information. The same piece of information that made no impact at a lecture makes a dramatic impact when we're ready for it."*

Learning from Mistakes

- **Oscar Wilde:** “Experience is the name every one gives to their mistakes.”
- **John Archibald Wheeler:** “Our whole problem is to make the mistakes as fast as possible.”
- **Henry Petroski: “**The colossal disasters that do occur are ultimately failures of design, but the lessons learned from those disasters can do more to advance engineering knowledge than all the successful machines and structures in the world.”

Learning from Mistakes

- **concepts associated with learning from mistakes:**
  - debugging
  - critics
  - error elimination (Popper: “Conjectures and Refutations”)
  - learning from real-world disasters (Lee: “The Day The Phones Stopped”)

- breakdowns in design
  - back-talk of the situation supports reflection-in-action
  - “artifacts do not speak for themselves” (Rittel) — implying conversations with design materials are limited in their back-talk
  - “ready-to-hand” ---> “present-at-hand” (Heidegger’s hammer)
  - activation of tacit knowledge

Potential Benefit and Impact of Computer on Mistakes / Breakdowns

- allow us to make mistakes more quickly
- allow us to make mistakes in safe environments (undo commands, flight simulators)
- help us in recognizing mistakes / breakdowns
  - at the tool / medium level
  - at the content level
- contextualize information to the task at hand
- make the *important* invisible visible

Symmetry of Ignorance as a Source for Creativity in Communication and Coordination Processes

designer(s) <----> user(s) / client(s)

- clients do not know what they want (analogies to architecture)
- software designers suffer from the “thin spread of application knowledge”
- languages of doing (instead of formal representations) are needed for mutual understanding and referential anchoring

design teams

- groups with different interests
- who is the beneficiary and who has to do the work? (e.g., comments, design rationale, design for redesign)

designer(s) and knowledge-based design environment

- indirect, long-term communication (group memory, design artifact memories)
- multi-user environments enhancing single users with shared artifacts for collaboration

From “Teacher / Learner” and “Expert / Novice” to Symmetry of Ignorance

- **we need to redefine** our models of teachers and learners, and of expert and novices from attributes of a person to attributes of a specific context

- **claim:** reciprocal learning and teaching (people helping each other) is more characteristic for real work environments than a role separation in teachers and learners, and in experts and novices

Human Creativity = f{Medium}

quote: “you cannot use smoke signals to do philosophy. Its form excludes the content” (Postman, “Amusing Ourselves to Death”, p 7)

claim: we cannot use most current computer systems to be creative

challenge: design of environments (social and technical)

- supporting creativity
- allowing us to think previously unthinkable thoughts, do previously undoable actions, and explore previously unfeasible questions

collaborative human-computer systems: to specify a division of labor between human and computer

- What part of the task should be exercised by human beings?
- What part by the of the task should be exercised by the computer?

Why have there been so few Creative Solutions in Computational Media?

- **weak generators -- no need for selection criteria / critics:**

  COMPUTER OUTPUT LOOKED LIKE THIS

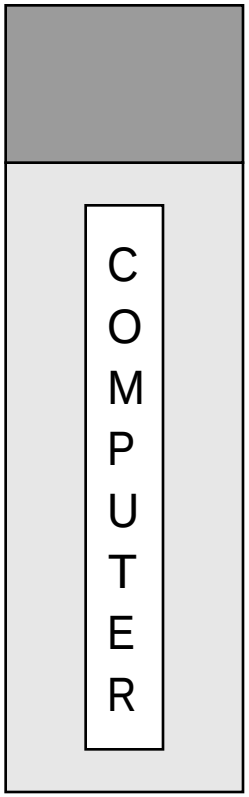
- **powerful generators — need for selectors and critics:**

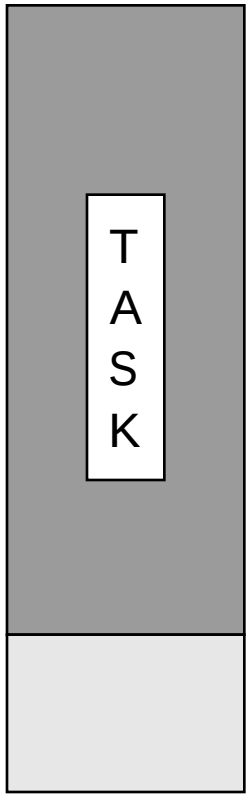
  Powerful *tools* can sometimes be **powerfully** abused!!

How can Creativity be Enhanced?

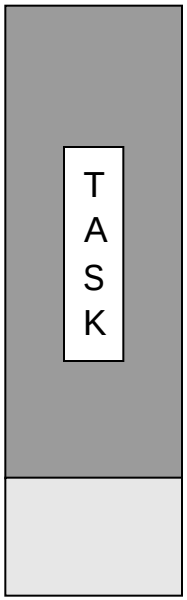
- develop intelligent support environments
- support distributed cognition: combine knowledge in the head with knowledge in the world
- look for analogies and the impact of representation on problem difficulty
- supporting the incremental unfolding of design spaces
- reuse and redesign
- exploit what people already know (user modeling, differential descriptions)
- support contextualized explanations and argumentation
- take care of low-level clerical details
- putting owners of problems in charge (delegation suppressing creativity by the fact that “situations” do not talk back to the owners of problems)
- create flow, affection and appropriation

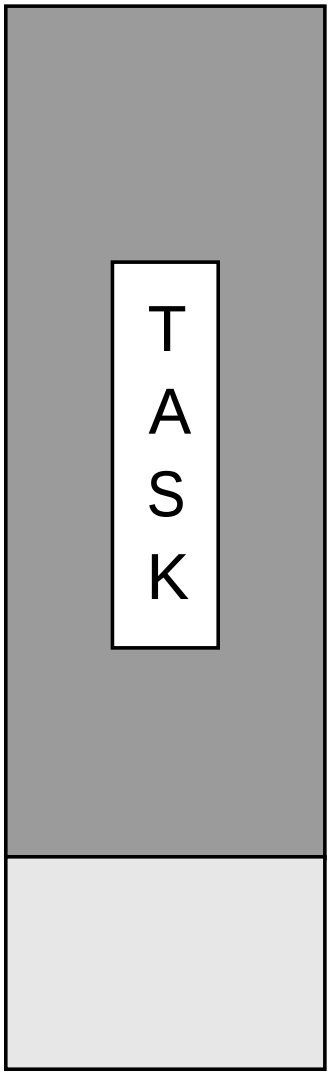
Human-Computer Interaction
-------->
Human Problem-Domain Interaction





Making New Tasks Possible





Requirements for Creativity Supporting Environments

human computer interaction	----->	human problem-domain interaction
usable	----->	usable + useful + authentic + engaging + motivating + self-directed + modifiable
problem solving	----->	problem framing and problem solving
action and reflection separated	----->	reflection-in-action
instructionism	----->	constructionism
consumers	----->	designers
individual creativity	----->	organizational / collective creativity

Integration of Theory, System Building and Assessment

theory: reflection-in-action, breakdowns, symmetry of ignorance,
human problem-domain communication

systems: layered architectures, agentsheets, domain-oriented design
environment, critiquing, learning on demand, end-user modifiability

assessment: naturalistic environments, role of critiquing, creativity is everywhere,
problems are not solved, but reformulated (dissolved)

Example: From General Purpose Programming Languages to Domain-Oriented Design Environments

General Purpose Environments	---->	limited reuse
Object-Oriented Design	---->	lack of domain-orientation
Domain-Oriented Construction Kits	---->	no feedback about quality of artifact
Constructive Design Environments	---- >	design is an argumentative process
Integrated Design Environments	---->	lack of shared context
Multifaceted Architecture	---->	limited evolution

End-User Modifiable (Programmable) Design Environments

Working Hypotheses (H), Challenges (C) and Environments (E) Supporting Creativity

- **H: the choice of tasks and goals must be under the control of the user/learner**
  C: creating systems which are *simultaneously* user-controlled and supportive
  E: domain-oriented design environments

- **H: new information must be relevant to "the task at hand"**
  C: saying the "right" thing at the "right" time in the "right" way
  E: shared understanding, agents, information delivery

- **H: "symmetry of ignorance" exists**
  C: requires representations for mutual understanding and learning
  E: shared artifacts and shared context

- **H: "breakdowns" and "suggestions" must lead to opportunities for learning**
  C: artifacts do not speak for themselves
  E: critics, simulation, argumentation, catalogs (case repositories)

Convivial Tools and Creativity

- **user control:**
  - end-user modification and programming
  - independence of high-tech scribes
  - putting owner of problems in charge

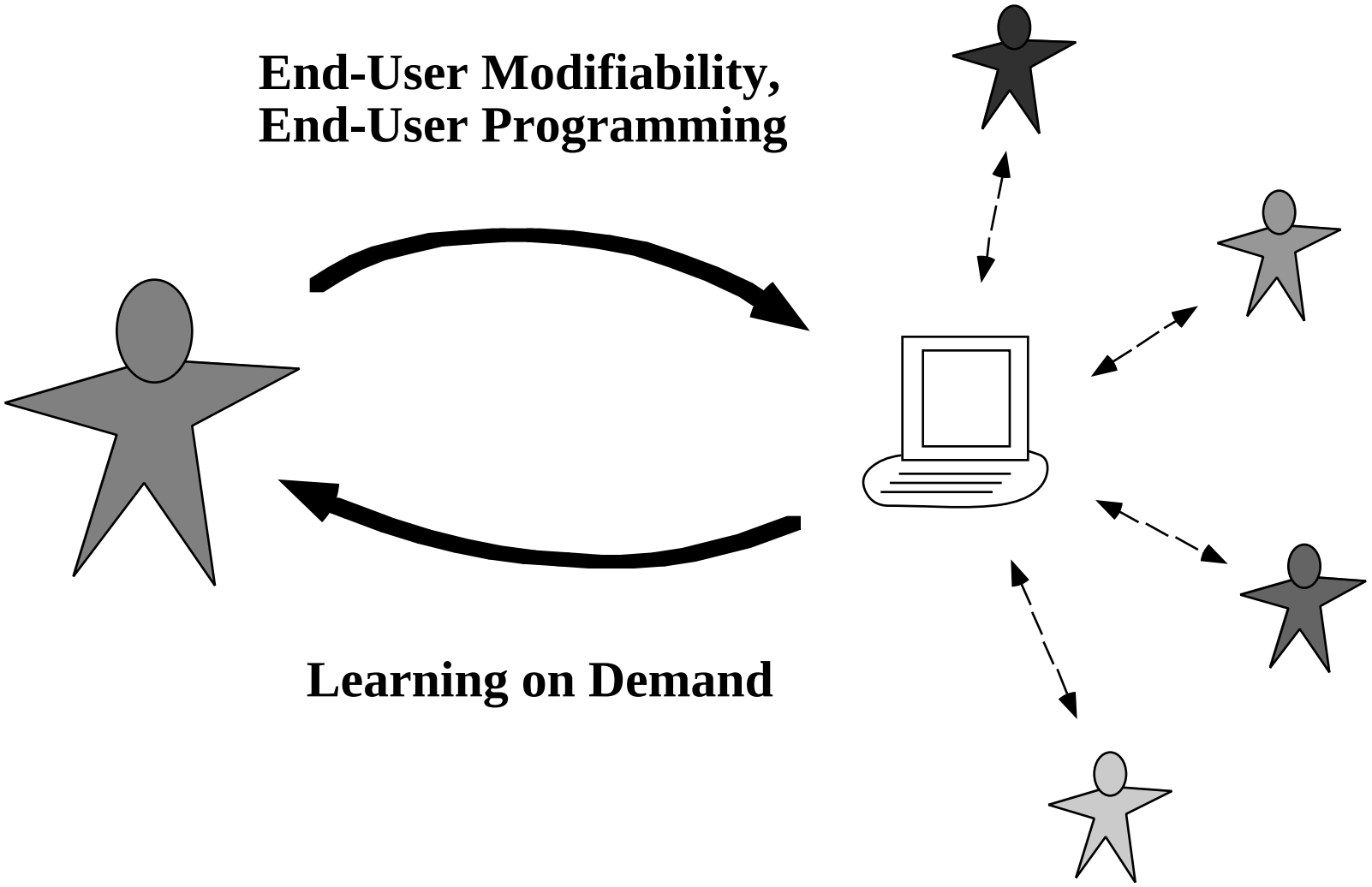
- **example: SimCity —** too much crime
  - solution supported: build more police stations (fight crime)
  - solution *not* supported: increase social services, improve education (prevent crime)

- **one of our goals:** create end-user modifiable versions of SimCity, because
  - background knowledge can never be completely articulated
  - the world changes

End-User Modifiability

- competent practitioners usually know more than they can say
- tacit knowledge is triggered by situations, by breakdowns
- impossibility of completely articulating background assumptions
- situations of practice: complex, unique, uncertain, conflicted, instable (“85% percent of the problems a doctor sees in his office are not in the book” (Schön)
- the initial moves must be reframed, as the changed situation most often deviates from the initial appreciation

Learning on Demand and End-User Modifiability



Assessment / Evaluation

- do critics enhance or hinder creativity (e.g., Fosbury Flop)? — Stravinsky: without constraints, there can be no creativity
- differences in performance, quality, and creativeness as a function of critics, catalog, simulation component?
- trade-offs between critiquing (breakdowns occur) versus constraints (breakdowns are prevented)
- trade-offs between different intervention strategies (active versus passive)
- does “making information relevant to the task at hand” prevent serendipity?
- under which conditions will designers challenge or extend the knowledge represented in the system?
- should the “back-talk” be embedded directly in the artifact or in a separate discourse?

Conclusions

- **Popper** (in “Conjectures and Refutations: The Growth of Scientific Knowledge”):

  “The way in which knowledge progresses, and especially our scientific knowledge, is by justified (and unjustifiable) anticipations, by guesses, by tentative solutions to our problems, by conjectures.

  These conjectures are controlled by criticism; that is, by attempted refutations, which include severely critical tests.

  Criticism of our conjectures is of decisive importance: by bringing out our mistakes it makes us understand the difficulties of the problem which we are trying to solve.”

- Popper focused on historical (universal) creativity ----> claims:
  - this process is of equal relevance for *psychological* (local) creativity
  - computational media can effectively support this process