



Center for
**LifeLong
Learning
& Design**

University of Colorado at Boulder

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

AI and Design

Gerhard Fischer

AI Course, Fall 1996, Lecture, Dec 2nd, 1996

The Sciences of the Artificial

Understanding the Natural and the Artificial World

- **natural science: “how things are”**
 - knowledge about natural objects and phenomena
 - primary interest: analysis
 - examples: physics, chemistry
- **sciences of the artificial: “how things might be (and ought to be in order to attain goals and to function)”**
 - knowledge about artificial objects and phenomena
 - primary interest: synthesis
 - artificial things are as they are only because of a system's being molded, by goals and purposes, to the environment in which it lives
 - examples: engineering, medicine, business, architecture, painting

Definition of “Artificial”

Definition of "artificial": human-made as opposed to natural

questions: where does mathematics / computer science / biology belong too?

claims by Simon:

- the world in which we live in today is much more a human-made, or artificial, world than it is a natural world
- a plowed field is no more part of nature than an asphalted street - and no less

Alan Kay (Scientific American, Sept 84, p 57)

“molecular biology has the advantage of studying a system already put together and working; for the composer of software the computer is like a bottle of atoms waiting to be shaped by an architecture he must invent and then impress from the outside”

Some Aspects of Artifacts

- artifacts are not apart from nature
- they are adapted to human's goals and purposes
- they are what they are in order to satisfy human's desires (e.g. to fly, to eat well, to take a hot shower,)
- as human's aims change, so too do his artifacts — and vice versa
- apply these characterizations to:
 - computer system?
 - user interface?

Science of Design

- **Definition:**
Everyone designs who devise courses of action aimed at changing existing situations into preferred ones. The intellectual activity that produces material artifacts is no different fundamentally from the one that prescribes remedies for a sick patient or the one that devises a new sales plan for a company or a social welfare policy for a state (Simon, “Sciences of the Artificial”, p 130)
- **examples:** architects, doctors, managers, politicians, teachers,
- **generic design — does it exist?**
 - design as an activity has a distinct conceptual and cognitive realization from nondesign activities
 - it can be abstracted away from the particulars of the knowledge base of a specific task or discipline and studied in its own right

Design Deals with Wicked or Ill-Defined Problems

Rittel in Cross "Developments in Design Methodology"

- there is no definitive formulation of a wicked problem. For any given tame problem, an exhaustive formulation can be stated containing all the information the problem-solver needs for understanding and solving the problem
- they have no stopping rule. In tame problems, problem solvers know when they have done the job. Problem solvers terminate work on a wicked problem, not for reasons inherent in the 'logic' of the problem
- solutions to wicked problems are not "true-or-false", but "good-or-bad"
- there is no immediate or ultimate test of a solution to a wicked problem
- every solution to a wicked problem is a 'one-shot operation'; because there is no opportunity to learn by trial-and-error, every attempt counts significantly

Wicked or Ill-Defined Problems — Continued

- wicked problems do not have an enumerable (or an exhaustively describable) set of potential solutions, nor is there a well-described set of permissible operations that may be incorporated into the plan
- every wicked problem is essentially unique
- every wicked problem can be considered to be a symptom of another problem
- the existence of a discrepancy representing a wicked problem can be explained in numerous ways. The choice of explanation determines the nature of the problem's resolution.
- the aim is not to find the truth, but to improve some characteristics of the world where people live

Complexity of Designs

from R. Dawkins: "The Blind Watchmaker"

- biology is the study of complicated things that give the appearance of having been designed for a purpose
- physics is the study of simple things that do not tempt us to invoke design
- treat complex human-made artifacts (e.g., computers, airliners, cars, books) as biological objects
- the behavior of physical, nonbiological objects is so simple that it is feasible to use existing mathematical language to describe it
- a complex thing is something whose constituent parts are arranged in a way that it is unlikely to have arisen by chance alone

Simulation

- artificiality connotes perceptual similarity but essential differences, resemblances from without rather than within
- the computer (based on its abstract character and its symbol-manipulating generality) has greatly extended the range of systems whose behavior can be imitated
- question: how can a simulation ever tell us anything that we do not already know?
- hypotheses:
 - a simulation is no better than the assumptions built into it
 - a computer can do only what it is programmed to do

Simulation as a Source for New Knowledge

- even when we have correct premises, it may be difficult to discover what they imply (e.g.: weather prediction; predict how an assemblage of components will behave)
- **simulation of poorly understood systems**
 - we are seldom interested in explaining or predicting phenomena in all their particularity
 - we are interested only in a few properties abstracted from complex reality (e.g.: neurophysiological level versus information processing level in understanding human cognition)”
- resemblance in behavior of systems without identity of the inner systems ---> feasible for: emphasis on organization of parts, not in the properties of the individual components

Artificial Intelligence and Human Intelligence

A Design Perspective

- the organization of components, and not their physical properties, largely determines behavior
- computers are organized somewhat in the image of humans
- computer becomes an obvious device for exploring the consequences of alternative organizational assumptions for human behavior
- psychology can move forward without awaiting the solutions by neurology of the problems of component design -- however interesting and significant these components turn out to be

Computers as Empirical Objects

- **claim:** the highly abstractive quality of computers makes it easy to introduce mathematics into the study of their theory — and has led some to the erroneous conclusion that, as a computer science emerges, it will necessarily be a mathematical rather than an empirical science.
- see Turing Award Lecture from Newell and Simon, “Computer Science as an Empirical Inquiry: Symbols and Search”, CACM, Vol 19, No 3, 1976, pp 113-136
 - example: time-sharing systems ---> main route to develop and improve them was: build them and see how they behave
 - perhaps theory could have anticipated these experiments and made them unnecessary ---> in fact: it did not

Design Perspectives Relevant to AI

- **satisficing versus optimizing**
 - optimal decision for an imaginary simplified world (operations research methods)
 - decisions that are “good enough” ---> "satisficing" solutions (heuristic search)
- **heuristic search** makes much weaker demands on the problem structure than mathematical tools of **operations research**
 - can cope with non-quantifiable variables
 - is applicable to non-numerical as well as numerical information

“The best is often the enemy of the good”

Finding Satisfactory Actions

- “No one will satisfice if (s)he can equally well optimize”
- examples of combinatorial problems
 - traveling salesman problem
 - warehouse location problem
 - location of central power stations
- characteristics of these situations:
 - the set of available alternatives is "given" in a certain abstract sense ---> i.e. we can define a generator guaranteed to generate all of them eventually
 - they are **not** “given” in the sense that it is practically relevant
- within practical computational limits we cannot generate all the admissible alternatives
- we cannot recognize the best alternative , even if we are fortunate enough to generate it early

Finding Satisfactory Actions

- **example of alternatives “given” versus “not given”:**
 - going from Los Angeles to Boston (by foot, by bike, by train, by airplane)
 - subjective computability versus objective computability
 - informational and computational efficiency
- **informational efficiency:**

two representations are informationally equivalent if all of the information in the one is also inferable from the other, and vice versa. Each could be constructed from the information in the other.
- **computational efficiency:**

two representations are computationally equivalent if they are informationally equivalent and, in addition, any inference that can be drawn easily and quickly from the information given explicitly in the one can also be drawn easily and quickly from the information given explicitly in the other, and vice versa
- **examples:**
 - Number Scrabble versus Tic Tac Toe
 - the mutilated chess board versus the Match-Making story
 - Roman Numerals versus Arabic Numerals

The Evolutionary Model

- things evolve in response to some kind of selective force
- simple scheme of evolution:
 - generate — produce variety (e.g. genetic mutation)
 - test — to evaluate the newly generated forms (e.g. natural selection)
- problems with evolution:
 - is myopic
 - reaches local maxima (instead of global ones)
 - moving away from a local maxima implies: going across a valley

Evolution and Design

- **Design different from Evolution:**
 - guided
 - there is a goal (question: can we design without a final goal in mind?)
 - one can look back over a design and “clean it up”
 - one can examine failures and see what went wrong
 - faster than evolution (guidance, remembering previous successes and failures)
- **Is Design the same as Evolution?**
 - "installed base" problem (“qwerty” typewriter, English measurement system, Fortran/Cobol,)
 - standards
 - knowledge is cumulative

Evolution and Knowledge-Based Design Support Systems

- why do complex systems evolve (do they?)
- why must knowledge-based systems be able to evolve? (changing requirements — different from algorithms)
- how do they evolve? — end-user modifiability, reuse, redesign, object-oriented architectures...
- powerful reasoning and search methods <----> large amounts of special case knowledge:
“the fundamental problem of understanding intelligence is not the identification of a few powerful techniques, but rather the question of how to represent large amounts of knowledge in a fashion that permits their effective use and interaction (Goldstein and Papert 1977)”
- expert knowledge is more bounded than common-sense knowledge

The Shape of the Design: Hierarchy

The Problem of Modularity

- **claim:** to design a complex structure, one powerful technique is to discover viable ways of decomposing it into semi-independent components corresponding to its many functional parts. The design of each component can then be carried out with some degree of independence of the design of others, since each will affect the others largely through its function and independently of the details of the mechanisms that accomplish the function.
- **examples:**
 - functional programming
 - object-oriented programming
 - rule-based systems
 - nearly decomposable systems

Integrating Problem Framing and Problem Solving

Simon:

in oil painting every new spot of pigment laid on the canvas creates some kind of pattern that provides a continuing source of new ideas to the painter. The painting process is a process of cyclical interaction between the painter and canvas in which current goals lead to new applications of paint, while the gradually changing pattern suggests new goals.

Computer Science Technology Board:

system requirements are not so much analytically specified as they are collaboratively evolved through an iterative process of consultation between end-users and software developers

Rittel:

one cannot understand a problem without having a concept of the solution in mind

one cannot gather information meaningfully unless one has understood the problem but one cannot understand the problem without information about it

Examples for Large-Scale Design

going to the moon: a “complex” problem along one dimension; sources for success:

- exceedingly cooperative environment
- employing a single new organization
- single, highly operational goal

the American Constitution:

“the founding fathers did not postulate a new man to be produced by new institutions but accepted as one of their design constraints the psychological characteristics of men and women as they knew them, their selfishness as well as their common sense” (Simon)

Topics for Large-Scale Design

- problem representation — e.g., in many cases: what is needed was not so much a “correct” conceptualization as one that could be understood by all the participants and that would facilitate action rather than paralyze it
- how to cope with inadequacies in our data
- who is the client?
- limits on the planner's time and attention
- ambiguity and conflict of goals in societal planning

Who is the Client?

- designers = dual role of the artist and professional
- in city planning, in the design of software systems (if we regard them as embedded systems) ---> the boundary between the design of physical structures and the design of social systems dissolves almost completely
- balance cost against quality (e.g., in computer systems, in medical care)
- the members of an organization or a society for whom plans are made are not passive instruments, but are themselves designers who are seeking to use the system to further their own goals
- example: Denver Public Library

Ownership of Problems — Example: Designing the Denver Public Library

- 1991: final design competition
- 1995: construction finished
- requirement: 15 years without major modifications
- question: a library in the year 2010? — books, CD ROM, ---> we are not only designing for a given context: we construct the context
- client: city of Denver ---> committee (old librarian, techi, ...)
- who owns the problem??
 - client(s), designers, customers, specialist,
 - architect, structural engineer, contractor, bricklayer,

Three Generations of Design Methods from the History of Architectural Design

- **1st Generation (before 1970):**
 - directionality and causality
 - separation of analysis from synthesis
 - major drawback: perceived by the designers as being unnatural; does not correspond to actual design practice
- **2nd Generation in the early 70'es:**
 - participation — expertise in design is distributed among all participants
 - argumentation — various positions on each issue
 - major drawback: insisting on total participation, neglecting expertise possessed by a well-informed and skilled designer
- **3rd Generation (in the late 70'es):**
 - inspired by Popper: the role of the designer is to make expert design conjectures
 - these conjectures must be open to refutation and rejection by the people for whom they are made (---> end-user modifiability)