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

Design and Domain-Oriented Design Environments

Gerhard Fischer, Hal Eden, and Holger Dick — Fall Semester 2010
gerhard@colorado.edu; haleden@colorado.edu; holger.dick@gmail.com;
September 20, 2010

Overview

- Design

- Domain-Oriented Design Environments (DODEs)

- Examples
  - video-tape of Janus: a DODE for kitchen design

- Critiquing in DODEs
  - reflection-in-action
  - intrusiveness
  - generic, specific, interpretive critics
Design

- **a fundamental and important objective:**
  - Karl Marx: “Die Philosophen haben die Welt nur verschieden interpretiert; es kommt aber darauf an, sie zu verändern!” (Entrance Hall of the Humboldt University in Berlin)
  - in English: “Philosophers interpret the world in various ways; what matters is to change it!”

- **design as a concept / an activity**
  - poorly understood
  - used differently by different people and different communities

- **Herbert Simon’s “Sciences of the Artificial”**
  - natural sciences: how things are → understand what is
  - design: how things ought to be → shape what will be
    - design = “everyone designs who devises courses of action aimed at changing existing situations into preferred ones”
Different Discourse

- **natural science:**
  - understanding the physical world: natural scientists are given a universe and seek to discover its laws
  - in the natural sciences there are right and wrong answers
  - studies tame problems

- **design:**
  - computer scientists create worlds in the form of programs and the computer brings these worlds to life
  - studies ill-defined, wicked problems
    - integration of problem framing and problem solving
    - problems have no stopping rule
  - trade-offs (e.g.: simplicity \( \leftrightarrow \) complexity) \( \rightarrow \) problems have no single answer (argumentation support is required)
  - there are no best, optimal solutions \( \rightarrow \) satisficing
Design Domains

- software design and software engineering
- architecture and urban design
- design in the creative arts
- design of learning environments
- design of collaboration environments
SchemePaint (Michael Eisenberg): a programmable application combining direct manipulation with interactive programming
Example: Crafts Technology — Hypergami

(Michael Eisenberg)
Example: Google SketchUp & 3D Warehouse (2009 Inaugural Stage)
(guest lecture: John Bacus)
Example: Design Kits
(general purpose ↔ special purpose)
Example: Architecture
Example: Architecture
Software Design / Engineering: Upstream and Downstream Activities

- **upstream: world → model / specification**
  - ill-defined problem
  - integration of problem framing and problem solving
  - collaboration and communication between different stakeholders
  - failure leads to *design disasters* (wrong problem is solved)

- **downstream: model / specification → implementation / system**
  - well-defined problem
  - dealing with difficult technical problems
  - creating reliable code
  - failure leads to *implementation disasters* (wrong solution to the right problem)
Design: Beyond Binary Choices

- **Turing Tar Pit:** “Beware of the Turing Tar Pit, in which everything is possible, but nothing of interest is easy.”
  - level of representation is too far removed from the conceptual world of the domain workers
  - claim: they emphasize objective computability → the challenge: subjective computability

- **The Inverse of the Turing Tar Pit:** “Beware of the over-specialized systems, where operations are easy, but little of interest is possible.”
  - domain-specific tools (such as SimCity) provide extensive support for certain problem contexts
  - the ability to extend these environments is limited — even minor incremental changes are often impossible in these systems
Domain-Oriented Design Environments

- **rationale:**
  - recognize the legitimacy of **specialization to a domain** — do not serve all needs obscurely, serve a few needs well
  - support Human Problem-Domain Interaction

![Diagram of domain-oriented design environments](image)
Human Problem Domain Interaction — Pinball Construction Kit
Human Problem Domain Interaction — **Music Construction Kit**
Examples of Domain-Oriented Design Environments

- kitchen design
- voice dialog design
- computer network design
- urban design and transportation planning — Envision and Discovery Collaboratory (EDC)
- multi-media design (color)
- website design
Domain-Oriented Design Environments (Janus-Construction)

**Janus-Construction**

- Appliance Palette
  - walls
  - doors
  - windows
  - sinks
  - stoves

- Catalog
  - L-Shaped-Kitchen

- Work Area
  - DW

**Messages**

- The length of the work triangle (Double-Bowl-Sink-1, Four-Element-Stove-1, Single-Door-Refrigerator-1) is greater than 23 feet.
- Single-Door-Refrigerator-1 is not near Four-Element-Stove-1.

**Commands**

- Critique All

---

Fischer & Eden & Dick

HCC Course, Fall 2010
Janus-Argumentation

Answer (Refrigerator, Sink, Stove)
The distance between sink, stove and refrigerator, the work triangle, should be less than 23 feet.

\[ d_1 + d_2 + d_3 < 23 \text{ feet} \]

Figure 10: the work triangle

Argument (Walking Distance)
The work triangle is an important concept in kitchen design. The work triangle describes the center front distance between the three main appliances: sink, oven and refrigerator. This length should be less than 23 feet to avoid unnecessary walking and to ensure an efficient work flow in the kitchen.

Argument (Small Room)
In small kitchens where the work triangle is less than 16 feet.
The Multi-Faceted Architecture behind DODEs
Domain-Oriented Design Environments (DODEs)

construction

specification
Is the cook right-or left-handed?

critics

design rationale
issue:
answer:
argument:
argument:
answer:
argument

perspectives
resale personal
electrical plumbing
American Japanese

catalog
Reflection-in-Action as a Problem Solving Theory

**designer's understanding** \rightarrow **situated action**

**reflection on knowledge**

**DESIGN**

**breakdown**

**Movie: The Janus DODE**
VDDE: Voice Dialog Design Environment
Computer Network Design

(1) NetDE

(2) Priorities to be used for devices in this area

1st priority
- Cost
  - weight: 10

2nd priority
- Expandability
  - weight: 0

3rd priority
- Reliability
  - weight: 4

(3) Publications OT 8-6, College of Engineering, University of Colorado

(4) Group Memory

- Meeting Notes
- Priorities
- Machinery
- Miscellaneous
- All email

(5) Design
Computational Critics (= “Virtual Human Critics”)

- **spelling correctors** — example of a “simple” critiquing system
  - simple: a “correct” answer exists
  - passive $\leftrightarrow$ active
  - suggestions for corrections $\leftrightarrow$ “auto-correct” in MS-Word

- **unlimited opportunities for application**: grammar checkers, color critics, graphs critics, webpage critics

- **webpage critics and universal access**
  - [http://analyze.websiteoptimization.com/](http://analyze.websiteoptimization.com/)
  - [http://validator.w3.org/](http://validator.w3.org/)
The Rationale / Need for Critiquing

  - “but when color is used inappropriately it can be very counter productive and few software designers have much experience with the use of color; the aim of this book is to synthesize our current knowledge in the area and specify **guidelines** so that programmers, engineers, and psychologist can use color.”
  - question: what are the benefits of “critiquing systems” compared to “guidelines”

  - “one reason for the abundance of bad graphs is the proliferation of low-cost microcomputers and ‘business graphics’ packages which often seduce the user into producing flashy but muddled displays; many graphs are designed without consideration of principles of human perception and cognition”
  - question: can a critiquing system be developed for “human perception and cognition”
EMMA (Environment for MultiMedia Authoring) and Color Critiquing
Computer-Based Critiquing: Examples and Mechanisms

- examples:
  - the length of the work triangle is more than 23 feet
  - a critiquing rule in the Envisionment and Discovery Collaboratory: “the maximum distance between two bus stops is 1 mile”

- mechanism:
  - enable relevant critics
  - analyze construction and specification (e.g., the specification states that this is a part of town where many old people live)
  - signal breakdowns
  - deliver relevant knowledge
  - identify the right level of intrusiveness: on demand ↔ critical points (“windows in Janus”) ↔ all the time (MS Word)
Giving Domain Designers Control about the Intrusiveness of Critics

![Image of UDDE-Stack interface](image)

- **Select Perspective:**
  - Select the standard(s) you want to use when analyzing your design. This allows you to view your design from multiple perspectives. Click on the button to the left to enable the rule set. Click on Argumentation to list all the rules which belong to the selected standard. You can disable and enable individual critic rules within each standard from this overview. To change priority, disable the rule set, then enable it again.

- **Priorities:**
  - **1** USWEST Rule Set
  - **0** UMUIF Rule Set
  - **0** International Rule Set
  - **0** Consistent with:

- **Type of Application:** General
  - **Conceptual Unit Critiquing**

- **Critiquing Thermometer:**
  - Active: 3, 6, 9, 12, 15
  - Passive
An Implementation of Critics

- Specification Component
- Construction Component
- Construction Analyzer
- Argumentation Component
- Argumentation Illustrator
- Catalog Component

- critic messages
- design rationale
- catalog examples
Embedding Critics in the Contexts of Design

**generic domain knowledge**
- "kitchen design"
- design rationale
- catalog of past designs

**construction**
- "this design"
- graphical construction

**specification**
- "left-handed kitchen"
- partial specification

**perspective**
- "the resale perspective"
- redefined knowledge

**generic critics**

**specific critics**

**interpretive critics**
**Generic Critics in Construction**

**Construction**

**Design Rationale**

**issue:**
Where should the dishwasher be placed?

**answer:**
Left side of sink.

**argument:**
Dishwasher on left provides efficient work flow for right-handed people.

**Generic Critic**

IF the dishwasher is right of sink, THEN "move dishwasher left of sink"
A Partial Specification of a Specific Client

<table>
<thead>
<tr>
<th>questions in specification component</th>
<th>answers by client:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- name:</td>
<td>Smith’s kitchen</td>
</tr>
<tr>
<td>- size of family:</td>
<td>four to six</td>
</tr>
<tr>
<td>- primary cook:</td>
<td>left-handed</td>
</tr>
<tr>
<td>- size of meals:</td>
<td>huge (big eaters)</td>
</tr>
<tr>
<td>- entertainment:</td>
<td>often</td>
</tr>
<tr>
<td>- cooking frequency:</td>
<td>often</td>
</tr>
<tr>
<td>- type of sink:</td>
<td>double bowl sink</td>
</tr>
</tbody>
</table>

**specification component in EDC:** questionnaire for citizens how long they would wait for the bus
Global Objective of Embedding Critics

- increase the “back-talk” of the situation
- support reflection-in-action
- support learning on demand
- reduce information overload: saying the ‘right’ thing at the ‘right’ time in the ‘right’ way to the ‘right’ person
- make information relevant to the task at hand