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

Meta-Design:  
A Framework for the Future of End-User Development (EUD)

Gerhard Fischer, Hal Eden, and Holger Dick — Fall Semester 2010  
gerhard@colorado.edu; haleden@colorado.edu; holger.dick@gmail.com;  
September 27, 2010

Topics

- meta-design
- end-users and end-users development
- design time and use time

examples:
- Macros (in Word)
- CLever/MAPS
- Envisionment and Discovery Collaboratory

- boundaries, control, motivation

- meta-design: democratizing design in many application domains
Meta-Design

- **meta-design** = creating new media and new technologies that allow users to act as **designers and be creative** (rather than being confined to consumers)

- **meta-design emphasizes**
  - the creation of context rather than content
  - puts the tools rather than the object of design in your hands
  - does not define a product, but the conditions for a process of interaction
Meta-Design: Exploring Middle Ground

Turing Tar Pit:
Saw + Wood

Inverse of the Turing Tar Pit:
Plastic Car

Construction Kits
Meta-Design and Technical Construction Kits: An Airplane
Meta-Design and Technical Construction Kits: A Tipper Truck
Why Meta-Design?

- design for **diversity** (for “a universe of one” → CLever Project)

- design as a process is **tightly coupled to use** and continues during the use of the system

- addresses and overcome problems of **closed systems**

- prerequisite for **social creativity** and innovation

- transcends a “**consumer mindset**”
Meta-Design — How We Think About It

- “if you give a fish to a human, you will feed him for a day — if you give someone a fishing rod, you will feed him for life” (Chinese Proverb)

- meta-design extends this to:

  “if we can provide the knowledge, the know-how, and the tools for making fishing rods, we can feed the whole community”
Computer Scientists and End-Users

- **computer scientist / programmers**
  - find computers intrinsically interesting
  - like computers because they get to program

- **end-users:**
  - are the owners of problems, have the domain knowledge, are competent practitioners and like computers because they get their work done
  - regard computers as useful machines capable of helping them work more productively, creatively, and with greater pleasure
  - end-user programmers are motivated by their domain and not by the merits of producing high-quality, dependable code
  - enormous diversity between end-users

- **ultimate goal / belief:** end-users will use, tailor, extend and create their own computational artifacts if they have a supportive socio-technical environment

- **communities of users will develop:** power users, local developers, gardeners
Number of “Programmers” (in the USA)

- **3 million** professional programmers

- **12 million** people in workplaces that “do programming” at work

- **50 million** use spreadsheets and databases → they may potentially program doing
  - spreadsheets systems
  - web authoring tools
  - business process authoring tools
  - graphical languages

**source:**
What Do Meta-Designers Do?

- use their own creativity to create socio-technical environments in which other people can be creative

- create **technical** and **social** conditions for broad participation in design activities which are as important as creating the artifact itself
## Concepts of Meta-Design

<table>
<thead>
<tr>
<th>Concept</th>
<th>Implications</th>
</tr>
</thead>
<tbody>
<tr>
<td>convivial tools</td>
<td>allow users to invest the world with their meaning and to use tools for the accomplishment of a purpose they have chosen</td>
</tr>
<tr>
<td>domain-orientation</td>
<td>bring task to the forefront; provide time on task</td>
</tr>
<tr>
<td>open, evolvable systems</td>
<td>put owners of problems in charge; in open systems, extension is an essential part of use</td>
</tr>
<tr>
<td>underdesigned systems</td>
<td>create seeds and constructs for design elaboration at use time</td>
</tr>
<tr>
<td>collaborative work practices</td>
<td>support design communities and the emergence of power users</td>
</tr>
</tbody>
</table>
Consumer and Designers — A Continuum
Design Time and Use Time

Key:
- System developer
- User (representative)
- End user

Design Time:
- World-as-imagined
- Prediction
- Planning

Use Time:
- World-as-experienced
- Reality
- Situated action
Computational Media
—
Extending Design Opportunities at Use Time

- **print media:** a fixed context for use time is decided at design time

- **computational media:**
  - presentations at use time can take advantage of contextual factors only known at use time (about tasks, users, social systems,.....)
  - examples: specification sheets and usage data, supporting dynamic forms, dynamic websites, user and task specific maps and traffic schedules....

- **evolving existing systems:** users (acting as designers) can transcend at use time the boundaries of the systems as developed at design time
Putting Owners of Problems in Charge: a Necessity not a Luxury —

An Interview with a Geoscientist at CU Boulder

- I spend in average an hour every day developing software for myself to analyze the data I collected because there is not any available software. → “reality is not user-friendly” and problems are unique

- Even if there is a software developer sitting next to me, it would not be of much help because my needs vary as my research progresses and I cannot clearly explain what I want to do at any moment. → ill-defined problems cannot be delegated; they require “unselfconscious cultures of design”

- Even if the software developer can mange to write a program for me, I will not know if he or she has done it right without looking at the code. → back-talk of the artifact under construction has to go back to the owner of the problem
Interview (continued)

- So I spent three months to gain enough programming knowledge to get by. Software development has now become an essential task of my research, but I do not consider myself a software developer and I don’t know many other things about software development.

  → this geoscientist obviously is not just an end-user (or a “none-professional”); his software has thousands of lines and he has considerable programming skills

  → it is equally obvious that he is not a software professional and does not intend to become one

  → the number of end users creating software is far larger than the number of professional programmers.
Meta-Design Concepts (in Microsoft Word) —

Users as Co-Developers

- **tailor** and **customize** the system by setting different parameters as their personal preferences

- **extend** and **evolve** existing information structures (e.g., menus, spelling dictionaries, auto-correct tables, …)

- write **macros** to create new operations (an example of “programming by example” or “programming by demonstration”)

- create **programs in VisualBasic** to extend the functionality of the system

- **share** the user-defined extensions
A Macro for Unwrapping Text

Is End-User Modifiability the Answer to HFAs? The message of this section about HFA is that they contain too much unused functionality — at least in the abstract. But on the other hand: there is often not enough functionality for specific problems. As argued at other places in this book, closed systems are inadequate to capture the unanticipated needs of users in the real world. No matter how much designers at design time try to anticipate the needs of users at use time (see Figure <design/usetime>), the effort will fall.
Meta-Design in the Cognitive Lever (CLever) Project
MAPS Script Editor
Use of MAPS in Mobility-for-All

This is your Bus, Get ready to get on

Script Use Time
Meta-Design Aspects in the Envisionment and Discovery Collaboratory

Closed versus Open Systems

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

- **important goal of EDC:** create end-user modifiable versions of SimCity, because:
  - background knowledge can never be completely articulated
  - the world changes
Web 2.0: A Focus on Meta-Design


<table>
<thead>
<tr>
<th>Web 1.0</th>
<th>Web 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Britannica Online</td>
<td>Wikipedia</td>
</tr>
<tr>
<td>personal website</td>
<td>blogging</td>
</tr>
<tr>
<td>publishing</td>
<td>participation</td>
</tr>
<tr>
<td>content management systems</td>
<td>wikis</td>
</tr>
<tr>
<td>scheduled software releases</td>
<td>continuous improvements</td>
</tr>
<tr>
<td>individual contributions</td>
<td>collective intelligence</td>
</tr>
</tbody>
</table>

- **claim:** network effects from user contributions (= knowledge sharing) are the key to market dominance in the Web 2.0 era
### Examples of Web 2.0 Environments Supported by Meta-Design

<table>
<thead>
<tr>
<th>Site</th>
<th>Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>iTunes U</td>
<td>distribute digital lessons by faculty members from “certified institutions”</td>
</tr>
<tr>
<td>YouTube</td>
<td>video sharing website</td>
</tr>
<tr>
<td>Encyclopedia of Life (EoL)</td>
<td>document 1.8 million living species known to science</td>
</tr>
<tr>
<td>Google-Maps</td>
<td>maps and task-related directions</td>
</tr>
<tr>
<td>Google-SketchUp and 3D Warehouse</td>
<td>model the whole world in 3D</td>
</tr>
<tr>
<td>Scratch</td>
<td>programming environment and support for sharing creations</td>
</tr>
<tr>
<td>PatientsLikeMe</td>
<td>enables patients who suffer from life-changing diseases to converse</td>
</tr>
<tr>
<td>Crisis Informatics</td>
<td>explores the synthesis between broadcast news and information provided by directly involved people</td>
</tr>
<tr>
<td>Stepgreen</td>
<td>library of energy saving actions</td>
</tr>
<tr>
<td>OpenEI</td>
<td>open-data source (Wiki) for energy information</td>
</tr>
<tr>
<td>CreativeIT Wiki</td>
<td>support researchers interested in creativity</td>
</tr>
</tbody>
</table>
**Cost-Scope Trade-offs in End-User Development (EUD) Tools**

<table>
<thead>
<tr>
<th>Cost of Learning / Scope</th>
<th>High</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>Java / C++</td>
<td>EUD Ideal?</td>
</tr>
<tr>
<td>Low</td>
<td>Hardware design</td>
<td>Domain-oriented design environments</td>
</tr>
</tbody>
</table>
Meta-Design: Expanding Boundaries

- power of the few → wisdom of the crowds

- socio-technical environments are living entities

- breaks down the sharp distinction between designers and users: users become co-designers

- democratizes design and innovation: meta-design eliminates the constraint that users are restricted to what is given to them

- revolutionizes the creation of systems: creates foundation for social production and mass collaboration
Meta-Design: Redistributing Control

- developers and user-designers: sharing control
  - benign dictatorship
  - council control
  - complete decentralized

- control is desired only for personally meaningful problems

- the pitfalls associated with a “do-it-yourself” society
Motivational Aspects and Meta-Design

- **what will make humans want to become designers/active contributors over time?**
  - serious working and learning does not have to be unpleasant but can be personally meaningful, empowering, engaging, and fun
  - comment by an artist: “programming is not hard, but it is boring”

- **what will make humans want to share?** → requires: mindset change, culture change, community knowledge bases, gift cultures, social capital

- **who is the beneficiary and who has to do the work?** → organizational rewards
Utility = Value / Effort

- **increase in value**: motivation and rewards for a “design culture”
  - feeling in control (i.e., independent from “high-tech scribes”)
  - being able to solve or contribute to the solution of a problem
  - mastering a tool in greater depth
  - making an ego-satisfying contribution to a group
  - enjoying the feeling of good citizenship to a community (“social capital”)

- **decrease in effort**:
  - meta-design is hard
  - extending meta-design to design for design communities
Consumer and Designers — Beyond Binary Choices

- **claims:**
  - there is nothing wrong being a consumer (watching a tennis match, listening to a concert, ...)
  - the same person wants to be a consumer in some situations and in others a designer
  - consumer / designer is not an attribute of a person, but of a context
    \[ \text{consumer / designer} \neq f\{\text{person}\} \rightarrow f\{\text{context}\} \]

- **problems:**
  - someone wants to be a designer but is forced to be a consumer \( \rightarrow \text{personally meaningful activities} \)
  - someone wants to be a consumer but is forced to be a designer \( \rightarrow \text{personally irrelevant activities} \)
Meta-Design: Democratizing Design in Many Application Domains


Democratizing Design in Many Application Domains — Continued


Conclusions

- **meta-design offers:**
  - to invent and design a culture in which all participants in collaborative design processes can express themselves and engage in personally meaningful activities

- **meta-design requires**
  - a new *mindset* of all participants
  - designers giving up some *control* at design time
  - *active contributors* and not just passive consumers at use time

- **meta-design raises many issues and research problems of fundamental importance** including
  - new design methodologies
  - a new understanding of cognition, collaboration, and motivation
  - the design of new media and new technologies