User-Centered Design

Workshop on User-Centered Design of Language Archives

Christina Wasson

20-21 February 2016

This material is based upon work supported by the National Science Foundation under Grants No. BCS-1543763 and BCS-1543828.

What is user-centered design?

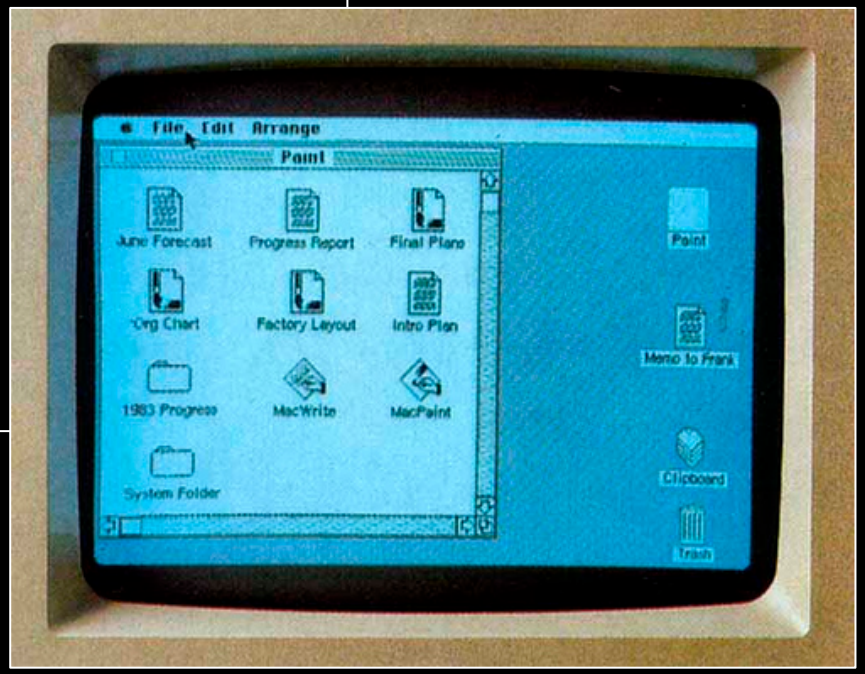
The chief difference from other product design philosophies is that UCD tries to optimize the product around how users can, want, or need to use the product, rather than forcing the users to change their behavior to accommodate the product.

-- Wikipedia

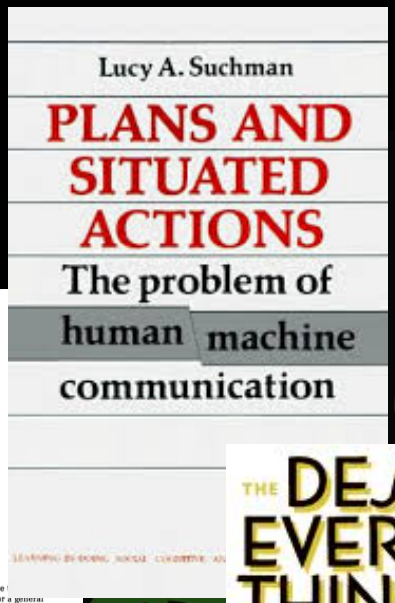
History of computer users

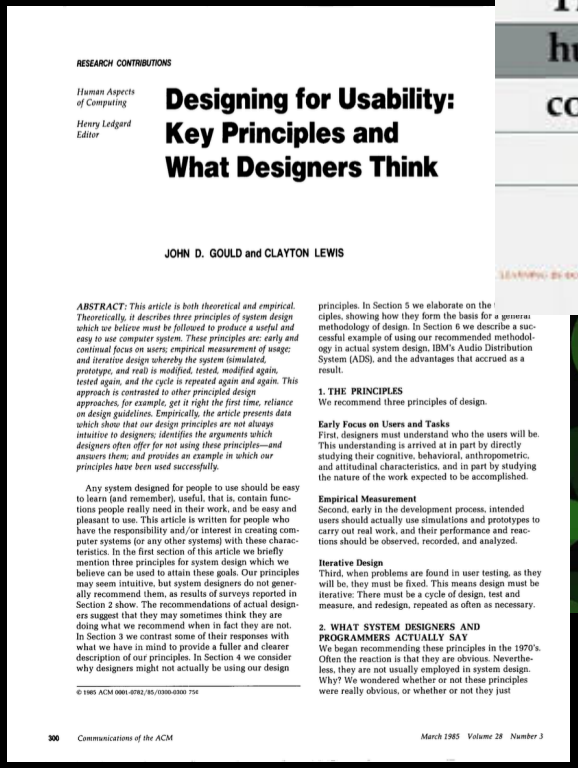
From DOS to graphical user interface (GUI)

```
Welcome to FreeDOS


CuteMouse v1.9.1 alpha 1 [FreeDOS]
Installed at PS/2 port
C:\>ver

FreeCom version 0.82 pl 3 XMS_Swap [Dec 10 2003 06:49:21]

C:\>dir
 Volume in drive C is FREEDOS_C95
 Volume Serial Number is 0E4F-19EB
 Directory of C:\

FDOS                <DIR>  08-26-04  6:23p
AUTOEXEC BAT          435  08-26-04  6:24p
BOOTSECT BIN          512  08-26-04  6:23p
COMMAND  COM       93,963  08-26-04  6:24p
CONFIG   SYS          801  08-26-04  6:24p
FDOSBOOT BIN          512  08-26-04  6:24p
KERNEL   SYS       45,815  04-17-04  9:19p
        6 file(s)        142,038 bytes
        1 dir(s)   1,064,517,632 bytes free

C:\>_
```



Paradigm shift to UCD





RESEARCH CONTRIBUTIONS

Human Aspects of Computing

Henry Ledgard Editor

Designing for Usability: Key Principles and What Designers Think

JOHN D. GOULD and CLAYTON LEWIS

ABSTRACT: This article is both theoretical and empirical. Theoretically, it describes three principles of system design which we believe must be followed to produce a useful and easy to use computer system. These principles are: early and continual focus on users; empirical measurement of usage; and iterative design whereby the system (simulated, prototype, and real) is modified, tested, modified again, tested again, and the cycle is repeated again and again. This approach is contrasted to other principled design approaches, for example, get it right the first time, reliance on design guidelines. Empirically, the article presents data which show that our design principles are not always intuitive to designers; identifies the arguments which designers often offer for not using these principles—and answers them; and provides an example in which our principles have been used successfully.

Any system designed for people to use should be easy to learn (and remember), useful, that is, contain functions people really need in their work, and be easy and pleasant to use. This article is written for people who have the responsibility and/or interest in creating computer systems (or any other systems) with these characteristics. In the first section of this article we briefly mention three principles for system design which we believe can be used to attain these goals. Our principles may seem intuitive, but system designers do not generally recommend them, as results of surveys reported in Section 2 show. The recommendations of actual designers suggest that they may sometimes think they are doing what we recommend when in fact they are not. In Section 3 we contrast some of their responses with what we have in mind to provide a fuller and clearer description of our principles. In Section 4 we consider why designers might not actually be using our design

principles. In Section 5 we elaborate on the
ciples, showing how they form the basis for a general
methodology of design. In Section 6 we describe a successful example of using our recommended methodology in actual system design, IBM's Audio Distribution System (ADS), and the advantages that accrued as a result.

1. THE PRINCIPLES

We recommend three principles of design.

Early Focus on Users and Tasks

First, designers must understand who the users will be. This understanding is arrived at in part by directly studying their cognitive, behavioral, anthropometric, and attitudinal characteristics, and in part by studying the nature of the work expected to be accomplished.

Empirical Measurement

Second, early in the development process, intended users should actually use simulations and prototypes to carry out real work, and their performance and reactions should be observed, recorded, and analyzed.

Iterative Design

Third, when problems are found in user testing, as they will be, they must be fixed. This means design must be iterative: There must be a cycle of design, test and measure, and redesign, repeated as often as necessary.

2. WHAT SYSTEM DESIGNERS AND PROGRAMMERS ACTUALLY SAY

We began recommending these principles in the 1970's. Often the reaction is that they are obvious. Nevertheless, they are not usually employed in system design. Why? We wondered whether or not these principles were really obvious, or whether or not they just

Cog psych and human factors

1 person +
1 machine

Anthropology

Social groups

Cultural practices and meanings

4 basic components of UCD

1. Identify user groups
2. Work with users to identify needs
3. Interdisciplinary and collaborative process
4. Iterative

1. Identify user groups

- What groups of people are currently using the technology?
- What other groups might benefit from using it?

Questions for language archives

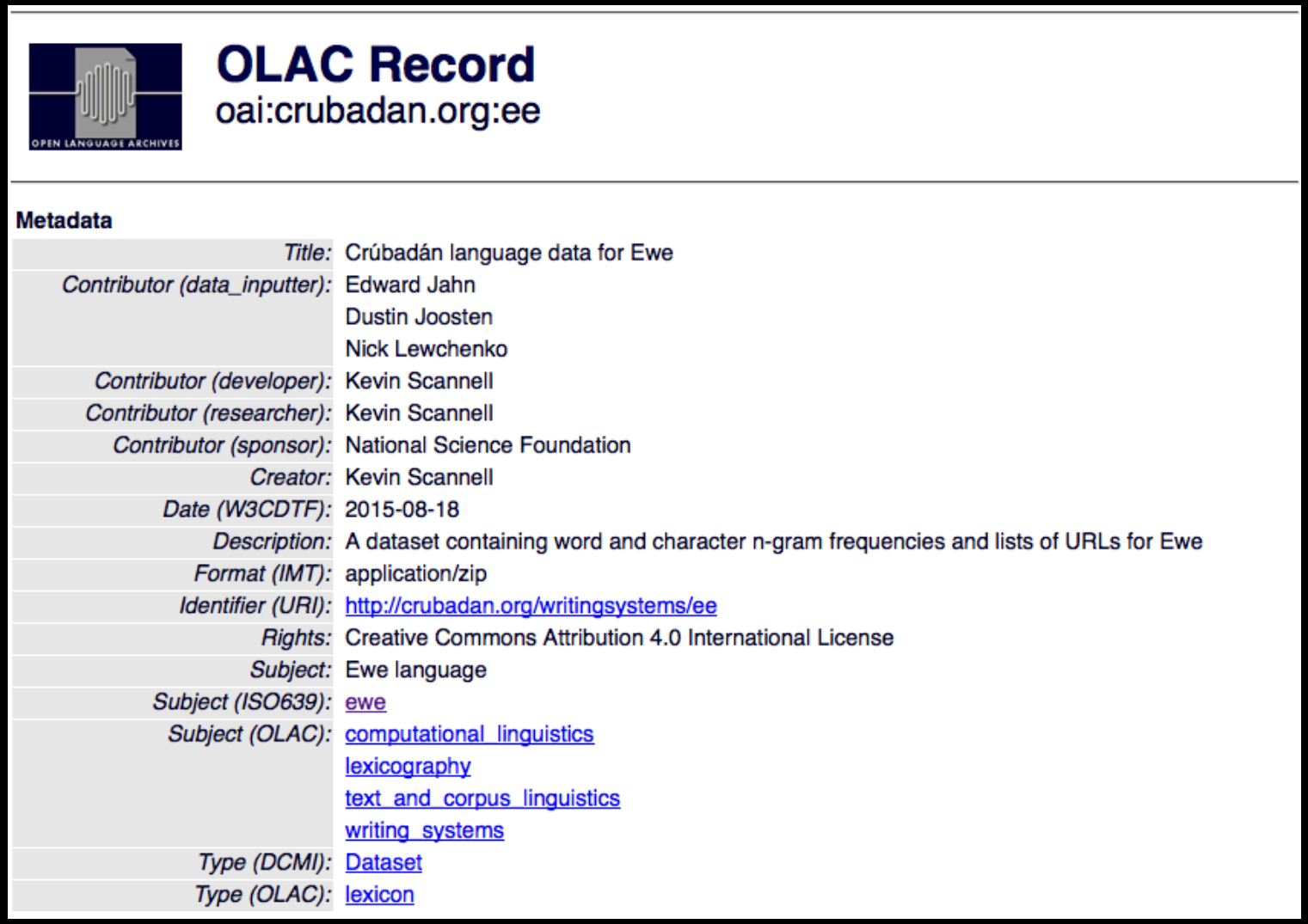
- Surprisingly little holistic consideration of user groups

- First pass at user groups
  - Linguists
  - Language communities

Questions for language archives

- Second pass at user groups
  - Users may belong to more than one group
  - Other user groups and divisions within groups
  - Users yet unborn
  - Users who aren't users
  - Archives are only user-centered for some users

Accessible to archivists and linguists



OLAC Record
oai:crubadan.org:ee

Metadata

Title:	Crúbadán language data for Ewe
Contributor (data_inputter):	Edward Jahn Dustin Joosten Nick Lewchenko
Contributor (developer):	Kevin Scannell
Contributor (researcher):	Kevin Scannell
Contributor (sponsor):	National Science Foundation
Creator:	Kevin Scannell
Date (W3CDTF):	2015-08-18
Description:	A dataset containing word and character n-gram frequencies and lists of URLs for Ewe
Format (IMT):	application/zip
Identifier (URI):	http://crubadan.org/writingsystems/ee
Rights:	Creative Commons Attribution 4.0 International License
Subject:	Ewe language
Subject (ISO639):	ewe
Subject (OLAC):	computational_linguistics lexicography text_and_corpus_linguistics writing_systems
Type (DCMI):	Dataset
Type (OLAC):	lexicon

Accessible to broader audiences

2. Work with user groups to identify needs

- Map “user experience” by conducting “user research”
- Create space for voices of users

Typical research questions

- What are the main ways that people use the technology?
- What would they like the technology to do that it doesn't do?
- What problems do they encounter, and how do they work around those problems?
- What is the broader cultural context within which the technology can help people achieve their goals?

Levels of participation

Participatory research

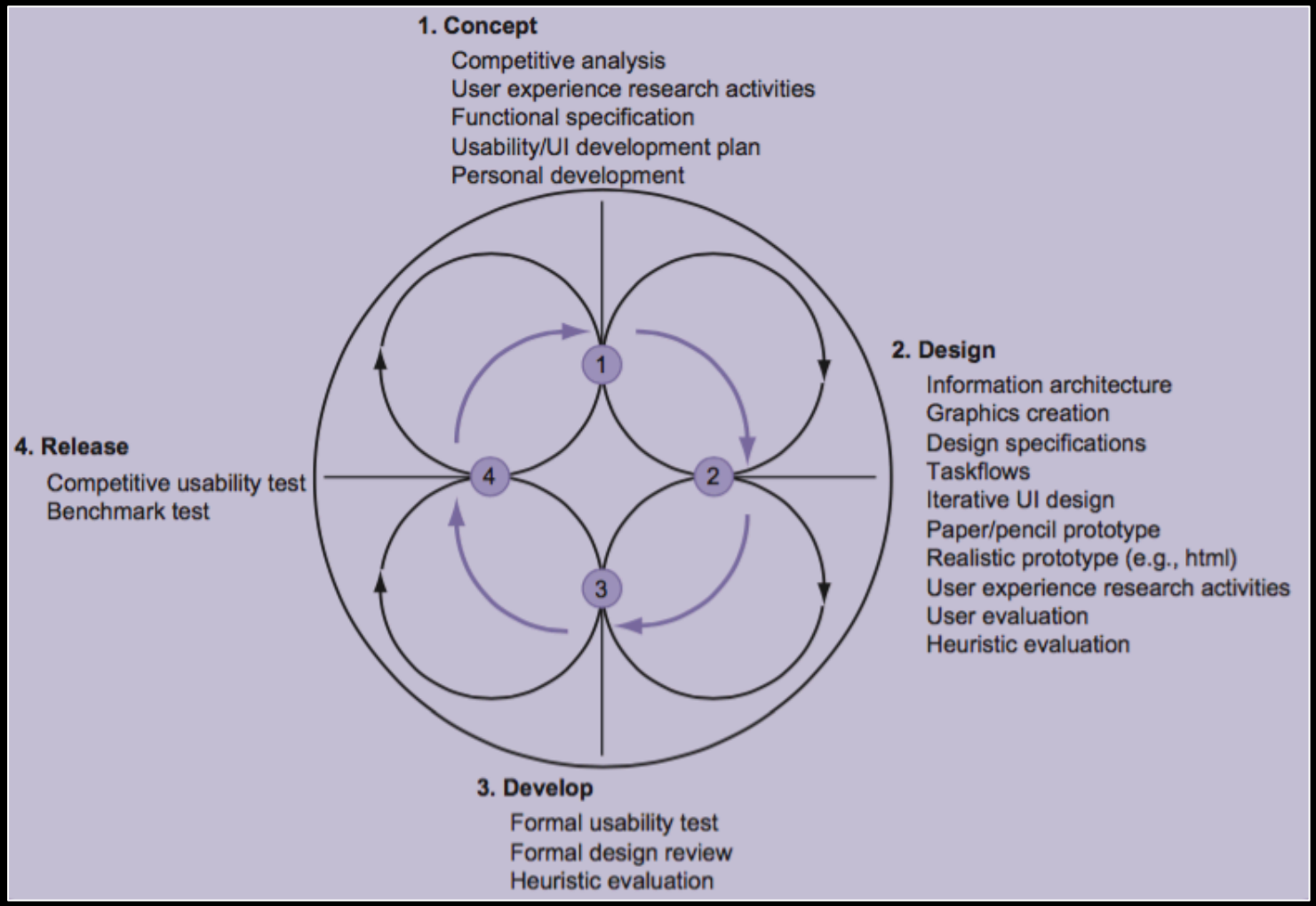
Traditional ethnographic research

Small, defined communities

Large, amorphous populations

3. Interdisciplinary and collaborative process

4. Iterative


1. Concept
Competitive analysis
User experience research activities
Functional specification
Usability/UI development plan
Personal development
2. Design
Information architecture
Graphics creation
Design specifications
Taskflows
Iterative UI design
Paper/pencil prototype
Realistic prototype (e.g., html)
User experience research activities
User evaluation
Heuristic evaluation
3. Develop
Formal usability test
Formal design review
Heuristic evaluation
4. Release
Competitive usability test
Benchmark test
1
2
3
4

What might it mean to bring UCD into field of language archives?