cs414

principles of user interface design, implementation and evaluation

Karrie Karahalios, Eric Gilbert
30 March 2007

Reaction Time and Motor Skills

Predictive Models

- *Hick's Law*
- *KLM*
- *Fitts' Law*

Descriptive Models

- *KAM*
- *Three-state Model of Graphical Input*
- *Guinard's Model of Bimanual Skill*

Predictive Models (*engineering models, performance models*)

In HCI, they allow metrics of human performance to be determined analytically without undertaking time-consuming and resource-intensive experiments.

Hick's Law

$$RT = a + b \log_2(n)$$

Where *a* and *b* are empirically determined constants.

Given *n* stimuli

Used for:

- Telephone operator selecting among 10 buttons when the light behind a button comes on.
- Measuring and predicting the time to select items in hierarchical menus (found that breadth should be favored over depth in hierarchical menus).
- Text-entry rates on soft keyboards with non-QWERTY layouts (no muscle or layout memory).

Keystroke-Level Model (KLM by Card et al)

Goal is to predict the time to accomplish a task on a computer system.
The model predicts expert error-free task-completion times, given the following parameters:

- a task or series of subtasks
- method used
- command language of the system
- motor-skill parameters of the user
- response-time parameters of the system

$$T_{execute} = t_k + t_p + t_h + t_d + t_m + t_r$$

Sum of subtask times

4 motor control operators (k is keystroking, p = pointing, h = homing, d = drawing) +

one mental operator (m) +

one system response operator (r)

t_k ranges from .08s for highly skilled to 1.20s for a typist working with an unfamiliar keyboard.

Fitts' Law

One of the most robust and highly adopted models of human movement. The model is, arguably, the most successful of many efforts to model human behavior as an information-processing activity (From John Carroll Text).

Fitts applied information theory to measure the difficulty of movement tasks and the human rate of information processing as tasks are realized.

amplitude of aimed movement ~ electronic signal

spatial accuracy of movement ~ electronic noise

human motor system ~ communication channel

Shannon's Theorem

$$C = B \log_2 (S/N + 1)$$

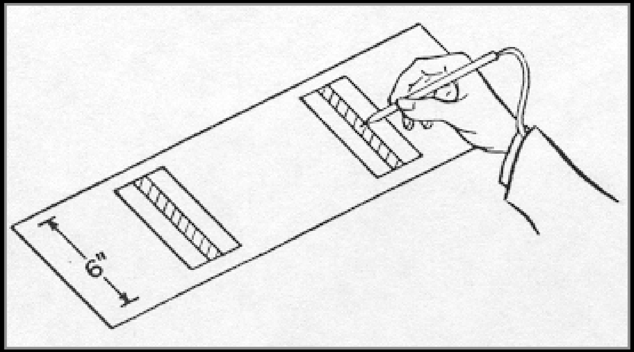
S is signal power, N is noise power

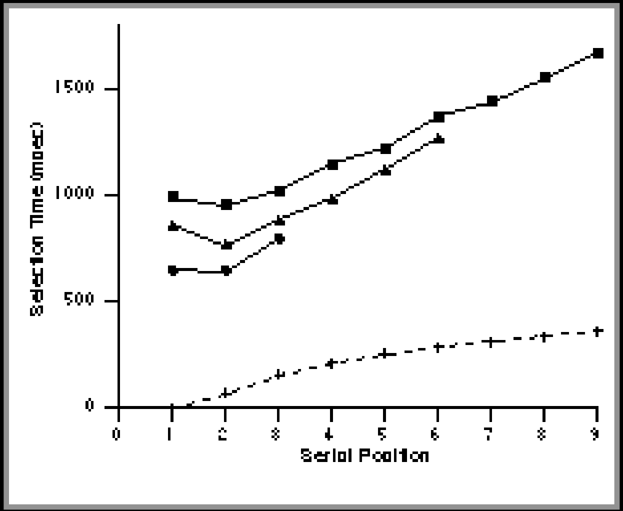
$ID = \log_2 (2A / W)$ where ID is the index of difficulty by the metric of bits

To improve the analogy, this became $ID = \log_2 (A / W + 1)$

$MT = a + b * ID$ where MT is movement time

Through empirical tests and linear regression…..





$$ID = \log_2 (A / W_e + 1)$$

$$TP = ID_e / MT$$

$$ID = \log_2 (A / W_e + .5)$$

For ID < 3 bits

Measured A(pixels), W(pixels), ID(bits), ER(%), MT(ms) for 2 devices

$W_e = 4.122 * SD_x$ where SD_x is the standard deviation in the selection coordinates over a block of trials. Replacing target W with the effective target width.

Good for predicting text entry rates on mobile phone.

Using Fitts' Law to predict text entry rates on mobile phones

(Mackenzie, 2000, 2002)

- On average, it takes 2.034 key presses to input a single English character.
- MT = 165 + 52 ID for index finger input
- MT = 176 + 64 ID for thumb input
- Results found through experiments and linear regression.
- Fitts' Law applies applies here because it is serial input with a single finger, whereas QWERTY is complex and overlaps.

Example empirical data using mouse:

- Data on the Microsoft 2.0 mouse
- Data gathered using 240 observations (12 participants x 20 trials per condition)
- Via regression, A was determined to be 545; B was 420
- Therefore, MT = 548 + 420 * ID
- ID = $\log_2 (A / W + 1)$

Combining the above 2 equations, one can solve for MT.

You can use standard pixel sizes for menu target areas.

Fitts' Law Example

Pop-up Linear Menu

Pop-up Pie Menu

Which will be faster on average?

From James Landay Notes

Fitts' Law Example

Pop-up Linear Menu

Pop-up Pie Menu

- Which will be faster on average?
  - *pie menu (bigger targets & less distance)*

From James Landay Notes

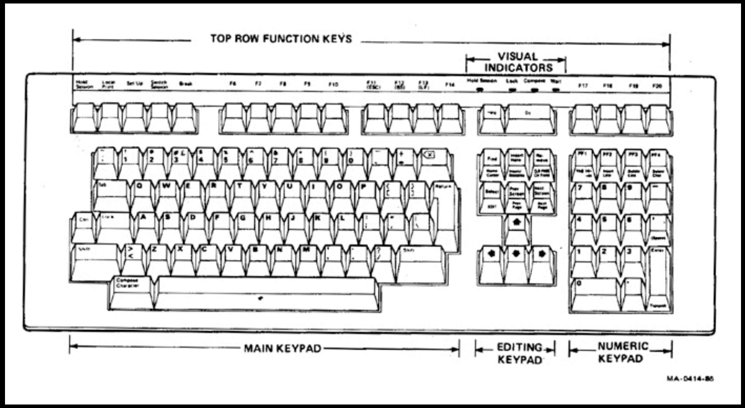
Power Law of Practice

$T_n = T_1 * n^{(-\alpha)}$

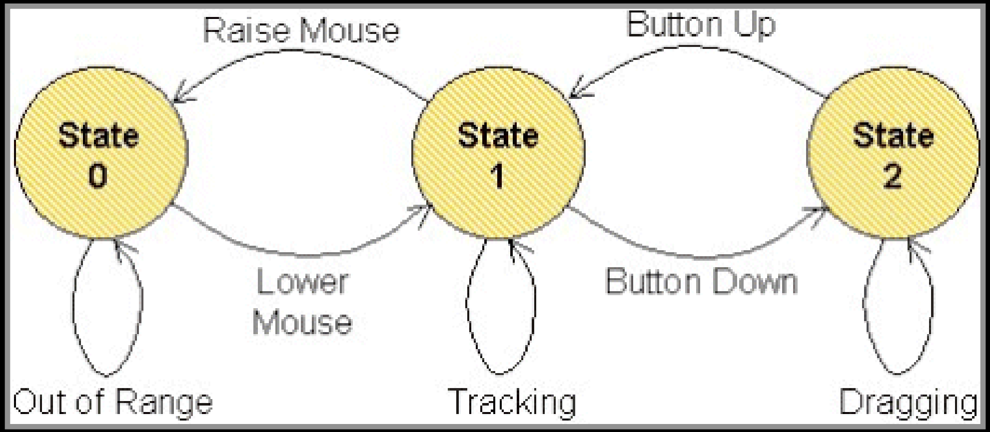
where α is learning constant (usually about 0.5),
n is the number of trials

Descriptive Models

- Key Action Model (KAM)



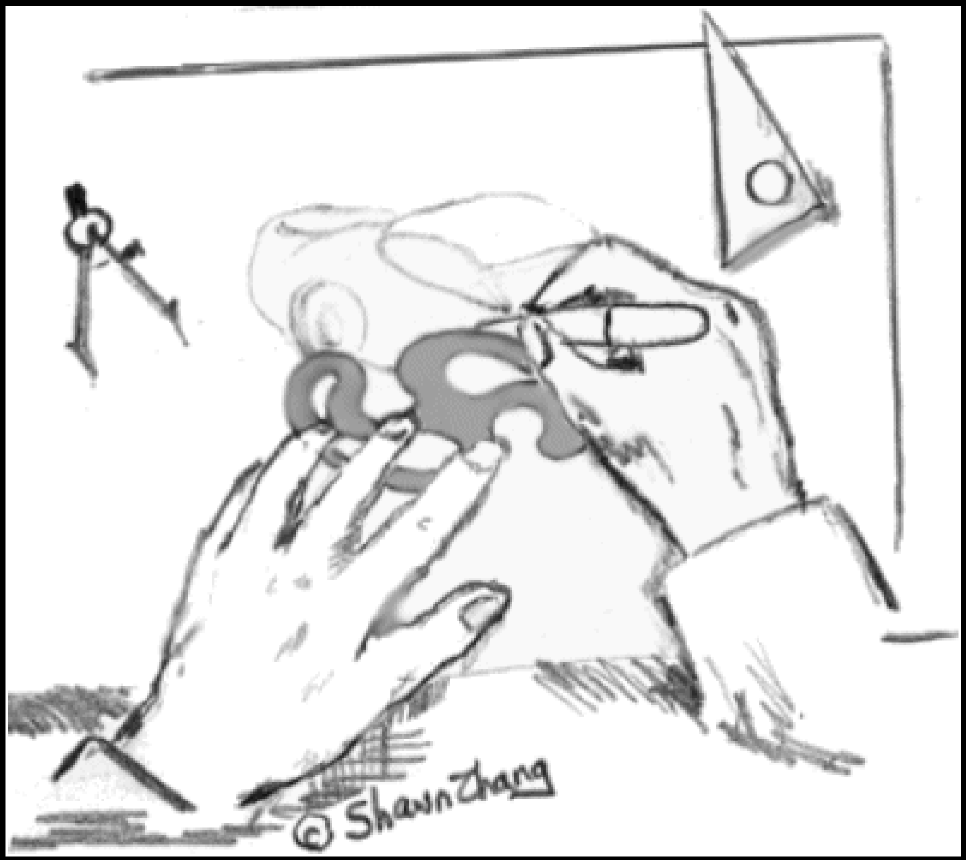
- Three-State Model of Graphical Input



Buxton, 1990

Guinard's Model of Bimanual Skill

Hand	Role and Action
Nonpreferred	• Leads the preferred hand • Sets the spatial frame of reference for the preferred hand • Performs course movements
Preferred	• Follows the nonpreferred hand • Works within the established frame of reference set by the nonpreferred hand • Performs fine movements


© ShawnZhang

low-fi prototyping

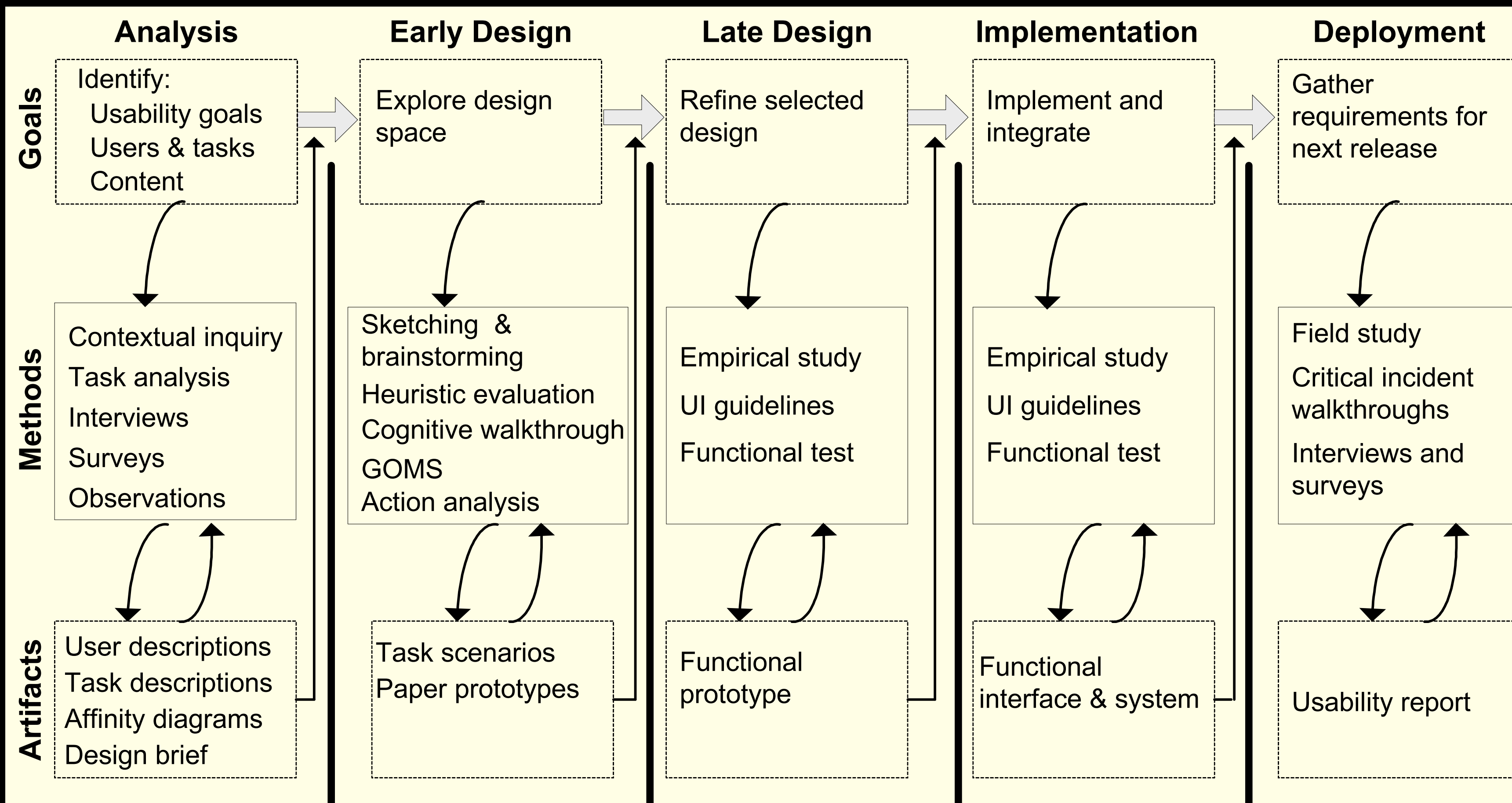
(or - how to get paid for playing with construction paper and crayons)

Brian Bailey, Karrie Karahalios, Ankur Kalra, Eric Gilbert

Messages

- Low fidelity prototypes allow rapid exploration with minimal investment
- Get your design down quickly, evaluate it, and iterate based on lessons learned
- Improves creativity and resulting design

In Context of TCUID



Prototyping in HCI

- **Build a scaled model of an interactive system**
  - externalize your design thinking
  - enables communication
- **Evaluate the model against some criteria**
  - measure how “good” the model is
  - compare against other models
  - learn how to improve it
- **Iterate as necessary**
  - integrate lessons, increase fidelity and detail
  - iterate rapidly at first, then slow as design solidifies

Prototype and Evaluation Stages

Early design

Rough out on paper	Low-fidelity (paper-based)
Cognitive walkthrough	
Formative evaluation	
Action analysis	Medium-fidelity
Heuristic evaluation	
Functional prototype	High-fidelity (computer-based)
Empirical studies	

Late design

Iterative Design

- **Build, evaluate, iterate, …**
  - iterate rapidly at first, slow as design solidifies
- **Increase fidelity and add detail**
  - change from paper to computer medium
- **Evaluate different aspects of the design at different stages of the design process**
  - **early**: metaphor, structure, usefulness
  - **late**: performance, learnability, satisfaction

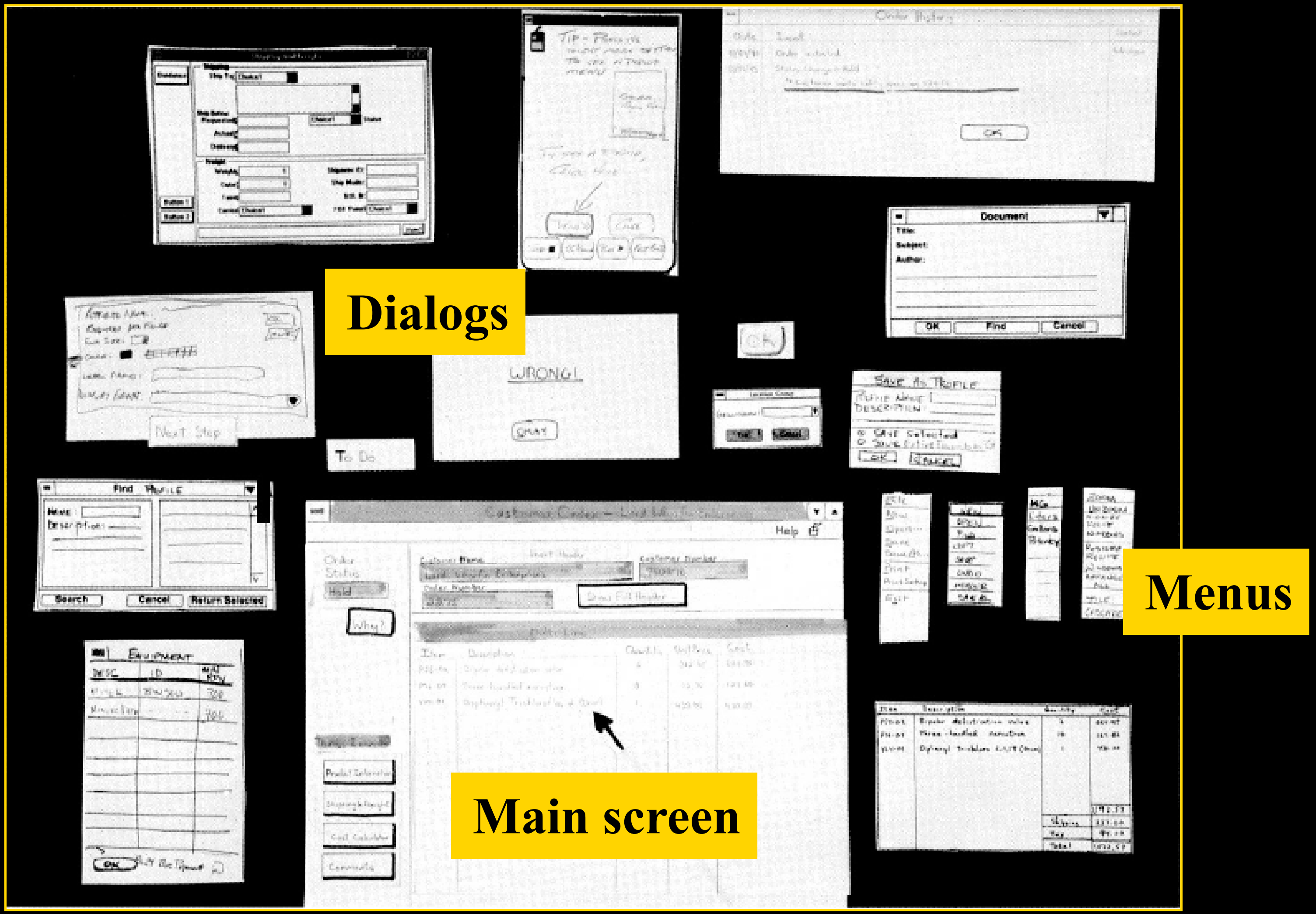
Prototyping Tools

- **Use tools to construct models**
  - paper, informal tools, development tools
- **In choosing a tool, consider:**
  - requirements of the project
  - expertise of design team
  - access to relevant libraries
  - *balance investment with effectiveness*

Low-fidelity Prototype

- Rough cut of your interface design created with paper, post-it notes, overlays, correction tape, etc.
- Sketch storyboards to show overall design concept and interface structure
- Use post-its and overlays to simulate critical interactions
- Build the prototype to support your tasks

Low-fidelity Tools



Benefits of Low-fidelity

- Quick and cheap to build prototype
- Communicates design concept and structure, can demonstrate interactions
- Facilitates brainstorming and invites discussion
- Enables early evaluation
- Maximizes number of design refinements before you commit to code
- Enables rapid and extensive exploration of the design space

Materials

- Paper: heavy paper for base interface screens
- Overlays: show changes in content
- Post-it notes: show changes in content
- Adhesives: glue sticks, correction tape
- Colored markers and pencils: enable highlighting, drawing in color
- Scissors: cut content to size
- Library: Pre-fabricate menus, buttons, tabbed panes, dialogs, etc.

Build the Prototype

- **Sketch interface screens on heavy paper**
  - called *storyboards*
- **Build interactions using “library”**
  - menus, dialogs, tabbed panes, buttons, etc.
- **Assemble components to enable users to perform each task from your analysis**
- **Do not debate the design too much, get your ideas down and ready for evaluation**
  - quality will improve with iteration

Evaluate Low-Fi Prototype

- **Identify "big" problems**
- **Perform rapid iteration**
- **May catch problems that empirical tests could miss (e.g., consistency issues)**
- **Evaluations are only effective if your team**
  - has the right skill set
  - wants to improve the design, not defend it

Evaluation Techniques

- Formative evaluations
- Wizard of Oz studies
- GOMS and action analysis

- Cognitive walkthroughs
- Heuristic evaluations

Evaluating the Techniques

- **Does the technique identify usability issues that users will encounter in practice?**
- **Can the technique help you identify design solutions? Does implementing these solutions actually fix the usability issue?**
- **When can you use the technique?**
- **What are the costs of learning and applying the technique? What types of usability issues does it discover?**

Formative Evaluation

- **Evaluate how well users can perform tasks with your low-fidelity prototype**
  - have a user perform a task with prototype
  - manipulate prototype to make it interactive
  - identify trouble points and solutions
  - revise prototype and perform again

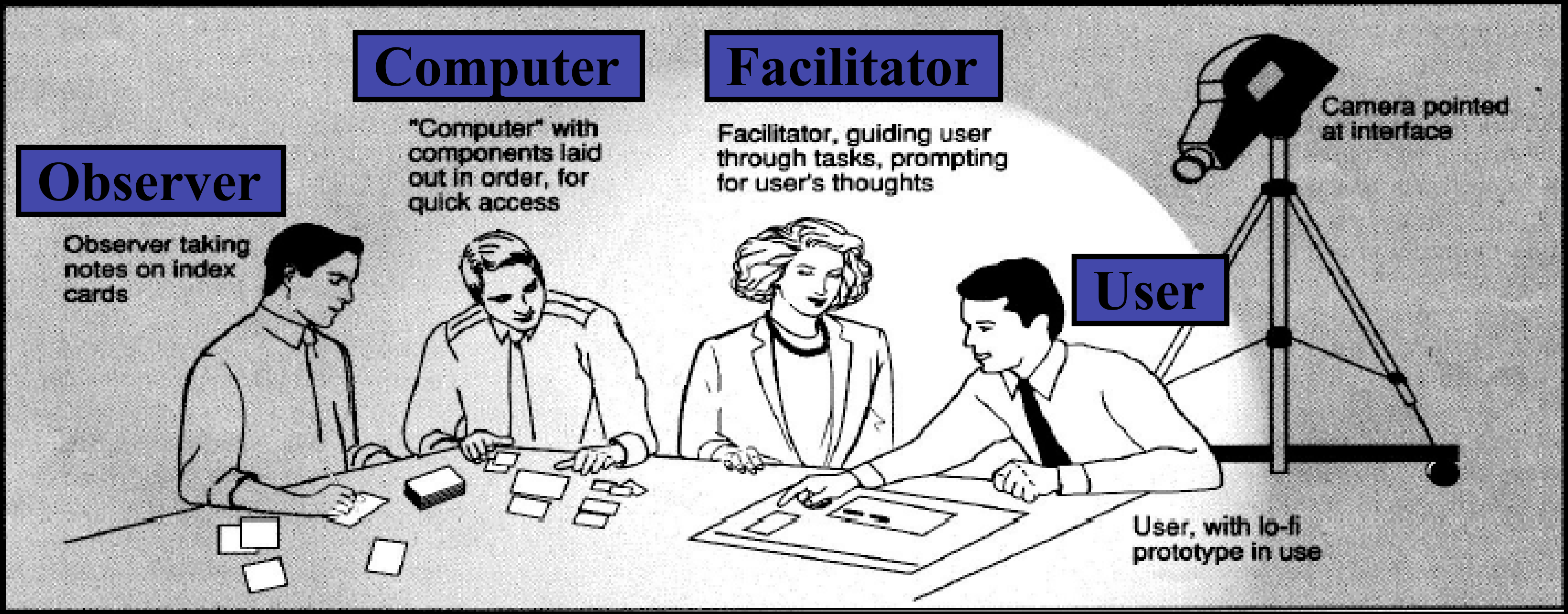
What You Need

- **User and task descriptions**
- **Low-fidelity prototype with enough "functionality" for several tasks**
- **An evaluation team that consists of a**
  - "computer"
  - facilitator
  - note taker

What To Do

- Ask the user to perform a task
- Manipulate prototype to reflect actions
- Ask the user to think aloud
- Identify trouble points with the interface
- Write down each usability issue

Formative Evaluation



[Rettig, 94]

After the Evaluation

- **Reflect on the results**
  - write each usability issue on a post-it note
  - identify severity and frequency
  - use affinity diagram techniques
- **Explore larger design changes to address multiple usability issues**
  - small variations may not be enough, this is the best time for “big” changes
- **Change your prototype and re-evaluate**