Agenda

6:00 ......Goals and Introduction

6:05 ......Origin and Need of HCI

6:15 ......Some Structure for Understanding HCI

6:50 ......Psychology in HCI

7:00 ......(break)

7:30 ......Psychology in HCI (continued)

8:10 ......Computer Science and HCI

9:00 ......Conclusion and Wrap-up

9:30 ......End

Objectives of the Course

- What is HCI and why is it important?
- Brief history of HCI
- Introduction to building usable systems
- Introduction to the psychology of HCI
- Introduction to computer technologies for HCI
- Future directions of HCI
- Where to learn more during the conference
- Where to learn more in the published HCI literature

Business Case: Corporate ROI for Usability is Huge

Joint study on overhead reduction by Boeing and Microsoft on Windows 2000

- Conservative set of assumptions:
  - User population = 100k
  - Product life = 5 years
  - Mean work-stoppage time = 4.1 minutes
  - Data on work-stoppage from survey of Boeing users
  - Compared work-stoppage for versions of Windows 2000
  - value estimate between $31m-$65m

- **Usability improvement ROI = 1203 for Boeing**

- Conclusion: Poor usability is a large, hidden source of uncontrolled overhead

For other examples, see:

- Cost-Justifying Usability. Randolph G. Bias and Deborah J. Mayhew, Eds., Academic Press, 1994

2.2 Design-induced Error

Bad usability can be catastrophic, even for highly-trained operators

- Analysis of 460 aircraft non-combat incidents
- Operator misunderstanding of status lights at Three Mile Island
- Radiation overdoses to cancer patients by the Therac-25

3.0 Some Structure & History for Understanding the Field of HCI

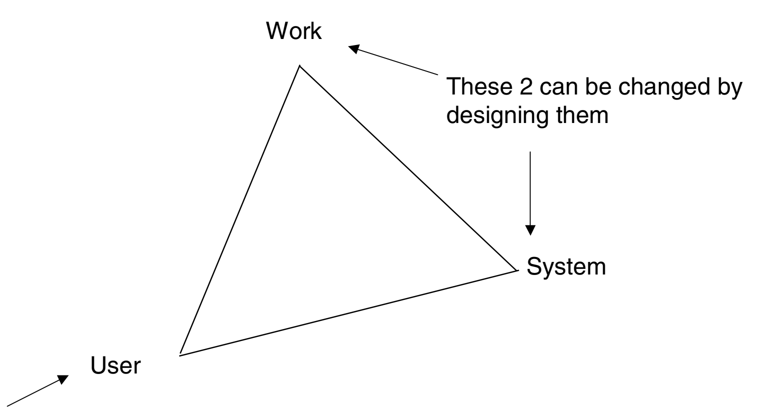
3.1 Designing Good HCI is Complex
- High-level task analysis to understand HCI design
- Usability is some inverse function of overhead
- Tools for Usability Engineering

3.2 Usability Engineering for Application Development
- Iterative prototyping methods
- New *ANSI* standard for software product usability
- Design support technology

3.3 R&D for HCI Technology
- HCI draws on many technologies
- The various technologies differ in maturity

3.4 Where to Learn More-
- In the published literature
- Here at CHI2003

3.1 Designing for Good HCI is Complex: A 3-way, Highly Dynamic Interaction



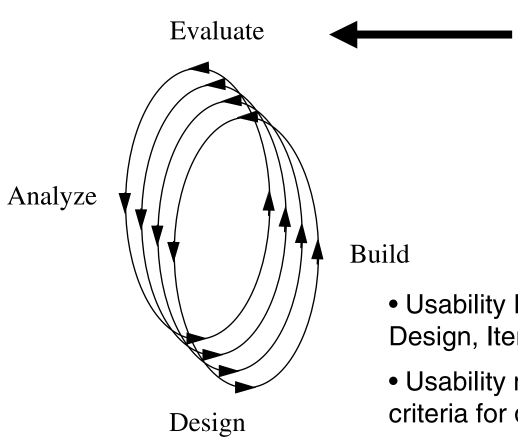
Users are self-adaptable, given enough chance to practice- but requiring them to change requires time, training, and imposes complexity, overhead, and error-proneness

Human Factors Design Methods

Usability Engineering inherited many design methods from cockpits -

- Both pilots and users participate in design
- Both the mission and user work are analyzed for re-engineering with technology
- Prototypes of interfaces are built to test concepts and discover problems before product delivery
- Both pilots and users test prototypes on representative tasks
- Product acceptance requires satisfaction of objective task performance criteria

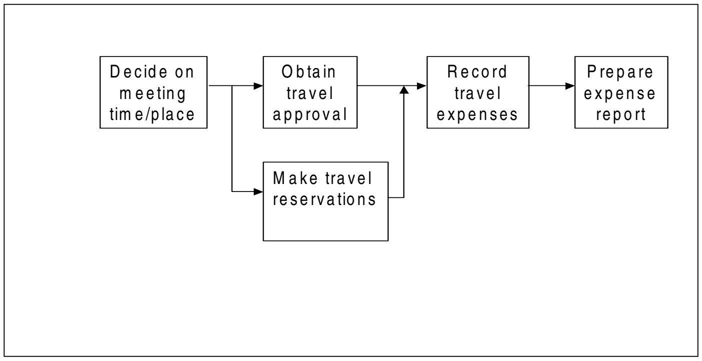
3.2 Usability Engineering for Application Development



- Usability Engineering (UE) is a.k.a. Spiral Design, Iterative Prototyping, etc.
- Usability metrics provide completion criteria for design cycles [20, 26]
- UE satisfies first stage of the Use-Case Driven Method (15) for object-oriented software engineering
- Steps defined in ISO 9241-11; ISO 13407

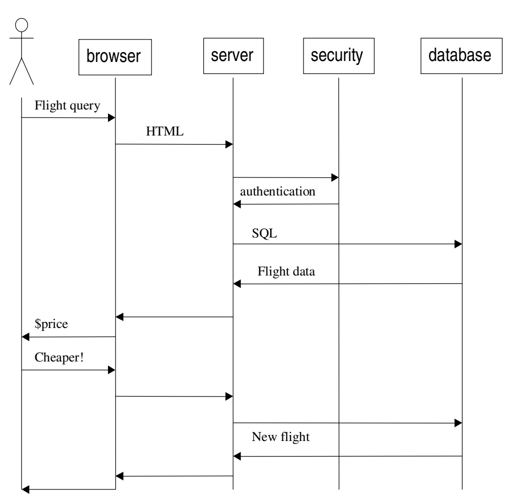
Popular Diagrams for Modeling Work

User-centered



From the perspective of the user, a UML activity diagram (see Fowler [12])

Technology-centered Diagram of Work

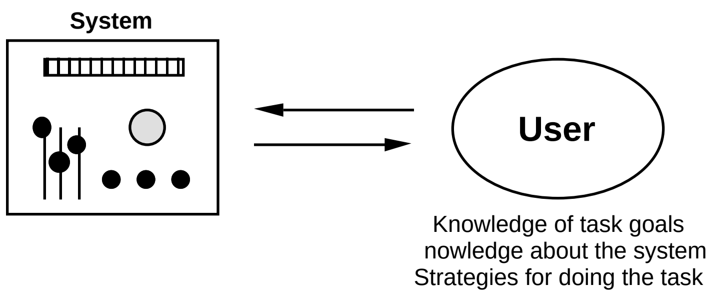


From the perspective of the application, a UML event-sequence diagram (see Fowler [12])

The Big Picture: Users and Systems

The user knows what tasks to accomplish, but also has to know *how* to accomplish the tasks.

- E.g., which buttons to press to obtain the desired result.



User must come up with a procedure by using strategies:

Execute an already known procedure.

- Simply using a routine skill.

Get a procedure from instructions.

- Can be made more efficient based on research results.

Infer a procedure by applying problem-solving strategies:

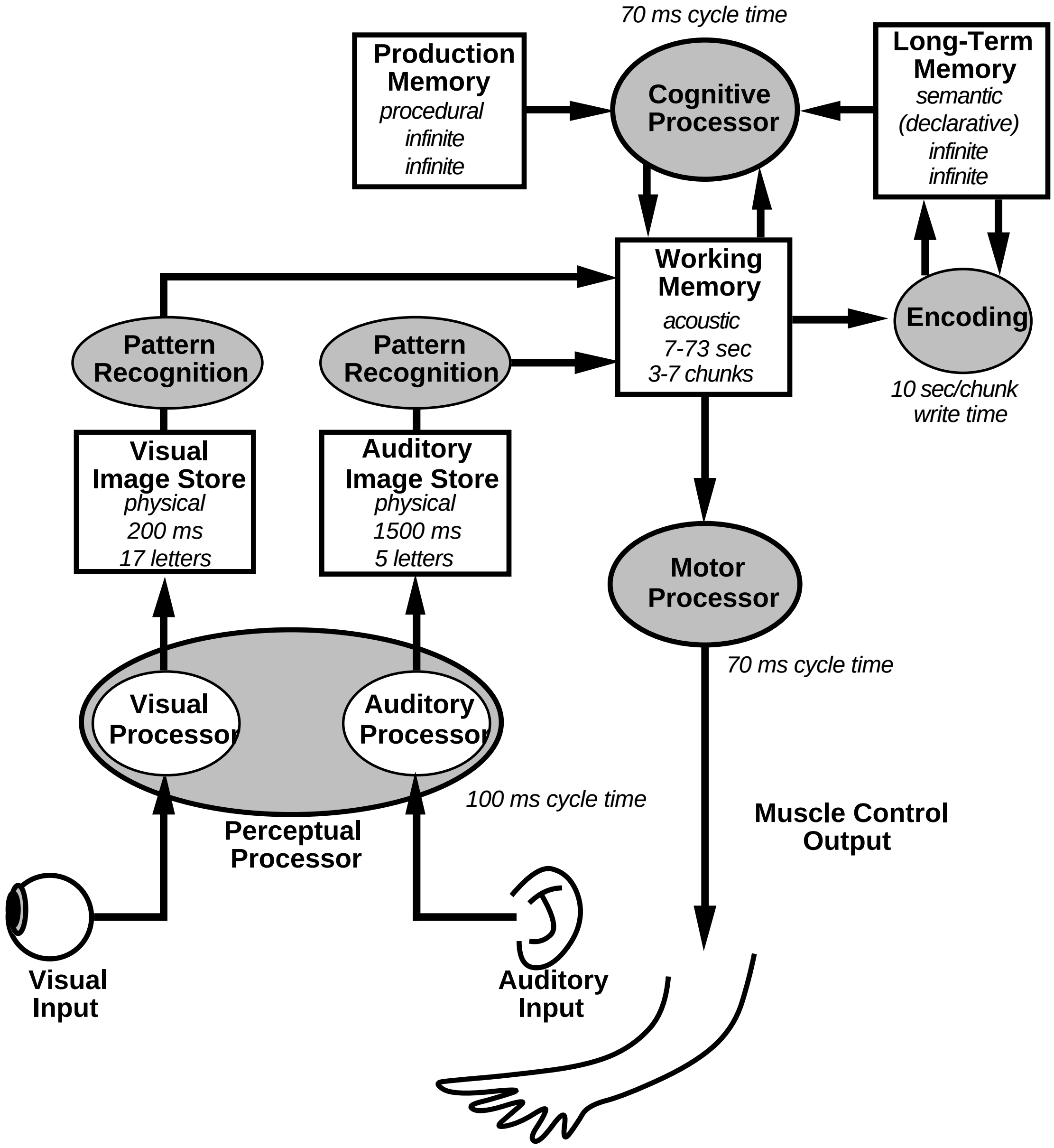
- Use analogy from a similar system with same function or purpose - tends to have similar procedures.
- Use trial and error based on:
  - Behavior upon manipulation.
  - "Affordances" - physical appearance and constraints.
- Use knowledge of how the system works.
  - Previously known or in documentation.
  - Infer based on a "mental model."

The Model Human Processor

Interconnected processors and memories.

- Each processor has a cycle time.
- Each memory has:

    Information representation type.
    Half-life of information (exponential decay).
    Storage capacity.



Modified from Card, Moran, & Newell (1983)

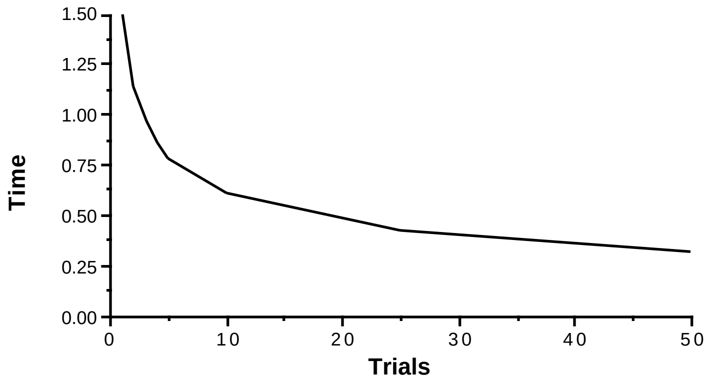
Power Law of Practice

Performance time for a skill decreases according to

$$T_n = T_1 n^{-\alpha}$$

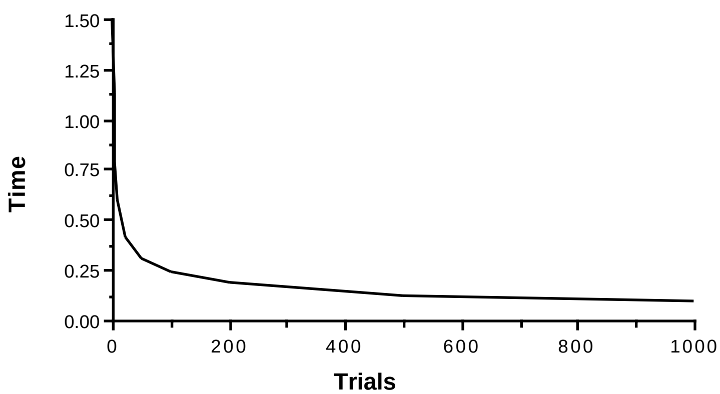
where *n* is the number of trials, typically $\alpha = .4$

Time over the first 50 trials:



Very rapid initial decrease, begins to level off.

Time over the first 1000 trials:

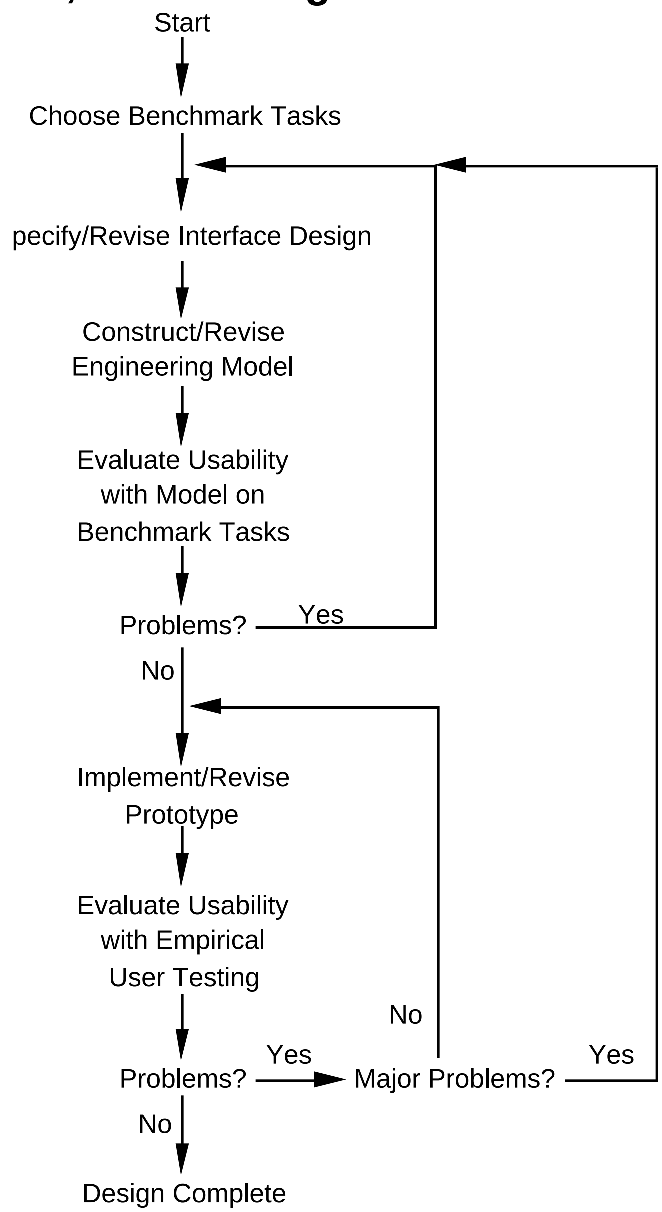


Initial decrease is really fast, then almost stable.

Many practiced skills have essentially stable performance times, enabling practical prediction.

Engineering Model Process

Use model first, user testing for final check

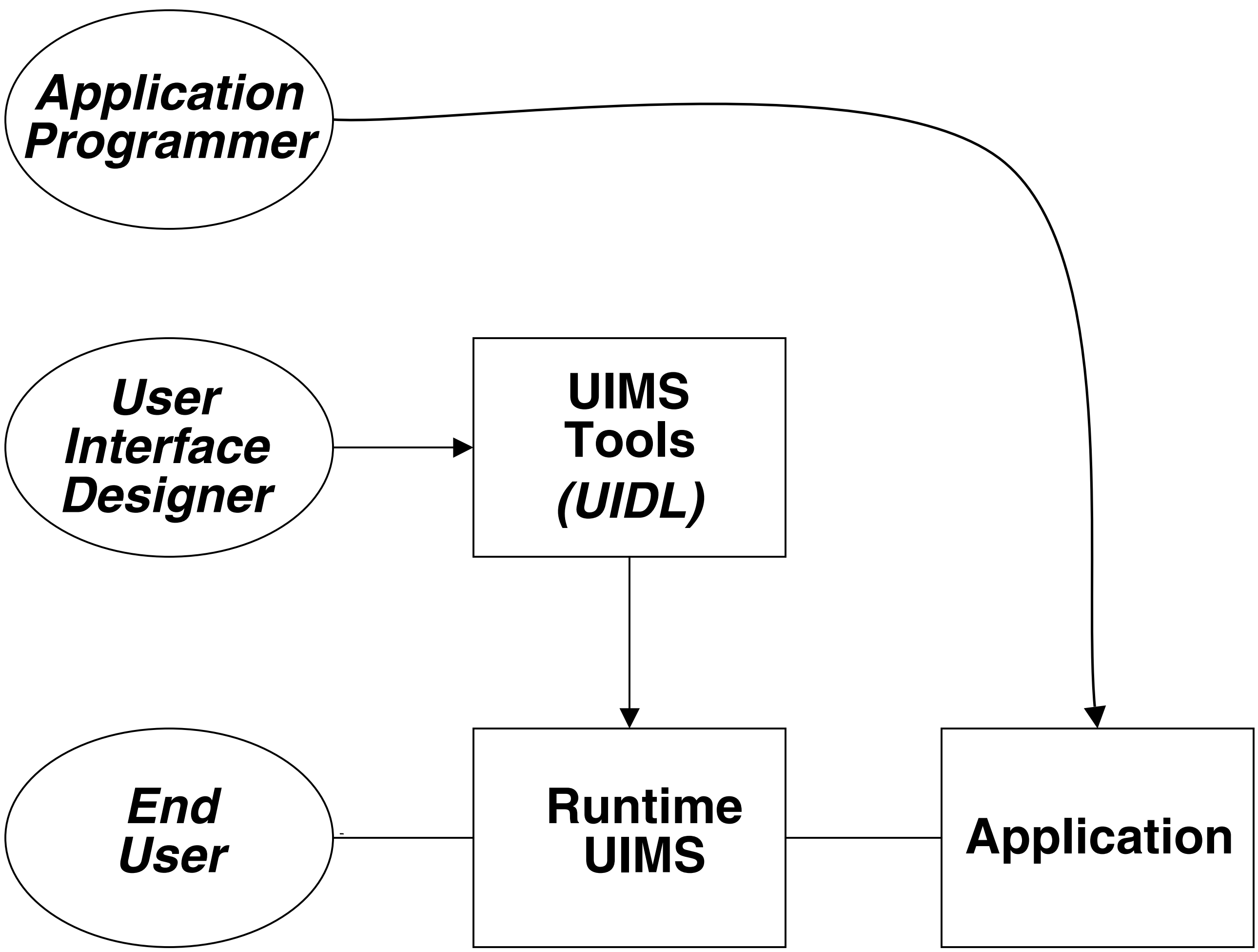


May need to loop all the way back, but model-based evaluation will help then also

5.2. User Interface Software Concepts

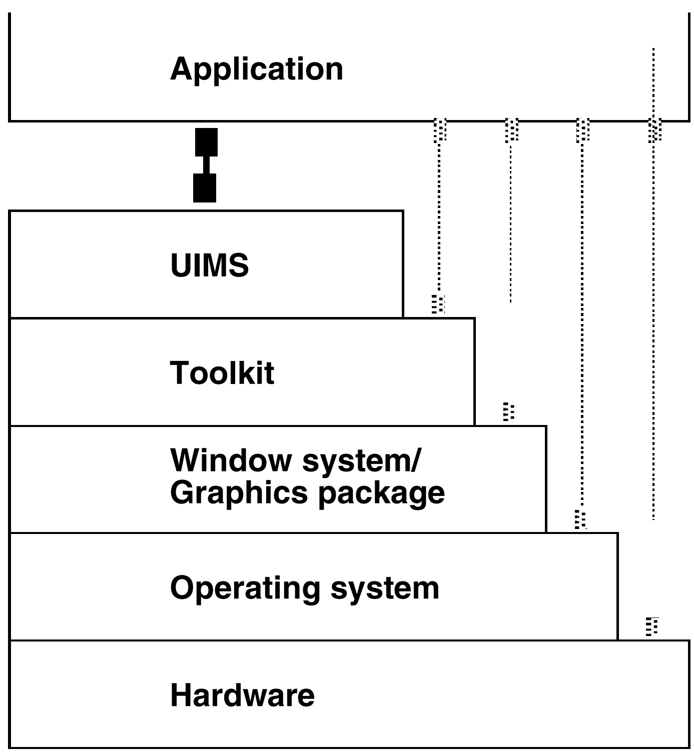
- Dialogue independence
  - Interface (syntax) vs. application (semantics)
- Interaction techniques
- Basic interaction styles
- Design levels: conceptual, semantic, syntactic, lexical
- User interface management system (UIMS)
- User interface description language (UIDL)
- User interface software tools
- New interaction styles (non-WIMP)

Classes of Programmers and Tool Users



5.3. User Interface Software Tools

- **Goal: Support dialogue independence and separation of programming roles [Foley, van Dam, Feiner, & Hughes]**



UIMS Reference Model