cs465

principles of user interface design, implementation and evaluation

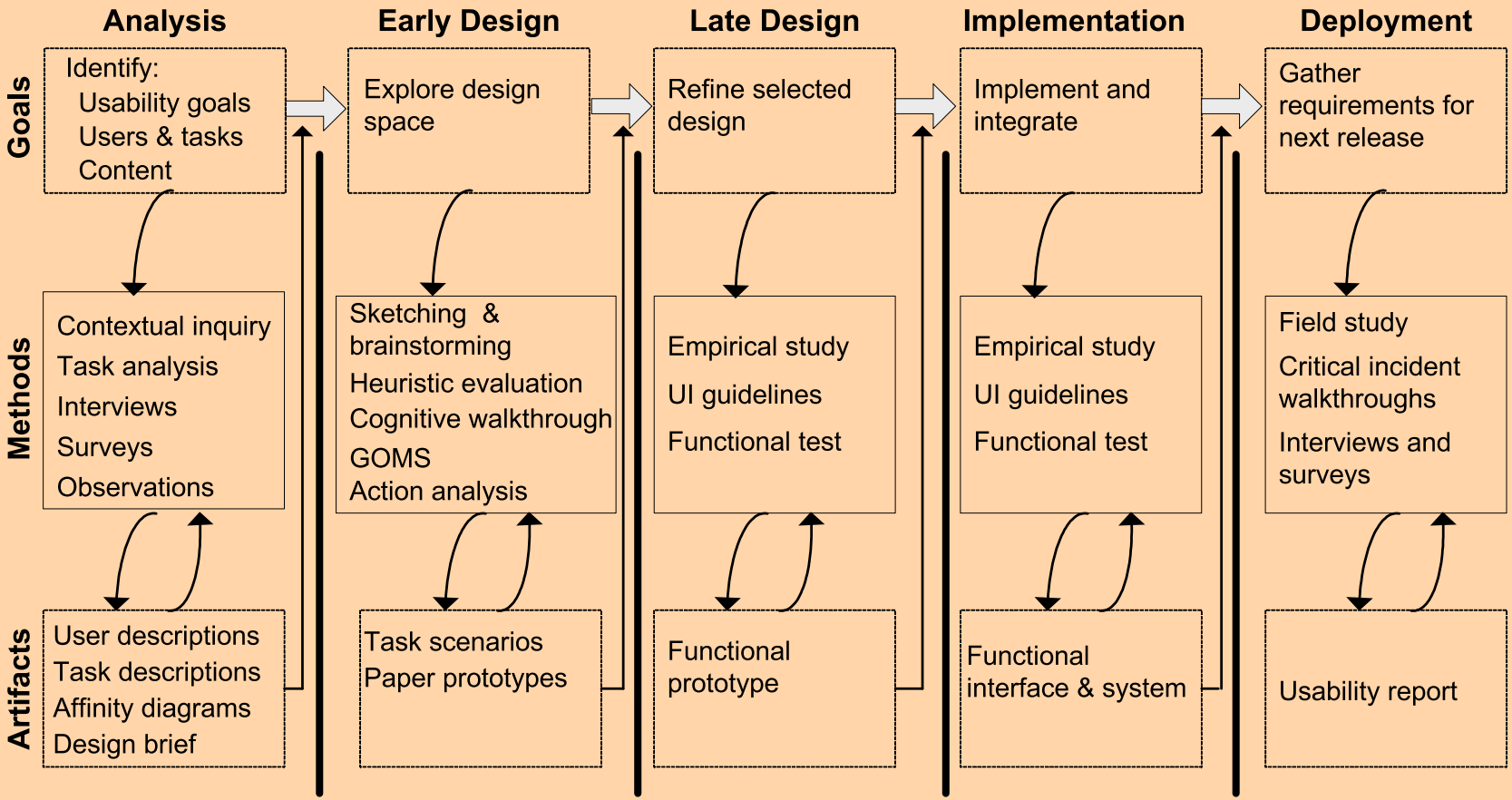
Karrie G. Karahalios
30. September 2008

1. GOMS
2. Action Analysis
3. Discuss Homework Assignment

Big Picture

- Quantitative analysis important in early design
  - performance, error rate, learnability
- Use GOMS and action analysis to predict task performance with interactive systems
  - no implementation required
- Perform GOMS and action analysis on low-fidelity prototypes or existing applications
  - assumes error-free, goal-directed, rational behavior

Context of TCUID - Task Centered User Interface Design



Task Performance

- Task performance is critical in
  - airline and automobile displays
  - emergency management systems
  - process control systems
  - customer service systems and more
- Should measure task performance early to
  - minimize task performance on high-frequency tasks
  - select among alternative designs
  - ensure that critical performance goals will be satisfied
  - cost justify replacement of an existing system

Real-World Example

- For every second saved in operator support, a company could save 3 million dollars per year
  - NYNEX estimate for its operator support, [Gray et al., GOMS Meets the Phone Company, Interact, 1990]
- Replace old workstations with new workstations
  - promised to reduce operator support time by 2.5s
  - weigh against investment of the new systems (about 1000 workstations at $10,000 each)
- Conduct empirical study to compare operator performance on old and new systems
- Perform GOMS analysis to help explain results

Findings

- Operators slower on new workstations
  - would have cost another $3 million per year
- GOMS analysis showed that an operator had to perform more operations along the critical path for the new systems
  - GOMS is a predictive and explanatory model

GOMS

- *Goals*: what a user wants to accomplish
- *Operators*: mental or physical actions that change the state of the user or system
- *Methods*: groups of goals and operators
- *Selection rules*: determine which method to apply, if more than one available

GOMS

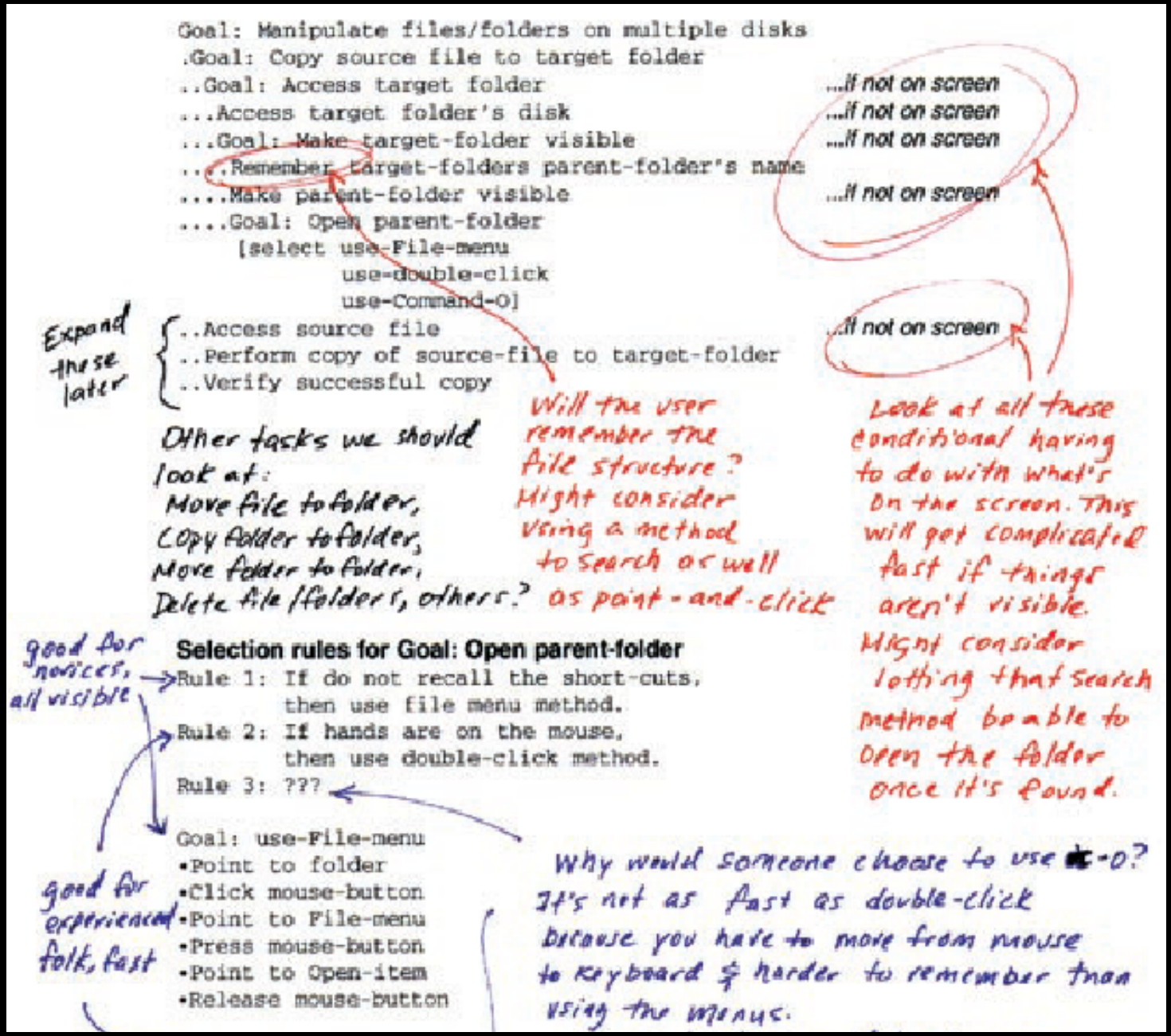
- A method to describe tasks and how a user performs those tasks with a specific design
  - bridges task analysis with a specific interface design
  - error-free, goal-directed, and rational behavior
- Views humans as information processors
  - small number of cognitive, perceptual, and motor operators characterize user behavior
- To apply GOMS:
  - analyze task to identify user goals (hierarchical)
  - identify operators to achieve goals
  - sum operator times to predict performance

GOMS Can Be Used To:

- Develop task-centered documentation
- Predict time to learn how to perform tasks
- Predict likelihood of errors
- Predict time to perform tasks
  - predictions tied to specific interface designs

Apply GOMS When

- Want a formal method of writing tasks
  - enables you to identify intersections across tasks, but requires a consistent vocabulary
  - generates discussion (concrete representation)
  - matches tasks to specific interface design
- Want to make tasks more efficient
  - or just the repetitive parts of larger tasks
  - even creative tasks have repetitive parts


Goal: Manipulate files/folders on multiple disks
.Goal: Copy source file to target folder
..Goal: Access target folder
...Access target folder's disk
...Goal: Make target-folder visible
...Remember target-folders parent-folder's name
....Make parent-folder visible
....Goal: Open parent-folder
[select use-File-menu
use-double-click
use-Command-O]
..Access source file
..Perform copy of source-file to target-folder
..Verify successful copy
...if not on screen
...if not on screen
...if not on screen
...if not on screen
...if not on screen
Expand these later
Will the user remember the file structure? Might consider using a method to search as well as point-and-click
Look at all these conditional having to do with what's on the screen. This will get complicated fast if things aren't visible. Might consider letting that search method be able to open the folder once it's found.
Other tasks we should look at:
Move file to folder,
Copy folder to folder,
Move folder to folder,
Delete file/folders, others?
Selection rules for Goal: Open parent-folder
Rule 1: If do not recall the short-cuts, then use file menu method.
Rule 2: If hands are on the mouse, then use double-click method.
Rule 3: ???
good for novices, all visible
Goal: use-File-menu
•Point to folder
•Click mouse-button
•Point to File-menu
•Press mouse-button
•Point to Open-item
•Release mouse-button
good for experienced folk, fast
Why would someone choose to use ⌘-O? It's not as fast as double-click because you have to move from mouse to keyboard & harder to remember than using the menus.

John, B.E. 1995

Who Can Use GOMS

- Just about everyone
  - formal training not required; experience helps
- Have multiple people perform analysis and compare results
  - results are often surprisingly consistent

How To Use GOMS

- Analyze hierarchical structure of a task
  - coarse analysis focuses more on the cognitive structure of a task
  - fine analysis focuses more on the structure imposed by the specific interface design
- Analyze alternative methods
- Assign operators to base level goals
- Assign times to operators
- Sum the operator times

Operator Times

Press key on keyboard	280 ms
Use mouse to point to object on screen	1500 ms
Move hand to pointing device	300 ms
Move eyes to location on screen	230 ms
Retrieve item from memory	1200 ms
Learn a single step in a procedure	25 seconds
Select among methods	1200 ms

More available in TCUID chapter 4

GOMS Example

- Retrieve the article entitled “Why Goms?”
  - written by Bonnie John, 1995, in ACM DL

Goal Structure

- Goal: Retrieve article from ACM DL
  - Goal: Go to ACM
    - Goal: Enter ACM URL
    - Goal: Submit URL
  - Goal: Go to DL
    - Goal: Locate DL link
    - Goal: Select the link
  - Goal: Select method
    - [Method: Search method
    - Goal: Search for article
      - Goal: Enter search parameters
      - Goal: Submit search
      - Goal: Identify article from results
    - Goal: Select the article]
    - [Method: Browse method - <take home exercise>]
  - Goal: Save article to disk
    - Goal: Initiate save action
    - Goal: Select location
    - Goal: save article to that location

Can GOMS Be Trusted?

- Predictions made by GOMS models validated in many research studies
  - assumes that you have a valid model!
- Build initial model based your own understanding of a task's execution
  - record other users performing the task
  - compare predicted versus actual sequence
  - refine and iterate

GOMS Worth the Effort When

- Want quantitative estimates of human performance without having to
  - build a working system
  - train people to use the system to measure performance
  - measure performance for many users

GOMS: Pros and Cons

- Pros
  - predict human performance before committing to a specific design in code or running empirical studies
  - no special sills required
  - many studies have validated the model (it works)
- Cons
  - assumes error-free, skilled behavior
  - no formal recipe for how to perform decomposition
  - may require significant time investment

In-Class Exercise

- Perform a GOMS analysis for a task that your initial interface design supports

Action Analysis

- Write down each step that a user must perform in your interface to achieve a task
- Multiple number of steps by [2, 3] secs
  - provides range of [best, worst] performance

Action Analysis Example

- Enter URL String
- Press “Enter” key
- Find “digital library link”
- Select the link
- [assume search method]
- Enter title of article into search field
- Select “Search”
- Find “Why GOMS” link
- Select the link
- Select “Save” button
- Select folder location
- Select “Save” button on dialog

Action Analysis Example

- Enter URL String
- Press “Enter” key
- Find “digital library link”
- Select the link
- [assume search method]
- Enter title of article into search field
- Select “Search”
- Find “Why GOMS” link
- Select the link
- Select “Save” button
- Select folder location
- Select “Save” button on dialog

12 Steps = [24, 36] seconds

My actual time = 28 seconds

Pros and Cons

- Pros
  - faster to perform
  - easier for a beginner
  - good for less performance critical apps
- Cons
  - less accurate (higher variance)
  - more difficult to compare alternative designs that are close in predicted performance

In-Class Exercise

- Perform an action analysis for same task and compare predicted time with GOMS