Interface Evaluation — Qualitative Evaluation

Chapter 9

The slides in this lecture are partially based upon this from Ms. Tiffany Tang and Drs. Vincent Ng and Saul Greenberg.

Use and Context

Human Social Organization

Human-Machine Fit and Adaptation

Applications

Human

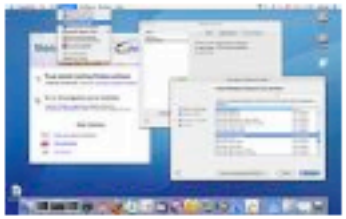
Human Information Processing

Language, Communication, Interaction

Ergonomics

I/O Devices

Computer



Interface Metaphors

Graphic Design

Dialogue Techniques

Prototypes

Evaluation Techniques

Implementation Techniques and Tools

Design Approaches

Development Process

Lecture Overview

- What you'll learn in this lecture
  - How to quickly debug a system by observing users.
  - How to find out "what people are thinking about" when they use your system.
  - How to conduct usability tests.

Usability Testing: What is it?

- Usability Testing is a means of measuring how well people can use an artificial object (e.g. webpage, pen, car, etc) for its intended purpose.

- In other words, measures the *usability* of the object.

Why should we test?

- Can’t tell how good UI is until it is tried for sure!
- Hard to predict what “real” users will do.
  - Most interfaces are “tested” by designers pretending to be users.
  - But most users do not have the same background information that the designer has.
  - Also, almost impossible to “forget” elements of an interface that you designed!
- Would you fly in an airplane that has only been “tested” by the person who built it?

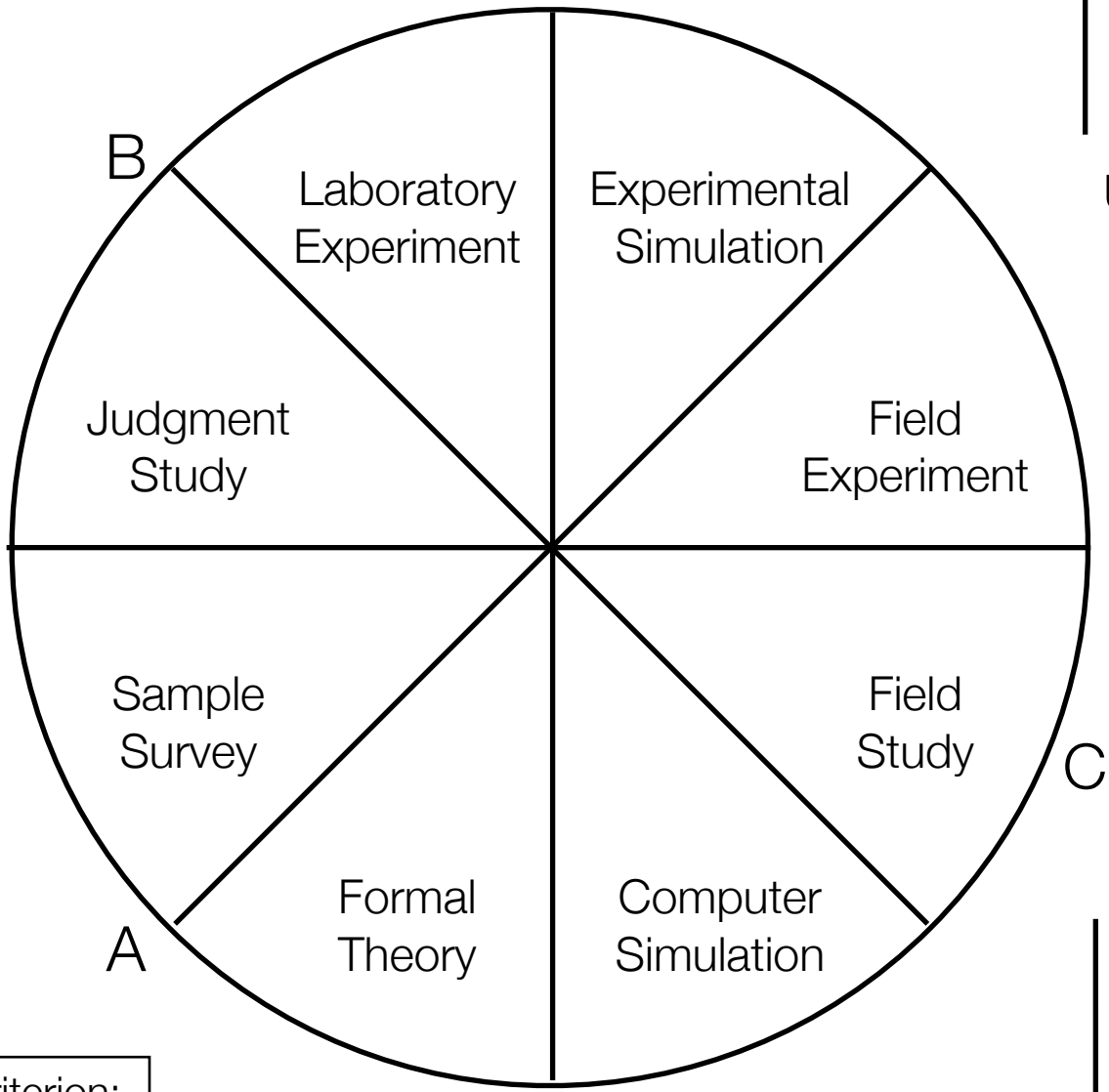
When to Test?

- Pre-design
  - Investing in new expensive system requires proof of viability
- Initial design stages
  - Develop and evaluate initial design ideas with the user
- Iterative design
  - Does system behavior match the user's task requirements?
  - Are there specific problems with the design?
  - What solutions work?
- Acceptance testing
  - Verify that system meets expected user performance criteria
    - E.g. "80% of 1st time customers will take 1-3 minutes to withdraw $50 from the automatic teller"

What to Test?

- Possible metrics to measure the “goodness” of an interface:
  - Time to learn: How long to learn commands relevant to a task
  - Speed of performance: How long does it take to carry out benchmark tests
  - Rate of errors by users: How many and what kinds of errors users make using the system.
  - Retention over time: How well do users maintain knowledge over time
  - Subjective satisfaction: How much did users like various aspects of the system

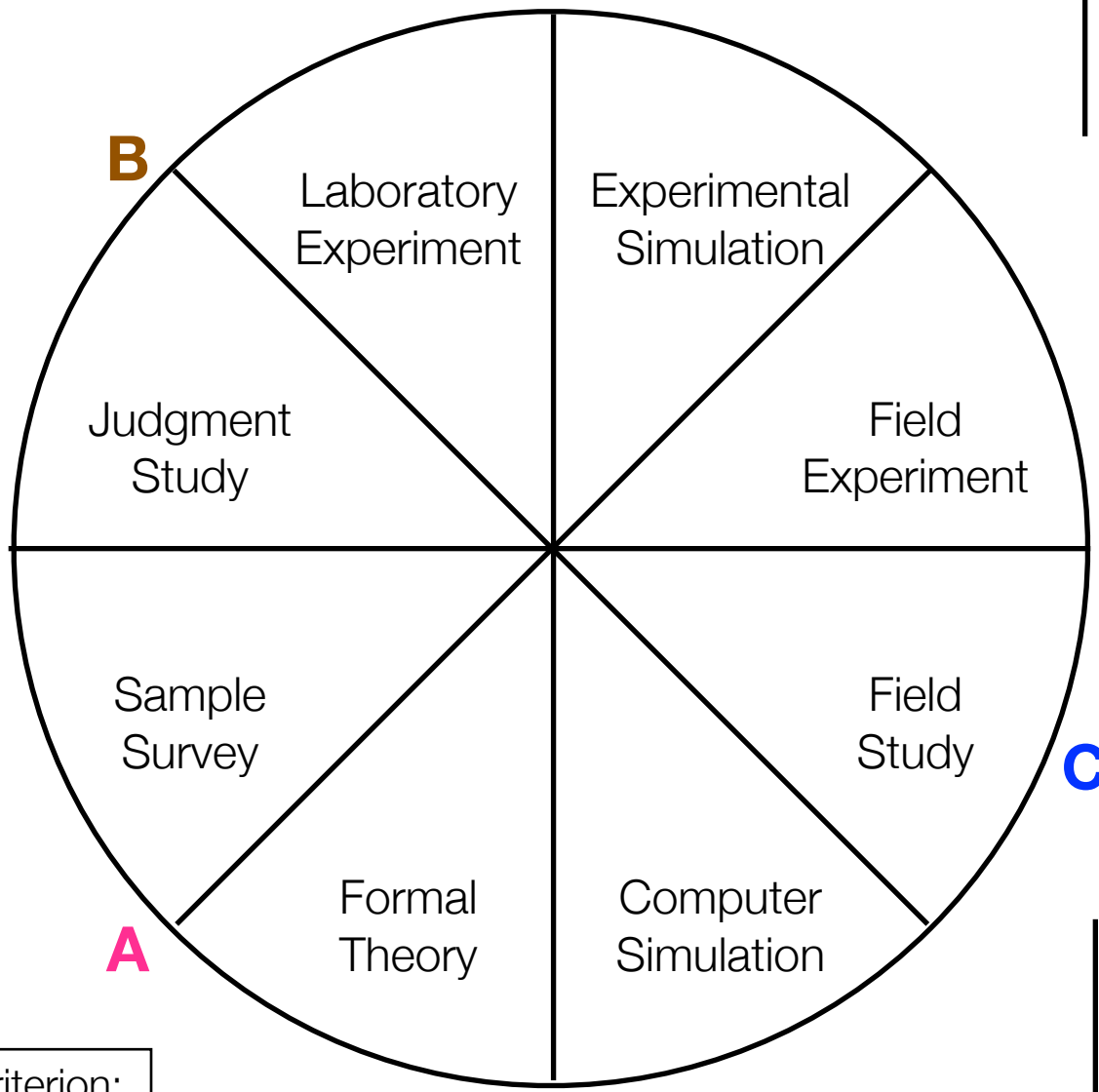
Ways to Test?



Maxima for each criterion:
A = Generalizability
B = Precision
C = Realism

McGrath et al, 1994

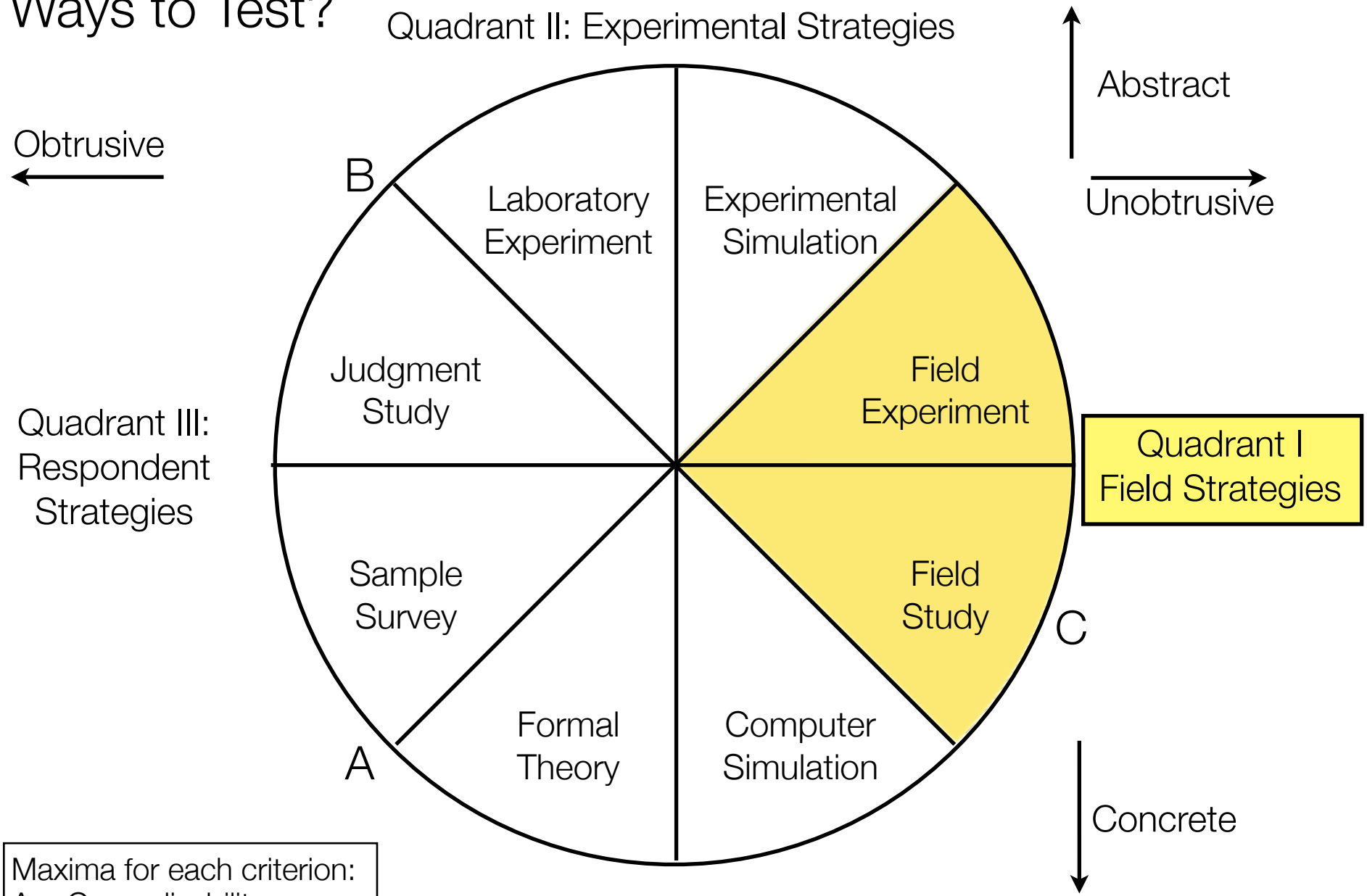
Ways to Test?



Maxima for each criterion:
A = Generalizability
B = Precision
C = Realism

McGrath et al, 1994

Ways to Test?



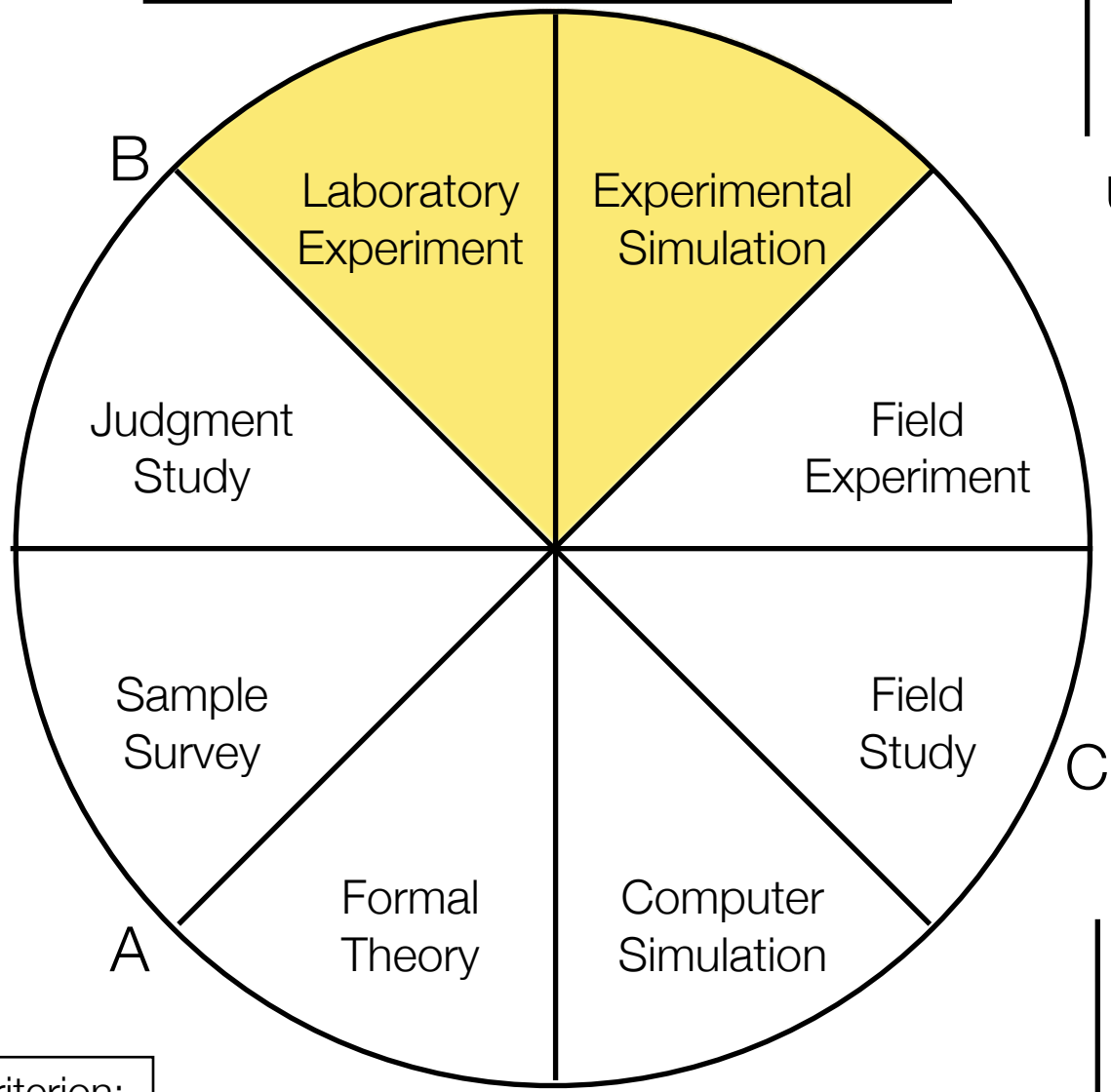
Maxima for each criterion:
A = Generalizability
B = Precision
C = Realism

McGrath et al, 1994

Field Studies

- Observing systems in use “in the wild”
- Advantages:
  - Natural Environment
  - Context Retained
- Disadvantages:
  - Distraction, noise
  - Hard to arrange and do, time consuming
- Appropriate when:
  - Context is crucial
  - Longitudinal Studies

Ways to Test?



Maxima for each criterion:
A = Generalizability
B = Precision
C = Realism

McGrath et al, 1994

Laboratory Experiment

- Observe people using systems in simulated settings
  - People brought in to artificial setting that simulates aspects of real world setting
  - Given specific tasks to do
  - Observations / measures made as people do their tasks
  - Look for problem areas / successes
  - Good for uncovering ‘big effects’

Laboratory Experiment Concerns

- Is the test result relevant to the usability of real products in real use outside of lab?
- Problems
  - Non-typical users tested
  - Non-typical tasks
  - Different physical environment
  - Different social context
    - Motivation towards experimenter vs motivation towards boss
- Partial Solution
  - Use real users
  - Task-centered system design tasks
  - Environment similar to real situation

Stupid Tests Give Useless Results
(Or: How NOT to test)

PC World's Techlog

News, opinion, and links from Editor in Chief Harry McCracken.

A Not-Very-Useful iPhone Keyboard Study

SLASHDO DIGG TI DEL.ICIO NEWSVI

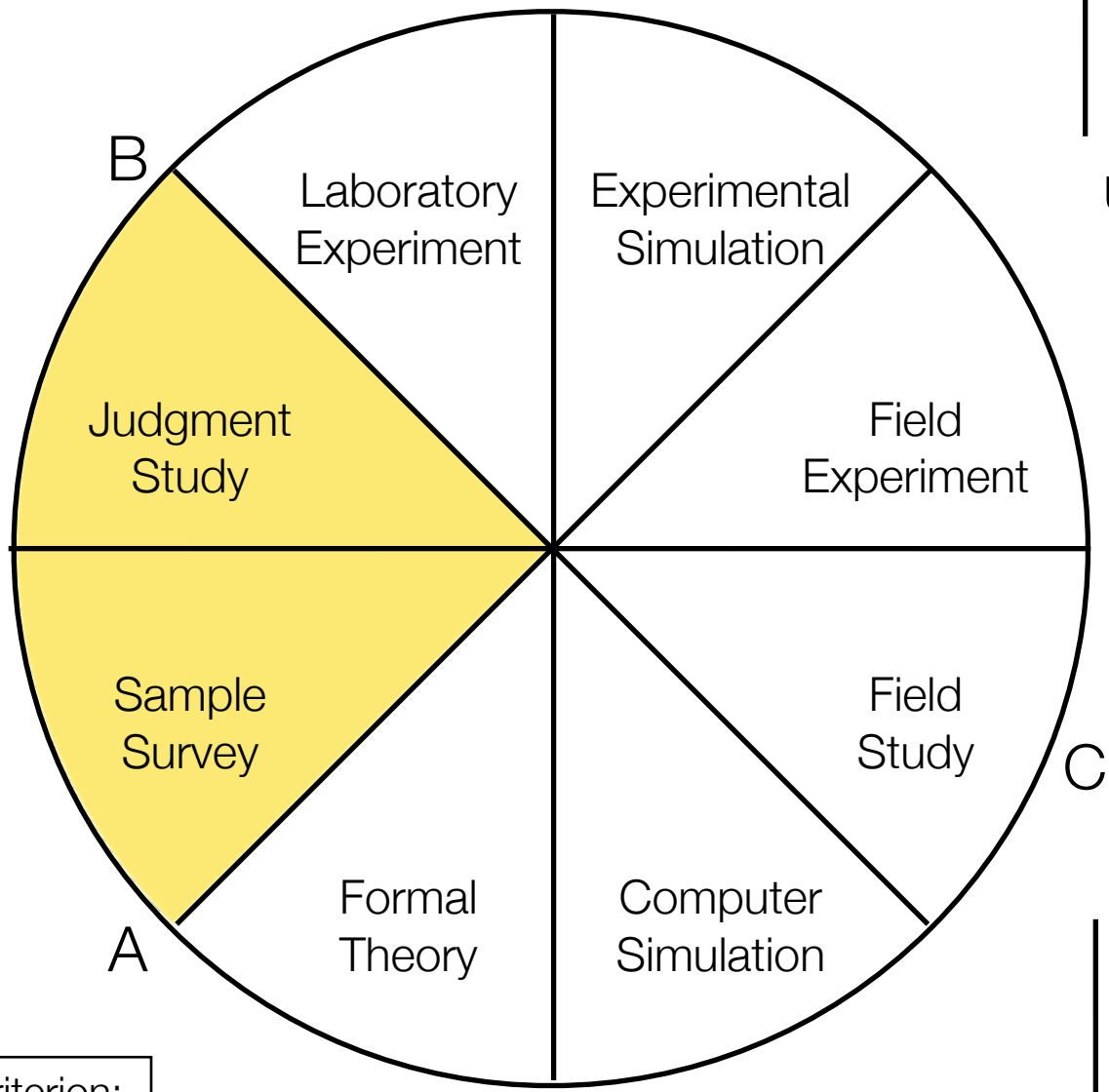
Research firm User Centric has released a study that tries to gauge how effective the iPhone's unusual on-screen keyboard is. The goal is certainly a noble one, but I can't say that the survey's approach results in data that makes much sense.

User Centric brought in twenty owners of other phones--half who had ones with QWERTY keyboards, and half who had ordinary numeric phone keypads. None were familiar with the iPhone. The research involved having the test subjects enter six sample text messages with the phones they already had, and six with an iPhone.

Logical end result: These iPhone newbies took twice as long to enter text with an iPhone as they did with their own phones, and made lots more typos.

Are we supposed to be surprised?

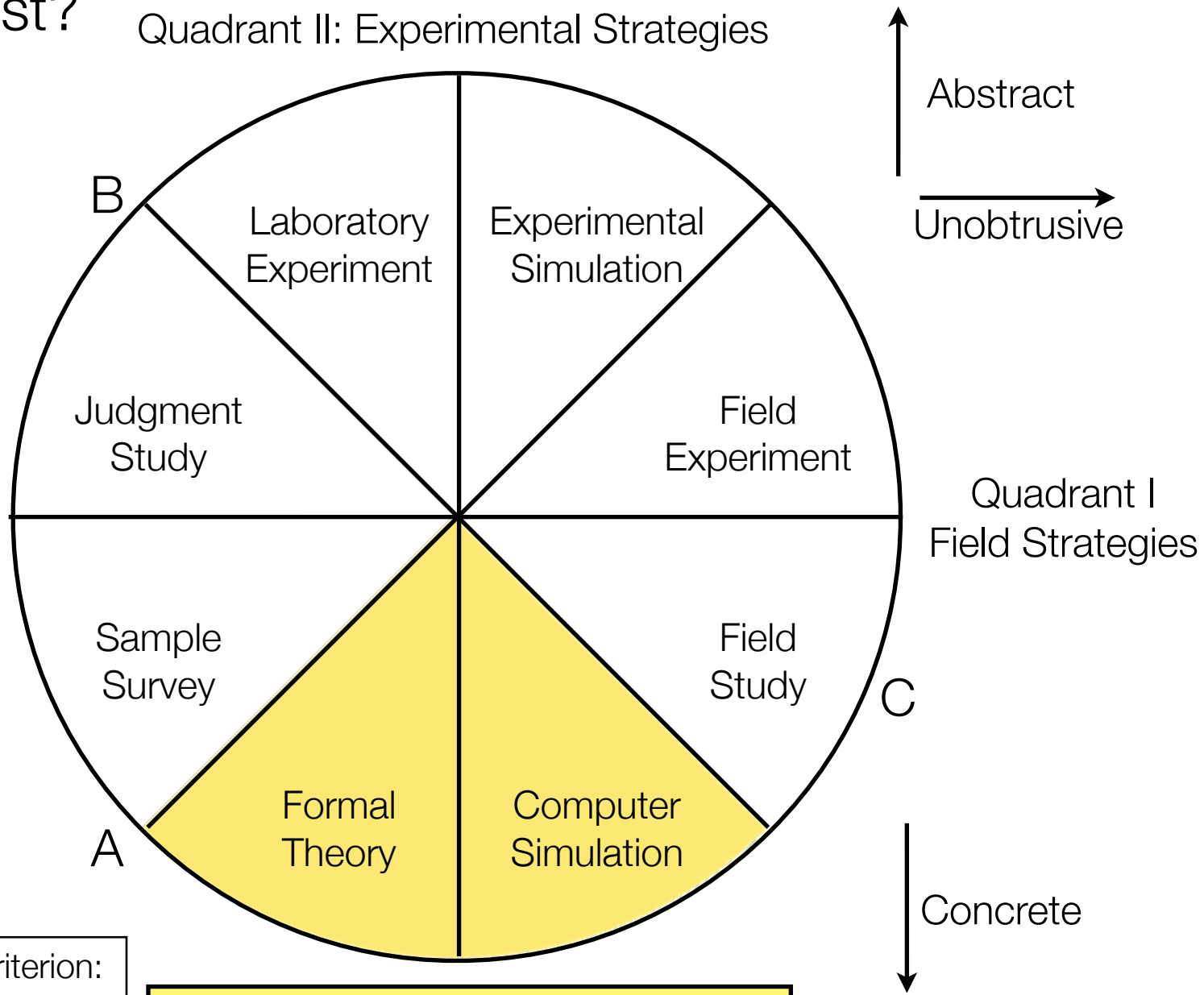
Ways to Test?



Maxima for each criterion:
A = Generalizability
B = Precision
C = Realism

McGrath et al, 1994

Ways to Test?



Maxima for each criterion:
A = Generalizability
B = Precision
C = Realism

McGrath et al, 1994

What is it?

- Quantitative evaluations test whether a given hypotheses are true.
  - E.g. Whether users remember Icon A more easily than Icon B
  - Require a clearly defined hypothesis and objective measurements.
- But there are some things that quantitative evaluations cannot measure.
  - E.g. Whether users can find information easily on a particular webpage.
  - Users' perceptions may differ from objective measurements

Measurements are always correct…

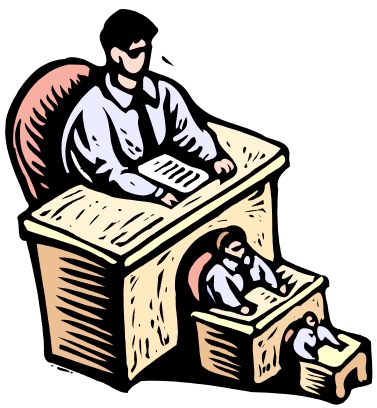
- … or are they?
- In a study at Apple [Tognazzini, 1992], users were asked to do the same task using a mouse and keyboard shortcuts.
  - Timing measurements showed that each and every user was able to perform the task significantly quicker using the mouse (on average, about 50% faster)
  - Interestingly, each and every user reported that they performed the task much faster using the *keyboard*.
- Moral: Don't trust users' opinions, but also, objective measurements don't always reflect users' perceptions and preferences.

Discount Usability Evaluation

- Many formal methods of usability evaluation are expensive and time-consuming
- *Discount usability evaluation* focuses on getting the most "bang for the buck"
- Inspection
- Extracting the conceptual model
- Direct observation
  - Think-aloud
  - Constructive interaction
- Query techniques (interviews and questionnaires)
- Continuous evaluation (user feedback and field studies)

Inspection

- Designer tries the system (or prototype)
  - Does the system “feel right”?
  - Benefits
    - Can catch some major problems in early versions
  - Problems
    - Not reliable as completely subjective
    - Not valid as introspector is a non-typical user
    - Intuitions and introspection are often wrong
- Inspection methods help
  - Task centered walkthroughs
  - Heuristic evaluation



Conceptual model extraction

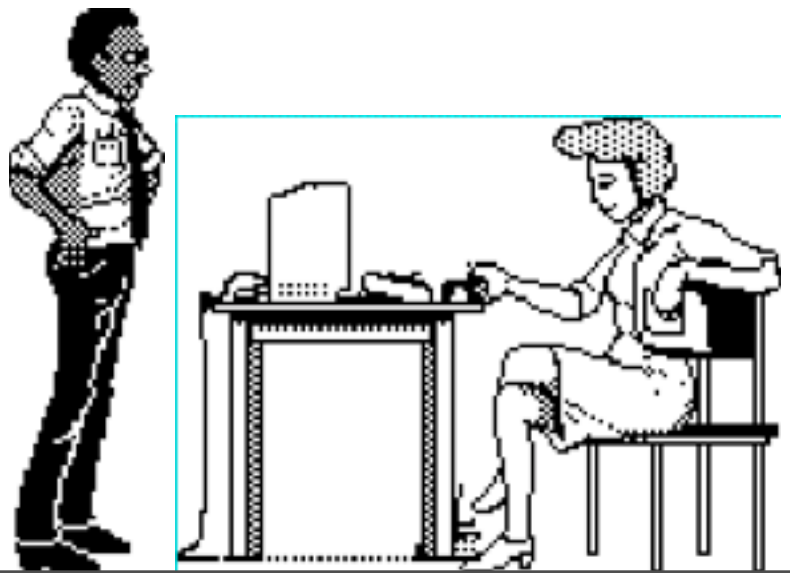
- How do we do it?
  - Show the user static images of the prototype or screens during use
  - Ask the user to explain
    - The function of each screen element
    - How they would perform a particular task
- What are we trying to acquire?
  - **Initial conceptual model**
    - How person perceives a screen the very first time it is viewed
  - **Formative conceptual model**
    - How person perceives a screen after its been used for a while
- Why do we want to do this?
  - Good for eliciting people's understanding before & after use
  - Poor for examining system exploration and learning

Direct observations

- Evaluator observes users interacting with system
  - In lab:
    - User asked to complete a set of pre-determined tasks
  - In field:
    - User goes through normal duties
- Value
  - Excellent at identifying gross design/interface problems
  - Validity depends on how controlled/contrived the situation is

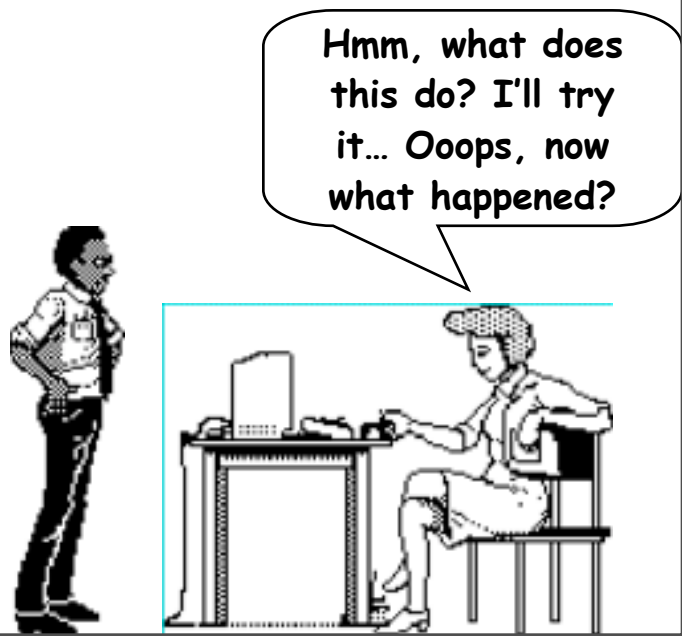
Simple observation method

- User is given the task
- Evaluator just watches the user
- Problem
  - Does not give insight into the user's decision process or Attitude.



Think aloud method

- Users speak their thoughts while doing the task
  - What they are trying to do
  - What they are reading
  - Questions that come up in their mind
  - Things they find confusing
  - Why they took an action
  - How they interpret what the system did



Think Aloud Method

- Facilitator's Role: try to facilitate spontaneous comments from the user
- Should user stop talking aloud, encourage flowing commentary with *unbiased* prompts:

- Good:
  - Non-committal "uh huh"
  - "Can you say more?"
  - "Please tell us what you are doing now?"
  - "I can't hear what you are saying"
  - "What are you thinking right now?"
- Bad:
  - "Why did you do that?"
  - "Why didn't you click here?"
  - "What are you trying to decide between?"

Why can't we ask why?

- Problem: Tie together two strings hanging from ceiling, too far apart.
- Solution: tie weight to one string, set it swinging, grab other string, wait for first one to come within reach.
- Scenario: When the facilitator "accidentally" brushed against one string, people were more likely to find solution. However, they did *not* say that the facilitator's moving the string gave them the idea.

Maier, 1931

Why can't we ask why?

- In market survey, most people preferred buying the rightmost pair of three *identical* pairs of underwear. When asked why, people made up plausible (but wrong) reasons.

- The real reason was that there is a natural bias towards last of a number of closely matched alternatives.

Nisbett & Wilson 1977

Think-Aloud Method

- Illustration of why a think-aloud method is useful (Louis and Rieman, 1993)
  - Menu-based administrative system for law offices
  - System messages used the word “parameter” a lot.
  - Users were reading “perimeter” when they saw “parameter”.
  - Hard to detect such problems just by watching people’s mistakes.

Pros and Cons of Thinking Aloud

++ Finds many usability problems

++ Gives insight into what the user is thinking.

- Finds *why* the problems occur

+ Requires little facilitator expertise

+ Generates colorful quotes for the report :-)

-- Slows users down by about 17% (Ericsson and Simon 1993)

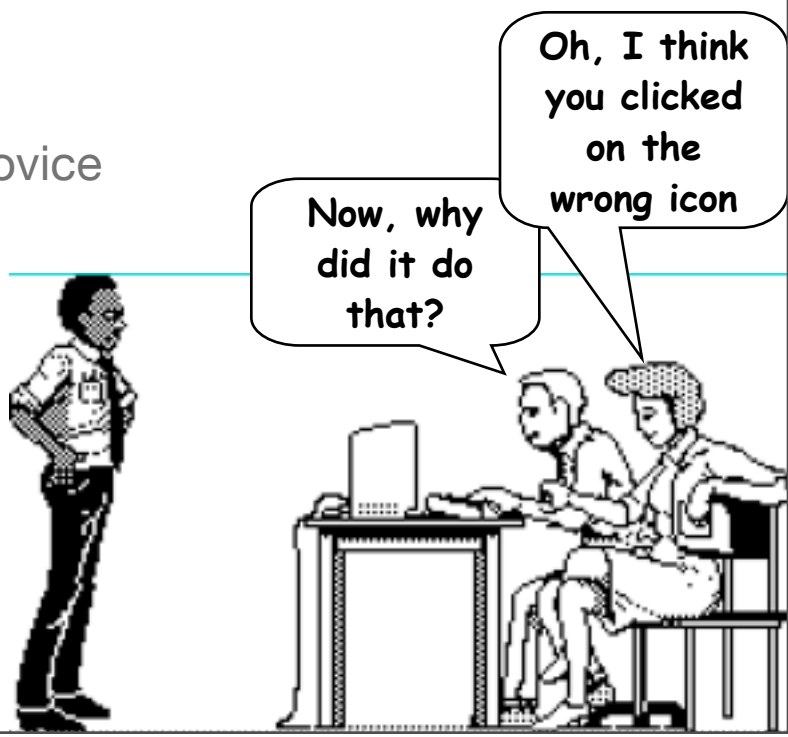
- May alter the way users do the task

- Cannot provide performance data

- Still, most-used evaluation method in industry.

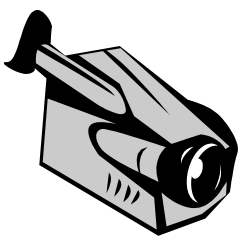
Constructive Interaction Method

- Two people work together on a task
  - Monitor their normal conversations
  - Removes awkwardness of think-aloud
  - Validity issue: would system normally be used by two people together?
- Co-discovery learning
  - Use semi-knowledgeable “coach” and novice
  - Only novice uses the interface
    - Novice ask questions
    - Coach responds
  - Gives insights into two user groups



Recording observations

- How do we record user actions for later analysis?
  - Otherwise risk forgetting, missing, or misinterpreting events
  - Paper and pencil
    - Primitive but cheap
    - Observer records events, comments, and interpretations
    - Hard to get detail (writing is slow)
    - 2nd observer helps…
  - Audio recording
    - Good for recording think aloud talk
    - Hard to tie into on-screen user actions
    - Best to combine with screen capture
  - Video recording
    - Can see and hear what a user is doing
    - One camera for screen, rear view mirror useful…
    - Initially intrusive



Interviews

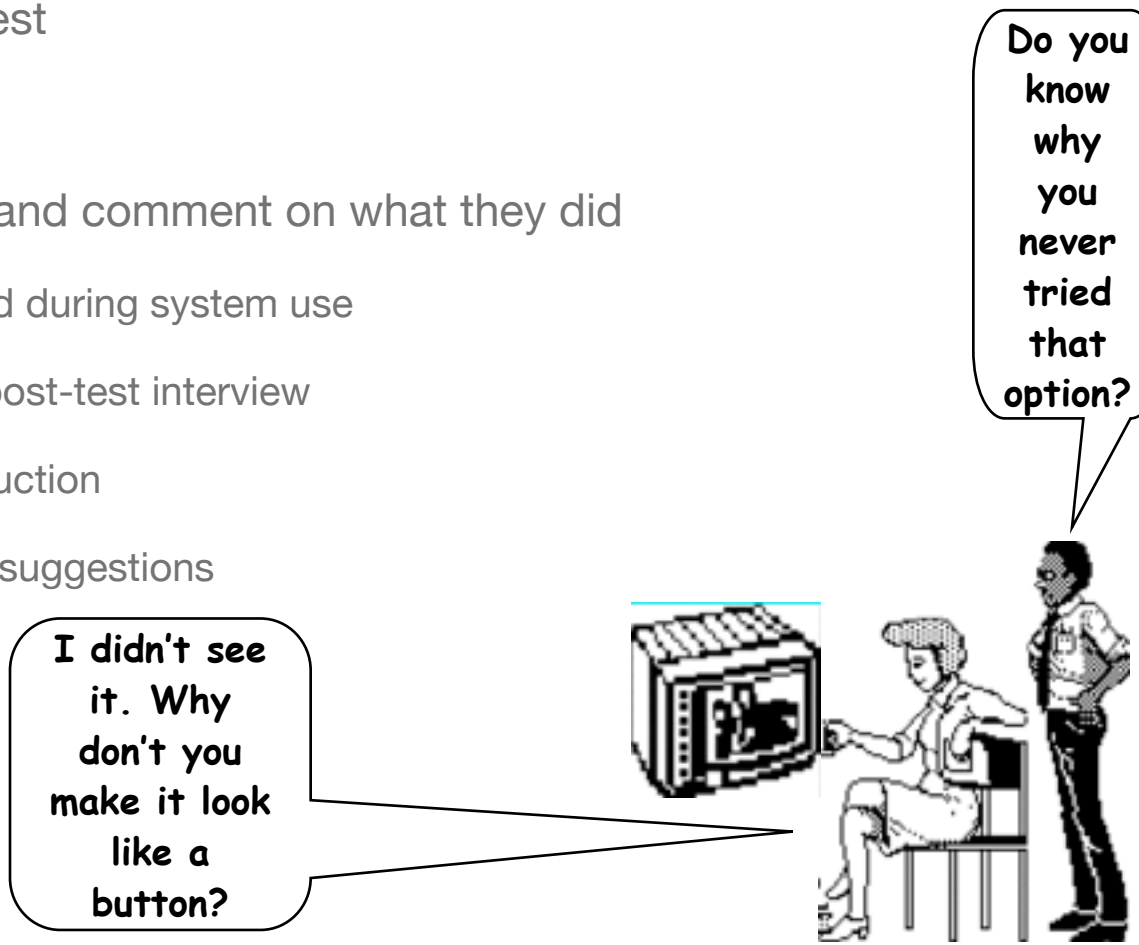
- Good for pursuing specific issues
  - Vary questions to suit the context
  - Probe more deeply on interesting issues as they arise
  - Good for exploratory studies via open-ended questioning
  - Often leads to specific constructive suggestions
- Problems:
  - Accounts are subjective
  - Time consuming
  - Evaluator can easily bias the interview
  - Prone to rationalization of events/thoughts by user
    - User's reconstruction may be wrong

How to Interview

- Plan a set of central questions
  - A few good questions gets things started
    - Avoid leading questions
  - Focuses the interview
  - Could be based on results of user observations
- Let user responses lead follow-up questions
  - Follow interesting leads *vs* bulldozing through question list

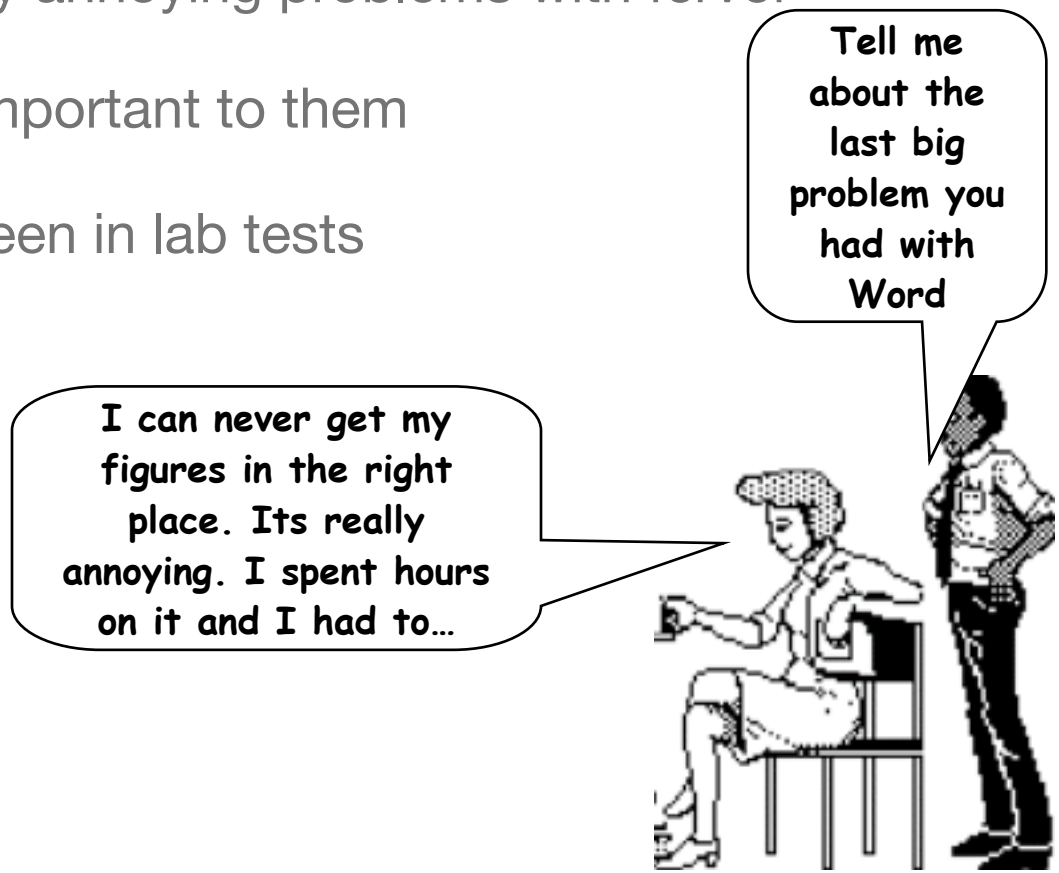
Retrospective testing interviews

- Post-observation interview to
  - Perform an observational test
  - Create a video record of it
  - Have users view the video and comment on what they did
    - Clarify events that occurred during system use
    - Excellent for grounding a post-test interview
    - Avoids erroneous reconstruction
    - Users often offer concrete suggestions



Critical incidence interviews

- People talk about incidents that stood out
  - Usually discuss extremely annoying problems with fervor
  - Not representative, but important to them
  - Often raises issues not seen in lab tests



Questionnaires and Surveys

- Questionnaires / Surveys
  - Preparation “expensive,” but administration cheap
    - Can reach a wide subject group (e.g. mail)
  - Does not require presence of evaluator
  - Results can be quantified
- But results are only as good as the questions asked
- Preparation of good questionnaires and surveys is a discipline of its own!

Styles of Questions

- Open-ended questions
  - Ask for unprompted opinions
  - Good for general subjective information, but difficult to analyze rigorously

    E.g. Can you suggest any improvements to the interfaces?

Styles of Questions

- Closed questions
  - restrict respondent's responses by supplying alternative answers
  - can be easily analyzed
  - watch out for hard to interpret responses!
    - alternative answers should be very specific

Do you use computers at work:

O often O sometimes O rarely

vs

In your typical work day, do you use computers:

O over 4 hrs a day

O between 2 and 4 hrs daily

O between 1 and 2 hrs daily

O less than 1 hr a day

Styles of Questions

- Scalar
  - ask user to judge a specific statement on a numeric scale
  - scale usually corresponds with agreement or disagreement with a statement

Characters on the computer screen are:				
Hard to read				Easy to read
1	2	3	4	5

  - Five and seven point scales give users a midpoint to “sit” on. Six-point scales force users to choose one way or another.
  - > 7 points: not much distinction between points.

Styles of Questions

- Semantic Differentials
  - Sliding scale between opposing pairs of adjectives (5 or 7 best)

Circle the number most closely matching your feelings about the interface

Simple	3 2 1 0 1 2 3	Complex
Professional	3 2 1 0 1 2 3	Unprofessional
Reliable	3 2 1 0 1 2 3	Unreliable
Attractive	3 2 1 0 1 2 3	Unattractive

Styles of Questions

- Multi-choice
  - respondent offered a choice of explicit responses

How do you most often get help with the system? (tick one)

O on-line manual

O paper manual

O ask a colleague

Which types of software have you used? (tick all that apply)

O word processor

O data base

O spreadsheet

O compiler

Styles of Questions

- Ranked
  - respondent places an ordering on items in a list
  - useful to indicate a user's preferences
  - forced choice

  Rank the usefulness of these methods of issuing a command

  (1 most useful, 2 next most useful..., 0 if not used

  __2__ command line

  __1__ menu selection

  __3__ control key accelerator

Continuous Evaluation

- Monitor systems in actual use
  - Usually late stages of development
    - ie beta releases, delivered system
  - Fix problems in next release
- User feedback via gripe lines
  - Users can provide feedback to designers while using the system
    - Help desks
    - Bulletin boards
    - Email
    - Built-in gripe facility
  - Best combined with trouble-shooting facility
    - Users always get a response (solution?) to their gripes

Continuous evaluation

- Case/field studies
  - Careful study of “system usage” at the site
  - Good for seeing “real life” use
  - External observer monitors behavior
  - Site visits

Conducting a Usability Evaluation Test

1. Develop the test plan
2. Select and acquire participants
3. Prepare test materials
4. Run a pilot test
5. Conduct the real test
6. Analysis and final report

Test Plan

- Main section headings for any test plan:
  - Purpose
  - Problem statement
  - User profile
  - Method (test design)
  - Task list
  - Test environment
  - Data to be collected
  - Content of report.

Task List

- Prioritize tasks by frequency and criticality
- Choose most frequent and critical to test
- Make a task list for test team internal use only
- For each task:
  - Define any prerequisites
  - Define successful completion criteria
  - Specify maximum time to complete each task (after which help may be given)
  - Define what constitutes an error

Sample Task List

Task	Description	Criteria
1	Open the Safari Application from Windows Explorer	Prereq: Computer booted up with no other apps running. Completed: Safari window open. Maxtime: 1 minute
2	Navigate to the PolyU Homepage	Prereq: Safari window open at a blank home page. Completed: PolyU homepage displayed in Safari Maxtime: 1 minute
3	Locate swimming pool opening hours	Prereq: Safari window showing PolyU homepage Completed: Swimming pool opening hours displayed on the Safari window Maxtime: 2 minutes
4	Etc…	

Selecting and Acquiring Participants

- Representative users based on user profile.
- Classify into one or more categories
- At least 4 users per category
- Acquire test users via employment agency, students, existing customers, internal personnel.
- Maintain a database of potential test users.
- Screening questionnaire (ensure users fit profile)

Sample User Profile

Characteristic	Range	Frequency Distribution
Age	18-55	70% 25-45 30% other
Sex		90% male
Education level	Secondary school College	20% 80%
Education major	Computer science Other	50% 50%
General computer experience	0-5 years	10% < 1 year 30% 1-2 years 40% 3-4 years 20% > 4 years
OS experience	Windows Unix Mac Other none	60% 80% 20% 10% 5%
Etc		

Test Materials

- Orientation Script
- Background Questionnaire
- Nondisclosure and Consent form
- Training script (if any)
- Task Scenarios
- Data Collection Forms
- Debriefing Topic Guide
- Post Test Questionaire
- Checklist

Orientation Script

- Introduce yourself and any observers by name, but do not tell them your title or job descriptions.
- Explain purpose of test (to collect input to help produce better products)
- Acknowledge software is new and may have problems.
- Do not mention any association you have with product. If you are not associated with product, mention it.
- Explain any recording (reassure confidentiality)
- Say user may stop at any time.
- Say user may ask questions at any time, but they may not be answered until the test is complete.
- Invite questions.

Sample Orientation Script

Hi, my name is Grace. I'll be working with you in today's session. [John and Tom here will be observing.]
We are here to test a new product, the PolyU webpage, and we'd like your help.
I will ask you to perform some typical tasks with the system. Do your best, but don't be overly concerned with results -- the system is being tested, not your performance.
Since the system is a prototype, there are certainly numerous rough edges and bugs and things may not work exactly as you expect.
[I am an independent researcher hired to conduct this study, and have no affiliation with the system whatsoever.] My only role here today is to discover the flaws and disadvantages of this new system from your perspective. Don't act or say things based on what you think I might want to see or hear. I need to know what you really think.
Please do ask questions at any time, but I may only answer them at the end of the session.
While you are working, I will be taking some notes and timings. We will also be videotaping the session for the benefit of people who couldn't be here today.
If you feel uncomfortable, you may stop the test at any time.
Do you have any questions?
If not, then let's begin by filling out a short background questionnaire and having you sign the nondisclosure agreement and consent to tape form.

Background Questionnaire

- This is needed so you can better understand the user's performance during the test.
- Fill in the background questionnaire, asking the test user the questions.
- If possible, all observers should have a copy before the test starts.
  - Administration data: date, test number, user number or id.
  - General Data: age (range), sex, educational level, …
  - Computer experience: total time, frequency of use, types of software, have used a GUI before, …
  - Application experience: total time, frequency of use, brand.

Sample Combined NDW and Consent to Tape Form

Thank you for participating in our product research. Please be aware that confidential information will be disclosed to you and that it is imperative that you do not reveal information that you may learn during the course of your participation. In addition, your session will be videotaped, to allow staff members who are not present to observe your session and benefit from your feedback. Please read the statements below and sign where indicated. Thank you.

I agree that I will disclose no information about the product research conducted by ABC Company Inc., or about the specifications, drawings, models, or operations of any machine, devices, or systems encountered.
I understand that video and audio recordings will be made of my session, I grant ABC Company Inc. permission to use these recordings for the purposes mentioned above, and waive my right to review or inspect the recordings prior to their dissemination and distribution.

Please print name: _____
Signature: _____
Date: _____

Training Script

- Exact written description of prior training:
  - Demonstration of GUI
  - Demonstration of special interaction styles: mouse keys, drag-and-drop, etc.
  - Walk-through of sample task
  - Demo of how to think aloud (for Thinking Aloud style tests)

Task Scenarios

- The task descriptions given to the test users
  - Simple introductory first task (early success)
  - Realistic scenarios in typical order
  - If sequential ordering not crucial, randomize presentation order to counterbalance learning effect
  - Each task scenario on a separate sheet -- do NOT hand user all the tasks at once!
  - Do not guide participants through the task (describe the goal, not the steps).

Data Collection Forms

- Define abbreviations for expected events
- Use ? to signal an event worth probing during debriefing
- Paper or electronic forms (or special software)

Code	Event
B	Begin Task
E	End Task
P	Prompted by test facilitator
T	Exceeded maximum time
X	Incorrect action
"…"	Verbatim user comment
M	Reading the manual
H	Accessing online help
?	Probe this during debriefing
C	Comment by facilitator
*	Very important action

Sample collection form

Test: *Edit HTML document*		User No: *3*
Date: *23-4-01*	Time: *11:50*	Page *1* of *3*

Task	Time Taken	Observations
1	*04:25*	*X Opened wrong file. Found mistake.* *X Opened wrong file again.* *Self-corrected due to error message.*
	06:00	*P*
	07:00	*T*
2	*11:30*	*" I wish it were always that easy!"*
	15:20	*? Very long hestitation, then correct action*
	16:15	*E*

Debriefing Guide

- Let user speak thoughts first: “So, how was it?”
- Top-down: probe high-level issues from topic guide first, then more detailed questioning about the task.
- Probe specific issues arising from test notes.
- Review answers to post-test questionnaire
- Accept questions from any observers.

Post Test Questionnaire

- Collect feelings, opinions, suggestions (hard to observe in other ways), for example:
  - Does the interface organization match real world tasks?
  - Too much or too little information on screens?
  - Similar information consistently placed?
  - Problems with navigation?
  - Computer jargon?
  - Appropriate use of color?

Chronological Order of Evaluation Test

- [ ] Scan your customized checklist
- [ ] Everything ready in test room
- [ ] Prepare yourself mentally
- [ ] Establish protocol for observers
- [ ] Greet the participant
- [ ] Read the orientation script and set the stage
- [ ] Have participant sign consent forms
- [ ] Administer background questionnaire
- [ ] Move to testing area
- [ ] Provide any prior training
- [ ] Provide demo of thinking aloud
- [ ] Record start time
- [ ] Distribute or read written task scenarios to participant one at a time
- [ ] Observe, note interesting and critical events
- [ ] Debriefing interview
- [ ] Administer post test questionnaire
- [ ] Thank participant, provide any remuneration, show participant out.
- [ ] Organize data sheets and notes
- [ ] Summarize thoughts about test
- [ ] Prepare for next participant

Pilot Test

- A pilot test must *always* be performed prior to any user testing.
- Look for:
  - Ambiguous Instructions
  - Unrealistic Time Estimates
  - Ambiguous Criteria
  - Misleadinq Questionnaire Questions

The Real Thing

- During the test:
  - Facilitator handles all interaction with participant (observers are completely quiet)
  - User should *never* be prompted or biased during the test (be careful of body language!)
  - User should only be assisted in cases of severe difficulties (a note should be taken)
  - Debriefing Interview or questionnaire should always be included.

Analysis and Final Report

- Compile and Summarize Data:
  - Mean, median, range and standard deviation of completion times
  - Percentage of users performing successfully
  - Bar chart of preference scores
  - Etc.
- Analyse data:
  - Identify errors and difficulties
  - Diagnose source of each error
  - Prioritize problems by severity or criticality

Final Report

- Title Page
- Description of Test Environment
  - Hardware, software version, test room, dates
- Executive Summary
  - Concise summary of major findings (a few pages)
- Description of test
  - Updated test plan, method, training, tasks.
- Test Person Data
  - Tabular summary of age, occupation, experience
- Results
  - Tabular and graphical summaries of times taken, number of errors made, questionnaire responses, etc.
  - Discussion and analysis, amusing quotes.

Final Report (cont'd)

- List of Positive Findings
- List of Recommendations
  - List of problems discovered, in descending order of severity, and recommended improvements. For each recommendation:
    - Diagnose why the problem occurred.
    - Illustrate with a screen shot
    - Rate its severity
    - Indicate exactly how many testers experienced this problem.
    - Include a reference to timestamps on video tape.
    - Possibly include an appropriate user quotation
    - Describe suggested improvement.
- Appendices (raw data and tables)
  - Background questionnaires, consent forms, orientation script, data collection forms, tapes, transcripts, etc.

Sample Recommendation

R12. Sort Order Panel (Severity: 3.2)

- Problem: Users had problems understanding the sort order panel. In particular, plus and minus icons used for ascending and descending order are non-intuitive.
- Reference: TP1, 00:09:17
  - “What does this plus mean?”
- Recommendation: Redesign the icon (for example, use a sloping ramp)

The IBM Real Phone Study