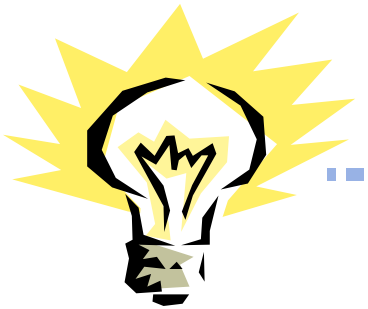
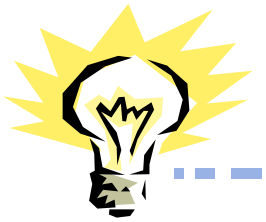


Empirical Evaluation: Just an Overview and Reminder

Assessing usability
(with users)



Agenda

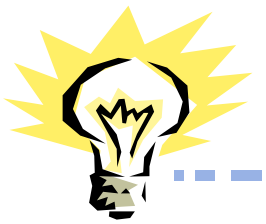
- Evaluation overview
- Analyzing & interpreting results
- Using the results in your design

Why Evaluate?

Recall:

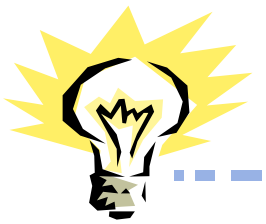
- Users and their tasks were identified
- Needs and requirements were specified
- Interface was designed, prototype built…

- *But is it any good? Does the system support the users in their tasks? Is it better than what was there before (if anything)?*



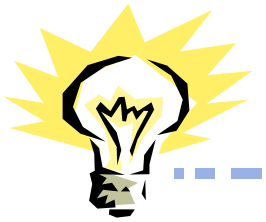
D3

- A key part of D3 is making sure your **prototype**, **evaluation plan**, and **usability specifications** align
- Your prototype should be designed to support your evaluation.
- Your evaluation plan should support determining adherence to your selected usability specifications
- Usability specifications should be selected based on the overall project goals and requirements
- Be thinking about this



Types of Evaluation

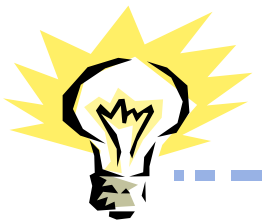
- Interpretive and Predictive (a reminder)
  - Heuristic evaluation, cognitive walkthroughs, ethnography…
- Summative vs. Formative
  - What were they, again?
- Focused on summative evaluation at present



Now With Users Involved

- Interpretive (naturalistic) vs. Empirical:

- Naturalistic
  - In realistic setting, usually includes some detached observation, careful study of users
- Empirical
  - People use system, manipulate independent variables and observe dependent ones

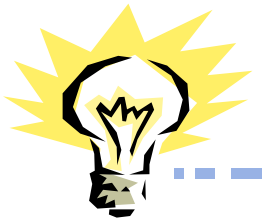


Why Gather Data?

- Design the experiment to collect the data to test the hypotheses to evaluate the interface to refine the design

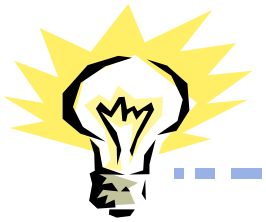
- Information gathered can be:
  objective or *subjective*
- Information also can be:
  qualitative or *quantitative*

Which are tougher to measure?



What kind of data do you need?

- Performance Metrics
  - Error counts
  - Success/fail rate
  - Task times
  - Number of clicks
- Behavioral and physiological metrics
  - Eye tracking
  - Physiological measures
- Self-report metrics
  - Survey response scores



Conducting an Evaluation

- Determine the tasks
- Determine the performance measures
- Develop the procedures
- (IRB approval)
- Recruit participants
- Collect the data
- Inspect & analyze the data
- Draw conclusions to resolve design problems
- Redesign and implement the revised interface

Writing Tasks

- Representative tasks - add breadth, can help understand process
- Benchmark tasks - gather quantitative data

- Issues:
  - Lab testing vs. field testing
  - Validity - typical users; typical tasks; typical setting?
  - Run pilot versions to shake out the bugs

Writing Tasks

- Tasks should form a **representative sample** of the things a user might do, but should also be **targeted** to answer important questions about your design
- When choosing tasks, consider:
  - Common tasks
  - Critical tasks
  - Coverage of most major system functionality
  - Research questions, if you have them
  - Usability criteria, system goals, etc.

Writing Tasks

- Tasks should be **plausible**, but not too easy
- If you want to write a hard task, think of a realistic scenario that is hard
  - Example: 'buy 10 different kinds of forks, and have them shipped to 6 different addresses' ?

Writing Tasks

- Tasks should be **described in terms of the user's end goals** and motivations, not the system.
  - Providing brief context can facilitate this
    - Good: "Your computer is slowing down when you have more than a couple windows open. Purchase a stick of 8GB DDR memory that will work with your computer."
    - Bad: "Use the shopping widget to add a stick of 8GB DDR memory to the cart and complete purchase."

Writing Tasks

- Tasks must be **possible** to complete, with a definable **success/ fail conditions**
  - Success: a person reaches the desired screen, and they know it
  - Failure:
    - participant indicates they would like to give up (give them this option)
    - participant reaches the wrong screen and thinks they are at the correct screen
    - participant reaches the correct screen and thinks they are at the wrong screen

Writing Tasks

- Tasks should have a **specific end goal**

  Good: 'Buy 8GB of Corsair DDR memory'

  Bad: 'Look around for some memory you might want to buy'

- But be careful not to tip the participant off by using terms or phrasings that exist in the interface.
  - You don't want participants to be able to just recognize a term
- Tasks should also **allow exploration, information-seeking, and decision-making**
  - Tell them <u>what</u> to do, not <u>how</u> to do it
    - Don't be too prescriptive! Tasks can be high-level, as long as there is a definable endpoint
      - "Find a restaurant that looks appealing and order the meal you want"

Writing Tasks

- When possible, tasks should relate to the **desired outcomes** from a system, in addition to whether the system is usable
  - If your goal is to improve cyclist safety, can you evaluate that?
  - What about helping people use more re-usable cups?
  - Oftentimes this is too difficult, and we will only test whether a user is able to use the system, without knowing about whether it might cause desired outcomes

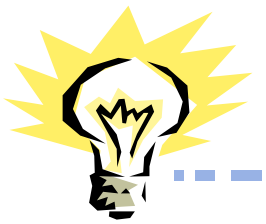
Writing Tasks

- Tasks should be **ordered in a realistic sequence**
  - You might start with a browsing task, followed by a selection task, followed by a purchasing task, followed by entering shipping information
    - It's about situating the user in a story and context that makes some sense

Writing Tasks

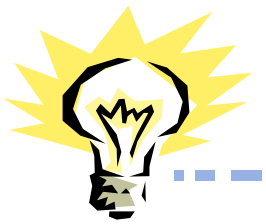
- Think about **what constitutes an error** in the context of your prototype
  - Participant clicks on an unnecessary button?
  - Participant moves the mouse over an unnecessary screen area?
  - Participant's eyes linger on the wrong page content?

- Try and anticipate and record in-task errors, in addition to a participant reaching the wrong endpoint or failing outright



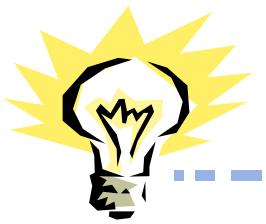
Defining Performance

- Depends on the task
- Specific, objective measures/metrics
- Examples:
  - Speed (reaction time, time to complete)
  - Accuracy (errors, hits/misses)
  - Production (number of files processed)
  - Score (number of points earned)
  - …others…?



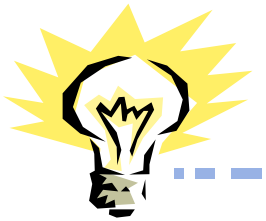
"Benchmark" Tasks

- Specific, clearly stated task for users to carry out
  - (don't make all tasks like this though)
- Can use these tasks to compare performance across versions
- Example: Email handler
  - "Find the message from Mary and reply with a response of 'Tuesday morning at 11' ."
- Users perform these under a variety of conditions and you measure performance



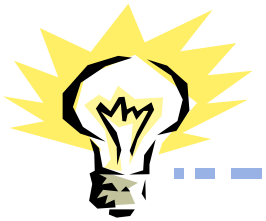
Empirical Evaluation Study Design

- Some evaluations will have the features of psychological experiment design:
  - Independent Variables
    - What you're studying, what you intentionally vary (e.g., interface feature, interaction device, selection technique)
  - Dependent Variables
    - Performance measures you record or examine (e.g., time, number of errors), in terms of how changes in the IVs affect them
  - Controlled Variables
    - Properties that are held constant (intentionally **not** varied)
  - Hypotheses: how do you predict the dependent variable (i.e., performance) will change depending on the independent variable(s)



Example

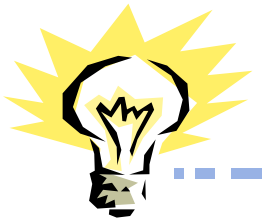
- Do people complete operations faster with a black-and-white display or a color one?
  - Independent - display type (color or b/w)
  - Dependent - time to complete task (minutes)
  - Controlled variables - same number of males and females in each group
  - Hypothesis: Time to complete the task will be shorter for users with color display
  - Note: Within/between design issues, next



Empirical Evaluation Study Design

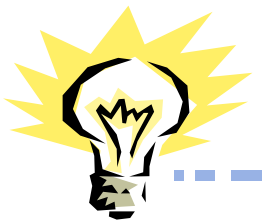
- Within Subjects Design
  - Every participant provides a score for all levels or conditions
    - More efficient, fewer participants needed
    - Greater statistical power
    - Need to avoid order effects

- Between Subjects Design
  - Each participant provides results for only one condition
  - Fewer order effects
    - Participant may learn from first condition
    - Fatigue may make second performance worse
  - Simpler design & analysis
  - Easier to recruit participants, shorter sessions
  - Less efficient



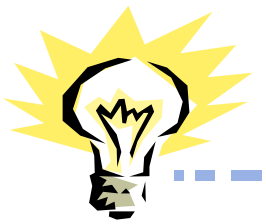
IRB, Participants, & Ethics

- Institutional Review Board (IRB)
  - http://www.osp.gatech.edu/compliance.htm
- Reviews all research involving human (or animal) participants
- Safeguarding the participants, and thereby the researcher and university
- Not a science review (i.e., not to assess your research ideas); only safety & ethics
- Complete Web-based forms, submit research summary, sample consent forms, etc.
- All experimenters must complete NIH online history/ethics course prior to submitting



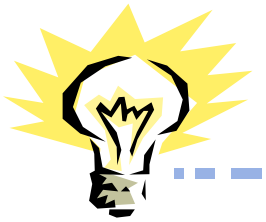
Recruiting Participants

- Various "subject pools"
  - Volunteers
  - Paid participants
  - Students (e.g., psych undergrads) for course credit
  - Friends, acquaintances, family, lab members
  - "Public space" participants - e.g., observing people walking through a museum
- Must fit user population (validity)
- Motivation is a big factor - not only $$ but also explaining the importance of the research
- Note: Ethics, IRB, Consent apply to *all* participants, including friends & "pilot subjects"



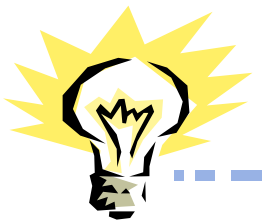
Ethics

- Testing can be arduous
- Each participant should consent to be in experiment (informal or formal)
  - Know what experiment involves, what to expect, what the potential risks are
- Must be able to stop without danger or penalty
- All participants to be treated with respect



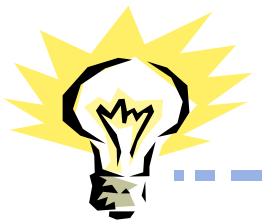
Consent

- Why important?
  - People can be sensitive about this process and issues
  - Errors will likely be made, participant may feel inadequate
  - May be mentally or physically strenuous
- What are the potential risks (there are always risks)?
  - Examples?
- "Vulnerable" populations need special care & consideration (& IRB review)
  - Children; disabled; pregnant; students (why?)



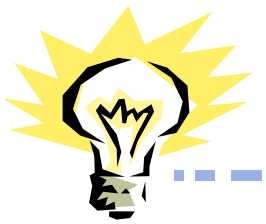
Before Study

- Be well-prepared so participant's time is not wasted
- Make sure they know you are testing software, not them
  - (Usability testing, not User testing)
- Maintain privacy
- Explain procedures without compromising results
- Can quit anytime
- Administer signed consent form



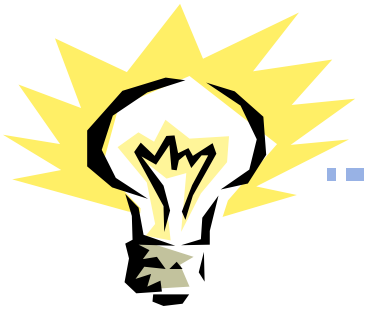
During Study

- Make sure participant is comfortable
- Session should not be too long
- Maintain relaxed atmosphere
- Never indicate displeasure or anger



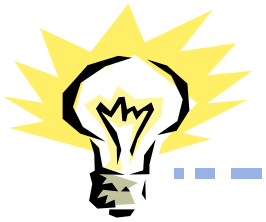
After Study

- State how session will help you improve system
- Show participant how to perform failed tasks
- Don't compromise privacy (never identify people, only show videos with explicit permission)
- Data to be stored anonymously, securely, and/or destroyed



Gathering Usability Data

Observing users & subjective data



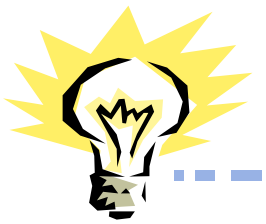
Directing Sessions

- Study design issues:
  - Are you in same room or not?
  - Single person session or pairs of people
  - Objective data -- stay detached

- In typical usability study, there will be a combination of procedure-following (list of tasks, etc.) , and more spontaneous interviewing by a moderator

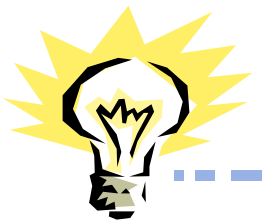
Moderator Tips

- Start with some easy rapport-building
- Then, first impression questions
- A good moderator will know when to intervene and ask a participant for more information



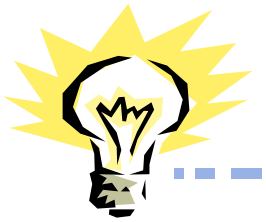
Moderator Tips

- **Probe for expectations-** before a user takes an action, ask them what they expect to happen. After they take an action, you can ask if it matched their expectations.
- **Ask for more information** if the participant is being vague
- **Investigate mistakes**
- **Probe nonverbal cues**
- However: **keep the interview task-centered**



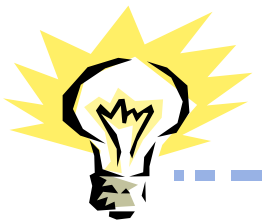
Moderator Tips

- → **Keep the participant focused on their own experience.**
  - Participants will try and think about the population in general, or hypotheticals
    - "I think that would be useful to someone"
    - "This is probably simple for most people but I just had trouble with it."
  - Remember that what you care about is **what the participant is experiencing, right in the present moment.**



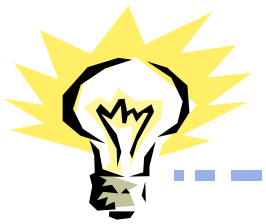
Moderator Tips

- **Attribution Theory:** Studies why people believe that they succeeded or failed--themselves or outside factors (gender, age differences)
- Explain how errors or failures are not participant's problem
- Instead, these are places where interface needs to be improved



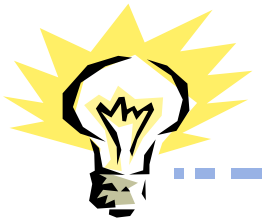
Moderator Tips

- Try not to give the participant positive or negative feedback on how ”well” they are doing
  - Minimize extrinsic performance feedback in the prototype
    - (no “success beep” when they find the goal)
- However, do show interest in their thoughts and experiences, and encourage them to share more detail
  - Ask for **clarification**, without asking **leading questions**
  - “So I think what I am hearing is that you would prefer the login page be a bit simpler. Is that correct?”



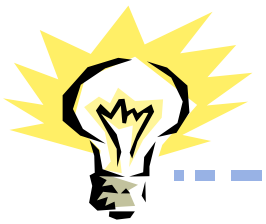
Moderator Tips

- If the user gets stuck on a task, or discouraged:
- You can ask:
  - "What are you trying to do..?"
  - "What made you think..?"
  - "How would you like to perform..?"
  - "What would make this easier to accomplish..?"
  - Maybe offer hints
- Ok to briefly explore solutions and design ideas
  - Participant is not a designer, but you can work with them to explore ways to address the problem



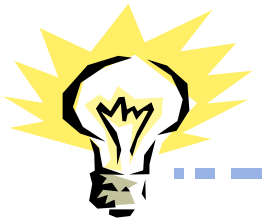
Location

- Sessions may be
  - In lab - Maybe a specially built usability lab
    - Easier to control
    - Can have user complete set of tasks
  - In field
    - Watch their everyday actions
    - More realistic
    - Harder to control other factors
- Either way, make sure the participant is comfortable, and also that the environment is as valid as possible



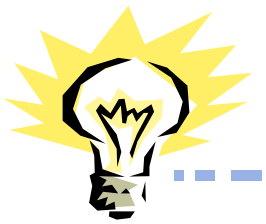
Observing Users

- One of the best ways to gather feedback about your interface
- Watch, listen and learn as a person interacts with your system
- Not as easy as you think…



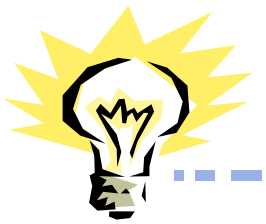
Observation

- Direct Observation
  - In same room
  - Can be intrusive
  - Users aware of your presence
  - Only see it one time
  - May use 1-way mirror to reduce intrusiveness
- Indirect Observation
  - Video recording
  - Reduces intrusiveness, but doesn't eliminate it
  - Cameras focused on screen, face & keyboard
  - Gives archival record, but can spend a lot of time reviewing it



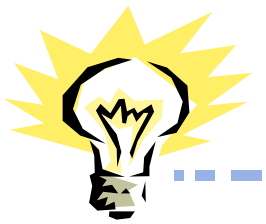
Challenge

- While observation of what users do is important, you don't know what's going on in their head
- In addition to observation, often utilize some form of *verbal protocol* where users describe their thoughts



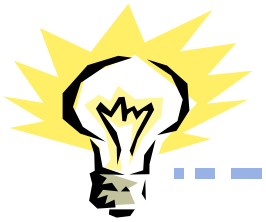
Verbal Protocol

- One technique: *Think-aloud*
  - User describes verbally what s/he is thinking and doing
    - What they believe is happening
    - Why they take an action
    - What they are trying to do
- Very widely used, useful technique
- Allows you to understand user's thought processes better
- Potential problems:
  - Can be awkward for participant
  - Thinking aloud can modify way user performs task



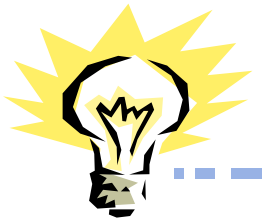
Post-Event Protocols

- What if thinking aloud during session will be too disruptive?
- Can use **post-event protocol** (also called retrospective think aloud)
  - User performs session, then watches video afterwards and describes what s/he was thinking
  - Sometimes difficult to recall
  - Opens up door of interpretation
  - With this method, you can still record data such as task times



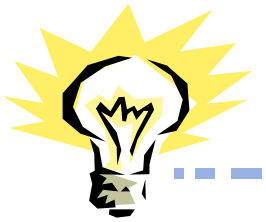
Collecting Data

- Note-taking
  - If you can manage, categorize errors, measure task times, etc. *during* the study
  - Remember to write down what they **do** (observation) not just what they say
- Video Recording
- Instrumenting the user/ interface
  - Eye tracking
  - Physiological measures
  - Cursor tracking, etc.
- Post-experiment questions and interviews



Collecting Data

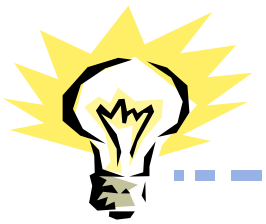
- Identifying errors can be difficult
- Qualitative techniques
  - Think-aloud - can be very helpful
  - Post-hoc verbal protocol - review video
  - Critical incident logging - positive & negative
  - Structured interviews - good questions
    - "What did you like best/least?"
    - "How would you change..?"



Capturing a Session

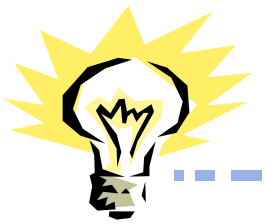
- 1. Paper & pencil
  - Can be slow
  - May miss things
  - Is definitely cheap and easy

		Task 1	Task 2	Task 3	…
Time	10:00		S		
	10:03		e	S	
	10:08			e	
	10:22				



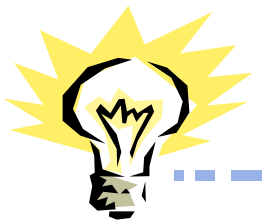
Capturing a Session

- 2. Recording (audio and/or video)
  - Good for talk-aloud
  - Hard to tie to interface
  - Multiple cameras probably needed
  - Good, rich record of session
  - Can be intrusive
  - Can be painful to transcribe and analyze



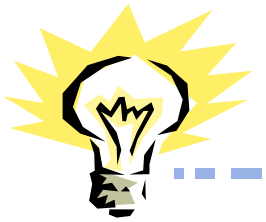
Capturing a Session

- 3. Software logging
  - Modify software to log user actions
  - Can give time-stamped key press or mouse event
  - Two problems:
    - Too low-level, want higher level events
    - Massive amount of data, need analysis tools



Subjective Data

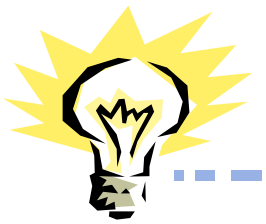
- Can ask about, for example:
  - Satisfaction (important factor in performance over time)
  - Preference
  - Workload



Methods

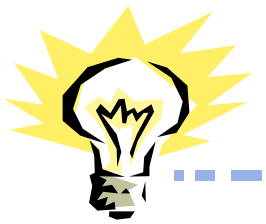
- Ways of gathering subjective data
  - Questionnaires
  - Interviews
  - Booths (e.g., trade show)
  - Call-in product hot-line
  - Field support workers

- (Focus on first two)



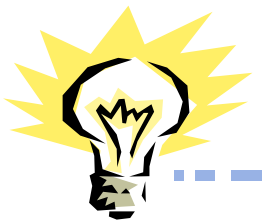
Questionnaires

- Preparation is expensive, but administration is cheap
- Oral vs. written/electronic
  - Oral advs: Can ask follow-up questions
  - Oral disadvs: Costly, time-consuming
- Forms can provide better quantitative data
- Lots of online survey tools



Questionnaires

- Issues
  - Only as good as questions you ask
  - Establish purpose of questionnaire
  - Don't ask things that you will not use
  - Who is your audience?
  - How do you deliver and collect questionnaire?



Questionnaire Topic

- Often, used to gather demographic data, and experience with technology/ the type of interface being studied
- Demographic data:
  - Age, gender
  - Task expertise
  - Motivation
  - Frequency of use
  - Education/literacy
  - Technology experience and attitudes

Question Format

- Closed format
  - Answer restricted to a set of choices
  - Typically very quantifiable
  - Variety of styles:

    Likert, multiple choice, rank order, check all that apply

Characters on screen

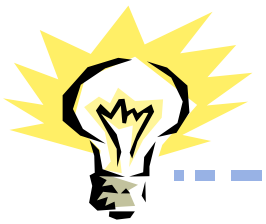
hard to read 1 2 3 4 5 6 7 easy to read

Which word processing systems do you use?

- ☐ LaTeX
- ☐ Word
- ☐ FrameMaker
- ☐ WordPerfect

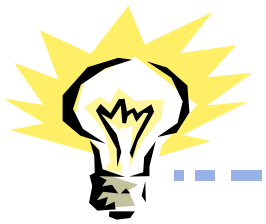
Rank from
1 - Very helpful
2 - Ambivalent
3 - Not helpful
0 - Unused

___ Tutorial
___ On-line help
___ Documentation



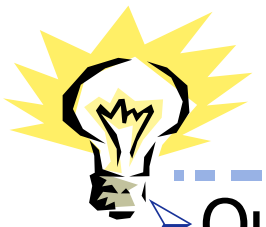
Closed Format

- Advantages
  - Clarify alternatives
  - Easily quantifiable
  - Eliminate useless answer
- Disadvantages
  - Must cover whole range
  - All should be equally likely
  - Don't get interesting, "different" reactions



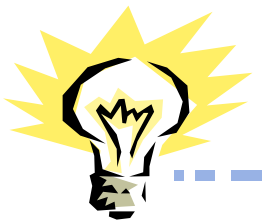
Open Format

- Asks for unprompted opinions
- Good for general, subjective information, but difficult to analyze rigorously
- May help with design ideas
  - "Can you suggest improvements to this interface?"



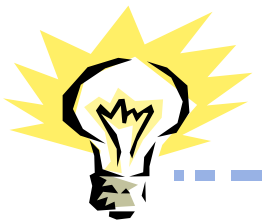
Questionnaire Issues

- Question specificity
  - "Do you have a computer?"
- Use language that will make sense to participants
  - Beware of terminology, jargon (in particular, internal corporate language)
- Clarity
  - There shouldn't be multiple possible interpretations
- Avoid leading questions
  - Can be phrased either positive or negative
- Double-barreled questions



Questionnaire Issues

- Prestige bias - (British sex survey)
  - People answer a certain way because they want you to think that way about them
- Bradley Effect
  - Respond one way in polls/questionnaires, behave in opposite or different way (political effect)
- Embarrassing questions
- Hypothetical questions
- "Halo effect"
  - When estimate of one feature affects estimate of another (e.g., intelligence/looks)

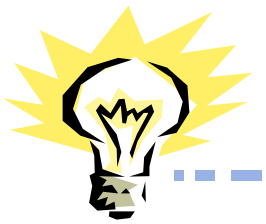


Questionnaire Deployment

- Steps:
  - Discuss questions among team
  - Administer verbally/written to a few people (pilot). Verbally query about thoughts on questions
  - Administer final test

Post-Task Interviews

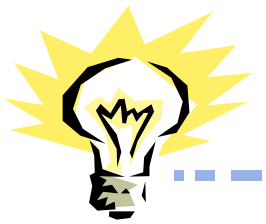
- Get user's viewpoint directly, but certainly a subjective view
- Advantages:
  - Can vary level of detail as issue arises
  - Good for more exploratory type questions which may lead to helpful, constructive suggestions
- Disadvantages
  - Subjective view
  - Interviewer can bias the interview
  - User may not appropriately characterize usage
  - Time-consuming



Archetypal Usability Test

- Have 5-10 participants think-aloud as they complete 10 tasks
- Moderator interviews participant throughout tasks
- Note-taker observes participant, records key utterances, takes note of errors, trends, preliminary findings
- After tasks are complete, a semistructured interview is conducted
- Lastly, participant completes a questionnaire with demographics, preference questions, satisfaction, etc.

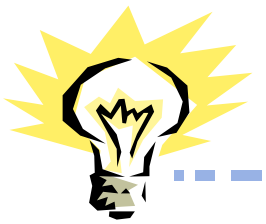
- After each session, notes and/or video data are gone over
- After a few participants, collate and meet to suggest design changes



Piloting

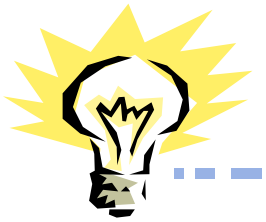
- Designing a study is similar to the UCD cycle.

- Run pilot versions to shake out the bugs
- Design->test->iterate



Basic Data Analysis

- In many cases, **immediate** analysis of your notes will yield good results
- Cross—check your observations with descriptive statistics
  - Determine the means (time, # of errors, etc.) and compare with target values (coming up…)
- Determine:
  - Why did the problems occur?
  - What were their causes?
  - The goal is to triage. Find the most prominent trends, and work on those.
  - But: if a problem only occurred once, and it was a valid problem, that is also worthy of attention

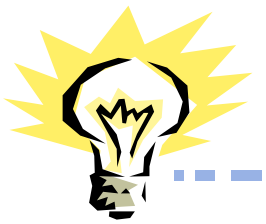


Inferential Statistics

- Sometimes you will be in a position to use statistical tests to compare alternative designs

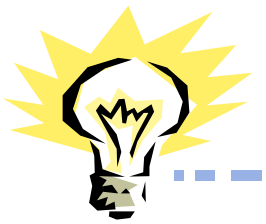
- For example:
  - 20 participants average 30 seconds to complete a task with design A, and 32 seconds to complete a task with design B

  - What do you conclude?



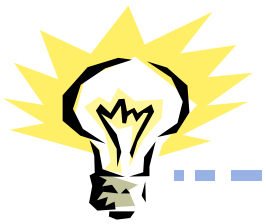
Drawing Conclusions from Results

- How does one know if an experiment's results mean anything or confirm any beliefs?
- Example: 20 people participated, 11 preferred interface A, 9 preferred interface B
- What do you conclude? Why?



Using the Results

- How do you use the results of your evaluation?
- How can you make your design better with this knowledge?
- How much user data do you need before drawing conclusions, or iterating?
  - Danger of over-correcting
- Often, the results of one round of evaluation will inspire the tasks that you will use for the next round, and will require new prototype features…



Upcoming

- Using the results of your evaluation
- More prototyping