Interaction Design

Chapter 8 (June 6, 2012, 9am-12pm):
Evaluating and Testing

What is evaluation?

- Part of the design-build-evaluate iterative design cycle
- A comparison of ‘built’ to ‘planned’
- A place to reflect on both this and the next design

Evaluation and testing

- Introduction
- Approaches to evaluation
- Evaluation Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

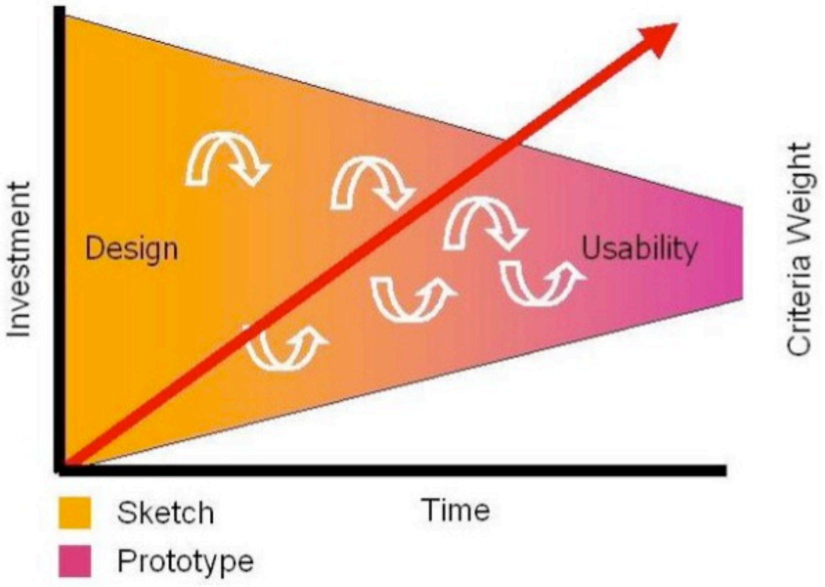
- Introduction
- Approaches to evaluation
- Evaluation Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

What is evaluation?

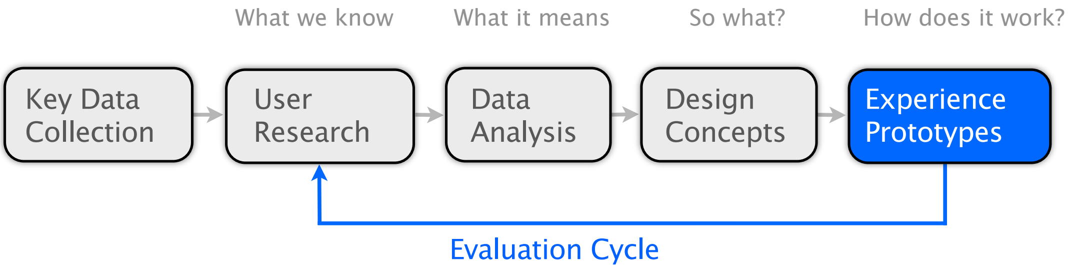
- Part of the design-build-evaluate iterative design cycle
- A comparison of ‘built’ to ‘planned’
- A place to reflect on both this and the next design

From ***Sketching User Experiences*** by Bill Buxton

Design process



When do you evaluate?



Being agile

- Fail fast to get success sooner:
  - Early iterations use cheap prototypes
  - Parallel design: build & test multiple prototypes
  - to explore design alternatives
- Increase fidelity progressively
- Reality checks, design for use cases, not specs

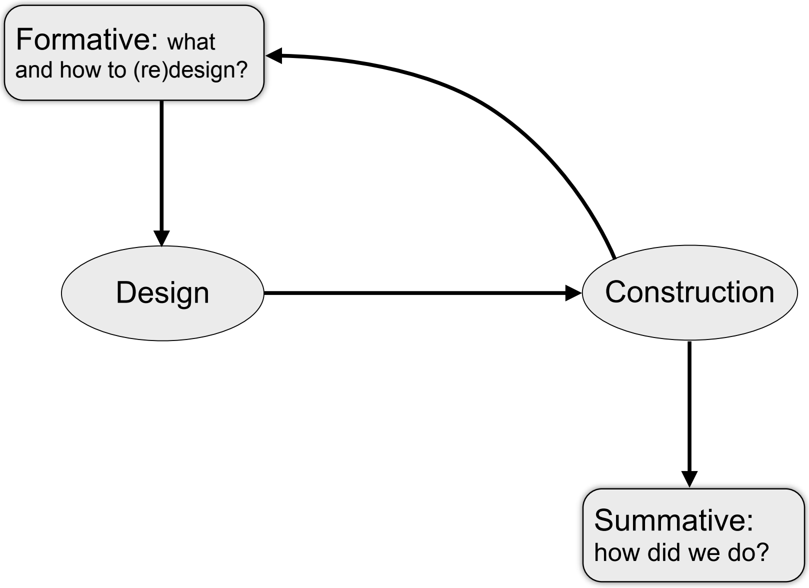
Evaluation and testing

- Introduction
- Approaches to evaluation
- Evaluation Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Evaluation Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

What do you evaluate?


Formative: what and how to (re)design?
Design
Construction
Summative: how did we do?

M. Scriven: The methodology of evaluation, 1967

Analytical vs. Empirical Evaluation

"If you want to evaluate a tool, say an axe, you might study the design of the bit, the weight distribution, the steel alloy used, the grade of hickory in the handle, etc., or you may just study the kind and speed of the cuts it makes in the hands of a good axeman."

[Scriven, 1967]

Empirical and Analytic Methods are Complementary

- Empirical evaluation helps to understand the context for object properties
  - If the axe does not cut well, what do we have to change?
  - Empirical evaluation produces facts which need to be interpreted

- Analytic evaluation identifies the crucial characteristics
  - Why does the axe have a special-shaped handle?
  - Analytical evaluation produces facts which need to be interpreted

Orthogonality of Approaches

	Formative	Summative
Analytical		
Empirical		

Orthogonality of Approaches

Without users, evaluates design choices

	Formative	Summative
Analytical		
Empirical		

Orthogonality of Approaches

Without users, evaluates design choices

Without users, evaluates the implementation

	Formative	Summative
Analytical		
Empirical		

Orthogonality of Approaches

Without users, evaluates design choices

Without users, evaluates the implementation

	Formative	Summative
Analytical		
Empirical		

With users, evaluates design choices

Orthogonality of Approaches

Without users, evaluates design choices

Without users, evaluates the implementation

	Formative	Summative
Analytical		
Empirical		

With users, evaluates design choices

With users, evaluates the implementation

Evaluation without Criteria is Useless

- Possible criteria (among many more):
  - Informal assessment of one idea against another
  - Detailed statistical analysis of average performance
  - using realistic user group (or actual field usage)
  - Fulfillment of informal usability heuristics
  - Fulfillment of formalized usability-related design metrics

Evaluation without Criteria is Useless

- Possible criteria (among many more):
  - Informal assessment of one idea against another
  - Detailed statistical analysis of average performance
  - using realistic user group (or actual field usage)
  - Fulfillment of informal usability heuristics
  - Fulfillment of formalized usability-related design metrics

You have to know in advance what you are looking for before you can evaluate!

From S. Klemmer - CS 147 @Stanford

Taxonomy of Methods [McGrath et al. 1994]

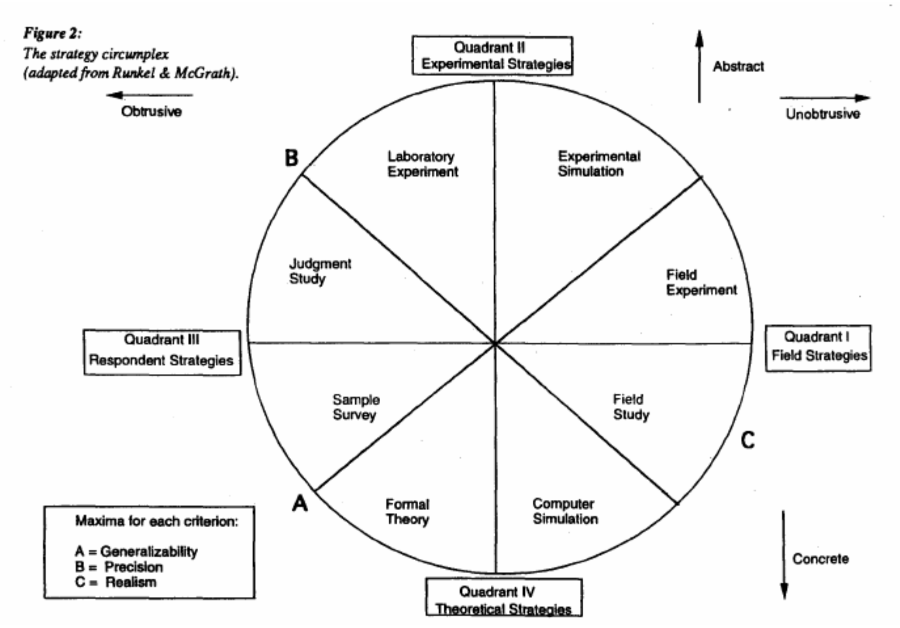


Figure 2:
The strategy circumplex
(adapted from Runkel & McGrath).

From S. Klemmer - CS 147 @Stanford

Taxonomy of Methods [McGrath et al. 1994]

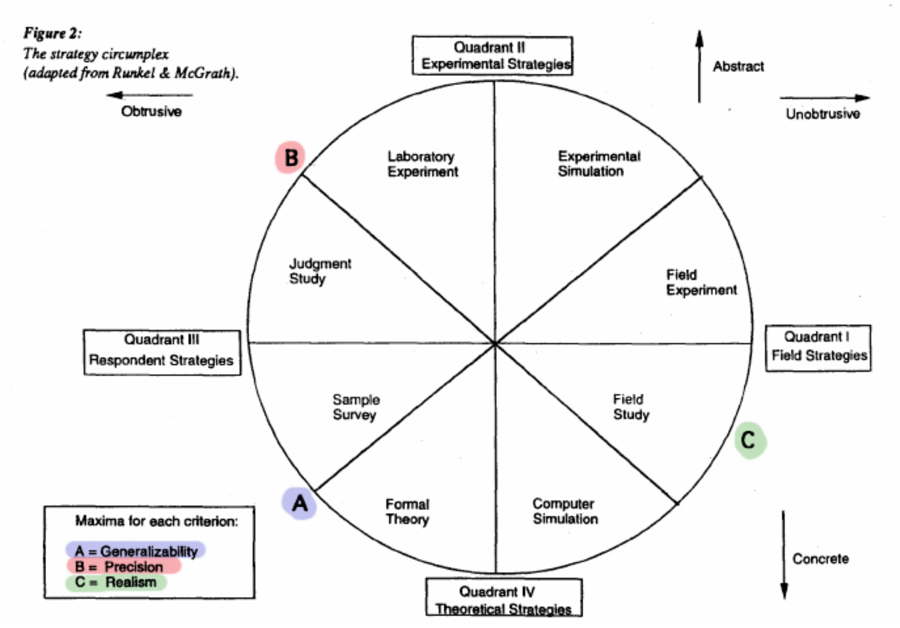


Figure 2:
The strategy circumplex
(adapted from Runkel & McGrath).

From S. Klemmer - CS 147 @Stanford

Taxonomy of Methods [McGrath et al. 1994]

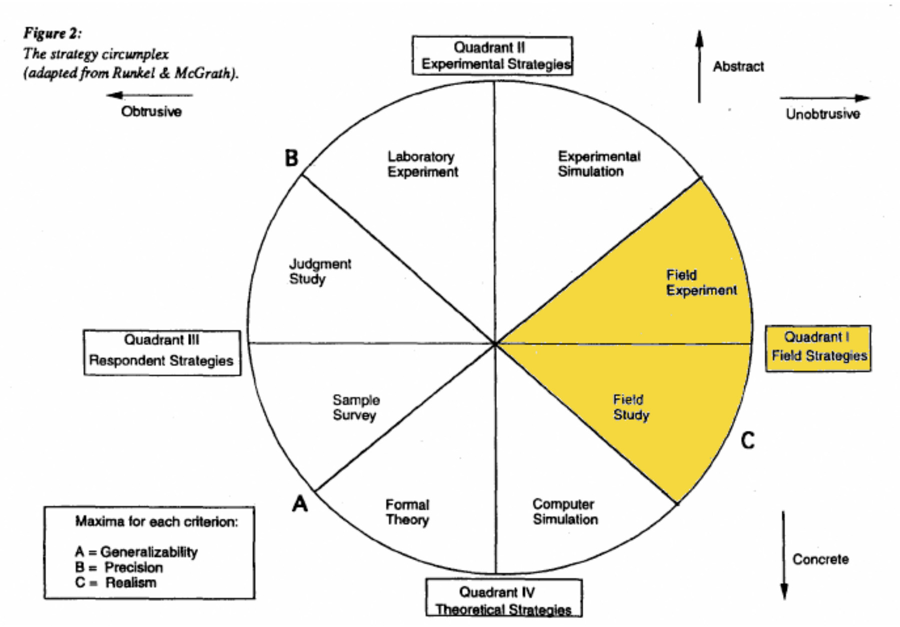


Figure 2:
The strategy circumplex
(adapted from Runkel & McGrath).

From S. Klemmer - CS 147 @Stanford

Taxonomy of Methods [McGrath et al. 1994]

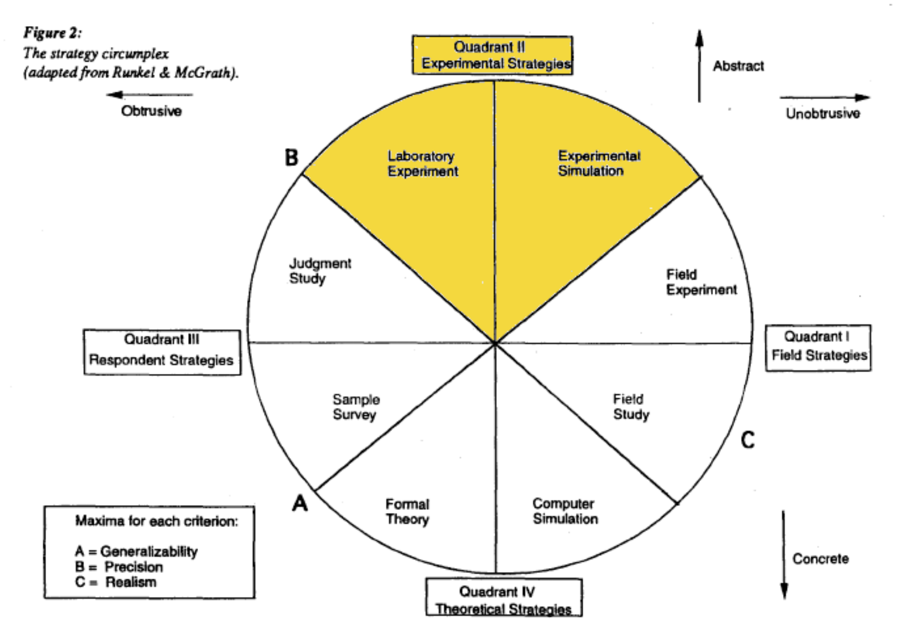


Figure 2:
The strategy circumplex
(adapted from Runkel & McGrath).

From S. Klemmer - CS 147 @Stanford

Taxonomy of Methods [McGrath et al. 1994]

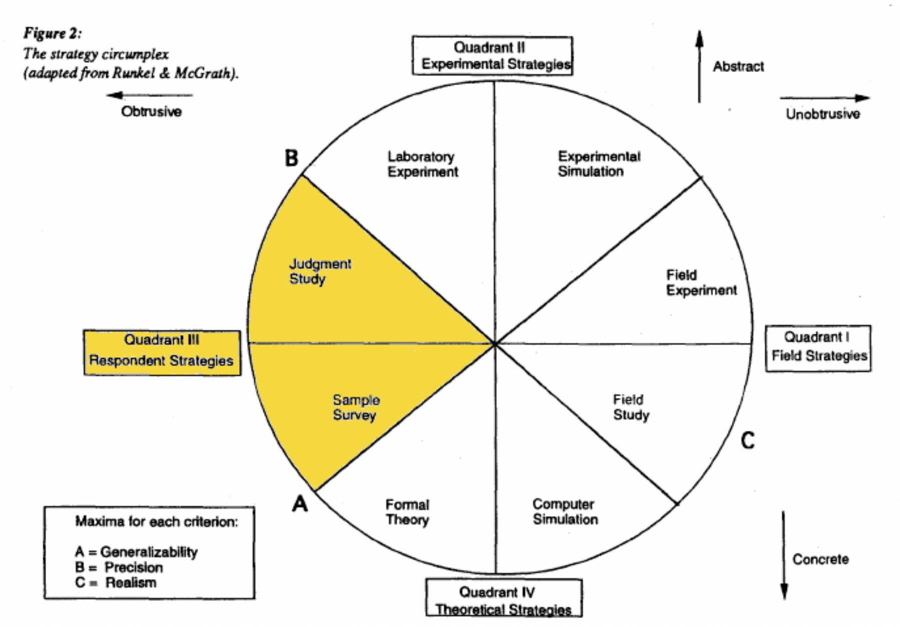


Figure 2:
The strategy circumplex
(adapted from Runkel & McGrath).

From S. Klemmer - CS 147 @Stanford

Taxonomy of Methods [McGrath et al. 1994]

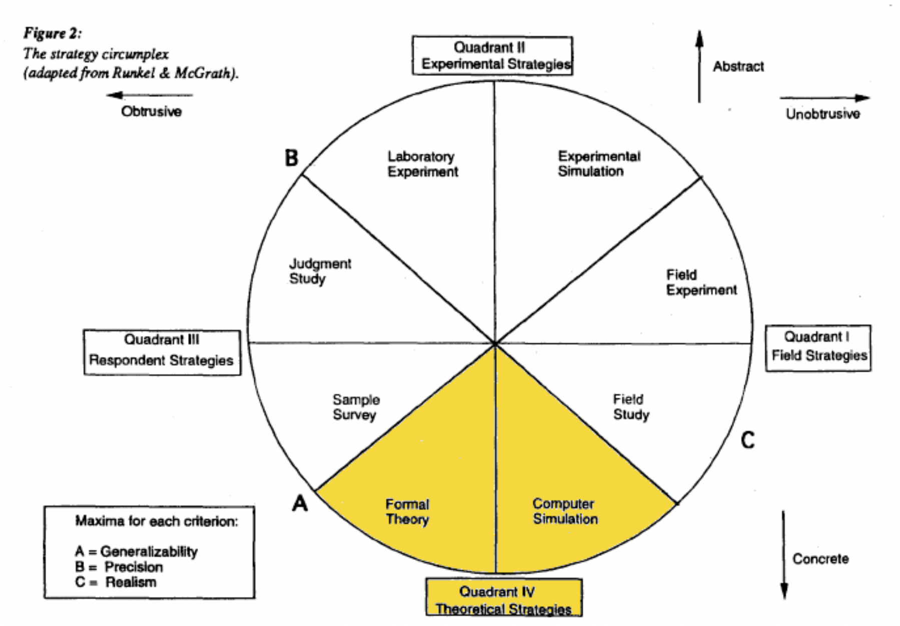


Figure 2:
The strategy circumplex
(adapted from Runkel & McGrath).

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Types of Analytical Evaluation

- Model-based evaluation
  - Evaluation according to models of how interaction works
- Inspection-based evaluation
  - Expert review
  - Cognitive walkthrough
  - Heuristic evaluation

Model based

- GOMS (Goals, Operators, Methods, and Selection rules)
  - Goals are what the user intends to accomplish.
  - Operators are actions that are performed to get to the goal.
  - Methods are sequences of operators that accomplish a goal. There can be more than one method available to accomplish a single goal, if this is the case then
  - Selection rules are used to describe when a user would select a certain method over the others. Selection rules are often ignored in typical GOMS analyses.
- KLM
  - Analyze an action and break it down into elementary steps
  - Read the duration of these steps from a table
  - Predict duration of the entire action
  - *Allows prediction before implementation!*

From http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/printer/goms.html

[John & Kieras, 1996]

GOMS analysis

```
GOAL: EDIT-MANUSCRIPT
.   GOAL: EDIT-UNIT-TASK ... repeat until no more unit tasks
.   .   GOAL: ACQUIRE UNIT-TASK
.   .       .                   GOAL: GET-NEXT-PAGE ... if at end of manuscript page
.   .       .                   GOAL: GET-FROM-MANUSCRIPT
.   .       GOAL: EXECUTE-UNIT-TASK ... if a unit task was found
.   .       .                   GOAL: MODIFY-TEXT
.   .       .                   .  [select: GOAL: MOVE-TEXT* ...if text is to be moved
.   .       .                   .      GOAL: DELETE-PHRASE ...if a phrase is to be deleted
.   .       .                   .      GOAL: INSERT-WORD] ... if a word is to be inserted
.   .       .                   .    VERIFY-EDIT

*Expansion of MOVE-TEXT goal
GOAL: MOVE-TEXT
.   GOAL: CUT-TEXT
.   .   GOAL: HIGHLIGHT-TEXT
.   .       .                  [select**: GOAL: HIGHLIGHT-WORD
.   .       .                  .  MOVE-CURSOR-TO-WORD
.   .       .                  .  DOUBLE-CLICK-MOUSE-BUTTON
.   .       .                  .  VERIFY-HIGHLIGHT
.   .       .                                 GOAL: HIGHLIGHT-ARBITRARY-TEXT
.   .       .                  .               MOVE-CURSOR-TO-BEGINNING          1.10
.   .       .                  .               CLICK-MOUSE-BUTTON                0.20
.   .       .                  .               MOVE-CURSOR-TO-END                1.10
.   .       .                  .               SHIFT-CLICK-MOUSE-BUTTON          0.48
.   .       .                  .               VERIFY-HIGHLIGHT]                 1.35
.   .       GOAL: ISSUE-CUT-COMMAND
.   .       .   MOVE-CURSOR-TO-EDIT-MENU                                         1.10
.   .       .   PRESS-MOUSE-BUTTON                                               0.10
.   .       .   MOVE-CURSOR-TO-CUT-ITEM                                          1.10
.   .       .   VERIFY-HIGHLIGHT                                                 1.35
.   .       .   RELEASE-MOUSE-BUTTON                                             0.10

...
```

From http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/printer/goms.html

[John & Kieras, 1996]

GOMS analysis

```
*Expansion of MOVE-TEXT goal
GOAL: MOVE-TEXT
.   GOAL: CUT-TEXT
.   .   GOAL: HIGHLIGHT-TEXT
.   .       .                  [select**: GOAL: HIGHLIGHT-WORD
.   .       .                  .  MOVE-CURSOR-TO-WORD
.   .       .                  .  DOUBLE-CLICK-MOUSE-BUTTON
.   .       .                  .  VERIFY-HIGHLIGHT
.   .       .                                 GOAL: HIGHLIGHT-ARBITRARY-TEXT
.   .       .                  .               MOVE-CURSOR-TO-BEGINNING          1.10
.   .       .                  .               CLICK-MOUSE-BUTTON                0.20
.   .       .                  .               MOVE-CURSOR-TO-END                1.10
.   .       .                  .               SHIFT-CLICK-MOUSE-BUTTON          0.48
.   .       .                  .               VERIFY-HIGHLIGHT]                 1.35
.   .       GOAL: ISSUE-CUT-COMMAND
.   .       .   MOVE-CURSOR-TO-EDIT-MENU                                         1.10
.   .       .   PRESS-MOUSE-BUTTON                                               0.10
.   .       .   MOVE-CURSOR-TO-CUT-ITEM                                          1.10
.   .       .   VERIFY-HIGHLIGHT                                                 1.35
.   .       .   RELEASE-MOUSE-BUTTON                                             0.10
.   GOAL: PASTE-TEXT
.   .       GOAL: POSITION-CURSOR-AT-INSERTION-POINT
.   .       MOVE-CURSOR-TO-INSERTION-POIONT                                      1.10
.   .       CLICK-MOUSE-BUTTON                                                   0.20
.   .       VERIFY-POSITION                                                      1.35
.   .       GOAL: ISSUE-PASTE-COMMAND
.   .       .   MOVE-CURSOR-TO-EDIT-MENU                                         1.10
.   .       .   PRESS-MOUSE-BUTTON                                               0.10
.   .       .   MOVE-MOUSE-TO-PASTE-ITEM                                         1.10
.   .       .   VERIFY-HIGHLIGHT                                                 1.35
.   .       .   RELEASE-MOUSE-BUTTON                                             0.10
TOTAL TIME PREDICTED (SEC)                                                       14.38
```

Based on the above GOMS analysis, it should take 14.38 seconds to move text.

From http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/printer/goms.html

KLM

Description	Operation	Time (sec)
Reach for mouse	H[mouse]	0.40
Move pointer to "Replace" button	P[menu item]	1.10
Click on "Replace" command	K[mouse]	0.20
Home on keyboard	H[keyboard]	0.40
Specify word to be replaced	M4K[word]	2.15
Reach for mouse	H[mouse]	0.40
Point to correct field	P[field]	1.10
Click on field	K[mouse]	0.20
Home on keyboard	H[keyboard]	0.40
Type new word	M4K[word]	2.15
Reach for mouse	H[mouse]	0.40
Move pointer on Replace-all	P[replace-all]	1.10
Click on field	K[mouse]	0.20
Total		10.2

Limitations

- Predictions are only valid for expert users not making any errors.
  - expert users will make mistakes
  - no consideration of novices or intermediate users who make occasional errors.
  - extensions try to model learning
- All tasks are goal-directed
  - Some tasks like problem-solving are less directed.
- Does not take into account individual differences among users,
  - Relies on statistical averages
- Does not take into account the social or organizational impact of the product.
- No insight on how useful or enjoyable the product under design.
- Not representative of current theories of human cognition.
  - Assumes a serial model of human cognition: One activity done at a time.

Inspections & Expert Review

- Throughout the development process
- Performed by developers and experts
- External or internal experts
- Tool for finding problems
- May take between an hour and a week
- Structured approach is advisable
  - Reviewers should be able to communicate all their issues (without hurting the team)
  - Reviews must not be offensive for developers / designers
  - The main purpose is finding problems
- Solutions may be suggested but decisions are up to the team

Inspection Methods

- Guideline review
  - Check that the UI is according to a given set of guidelines
- Consistency inspection
  - Check that the UI is consistent (in itself, within a set of related applications, with OS)
  - Bird’s eye view can help, e.g. printout of a web site and put it up on the wall)
  - Consistency can be enforced by design (e.g. CSS for Web sites)

- Procedure for inspections:
  - Find reviewers, define schedule
  - Prepare material for reviewers, including criteria
  - On-site or off-site review
  - Review report, definition of consequences

Expert evaluation pro&cons

- Results of informal reviews and inspections are often directly used to change the product
  - ... still state of the art in many companies!
  - The personal view of the CEO, or his partner ...
- Really helpful evaluation
  - Is explicit
  - Has clearly documented findings
  - Can increase the quality significantly
- Expert reviews and inspections are a starting point for change

more at http://hcibib.org/tcuid/chap-4.html#4-1

Cognitive Walkthrough

One or more evaluators going through a set of tasks

- Evaluating understandability and ease of learning

Procedure:

- Defining the input:
  - Who will be the users of the system?
  - What task(s) will be analyzed?
  - What is the correct action sequence for each task?
  - How is the interface defined?
- During the walkthrough:
  - Will the users try to achieve the right effect?
  - Will the user notice that the correct action is available?
  - Will the user associate the correct action with the effect to be achieved?
  - If the correct action is performed, will the user see that progress is being made toward solution of the task?

Usability guidelines

- Don Norman’s principles:
  - visibility, affordances, natural mapping, and feedback
- Ben Shneiderman’s 8 Golden Rules of UI design
- Bruce Tognazzini’s 16 principles:
  - http://www.asktog.com/basics/firstPrinciples.html
- Christian Bastien’s Ergonomic Criteria
- Jakob Nielsen’s Heuristics

Heuristic Evaluation

- Heuristic evaluation is a “discount” usability inspection method
  - Quick, cheap and easy evaluation of UI design
  - http://www.useit.com/papers/heuristic/
- Implicit assumptions:
  - There is a fixed list of desirable properties of user interfaces (the “heuristics”)
  - These heuristics can be checked by experts with a clear and defined result

Ten Usability Heuristics

http://www.useit.com/jakob/photos/

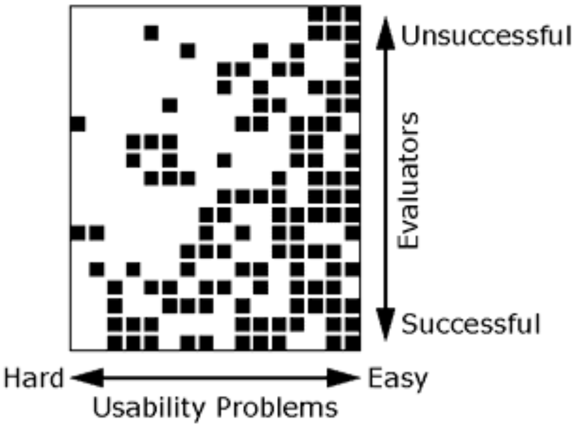
- Meet expectations
  1. Match the real world
  2. Consistency & standards
  3. Help & documentation
- User is boss
  4. User control & freedom
  5. Visibility of system status
  6. Flexibility & efficiency
- Errors
  7. Error prevention
  8. Recognition, not recall
  9. Error reporting, diagnosis, and recovery
- Keep it simple
  10. Aesthetic & minimalist design

Procedure

- Small set of evaluators examine the interface and judge its compliance
- with recognized usability principles (the "heuristics").
- Either just by inspection or by scenario-based walkthrough
- Critical issues list, weighted by severity grade
- Opinions of evaluators are consolidated into one report

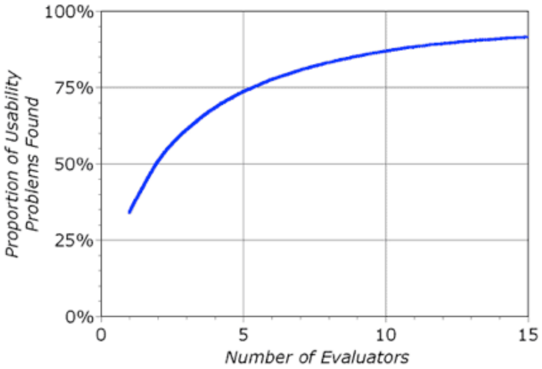
Number of evaluators

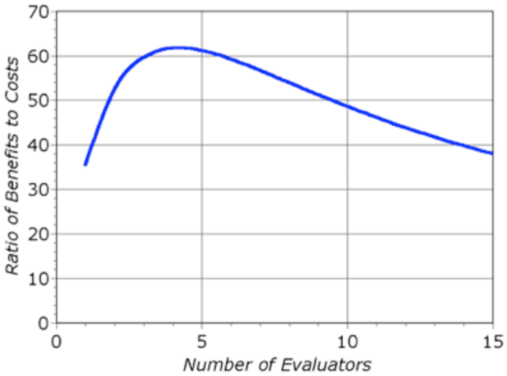
- Every evaluator doesn’t find every problem
- Good evaluators find both easy & hard ones



Number of evaluators

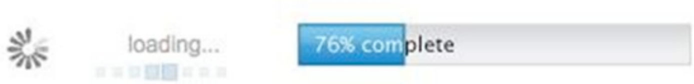
- Single evaluator achieves poor results
- Only finds 35% of usability problems
- 5 evaluators find ~ 75% of usability problems





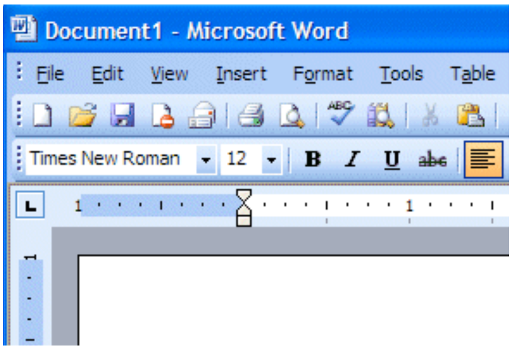
Heuristics

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



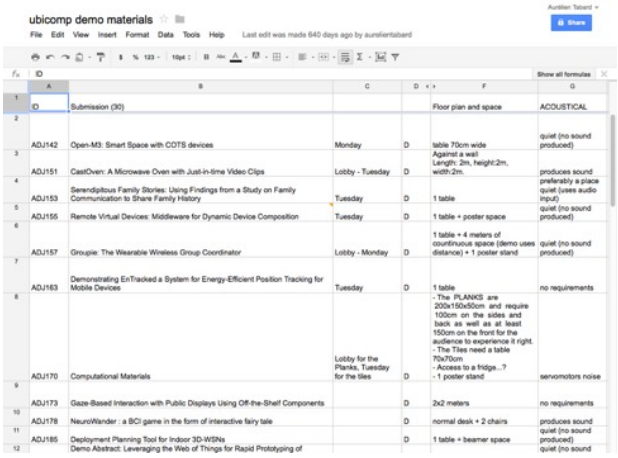
Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



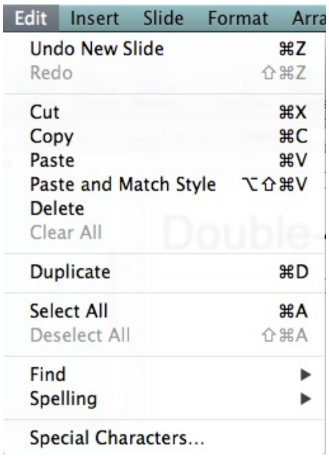
Aurélien Tabard
ubicomp demo materials
Share
File Edit View Insert Format Data Tools Help
Last edit was made 640 days ago by aurelientabard
10pt
Show all formulas

fx ID

	A	B	C	D	F	G
1	ID	Submission (30)			Floor plan and space	ACOUSTICAL
2	ADJ142	Open-M3: Smart Space with COTS devices	Monday	D	table 70cm wide	quiet (no sound produced)
3	ADJ151	CastOven: A Microwave Oven with Just-in-time Video Clips	Lobby - Tuesday	D	Against a wall Length: 2m, height:2m, width:2m.	produces sound
4	ADJ153	Serendipitous Family Stories: Using Findings from a Study on Family Communication to Share Family History	Tuesday	D	1 table	preferably a place quiet (uses audio input)
5	ADJ155	Remote Virtual Devices: Middleware for Dynamic Device Composition	Tuesday	D	1 table + poster space	quiet (no sound produced)
6	ADJ157	Groupie: The Wearable Wireless Group Coordinator	Lobby - Monday	D	1 table + 4 meters of countinuous space (demo uses distance) + 1 poster stand	quiet (no sound produced)
7	ADJ163	Demonstrating EnTracked a System for Energy-Efficient Position Tracking for Mobile Devices	Tuesday	D	1 table	no requirements
8	ADJ170	Computational Materials	Lobby for the Planks, Tuesday for the tiles	D	- The PLANKS are 200x150x50cm and require 100cm on the sides and back as well as at least 150cm on the front for the audience to experience it right. - The Tiles need a table 70x70cm - Access to a fridge...? - 1 poster stand	servomotors noise
9	ADJ173	Gaze-Based Interaction with Public Displays Using Off-the-Shelf Components		D	2x2 meters	no requirements
10	ADJ178	NeuroWander : a BCI game in the form of interactive fairy tale		D	normal desk + 2 chairs	produces sound
11	ADJ185	Deployment Planning Tool for Indoor 3D-WSNs		D	1 table + beamer space	quiet (no sound produced)
12		Demo Abstract: Leveraging the Web of Things for Rapid Prototyping of				quiet (no sound

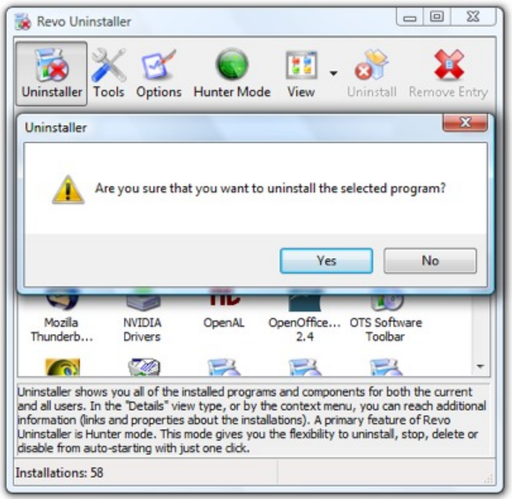
Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



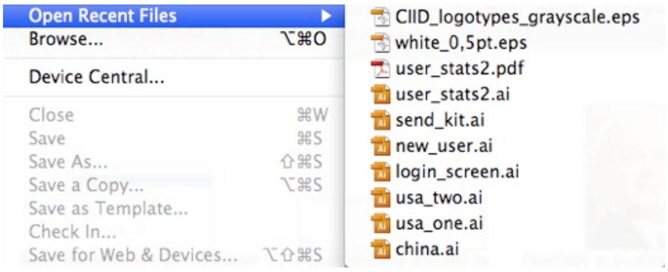
Are you sure that you want to uninstall the selected program?

Uninstaller shows you all of the installed programs and components for both the current and all users. In the "Details" view type, or by the context menu, you can reach additional information (links and properties about the installations). A primary feature of Revo Uninstaller is Hunter mode. This mode gives you the flexibility to uninstall, stop, delete or disable from auto-starting with just one click.

Installations: 58

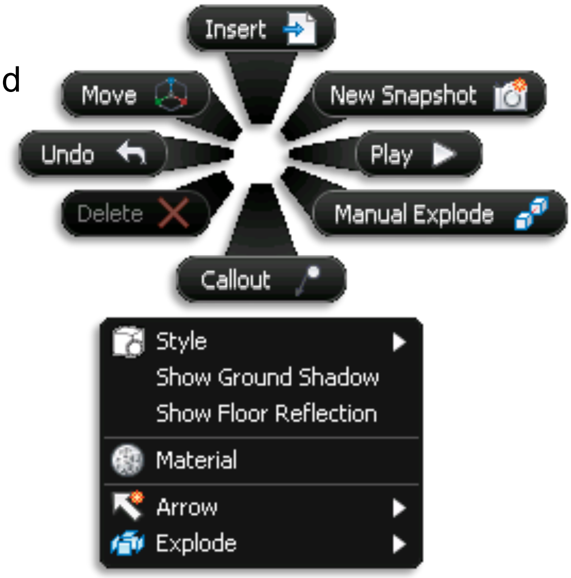
Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation

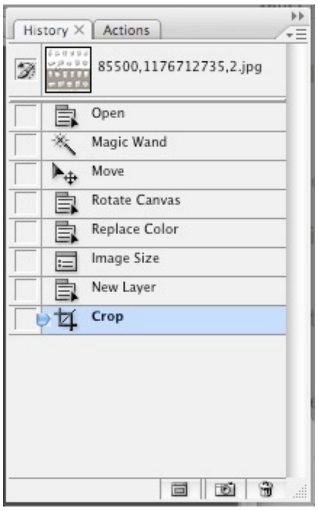


Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation

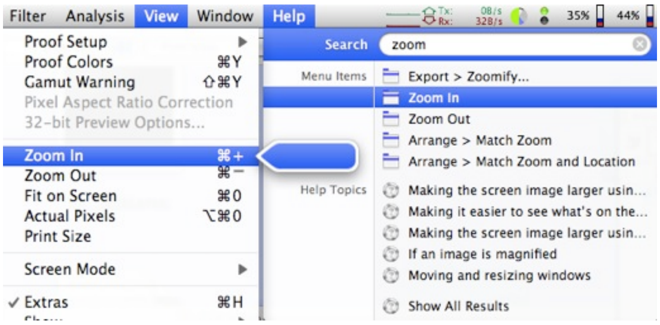
Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Heuristic

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and documentation



Severity scale

- Contributing factors
  - Frequency: how common?
  - Impact: how hard to overcome?
  - Persistence: how often to overcome?

- Severity scale
  - Cosmetic: need not be fixed
  - Minor: needs fixing but low priority
  - Major: needs fixing and high priority
  - Catastrophic: imperative to fix

Writing good heuristic evaluations

- Heuristic evaluations must communicate well to developers and managers

- Include positive comments as well as criticisms
  - Good: Toolbar icons are simple, with good contrast and few colors (minimalist design)

- Be tactful
  - Not: the menu organization is a complete mess
  - Better: menus are not organized by function

- Be specific
  - Not: text is unreadable
  - Better: text is too small, and has poor contrast (black text on dark green background)

Example

- What to include:
  - Problem
  - Heuristic
  - Description
  - Severity
  - Recommendation (if any)
  - Screenshot (if helpful)

Severe: User may close window without saving data (error prevention)

If the user has made changes without saving, and then closes the window using the Close button, rather than File >> Exit, no confirmation dialog appears.

Recommendation: show a confirmation dialog or save automatically

Summary

- Heuristic evaluation is a discount method
- Have evaluators go through the UI twice
  - Ask them to see if it complies with heuristics
  - Note where it doesn’t and say why
- Have evaluators independently rate severity
- Combine the findings from 3 to 5 evaluators
- Discuss problems with design team
- Cheaper alternative to user testing
- Finds different problems, so good to alternate

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Empirical evaluations

- Field study
  - Find problems in context
  - Evaluates working implementation, in real context, on real tasks
  - Mostly qualitative observations
- Focus groups
- Usability evaluation
  - Find problems for the next design iteration
  - Evaluates prototype or implementation, in lab,on chosen tasks
  - Qualitative observations (usability problems)
- Physiological measurement (e.g. eye tracking)
- Controlled experiment
  - Tests a hypothesis (e.g., interface X is faster than interfaceY)
  - Evaluates working implementation, in controlled lab environment, on chosen tasks
  - Mostly quantitative observations (time, error rate, satisfaction)

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?
  - people use it!

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?
  - people use it!

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?
  - people use it!
- Other methods are based on evaluators who

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?
  - people use it!
- Other methods are based on evaluators who
  - may know too much

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?
  - people use it!
- Other methods are based on evaluators who
  - may know too much
  - may not know enough (about tasks, etc.)

From S. Klemmer - CS 147 @Stanford

Why use empirical methods?

- Can’t tell how good UI is until?
  - people use it!
- Other methods are based on evaluators who
  - may know too much
  - may not know enough (about tasks, etc.)
  - Hard to predict what real users will do

Motivations

- identify any usability problems that the product has
- collect quantitative data on participants' performance
- determine participants' satisfaction with the product

Field Studies

- Activities studied in situation (where they usually happen).

- Advantages:
  - Can reveal results on user acceptance
  - Allows longitudinal studies, including learning, collaboration and adaptation
- Problems:
  - In general very expensive
  - Reliable product (or prototype) needed
  - How to get observations?
    - Collecting usage data
    - Direct observation, regular interviews
    - On-line feedback
    - Retrospective interviews, questionnaires

Focus Groups

- Informal, qualitative group discussion of specific topic
  - Get indication of how people think and feel
  - Collecting opinions, attitudes, feelings, needs and ideas
  - Understand why people act or react in a certain way
- Early in the design process, before UI design or implementation
- Complementary to more elaborate, quantitative studies
- Setup:
  - Groups of 6 to 8 participants
  - Conducted by a moderator
  - Duration 1,5 to 2 hours
- Analysis of script, video recording => simple report with quotes

Focus groups pro&cons

Advantages

- Fast, easy, cheap
- In depth information about users' opinions, motives, motivations
- Flexible, exploration of different topics and materials

Disadvantages

- Not representative, hard to generalize
- What users think vs. what users actually do
- Analysis can be laborious
- Can be biased by moderator or people with strong opinions

Getting Participants

- Representative of target users
  - job-specific vocabulary / knowledge
  - tasks
- Approximate if needed
  - system intended for doctors
    - get medical students
  - system intended for engineers
    - get engineering students
- Use incentives to get participants

From R. Miller - 6.831 @MIT

Ethics!

- Pressures on users:
  - Performance anxiety
  - Feels like an intelligence test
  - Comparing self with other subjects
  - Feeling stupid in front of observers
  - Competing with other subjects

From R. Miller - 6.831 @MIT

Respect and control

- Time
  - Don’t waste it
- Comfort
  - Make the user comfortable
- Informed consent
  - Inform the user as fully as possible
- Privacy
  - Preserve the users privacy
- Control
  - The user can stop at any time

From R. Miller - 6.831 @MIT

Before a test

- Time
  - Pilot-test all materials and tasks
- Comfort
  - We’re testing the system; were not testing you.
  - Any difficulties you encounter are the systems fault. We need your help to find these problems.
- Privacy
  - Your test results will be completely confidential.
- Information
  - Brief about purpose of study
  - Inform about audio-taping, video-taping, other observers, make sure it is ok or disable the ones the subject is not comfortable with
  - Answer any questions beforehand (unless biasing)
- Control
  - You can stop at any time.

From R. Miller - 6.831 @MIT

During the test

- Time
  - Eliminate unnecessary tasks
- Comfort
  - Calm,relaxed atmosphere
  - Take breaks in long session
  - Never act disappointed
  - Give tasks one at a time
  - First task should be easy, for an early success experience
- Privacy
  - Users' boss shouldn't be watching
- Information
  - Answer questions (again,where they won't bias)
- Control
  - User can give up a task and go on to the next
  - User can quit entirely

From R. Miller - 6.831 @MIT

After the test

- Comfort
  - Say what they've helped you do
- Information
  - Answer questions that you had to defer to avoid biasing the experiment
- Privacy
  - Don't publish user-identifying information
  - Don't show video or audio without users permission

From Wikipedia via S. Klemmer - CS 147 @Stanford

What is usability testing?

Usability testing is a means for measuring how well people can use some human-made object (such as a web page, a computer interface, a document, or a device) for its intended purpose, i.e. usability testing measures the usability of the object.

Metrics

- Ease of learning
  - – learning time, …
- Ease of use
  - – performance time, error rates…
- User satisfaction
  - – surveys…

Metrics

- Ease of learning
  - learning time, …
- Ease of use
  - performance time, error rates…
- User satisfaction
  - surveys…

Not “user friendly”!

Metrics

- Ease of learning
  - learning time, …
- Ease of use
  - performance time, error rates…
- User satisfaction
  - surveys…

Not “intuitive”!

Not “user friendly”!

Metrics

- Ease of learning
  - learning time, …
- Ease of use
  - performance time, error rates…
- User satisfaction
  - surveys…

Not "natural"!

Not "intuitive"!

Not "user friendly"!

What data to gather

	Process	*Bottom-line*
Qualitative		
Quantitative		

What data to gather

observations of what users are doing & thinking

	Process	*Bottom-line*
Qualitative		
Quantitative		

What data to gather

observations of what users are doing & thinking

summary of what happened (time, errors, success)

	Process	*Bottom-line*
Qualitative		
Quantitative		

What you gather (quantitative)

- Quantitative data, which might include:
  - Success rates
  - Accuracy / Error rates : How many mistakes did people make? And were they fatal or recoverable with the right information?
  - Time on Task: How long does it take people to complete basic tasks? (For example, find something to buy, create a new account, and order the item.)
  - Pages visited, number of steps to reach goal...
  - Recall: How much does the person remember afterwards or after periods of non-use?
  - Emotional Response: Ratings on a satisfaction questionnaire, How does the person felt about the tasks completed? (Confident? Stressed? Would the user recommend this system to a friend?)

What you gather (qualitative)

- Qualitative data, which might include notes on:
  - How people reacted to the system.
  - How participants understood it.
  - Which the pathways participants took.
  - Which problems participants had (critical incidents).
  - What participants said as they worked.
  - Participants' answers to open-ended questions.

You need a plan!

- A good plan for usability testing gives the participants:
  - a goal/task (what to do or what question to find the answer for)
  - data, if needed, that a real user would have when going to the site to do that task
- You can give the scenario as just the statement of the goal/task or you can elaborate it a little with a very short story that adds motivation to get to the goal.

Participants

- The participants must be like the people who will use your product.
- Be ready to screen participants (do not grab the first person in the corridor)
- Plan on a cost associated with finding the people
  - you may still need to plan on incentives to get participants to participate ...

Test!

- Make sure you have everything you need
  - the prototype you are going to test
  - the computer set up for the participant with the monitor, resolution, and connection speed that you indicated in the test plan
  - note-taking forms on paper or set up on a computer
  - consent forms for participants to sign and a pen in case the participant does not bring one
  - questionnaires, if you are using any
  - the participant's copy of the scenarios
  - cameras, microphones, or other recording equipment if you are using any
  - folders to keep each person's paperwork in if you are using paper
- Do a dry-run and a pilot test

Before starting

- You should know, and have written down
  - objective
  - description of system being testing
  - task environment & materials
  - participants
  - methodology
  - tasks
  - test measures

- Will help you design a good usability test
- Will help you figure out how to analyze your data

Usability laboratory

- Specifically constructed testing room
  - Instrumented with data collection
  - devices (e.g. microphones, cameras)
- Separate observation room
  - Usually connected to testing room
  - by one-way mirror and audio system
  - Data recording and analysis
- Test users perform prepared scenarios
  - “Think aloud” technique
  - Decide whether to interrupt or not
  - Keep variances among tests low
- Problem:
  - Very artificial setting
  - No communication

From C|Net “How Google tested Google Instant”
http://news.cnet.com/8301-30684_3-20019652-265.html

Think aloud

- Need to know what users are thinking, not just what they are doing
- Ask users to talk while performing tasks
  - tell us what they are thinking
  - tell us what they are trying to do
  - tell us questions that arise as they work
  - tell us things they read
- Make a recording or take good notes
  - make sure you can tell what they were doing
  - use a digital watch/clock
  - take notes, plus if possible record audio & video (or even event logs)
- Prompt the user to keep talking
  - “tell me what you are thinking”
  - Only help on things you have pre-decided
  - keep track of anything you do give help on

Usability testing analysis and limitations

- Summarize the data
  - make a list of all critical incidents
    - positive & negative
  - include references back to original data
  - try to judge why each difficulty occurred
- What does data tell you?
  - UI work the way you thought it would? users take approaches you expected?
  - something missing?
- Update task analysis & rethink design
  - rate severity & ease of fixing CIs
  - fix both severe problems & make the easy fixes
- Will thinking aloud give the right answers?
  - not always
  - if you ask a question, people will always give an answer, even it is has nothing to do with facts
  - try to avoid specific questions

- Situations in which numbers are useful
  - time requirements for task completion
  - successful task completion
  - compare two designs on speed or # of errors
- Ease of measurement
  - time is easy to record
  - error or successful completion is harder
  - define in advance what these mean
- Do not combine efficiency measures with thinking-aloud.
  - talking can affect speed & accuracy

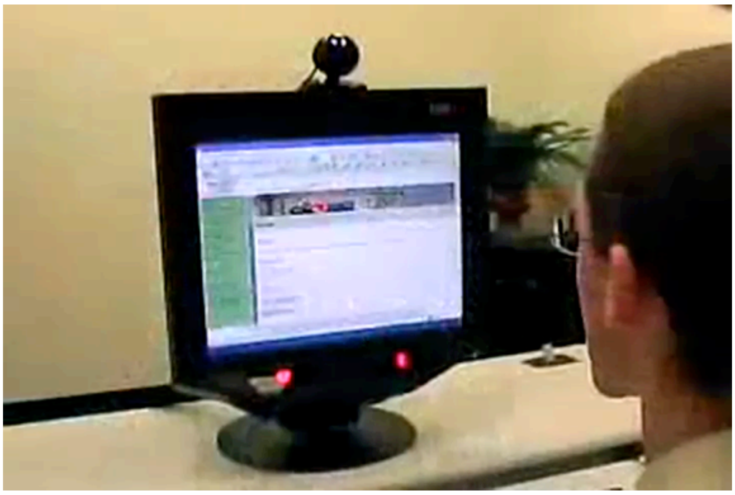
Physiological measurements

- Eye tracking
  - well developed and robust
- Stress
  - e.g. skin conductivity
- Brain activity
  - experimental

Eye-tracker - © Kent State University (US)

Physiological measurements

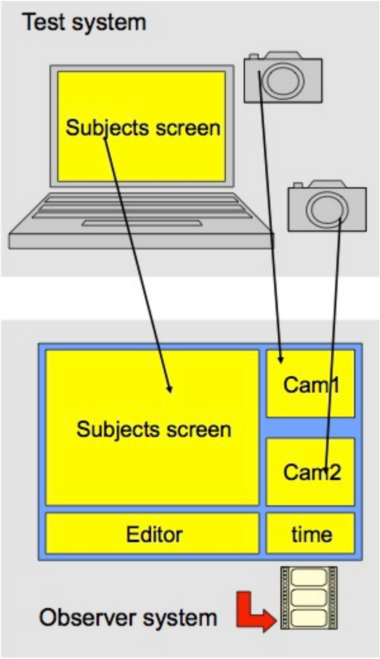
- Eye tracking
  - well developed and robust
- Stress
  - e.g. skin conductivity
- Brain activity
  - experimental



Eye-tracker - © Kent State University (US)

Usability lab on the cheap

- Goal: Integrate multiple views
  - Capture screen with pointer
  - View of the person interacting with the system
  - View of the environment
- Setup:
  - Computer for the test user
    - Application to test
    - Capture tool
  - Computer for the observer
    - See the screen of the subject
    - Attach 2 web cams (face and entire user)
    - Display them on the observer‘s screen
    - Have an editor for the observer‘s notes
    - Capture this screen
- Debrief with the users afterwards



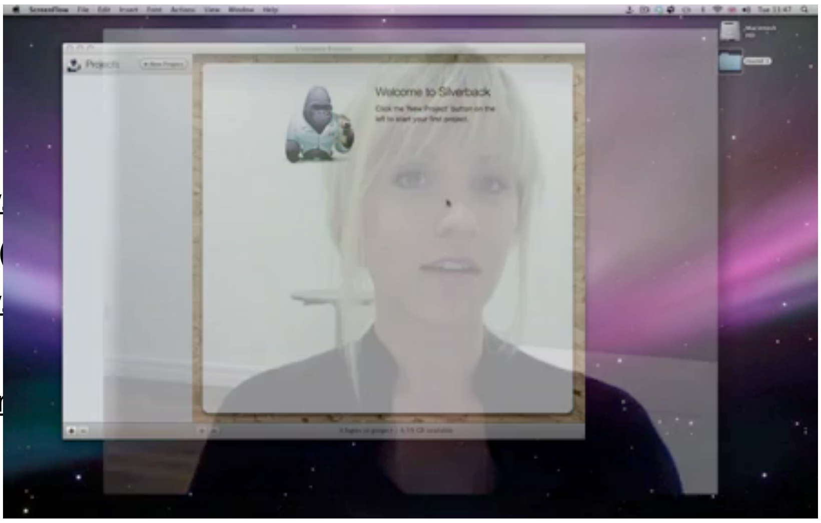
Really cheap

Existing tools

- Morae
  - http://www.techsmith.com/morae.html
- Ovo studio (free for students)
  - http://www.ovostudios.com
- Silverback
  - http://silverbackapp.com/

Existing tools

- Morae
  - http://www
- Ovo studio (
  - http://www
- Silverback
  - http://silver



Welcome to Silverback

Click the 'New Project' button on the left to start your first project.

Evaluation and testing

- Introduction
- Approaches to evaluation
- Evaluation Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Evaluation Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

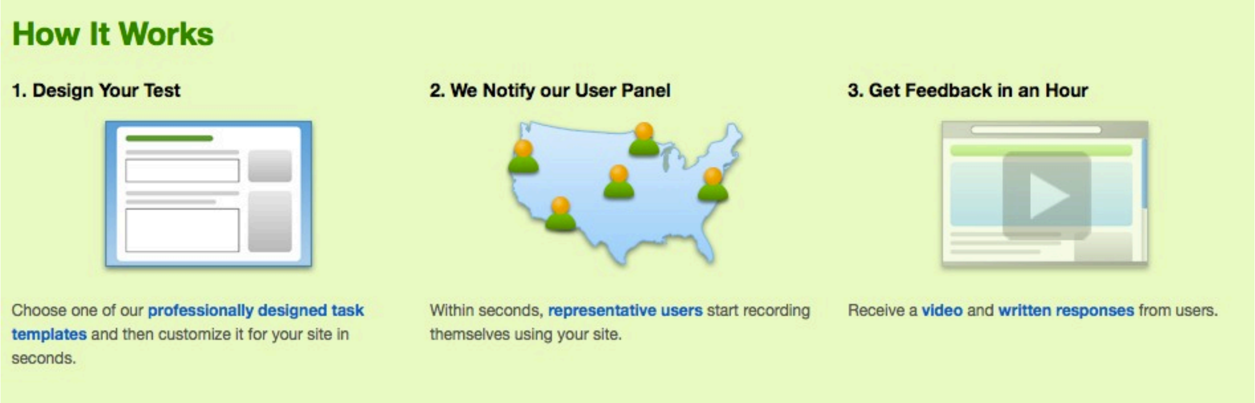
Scaling usability studies

- Large web audiences
- Large mobile audiences

- Easy distribution and updates

Remote usability studies

E.g. Usertesting.com...



How It Works

1. Design Your Test

Choose one of our **professionally designed task templates** and then customize it for your site in seconds.

2. We Notify our User Panel

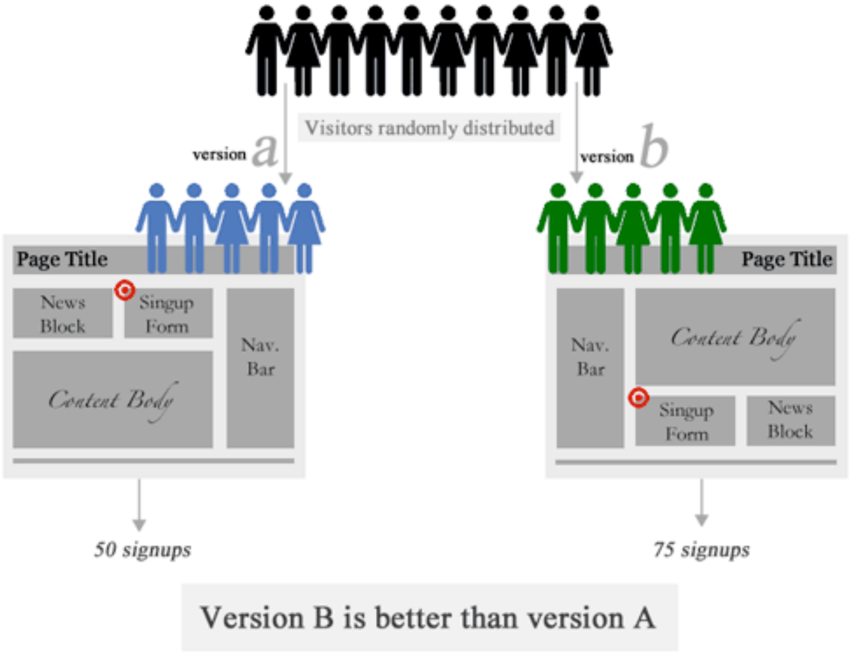
Within seconds, **representative users** start recording themselves using your site.

3. Get Feedback in an Hour

Receive a **video** and **written responses** from users.

From http://www.smashingmagazine.com/2010/06/24/the-ultimate-guide-to-a-b-testing/

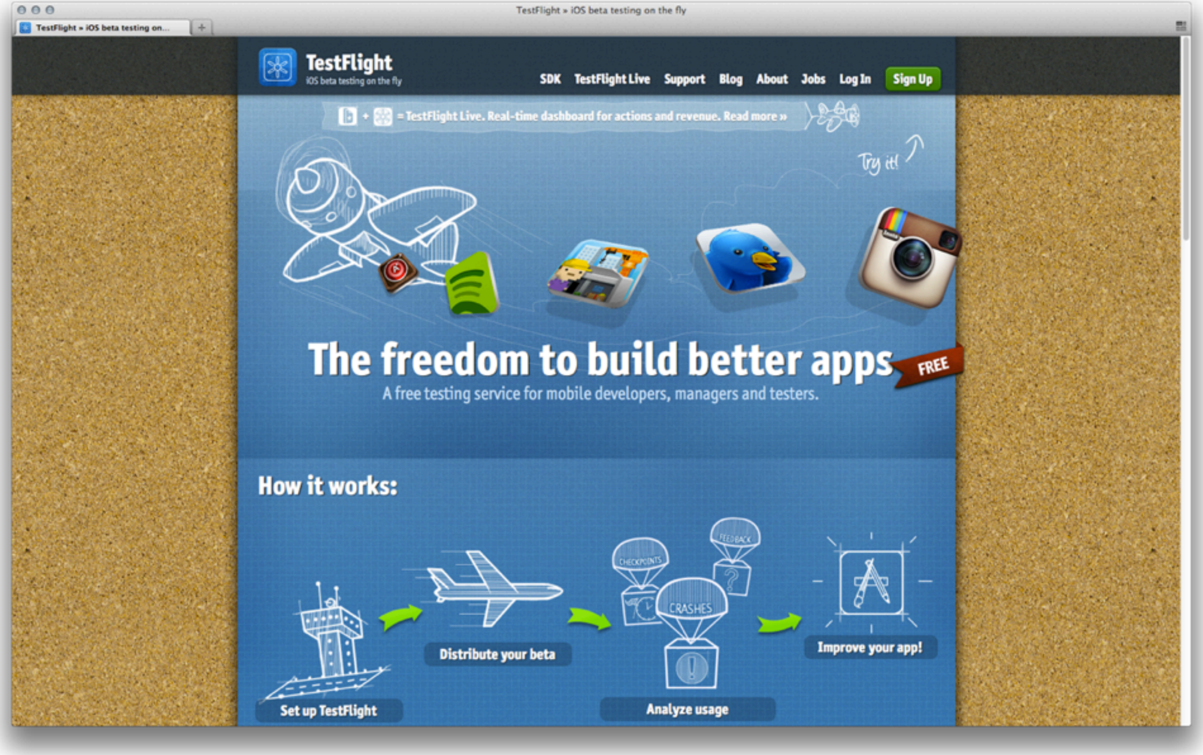
A/B testing



A/B testing

- Test better landing pages,
  - more efficient form design,
  - better conversion rates...
- Limitations
  - Does not replace user studies!
  - Does not provide explanation.
  - Arbitrary changes can be disturbing to existing users
  - Usually used to compare incremental changes (tricky to test complete re-designs).
- Tools:
  - Google Website Optimizer

Intelligently distributing betas



TestFlight » iOS beta testing on the fly

= TestFlight Live. Real-time dashboard for actions and revenue. Read more »

The freedom to build better apps

A free testing service for mobile developers, managers and testers.

How it works:

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let's do it!

Controlled experiments

- A scientific approach
  - Answering specific questions with data
    - Performance
    - Learning
    - Satisfaction
  - Providing basic knowledge generalizable across contexts.
  - Demonstrate causality between different factors
    - correlation: show that a change in A occurs with a change in B
    - order: show that A takes place before B
    - no hidden cause: show that there is no C with C -> A and C -> B

Examples

- Compare different input devices
- Compare gesture mechanisms
- Compare browsing mechanisms

Input devices

- From two weeks ago:

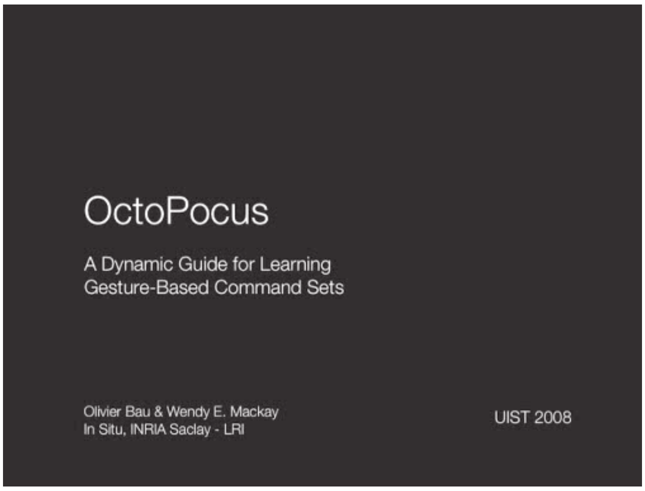
Input devices

- From two weeks ago:

Device	Study	IP (bits/s)
Hand	Fitts (1954)	10.6
Mouse	Card, English, & Burr (1978)	10.4
Joystick	Card, English, & Burr (1978)	5.0
Trackball	Epps (1986)	2.9
Touchpad	Epps (1986)	1.6
Eyetracker	Ware & Mikaelian (1987)	13.7

Gestures

- Octopocus: a better way do learn gestures



OctoPocus

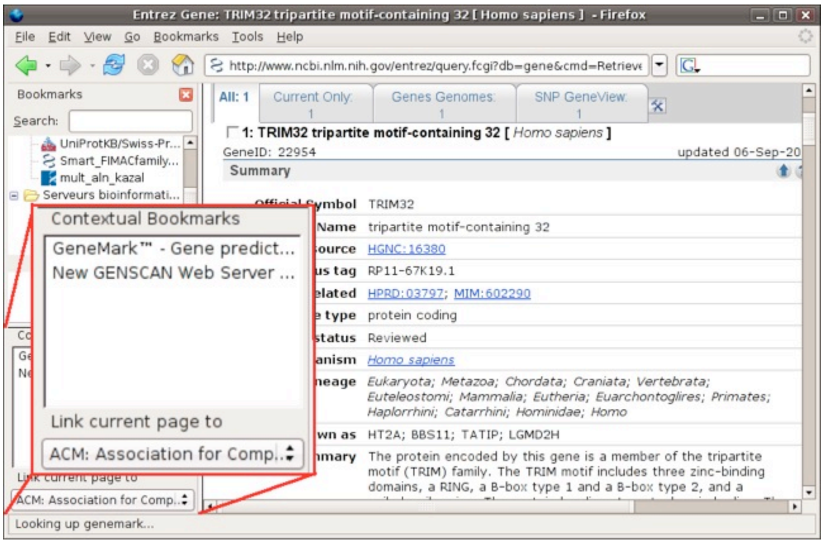
A Dynamic Guide for Learning
Gesture-Based Command Sets

Olivier Bau & Wendy E. Mackay
In Situ, INRIA Saclay - LRI

UIST 2008

Navigation

- PageLinker: improving web page revisitation



Entrez Gene: TRIM32 tripartite motif-containing 32 [Homo sapiens] - Firefox

http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene&cmd=Retrieve

1: TRIM32 tripartite motif-containing 32 [*Homo sapiens*]

GeneID: 22954 updated 06-Sep-20

Summary

Official Symbol TRIM32
Name tripartite motif-containing 32
ource HGNC:16380
us tag RP11-67K19.1
elated HPRD:03797; MIM:602290
e type protein coding
status Reviewed
anism *Homo sapiens*
neage *Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini; Catarrhini; Hominidae; Homo*
wn as HT2A; BBS11; TATIP; LGMD2H
mmary The protein encoded by this gene is a member of the tripartite motif (TRIM) family. The TRIM motif includes three zinc-binding domains, a RING, a B-box type 1 and a B-box type 2, and a

Contextual Bookmarks

GeneMark™ - Gene predict...
New GENSCAN Web Server ...

Link current page to

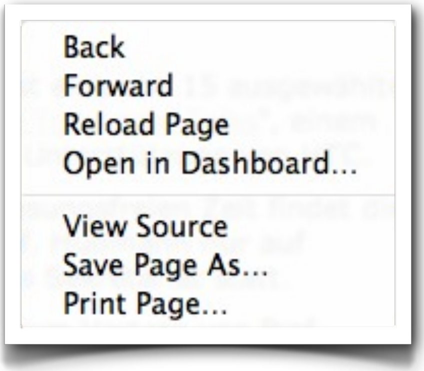
ACM: Association for Comp..

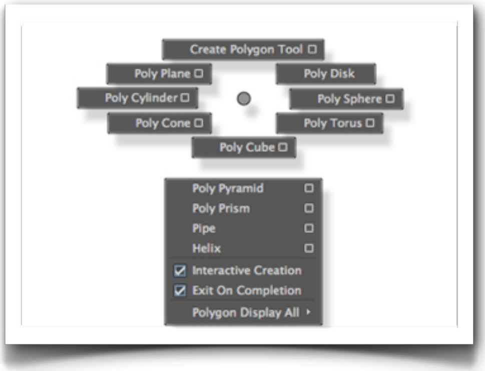
Looking up genemark...

Process

- Define what you are looking for
  – write down an experimental protocol
- Selected participants carry out the well-defined tasks
  – Make sure that you experiment respects the participants
- Run the experiment and gather data
- Analyze it and assess its significance
  – use statistical analyses

Example: comparing two menu designs





Data:

- Factors : **independent** variables
  - Variables we manipulate in each condition
- Levels (a.k.a. possible values for independent variables)
- Measures (or response) : **dependent** variable(s)
  - Outcomes of experiment
- Replication (number of subjects assigned to each level)

Independent variables (factors)

- The conditions of the experiment are set by independent variables
  - The number of items in a list, text size, font, color
- The number of different values used is the **level**
  - The number of experimental conditions is the product of the levels
  - E.g., font can be times or arial (2 levels), background can be blue, green, or white (3 levels). This results in 6 experimental conditions (times on blue, times, on green, ..., arial on white)

Dependent variables

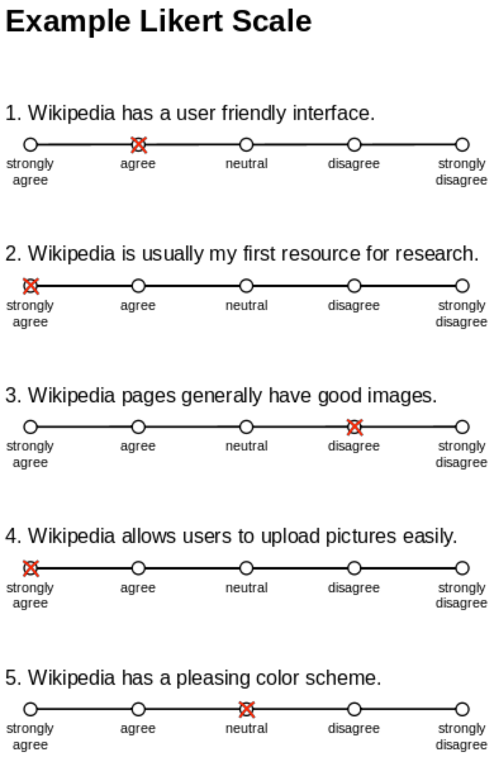
- The dependent variables are the values to be measured:
  - Objective values: e.g. time to complete a task, number of errors, etc.
  - Subjective values: ease of use, preferred option, etc.
  - They should only be dependent on changes of the independent variables.

Objective measures

- Measures (largely) independent from users' opinion:
- Examples:
  - Time
  - Errors
  - Steps to goal
  - Galvanic Skin Response

Subjective measures

- Measures dependent on users' opinions.
- Examples:
  - Likert scales
  - Questionnaires



Example Likert Scale

1. Wikipedia has a user friendly interface.

○ strongly agree ☒ agree ○ neutral ○ disagree ○ strongly disagree

2. Wikipedia is usually my first resource for research.

☒ strongly agree ○ agree ○ neutral ○ disagree ○ strongly disagree

3. Wikipedia pages generally have good images.

○ strongly agree ○ agree ○ neutral ☒ disagree ○ strongly disagree

4. Wikipedia allows users to upload pictures easily.

☒ strongly agree ○ agree ○ neutral ○ disagree ○ strongly disagree

5. Wikipedia has a pleasing color scheme.

○ strongly agree ○ agree ☒ neutral ○ disagree ○ strongly disagree

http://en.wikipedia.org/wiki/Likert_scale

Validity

- Internal validity
  - Manipulation of independent variable is cause of change in dependent variable
  - Requires removing effects of confounding factors
  - Requires choosing a large enough sample size, so the result couldn't have happened by chance alone.
- External validity
  - Results generalize to real world situations
  - Requires that the experiment be replicable
  - No study "has" external validity by itself!

Strategies

- Within-subjects design:
  - Same participant exposed to all test conditions
- Between-subjects design:
  - Independent groups of participants for each test condition

Randomization and control

- Control: holding a variable constant for all cases
  - Lower generalizability of results
  - Higher precision of results
- Randomization: allowing a variable to randomly vary for all cases
  - Higher generalizability of results
  - Lower precision of results
- Randomization within blocks: allowing a variable to randomly vary with some constraints
  - Compromise approach

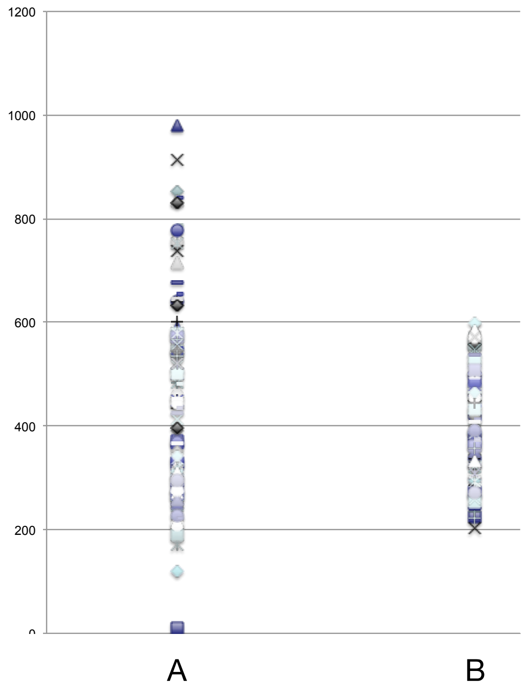
Hypotheses

- Prediction of the result of an experiment
- Stating how a change in the independent variables will affect the measured dependent variables
- With the experiment it can be tested whether the hypothesis is correct
- Usual approach
  - Stating a null-hypothesis (predicts that there is no effect)
  - Carrying out the experiment and using statistical measures to disprove the null-hypothesis
  - When a statistical test shows a significant difference it is probable that the effect is not random
  - Carefully apply statistical significance tests (see statistical methods)

Collecting data

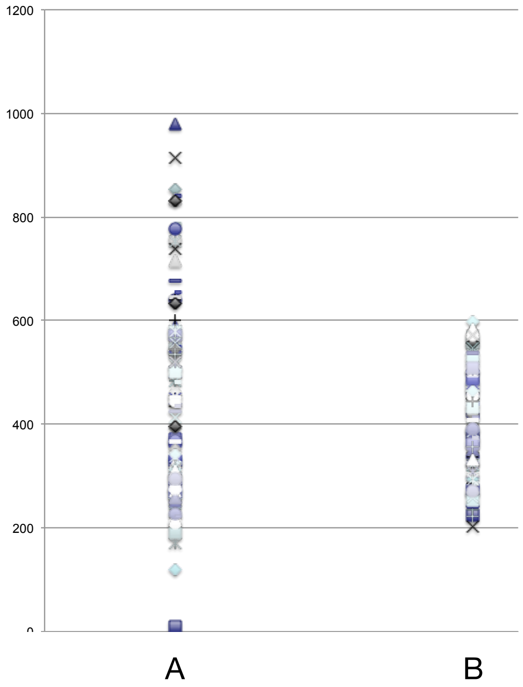
- process data
  - observations of what users are doing & thinking
- bottom-line data
  - summary of what happened (time, errors, success)
  - i.e., the dependent variables

Is A faster than B?



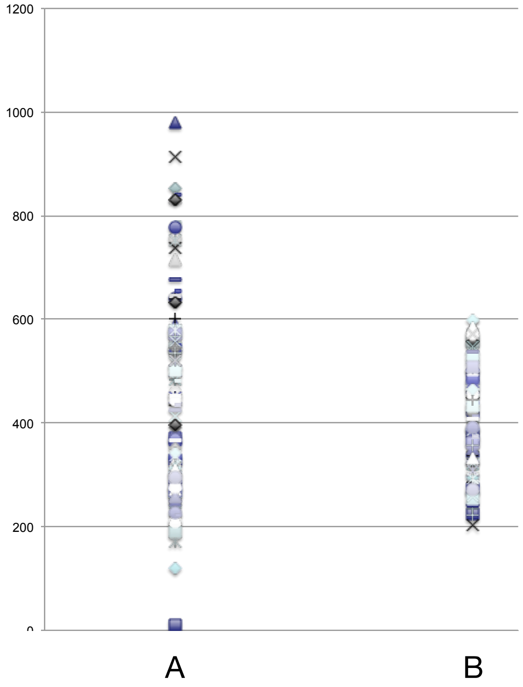
Is A faster than B?

- are these two means significantly different?



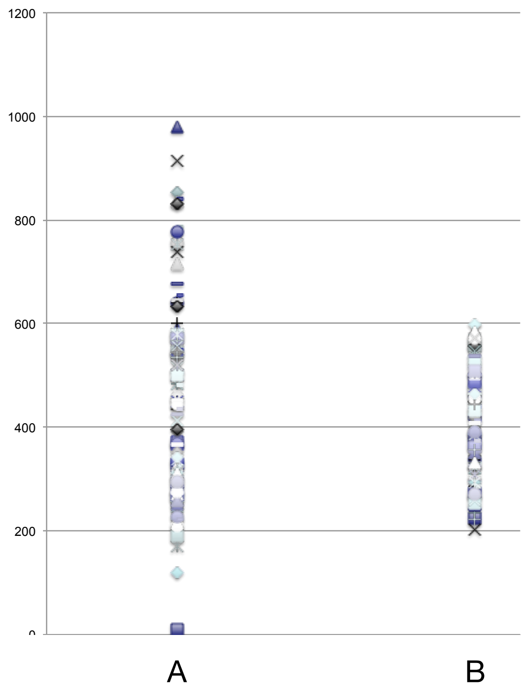
Is A faster than B?

- are these two means significantly different?
- depends on difference between means



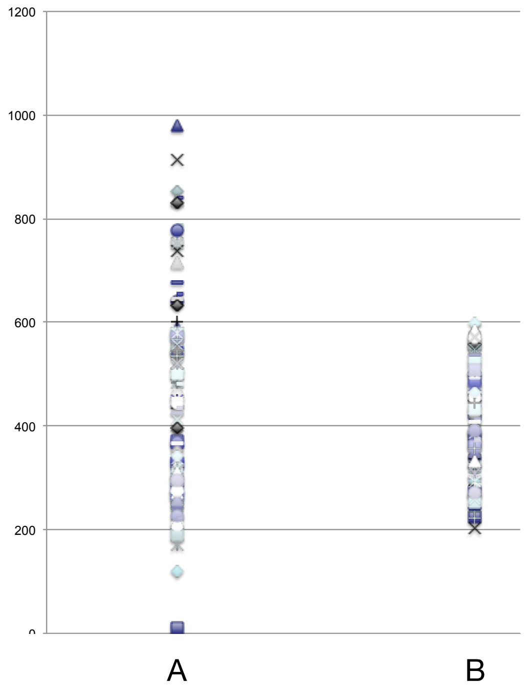
Is A faster than B?

- are these two means significantly different?
- depends on difference between means
- depends also on spread (i.e. standard deviation)



Is A faster than B?

- are these two means significantly different?
- depends on difference between means
- depends also on spread (i.e. standard deviation)
- depends also on sample size

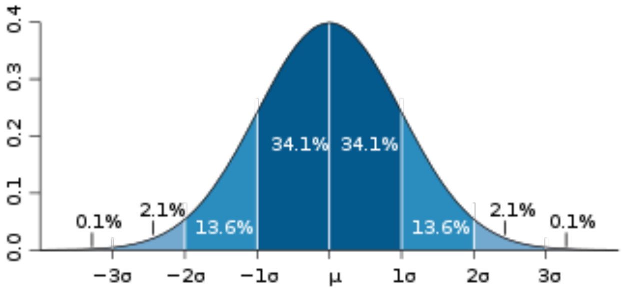


Is A faster than B?

- are these two means significantly different?
- depends on difference between means
- depends also on spread (aka standard deviation)
- depends also on sample size

(Student's) t-test

- Looks at the relationship between two data sets
- Designed for
  - – small sample (= few measurements)
  - – unknown (mean and) standard deviation
  - – but has to be normally distributed



t-test

- Gives *p*: the probability (i.e., 0 < p < 1) you got the difference between two data sets is due to chance
- A low probability (< 0.05) means “unlikely that this difference in means was the result of chance reject null hypothesis”
- The risk of erroneously rejecting the null hypothesis (= supporting the hypothesis) is less than percentage *p*.
- In our field usually 0.05 (= 5% chance).

PLEASE DON’T

- If p>0.05 say:
  - “our tests showed that there was no difference”.

  - significant difference -> impact
  - no significant difference -> nothing

- You cannot show that there is no difference!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

Evaluation and testing

- Introduction
- Approaches to evaluation
- Analytical Methods
- Empirical Methods
- Evaluation 2.0 : scaling up
- Experimental Design
- Let’s do it!

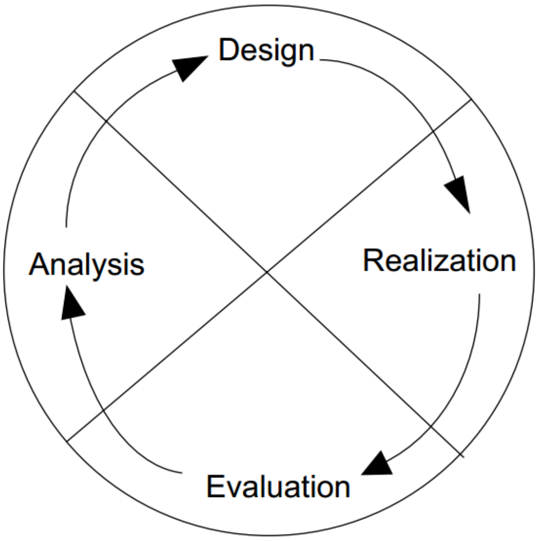
Breakoutsession No. 6

Evaluation

Evaluation

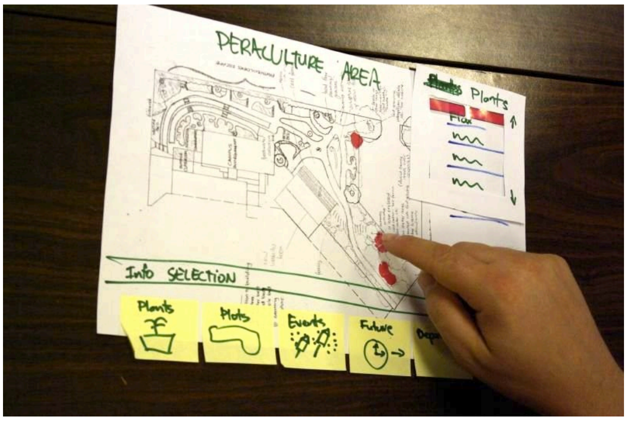
- analytical evaluation methods:
  - – model-based evaluation:
    - GOMS
    - KLM
  - – inspection-based evaluation:
    - Cognitive Walkthrough
    - Inspections & Expert Review
    - Ten Usability Heuristics (Nielsen)

- empirical evaluation methods:
  - – usability evaluation
  - – field study
  - – controlled experiment



Cognitive Walkthrough

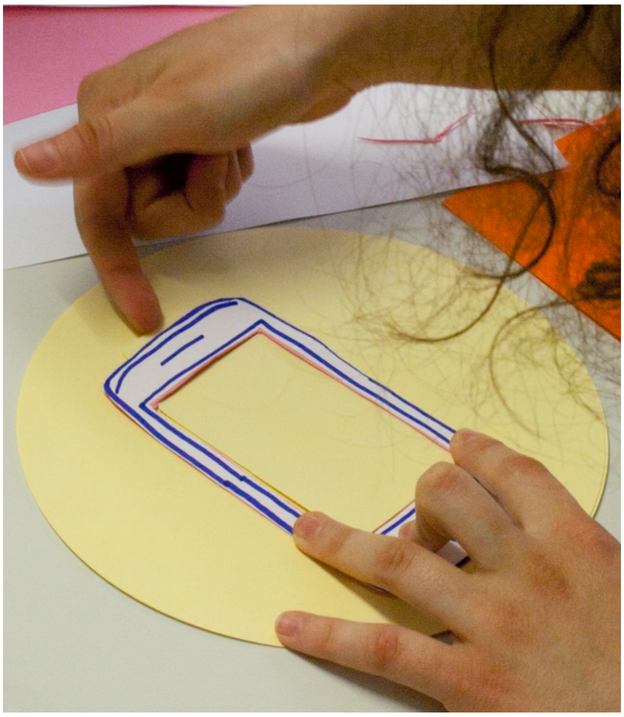
- goal:
  - – evaluate understandability
- method:
  - – evaluator goes through a set of tasks
- procedure:
  - – defining the input:
    - who will be the users?
    - what tasks will be analyzed?
    - what is the correct action sequence for each task?
  - – during the walkthrough:
    - will the users try to achieve the right effect?
    - will the users notice that the correct action is available?



http://commons.wikimedia.org/wiki/File:ELiving_Campus_Paper_Prototype_2.jpg

Task

- prepare for a cognitive walkthrough:
  - – define tasks to be analysed and the correct action sequence for each task
  - – time: 5 min
- choose one person of your group who tests the prototype of the group next to you
- do a cognitive walkthrough:
  - – write down the results of the test



Homework

- iterate the design process:
  - build an improved prototype
- prepare for a presentation:
  - 5 minutes, not too many slides ☺
  - slides should contain :
    - » explanation of your concept
    - » first prototype (pictures, annotations)
    - » findings of evaluation
    - » improved prototype (bring it with you)
  - send it via email to sebastian.loehmann@ifi.lmu.de
  - file format: PDF
  - deadline: Monday, 25.06.2012 – 24:00
  - date of presentation: Wednesday, 27.06.2012