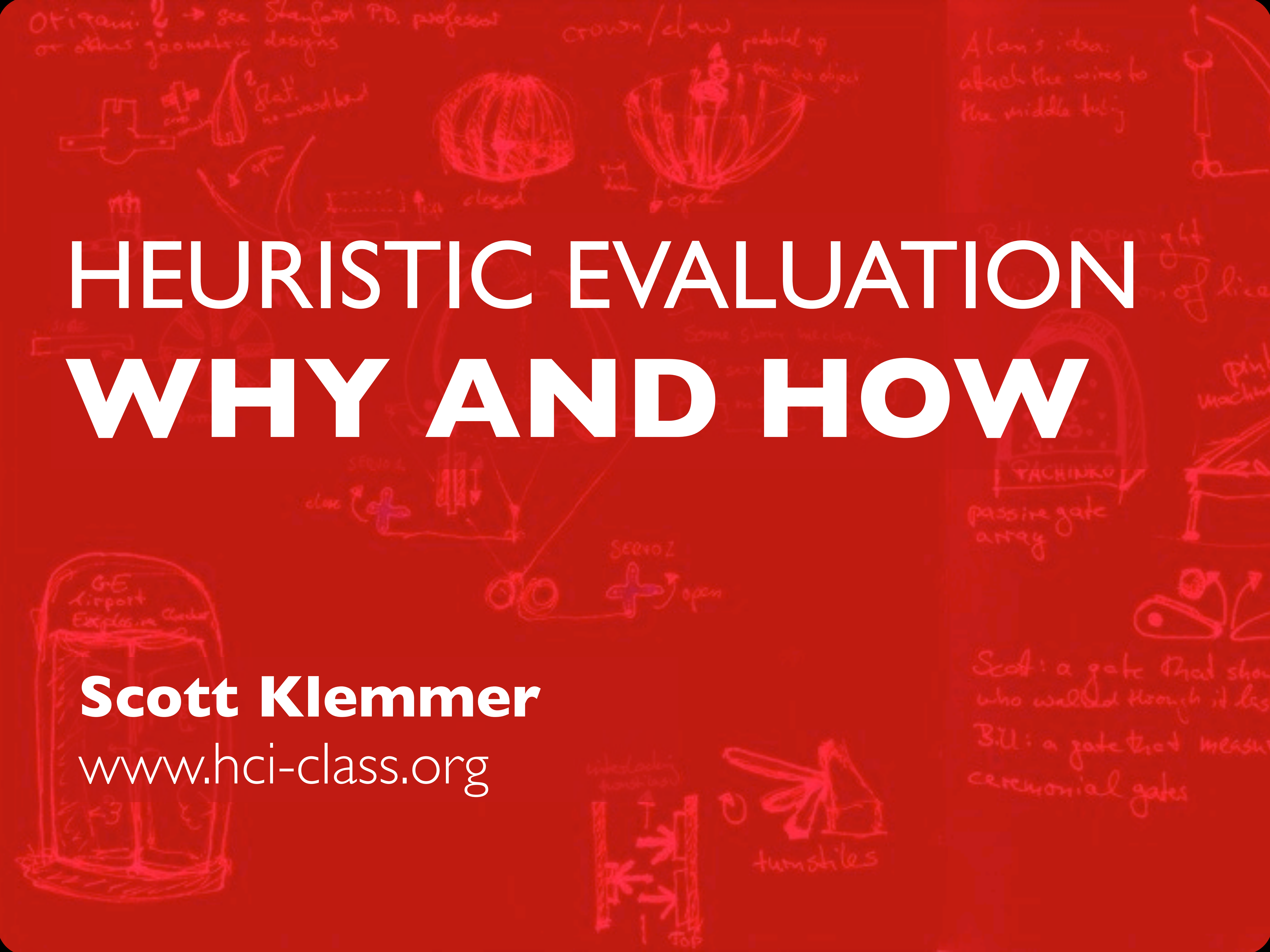


HEURISTIC EVALUATION

WHY AND HOW

Scott Klemmer

www.hci-class.org

Multiple ways to evaluate

Empirical	Assess with real users
Formal	Models and formulas to calculate measures
Automated	Software measures
Critique	Expertise and heuristic feedback

When to get design critique?

- **Before user testing.** Don’t waste users on the small stuff. Critique can identify minor issues that can be resolved before testing, allowing users to focus on the big issues.
- **Before redesigning.** Don’t throw out the baby with the bathwater. Critique can help you learn what works and what should change.
- **When you know there are problems, but you need evidence.** Perhaps you've received complaints from customers or found yourself stumbling around your own site. Critique can help you articulate problems and provide you with ammunition for redesign.
- **Before release.** Smooth off the rough edges.

http://**www.etre.com/usability/inspection**

Begin Review with a Clear Goal

Heuristic Evaluation

- Developed by Jakob Nielsen
- Helps find usability problems in a design
- Small set (3-5) of evaluators examine UI
    - independently check for compliance with usability principles (“heuristics”)
    - different evaluators will find different problems
    - evaluators only communicate afterwards
        - findings are then aggregated
- Can perform on working UI or sketches

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Ten Design Heuristics

- Show system status
- Familiar metaphors & language
- Control & freedom
- Consistency
- Error prevention
- Recognition over recall
- Flexibility & efficiency
- Aesthetic & minimalist design
- Recognize, diagnose, & recover from errors
- Help

My somewhat revised names for Jacob Nielsen's heuristics, http://www.useit.com/papers/heuristic

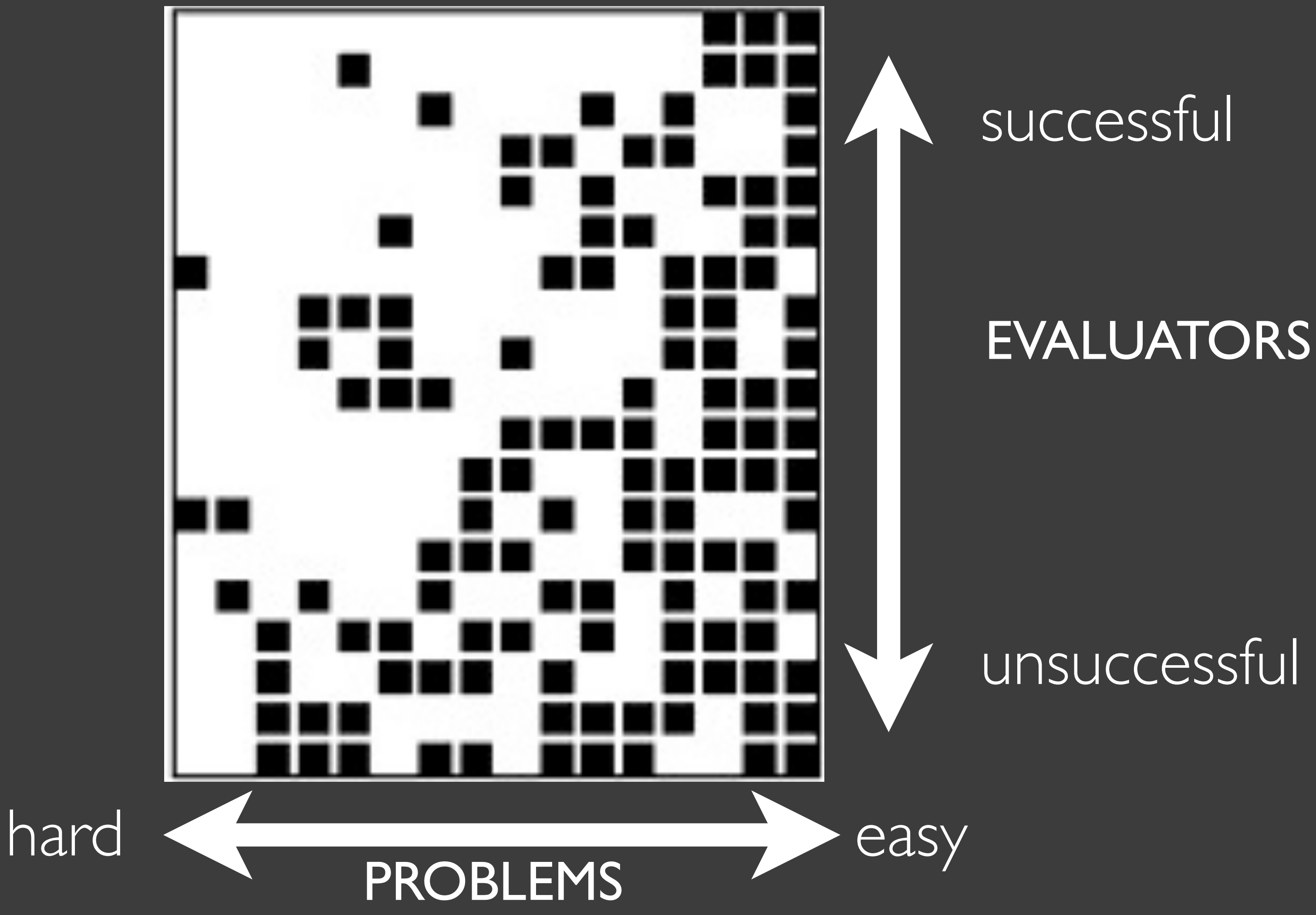
Evaluators’ Process

- Step through design several times
  - Examine details, flow, and architecture
  - Consult list of usability principles
  - ..and anything else that comes to mind
- Which principles?
  - Nielsen’s “heuristics”
  - **Category-specific heuristics** from e.g., design goals, competitive analysis, existing designs
- Use violations to redesign/fix problems

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Why Multiple Evaluators?

- No evaluator finds everything
- Some find more than others



Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Decreasing Returns

- Caveat: graphs for a specific example

problems found

benefits / cost

Heuristic Eval: Cost-effective

- In one case: benefit-cost ratio of 48
  - estimated benefit $500,000; cost $10,500
  - value of each problem ~$15K
  - how might we calculate this value?
    - in-house -> productivity; open market -> sales
- Severe problems found more often
- Single evaluator achieves poor results
  - only finds 35% of usability problems
  - 5 evaluators find ~ 75% of problems

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Heuristics vs. User Testing

- Heuristic Evaluation often faster
    - 1-2 hours each evaluator
- HE results come pre-interpreted
- User testing is more accurate (by def.)
    - takes into account actual users and tasks
    - HE may miss problems & find "false positives"
- Valuable to alternate methods
    - find different problems
    - don't waste participants

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Phases of Heuristic Evaluation

1. **Pre-evaluation training:** give evaluators needed domain knowledge and information on the scenario
2. **Evaluation:** individuals evaluate and then aggregate results
3. **Severity rating:** determine how severe each problem is (priority). Can do first individually and then as a group
4. **Debriefing:** review with design team

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

How-to: Heuristic Evaluation

- At least two passes for each evaluator
  - first to get feel for flow and scope of system
  - second to focus on specific elements
- If system is walk-up-and-use or evaluators are domain experts, no assistance needed
  - otherwise might supply evaluators with scenarios
- Each evaluator produces list of problems
  - explain why with reference to heuristic or other information
  - be specific and list each problem separately

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

How-to: Heuristic Evaluation

- Why separate listings for each violation?
  - risk of repeating problematic aspect
  - may not be possible to fix all problems
- Where problems may be found
  - single location in UI
  - two or more locations that need to be compared
  - problem with overall structure of UI
  - something is missing
    - ambiguous with early prototypes; clarify in advance
    - sometimes features are implied by design docs and just haven't been "implemented" – relax on those

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Severity Rating

- Independently estimate after review
- Allocate resources to fix problems
- Estimate need for more usability efforts
- Severity combines
    - frequency
    - impact
    - persistence

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Severity Ratings

0 - don’t agree that this is a usability problem

1 - cosmetic problem

2 - minor usability problem

3 - major usability problem; important to fix

4 - usability catastrophe; imperative to fix

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Severity Ratings Example

- *Issue*: Unable to edit one's weight
- *Severity*: 2
- *Heuristics violated*: User control and freedom
- *Description*: when you open the app for the first time, you have to enter your weight, but you cannot update it. It could be useful if you mistyped your weight, or if one year or two after the first use of the app, your weight has changed.

Courtesy Sebastien Robaszkiewicz, cs147 2011

Debriefing

- Conduct with evaluators, observers, and development team members
- Discuss general characteristics of UI
- Suggest potential improvements to address major usability problems
- Dev. team rates effort to fix
- Brainstorm solutions

Jacob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay