Usability Testing

Nico Zazworka

Motivation

- Your GUI or web application might be
  - Correct
  - Robust, Reliability
  - Fast enough
- But if your users cannot interact with it in an efficient way they won't use it!
- Usability Testing
  - A method to find usability faults

Testing for Correctness vs. Testing for Usability

	Correctness	Usability
Subject	Program, Code, Algorithm	User interface
Input	Testcases: configuration of parameters or sequence of events	Tasks and users
Goal	Find correctness faults	Find obstacles
Measures	# of faults, types of faults	Efficiency, accuracy, recall, emotional response, heuristics
Action/solution	Correct faults in program	Improve user interface

How to validate a user interface

- Formally
  - Meeting of usability experts to discuss merits and weaknesses
  - “Usability experts” in ideal case not the UI developers
  - No real users involved
- Empirically
  - Set up an experiment with a set of hypothesis to show that UI A is better than UI B
  - Expensive – eventually lots of subjects needed
  - Eventually replication needed
  - How to choose users? How many? What tasks? What quantitative and qualitative measures are we taking?
- Heuristically
  - Use a small set of users (not experts), give them some guidelines (heuristics) and ask them about their opinion
  - Goal: find usability problems
  - Critique: what if we choose the wrong users?

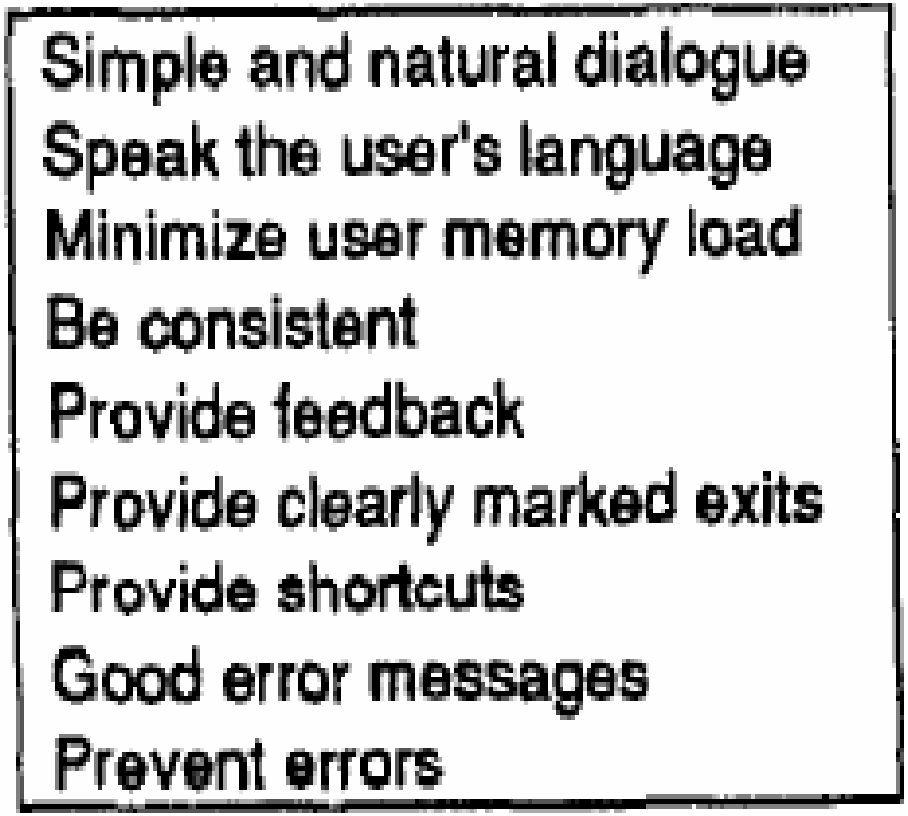
Heuristic Evaluation

- Nielsen and Molich 1990 study
- Process:
  - Subjects are looking at interfaces and note down what is good and bad
  - According to a very small set of rules or heuristics (in this case nine)
  - Subjects are not usability experts
- Empirical study with students as users and four applications

Experiment (short name)	*No. Evalu-ators*	*Total Known Usability Problems*	*Average Problems Found*
Teledata	37	52	51%
Mantel	77	30	38%
Savings	34	48	26%
Transport	34	34	20%

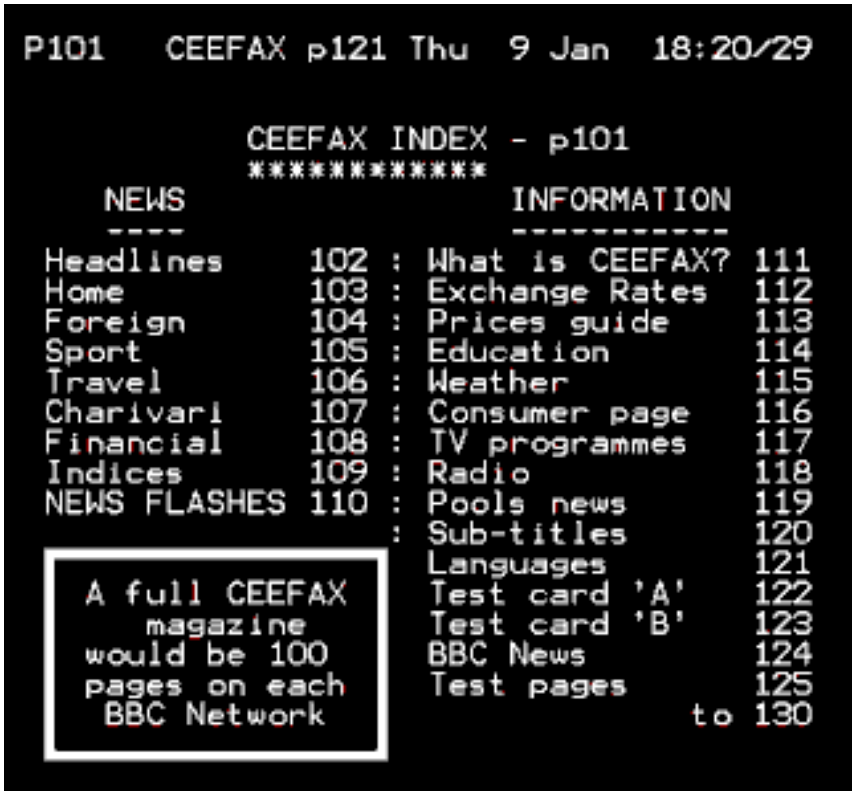
Table 2. Summary of the four experiments.

Heuristics


Simple and natural dialogue
Speak the user's language
Minimize user memory load
Be consistent
Provide feedback
Provide clearly marked exits
Provide shortcuts
Good error messages
Prevent errors

Teledata

- Screenshots of 10 pages
- 2 Systems:
  - General search
  - Airline search
- 52(!) usability problems



Mantel

T E L E P H O N E I N D E X

Telephone number (212) 345-6789 has the following subscriber:

Jim E. Jones
17 Pine Street
New York, NY 10012

Press:
RETURN to be able to enter a new telephone number
ESC to leave the Telephone Index
PF1 to get Help about how to use this system
PF2 to go to the Directory Information system
PF4 to go to the general Videotex service
PF5 to get a list of Other Services available

Savings & Transport

- Two voice response systems
- If you think this is out of date think about
  - Your voicemail
  - Voice recognition software
  - Blind users

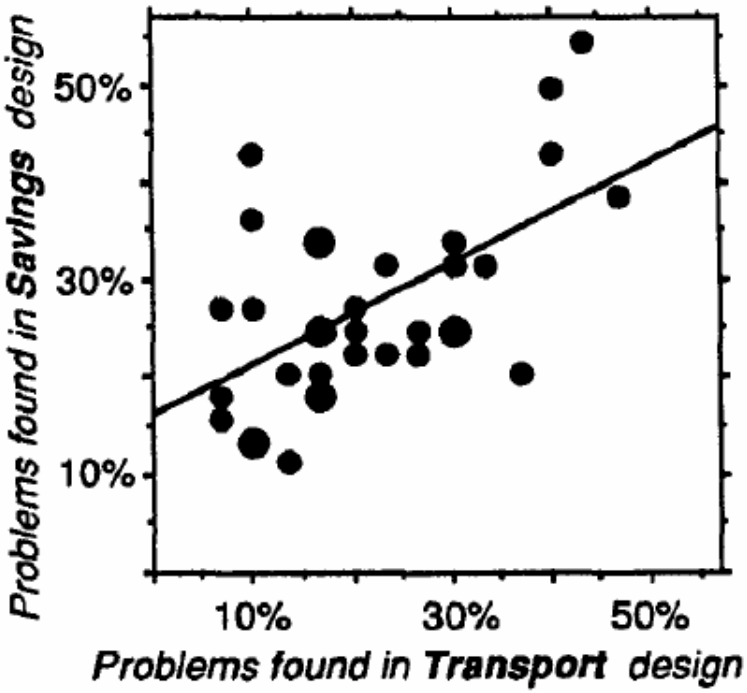


Figure 1. *Scatterplot of the proportion of usability problems found by the same evaluators in two different interfaces. The regression line has* R^2*=0.33 and shows that there is only a very weak correlation between the evaluators' performance in the two experiments.*

Process

- The authors reports a set of usability problems for each application
  - That they had to extend based on the one found by the students
- False positives
  - Can cause discussion if it is a problem or not
  - Very few occurred in study and only each of them reported by one finder
- False negatives
  - In general is impossible to find all usability problems
  - Some problems are reported by only one evaluator
  - To find all problems all possible users have to go through this process
    - Infeasible in most cases (e.g. web applications, office applications)

Results

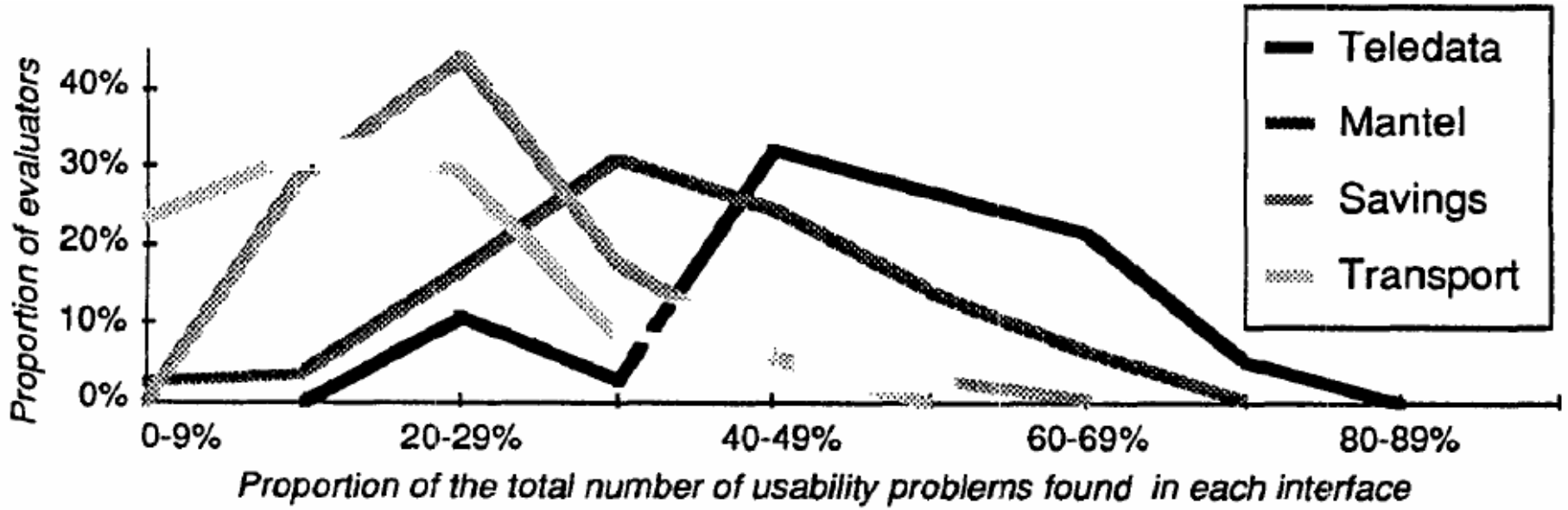
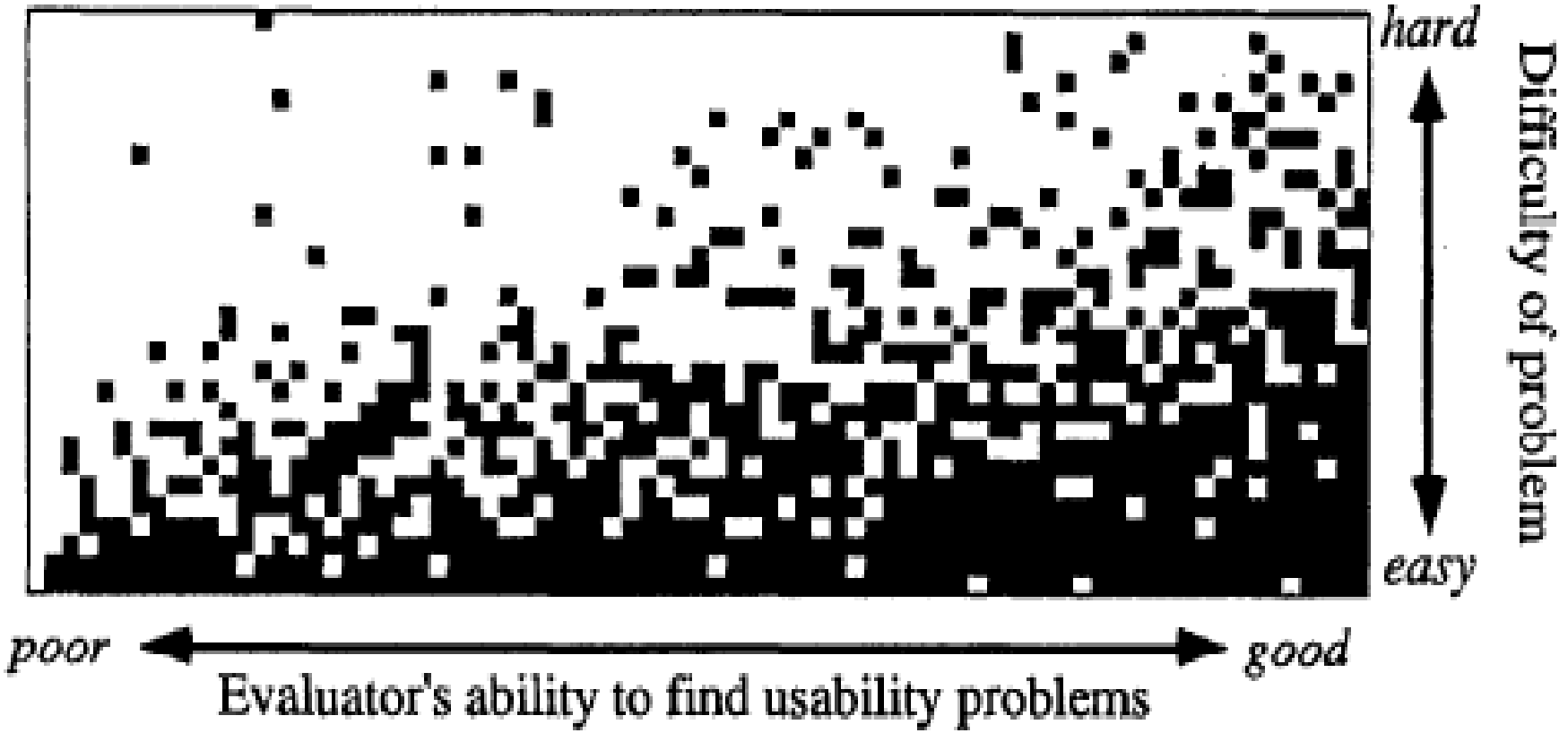


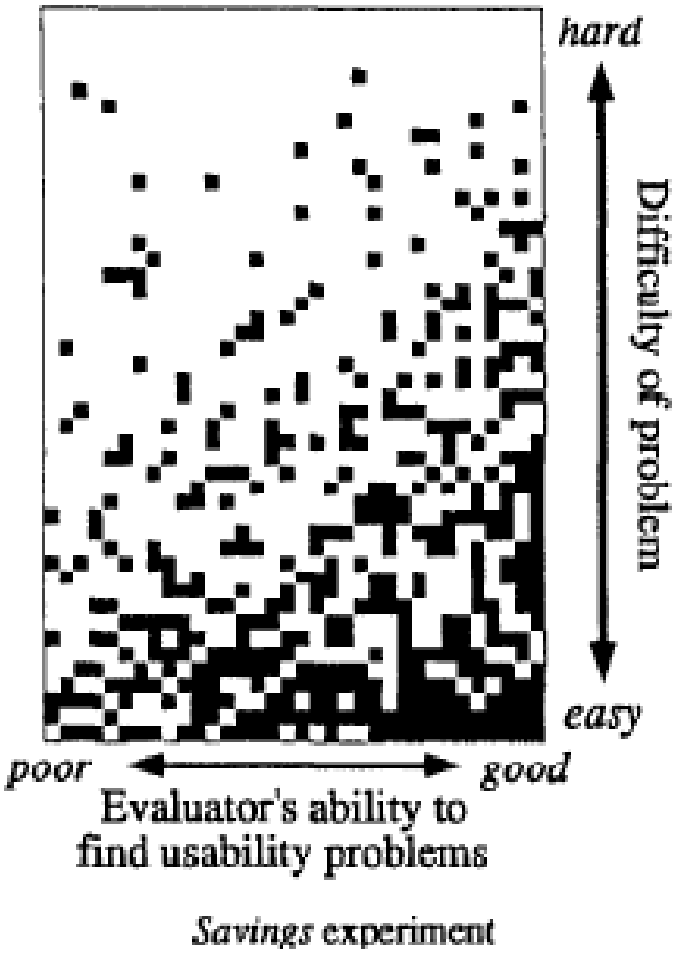
Figure 2. *Distribution for each of the four experiments of the number of usability problems found by the evaluators (expressed as percent of the total number of problems in each interface to enable comparisons).*

Interesting Results



Mantel experiment

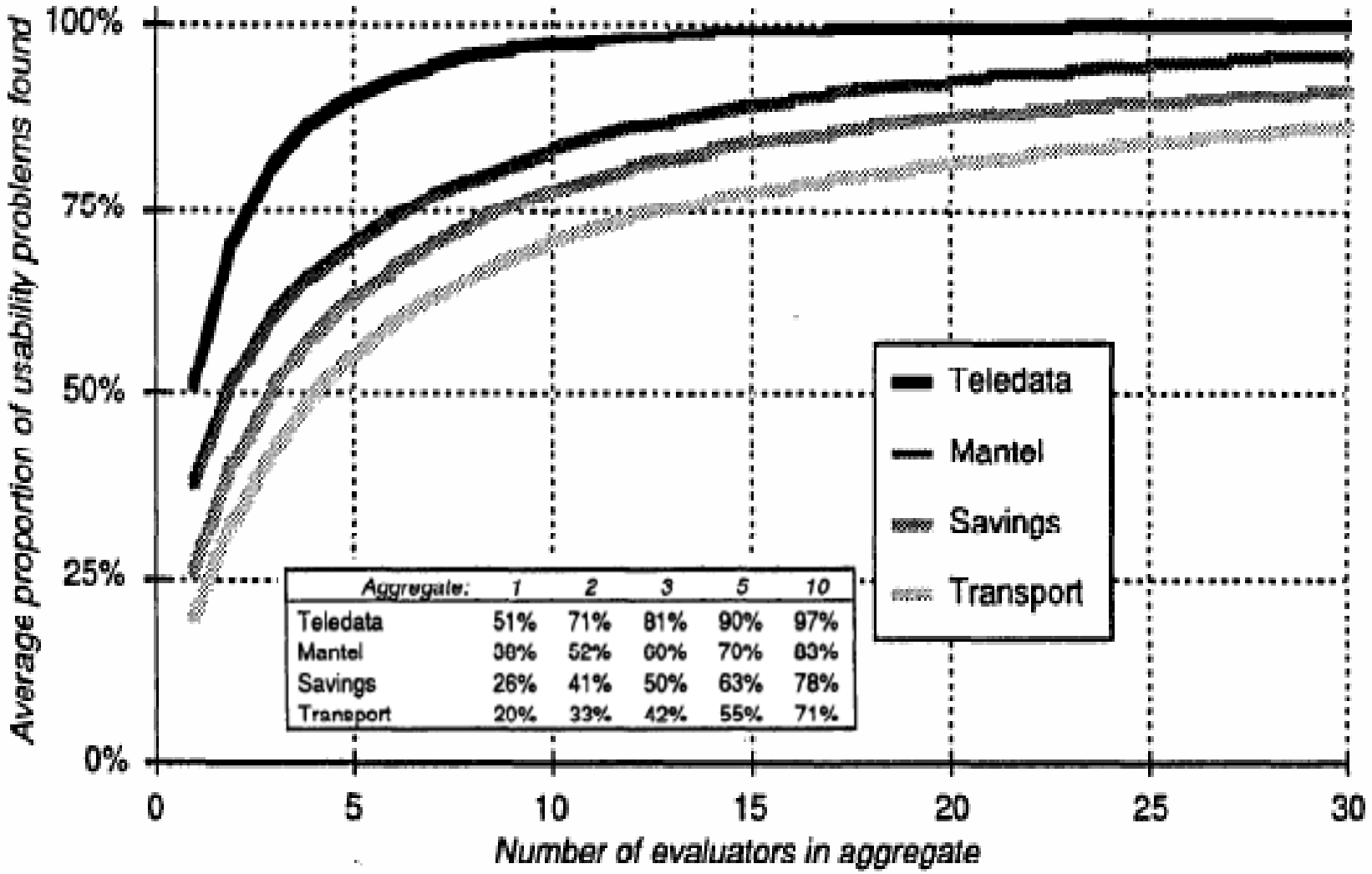
Interesting Results (2)



Savings experiment

- Each of the evaluators finds a different set of usability problems
- Like having test cases that find very different sets of faults
- Idea: how many test cases do I have to pick to find enough or all faults?

Monte Carlo Simulation



Aggregate:	1	2	3	5	10
Teledata	51%	71%	81%	90%	97%
Mantel	38%	52%	60%	70%	83%
Savings	26%	41%	50%	63%	78%
Transport	20%	33%	42%	55%	71%

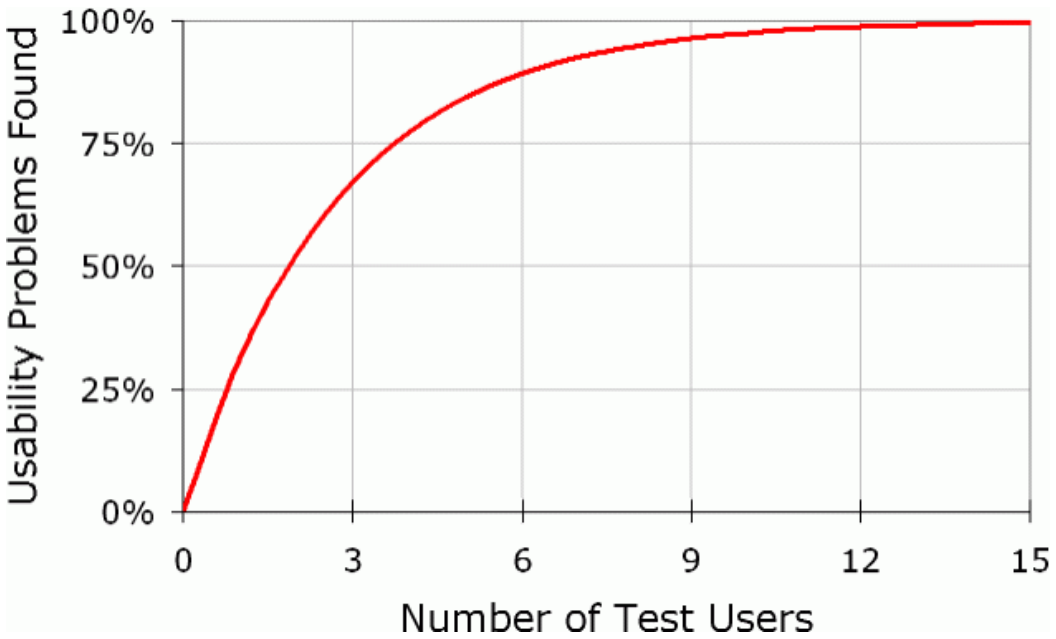
Figure 4. *Proportion of usability problems found by aggregates of size 1 to 30.*

Conclusions

- Having a single person look at the interface is not a good idea:
  - There are poor and good evaluators
  - Even the good ones oversee easy problems
- Number of usability faults found grows rapidly in the interval from 1 - 5 evaluators
- 5 is sometimes enough
- Technique is:
  - Cheap
  - Intuitive and easy to learn
  - Can be used early in the development process
- Technique does not suggest solutions to the problems and therefore not generate breakthroughs in the evaluated design.

Later work

- Formalized it further:
  - Nielsen, Jakob, and Landauer, Thomas K.: "A mathematical model of the finding of usability problems," *Proceedings of ACM INTERCHI'93 Conference* (Amsterdam, The Netherlands, 24-29 April 1993), pp. 206-213.
- *N*(1-(1-*L*)^*n*)
  - where *N* is the total number of usability problems
  - *L* is the proportion of usability problems discovered while testing a single user
  - n number of users
  - Typical value of L = 31%



Critiques in 2001

- Spool and Schroeder: “Testing Web Sites: Five Users Is Nowhere Near Enough”
- Replicated the experiment on 4 shopping websites
- After 5 users only 35% of problems captured
- Today’s systems more complex: coverage is an important factor

	Site	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
All	I	14	6	8	7	12	9	12	7	11	14	7	8	6	10	7	4	8	14
New	I	14	5	7	5	7	5	11	6	9	10	5	5	2	8	3	1	2	9
All	II	15	5	2	7	6	12	5	6	1	13	6	6	1	7	3	6	2	7
New	II	15	4	1	5	2	9	5	2	0	8	2	4	0	2	2	3	1	1
All	III	6	7	4	11	7	10	5											
New	III	6	7	1	9	5	6	2											
All	IV	6	7	17	11	6	7												
New	IV	6	5	13	6	5	2												

Table 1. Obstacles Found By Test

Questions?