Historic note

What happened to Alias?

This presentation describes work that was done by the Alias User Experience team, based in Toronto, Canada.

After this presentation was accepted for UPA 2006, Alias was acquired by Autodesk, Inc. of San Rafael, California.

For that reason, the company name listed in your program will not match what you see on this screen.

Sorry for any confusion.
Agenda
for the next 1.5 hours

Background
What makes formative usability testing different?
What are the steps?
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, discussion

Ground Rules
for this presentation

Ask questions or make comments at any time
This is about sharing best practices, so please share yours
Agenda
where we are

Background
What makes formative usability testing different?
What are the steps?
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, discussion

Background
There are many user-centred design activities

- Ethnographic research
- Contextual inquiry
- Task analysis
- Creating personas
- Creating use-case scenarios
- Interaction design
- Graphic design
- Heuristic evaluation
- Usability testing
- Support log analysis
- Focus groups
- …
Background
There are many user-centred design activities

- Ethnographic research
- Contextual inquiry
- Task analysis
- Creating personas
- Creating use-case scenarios
- Interaction design
- Graphic design
- Heuristic evaluation
- Usability testing
- Support log analysis
- Focus groups
- …

Background
Usability Testing can be used at different times

- To improve the usability/learnability of interaction prototypes before they are passed on to development for implementation.
- To verify the usability/learnability of the implemented feature after it is built.
- To find out if there are any usability problems in the feature when it is placed in the larger context of the program or system.
- To compare the usability of different programs or designs.
- To determine the overall usability of a complete system.
Background

…but the goals of early and late testing are different

- To improve the usability/learnability of interaction prototypes before they are passed on to development for implementation.

- To verify the usability/learnability of the implemented feature after it is built.

- To find out if there are any usability problems in the feature when it is placed in the larger context of the program or system.

- To compare the usability of different programs or designs.

- To determine the overall usability of a complete system.

Background

Formative and Summative Testing

**Formative** Usability Testing is done iteratively at the prototyping stage, to help guide the design.

**Summative** Usability Testing is done later, to measure the usability of a completed component or system, and to identify problems in it.

Although these two tasks have different goals and needs, the industry for the most part prescribes a common methodology for both.

But is that right?
Background

Our problem: faster, faster!

Some years ago, my company Alias adopted many Agile development methodologies.

The User Experience Team needed to become more agile, too. We needed to develop ways to go from requirements to finished, validated UI designs in one agile ‘cycle’.

We design highly interactive and innovative graphic interfaces, so we need a lot of iteration to get it right.

footnote

[If you are interested in the details of how we do Agile User-Centred Design, you can read this paper:

Miller, L.  A Case Study of Customer Input for a Successful Product.  AGILE 2005 Conference, Denver, Colorado, USA. ]
Background
What’s wrong with the methodology?

Regular usability testing methodology was too slow and cumbersome for our needs.

The turnaround time was too long.

It took too long to write reports -- even short ones.

Big initial problems completely hid many other problems.

Fixing a big problem could change the user’s response to many other things, invalidating much of the data collected in previous test runs.

Much of the information we collected never got used.

Background
Looking Around

What were other people doing to make formative usability testing better and faster?

In the early nineties, there was a lot of research done on using fewer test participants.
Background
1992: Using Fewer Test Subjects


Ideas from this paper:

- More test participants gave decreasing levels of return
- Participants found the most important problems first.
- The first 5 participants found 80% of the problems

Background
1993: Determining the Optimal Number of Testers

Background

1994: Further considerations

Lewis, J.R. *Sample Sizes for Usability Studies: Additional Considerations.* Human Factors 36, p. 368-378

Ideas from this paper:

- The number of testers you need depends on the probability of a tester finding a given problem.
- Highly-usable interfaces require more testers to find a given percentage of problems.

Background

So that’s it, then?

Others got into the fray --- for example, Jared Spool argued that you need many more testers for some web site, because of the huge diversity of potential users.

But we had good models for determining how many testers you need to uncover a given percentage of the usability problems of a given system.

But for *formative* usability testing, is that actually what you want to do?
Background

Things that made us go hmmm

In formative usability studies, is the goal to find some minimal percentage of all the existing problems in a given prototype?

This is a good goal, if the usability problems you find are independent of one another --- that is, fixing one will still leave all the others.

We didn’t observe that to be the case.

Background

2002: The RITE method


Ideas:

- Using only one or two testers before changing the interface and re-testing

- “…more important to get the team to fix problems and to determine the likelihood that a ‘fix’ had solved a problem, than to agonize over if every problem had been uncovered.”

- This practice was neither new nor unique.
Background
2003: We admit it, too.


Mentioned:
- Changing the prototype after only one or two testers, *if it failed the design goal*.

Background
2005: Formative testing has different needs

Theofanos, M., Quesenbery, W. *Reporting Formative Test Results*, A Workshop at UPA 2005 (Montreal, Canada), June 27 to July 1, 2005.

Mentioned:
- The Common Industry Format (CIF) for reporting usability test results has been an ANSI/INCITS Standard since 2001. But it’s *not suitable for formative testing* without modification.
- Need to get rid of information that does not immediately impact design decisions, or at least move it to the end.
- Need to include additional data: “how” and “why” of events.
- Don’t write a report if no one will read it.
Background
2005: The Masking Problem


Mentioned:
- In early stages, there is no point in testing more than five or six people, because the major problems you find will mask the other problems.
- “…a properly conducted test will identify the major usability issues…as well as provide direction for design along the way.”
- “The most important thing is to incorporate those findings into the design so that a better version can be tested later.”

Background
2005: Sometimes one user is enough


Mentioned:
- There are certain circumstances when it is appropriate to report a problem you’ve seen only once, such as:
  - Face Validity: “Some problems are obviously valid the moment you see them.”
  - Corroborating evidence
Background
Some terminology for this presentation

We spent some time formalizing the differences between Formative and Summative testing, in terms of their properties and our practice of them.

**Formative** usability testing is used as a high-efficiency driver of interaction design.

**Summative** usability testing is used to examine and measure the usability and learnability of a design or implemented system.

Each kind of testing has its own **goals, needs, and best practices**.

---

Background
Some terminology for this presentation

Sometimes I will be comparing our past practices to our current practices.

- **“Light” Formative Testing** refers to our current practice, or similar practices such as the RITE method.

- **“Heavier” Formative Testing** refers to the more traditional way of doing formative usability tests.
Background

the end of this section

Questions? Stories?

Agenda

where we are

Background
What makes formative usability testing different?
What are the steps?
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, discussion
Formative Usability Testing

Why should it be different?

Many common practices of usability testing are not suitable for driving early prototype design.

Why?

- They can take too long
- They collect information that is not needed
- They lose information that is needed: *behavioural insight*

Formative Usability Testing

The differences

**Used When:**

<table>
<thead>
<tr>
<th>Formative Testing</th>
<th>During early design and prototyping phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summative Testing</td>
<td>During late development or post-development</td>
</tr>
</tbody>
</table>
Formative Usability Testing
The differences

**Goal:**

<table>
<thead>
<tr>
<th>Formative Testing</th>
<th>Maximize iterations to hone in on the best achievable design.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>“How usable can we make this product?”</td>
</tr>
<tr>
<td>Summative Testing</td>
<td>Find and measure all usability problems.</td>
</tr>
<tr>
<td></td>
<td>“How usable is this product?”</td>
</tr>
</tbody>
</table>

**Inputs:**

<table>
<thead>
<tr>
<th>Formative Testing</th>
<th>Fastest/cheapest prototype that still captures basic behaviours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summative Testing</td>
<td>Finished product (or feature)</td>
</tr>
</tbody>
</table>
Formative Usability Testing
The differences

**Outputs:**

<table>
<thead>
<tr>
<th>Formative Testing</th>
<th>Interaction design verified to be complete and correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summative Testing</td>
<td>Report, containing found problems, their severity, possible statistical analyses (task time, error recovery), and recommendations for improvement</td>
</tr>
</tbody>
</table>

Formative Usability Testing
The differences

**What to do between test participants:**

<table>
<thead>
<tr>
<th>Formative Testing</th>
<th>The prototype may be updated, but the test protocol remains unchanged</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summative Testing</td>
<td>The prototype and protocol remain unchanged</td>
</tr>
</tbody>
</table>
## Formative Usability Testing

**The differences**

**Overall:**

<table>
<thead>
<tr>
<th>Formative Testing</th>
<th>Finds solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summative Testing</td>
<td>Finds problems</td>
</tr>
</tbody>
</table>

## Agenda

**where we are**

- Background
- What makes formative usability testing different?
- What are the steps?
- Cost/benefit and discussion
- Case study
- Tips for doing good formative testing
- Questions, discussion
Doing a Formative Usability Test

It starts off like any other usability test…

1. Establish the usability and learnability goals of the feature
2. Build a low-fidelity prototype
3. Create your test plan
4. Do a dry run

Running the test

5. Run a test session. Make sure the Interaction Designer is an observer.
6. At the end of the test session, decide whether or not to change the prototype [more on this later].
7. Make changes to the prototype, if required
8. Go back to Step 5. Repeat until you have a sufficient number of sequential, successful test sessions following the last prototype change.
9. Your updated prototype is your verified interaction design. Pass it on to engineering for implementation.
Doing a Formative Usability Test

When do you change the prototype?

The interaction prototype should only be changed under certain circumstances:

• The user encounters a problem that blocks him or her from completing the task.

• The prototype fails its design goals.

• You’ve seen a problem enough times to warrant a change.

Doing a Formative Usability Test

Example Test Run

Say you decide that you need 5 testers to verify the usability goals. Your test may go like this:

1. Test run (succeeds) – 4 more testers required
2. Test run (fails) – changes made – 5 more testers required
3. Test run (fails) – changes made – 5 more testers required
4. Test run (succeeds) – 4 more testers required
5. Test run (succeeds) – 3 more testers required
6. Test run (fails) – changes made – 5 more testers required
7. Test run (succeeds) – 4 more testers required
8. Test run (succeeds) – 3 more testers required
9. Test run (succeeds) – 2 more testers required
10. Test run (succeeds) – 1 more testers required
11. Test run (succeeds) – DONE and VERIFIED
Doing a Formative Usability Test

Example Test Run (continued)

Total number of test runs: 11
Versions of prototype examined: 4

To examine four prototypes with “heavier” formative testing to the same level of verification, you would need to conduct four tests, each with five test runs – about double the number of test runs.

Plus, you’d have four reports to write.

Doing a Formative Usability Test

Writing a report

Your hand-off to engineering should be the updated prototype. If you choose to write a report, do so for historic reasons, e.g.:

- In case someone asks you later “why didn’t you do X?”
- If you re-visit a feature in a later release, and need the context of earlier decisions
- If you change usability staff during a project.

In the report, record each version of the prototype, the changes that you made, and what prompted the change.
Agenda

where we are

Background
What makes formative usability testing different?
What are the steps?
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, discussion

Cost/Benefit

Costs

Formative testing does not give you statistically valid, repeatable metrics – you’re testing a moving target.

It’s hard to plan tests when you don’t know how many testers you’re going to need.

It reduces your visible deliverables.
Cost/Benefit

Benefits

**Faster Iteration**: “Light” Formative usability testing can move you through the “design space” of your solutions quickly.

**Better Insight**: When interaction designers observe the test runs, they gain insight into user behaviour, which they can apply immediately to the design.

**No number crunching**: Unless measuring your usability goals requires it. You just look for the biggest problems and fix those.

You are always working on the most important problems, since they are the ones users find first.

Cost/Benefit

Problems with heavier testing practices

If you use heavier testing practices on your early designs, you often spend more time examining the prototype than it deserves.

*Time spent testing a design that has already failed its usability goals is time not spent trying something different, something that might work.*
Cost/Benefit
end of this section

Questions? Stories?

Agenda
where we are

Background
What makes formative usability testing different?
What are the steps
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, discussion
Case Study

The problem: calibrating a large display

Two steps to calibration:

1. The computer displays a graphic of known size (in pixels)

2. The user measures how big that graphic appears on the screen with a tape-measure, and tells the computer.

The computer now knows how big a pixel is in real-world space, and can display its images accordingly.
Case Study
The problem: calibrating a large display

Some facts about our users and this task:

1. Many of our users are unfamiliar with this task
2. Most find the steps non-obvious
3. This task is done rarely --- perhaps only once
4. Our users are reading-averse (i.e. they don’t read text in dialogs)
5. Some users know a different (more difficult) workflow from another application

Case Study
Goals and original design direction

The usability/learnability goal for this feature was:

Users will be able to correctly use the feature on their first attempt without consulting the on-line help or another person.

Since our users are reading-averse, and this is an unfamiliar task, we decided to show them an image of someone doing the task correctly.
Case Study
First Prototype

This was the first design we tested, with graphics from our graphic designer.

Since we were testing for comprehension, we thought it was important to use final-quality graphics.

The first tester typed in what he guessed was the pixel resolution of his monitor.

Second tester used a ruler to measure his car model on the screen and typed in its size.

Neither tester understood that they had to measure the LINES.
Case Study
Second Prototype

We punched up the lines, and added a word or two to show users exactly on the dialog where to measure.

We removed step one, since neither tester even saw it.

The first tester measured the size of his monitor screen in inches and typed in those values.

He did see the “measure here” message, but thought that it was referring to the edges of the screen.
Case Study
Third Prototype

We really punched up the instruction that tells the user that they should measure between the LINES.

The first tester used a ruler to measure his car model on the screen. He thought that the orange lines showed where on his car he should measure.

The second tester -- a reader -- used the dialog correctly.
Case Study
Fourth Prototype

We removed the car from the graphic, and made the screen more of a solid object.

The major activity happening in the image now is the measuring hands.

And they are clearly measuring between the orange lines.

Case Study
Fourth Prototype

All remaining users in the test run used this interface successfully. This was a sufficient number to validate the design against its goals.
Case Study

Final results

We were able to eliminate 3 failing designs and arrive at a working one, with only 10 test participants (total).

We handed off a successful design to development, which was successfully deployed.

Questions? Stories?
Agenda
where we are

Background
What makes formative usability testing different?
What are the steps?
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, discussion

Useful Tips
Conducting good formative usability tests

- The selection of test participants is no different than in standard testing.
- The interaction designer and test facilitator need to work closely together. Especially, the interaction designer needs to observe all the test runs.
- Fast iteration is no excuse for sloppy design.
- Never put a prototype in front of a user without an intent. Don’t spend time exploring “what if” scenarios.
- Don’t make changes lightly.
Useful Tips
Conducting good formative usability tests

- Do use your judgement – sometimes a problem is a user idiosyncrasy, not a UI issue.

- Don’t make a change unless you can verify that it fixes the problem.

- Try to use the same designers and test facilitators on the same product during its development. Customer insight is cumulative.

- Space out your test sessions so that there is time to change the prototype between testers.

- Consider scheduling continuous, regular test sessions.

Agenda
where we are

Background
What makes formative usability testing different?
What are the steps?
Cost/benefit and discussion
Case study
Tips for doing good formative testing
Questions, stories
Summing up
only one more slide after this one

We have used this kind of formative usability testing successfully to reduce design time while maintaining or increasing the quality of the interaction designs we pass on to our development teams.

For early prototype design, formative testing provides the most important information in a timely manner, and wastes little time on less important details.

Summative testing is still a key component of a complete user-centred design process.

Questions? Stories?
Thank you.

John Schrag
john.schrag@autodesk.com