

An example of breaking a feature into stories

Brian Marick

marick@exampler.com

<http://www.testing.com/cgi-bin/blog>

I'm a reviewer for the Agile2007 conference. Here's an example of a page where reviewers can tell the track chair which papers they want to review, can review, can't review because they don't have the needed expertise, or can't review because they have a personal relationship with an author.

Reviewer: Brian Marick						
#	Title	Attachment	I would like to	I can	I cannot	Conflict of Interest
RESEARCH PAPERS						
347	'Talking the talk': Is intermediate-level conversation the key to the pair programming success story?	Open	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
349	Using Agile Practices in a Maintenance Environment	Open	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
368	Artifacts in System Archeology	Open	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
372	The Role of Agile Practices & Principles in Design Decision Making	Open	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
404	Automated Recognition of Test-Driven Development with Zorro	Open	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
436	A Case Study Validation of an Extreme Programming Bayesian Network Model	Open	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

How could the “epic” of building this page be broken into smaller stories, each with business value? To start, I'm going to assume that there is already a database that holds the submissions. Here are some stories, together with sketches. Imagine these sketches being drawn on a whiteboard as the product owner explains the story to the team. (Some of the time, additions to existing screens would instead be explained by gesturing at the current version and saying what needed to change.)

- *Reviewer can see a list of authors.*

If this story were the only one implemented, the reviewer would have to email the track chair any comments about conflicts. The track chair would have to assign papers to reviewers without knowing anything about their preferences. That's all pretty awkward, but there's still some business value to this story.¹

- *Reviewer can see a list of authors with checkboxes. Checking the box means there's a conflict of interest. The list of checked authors gets sent to the track chair via email.*

This new story makes the previous one rather more useful.

Note that we'll eventually end up with radio buttons, not checkboxes. This story *could* have asked for two radio buttons: one for a conflict, one for no conflict. But that would be a worse user interface. The only reason to choose it would be if undoing the checkboxes and putting in radio buttons were expensive. But it's the programmers' job to make changes cheap. When they're not, the expensive changes give them motivation and opportunity to improve the code, their skills, or their tools.

¹ Note that the form on the first page doesn't show authors. In the real project, that's a story that never got implemented. I happened to focus on author conflicts in my early stories; the real product owner didn't. Product owners have a lot of flexibility in how they grow their product.

- *Reviewer can see a list of titles and authors, together with a checkbox for conflict of interest.*

This story will mean reworking the page. What was a simple list of authors will now be a list of titles with authors indicated alongside them. Further, the conflict will now be associated with a paper, not a particular author.

Authors	Conflict?
"My paper", Joe Bergin	<input checked="" type="checkbox"/>
"Another paper", Joe Bergin, Donald Holmes	<input checked="" type="checkbox"/>
"Holmes Alone", Donald Holmes	<input type="checkbox"/>

Submit

- *Reviewer can, in addition to conflicts, indicate "I want to review", "I can review", or "I can't review".*

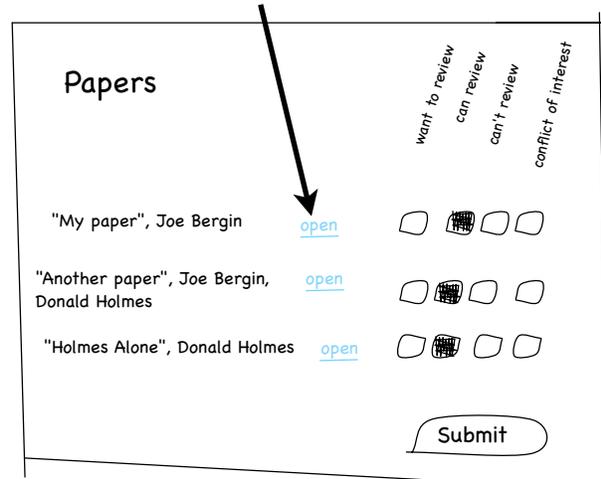
Here come the radio buttons. Note that I haven't said anything more about how the information is delivered to the track chair. Until I do, it'll continue to be by email. In real life, I'd be developing the track chair control panel too, likely in parallel, but I'll stick to bids for this example.

Papers	want to review	can review	can't review	conflict of interest
"My paper", Joe Bergin	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
"Another paper", Joe Bergin, Donald Holmes	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
"Holmes Alone", Donald Holmes	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Submit

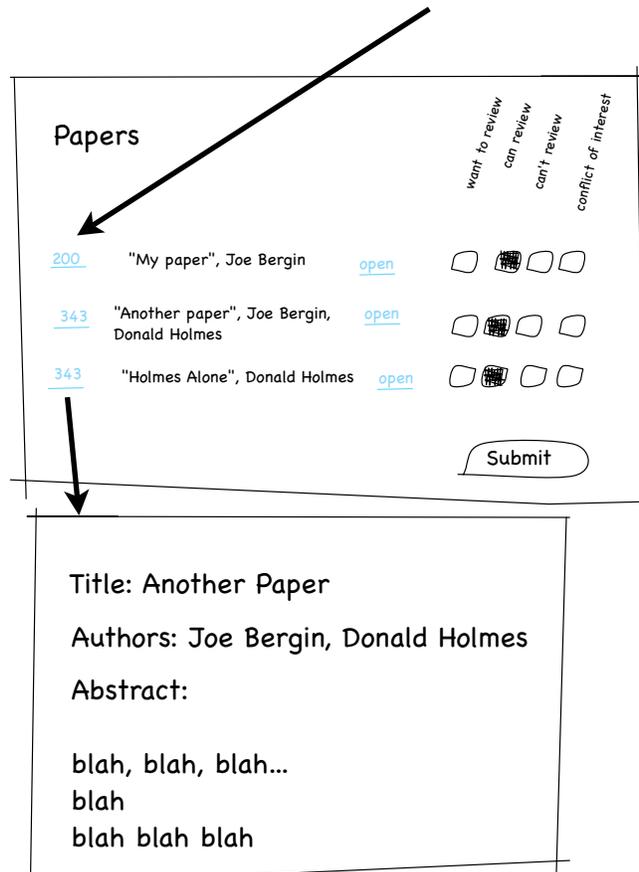
- Reviewer can click on a link to open the actual paper. (A PDF would probably open in her browser; a Word document would probably be downloaded. It all depends on what plugins she has.)

Because downloading and opening a file is slow, this new feature is inconvenient when all you want to do is see the abstract. But it's a cheap way to add value.



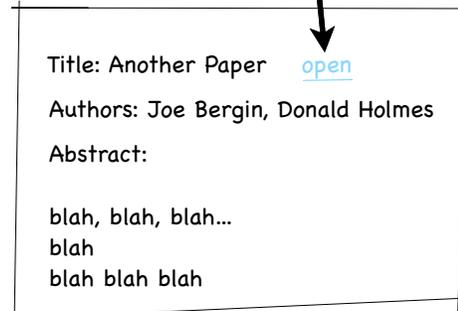
- Reviewer can click on a link (labeled by paper number) to pop up a new window with at least the author, title, and abstract.

If there's other information in the database, and it's easy to add to the popup page, the programmers might ask if I want it. If I do, I'll have them add it.



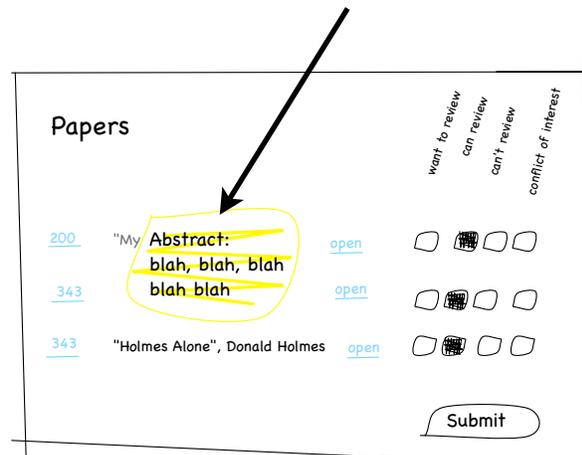
- *The paper abstract page (just described) should have a link to open the actual paper.*

Oops. When real people made use of the previous story, they would often read the abstract, want to scan the whole paper, and be annoyed that they had to find the list-of-papers window to do it. One of the benefits of Agile is that you rarely have to get things right the first time.



The position of the link on the page is probably wrong. As a product owner, I'd expect programmers to show me the page when they thought they were done with the story. It would be OK for me to then ask them to move the link elsewhere. (Moving it should be a matter of minutes, so it's not a new story.) Or they might think of a better place as they coded and immediately ask me if I wanted to move it.²

- *Make the abstract pop up in a tooltip when you hover over the paper title. I'm adding this toward the end because its business value is lower. I consider the previous set of stories the minimum tolerable bid system. This story is just a convenience.*



(Note: the yellow of the tooltip won't show up on a black-and-white printer.)

² Note that all this is *much* easier when the product owner is in the bullpen with the team.

- When you click on the paper title, the abstract will scroll out below it in an Ajaxy way.

Another convenience feature: a lot of people don't think to hover.

As an aside, this story will introduce a new technology (Javascript). That affects the estimate.

Even though this story and the previous one do the same thing (display the abstract in the list-of-papers window), this story is likely to cost more. Suppose it costs a *lot* more. I might talk over my plans with the programmers. If this seemed to be the only story that would require Javascript, I might well drop it or defer it to see if other Javascript stories pop up. Alternately, the programmers might find ways to slice bits off the story to make smaller stories that still have immediate business value.

What if the story can't be slimmed down? And what if the estimate is longer than the ideal three days? Worse yet, what if it's longer than the sprint? In that case, I'd reluctantly have the programmers create so-called *technical stories*. Those are stories that still have the virtue of being small enough for good estimates, but they don't deliver business value until they're all done. That makes them riskier: more can go wrong, for a longer time, before the business can see or control it.

