

November 2008

\$9.95 www.StickyMinds.com

BETTER SOFTWARE

SIMPLIFY, SIMPLIFY
Status reports
executives can use

**UNCHARTED
TERRITORY**
The route to great
test coverage

The Print Companion to StickyMinds.com

GETTING
AGILE with
USER-CENTERED
DESIGN
**CREATE SOFTWARE
USERS CAN'T LIVE WITHOUT**



Cover or Discover?

by Michael Bolton

Most of the time, excellent testing and our evaluation of the quantity and quality of our test coverage start with asking who our client is and what he wants to know about the system. Yet, if our client isn't knowledgeable about testing, development, or some other aspect of the context in which the product must work, it's possible that neither we nor the client knows what he wants to know. So where could we start? We might want to start by looking at a map of the system and asking questions about it. Suppose that I showed you a map of a subway system and asked you to cover the map. How would you do it?

You might start by choosing a station at the end of one of the lines on the system and then riding the train to the other end of the same line. Then you might do this for each of the other lines in turn. When you had covered all of the lines, would you have achieved complete coverage of the map?

Maybe not. Most subway lines consist of two or more tracks in each direction. To cover the system, you'd have to take each line in both directions. So you do that. Would you be done then? Maybe not. There are also sidings off the main lines, so that trains can turn around or pass one another. After you have visited all the sidings, would you have achieved complete coverage? Subway lines don't exist on their own; there are transition points between the lines. To cover the map, you'd likely want to cover the interchanges between the lines in every direction. And is it sufficient merely to travel the lines, or would we need to stop at each station and have a look around?

So far we've been thinking about covering the map in an abstract, structural way, but we haven't been thinking very much about what actually happens on or with the system. We haven't talked much about the rolling stock—the trains themselves or other vehicles like maintenance wagons. We have yet to discuss the infrastructure—track, signaling systems, elec-

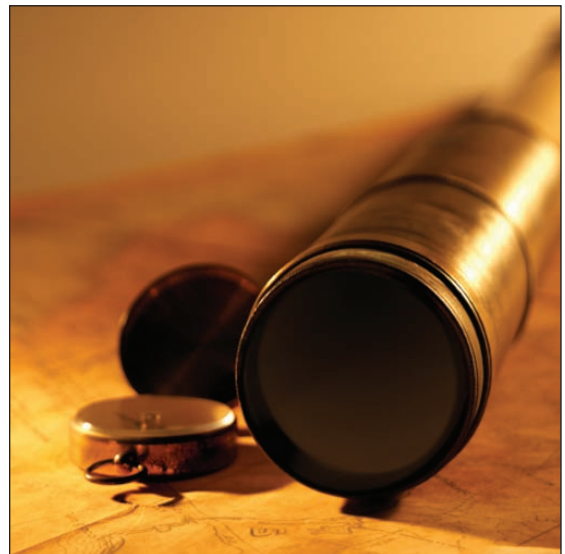
trical supply—that is needed to support the operations of the system. We could look at the interconnections with other systems—sidewalks, buses, streetcars, and commuter train services. We might talk about the different types of people who use the system—commuters, students, and people with strollers or in wheelchairs. We might consider why they are using the system—to get to work or school during peak hours, to get to cinemas and art galleries on the weekends, or to get home after the bars close. We could discuss aspects of the system that might seem peripheral or incidental to its primary purposes—signage, collection booths, janitorial services, newsstands, and advertising. We should consider the relationship between the system and time—scheduling, actual time between trains, boarding time. We haven't considered how our observations might be affected by the times at which we make them—at rush hour, in the middle of the day, in the middle of the night when the system is closed. And then, we could think about combinations of circumstances and how they might result in behavior that is non-linear—at rush hour, trains are more crowded, so they take longer to board, so the trains can't leave the stations as quickly as at other times, so the system backs up, so the trains become more crowded.

Maybe we haven't talked about these things because we—or our clients—haven't completely realized the motivation for why we're testing. Yet maps rapidly inspire thinking about what we might think or do to cover them and raise possibilities about what we might see.

Some representations of a system actually look like maps—flowcharts, diagrams, UML models, and the like. Usually we design tests by asking ques-

tions about specific things that are labeled on the map such as the steps in some task that the map models. But if we're stuck for ideas on what might be most important to cover, we could start with some map and ask questions that apply generally to maps or apply some general guideword heuristics—*simplest, popular, critical, complex, pathological, challenging, error handling, periodic*. “What's the easiest way to get from here to there? What's the *standard* way? What routes are essential? What parts of the route are potentially confusing or error prone? What could interfere with this task? What would happen if this route were missing or compromised? Does the map suggest what workarounds might be available? How much traffic goes along this path? How fast does it go? Are there possibilities for collisions?”

But another class of questions might arise, too, based on the idea that we might add details to an existing map. An older version of the Toronto subway map that I have indicates the lines and their connections, the stations, and interconnections with suburban rail. A newer version adds extra labels—the street addresses of the subway stations and the locations of parking lots, public restrooms, elevators for people in wheelchairs or with strollers, and surface route transfer points—but drops the labels for the suburban rail transfer points. This reminds us that any map includes some kinds of information that might be useful



ISTOCKPHOTO

and leaves out other information. Comparing and contrasting two maps may suggest ideas that neither map covers.

Some suggest that a map must accurately reflect the territory in order to be useful, but there are plenty of reasons to think otherwise. Subway maps often display the relationships between stations but not the actual distances between them. A story in *Sensemaking in Organizations*, by Karl Weick, suggests a more profound example of usefulness despite inaccuracy. A Hungarian Army unit is on patrol in the Swiss Alps. A big storm comes up, and one platoon doesn't return to camp for a day, two days, three days. The lieutenant is now panicked ... and suddenly the entire platoon walks back into camp. "We thought you were lost for good! Where have you been?" "Well, when the storm came up, we took shelter. When the weather cleared, we realized that we were lost. One of us had a map, though, so we looked at it, and we realized that if we went down the hill, we'd hit a river, and if we followed the river, we'd get to the town ... and here we are!" The lieutenant looked at

the map and realized to his surprise that it wasn't a map of the Alps, but of the Pyrenees. Says Weick:

This raises the intriguing possibility that when you're lost, any old map will do ... maybe when you are confused any old strategic plan will do. Strategic plans are a lot like maps. They animate and orient people. Once people begin to act, they generate tangible outcomes in some context, and this helps them discover what is occurring, what needs to be explained, and what should be done next. Managers keep forgetting that it is what they do, not what they plan, that explains their success. They keep giving credit to the wrong thing—namely, the plan—and having made this error, they then spend more time planning and less time acting. They are astonished when more planning improves nothing [1].

It's easy to see how covering a map might be useful. Ultimately, though, it's

what people think and what people do that make the difference. Analyzing a map and planning how to cover it—test design—can suggest ideas that the map on its own doesn't cover. That might spark us to annotate the map we've got or to create a new one. Comparing the map to the territory—test execution—takes us places where we learn things, even when the map is limited or inaccurate. Excellent testing isn't just about covering the map—it's also about exploring the territory, which is the process by which we discover things that the map doesn't cover. {end}

REFERENCES:

[1] Weick, Karl E. *Sensemaking in Organizations*. Sage Publications Inc., 1995. p 54-55.



Of course there are ways to learn about the product other than maps or diagrams. What representations do you use and how do you cover them efficiently?

Follow the link on the StickyMinds.com homepage to join the conversation.

S U C C E E D I N G W I T H A G I L E SM

DON'T BE SHY about transitioning to agile.

Face and overcome the challenges. Master agile software development techniques and deliver valuable, functional software in a world of uncertainty and change.

Join **Mike Cohn**, author of *User Stories Applied* and *Agile Estimating and Planning*, in courses exploring the principles of agile software development: people over process, working software over documentation, collaboration over negotiation, and flexibility over rigidity.

To learn more, visit
www.mountaingoatsoftware.com/better

Upcoming Courses

Dallas

January 27-28
Certified ScrumMaster

January 29
Agile Estimating and Planning

Boulder

February 18-19
Certified Scrum Product Owner*
*with Ken Schwaber

Seattle

March 31-April 1
Certified ScrumMaster
April 2
Agile Estimating and Planning

Orlando

May 12
Effective User Stories
for Agile Requirements

May 13-14
Certified ScrumMaster

May 15
Agile Estimating and Planning



**MOUNTAIN GOAT
SOFTWARE**