

13 April 2006

Michael Bolton

mb@developsense.com

<http://www.developsense.com>

Back issues: <http://www.michaelbolton.net/newsletter/index.html>

Blog: <http://www.developsense.com/blog.html>

Welcome!

Please note: I've sent this newsletter to you either because you asked me for it explicitly, or because I genuinely thought that you would be interested in it. If neither is the case, please accept my apologies and let me know by clicking [here](#), or send a message to remove@developsense.com.

On the other hand, if you like this newsletter, *please take a moment to forward it to friends or colleagues that you think might be interested*. If you'd like to get on the list, please click [here](#), or send a message to addme@developsense.com.

Your email address is just between you and me. I won't give your email address to anyone else, nor will I use it for any purpose other than to send you the newsletter and to correspond directly with you.

Your comments and feedback are very important to me, and I'd love to share them with the rest of the recipients of the letter. Please send them on to me at feedback@developsense.com.

In This Newsletter...

- What I've Been Up To
- One-Day Workshops and Presentations
- Teaching Rapid Testing in India
- Book of the Month: The Moleskine Notebook
- Notes From An Exploratory Testing Session
- Coming Up

What I've Been Up To

Since the last newsletter, one highlight was bringing James Bach to the Toronto area for three special events; more on that below.

I wound up my assignment at a Canadian financial services firm at the end of 2005, and took most of January off. January ended with the Exploratory Testing Research Summit, and February began with the 2006 Workshop on Teaching Software Testing. The middle of March was taken up with one-day Rapid Testing tutorials in Delhi, Mumbai, and Bangalore, and a couple of days of a conference, all sponsored by QAI India. Again, more on that below.

One-Day Workshops and Presentations

For a long time, I wanted to bring James Bach to Toronto to address the Toronto Association of System and Software Quality, and this became possible thanks to his schedule and his very generous terms. In the early summer, Mike Cookson of the South Western Ontario Software Quality Group contacted me; his group wanted to host a “Build The Skill” day for its members, and was interested in having either James or me present a workshop on Rapid Testing. We realized that we could co-present a one-day Rapid Testing sampler on Monday, November 28, 2005, and that this would dovetail perfectly with the TASSQ presentation the following evening. TASSQ jumped at the opportunity to present a workshop of its own, so the final itinerary was Rapid Testing in Kitchener, Ontario on Monday; another day of Rapid Testing in Toronto on Tuesday, and James’ presentation “Skilled Testers and Their Enemies” Tuesday evening.

The results of all this activity were wonderful. The organizations made a good deal of money on the workshops. We provided over 60 people with a one-day sampler of the Rapid Testing philosophy and course. TASSQ attracted 150 or so people to James’ evening presentation—enough to fill its war chest for some time. And after that, a big group of people went to a local pub for an informal follow-up chat.

For me, the best thing about all three events was that I could feel a real sense of community building. James was inspiring on that front; he urged us all to develop skills, build and foster a network of skilled, engaged, respected professional testers. Lots of people talked to James about their experience, their problems, and their victories, but even better, lots of people met new friends and started talking to each other.

We look forward for the opportunity to do more of the same; we extend the idea of group-sponsored workshops to testers in other communities. If you’d like us to bring the road show to you, let us know.

Teaching Rapid Testing in India

I went to India in the middle of March to teach three Rapid Software Testing One-Day Sampler courses for QAI India in Delhi, Mumbai, and Bangalore. The turnout was very heavy in Bangalore, and I was fortunate to have help from a colleague, Shrini Kulkarni. James and I met Shrini in Toronto; he attended the Rapid Software Testing workshop in Toronto, because he happened to be leading an automation project there at the time. After the workshops, it was back to Delhi to present two keynotes at STC2006, QAI’s annual software testing conference. The reaction to my two conference presentations (Two Futures of Software Testing, and Testing Skills That (Almost) Nobody Talks About) was very gratifying.

Mahantesh Ashok Pattan gave a really exciting presentation just before my afternoon talk. He described his work on a test automation project at HP, where he rationalized several different approaches, sets of tools, classes of tests, and volumes of data. In his presentation, he did two things that were singular. First, he claimed that he had achieved 100% automation coverage--which is the sort of statement that I usually dismiss out of hand. Then, to my astonishment, *he actually said what he meant by that*, and what's more, *it actually made sense*. He defined 100% coverage as "successfully automating all of the tests that we set out to automate", while recognizing that this was neither all the tests that they *could have* automated, nor all of the tests that could be performed (since, as he carefully and surprisingly pointed out, automation simply isn't feasible for all tests; only for those that it makes sense to automate). Note that his definition

of coverage, in this sense, fits with Rapid Testing's open-ended definition of coverage, "the extent to which we've tested the product according to some model".

Under extreme time pressure (the conference schedule was running late), Mahantesh demonstrated an architecture and an approach to performing and thinking about automation that was first-rate. He had great energy, was obviously passionate about his trade, and he had real content *and* real context for his remarks.

Vipul Kocher invited me to speak to his testing services company, PureTesting. Vipul also runs the WhatIsTesting.com Web site, which is soon to be published in magazine format. Vipul is a strong believer in developing expert testers, and in creating a worldwide community—and he's done something about it: check out <http://www.whatistesting.com>.

It was my third visit to the India. Each time, I've been impressed by an enormous ambition and optimism in the testers that I've met. People like Mahantesh, Shrini, and Vipul make me believe that India will develop a strong community of first-rate testers that will match or exceed North America's. There is a risk of selection bias here; I'm most likely to meet people who are most interested in meeting me, and people who don't like testing or aren't engaged by it won't tend to approach me. However, the people who have approached me have all impressed me a lot. In response to the talks, I found that a lot of people were asking about how to improve skills in themselves, in their organizations, and in the country. So...

One Way To Create A Testing Community

...one answer for everyone, whether in India or anywhere else: *let's talk to each other*. If you're a tester, you almost certainly have a dozen or so colleagues within your company—not only testers, but developers, business analysts, support people, and so on—and you may have friends outside your company who are also in a line of work that's related to testing. Ask each of those people to refer you to two or three other people, involved in testing in some way, that they believe that you've never met. Find a space that seats at least a dozen people—a meeting room in your company, a local library or community centre, or a restaurant or pub. Choose some date and time for an evening or lunchtime meeting, and send out the invitations. Meet once a month or so. Ask people for presentations of an hour or so. Hold question-and-answer sessions afterwards. If you don't get volunteers for long talks, try panel discussions or "lightning talks"—a set of five-minute presentations followed by discussion and conversation.

Book of the Month: The Moleskine Notebook

In 2005, I met Jon Bach for the first time. Jon, brother of James, is an expert exploratory tester (I guess that runs in the family) and a wonderful writer on the subject of E.T., and in particular how to make it accountable. The first thing that I noticed on meeting Jon is that he's an assiduous note-taker—he studied journalism at university—and over the last year, he has inspired me to improve my note-taking processes.

Another factor in that improvement was James' recommendation of the Moleskine pocket notebook. I got my first one at the beginning of the year, and I love it. It fits in a pocket; it's bound to lie flat; it has a fabric bookmark and an elasticized loop that holds the book closed. They come in larger sizes, if that's more to your liking, and the pages can be unlined, lined, or

squared (graph paper). I prefer the latter; I find that it helps with sketching and with laying out tables of information. You'll see some examples below.

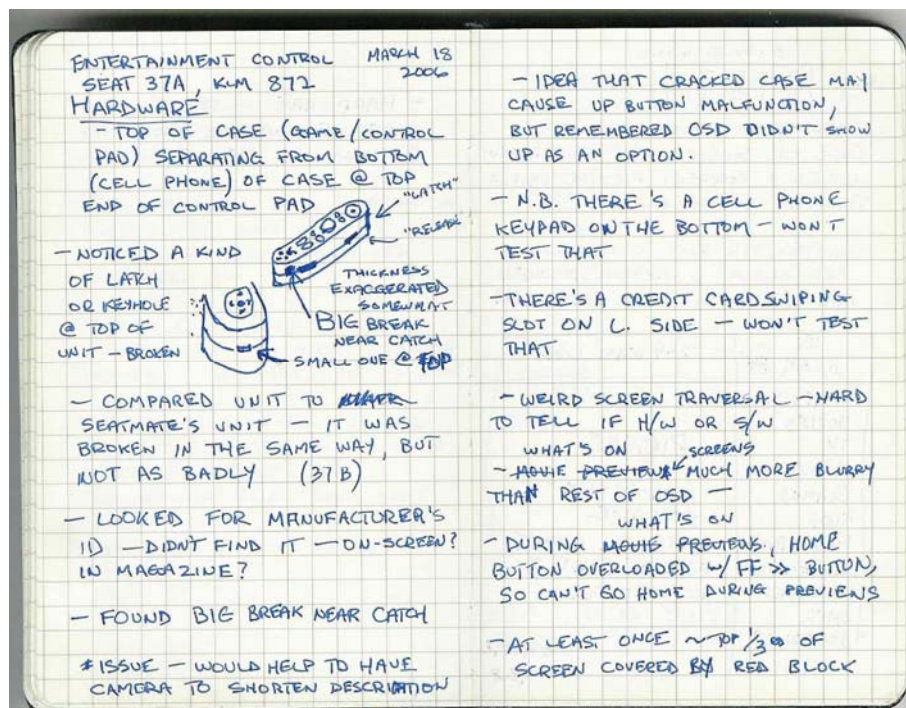
Moleskines are available at better bookstores all over the place, and from <http://www.moleskineus.com>.

Notes from an Exploratory Testing Session

I flew from Delhi to Amsterdam. I was delighted to see that the plane was equipped with a personal in-flight entertainment system, which meant that I could choose my own movies or TV to watch. As it happened, I got other entertainment from the system that I wouldn't have predicted.

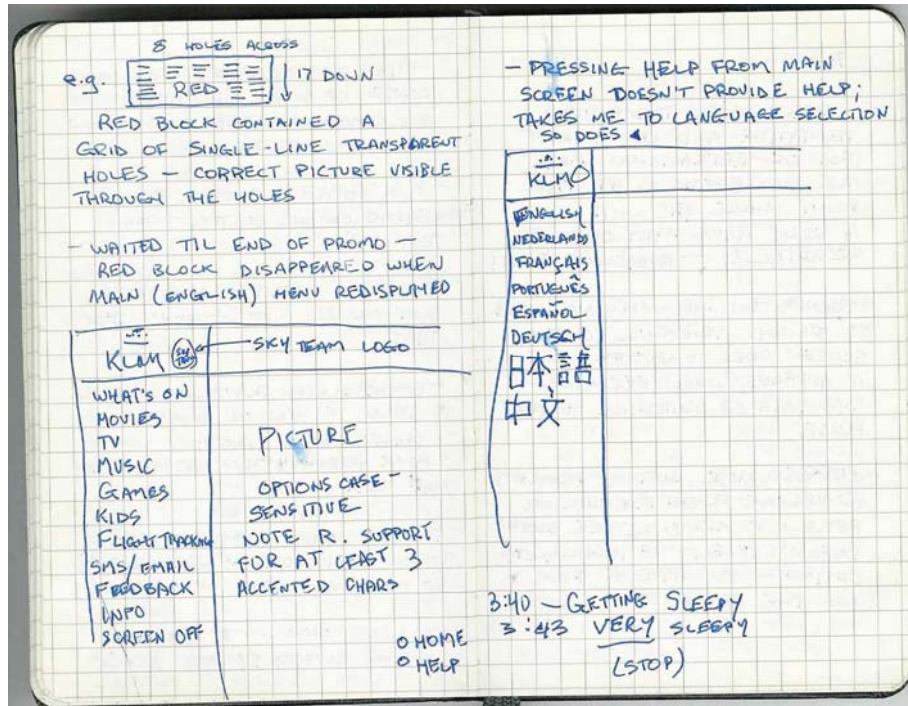
The system was menu-driven. I went to the page that listed the movies that were available, and after scrolling around a bit, I found that the "Up" button on the controller didn't work. I then inspected the controller unit, and found that it was cracked in a couple of places. Both of the cracks were associated with the mechanism that returned the unit, via a retractable cord, to a receptacle in the side of the seat. I found that if I held the controller just so, then I could get around the hardware—but the software failed me. I found lots of bugs.

I realized that this was an opportunity to collect, exercise, and demonstrate the sorts of note-taking that I might perform when I'm testing a product for the first time. Here are the entries from my Moleskine, and some notes about my notes.



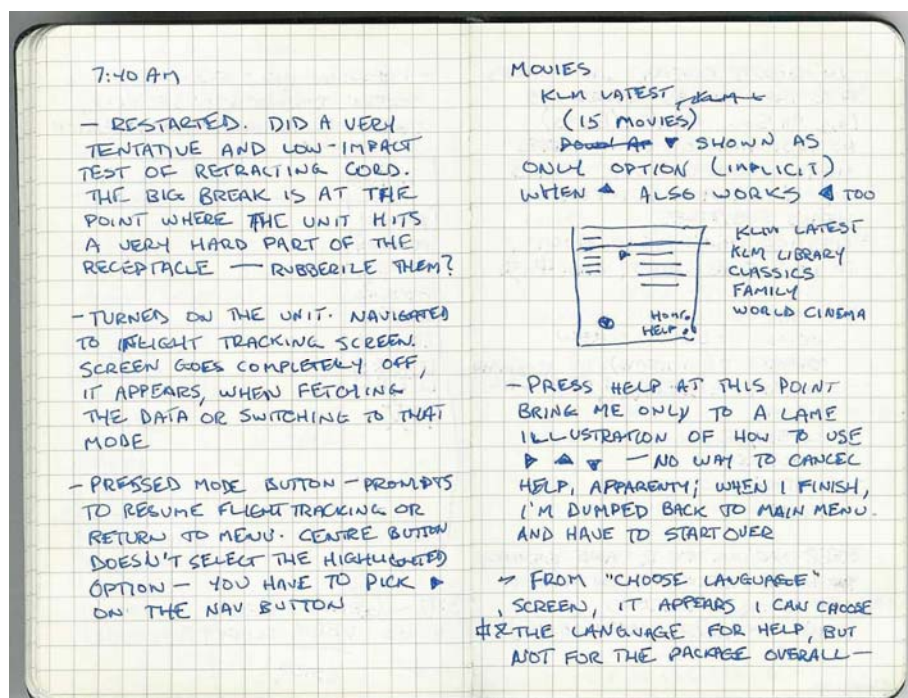
When I take notes like this, they're a tool, not a product. I don't expect to show them to anyone else; it's a possibility, but the principal purposes are to allow me to remember what I did and what I found, and to guide a discussion about it with someone who's interested.

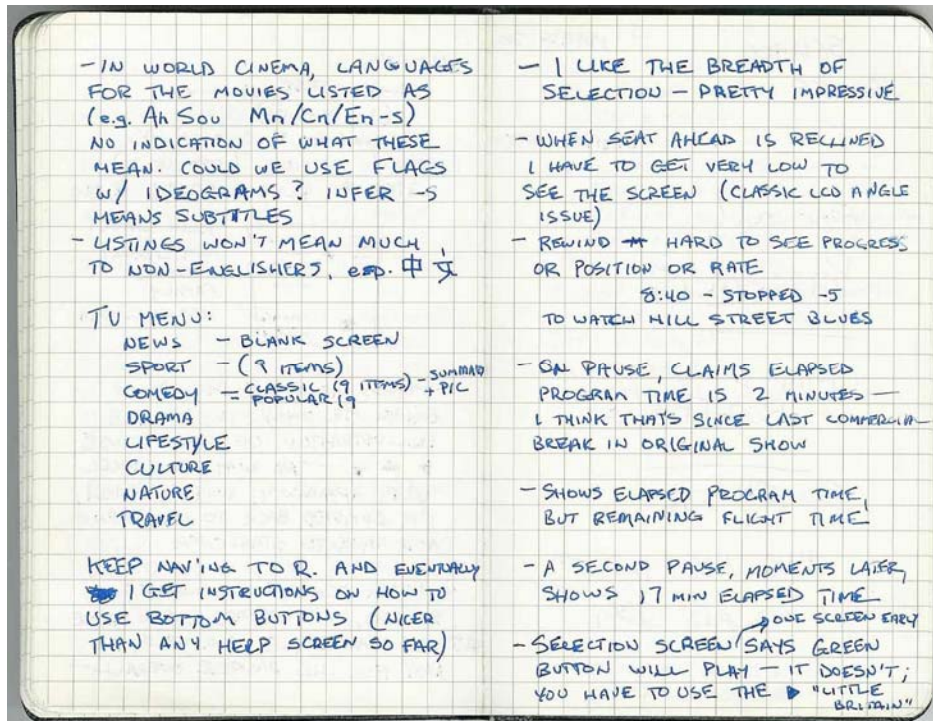
I don't draw well, but I'm slowly getting better at sketching with some practice. I find that I can sketch better when I'm willing to tolerate mistakes.



In the description of the red block, at the top of the left page, I failed to mention that this red block appeared when I went right to the “What’s On” section after starting the system. It didn’t reproduce.

Whenever I look back on my notes, I recognize things that I missed. If they’re important, I write them down as soon as I realize it. If they’re not important, I don’t bother. I don’t feel bad about it either way; I try always to get better at it, but testers aren’t omniscient. Note “getting sleepy”—if I keep notes on my own state, they might suggest areas that I should revisit later.





Jon Bach recently pointed out to me that, in early exploration, it's often better to start not by looking for bugs, but rather by trying to build a model of the item under test. That suggests looking for the positives in the product, and following the happy path. I find that it's easy for me to into the trap of finding and reporting bugs. These notes reflect that I did fall into the trap, but I also tried to check in and return to modeling from time to time. At the end of this very informal and completely freestyle session, I had gone a long way towards developing my model and identifying various testing issues. In addition, I had found many irritating bugs.

Why perform and record testing like this? The session and these notes, combined with a discussion with the project owner, might be used as the first iteration in the process of determining an overall (and perhaps more formal) strategy for testing this product.

Coming Up...

- I'll be presenting a half-day tutorial on exploratory testing at [QAI's 26th Annual Software QA Conference](#) in Orlando, Florida, April 25. The conference runs from April 24-28, 2006.
- I'll be giving a one-hour presentation on Rapid Software Testing at the QAI Conference on April 27, 2006.
- I'm teaching Rapid Software Testing to a corporate client in Altoona, PA, May 3-5, 2006.
- I'm teaching Rapid Software Testing to a corporate client in Toronto, ON, May 9-11, 2006.
- I'll be presenting at [Software Testing Analysis and Review](#) (STAR) East in Orlando, Florida, May 15-19, 2006.
- I'm attending the Workshop on Heuristic and Exploratory Testing, May 19-20, Palm Bay, Florida.

I still owe you a report on the Exploratory Testing Research Summit, too. I hope my writing can keep up with my schedule.

---Michael B.