10 Things You Might Not Know

About Sources of Testing Ideas

by Robert Sabourin

I am always on the lookout for new testing ideas. Some of the most valuable test ideas come to mind when I am actually testing. Here is a list of ten sources of test ideas I use on Just-in-Time (JIT) testing projects.

1 Capabilities. Capability-based test ideas focus on confirming that an application does what it is supposed to do. Requirement and functional specifications can be used as a source of capability-based testing ideas.

2 Failure Modes. Failure mode-based test ideas are "what if" questions that often are inspired by how a system is designed. I look at all of the objects, components, and interfaces in a system and ask myself, what if they break or exhibit some sort of unanticipated failure. Failure modes can be the result of harshly constrained system resources or forced error conditions.

3 Quality Factors. Quality factors are characteristics of a system that must be present for the project to be successful. Quality factors are the "ilities" including: usability, reliability, availability, scalability, and maintainability. Quality factor test ideas often involve experiments to determine if a quality factor is present. Examples include performance, load, and stress testing.

4 Usage Scenarios. A usage scenario test idea challenges whether a user can achieve his tasks with the software under test. To paraphrase the Kennedy inaugural address—we ask not what the software does for the user, but rather we ask what the user does with the software. Usage scenario-based test ideas involve identifying who is using the system, what he is trying to achieve, and in what context.

5 Creative Ideas. Creative test ideas come from many sources. I often use deliberate lateral thinking techniques (for example, Edward De Bono's Six Thinking Hats) to come up with cool and effective tests. I also use metaphors to come up with testing ideas. I wonder what would happen if the Tasmanian Devil used the system. Perhaps a Dr. Seuss story inspires some testing ideas. Perhaps great detectives or movies. Pretty much anything goes here.

6 States. When testing an application, a state model helps me come up with test ideas. For example, a transaction goes through many states of existence from creation through approval, payment, and delivery. I use state models to identify test ideas such as getting to states, exercising state transitions, and navigating paths through the system.

7 Data. Data is a rich source of testing ideas. Data flow paths can be exercised, different data sets can be used, data can be cooked up and built from combinations of different data types, and stored procedures can be verified. Test ideas can be developed to create, update, and remove any persistent data.

8 Environments. Exploring how the application behaves in different operating environments is a rich source of testing ideas. Environment test ideas can relate to varying the platform, hardware, software, operating system, co-resident third-party software, and locales.

9 White Box. White box test ideas come from reviewing low-level design, code, and data schemas. White box test ideas include exercising the code, decisions, and paths through the program. White box test ideas look at the work products of developers to find ways to challenge the application under test.

10 Taxonomies. One of my favorite sources of test ideas is bug taxonomies. These are organized, documented collections of bugs. I peruse taxonomies randomly in search of interesting bugs. If the bug could happen in my application, I come up with a test idea to expose it. Three taxonomies I often use are:

- Appendix A of *Testing Computer Software* (Kaner, Falk, and Nguyen)
- Boris Beizer Taxonomy (Otto Vinter)
- Shopping Cart Taxonomy (Giri Vijayaraghavan)