

# The Future of Software Testing Is Ours To Make!

© 2008 Fiona Charles

*Originally published in the EuroSTAR Newsletter August 2008*

“The era of the independent tester is over”

“If you can't code, there's no room for you on an Agile team. Agile is the future and Agile teams want only generalists.”

Do you buy these? I don't!

When I look at many of my colleagues, I see career testers who are uniquely placed to find the important bugs that developers miss—even after doing test-driven development and running automated unit and acceptance tests. I see pioneers developing exciting and important new ways of testing and thinking about testing.

What I don't see is much recognition of these things in the mainstream world of software development. Disturbingly, I often don't see it among mainstream testers, many of whom seem not even to have heard that there is more than one way to test software.

There are many forces at work in our industry that could make us wonder if we are indeed obsolete. Managements are continually pressured to bring costs down. Development thought leaders behave as if only developers should own and control the entire software development cycle. Most IT managers and project managers were developers first. They understand development better than testing and tend to identify with the developers' point of view, discounting the testers'.

Add to this mix the perception—and, too often, the fact—that testing is unduly slow, cumbersome and expensive for the results it achieves, and it's not surprising when managements limit testing, or try to eliminate specialized testers from their projects.

The future doesn't have to be like this.

In reality, software projects cannot well do without the value testers bring, with our particular skills and special mindset. And we can turn that same creativity, independence of mind, empathy, courage and tradecraft to shaping the future of software testing.

## **Creativity**

To assure that future, we have substantive, creative work to do.

Just as development methods are continually changing, we need to keep reinventing and improving what we do:

- Develop new ways to test better and faster
- Select the best and most appropriate techniques, tools and practices for each job from a well-stocked—and constantly growing—bag of tricks

- Decide how we can best contribute our skills, as project strategies and development methods inevitably grow and change

And,

- Continue to find new ways testers can add value to projects and to software quality

Good testing requires creativity. The best testers have it (and can always enhance it), so let's use our creativity to make the future we want to work in.

## **Independence**

Creativity relies on independence of mind. The tester who is an independent thinker is a critical thinker by definition. That tester designs an independent test model to evaluate the validity of the model implemented by development, rather than repeating in the tests the assumptions and constraints the developers adopted.

Testing means also testing the veracity of our information and our sources. And it means continually testing our own assumptions, processes, practices and techniques to verify their fitness for each current purpose.

Smart testers will always be truly driven by context:

- Not wedded to any particular camp, but aware of and open to the ideas of all
- Able to adapt and take the best from any community of testing, as appropriate for each project
- Inventive in developing their own techniques and practices, or in applying the ones they have learned.

## **Integrity**

Implied in independence of mind is the integrity to speak truth to power. Our job is to seek out and provide information about software quality. Though others may sometimes obfuscate or attempt to hide problems, testers are paid to give our best professional judgment of quality truthfully and completely.

## **Empathy**

Just as we need to bridge the gaps between test communities, so must we work as partners with developers. Independence of mind does not have to mean the structural antagonism between testers and developers that we have experienced in so many organizations. We think differently, and we have different roles, points of view, and skills. We share the goal of producing quality software.

We must especially extend our empathy to project stakeholders, so that we seek out and hear what will be the most useful information for them to receive. We need to know what matters to stakeholders, so we can work for:

- Improved quality where it matters most to them
- Reduced risk where it matters most

## **Courage**

Creativity, independence, integrity, empathy—exercising all of these requires courage, and no tester can be successful without it. It takes courage to work side-by-side with your developer colleagues and yet retain independent professional judgment. It takes courage to call out risks that management would prefer to ignore.

## **Tradecraft**

Finally, there are the skills a tester brings to a project. Whether these are technical, test technical or the so-called “soft” people skills of teamwork and communication, the good tester’s particular combination of skills is both unique among software development practitioners, and rightfully needed.

## **We can make the future, or we can let it make us**

Of course, there will always be “outside” influences we can neither predict nor control. Economic forces, business trends, and new software development and project management practices will all have an impact on what software testers do and how we do it. But the principal challenge of shaping and managing the profession of software testing—its standards, skills, techniques and practices—belongs to us. We the practitioners own that responsibility, and have the right to carry it out.

The future of software testing is ours to make! We have the skills, the character, and the qualities of mind to do the job. Let’s work together and decide what we want it to be.