# Clearing a Career Path for Software Testers

*As crucial as software testing is to product success, it gets little respect in most organizations. AT&T has taken its tradition of strong support for software testers a step further by establishing a formal career plan for testers that includes an ongoing education program.*

**Elaine J. Weyuker, Thomas J. Ostrand, JoAnne Brophy, and Rathna Prasad,** *AT&T*

**E**ffective testing is crucial to producing reliable and dependable software systems. However, few guidelines or formal criteria exist that determine whether someone is or might become an effective software tester. Also, testing is often viewed as an undesirable and unchallenging career track. Some organizations' staff testing groups primarily with new hires and mediocre or failed programmers. The implication is clear: Testing is not that important to product success.

However, this is clearly not the case. Many large-scale systems consist of millions of lines of source code. When a testing group determines that such a system is ready for release, it is essentially certifying that the software will behave with an acceptable level of dependability. The steps leading to this release decision must be based on sound engineering practices and carried out by qualified people.

AT&T has always considered software testing an important and central activity. Software testers are generally treated with respect and viewed as essential to product success. To reinforce this view and to make sure that potential software testers are offered a varied and challenging career, AT&T has recently begun a Software Test Engineer Professional Development Program. The program's primary goals are to attract highly qualified people to a career in software testing and to foster the growth of a permanent staff of highly competent test engineers. Achieving these goals should in turn lead to more uniform, more efficient, and more effective software testing practices, resulting in shortened product development and maintenance cycles and more reliable products. The program has three parts:

- identify and define the competencies needed for effective software testing,
- define a career path for software testers and support them in pursuing it with education and training, and
- encourage software testers to get formal certification from recognized organizations.

We are part of a corporation-wide team charged with designing this program. Cur-

rently, our work focuses on defining the role of the software test engineer, identifying associated skills and competencies, defining an interesting and varied career path, establishing appropriate training, and determining relevant certification opportunities to recognize outstanding software test engineers. In this article, we describe that career path and the core competencies that we've identified as key to successful testing.

## Core Competencies

We have identified two core competency categories for prospective software testers. The first consists of general engineering skills and knowledge. Employees can acquire some of these competencies during study for an undergraduate or Master's degree and others during the early years with the company. Such skills include general software engineering knowledge; topics in telecommunications and networking, which are AT&T's key technologies; and technical analysis and presentation skills. The second category consists of core testing-specific skills and knowledge that potential test engineers need and can acquire during the first two years of employment.

### General Skills

At AT&T, a new hire with a computer science degree is expected to have knowledge of several topics, including

- general knowledge of computers, information systems, and system architecture;
- software development methods;
- operating systems; and
- database systems.

AT&T also expects such a hire to have a basic understanding of quality principles and to be capable of applying these principles to the design, development, deployment, and maintenance of all AT&T products, processes, and services. These basic software engineering competencies are part of the Software Engineering Body of Knowledge developed by the joint IEEE-ACM Software Engineering Coordinating Committee (www.computer.org/tab/swecc). Although we expect new employees to understand the basics of each of these topics, we do not expect expertise in any of the areas.

In addition to this basic software engineering education, knowledge of networks and telecommunications is helpful in our environment. During their initial assignments at AT&T, individuals learn about our customers' business requirements, operating processes, and hardware and software systems, and become familiar with the computer systems and peripheral equipment used in AT&T networks and IT systems. In addition, technical personnel must become proficient at business requirements analysis. This typically involves analyzing a customer's business functions, processes, and operations to understand how they affect system design and development.

AT&T also expects software engineers to make effective oral and written presentations of technical material, including presentations to audiences with nontechnical participants. Oral presentation skills consist of the ability to organize and abstract the information and the use of appropriate media and technology to optimize information transfer. Written presentation skills include being able to write technical papers and proposals for both technical and nontechnical audiences.

As software testers become more knowledgeable and fluent in their skills, we sometimes call on them to perform as consultants, either for a customer or an internal organization. In this role, they must provide professional or technical expertise and ad-

> As software testers become more knowledgeable and fluent in their skills, we sometimes call on them to perform as consultants, either for a customer or an internal organization.

Testing has been a neglected area in the software industry for decades. There never seem to be the right resources, nobody knows how much testing is enough, and the test schedule gets squeezed as development overruns and delivery milestones are considered nonnegotiable. Testing should not be seen as a nuisance, sandwiched between development and deployment, but as a critical, professional activity that we need to perform in parallel to design and development activities.

This article describes a very thoughtful approach to fostering an environment in which testing positions can flourish as a professional career path. It also offers many insights that large and small companies can use as a blueprint to define a knowledge and skills inventory for testers and a career path for software testing that is an interesting and viable alternative to development. Even if a company cannot afford the full-blown curriculum for test engineers here, it can use many of the concepts in this article to help with software testing problems.

—Wolfgang Strigel, From the Trenches editor

vice that will aid the organization in making relevant decisions.

### Tester-Specific Skills

At AT&T, newly hired software testers typically have an undergraduate or advanced degree in computer science or a closely related field, but little or no experience in software testing. In fact, we've found it extremely rare for even the best-educated hires to have had formal testing instruction as part of their academic training. For this reason, software testers initially serve in an apprentice role, typically for two years. During this time, they take formal courses in software testing at our corporate education center and are closely mentored by experienced testers. Their goal is to acquire knowledge of basic methods, procedures, tools, and environments used in software testing. They must also display proficiency in the following testing-specific skill areas:

- In the *test planning* area, apprentice testers learn to specify test objectives; test scope and level; and use hardware, software, network, laboratory, and tool resources. They also learn to review system and function testability and determine test scheduling and personnel requirements, sometimes designating specific individuals for specific tasks.
- In the *test case design* area, apprentice testers learn to design and develop test cases and scripts, test specifications, and acceptance criteria, as well as to define expected results. Testers also study functional and specification-based test selection methods and learn about design methods and goals for the various testing stages: unit, integration, system, acceptance, and regression testing. They also learn to properly use hardware and software tools, including test data generators, test drivers and harnesses, capture and replay tools, and system-performance measurement tools.
- In the *test data adequacy* area, software testers learn to make decisions about how much testing to perform and how to determine when something has been sufficiently tested.
- In the *test automation area,* testers learn to use automated tools, which—given the size and complexity of today's soft-

ware systems—are indispensable for efficiently managing test data, executing test cases, and recording and evaluating test results. Software testers must be thoroughly familiar with the functionality, performance, and usability of currently available test tools. It's also essential that they stay abreast of changes in the rapidly evolving tools marketplace and learn to make informed decisions as to whether to build a custom tool or acquire an existing one.
- In the *test execution* area, learning starts with setup activities such as establishing the testing environment, verifying installation and documentation, and coordinating remote resources. Software testers must also be proficient at running test cases and scripts, documenting system performance, identifying problems that occur during test execution, and retesting changes. Following execution of test cases, testers are responsible for results analysis, which includes verifying dataflow correctness within and among tested system applications; defect analysis, which measures and categorizes defects; and root-cause analysis, which ascertains the sources of the problems. In certain circumstances, software testers must perform technical troubleshooting to restore services and solve equipment problems. These tasks require the ability to analyze, identify, isolate, and diagnose faults and their symptoms. In all stages of test execution, testers must assess test results and quality and prepare a software test summary.

Once testers have demonstrated the general and test-specific skills, they can become software test engineers, and optionally apply for test engineer certification.

### Creating a Career Path

Because software testing is sometimes viewed as a dead-end job, the most skilled testers are typically eager to "move up" and become programmers, analysts, or system architects. Software testing is thus often seen as a stepping stone: Good testers move on, regardless of how well they do their job or how valuable their testing work is to the organization. Companies often add to this problem by staffing testing groups with new

hires and allocating only a small percentage of senior people to manage them and provide guidance.

AT&T is attempting to change this. We've developed a software tester career path to give software testers a more defined, challenging, and varied career. We expect this program to increase the professionalism of our testing community, thereby improving the quality of our software and increasing job satisfaction among software testers. Figure 1 shows the different levels in this career path.

### Apprenticeship

The career path's entry point is the software tester position. New hires or transfers from another position start at this point and serve their apprenticeship. In addition to the studies we outlined earlier, testers are expected to attend relevant courses, seminars, and professional conferences to learn about new directions in software testing both inside and outside AT&T. A test manager or highly experienced tester also mentors and closely guides them. In this way, apprentice testers receive both formal education and hands-on experience to acquire the core software testing competencies.

### Mastery

After they have acquired the core competency skills, testers advance to a new level of responsibility as full-fledged software test engineers. They can now be key participants in test planning, test management, and test execution for development projects. Software test engineers must be thoroughly familiar with the quality goals of both the company and their projects. To help testers progress to the next career level, a manager or mentor will continue to guide their development and recommend more advanced courses or other professional development sources.

### Specialization

Following designation as software test engineers, testers can continue to gain experience and knowledge and advance to the next level: software test specialists. To do this, they must establish expertise in one of several areas:

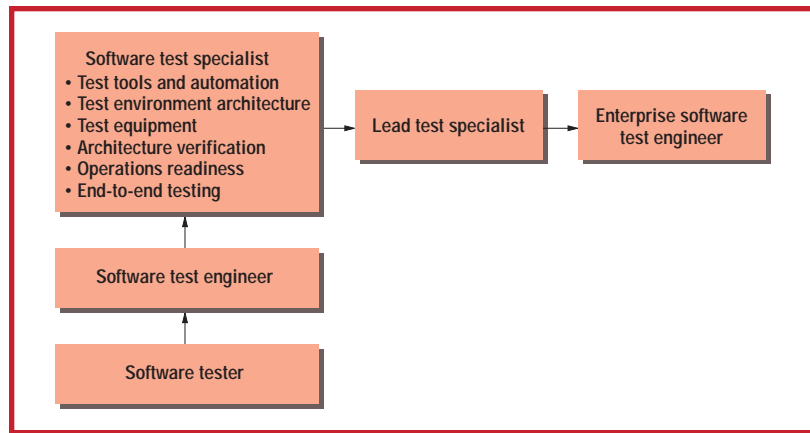- *Test tools and automation.* Specialists in this area design, build, implement, and



maintain automated test cases. They must be proficient in the latest test tools and technology, which requires software development skills. Specifically, we require knowledge of programming languages (such as C, C++, or Java), operating systems (such as Unix or Windows), databases (such as Oracle, Informix, or Access), and testing tools (such as Xrunner or Segue).

- *Test environment architecture.* Specialists in this area design, build, and maintain the software testing environment. This requires experience with the various hardware and software platforms we use and knowledge of system architectures and network communications techniques. Specialists also need knowledge of configuration management and system administration.

- *Test equipment.* Specialists in this area are experts in both the hardware and software of the specialized equipment we use to build and maintain the test environment, including simulators and load generators. Proficiency here requires in-depth knowledge of network elements and telecommunications.

- *Architecture verification.* Specialists in this area design, build, implement, and maintain test suites to ensure the system architecture's performance, reliability, stability, and security. Testers must have knowledge and experience in such areas as performance modeling, software reliability engineering, and security standards.

- *Operations readiness testing.* Specialists in this area design, build, implement, and maintain test suites to determine whether a software system is ready for production. This typically includes all operations administration and maintenance activities, including issues con-

**Figure 1: AT&T's software testing career path provides testers with an opportunity for a challenging and varied career.**

## Certification: Existing Options

Currently, a few organizations offer software testing certification, such as the Quality Assurance Institute, the American Society for Quality, and the British Computer Society (www.bcs.org.uk). At present, the IEEE and ACM do not offer certification programs for testers.

The QAI (www.qaiusa.com) administers a Certified Software Test Engineer Program. Prior to 1999, candidates for this certificate were required to subscribe to a code of ethics, meet certain educational and experience requirements, and provide personal references. As of 2000, QAI is also requiring a certification examination. To maintain certification, successful candidates must complete a minimum amount of professional education each year. The QAI program is approximately four years old. As of January 2000, approximately 2200 individuals were QAI Certified Software Test Engineers.

The ASQ (www.asq.org) began its Certified Software Quality Engineer program in April 1996. ASQ certification requires on-the-job experience—part of which engineers can satisfy with an equivalent educational degree—as well as proof of professionalism and successful performance on a written examination. As of January 2000, approximately 70% of the 1850 candidates for this program have been certified.

The ASQ requires candidates to prove knowledge in several software areas, including

- conduct and ethics;
- quality management;
- processes;
- project management;
- metrics, measurement, and analytical methods;
- inspection, testing, verification, and validation;
- audits; and
- configuration management.

The ASQ compiled its body of knowledge based on the work of software quality assurance practitioners and from a survey of ASQ members. The survey asked members to report on their own activities in the field and asked for recommendations on the basic knowledge newcomers should possess. ASQ has revised the material several times to create the current body of knowledge and examination, both of which focus on common practices.

Neither QAI nor ASQ are licensing boards, and their certifications have no legal significance. However, certification does offer testers a peer-recognized process for demonstrating familiarity with software's standard methods and tools. Thus far, QAI and ASQ have certified a relatively small number of testers. For example, ASQ has certified 1,299 software quality engineers, which is less than 5% of the number it has certified as general quality engineers since 1968. The reason is obviously time-related, but the small numbers of testers certified also underscore the lack of recognition of testing as a legitimate specialty in relation to other types of engineering.

Given the steadily increasing importance of software systems, however, and the increasing pressures to evaluate and attest to their reliability, dependability, and safety, the number of certified software test engineers is likely to grow rapidly. Also, the increasingly pervasive role that software systems play in our lives is likely to stir the engineering community and regulatory agencies to insist on well-defined standards of education, training, and experience for people producing and evaluating software. This might ultimately lead to more certification programs and to governmental licensing of software testers.

Already, there has been some movement in this direction. In 1993, for example, the IEEE Computer Society and the ACM established a joint Steering Committee for the Establishment of Software Engineering as a Profession (replaced in 1998 by the Software Engineering Coordinating Committee). This is relevant to software testers in that effective performance requires a solid understanding of the fundamentals of software engineering. Also, as of 1998, Texas became the first US state with an explicit licensing program for software engineers.[1] This move was particularly significant to testers in that the motivation for establishing the licensing program grew out of concern over the qualifications of people developing and testing embedded-system software components.

### Reference

1. D.J. Bagert, "The Licensing of Software Engineers and Its Effect on Certification," *Proc. Int'l Software Assurance Certification Conference*, Reliable Software Technologies, Dulles, Va., 1999.

cerning user administration, back up and recovery, fail-over, and disaster recovery. Specialists must also be skilled system administrators and have in-depth knowledge of operating systems, hardware, mirroring systems, and databases.

- *End-to-end testing.* Specialists in this area design, build, implement, and maintain test suites for products that span multiple applications or business units. This typically involves coordinating the efforts of many software testers. Specialists must have extensive testing experience, in-depth business knowledge, and project-management expertise.

At AT&T, we expect that most software test engineers will eventually become software test specialists, a process that generally requires several years of technical experi-

ence and additional knowledge acquisition through formal courses, professional conferences, and on-the-job training.

At this stage, testers should also be capable of making sound judgements about alternative approaches, equipment, or tools in their area of specialty. For example, a test-automation specialist should be able to evaluate different automation tools and recommend the one most suitable for a particular application.

### Leadership

Once a test engineer is designated as a specialist in at least one area, he or she can then work toward the next level: a lead software test specialist. There are two routes to this designation: depth or breadth. Following the depth route, test specialists can increase their knowledge and experience in their particular area, demonstrate the ability to lead software testing efforts, and become competent at coordinating many software testers working on complex projects.

Following the breadth route, testing specialists can attain the knowledge and experience necessary to qualify as specialists in more than one area. This makes testers valuable across a range of testing activities and make him or her a particularly good mentor for software testers.

Along with technical testing expertise, lead specialists must have outstanding negotiation, consultation, process management, project management, and oral and written communications skills. Lead specialists are the people others come to for advice and are expected to be mentors for lower-level testers.

Although we expect that most motivated software testers will be certified and eventually specialize, a much smaller and highly select group will become lead specialists.

### Reaching the Top

The final level in our proposed career path is an enterprise software tester. Such individuals must be highly experienced and have a great breadth and depth of knowledge. Enterprise testers will be regularly consulted by top management, and asked to provide strategic direction and advice on emerging technologies for software testing. Enterprise testers must have expertise in all core software testing competencies and

must be test specialists in at least four areas, including architecture verification. Finally, at this level, testers will frequently mentor lead specialists and might be asked to speak at internal software testing courses and symposia. We expect only 5 to 10% of all software test engineers to make it to this level.

### Certification

In addition to our in-house programs, software test engineers have the option of applying for certification, which formally recognizes testing skills and accomplishments. Because the costs of developing an internal certification program are prohibitive, even for a large company such as AT&T, our team's task was to select an ex-

## About the Authors

**Elaine Weyuker** is a technology leader at AT&T Labs–Research in Florham Park, New Jersey. Before moving to AT&T Labs in 1993, she was a professor of computer science at New York University's Courant Institute of Mathematical Sciences. Her research interests are in software engineering, particularly software testing and reliability, and software metrics. She is also interested in the theory of computation, and is the author, with Martin Davis and Ron Sigal, of *Computability, Complexity, and Languages* (Academic Press, 1994). She received an MSE from the Moore School of Electrical Engineering, University of Pennsylvania, and a PhD in computer science from Rutgers University. She is a senior member of the IEEE and was on the IEEE Computer Society's Technical Committee on Software Engineering executive board. She is also a fellow of the ACM, for which she was a National Lecturer and Secretary/Treasurer of SIGSOFT. Contact her at AT&T Labs–Research, 180 Park Ave., Floram Park, NJ 07932-0971; weyuker@research.att.com.

**Thomas Ostrand** is a technology consultant in the Network Services Research Center at AT&T Labs–Research in Florham Park, New Jersey. He previously worked in the Software Engineering Department of Siemens Corporate Research. His interests are in testing methods for large software systems, test generation techniques and tools, and software reliability. He has been an associate editor of *IEEE Transactions on Software Engineering* and a member of the ACM SIGSOFT executive committee. Contact him at AT&T Labs–Research, 180 Park Ave., Floram Park, NJ 07932-0971; ostrand@research.att.com.

**JoAnne Brophy** is a principal technical staff member at AT&T, where she is involved in solution testing, Y2K application conformance, and software test engineering advancement. She has worked at AT&T for 15 years and previously worked as a software developer for scientific research and business applications for Mathematica and the US Defense Department. She has a BS in mathematics from Temple University and an MS in software engineering from Monmouth University in Long Branch, New Jersey. She is a Certified Software Test Engineer, as granted by the Quality Assurance Institute. Contact her at AT&T–Network & Computing Services, 200 Laurel Ave., Middletown, NJ 07748; jbrophy@ems.att.com.

**Rathna Prasad** is a division manager at AT&T in Middletown, New Jersey, where she leads software process and metrics initiatives. She has been a software testing practitioner for more than 15 years, actively involved in verification and validation of large, complex applications. She championed and established AT&T's Testing Center for Excellence and is the chairperson of the Professional Development Team for Software Test Engineers. She has a mechanical engineering degree from Bangalore University, India; an MS in internal combustion engineering from the Indian Institute of Science; and an MS in mechanical engineering and an MS in computer science from the Illinois Institute of Technology. Contact her at AT&T–Network & Computing Services, 200 Laurel Ave., Middletown, NJ 07748; rkprasad@ems.att.com.

# IEEE Software

## Advertising Sales Contacts

**Sales Representative**
Sandy Aijala
saijala@computer.org

**Production information, conference and classified advertising**
Marian Anderson
manderson@computer.org

Debbie Sims
dsims@computer.org

**Advertising Manager**
Patricia Garvey
pgarvey@computer.org

**IEEE Computer Society**
10662 Los Vaqueros Circle
Los Alamitos, California
90720-1314
Phone: + 1 714 821 8380
Fax:  +1 714 821 4010

*http://computer.org*
*advertising@computer.org*

---

isting external certification program. We examined several, looking for one that used criteria similar to our own software testing competencies. In the sidebar "Certification: Existing Options," we offer an overview of two US programs. As our initial choice, we selected the Quality Assurance Institute program, as its requirements most closely match our target software testing skills. Ideally, we'd prefer a certification program that is administered by a professional society, rather than by a for-profit company. However, no such programs exist in the US at this time.

At AT&T, certification does not represent a step on the software testing career path, nor is it a requirement for advancement along the path. However, the company recognizes it as a technical accomplishment and views it as additional evidence of a tester's professional mastery. We've encouraged several experienced testers to apply for certification through QAI, which will help us more concretely assess its strengths. We are also continuing to evaluate additional programs and hope to find or help establish certification programs for the more advanced levels of our career path.

**B**ecause software testers determine whether or not software is ready for release, it is crucial that their decisions be backed by education and training and the confidence that such experiences bring. Currently, we have in place a well-designed first course for software testers and are designing a series of courses for the specialization areas. We are also hosting an annual in-house conference as a forum to discuss all aspects of software testing. The meeting consists of a two-day program of invited talks, refereed papers, panel discussions, and roundtables. Approximately 400 testers from many different organizations within AT&T attended the first conference in June 1999; they were generally enthusiastic about learning current technology and exchanging experiences and expertise with colleagues from organizations across the company. Planning is currently underway for the June 2000 conference. ⊕