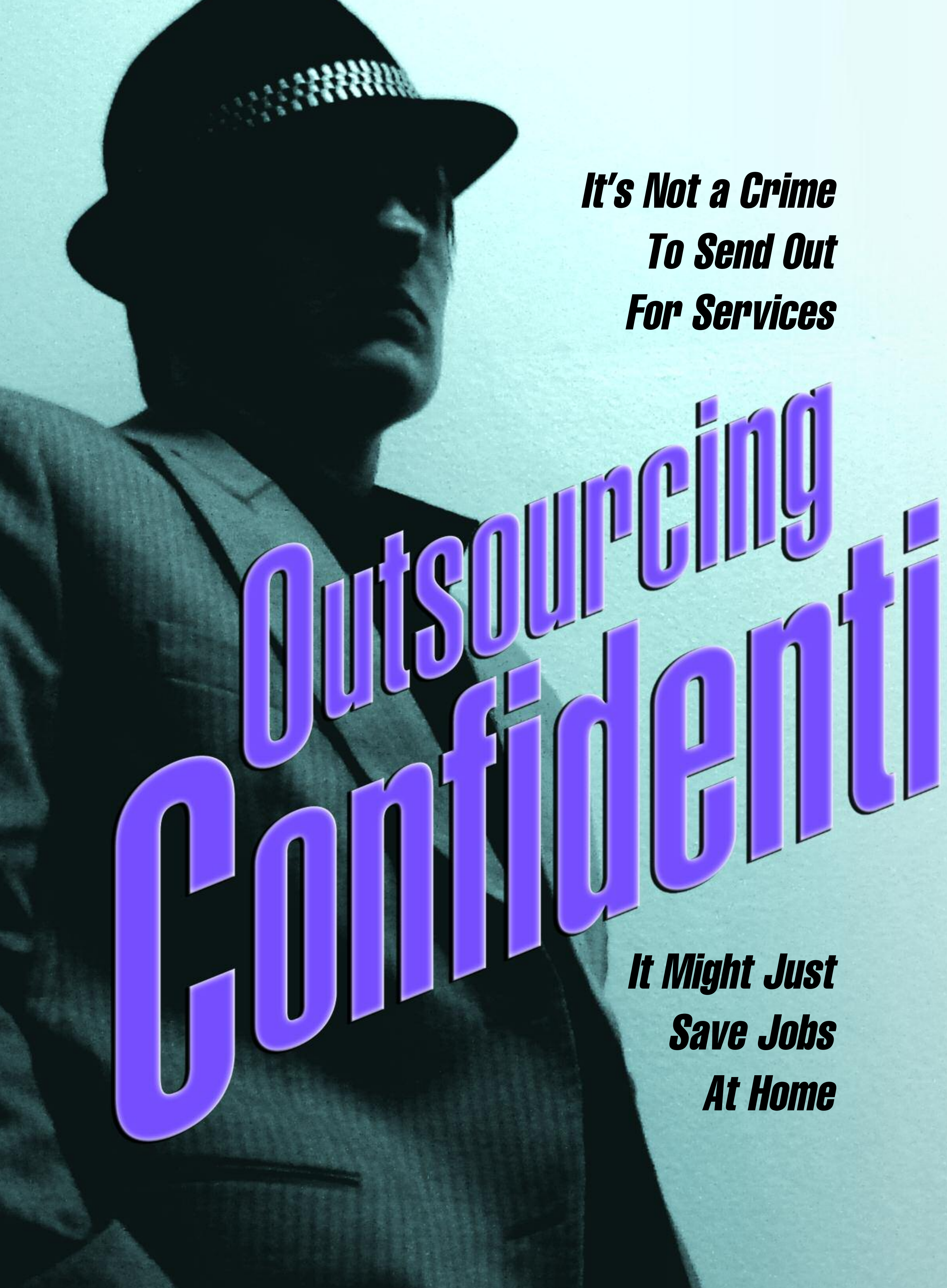

It's Not a Crime
To Send Out
For Services

Outsourcing Confidenti

It Might Just
Save Jobs
At Home

al

By Karen N. Johnson

Outsourcing has been a topic of some controversy in recent years, in part because of the perception that it sends jobs overseas. While numerous software testing centers exist outside the U.S., the practice might just as often engage test organizations around the corner. Regardless of where the work gets done, outsourcing, like any solution, sometimes works and sometimes does not.

In this article, I recount some of my experiences with outsourced software testing organizations in the U.S. and abroad—along with an explanation of the reasons for their success or failure. If your test organization is in need of outside help, you might benefit from the lessons I've learned.

CAN OUTSOURCING BE USED ONLY EXACTLY WHEN IT'S NEEDED?

A few years ago, I was managing a team of 14 software testers. As is typical of software projects, the workload would spike and the schedule would go through a frenetic period. Then the work would drop off after the product was released. During the frenetic period, I would appeal to the contract staff to work long days, nights and weekends to meet release schedules. Within days, I would then need to cut those same people.

[How do you know when it's time to outsource?]

None of this process was a surprise. In fact, the fast ramp-up and quick release of resources was actually part of the plan. It was a resource plan that addressed the needs—the relentless repeatable needs—of testing and shipping a software product.

For such spiky work patterns, the ideal scenario is to find dependable people to use when you need them and cut them loose to keep costs down, yet maintain the dignity of each individual and the overall morale of the team. If you want to be able to use and reuse the same resources, you have got to be able to find a way to let people go without making them feel as if they have been sacked.

Being at a large company, I was required to use only vendors from the preferred vendor list. Resources could not be hired directly by me and had to be brought in through an agency that referred to itself as an outsourced solution provider. I worked with what I had. I had a product to get tested and I needed outsourced staff willing to jump onboard and off with short notice.

Where did I find those magical resources? They were right in my own backyard. Rumor had it that there were a few retired engineers who were talented and trustworthy within miles of the office. They were willing to ramp hours up and down as needed for somewhat unpredictable durations. While the reasons for their flexibility varied, I could always rely on my small pool of retired temporary workers.

What a fantastic solution part-time local outsourcing can be. Factors behind this successful outsourcing were:

- Resources in close physical proximity to the office
- Talented engineers who could quickly integrate with the team
- Relatively few communication issues with the outsourced resources
- Familiarity of contract resources with internal staff and procedures
- More contribution of ideas from contract resources

I recall one time when product development was progressing at a lurch-and-regress pace. The testing team would get a build, and then a critical defect would be found that would block further testing. This pattern had been repeating for a couple of weeks, making it tough to predict workload.

Dave, one of the retired resources, asked me if he could take a few extra days near a weekend to visit with his grandchildren. I said yes. The project manager, sure the software would be ready and the resource would be needed, was furious. But it was my call, and I did not want to go back on my word. Development ran into additional roadblocks. All testing went on hold for several days, the same days Dave was out of town.

The weekend passed, the development team worked long hours and on Monday afternoon the development team announced the software would be ready for the test team first thing in the morning. Dave walked in Tuesday morning just as the build was up and testing was ready to go. Still, the project manager glared at me. The situation could have unfolded in many different ways, but that time, I got lucky.

LUCK IS NOT A PLAN

How do you know when it is time to outsource? The theory that another set of free hands will free you from your problems is just that, a theory. And managers who have done hiring and staffing for a while know it is a job in itself that is never done or stays done for long. In reality, hiring and staffing prob-

Karen N. Johnson is an independent software testing consultant.

Photograph by Alec Maharad

lems are more like a chronic jigsaw puzzle for which the picture continues to change.

Then there is workload, which is like tension on a drum. Too much and the team splits apart; too little and nothing really resonates. With just the right workload, your team hums along like a fine instrument. Tuning workload and resources is a composition to balance.

I am a firm believer in Management by Walking Around. I get to know how the team feels by being around all the time. It is like driving in the car with the radio off; you get to know what the engine sounds like. Sometimes I will sit in someone's cube long enough so that my presence does not cause a Hawthorne effect (see SIDEBAR). Then I can find out what normal is and listen for problems that might not otherwise surface.

After a few release cycles, I begin to see repeat patterns. I developed a sense of when I would need more staff and how I was going to go about acquiring resources. I usually contract staff just ahead of when they are needed. When possible, I found it is a good idea to keep a slight testing backlog in "reserve," so that outsourced resources are never kept idle.

FINDING THE LOWEST LABOR COSTS

On one project, the decision to outsource testing had come from an executive level so high up the corporate ladder that no one on the team knew who. From up there, the process must have appeared simple: Write test cases using on-site contract resources who are knowledgeable of the product, then outsource the test execution using offshore resources to save money.

But if you want to see the cracks in the foundation, you have to be down low and walk its length one step at a time. And the view from down in the cubicles was grim. A closer look at the process revealed redundant and unexpected costly steps.

[A closer look at the process revealed redundancy and unexpected costly steps.]

Step 1: Assemble a group of moderately priced U.S. contract staff, minimally trained on the product, and have them write detailed, step-by-step test cases. There were too many requirements (in the thousands) for them to be read by all of the contractors; each would read only the requirements in the set they were assigned. The theory was that each contractor would acquire the remaining knowledge needed to write the distinct subset of test cases for their particular requirements. Reading and learning was not an activity they were being asked or paid to perform.

Step 2: Ship the test cases to India for execution. The test team in India was trained even less on the product than the team in the U.S. The belief was that if the test cases were drafted in such a way that "anyone" could execute them, the testers could "just" follow the steps. If minimal thinking and knowledge are needed, the decision was made to find the lowest-cost resources possible. Previous testing experience was deemed not as valuable but as a potential cost inflator if the hourly rate went up.

The Indian testing team was instructed to execute the test cases and return the hard copy proof that the cases were executed. Since the product was medical software, the FDA would later audit all the test materials that demonstrated the product was safe for market use. The test materials had to be documented using so-called Good Documentation Practices (GDP) (see SIDEBAR).

Step 3: Each packet of test cases returned from India was reviewed by U.S. resources to ensure that testing had been executed and marked appropriately during execution, in accordance with GDP and FDA guidelines. As a result, executed test cases have what might be viewed as a market value to the overall product

DRY RUN TESTING

Within the community of regulated software testing, the term Òdry runÓ testing refers to testing that is executed according to the drafted test case in order to verify that the test case has been written accurately to the software being tested. The dry run test checks the test setup and steps to ensure the test case has been written with sufficient information and the steps are in the correct order; no recordings or markings are made on the test case itself. What constitutes sufficient information and how detailed the test setup and test steps are written is determined by the team. In many cases, the steps are written in specific detail, especially when the test execution will be done by an outsourced test team.

During or after a dry run, the test case gets corrected until the test case is executable, which in this case means that a tester can run the test case without any issues in terms of the test steps written. Hoping the software works flawlessly is not the goal of the dry run. The focus here is to debug the test case. The test cases in a process such as this become a product of their own, and must work as written. In some environments, test-case errors are written up in the defect tracking system.

In the case of a regulated software environment, dry run test cases are often not maintained or reviewed by an outside authority such as the FDA or any internal audit groups, even though many software defects are frequently found during the initial creation of a test case. In subsequent executions in which the test cases are marked and recordings are made during testing, this testing is often called ÒformalÓ testing, as these are the hard copy documents that will be k ept and shown during an audit as pr oof of testing.

THE HAWTHORNE EFFECT

The Hawthorne Effect states that when someone is being observed, a personÕs behavior changes based on the presence of the observer or on the awareness of being observed. The theory came from a study in 1955, and there have been debates about its merits ever since.

Regardless of whether the theory is valid, it is best to be aware of it when you are trying to observe someoneÕs behavior. Does your presence have an effect on the person or process you are observing?

cost. Mistakes made on those test executions also carried a related cost. The volume of Post-it notes attached to each test case was disheartening; each represented one or more issues that had to be addressed.

It was preferable that the original executor make all documentation corrections (GDP) whenever possible. Mistakes (on documenting test execution) were made, which created a whole review and correction process—a secondary process with associated costs that was not calculated into the cost savings.

Step 4:U.S. resources marked the needed corrections, which were then shipped back to India.

Step 5: The Indian testers would make the corrections and ship the documents back to the U.S. for another review.

Step 6:The U.S. team would make sure the corrections were addressed sufficiently. About a third of the test cases had to be sent through steps 4, 5 and 6 again.

During these weeks and months, packages shipped back and forth. There were packets with test cases for fresh execution, packets with corrections and packets with corrections that were not done correctly and had to be reshipped and corrected again. Packets after packets, FedEx after FedEx. Eventually, the regular FedEx agent arrived each day with a hand cart to move the boxes.

> [It is best to iron out which person is going to do what, at what cost.]

In addition to the obvious courier charges, other costs were harder to measure and would certainly not be visible from the executive level. Among them was the cost of motivation. To turn testing into an assembly line of mini-jobs was disheartening to the team. The process had so many steps and divisions of labor that people felt as if they never completed any work. There was no rhythm, no real workflow and no sense of ownership or accomplishment because each person held the process for only a moment.

From the lower rungs of the corporate ladder, the situation was clear: This process was just not working. It felt as if the decision to outsource was based on the "everybody is doing it" principle, but I did not have the political clout to change things. Later in the project, an executive was let go, and I heard it was political fallout.

The lesson here would have been to cut and run, to abandon the failed plan and engage the team to rethink and redesign.

IN OR OUT OF COUNTRY?

One project I managed included a team in India and another in the U.S. working on two different components of the same system. These two very different teams presented very different issues. In addition, I had to learn much about the company, the product and the project in a short amount of time, as I was contracted as a short-term test manager.

The team in the U.S. consisted of three testers located in a different state and time zone than I was in. This team was hired to perform exploratory testing on a product that was being built in iterations. They learned the product surprisingly fast based on the defects they reported, the types of issues found and how well they were able to report them. I also looked at what priority they assessed and set each defect as another way to determine their project understanding.

One tester—I'll call her Debbie—was especially engaged in the work and began reporting defects early in the process. Debbie's work remained constant and dependable throughout the project. In contrast, "Bob" suffered from a lack of motivation and seemed more interested in his personal life and social events than the testing job at hand. Bob's motivation issues first surfaced with a few subtle comments during conference calls, and his overall lack of motivation became an increasing concern over time.

The third resource was the team lead at the local site. While it was useful to have a point person at the outsourcing company during the kickoff phase, "Alan" had limited involvement during the remainder of the project, and over time I began to wonder if we should be paying for him.

For this engagement, a small amount of advance planning would have saved money later on. Whenever possible, it is best to iron out which person is going to do what, at what cost and for how many hours per week. It is also a good idea for contracts to include language about scaling of hours and resources based on evolving project needs. I also recommend weekly or biweekly resource reviews to keep finances and resources on track.

GOOD DOCUMENTATION PRACTICES

As the name suggests, Good Documentation Practices, or GDP, refers to the use of documentation practices that are considered Ògood.Ó Of course, good is a subjective term and can be an elusive standard to achieve. Overall GDP involves providing clear and sufficient documentation that will address concerns that might arise during an audit. More specifically, companies and products that use GPD have their own training and documentation about what GDP is and how to follow practices that would be considered good. Good implies that the documentation allows an auditor to read any handwritten notes or code comments and eliminates the need for follow-up questions.

As you might infer, GDP leaves considerable space for interpretation. In cases where a software product will undergo an FDA audit, GDP is expected to be used according to how the specific company or division defines itÑor ho w it is defined at the pr oduct level. In the case of testing software that is regulated, testers will usually be trained in GDP before they can begin work. Companies with regulated products are required to maintain training logs for employees, including software testers, and provide sufficient proof that each individual has been trained on GDP.

GDP is one form of Ògood practices.Ó There are many related good practices, including Good Manufacturing Processes (GMP) and Good Lab Practices (GLP). The term cGMP means current Good Manufacturing Practices. Practices vary according to the regulations the product is trying to adhere to and according to what regulatory agency the product is required to submit to. A look on Wikipedia for GxP shows a list of some of the good practices that exist in different industries.

Another outsourcing issue that quickly came to light was motivating people from a distance. How can you get an outsourced resource to feel the same time crunch and overall interest in the work when they are not in the same office and cannot feel the "heat" of impending deadlines? Unfortunately, there is no single answer to this question. Motivating people depends largely on the individual.

I had also contracted an Indian team to perform exploratory testing. The main difference between the Indian team and their U.S. counterparts was that the foreign team did not come up to speed quickly. Team India also entered volumes of defects, but the issues reported were minor. As the volume of defects swelled, the outsourced testers seemed to feel successful, while the local development team was becoming frustrated that so many minor issues were being reported with high priority while other areas of the product appeared untested.

I was also challenged by language barriers and a time difference. Two of the four testers required a translator. Early-morning and late-night conference calls were necessary, and testers were confused that the defects they were reporting were not perceived as valuable.

When the India team ran into issues during their workday, it would be in the middle of the night in the U.S. So I would awake to a flurry of e-mails with questions or partially reported defects. I could not follow up with them personally as it was now the middle of the night for them. Time delays were inevitable.

As I continued to work with the team in India, I discovered that, although the test lead had experience with exploratory testing, the other testers had always worked from test cases. They did not know how to test software with so much freedom and permission to be creative. I found myself pondering different questions: Do we trust who we hire? Do we trust other people to be intelligent, especially when the other people are from a different country and culture than our own?

As productivity-focused people in the U.S., the concept of having someone advance a project while we sleep sounds like a panacea. But is it a reality? The concept of following the sun was beginning to wear on me as I spent more time calculating time differences so I could decide when to hold conference calls. I longed for local retired people willing to work part-time.

Outsourcing is as varied as hiring. It is hiring; it just involves different financial and geographical arrangements. It can work, or it can fail. Outsourcing is not a silver bullet, but it can be part of the mix in solving spiky workloads.

Here are some useful tips to keep in mind when looking to outsource a project:

- Interview everyone on the team, regardless of location.
- Seek retirees; they are often flexible in terms of their hours.
- Tailor contracts to allow for fluctuations in workload.
- Arrange to have a team lead or point person for each location.

> [The concept of following the sun was beginning to wear on me...I longed for local retired people willing to work.]

- Build extra time into the project to accommodate for time zone differences.
- Look for patterns and use them to plan the next cycle.

For a project to be successful, the manager needs to know the team. For mature projects, patterns in the release cycle can provide insight about how the development cycle will work out and help you gauge when to adjust resources. And remember that every resource takes time to ramp up. Account for this by building extra time into your schedule. Every individual on the team requires at least some individual management attention. A good manager pays attention to workers and focuses on building a working rapport. None of these efforts show up on a project plan, but, in the end, they can be as valuable as your budget.