

# Contracts for Testing Services: Settling Disagreements

Copyright © Cem Kaner, 1997, All Rights Reserved.

A story often makes the rounds among lawyers about two companies (and their lawyers) who negotiated the perfect outsourcing agreement. After six months of intense negotiations, their contract covered every contingency. Unfortunately, a few weeks after the project finally started, the two companies had a disagreement. By this point, they'd used up their entire reserve of goodwill and all of their patience for negotiation. The project fell apart. All that negotiating served no purpose.

The lesson of this story is that you can't afford to negotiate every detail of a complex relationship. Instead, you want to reach agreement on the major issues, and on as many other points as you can comfortably cover.<sup>1</sup> To handle the rest, you need a solid, friendly, fair process for making decisions and resolving disagreements as the project goes forward.

In this article, I write from the perspective of the customer (the software publisher or developer who is contracting with a test lab), but the ideas are neutral. The goal is to provide ideas for managing and resolving disagreements, not to provide an advantage for either side.

## ***The Problem***

The typical contract either leaves disputes to the courts or includes an arbitration clause. An arbitration is a formal proceeding, like a trial, but it's run privately rather than by the government. The problems in both cases are essentially the same. By the time you get this far, you and the test lab will probably be furious with each other. The project will have fallen apart long ago. All that's left to argue about is who was right, who was wrong, who owes who money, and how much. This is a disaster, not a dispute resolution process.<sup>2</sup>

Some day, you might need to resort to trial or arbitration after a project fails. But this shouldn't be the only dispute resolution process, or the main one, in your contract. You should make it as easy and as natural as possible to identify and resolve problems quickly, long before they get this serious.

The goal of a dispute resolution strategy should be to help you finish the project successfully, while preserving your good relationship with the test lab. Anything else is a project failure.

## ***The Liaison***

The foundation of effective dispute resolution is good communication.

Your best strategy for preventing, detecting, and quickly dealing with problems with a test lab is to appoint someone from your staff to work on the project with the test lab.

This is not a controversial view:

- Mylott<sup>3</sup> calls this person the *liaison*.
- XXCAL asks its customers to make sure that "a client representative be available for telephone support during the entire course of the test cycle."<sup>4</sup>
- ST Labs also urges its customers to appoint an internal test team to delegate work to the lab and supervise the results on an ongoing basis. "Over the course of hundreds of projects, we've found that an outsource project probably will not satisfy either party if there isn't frequent communication between them."<sup>5</sup>

The liaison serves as the primary contact between your company and the test lab. She is the primary reviewer of the lab's work, including test plans, bug reports, status reports, test suites, and all other deliverables from the lab to you. She also works to understand this lab's business and technical practices and the practices of test labs in general. With this knowledge, she provides your company with insight and she serves as a more effective negotiator with the lab.

The liaison is the person that your staff complain to when they have problems with the test lab, and she is the person that the lab staff complains to about you. For example, when one side says that the bug reports are badly written and irreproducible, and the other side says that the programmers are just trying to avoid taking responsibility for their own bugs, your liaison is the person who is digging through the bug reports trying to understand who needs more technical training and who needs attitude adjustment.

Your liaison should be calm, able to assert herself without shouting, and able to deal with pressure and whining from all sides. She should be diplomatic but firm, able to explain your company's needs to the outsourcer without being obnoxious, and able to present the outsourcer's legitimate gripes with your company to your management without getting fired. She should be an excellent listener, tightlipped, and personable enough that the outsourcer will confide in her. She should have, and deserve, an air of solid credibility and integrity. She should be detail oriented and methodical. She should understand testing, test management, software development, and software project economics. Finally, she must be loyal to your company.

The test lab will probably have its own liaison (they might call him an account manager or a project manager).

The two liaisons, yours and the lab's, are the first-level negotiating team for resolving problems as they arise.

## ***Disagreements***

What kind of disagreements come up between a software developer and the test lab? Here are some examples:

- The latest version of the software is too unstable to test. It takes days (or weeks) for the developer to deliver a stabilized version. What is the test lab's staff supposed to do during this time? What if they have little productive work to do on your project? Should they work inefficiently, at your expense? Should the lab reassign them to another project, saving you money but possibly imposing an additional delay until they can rejoin your project?
- The lab promised four seasoned testers, full time. It seems to you that you're only getting about two testers' worth of productivity. Testing is going very slowly, but the lab has no additional testers to add to the project.
- The lab is testing your competitor's software at the same time as yours. The same testers are seeing both products. One tester told your liaison about a slick new feature in your competitor's program. You believe that your competitor would consider this feature a secret. You don't want your competitor to learn about your slick new features.
- You loaned the lab several computers and printers. It has returned a few to you, but they were so badly packaged that they were trashed in shipping. The lab also seems to have lost your \$10,000 color laser printer, and they're asking you about some equipment that they claimed they loaned you, that you've never heard of.
- The lab gave you a sketchy test plan and you didn't like it. You made extensive comments. They didn't make many changes, and you're still not satisfied. The lab is entitled to a payment when they reach the first milestone, "Test Plan Draft 1 Complete." They want their money, but you want a better first draft.
- The lab's testers seem clueless, at least the ones assigned to your project. The people that you met when you interviewed the lab, and toured the lab, were very sharp but none of them are assigned to your project. You want testers who are more senior, but the lab says they'll cost more money. You don't want to pay more because you thought that the rate you'd agreed to was for the senior testers.
- The lab agreed to test your software for a flat rate of \$100,000. Since then, you've changed the feature set six times. The latest feature includes modem support, and now several people can use the program at the same time, seeing and editing each other's data. You also want the program tested for compatibility with the new (still in beta) Windows 98 and NT (these didn't used to be a requirement, but they are now). The test lab wants more money to pay for the extra testing. You want to pay the flat fee, \$100,000.

There are plenty of other possible problems, but these illustrate the point. Some of these are big—a lot of money is involved. Others might just involve some retraining and some resetting of expectations.

## ***A Strategy for Resolving Disagreements***

You want to identify and solve problems as early as possible. The longer they drag on, the more damage they do. They hurt the project, and they hurt your ability to trust and work with each other.

### **Start with discussions at the liaison level**

If the liaisons can straighten the problem out, no one else has to deal with it.

Negotiating takes some skill and your liaison may need training in it. You want to use ethical methods of negotiating with the goal of preserving a business relationship.<sup>6</sup>

## Gradually escalate through management levels

If the liaisons can't resolve the disagreement, they should both take up the matter with their managers. The contract should describe this as part of the process—the first level managers will “meet and confer” to try to resolve the problem. If they can't reach agreement, the dispute goes up another level, and two more managers are required to meet and confer.

Sometimes, this sounds better than it works. Go up a few management levels and neither side might understand the issues or the technology. The result can be a deadlock in negotiations or a set of uninformed and ill-advised decisions.

## Use independent experts for fact-finding, arbitration, and mediation

Rather than continuing up the management chain, it might make sense to try something else. If your companies can't reach agreement when the first or second or third level (decide what level and write it into your contract) managers meet, get some help from an independent expert.

The expert should be someone that both of you trust. In your contract, you should list a few mutually acceptable people. You want to make this list before there's a problem because it can be difficult to agree on which people to consult if you wait until you're in the middle of a significant dispute. The list includes a few people so that if one isn't available, you have other choices.

The expert should have no other role in the project or in either of your companies—you want to avoid conflicts of interest. When you have a dispute to settle, and you call one of the experts for help, require him to list his history with both sides and any other conflicts of interest involving the project. If there are conflicts, you should be able to refuse the services of this expert. Call the next person on the list.

### Independent fact-finding.

In this case, the expert meets with your staff and the test lab's staff. He looks at documents and writes a detailed report that describes the disagreement between you and the lab. He recommends a course of action. The report should be direct, but should not insult either side. The report goes to the managers who couldn't reach an agreement before. They should then meet and discuss it.

If the main problem was a misunderstanding, or if one side was trying to get away with something that just won't fly once the other side understands what's going on, the expert's report should help settle the dispute.

Even if you and the lab don't reach an agreement when you read and talk about the report, the report will provide useful background to the mediator or the arbitrator.

### Mid-project mediation.

A mediator doesn't have to be an expert in the technical matters because he doesn't make the decisions. But he must be literate in the technology. The mediator meets with you and with the lab, tries to understand each of your positions, tries to help each of you understand the other's point of view, and tries to help you do creative problem-solving. The goal is to help you both reach a agreement that satisfies both of your needs. Often the agreement that results from mediation involves a plan that neither side would have thought up on their own.

If you reach an agreement, write it down, sign it, and move forward with the project. If not, try arbitration.

### Mid-project arbitration.

This is a session lasting 4-8 hours, in which both sides present the dispute to the expert. The arbitrator listens carefully, asks questions, and issues a ruling within a week. She decides who has to do what and who has to pay for it. The goal is to get a clear decision on the issue right away, and move forward. Even if the arbitrator makes the wrong decision, it is often better to move forward than to drag down the project by prolonged fighting over an issue that won't go away.

The contract should let you and the lab settle the dispute on your own terms before the arbitrator gives her decision. People often settle disputes at the last minute, when it looks like someone else will make the decision for them if they don't deal with their issue themselves. If you reach an agreement after the hearing and before the arbitrator gives a decision, call the arbitrator and tell her not to give you a decision.

Your contract might have special rules for large disputes (involving large sums of money). For example, the negotiations might go to more senior managers before they go to arbitration. You might use a three-arbitrator panel instead of a single arbitrator. At least one of the arbitrators should be a lawyer with extensive arbitration experience. The hearing might last longer than a day. The goal is still the same—make a decision quickly and get the project moving again.

## **Conclusion**

A contract should provide rules for protecting and preserving the deal you've agreed to, not just rules of engagement when the deal falls apart. It should help you protect the deal by providing simple rules for settling disagreements before they kill the project. The approaches discussed here are designed to help you resolve disagreements while your project is still alive and promising. They appear too rarely in contracts. Think seriously about including them in yours.

THE FOLLOWING SECTION IS AN OPTIONAL SIDEBAR. I THINK THAT READERS WILL FIND IT USEFUL, BUT IT IS A DIGRESSION. THEREFORE I SEPARATED IT SO THAT YOU CAN STRIP IT IN IF YOU HAVE SPACE.

## **Typical Tasks of the Liaison**

The liaison might not personally do every one of these tasks, but all of them should be done under her supervision.

1. Source of all material sent to the outsource test lab. *Anything* that goes to the test lab goes through the liaison.
2. Archives all releases sent to the outsourcer.
3. Receives all material from the outsourcer.
4. Archives all materials received from the outsourcer.
5. Reviews all test plans, automation architecture documents, test suites, and all other documents from the outsourcer.
6. Identifies tasks that the outsourcer is not going to do (tasks you would normally do but that the contract does not require the outsourcer to do.) Flags these in memos to management.
7. Control point for configuration testing: approves the list of configurations to be tested, probably providing market data to support construction of the list.
8. Carries and manages Tech Support requests for information on the results of testing. For example, Support might want results of configuration testing in a standard format that makes it easy to handle customer queries about the tested devices.
9. Reviews all bug reports. Tries to replicate and extend each bug. This is valuable. The liaison understands your quality standards and your market better than the outsourcer. She will see implications of these bugs that the outsourcer misses. She will also understand the work product of the outsourcer better, in more detail, this way.
10. Looks for communications issues in handling bugs. Are your staff being reasonable when they comment on bug reports? Are they legitimately asking for more information, or claiming irreproducibility, or are they wasting the outsourcer's time? Is the outsourcer competently and quickly handling requests for more and better information?
11. Audits the outsourcer's testing effectiveness. This may include running independent tests, monitoring bug reports from other people, inspecting the test plan to see whether it would catch specific bugs, etc.
12. Monitors the outsourcer's compliance with contractual performance standards. For example, if the contract specifies 90% branch coverage, the liaison should ask for output from a coverage monitor that shows the amount of coverage actually obtained.
13. Reports on project progress to you and your management.
14. Meets with the outsourcer to convey information about schedule, timing and content of upcoming deliveries, and to communicate your problems and frustrations.
15. Meets with the outsourcer to get information about schedule and progress, and to listen to the outsourcer's difficulties and frustrations.

16. Negotiates with the outsourcer to try to resolve disagreements, and to obtain factual information that can help rationally frame negotiations between higher-level managers.

---

<sup>1</sup> In another column in this series, I discussed the liability and indemnification clauses that often appear in these contracts and sometimes cause interminable negotiations. Kaner, C. (1996) "Contracts for Testing Services," *Software QA*, Volume 3, #5, p. 20. Future columns will examine a few other troublesome clauses.

<sup>2</sup> Kubey, C. (1991) *You Don't Always Need a Lawyer*. Consumer Reports Books.

<sup>3</sup> Mylott, T.R. III (1995) *Computer Outsourcing: Managing the Transfer of Information Systems*, Prentice-Hall.

<sup>4</sup> XXCAL Inc., *Why XXCAL*. Document downloaded from XXCAL's web site, [www.xxcal.com/why.htm](http://www.xxcal.com/why.htm), on March 25, 1997.

<sup>5</sup> Bach, J. *Things to Think about Before Outsourcing*. Document downloaded from ST Labs' web site, [www.stlabs.com/outsource.htm](http://www.stlabs.com/outsource.htm), on March 25, 1997. Also, see the summary of a USENET discussion of this by Kaner, C. and Bach, J. (1995), "Getting the most from outsourced testing," *The STL Report*, Nov./Dec. issue, available at [www.stlabs.com/most\\_out.htm](http://www.stlabs.com/most_out.htm).

<sup>6</sup> Freund, J.C. (1992) *Smart Negotiating: How to Make Good Deals in the Real World*. Simon & Schuster. Fisher, R. & Ury, W. (1991, 2<sup>nd</sup> Edition) *Getting to Yes: Negotiating Agreement Without Giving In*. Penguin Books.