

# Contracting Issues in Software Testing Outsourcing

Cem Kaner, J.D., Ph.D.

Professor & Director

Center for Software Testing Education & Research

Florida Institute of Technology



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH

# Outsourcing

- Delegation of work to some organization out of your immediate control
  - might be an outside firm or another organization in-house
  - might be your entire test organization (all projects) or just a single project or just a single task on a single project
  - might be done with the goal of removing work from your company or with the goal of developing expertise inside your company
  - might be done to satisfy your own requirements or to satisfy a contractual obligation of independent testing



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH

## Some Risk Factors: Technical & Process

- Clients (including management) engage in wishful thinking about schedules, capability & cost.
- Outsourcing requires understanding of the application subject matter, the specific product, the client's development process, and the customer's needs. The more remote the outsourcer (distance, culture, experience with this market), the more serious the problems.
- Outsourcer needs documentation of product and process, which can create new costs or substantially increased costs.
  - We should not make the mistake of thinking that it is self-evident that such documentation is intrinsically desirable.
  - This drives client to lock down the design.
  - May cut off iterative refinement / discussions among stakeholders.
- ***Client fails to manage the work.***



## Some Risk Factors: Legal

- Clients (including management) engage in wishful thinking about schedules, capability & cost.
- All sides often lack contracting experience (except the lawyers, who lack software experience).
- Outsourcers may promise too much, or appear to promise too much. Strong cultural differences in presentation and self-perception of expertise.
- Potential for security leaks, loss of intellectual property rights, conflicts of interest. These are exacerbated by cultural differences.
- Performance standards are unknown, unused, or not costed out in a rational way.
- Poor contingency planning.
- Conflict resolution process is weak/undefined.
- ***Client fails to manage the work.***



# Common Resulting Problems

- Outsourcer may be blind to usability and capability issues because they don't know the market, don't understand the culture or application area, or don't feel authorized to challenge the design.
  - Design bugs missed or unreported.
  - Design discussions triggered by local testing may be delayed or never take place.
- Increased bureaucracy and bureaucracy costs may be unappreciated, and blamed on the people who advocate for them.
- Project may stall when outsourcer staff stop work because they don't know what to do or how to do it.
- Remote testing may be superficial because testers are not subject matter experts and are not supervised by people who understand the relevant risks.
  - Product looks more stable than it is until after release.
- Iterative development can cause huge cost overruns.



# Suggestions

- There are many things you can do to mitigate outsourcing risk, some of those are contractual / legal.
- In 2000, I listed hundreds of issues to consider in a long paper at <http://www.kaner.com/pdfs/outsource.pdf>
- This talk adds some notes from a meeting of the 9th Software Test Managers Roundtable, held at Florida Tech in May 2004 on “The Offshoring of Software Testing.” Participants in the roundtable included:

James Bach

Scott Barber

Bernie Berger

Ross Collard

Terry Dirr

Ken Duesling

Michelle Gross

Douglas Hoffman

Paul Holland

Kathy Iberle

Cem Kaner

Michael Kelly

Dave Liebreich

Todd Little

Hung Quoc Nguyen

Drew Pritsker

Patrick J. Schroeder

Dana Spears

Mary R. Sweeney

Andy Tinkham

Dennis Van Horn



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH

# Role of Contract and Legal Structure

- Lawyers often draft contracts from checklists (like the one at [kaner.com](http://kaner.com)), but the risk is that we'll spend too much time trying to specify everything.
- You want a formal agreement.
- You can't plan for every contingency. You want
  - a broad agreement on scope, objectives, and schedule
  - clear agreements about money and authority
  - insight into
  - a solid dispute resolution procedure.
  - If you know more details, include them. At some point, you have to shake hands and start working.



# A Few Observations

- Several test groups burn off credibility and bargaining power fighting inevitable changes.
- With the management and process costs involved, offshoring may or may not yield net savings. It may yield the savings at the expense of quality.
- Sometimes offshoring **improves** quality, especially when the tasks offshored are specialized and the outsourcer is expert in that area.
- Full-project offshoring might be more successful than offshoring only testing.
- Keep in mind the difference between outsourcing of manufacturing and outsourcing of services.
  - Services are inherently local, with service provider constantly adapting to the needs of the client.
  - Some organizations (and some “quality” advocates) see outsourcing as an opportunity to drive software development to be more like manufacturing than service. This is an enormous cultural change.
- Offshore testing services are expanding rapidly, hiring junior staff.
- How will your outsourcer manage staff turnover?





# Suggestions

1. Appoint a local liaison.
2. In the contract, try to be clear about the scope of the work.
3. Clarify key issues of roles and responsibility
4. Be clear about the pay structure and its implications
5. Look for the strengths of the outsourcer and manage other resources to fill in for the outsourcer's weaknesses.
6. Don't rely only on written agreements and documentation.
7. Qualify the training and facilities of the outsourcer. Evaluate their training program and their work on a continuing basis.
8. If specific people are touted as "on the project" have them assigned to the project by name.
9. Create a mid-project arbitration procedure.
10. Create clear schedules with deliverables that you appraise. Plan to revise your schedules to conform to reality.
11. Apply or create a clear status reporting process.
12. Apply or create a clear bug management process.
13. Plan for unexpected contingencies.



# Local Liaison

- Your company **MUST** supervise the effort.
  - replicate all bugs and look for more serious consequences.
  - audit (don't just review) the test plan. (How will you find holes?)
  - look for communications and technical problems
  - monitor progress
- Your company needs a liaison to supervise / talk / settle problems with the outsourcer.
- Your company's liaison needs enough time to do a competent job.
- If work is done offshore, you and/or the outsourcer should manage the project with someone familiar with the outsourcer's workforce culture, who visits the outsourcer's work site frequently.



# In the contract, try to be clear about the scope of the work.

- Entire testing department (*We give up testing.*)
- Entire platform (*We give up testing Macs.*)
- Entire product line (*We give up testing word processors.*)
- Entire product (*We give up testing BuggyWord 5.0.*)
- Maintenance release (*We give up BuggyWord 5.01.*)
- Entire language (*We give up BuggyWord in French.*)
- Class of issue (*We give up testing printer compatibility. We give up testing against IRS regulations.*)
- Buy skill (*Give us a test plan or an automated test suite and show us how to use it. Show us how this tool works.*)
- Additional testing (*We want audit / usability test / sanity check.*)



# Clarify key issues of roles and responsibility

- Who has final authority on testing priorities?
- Is the outsourcer to use independent judgment or do you (or someone else) have authority?
  - If you have authority, you have pragmatic tradeoffs to make between the (hopefully) expert testing judgment of the outsourcer and the (hopefully) informed judgment of your staff as to what features and issues are important to your customers.



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH

# Be clear about the pay structure and its implications

- There are choices
  - hourly
  - flat-fee
  - flat-fee plus time-and-materials
  - flat-fee with incentives and disincentives
  - piece-work
    - bugs?
    - function points or KLOC?
    - configurations?
    - what else?
- Each type of fee structure is open to abuse
- Hourly contracts give you more control than flat-fee



# Outsourcer Strengths & Weaknesses

- Some outsourcers are specialists.
  - Example, some companies outsource localization testing because they believe the outsourcer does a better job than they would.
- Other assignments are more broad, and may be more broad than the skill set of the outsourcer
  - Example, suppose the outsourcer is primarily skilled at routine test automation. How will you collect information on design inadequacies?
- Do they know your market?
- Do they have an equipment lab?
- How junior are their staff?
- Who will do your glass box testing?
- Can they measure coverage for you?
- How well do they understand your application domain?
- How well do they understand your environment (O/S, etc.)?

***Work with the outsourcer to appraise strength and weakness.***

***Develop a mitigation plan for weaknesses (and get mgmt approval)***



# Don't Rely Only on the Document

- Misunderstandings are routine.
- Written documents are ambiguous because the language we use is ambiguous. This will not change soon.
- Even if the document is not ambiguous to an American reader, it will gain ambiguity as it is translated by a reader from another culture.
- To be sure of communication, you need to monitor the work products and the progress, and give feedback about the details of the differences between what you are receiving and what you expected.



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH

# Qualify and Monitor the Training Outsourcer's Staff

- Look at the discussions on the yahoogroups testing lists. It is common to see someone who claims to have just been hired as a tester (or test manager) who is asking the most basic questions, and others who claim to be experienced, providing answers that are wrong or are textbook-correct but reflect no personal experience.
- Even if your outsourcer shows you experienced staff, why would these staff stay at work with that outsourcer? How will the outsourcer manage through this turnover?
- Certification is a weak answer, because the exams are too easy, rely too heavily on memory work rather than experience, and answers are frequently posted on the Net.
- Free courseware is available
  - (e.g. <http://www.testingeducation.org/k04/index.html> )
  - might become basis for required self-study and harsher certification testing



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH



# Assign Key People as Part of the Contract

- A common trick is to introduce you (whoever qualifies the outsourcer or negotiates the contract) to specific senior people or to show you specific facilities,
  - The implication is that these people or facilities will be assigned to your project
  - But then they are not put on your project or they are available for only a miniscule amount of time.
- If you rely on specialized individuals or facilities, it is essential that you include them (by name) in the contract



# Mid-Project Arbitration

- Why is this so important to resolve upfront?
  - a good process lets you work with less tightly defined deals.
  - typical processes (lawsuits, arbitrations, putting up with it) are unsatisfactory.
  - ***Resolving disputes can be done in a friendly way, if you plan for it.***
- Ideas on resolution, ***mid-project*** (before it's too late)
  - meet and confer
  - escalating discussions through management
  - mid-project mediation
  - independent mini-arbitration
  - independent fact-finder
  - ***Agree up-front about these third party helpers***



# Schedules & Status Reports

- Of course you want regular status reports.
- You need a way to predict progress and map actual progress against predictions
- You need factually verifiable milestones or other status indicators
- How much detail are you willing to pay for?

Detail = money



CENTER FOR SOFTWARE TESTING  
EDUCATION AND RESEARCH

# Bug Management Process

- Whose bug tracking system, yours or theirs? Why?
- Replicate and evaluate the bugs you receive.
  - Evaluation suggestions at [http://www.testingeducation.org/k04/bbst28\\_2004.pdf](http://www.testingeducation.org/k04/bbst28_2004.pdf)
  - Provide feedback to the reporting testers
- What is the bug lifecycle from report to closure?
  - Who at your site sees the reports?
  - Will they stimulate the level of discussions that in-house reports stimulate? If not, how can you convert these outsourcer reports to local reports?



# Plan for Unexpected Contingencies

- What if the outsourcer completely screws up?
- What about the unexpected (if inevitable) last-minute shuffling of the feature set?
- What if the software is untestable?
  - For how long?
  - What do the outsourcer's staff do in the interim?
- What about staff turnover?
  - *(How much are you willing to spend to manage this risk in advance? Maybe you will choose to leave this an open risk.)*

