

Offshore Outsourcing - Gather Data (Part 6)

Surendra Saxena

No outsourcing can work unless you can nail down requirements. Just as important is how well you can measure how these requirements can be met. In essence:

- Get complete requirements
- Identify acceptance and performance criteria
- Put measurements in place

The more fine grained the requirements the better will be the results. Vagueness will result in deliverables that just may not meet the needs of the users. Time and distance will add to the unpredictability of the output if the requirements are vague.

The foundation to a successful relationship is to understand your own responsibilities. There are some things that you just have to do before you plan to outsource applications development:

- Formulating Requirements
- System Analysis including use case scenarios
- Overall System Design
- Infrastructure specifications and design. This should be owned by you. Infrastructure includes hardware, operating systems, network, tools, contractual and billing implications (especially in Web services architectures), application servers, databases, eai technologies etc.
- Performance specifications
- Service levels desired
- Standards to be followed for development, testing and documentation
- Standards for release/ packaging
- Acceptance criteria

To be fair to your OSP have your acceptance criteria ready before you outsource. Your OSP should be clear from the beginning on the basis of acceptance. Thus define every thing that is required for you to accept the software. This could include Functional acceptance, Usability acceptance, maintainability acceptance, and performance acceptance. Define documentation expected and standards to be followed for documenting the code.

What cannot be measured cannot be improved, put measurements in place. Look at your existing operations and establish

- Performance measures
- Fiscal measures

Typical performance measures for a Maintenance project could be modules by type and complexity, productivity of a team and individual for complex, average and simple bugs, average trouble tickets fixed by complexity, percentage of trouble tickets to existing trouble tickets (Release change ratio), defect density, detection efficiency, removal efficiency, average fix per period, average time to repeat simple, medium or complex bug, average time lost due to improper fix.

Cost measures could be current cost per fix by complexity, existing cost of operations.

When you outsource a piece of work the expectation is that your OSP will significantly beat your performance in cost and performance. This will not happen at the outset but progressively they will.

Therefore when you gather data you are actually looking at your current service level. This will establish a desired service level for your OSP:

- For volume of work to be done
- By value at which that work needs to be done

This is your baseline, now as the OSP gets more and more experienced with your work improve the baseline. Incentivize and reward your OSP as he betters his performance.

When you outsource new applications development the process can be even more challenging. If you lack the rigor of formal processes or are unwilling to set formal processes there is an increasing likelihood of failure. Checks and balances are required not for micromanaging the service provider but to ensure that you are getting what you want. Be prepared for iterative approaches to building your solutions prototyping and testing at every stage of the process to ensure conformance to requirements. Set your team to be able to test incremental releases.