How to Manage an Outsourcing Relationship

For Executive Management and IT directors

A few years back, outsourcers in countries like India, Russia, and China began capturing a trickle of software development projects. Today, the outsourcing trend has turned into a tidal wave: Meta Group estimates that outsourcers this year handled 41 % of new development activity. And the percentage is rising, even among commercial software companies that have long insisted that in-house developers are part of their competitive edge.

But working with programmers on the opposite side of the world can be a challenge. Managing an outsourcer is clearly very different from managing an in-house development group, and we've found remarkably little discussion of how to make outsourcing relationships more successful. So, we decided to share our expertise in Offshore IT Outsourcing that our Project Management Team has gained during the implementation of more then 200 projects in the last 3 years. We offering the following insights:

1. **First, think about how you'll interact with the outsourcer:** "Most people will agree that you can't just define a project and 'throw it over the wall' to an outsourcer, "Yet there's a common misconception that you start out by preparing a product spec and then just hand it to an outsourcer." In actual practice, project specs emerge through "a process of progressive investigation and analysis"—and the success of this process largely depends mostly on how completely the outsourcer and client *communicate* with each other about the project goals and the client's overall business strategy. "Here's a true story," "A software company hired an IT outsourcer to develop a prototype, with the stated goal of attracting investor funding. However, the company also planned to use this prototype for product testing among potential users, a goal that was never communicated to the outsourcer. Because of this failure to clarify the project goal, the software didn't have the performance and scalability necessary for user testing."

2. **Start small:** "Your first goal, and the Offshore IT Provider's, should be to make your initial project a success, This goal is a prerequisite for meeting your other goals for outsourcing, such as cutting costs or getting to market faster. One way to succeed is to choose the right kind of pilot project, one that's amenable to success, so you can forge a relationship and learn how to work effectively together." Good projects involve minimal risks, are relatively easy to define, and have few dependencies

3. **Don't get derailed by the RFP:** Clients sometimes rely on a detailed Request for Proposal to collect bids from a dozen or so potential Offshore IT Service Providers, and then hand the bids over to a purchasing department for final selection and pricing. That's amateurish. "From the outsourcer's point of view, RFPs usually contain only 50%-75% of the real requirements, at best. At worst, they contain errors in estimating the size of the work and the skill levels required."

4. **A better approach** is to start by identifying just two or three likely outsourcers and then work with them to prepare a more substantive Project Overview document. "The PO should start with high-level descriptions of the business need and the outsourcer's solution, It should also describe delivery mechanisms, acceptance criteria and procedures, and, of course, a timeline and price for delivering the solution. A PO also covers legal and contractual issues such as ownership of property, terms and conditions of payment, and limits of liability.

5. "The PO will tell you a lot about the outsourcer besides its price. The more details you have about how the product will be delivered, tested, and accepted, the fewer unpleasant surprises you'll receive in the future."

Do a reality test on the schedule and deliverables: It's tricky to compare an outsourcer's hourly costs against the client's in-house engineering staff, but clients can certainly see if the project is adequately staffed. "The PO should break down what type of resources will be employed at each phase of the project—for example, developers, architects, quality assurance managers, project managers, etc.—and how much time they'll spend. A low price estimate may be due to the fact that one outsourcer only allotted eight hours for acceptance testing with a junior developer, while another allotted 40 hours and included both a developer and a QA manager."

6. What's a realistic allotment of time for specific phases? A good rule of thumb is to expect the following percentages:

    - Requirements Analysis: 10% of total project time
    - Specifications: 20% of total project time
    - Coding and testing: 40% of total project time
    - Integration and system testing: 20% of total Project time
    - Acceptance testing: 10% of total project time

7. “In addition, you should expect the price given in the initial PO to include a firm figure for a discovery or analysis stage and an *estimated* cost for the design, development, and deployment phases, which will be finalized after the discovery phase,” he says.

8. **Be clear about milestones and acceptance:** For commercial software developers, the decision to stop work and ship a product is often driven by market demand and competitive issues.

9. Points out that outsourcing relationships generally work best with more **formal**, measurable **decision points**. “Acceptance shouldn’t hinge on a subjective judgment such, as ‘I don’t think it does enough,’” he says. “Be sure to designate someone as an ‘acceptor’ responsible for making a timely decision, and provide a mechanism for bringing together both parties to discuss changes to the requirements and how the change will affect the schedule and the price.”

10. **Clarify the business requirements:** Typically, a newly hired outsourcer will spend time talking to end users as well as management to create a comprehensive specifications document that drills down even further than the PO does. “The goal is to ensure that developers are grounded in the business context. By understanding the business reasons for what they’re doing, they can foresee problems and devise better solutions. Linking the technical and business requirements in writing keeps everyone on track

11. **Your Intelligence Solution**

SCR SOFT can help you measure, monitor and ultimately improve the quality of your customers’ experience. SCR SOFT delivers information addressing critical questions regarding wireless data performance, such as:

- What is the actual, end-to-end experience of your Internet/ wireless users?
- How does your average download time compare to your target?
- How does your download time or network performance compare with the competition?
- Does the customer experience vary by geography?
- Does performance vary by carrier, by platform, by protocol?
- What can you do to quickly diagnose a problem and fix it?

SCR SOFT’s system monitors the same traffic route as a real end user *(i.e. from the browser, through the carrier gateway, etc.).*

Inside a Business Requirements and Functional Specification

Here are some of the topics that in our opinion a good requirements document should contain:

Business Requirements: This describes the business problem to be addressed and defines and enumerates both the business requirements necessary to solve the problem and the business objectives. Business requirements include new system and user activities such as “will generate usage reports,” “will provide user authentication functionality,” or “will allow designated users to perform a named function or action.” Examples of business objectives are “will decrease the lead-to-sales-closing time,” or “will reduce the maintenance costs of the service center.”

Technical Requirements: This outlines the hardware and software environment, platform considerations, any existing system information, and any known limitations and/or technical problems. It notes system functions and components, including changes and additions to existing software. Examples might include report, data, and security components. Also called out are out-of-scope features and exclusions, assumptions, and dependencies. For example, “the software must be accessed over the Internet,” or “the solution assumes that user authentication is already in place.” To be sure you get what you want–be sure this information is defined.

General Requirements: This section defines functional requirements and non-functional requirements.

- **Functional requirements.** Each system function or component listed in the technical rationale is broken out and explained. For example, for a report component, the description would name the reports generated by the solution.

- **Non-functional requirements.** This section describes the tools and technologies used by the proposed software, its hardware and software environment, and number of users and concurrent users. It also defines critical system characteristics such as performance (with metrics), integrity (“data for reports must be complete and valid”), and portability (“the software must comply with J2EE v1.5”). Characteristics also include what the new software should not do–for example, “it will not affect existing system X.”

- **Specific Requirements:** The more you break down the software and define how each part works the better. Details should include the purpose of each function, inputs and outputs, interaction points out that outsourcing relationships generally work best with more formal, measurable decision

points. “Acceptance shouldn’t hinge on a subjective judgment such, as ‘I don’t think it does enough,’” he says. “Be sure to designate someone as an ‘acceptor’ responsible for making a timely decision, and provide a mechanism for bringing together both parties to discuss changes to the requirements and how the change will affect the schedule and the price.”

About SCR SOFT:

SCR SOFT helps clients to improve their business operations and IT effectiveness by delivering a broad range of business consulting and outsourcing services designed to achieve near-term and sustainable business benefit. Specifically, SCR SOFT focuses on highly synergistic service offerings, including: Application Development & Integration, Application Outsourcing, and Maintenance.

SCR SOFT believes that business and IT improvements are best realized by streamlining and optimizing business and IT processes, implementing rigorous management disciplines, and fostering a culture of accountability through meaningful performance metrics. SCR SOFT delivers its services through an integrated network, and via ISO 9001-certified software engineering processes. Information on SCR SOFT is available on the Internet at www.scrsoft.com.

www.scrsoft.com