

Real World Offshore Development Practices

By Bryan Campbell

Offshore development will only help your project save about 15 to 25 percent of its costs. Why? The most logical piece of a project that can be outsourced is the commodity-oriented, programming component. Unfortunately, programming represents only 30 to 50 percent of the total costs. The rest is consumed by project management, analysis and design, testing, and ancillary activities (e.g., management and environment preparation). Unless the project has well defined requirements and has been well designed (which is why many mainframe development projects are often outsourced), you will not see the 100 to 200 percent returns many offshore development organizations are touting. However, the opportunity to save up to 25 percent of your project costs can be quite significant, particularly these days.

If your system is a high novelty system, you will not be able to avoid the need for proper requirements gathering and analysis and design. This process needs direct access to users and business stakeholders, which requires an *onsite* presence. Before embarking on an offshore development project, you need to understand where your organization's capabilities reside. If requirements gathering and analysis and design are skills that are readily available and which have hardened processes, you are probably in a good position to capitalize on offshore development. If not, then spend the time necessary to develop processes and skills to support these important workflows. It is easy to lose the benefits of cheaper offshore resources by inefficient re-work if requirements are not adequately documented or communicated.

Communicate, Communicate, Communicate

If current projects are challenged by lack of communications, you might want to re-assess your capacity to support an offshore development project. Regularly scheduled conference calls, e-mails, and chat sessions are necessary for managing these projects. Not only are these sessions for sharing information and establishing a clear perspective on the status of the project, but also they are important for avoiding any of the miscommunication that can plague an offshore development project.

I would strongly encourage defining a *communication strategy* for your offshore development project. This should document a number of important communication techniques such as conference calls, chat tools, video conferencing, application sharing tools, e-mail, and a repository.

Conference Calls: It is important to have regularly scheduled conference calls with the whole project team. Calls should be carefully managed to ensure they start and end on time. Detailed issues should be assigned to individuals to be followed-up offline. Issues and tasks should be documented and entered into a tool, which can be accessed by the entire project team at any time. It might be necessary to have a variety of project conference calls for different teams based on the size of the project (some involving the business owner/customer and others with only the project participants).

Chat Tools: Tools such as MSN Messenger and Yahoo are indispensable to managing offshore projects. Everyone should have an assigned ID, which is communicated throughout the team. Project members should follow some form of etiquette, since not all issues can be resolved by instant messages.

Video Conferencing: If you have access to good video conferencing facilities I would recommend using them; however, these require high bandwidth and more sophisticated equipment than the \$100 Web cams that are becoming increasingly popular. Video conferencing – due to its cost and the challenges of setting it up – is probably best for formal status updates, which should only occur every week.

Application Sharing: Explore the application sharing tools to aid collaborative development with an offshore team. These tools allow teams to whiteboard different solutions and to work on a common

document together (such as a use case or architecture diagram). These tools are very useful in place of the brainstorming that onsite teams often use.

E-mail Usage: E-mail is an important mechanism for sharing information between onsite and offsite teams. It should not be used to replace other components of your communication strategy, particularly a document repository. All too often, e-mail is used as a mechanism to share artifacts; the risk is that these artifacts can be difficult to track and manage. It can also be difficult to ensure that everyone has the most current version of an artifact. For offshore development projects this becomes a critical risk. To address this, guidelines should be drafted for the project where documents are transmitted via e-mail only in unique circumstances. Otherwise, all documents should be placed in a repository and notifications of these deposits provided through e-mail with a link back to the repository. Using e-mail for short, focused forms of documented communication is the best way to use this technology for project communication purposes.

Document Repository: A commonly accessible document repository is an important tool for facilitating communication within an offshore project. A good document repository should be accessible 24/7 (remember your offshore team will need access to information primarily during the night). The purpose is to provide a common location for all project related material. It should support versioning of artifacts and provide a check-in/check-out feature. Keep the interface and any document properties fields as simple as possible. Repositories with rich interfaces can cause unnecessary delays in loading pages particularly from offsite locations. If there are any bottlenecks in network speed, also adding too much detail about the properties of a document (e.g., document class, sub-class, status, etc.) can be needlessly demanding.

Tool Selection

Tools support most communication needs; however, they also can be important in managing more discreet elements of an offshore project, such as requirements management, source code, and defect tracking. Selecting the appropriate tools for your offshore development project should align with your enterprise tool requirements. It is important to *validate* that there are no barriers for their use in your offshore environment. Tools need to have the ability to *synchronize* easily through either the Internet or a VPN. Agree on the use of tools *before* the project and *test* them to make sure they work as required. If you do not have any tools to manage your development effort, you might want to consider this before embarking on an offshore development project.

Culture

Regardless of where you decide to outsource your development efforts (India, Pakistan, China, or elsewhere), it is important to recognize that there may be cultural differences. Although this is a difficult area (fraught with all sorts of political correctness issues), discuss and document the following areas before the start of your project:

Holidays: Make a record of national or other holidays. These holidays can have a significant impact on a project, particularly since most holidays do not align well between countries. For example, November can have many holidays between the US (Thanksgiving) and India (Deepavalli) which can significantly reduce project productivity.

Infrastructure: Many areas where offshore development is conducted can be prone to infrastructure challenges, such as brownouts and entire city power failures. When selecting an offshore development partner, ask how they are supported for power outages (back-up generators, etc.) and whether they operate in a high-priority area for phone and electrical services.

Language: It can take some time to get familiar with the accents of individuals from different states or counties let alone different countries. Expect to see improvements throughout the lifecycle of the project. Also, remember that verbal communication can be more effective than written communication since writing skills in the software development industry are typically weak and are only exacerbated when you involve writing in a second language. Review correspondence before it is issued to customers or business stakeholders. This quality assurance function should be formally

embedded in your project in some capacity simply to avoid any confusion (or possibly hurt feelings) caused by a language barrier.

The CMM Conundrum

Increasingly, many offshore development organizations are touting their Capability Maturity Model (CCM) Level 5 certifications as a way to differentiate themselves in the offshore market. How are these certifications assessed and with what credibility? A number of offshore organizations with CCM Level 5 certifications do not have true standardized processes and fail to have consistent and repeatable software development results that should be a hallmark of a CMM Level 5 organization. In addition, there is a difference between an *organization's* CMM certification and a *project's* CMM level certification. Although your offshore development partner might be a CMM level 5 certified organization this does not necessarily translate to your individual project. Ask how your offshore development partner will support your project with its CMM processes; ask if any of their *projects* have been assessed using the CMM process.

Also, be aware that while a high CMM certification might give you greater comfort that you are dealing with a credible and quality oriented offshore development partner, this can come at the expense of flexibility and agility. This might be fine for a mission critical system but you might want to think about whether something so important is the right candidate for an offshore development project. The most important thing to focus on is how your offshore development partner adjusts to deviations from its CMM processes. How these differences are managed can influence how successful your project will be.

The Final Analysis

Offshore development *does* work and it *does* save you money. There are many things to consider, however, before embarking on your first offshore development project. Two recommendations: First, conduct an *internal readiness assessment* to see if your existing processes, infrastructure, and tools can support an offshore development initiative. Second, *try offshore development* with *small* and relatively *low risk*, internal application. These two steps can help you move through the learning curve with a lower degree of risk than betting the farm on a large engagement.

Bryan Campbell is Director of Consulting (South East Region) with [Valtech Inc.](#), a software development company specializing in object-oriented development using the Unified Process. Mr. Campbell has spent more than 15 years in the information technology industry across a wide range of disciplines from network operations to enterprise architecture to software development. Within software development, Mr. Campbell has considerable experience managing large scale, Web development projects applying iterative development techniques using a blend of the Unified Process and agile development. Other skills include strategic architecture assessments, enterprise project management, and IT outsourcing. Mr. Campbell can be reached at bryan.campbell@valtech.com.