P2E From Practice to Experience

PROJECT PRESS

Volume 1, Issue 4 | Meet Your Customers' Expectations On Time and Within Budget | October 1999

From Impasse to Win/Win—Dealing With Difficult Customer Relationships

Outsourcing Software Development

Increasingly, software systems are developed by groups of people brought together in a customer and vendor arrangement. For example, a mortgage company may outsource the design and customization of its new loan origination system to a software product and service company. A customer/vendor relationship is similar to a marriage. The success of these relationships depends on how much both parties depend on each other and the degree to which they have each other's interest in mind. And, as with any relationship, customers and vendors are aware that they have their own motives and goals.

Customer Meets Engineering Process

Best Insurance, a large insurance company, wants to hire Omni Systems, a custom software development company, to develop a new system for their salespeople. When representatives from Omni Systems, the vendor, and Best Insurance, the customer, meet for the first time, the vendor elicits information from the customer in order to understand the customer's requirements and expectations.

INSIDE THIS ISSUE...

This exchange of information leads Omni Systems to prepare a proposal that outlines the technical solution and a cost estimate. Omni Systems and Best Insurance agree to an initial six-week contract period, during which Omni Systems will define the software requirements and prepare a detailed project plan. During this period, Omni Systems' engineers are glued to the customer's words, mining every sentence for relevant information, facts, and constraints that allow them to analyze and specify the software requirements.

The engineers discover that Best Insurance has a big vision to leapfrog their competition with a revolutionary quoting system. Unfortunately, people at Best Insurance are busy fighting fires with their existing system and have little time to create detailed requirements for the new system. Engineers at Omni Systems complete the requirements document as best as they can, and also develop a prototype based on the requirements in the hope that it will help the customer provide input and feedback. After demonstrating the prototype, the engineers at Omni Systems receive feedback that the prototype addresses a part of the desired new software system, but the initial rollout needs to include a hand-held device using a mobile communications facility. Omni Systems prepares a detailed schedule including the requested changes.

When the project manager at Best Insurance receives the project plan, he takes a fine-toothed comb to it, questioning the milestones and deliverables. The project manager tells Omni Systems that he is not ready to proceed. At this point, the engineers at Omni Systems are frustrated, as they have not been able to get their hands around the requirements for the system as envisioned by the customer.

Differences, Differences, Differences

Reaching a common understanding of the requirements for the software system is only one of the difficulties that exists in a customer/vendor relationship. Both parties have to bridge many other differences during their journey of working together.

Differences in Language

The customer is comfortable conversing in the language of its business domain, such as telecommunications, healthcare or insurance.

The software vendor is trained in the language of software engineering. Words have specific meaning in each domain. Even words such as “leadership” and “process” may be interpreted differently.

Differences in Goals

The customer may see the vendor not only as a contractor hired to produce a particular software system, but also as a consultant available for research and advice regarding software-related questions. The vendor’s goal is to develop and deliver a software system that will improve the customer’s business operations, and in turn will provide the vendor with revenue and new references for its portfolio.

Differences in Expectations

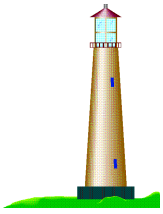
Customers most likely focus on the benefits the software system provides and how it helps them meet business objectives. From the customer’s perspective, the system needs to be available in its entirety as soon as possible. Instead, the software vendor sees the development of the system in successive deployments of well-specified functionality.

Differences in Decision Making

The customer may purchase software components to integrate into the new system based on favorable financial arrangements. The software development company will most likely focus on the technical feasibility of integrating different software components.

The following are more examples of differences between customers and vendors:

- Differences in expertise
- Differences in values
- Differences in solving problems
- Differences in interaction styles
- Differences in priorities
- Differences in communication styles



The Win/Win Objective

The customer or vendor or both may realize after several months into the engagement that many of the differences cannot be reconciled. An impasse is reached.

Sometimes it is difficult to see these differences before the business relationship is established. It is in the best interests of both parties to resolve the impasse through a win/win solution.

Creating a win/win situation means improving the business relationship by changing:

- the scope,
- the commitment, and
- the type of work.

These changes are implemented while honoring the differences between the two parties. For example, if the customer is looking for permanent staff, rather than project-related temporary contractors, the software vendor can replace the staff and negotiate a support agreement. If the customer is primarily looking for consulting, the contract can be switched from fixed time and cost to a retainer based on time and materials.

Bridging the Differences

In order to achieve a win/win outcome, both parties need to work together as collaborators rather than critics. Collaboration needs to exist on three levels:

- The business level
- The process level
- The personal level

The Business Level

As Rajiv Sabherwal points out in his article, “The Role of Trust in Outsourced IS Development Projects” (Communications of the ACM, Vol. 42, No. 2), a contract is necessary but not a sufficient condition for a successful outsourced project. The contract should be tailored to the customer’s needs with respect to service or product development. **Does the work involve research and recommendation or does it involve software development?** The contract needs to be adjusted according to the expected time frame. **Is this a short-term engagement (less than 3 months) or a long-term engagement?** The contract should match the role the vendor is expected to play. **Is the vendor: (1) the exclusive contractor, (2) the general contractor fulfilling the role of the integrator, or (3) a sub-contractor?**

The contract spells out the structural controls that establish an element of trust between the customer and vendor. Examples are performance penalties, acceptance criteria, monitoring and reporting requirements, payment in equity, fixed bid, or time and materials. In order to achieve a win/win outcome, a psychological contract needs to be established through trust. Trust can develop through:

- Getting to know each other personally,
- Sharing goals, or
- Establishing an early success.

The Process Level

To coordinate their work, vendor and customer need to agree on a process of how to work together as well as their roles and responsibilities. If the contract involves the development of a software system, the process most likely is project-oriented. For the project process to succeed, roles and responsibilities are expected from each other.

The customer is expected to contribute the following:

- Domain expertise
- Decisions based on evaluating technical solutions against business objectives
- Acceptance of work products
- Information about the customer's existing business processes

In case the customer is not available for providing domain expertise or acceptance of the work products, surrogates can be found to take the place of the customer. Information about the existing business processes requires the customer's involvement. The customer cannot delegate decisions.

The vendor is expected to contribute the following:

- Technology expertise
- Expertise in the best practices for developing software systems
- Deployment and technology transfer
- Quality control
- Review of intermediate work products with the customer

Only the deployment and technology transfer can be delegated to another party.

The Personal Level

In reality, the vendor organization does not work with the customer organization, but individuals from both organizations work together. Just like in any other working relationship, people have to come to know each other in order to trust in each other's abilities.

Frequent contact and open communication is important. It is important that the primary contact in the vendor organization and the contact in the customer organization work well together. If it turns out that their personalities are not compatible, they can learn to overcome incompatibilities by understanding the objectives of each other's roles, listening and operating on facts rather than assumptions, and establishing common ground. Alternatively, the contact roles can be re-assigned to people that have similar personality profiles.

Conclusion

Once engaged in a contract, the customer and vendor have agreed to establish a business relationship and bring the contract to closure. Unilateral termination is not a solution. The solution lies in redefining the relationship, putting a contract in place that defines the appropriate scope and commitment, and establishing trust.

Tracking Status Using Microsoft Project

Microsoft Project provides two ways for entering the project status:

- Based on **actual task** data
- Based on **timesheet** data

Entering Actual Task Data

To keep the amount of effort invested in updating the schedules at an acceptable level, do the following:

Select the reporting period and set the status date to the date following the last date for which actuals have been recorded in the current reporting period. For example, if the reporting period goes through Friday, July 10, then set the status date to Saturday, July 11.

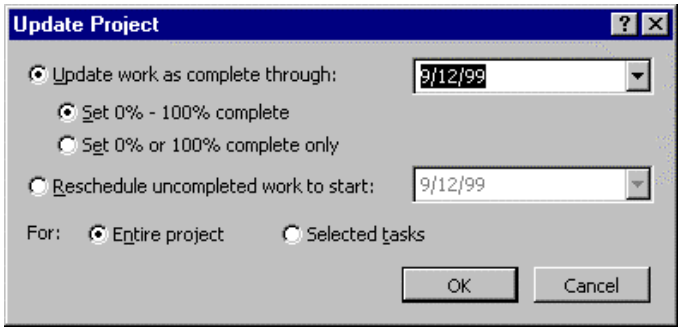
Solicit input from people who do the work with respect to start date, end date and duration or any revision thereof.

First Pass—For all tasks in the reporting period, make the appropriate changes with respect to the start date. Enter the actual start date in the **Start** field and the actual duration or new estimate for the duration in the **Duration** field of the **Entry** table.

Second Pass—Do one of the following:

Select the tasks in the task list for which you want to record actuals, and then select **Tracking** in the **Tools** menu and **Update Tasks**. Enter 100 for %Complete.

Select **Tools** and then **Tracking** and **Update Project**. Set the date for "Update work as complete through" to the day following the end date of the current reporting period. Select whether tasks in the reporting period that have been started but are not yet complete should be recorded as percent complete or not (partial credit).

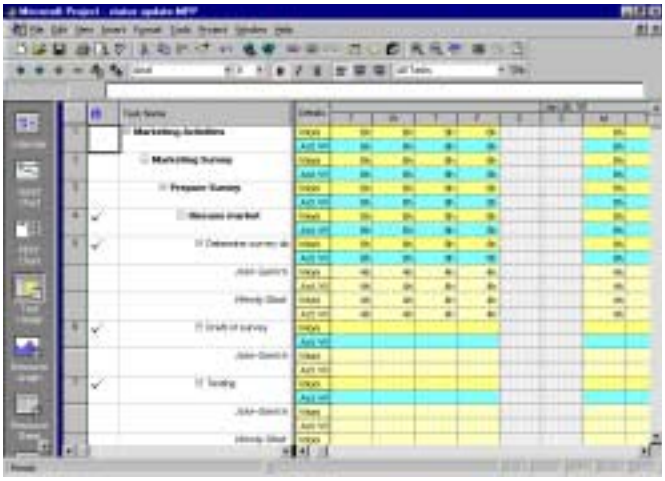


Note: Microsoft Project displays the current date in the dialog box for Update Project. The value entered in the dialog box for "Update work as complete through" will be stored in the **Status** field of the **Project Information** dialog box.

Entering Actual Timesheet Information

When using timesheets to keep track of work, the task usage and resource usage views are the best views to capture this information. Select the task usage icon from the view bar. Right-click anywhere in the time scale portion and choose **Actual Work** from the menu.

Enter the timesheet information in the field for actual work.



The actual hours entered for each resource are totaled in the row for the actual work of the task.

P2E Calendar of Activities

- **Monday, October 18 and Tuesday, October 19, 1999**—Introduction to Microsoft Project98, 2 days, Registration: (303) 492-8668, Location: CU Computing Center, Cost: $190.
- **Saturday, October 23, 1999**—Using the Unified Modeling Language for Object-Oriented Analysis and Design, Professional Development Seminar, Boulder Chapter of the ACM. Cost: $90. For more information, call 303-938-9400.
- **Tuesday, November 16, 1999**—Microsoft Project98 Advanced course, 1 day, Registration: (303) 492-8668, Location: CU Computing Center, Cost: $150.

Ulla Merz
P2E
5501 Aztec Court
Boulder, CO 80303

ADDRESS CORRECTION REQUESTED

INSIDE...
FIND OUT HOW P2E
HELPS SOFTWARE COMPANIES
MANAGE THEIR PROJECTS!