



Real World Project Management: Procurement Management

By Joseph Phillips

Projects typically need stuff: servers, software, subject matter experts, pizza, etc. And to buy all this stuff, you need to go through procurement processes. That's just a fancy way of saying you need to follow some rules and procedures within your organisation to get the things you need to complete your project. Well, duh.



In some organisations where I've consulted, the project managers can spend carte blanche up to \$10,000 on any purchases they need. In other, less fun organisations, the project managers can't buy a soda pop without an accountant's permission.

So where are you? Do you get to buy, buy, buy, or is every purchase considered, weighed, and meditated on before someone reaches for the wallet? In either shop, there are some guidelines you should consider. Really, there are.

Planning What to Buy

All purchases require some level of planning. The intensity of the planning is relevant to the purchase being made. You do this already, right? If you're about to install a new piece of hardware, you'll consider all the functions that the hardware should have, shop around a bit for prices, and then see how much your project or organisation can spend (or is willing to spend).

Planning for procurement includes more than window shopping. Think way, way back to the start of any project that required procurement. Early in the project planning, it was easy to identify those things or services that you needed to buy for the project to succeed. As the project moved forward, "emergencies" popped up, requiring you to buy more stuff: cables, software, additional hardware, tools, training, spaghetti sauce, whatever. So how did you go about getting all this stuff? Did you go to management, hat in hand, and plead your case for your much-needed spaghetti sauce, or did you dip into a project contingency fund?

How you go about purchasing depends on the structure within your organisation. It's difficult, if not impossible, to define a universal approach to procurement. Everybody, every organisation, has a specific approach to procurement. The moral of the story? Follow the rules. Once you know the rules of how to procure, then you can play the game.

Gettin' to the Gettin'

I know lots of people who like to go shopping. One person (who shall remain nameless, but her initials are Lisa) plans her vacations based on the shopping malls in the vicinity of her hotel. She buys an extra seat for the flight home, just to carry all of her new shoes and fancy outfits.

As a project manager, you can't go project shopping just because shoes are on sale. While sales are good, they don't always help the project to acquire the things it needs to reach project closure.

There's nothing better than finding a sale on the hardware or software that your project needs, but you and I know that's just not the way technology and procurement usually works. We have to shop, compare, evaluate, and eventually cough up the cash to get what our projects need. But here's some Econ 101 for you: Prices are affected by supply and demand, pending changes, and other factors, from government regulations to the economy as a whole.

I can hear you again: Duh.

But hold that "duh" for one moment. Three specific conditions affect how much you pay for the things your project needs:

1. **Sole source.** In this condition, you'll likely pay big bucks. Sole source means there is only one qualified seller in the market. This is supply and demand at its finest. If your project needs a Cisco CCIE-certified consultant who must also know how to program in COBOL, speak Spanish, and cook spaghetti for up to forty people, and must live local to your firm, those are some high requirements, you'll likely have to pay a higher dollar for this expert than for your average spaghetti-cooking hack.
2. **Single source.** You're in love. When there's a single source provider, your organisation prefers to work with this provider even though other providers may be less costly or more qualified. The danger is that your single source provider may know your attachment and take advantage of the situation. Or get lax in their commitment to quality. Or go out of business. (Or not.)
3. **Oligopoly.** This one is just fun to say. Try it: Oh-lig-AH-polly (sounds like monopoly). This market condition means that there are so few providers of the particular good or service that the events, actions, or circumstances of one seller affect the events, actions, or circumstances of the other sellers. Examples: Airline fares; oil prices; hardware costs; or possibly availability of spaghetti-cooking, COBOL-programming CCIEs who live in your neighborhood.

Cash and the Law of Diminishing Returns

One of my favorite economic laws is the Law of Diminishing Returns. It's basic stuff at first glance, but can really haunt a project manager if he's not careful. I know you're familiar with the Law of Diminishing Returns, but that guy from Sheboygan is also reading, so let me help him out.

Imagine that you have a wheat field. Wait, he's from Sheboygan. Imagine that you have a corn field and you know that you can get 100 trucks of corn out of the field. That's the most corn you'll ever get from the field. You also know that if you hire 10 guys to harvest the corn for you, they'll be done in 2 days. So you reason that if you hire 20 guys you'll be done in 1 day. So this must mean that if you hire 40 workers, all the corn will be harvested in half a day, right? Maybe, but if you continue to

add workers to the field, a few things will happen:

- Your yield-100 trucks of corn-remains the same no matter how quickly you harvest the corn.
- Your profit will decline because you have to pay all those workers to harvest the corn for you. At some point, you may even be upside down on profitability because of the expense of labour.
- The workers will become counterproductive because they'll start getting into each other's way.

So how does all this corn relate to an IT project?

The obvious answer is that you can't exponentially add labour to get a project done faster. And just because you add labour doesn't mean that the project will get done faster. (Have you ever had two programmers, two systems engineers, or even two Spanish-speaking, spaghetti-cooking, COBOL-programming CCIEs argue over how a task should be completed? The argument could go on for years before the work actually gets started.)

But the Law of Diminishing Returns also applies to the technology that you purchase. Have you ever bought an application that was so rich with features that the cost and time of learning the application was more than the returns from using the application? Or have you ever installed a massive powerhouse print server where many of the features of the OS go ignored? Or maxed out the RAM on a laptop that's only used for PowerPoint presentations and solitaire? When it comes to IT hardware, as with labour, project managers should procure only what's needed-not the maximum that the budget will allow.

Build or Buy?

Ah, one of the great arguments of all time. Should we buy it or should we build it? Well, it's probably not that great of an argument, but I'll bet you've been in some heated discussions on the value of either side of the debate. If not, let's start one now.

Sometimes, like it or not, it's more cost-effective to spend the cash and pay someone else to build the thing for you. Why? Your crew is busy doing other jobs, they don't have the competence to build the thing you need, or your organisation doesn't want to take the risk of creating the thing in-house. Lots of reasons.

Other times, like when your project team is lounging by the company pool sipping pinot noir and snacking on spaghetti, it's ideal to put them back to work building something. Again, there are lots of reasons why it may be better to build versus buy, or the other way around.

But sometimes it's purely a price decision. Here's the deal: Let's say that if your organisation builds a piece of software, it'll cost \$45,000 to create and then \$4,500 each month to support. Now, a vendor says they'll only charge you \$23,000 to build the initial product, but they'll need \$6,500 each month to support it as part of the deal.

Hmmm... so what's a project manager to do?

Maths.

Here's how it works: Take your build option of \$45,000 and subtract the vendor's quote of \$23,000. The difference is \$22,000. Now take the monthly support fees of the vendor, \$6,500, and subtract

your lesser in-house fees of \$4,500. The difference is \$2,000.

Now, drum roll please, divide the initial out-of-pocket expense difference of \$22,000 by the monthly support fees difference of \$2,000 and you'll get 11. Eleven what?

Well, 11 in Blackjack means double down. Here it means that you can pay for the out-of-pocket expenses of creating the software in-house in 11 months. So, Copernicus, if your software creation will exist for less than 11 months, hire the vendor to do the work for you. If your solution will be around longer than 11 months, and price is the only factor, build the software yourself.

Taking Out a Contract

Contracts override everything: promises, email, secret handshakes. As long as they don't include illegal activities, contracts are backed by the U.S. legal system. A contract is what makes the deal a deal.

To get to the contracting activities, you need to create the procurement documents. The initial document is usually the statement of work (SOW), which describes the thing or service you want to buy. The SOW is provided to the vendor with an invitation to bid (IFB), which you probably also know as a request for quote (RFQ). The IFB and the RFQ are basically the same thing and are focused just on price, not ideas.

A request for proposal wants a price, but also suggestions and ideas on how the project work should be done. Proposals are more than just costs, they're a bit of consulting from the vendor.

In big-dollar contracts, you'll likely host a bidders' conference in which all vendors that want to create a proposal or bid will meet with you at once and ask questions concerning the SOW. This setup ensures that all the vendors have the same information on which to base prices and proposals.

Once you make a decision on which vendor to use, you go through negotiations, which lead to a contract. The contract-type selection might also be negotiated, but usually the type of work or goods being procured dictate the appropriate contract type. The following table lists the most common contract types and their attributes.

Contract Type	Description	Risk
Fixed fee	Fixed fee for the goods or services provided. Nice and easy.	The vendor has the risk of cost overruns.
Time and materials	You pay for the time and materials to complete the work.	Low risk as long as the contract includes a "not to exceed" (NTE) clause as a price cap.
Unit price	Fee per item or hour purchased. Sometimes these plans include incentives such as "Buy more items and the cost per item drops."	Low risk.

Cost plus (anything)	Cost of the goods and services plus a fee or percentage of the cost. Who uses these anymore? Not too many folks.	High risk for the buyer, as waste means you get to buy more and pay more.
Incentive fee contracts	These contracts award a bonus if the project is done early and can include penalties if the work is late or lacking in quality.	Usually low risk.

Bring Your Wallet

Procurement is, uh, big business. There are lots of details in procurement planning, the creation of procurement documents, bidder conferences, and in the contracting of the work. All organisations have their approach to procurement and contracting. You, the project manager, need to understand your organisation's approach and then follow the rules to get the stuff you need.

Now, if you'll excuse me, my Spanish-speaking, CCIE-certified, COBOL programmer has spaghetti ready. And I need to pay his invoice.

Joseph Phillips is the author of five books on project management and is a, PMI Project Management Professional, a CompTIA certified Project Professional, and a Certified Technical Trainer. Joseph has taught for the Project Management Institute, the US Military, Fortune 50 companies, and has spoken at international conferences on project management. Please visit Joseph's website at projectseminars.com. 

© Project Smart 2000-2010. All rights reserved.