Introduction

It was a fine spring day in Denver. The snow was melting from a late-season snowstorm, not unusual for Colorado this time of year.

The CIO was busy dealing with a networking problem. The firewall system was not letting all the traffic through for the new web services interface for the supply chain software that had just been installed.

He was interrupted by a call on his cell phone. "Jack, we have a problem." What else is new? Jack thought and almost said it.

"It's the new data warehousing project," Mary said. "It's behind schedule again."

"So what else is new?" This time he did say it.

Mary said, "Yes, I know, but we really need to do something about it. These consultants we've hired are costing us an arm and a leg, and progress has been so slow."

That's what we get for paying them by the hour, thought Jack. As the saying went, use a small brush if you are paid by the hour to paint.

Mary continued, "They are working as hard as they can, and they are even spending some extra hours they are not billing us for."

"Then what do you suggest?" Jack asked.

"It's the way the project is being managed," Mary said. "We thought having all the programmers here would reduce the need for specifications. But at the rate we are going, we are going to blow our budget for this project in the first three months."

Jack could see it coming. The pressure from the CEO to outsource this project would be inexorable now. "We are going to have to find a way to make offshore outsourcing work for this project."

"I know," Mary said reluctantly.

It was a rainy Wednesday in Silicon Valley. The young entrepreneur was getting into her car to drive up to Sand Hill Road for an appointment with a venture capitalist. Kelly had made the connection through a friend who was working at another startup in which this VC had invested.

Even though this was not her first investor presentation, she was a bit nervous. "I won't need a Peet's coffee this morning," she thought. That would make her too jumpy.

She had just the right amount of information in her PowerPoint presentation covering sales, marketing, and software development. She was looking to raise $5 million in this first round.

"It's nice to know you think you can build this into a $100 million company in four years," said the VC, "but what we really want to know is how you are going to get customers to pay your first million in sales."

Another VC in the room asked, "Do you really need $5 million? You know this isn't 1999 anymore. How will you spend the money we give you?"

Kelly jumped ahead to the PowerPoint slide showing the budget. So much for her carefully planned presentation!

"Those engineering costs seem rather high. Aren't you outsourcing your software development?"

"We feel it is too risky," said Kelly.

"You'd better reconsider or at least find some way to reduce those product development costs," said the other VC.

Afterwards, Harry, the VP of engineering, asked Kelly, "How did it go?"

"I am not going back into another VC meeting until you tell me how we are going to outsource the product development."

"I thought we went over this already. It is too risky to outsource."

"Well, they say it is too expensive *not* to do it. And they are not going to fund us if they think we are throwing money away on

engineers here in Mountain View. So you are going to have to find a way to make it work."

"What about programming jobs leaving the U.S.? Don't they care about that?"

"No, of course not. And frankly, I am losing interest in that argument as well. If we don't find a way to make outsourcing work, we won't get funded and none of us are going to have jobs. And I don't mean just you and me—I mean the dozens of marketing, sales, and support jobs we are creating. They will all go away. Unless you can make outsourcing work, and make it work quickly."

Vijay had just hung up the phone. It was midnight in his condo in New Jersey, and he had just finished a conference call with his brother Sandeep back in Bangalore, India. It was 9:30 in the morning of the next day in India. Sandeep was having trouble with the team. Another engineer had quit.

Or more accurately, he never came to work to begin with. It is common for an engineer to accept multiple job offers and just go to the one he likes the most without letting the others know. It would be impolite to turn down a job offer in India.

But Vijay knew that Sandeep would handle it. He always did. After all, he had built up their small team of six engineers and created the software product in just eight months. They had funded it with their combined savings of just under $50,000.

Now Vijay was about to close a big deal with a client he had first met almost a year ago. Seeing the problems they had managing inventory gave him the idea for the software and caused him to start the company to begin with. The risk of using his savings to develop the software was finally about to pay off.

Thank God for Sandeep and the engineers in India! Vijay thought. There was no way they could have afforded to develop this software in New Jersey. All that uproar about jobs moving offshore, he thought. Vijay shook his head. "Some people just don't know an opportunity when they see one." He said it out loud to no one in particular.

From San Jose to Boston in the United States and in many other countries as well, these scenes are played out every day. Many people know about the promise of outsourcing—lower costs and accelerated completion of programming projects and software products. And yet deciding to outsource is not an easy decision for many companies. Is outsourcing only for the foreign-born entrepreneurs and executives?

In his book *Crossing the Chasm*, Geoffrey Moore introduces the concept of innovators and early adopters, who are the first to become involved with a new trend or product. When it comes to outsourcing, it seems that we have crossed the chasm from this first stage to encourage an early majority of people and companies that now find value in outsourcing. They believe that outsourcing of software development has been tried and tested by the early adopters and now is safe for them to try.

Or is it? Studies have shown[1] that more than half of outsourced work fails to meet financial goals. So have all the problems, risks, and dangers really been eliminated? Is it safe for you to outsource your software development? Given the pressure to outsource, how can you decide whether outsourcing is even appropriate for your company?

This book will help you decide. I have faced these questions in deciding to outsource my own software development. In some cases I was told I had to outsource; in others I made the decision myself. Together we will explore the issues of outsourcing—when to do it, where to do it, and how to do it. Even when *not* to outsource.

In this book, you will discover practical solutions to the problems that arise when outsourcing. The overall goal is to make your outsourcing as risk-free as possible. Each chapter presents an outsourcing issue and shows how you can navigate past the shoals of other people's mistakes to arrive at the end goal of delivering your software on time, within scope, and on budget. And that budget will be much, much lower when you use offshore outsourcing.

Here is what you'll find in each chapter:

Chapter 1: Deciding to Outsource. What factors should you consider in making your decision? When are the risks of outsourcing too great to bear? If you do decide to outsource, how should you do

1. For example, see *Calling a Change in the Outsourcing Market: The Realities for the World's Largest Organizations* published by Deloitte in April 2005. A link to this in-depth study is on the book's web site: www.SoftwareWithoutBordersBook.com.

it? Should you contract with an offshore vendor or one in your home country? Or should you use more than one vendor?

Chapter 2: Where to Outsource. More than 80 percent of outsourcing is done in India. But is that the best place for you? Outsourcing to China is growing rapidly. Is that the right destination for your outsourcing? What about Eastern Europe or Latin America?

Chapter 3: How to Select Your Outsourcing Vendor. What criteria should you take into consideration when selecting your outsourcing vendor? Just having a personal referral to a vendor from a friend or relative is usually not enough. The vendor you choose should meet the business, technical, and time zone criteria that work well for your organization.

Chapter 4: Offshoring, or Creating Your Own Offshore Subsidiary. You can save even more money when you create an offshore subsidiary, because the salary you pay your offshore engineers is lower than the outsourcing rate. But what about the setup costs and management overhead? See if it would make sense for your company to use a partner to build, operate, and then transfer (BOT) your offshore engineering team into your own subsidiary later.

Chapter 5: Describing Your Software for Outsourcing. It is critical for you to have a specification for your software that explains what it should do. Yet you don't need to spend months creating a huge document that rivals the size of *War and Peace* in order to direct your outsourcing effectively. This chapter describes how big your specification should be and what it should contain.

Chapter 6: Controlling Your Outsourced Software Development. Will your internal engineers be working from the same code base as your outsourced team? How will you handle integration if they are working separately? What are agile software development methods, and when should you be using them, whether you are outsourcing or not?

How will you be able to keep in touch with your outsourcing vendor? What tools, in addition to email and instant messaging, can you use to collaborate and to control their work?

Chapter 7: Software Outsourcing Metrics. How will you measure the performance of your outsourced team? What levels of productivity should you expect? Should you measure your results differently for new development than for maintenance programming?

Chapter 8: Protecting Your Intellectual Property. Are there countries you should avoid? What needs to be in your contract to give

you the legal protection you need? Besides a contract, what else do you need to protect your intellectual property?

Chapter 9: Outsourcing Your Quality Assurance. Maybe you should start outsourcing with this often neglected part of software development. Do you have a robust QA process in place? Will affordable testing and QA increase the value of your software and company?

Chapter 10: Five Situations Right for Outsourcing. Some companies outsource all software development, and others are more picky. What should you look for in a software project to make it a prime candidate for outsourcing?

Chapter 11: The Future of Global Software Development. Is outsourcing a fad, a flash in the pan? Is globalization good or bad? What can you do to protect your career and thrive if outsourcing continues to grow in popularity?

Appendix A: The Outsourcing Strategy Decision Matrix. Presents the details of a decision matrix described in Chapter 1. You can use the matrix to decide which of five outsourcing strategies is the best for you, if any, or if in-house software development is your best choice. Download the matrix from the book's web site to automatically compute your scores.

Appendix B: The Outsourcing Readiness Test. Twenty questions you can use to determine your readiness for offshore outsourcing your software development.

Appendix C: Avoiding the Seven Deadly Dangers of Outsourcing. Mastering global outsourcing and distributed software development is critical to the success of the software industry. This chapter summarizes the stumbling blocks others have encountered when offshoring their software development and recommends techniques that can help your company avoid months of frustration and could save you hundreds of thousands of dollars.

This book gives you an overview of what it takes to outsource successfully and then delves more deeply to give you the specific tools and techniques required. These include spreadsheets and documents you can use directly as well as pointers to other books and reports to give you the background, knowledge, and confidence you need to get you as close to risk-free outsourcing as possible.

Specific real-world examples are used throughout the book to show you how outsourcing is done successfully. Some examples show the pitfalls others have fallen into, and how you can avoid them.

You are welcome to visit the book's web site for updates and corrections as well as links to many useful tools.

Here is the URL:

www.SoftwareWithoutBordersBook.com

I welcome your thoughts on the book and on outsourcing of software in general. You can send me your comments via the book's web site or directly by email to:

Steve@SoftwareWithoutBordersBook.com