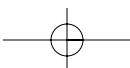
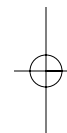
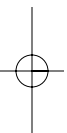
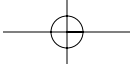


Part 1

The first two chapters set the stage for the rest of this book. The first chapter introduces the people, process, and product of the Delphi project. Delphi is the subject of most of this book, so we urge you to learn the terms we will use throughout. The second chapter discusses the project room. This is a public place in which to put all the information related to the project. We recommend that all projects have and use a project room.



Chapter 1

*It Sounded Good
When We Started*

In November of 1998, I* boarded a United Airlines flight to Los Angeles from Dulles Airport near Washington, DC. I would be spending my 40th birthday listening to a contractor describe the achievements, or lack thereof, made during the previous month on a project. My managers had asked me to go on this trip because I was the software specialist in our office and they anticipated that the contractor would soon be entering the phase for upgrading the software on a system. They were wrong, as this upgrade would not occur for several years. Nevertheless, my managers would soon volunteer me to be the lead on the project, and I would be taking these monthly and sometimes twice-monthly flights to the West Coast to monitor progress.

Accompanying me on this and many of the trips was Roy O'Bryan. He had retired from government service and was working for a company that provided technical assistance on this and other projects. He was the "experienced" one. He had worked four years on the project that preceded this one and had been working on this project since its start-up, almost a year before I arrived on the scene. What neither of us knew or anticipated at this point, was that this project was neither going to be like its predecessor, nor like any that we had experienced previously.

This project, which we shall call "Delphi," was not going well. A big problem was that people in our East Coast office thought the project was on firm ground. There had been a few problems in the first few months, but "the contractor had taken the appropriate action, fixed the problems, and all would be well from here on." That would not be the case. We experienced

*Throughout this book, "we" refers to both authors; "I" refers to Dwayne Phillips.

months of discovering problems, denying them, concentrating on short-term fixes, and punishing innocent bystanders before we faced the facts. We had been suckered into dispensing Band-Aids to fix systemic problems and had failed miserably when it came to endemic diagnosis. Initially, we spent far too much time on the dance floor before we decided to go up to the balcony to really see what was happening. The efforts of a few smart and courageous people brought Delphi around. To some of the bystanders, the project was a failure, but to users—the real stakeholders—it was successful and remains so today.

Delphi was not supposed to have all these problems, denial, and tribulations. It was to be a great project that went smoothly and produced an excellent product on time and on budget. In other words, it sounded good when we started.

There is an old adage about teaching old dogs new tricks. Roy and I had more years than we wish to admit running projects. We thought that we were pretty savvy when it came to the various aspects of project management, but during the three-plus years of working on Delphi we experienced much and learned even more. There was a lot of grief, but somewhere in midst of all our pain there was irony and some humor. Roy, being somewhat perceptive, recorded this wit as one-liners on 3 × 5 cards. Periodically, we would shuffle through the cards, muse about the one-liners as prospective chapters titles, and facetiously opined that the Delphi experience—what we had learned and had experienced—could fill the pages of a book. At some point in the spring of 2001, the thoughts could no longer be contained and I started writing. The one-liners became topics, and the topics became the titles of chapters. As we wrote, other chapter titles came to mind.

This chapter sets the stage for the rest of the book. We define terms here so that we do not have to repeat them throughout the rest of the chapters.

THE PRODUCT

The product of Delphi was a system with both hardware and software. The hardware portion was mostly analog circuitry. Our desire to make the system as small and light as possible drove us to use technologies that were difficult to handle physically. Assembly workers did much of their work using microscopes. They touched the parts with machines instead of their hands. These technologies are common when people need to build small, high-performance analog systems.

The software portion was split between an embedded processor in the system and its controller. About half of the software ran on an embedded processor in the system. This controlled the flow of signals through the system. Most of the system comprised analog circuitry, with the control portions being digital. We admired the people who worked on the software, as it was complex and difficult to write and maintain. It was conceived when object-

oriented design was relatively new, DOS was the standard operating system, and a 30 MHz embedded processor was really fast. By the time Delphi was delivered, object-oriented design was an accepted development methodology, Pentium processors were in their third generation, and Windows dominated the operating environments.

The system's controller was a laptop computer. This software was also difficult to maintain, as we were using laptop computers from the early 1990s (later chapters explain why). The programmers had to squeeze the software into limited memory and make it perform efficiently to meet the timing requirements of real-time control.

Delphi produced nine systems. Because tiny analog circuits were a major part of the project, each system had challenges of its own. We never reached the end of a learning curve where the engineers could turn their backs while assemblers built and tested the systems.

THE PEOPLE

There were close to 100 people involved with Delphi. These people were divided into three major groups. First were the users. These people needed the systems to do their jobs. They stated the requirements of the product during the early 1990s. The users took several trips a year with us to the West Coast to monitor progress. They also tested the systems by using them in their environment.

The second group was the builders. These people worked for the contractor who built the systems. They toiled long hours and weekends to deliver systems that the users needed. There were several groups of people working for the builder. One group was the engineers who designed, integrated, and tested the system. Another group was the programmers who wrote, tested, and maintained the software. A less appreciated but equally important group was the assembly workers. These people sat in front of microscopes and machines and assembled the tiny components into a system.

We worked in the group we will call "the buyer." Our job was to take the user's requirements to the builder and monitor the builder's progress. We managed the contract between our organization and the builder. If we had done Delphi in the 1960s, we would have built the product ourselves, but in the 1990s, like today, we worked with someone else to build the product.

These three groups—the users, builders, and buyers—had managers. The managers of the users and buyers funded the project. The builder's managers represented their company to our managers. It would be easy to blame all the problems we experienced with Delphi on the managers. That, however, would not be the truth. The managers on the user and buyer sides listened to our concerns and gathered the funds we needed to complete the project. The managers on the builder side put the necessary people on the project and did not quit when business logic showed otherwise.

We had a fine collection of individuals working on Delphi. There were some days when some people lashed out at others from fatigue and frustration. Some people quit the project and moved elsewhere. Despite the troubles and trials of Delphi, most of the people stayed with it for most of the project. The most satisfying part of this project was watching people come to work everyday and push against a mass of thousands of interlaced problems until that mass went away.

Throughout this book are the names of many people. We changed the names to protect the privacy of the people who worked on Delphi.

THE PROCESS

Delphi was a government project. I was an employee of the U.S. Federal government (still am), as were the users. We encourage readers to keep reading even if they don't work for or with the government. The challenges and lessons of Delphi apply to almost any human endeavor.

Delphi was part of a larger program involving the users, buyer, and builder. This program started in the early 1990s. The builder won a competitive contract from the buyer. They ran a four-year, \$40M project in which they delivered two prototype systems. I was not involved with that project, but Roy was. Those systems were so expensive because the builder had to invent much of the hardware and develop all the software algorithms. The users took the systems into their environment and used them. Their experiences produced a new set of requirements.

Shortly after finishing this first contract, the buyer awarded the builder the Delphi contract to build more systems. The builder would not have to invent things as they had to on the first project. Instead, "all they would have to do" was produce nine more systems and include a few enhancements so these systems would meet all the user's requirements. At least that is all that most people felt the builder would have to do. Delphi proved otherwise.

OTHER THINGS

Delphi involved many other things that made the project both possible and difficult. We had a small mountain of documents, including requirements, high-level designs, low-level designs, plans, drawings, specifications, etc. We had hundreds of meetings both formal and informal. The buyer and builder lived and worked on opposite coasts. This meant plenty of plane rides, daily phone calls, and a handful of faxes each week.

These items—documents, meetings, phone calls, and faxes—meant countless attempts from different groups of people to communicate with one another. This meant countless opportunities for miscommunication. Many of the problems and lessons in this book are about this subject of human com-

munication. If the reader is to learn one lesson from this book, it is that good, hard working, conscientious people frequently misunderstand one another. Such misunderstandings should not surprise or anger anyone. They are normal, and we need to work our way through them.

CONCLUSION

Delphi was one of the most challenging and rewarding projects we have worked on in our careers. We spent three and a half years on it. Several of us had children go from elementary school to high school during this time. Delphi ended by delivering what the users needed. It cost more than people wanted to spend and the wait was longer than people wanted to endure, but the people persevered. In the end, our reward was that we were able to work with people who had the knowledge, skill, and devotion to deliver what many others felt they could not. This is what makes projects worth the years of our lives—the people.