Draft Preface to the First Drafts

Give me a fish and you feed me for a day.
Teach me to fish and you feed me for a lifetime.

This is a book about fish. But since you'll eventually starve it also talks about fishing. The fish are all those pieces of information that you need as a developer to implement OLE 2.0 features into your application. Teaching you to fish means describing why the specific pieces you are using were designed and what path they lay out towards the future. Of course, you always need a reason to keep fishing even if you're well fed, so I will attempt to provide the necessary goals and motivation at the beginning of each chapter.

My first reason for writing this book is that the product called OLE 2.0 is much more than what has been termed "Compound Documents." My first experience with OLE 2.0 was to write two small applications: a container application that held objects and an object application that created some objects. Certainly writing applications that are enabled for compound documents is included in all the functionality contained in OLE 2.0's various DLLs.

But OLE 2.0 is much more than just compound documents. It introduces the future of the programming model for Windows applications, changing from an API-based system to an object-based system. Indeed, with just a focus on compound documents, vendors that produce accounting software or games see little use for OLE 2.0. However, these and all other classes of applications can greatly benefit from the technologies that provide the foundation for compound documents.

For this reason I wanted to show each component technology in the OLE 2.0 package in its own light as well as demonstrating how that component fits into the compound document model. Plenty of fish. Through this I hope to provide every applications vendor with the skill and understanding to fully exploit the technologies. Lessons in fishing. These technologies work on the existing Microsoft systems platforms of Windows 3.1 and Windows NT 1.0. Since the OLE 2.0 DLLs are redistributable with your application, your users are not required to update their systems past the versions above in order to use your application. You benefit from these new technologies without abandoning the installed base.

I also decided to write this book to provide an organization of OLE 2.0 such that my presentation of each chapter depends *soley* on information in previous chapters with absolutly no dependencies on later chapters. This sort of treatment is simply impossible with Window's traditional API–in order to discuss even the smallest piece of Windows with a sample you have to write a skeletal application involving message loops, instance handles, window classees, message procedures, icons, cursors, and possibly device contexts and various GDI calls. Every book on programming Windows has been forced to forward references later chapters, only adding to the laborious learning curve.

Since OLE 2.0 is not a technology for writing whole applications (because certainly we still use many Windows APIs) I am afforded the luxury to concentrate on OLE 2.0's features and how you use them to implement features in your application. I am able to present the material a little at a time, solidifying your understanding of that building block before moving on. I will take you on an evolutionary path where the work you do early in the book will be reusable in work you do in the later stages.

This evolutionary path will demonstrate how Windows itself is evolving from an API-oriented system to an object-oriented system. OLE 2.0 is just the first step in that evolution; what you learn in this book will carry you forward into future Windows with a definite edge.

OLE 2.0 is not a huge monolithic behemoth waiting to devour all availably resources. But you will have to learn how to fish. This book will keep you from starving in the meantime.

Kraig Brockschmidt
Redmond Washington
1 March, 1993