

Preface

This book grew from our own need to write good computer programs. We work in an interdisciplinary research center whose mission is to develop and evaluate statistical methodologies and disseminate them to a broad scientific community. Computing and software development play an increasingly important role both in our research and in dissemination. When faced with the task of actually writing a statistical program, however, we found ourselves without a road map. We needed an overall strategy.

After initial discussions, we drafted a brief document for our colleagues that contained guidelines for programming in Fortran. We also wrote libraries of Fortran procedures that could be shared among applications and a sample program to serve as a template for other projects. These materials proved extremely valuable. Those who followed the template—even if they had little or no prior experience in Fortran—quickly learned to produce code that was easy to maintain and debug.

At the same time, we wrestled with another crucial issue: how to disseminate statistical software to reach a wider audience. Some of our consumers would have no problem using an old-fashioned program that runs from the command line. But others had grown accustomed to graphical applications with windows, menus, buttons, etc. Still others were clamoring for add-ons to statistical programs such as SAS®. Given the diversity of environments in which data are being analyzed today, it seemed impossible to satisfy everyone without implementing our methods many times. As we began to learn about the Component Object Model (COM), however, we soon realized that packaging statistical routines as COM servers would help us to reach more potential users. With COM, we found that computational

procedures would have to be written carefully and methodically, but they would only have to be written once, as a single COM server can be called by many different client programs.

As this book took shape, a number of thorny (but in hindsight not too difficult) obstacles arose that needed special attention. One was the issue of how to invoke array-valued properties in a COM object from S-PLUS®. We learned from Insightful Corporation, the producers of S-PLUS, that this feature was available but undocumented. More generally, we struggled with how to best package Fortran arrays as variants. Our solution, which we implemented in our `variant_conversion` module, was to follow the practice of Excel and store an array as a variant containing an array of variants.

Another major hurdle was figuring out how to invoke a COM server from SAS. Interoperability with SAS was offered through the SAS Component Language but not through ordinary SAS programs. This prompted us to create the tool we call `sascomio`, a dynamic-link library that allows a SAS user to write COM client within PROC IML.

Finally, our preferred method for creating COM servers from Fortran code, which relied on a Compaq tool called the COM Server Wizard, became unavailable when the product was acquired by Intel in 2003. We suddenly found that the latter part of the book hinged upon a commercial product that had been discontinued! Fortunately, we were able to reimplement the functions of the COM Server Wizard through a Perl script that creates all of the necessary extra code. This new method for creating COM servers, which is described in Chapter 7, is almost completely automatic and requires very little knowledge of COM on the part of the programmer.

We would like to thank Linda Collins, Susan Murphy, and other folks from The Methodology Center at Penn State who encouraged us to publish this material as a book. Along the way, many individuals tested our ideas, provided suggestions, and helped to solve problems small and large. Brian Flaherty participated in the initial discussions as we developed our own unique style of pseudo object-oriented programming. Hwan Chung, Recai Yucel, and Hakan Demirtas tried these strategies and gave feedback. Jerry Maples did an outstanding job in writing the first version of our Perl script.

Portions of Chapter 8 rely on features of commercial software products that are not well documented. We wish to thank Steve Lionel of Intel for promptly answering our technical questions and giving us permission to modify the old Compaq templates for generating COM server code. Bob Rodriguez and Simon Smith at the SAS Institute were friendly and responsive as we worked out strategies for interfacing COM servers with SAS. Insightful's Jim Schimert helped us to learn about S-PLUS's COM interface. Thomas Baier, the author of `rcom`, gave helpful tips on calling COM servers from R.

Finally, we wish to thank John Kimmel and the staff at Springer for their help and patience as they waited for the completed manuscript. Three

anonymous reviewers gave many excellent suggestions that we took to heart. Based upon their advice, we expanded the introductory material on Fortran 95 in Chapters 2 and 3; added Chapter 4 on computational routines and numerical issues; added Chapter 6 on dynamic-link libraries; and removed all but the most necessary jargon and technical descriptions about COM and Windows® from Chapters 7 and 8.

Support for this project was provided by the National Institute on Drug Abuse grant 1-P50-DA10075 and by the National Institute on Mental Health grant 1-R01-MH60213.

David R. Lemmon
Joseph L. Schafer

University Park, Pennsylvania
September 2004



<http://www.springer.com/978-0-387-23817-3>

Developing Statistical Software in Fortran 95

Lemmon, D.R.; Schafer, J.L.

2005, XVI, 324 p., Softcover

ISBN: 978-0-387-23817-3