Scientific Software Design

The Object-Oriented Way

Damian Rouson
Sandia National Laboratories

Jim Xia
IBM Canada Lab, Markham

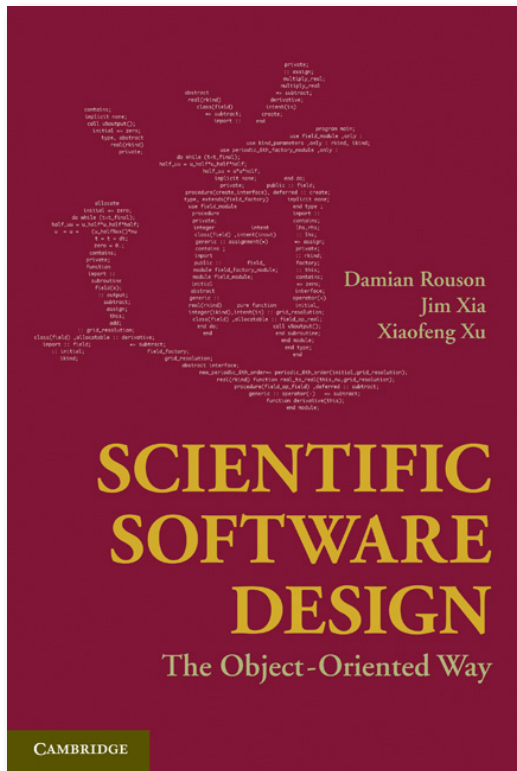**Xiaofeng Xu**
General Motors Corp.

About the Book

This book concerns software design. Although many current discussions of scientific programming focus on scalable performance, this book focuses on scalable design. The authors analyze how the structure of a package determines its developmental complexity according to such measures as bug search times and documentation information content. The work presents arguments for why these issues impact solution cost and time more than does scalable performance. The final chapter explores the question of scalable execution and shows how scalable design relates to scalable execution. The book's focus is on program organization, which has received considerable attention in the broader software engineering community, where graphical description standards for modeling software structure and behavior have been developed by computer scientists. These discussions might be enriched by engineers who write scientific codes. This book aims to bring such scientific programmers into discussion with computer scientists. The authors do so by introducing object-oriented software design patterns in the context of scientific simulation.

April 2011 | 404 pages
119 b/w illus. | 9 tables | 23 exercises
Hardback | 978-0-521-88813-4
~~List Price: USD 99.00~~
Discounted Price: USD 79.20

Praise for the Book

"This book makes a good case for the usefulness of
design patterns and object–oriented programming for
maintainable code, but disregards runtime performance
and scienti!c libraries ... one of those books that I wish I"d
read earlier in my programming career."

– **Ramses van Zon, Computing in Science and
Engineering, Jan/Feb 2012**

Key Features

• This book discusses object-oriented programming in For-
tran 2003 and C++

• This text is designed to help students and professionals
solve complex computer problems

• Each chapter has a set of exercise problems

Contents

Preface; Part I. The Tao of Scientific OOP: 1. Development
costs and complexity; 2. The object-oriented way; 3. Scien-
tific OOP; Part II. SOOP to Nuts and Bolts: 4. Design pat-
terns basics; 5. The object pattern; 6. The abstract calculus
pattern; 7. The strategy and surrogate patterns; 8. The pup-
peteer pattern; 9. Factory patterns; Part III. Gumbo SOOP:
10. Formal constraints; 11. Mixed-language programming;
12. Multiphysics architectures.