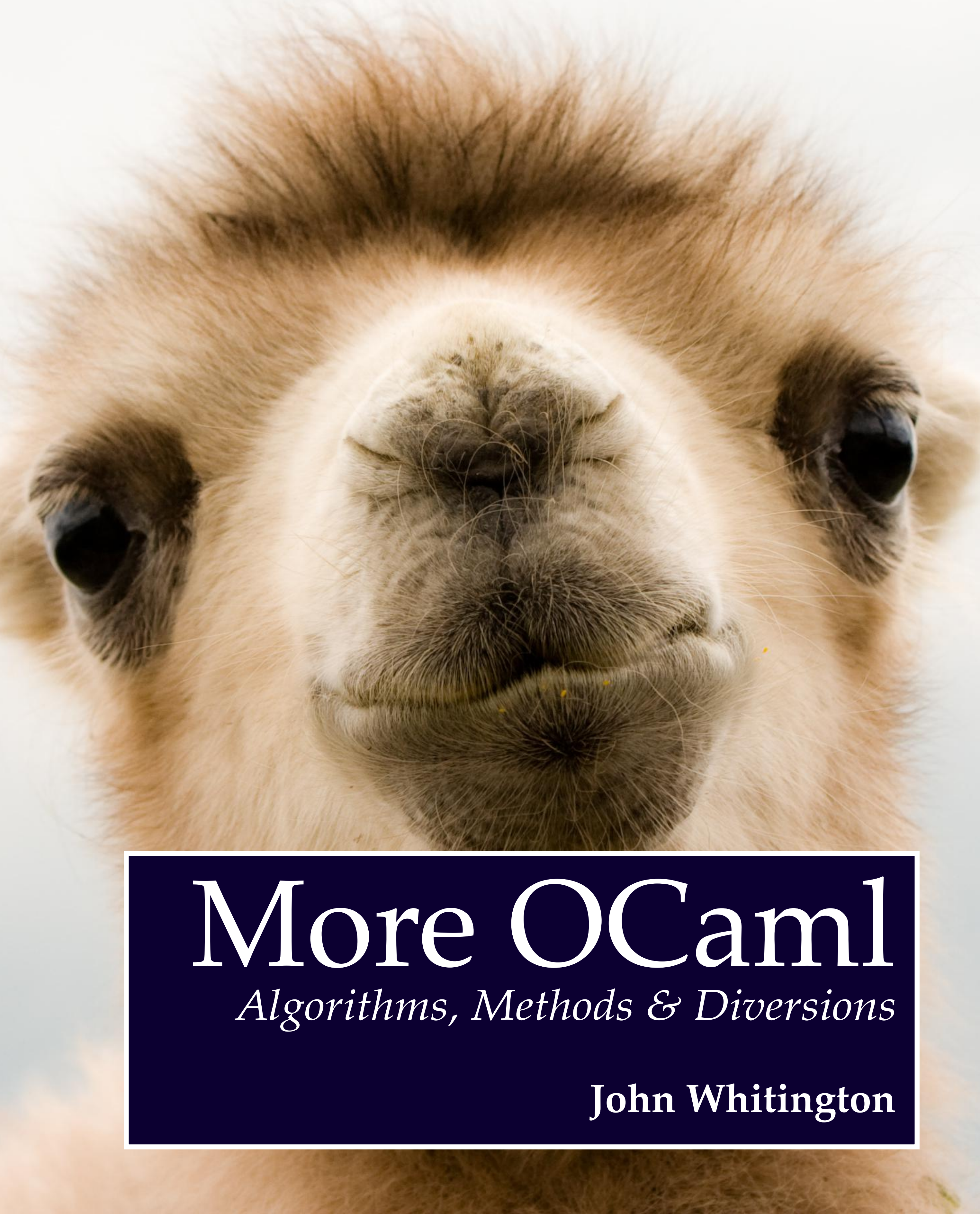

More OCaml

Algorithms, Methods & Diversions

John Whitington

MORE OCAML *Algorithms, Methods & Diversions*

In *More OCaml* John Whitington takes a meandering tour of functional programming with OCaml, introducing various language features and describing some classic algorithms. The book ends with a large worked example dealing with the production of PDF files. There are questions for each chapter together with worked answers and hints.

More OCaml will appeal both to existing OCaml programmers who wish to brush up their skills, and to experienced programmers eager to explore functional languages such as OCaml. It is hoped that each reader will find something new, or see an old thing in a new light. For the more casual reader, or those who are used to a different functional language, a summary of basic OCaml is provided at the front of the book.

JOHN WHITINGTON founded a software company which uses OCaml extensively. He teaches functional programming to students of Computer Science at the University of Cambridge. His other books include *"PDF Explained"* (O'Reilly, 2012) and *"OCaml from the Very Beginning"* (Coherent, 2013).

More OCaml

Algorithms, Methods & Diversions

John Whitington

COHERENT PRESS

COHERENT PRESS
Cambridge

Published in the United Kingdom by Coherent Press, Cambridge

First published August 2014
Reprinted with corrections July 2015
Reprinted 2016
Updated for OCaml language changes October 2017

A catalogue record for this book is available from the British Library

ISBN 978-0-9576711-1-9 Paperback

by the same author

PDF Explained (O'Reilly, 2012)
OCaml from the Very Beginning (Coherent, 2013)
A Machine Made this Book: Ten Sketches of Computer Science (Coherent, 2016)

Contents

Preface

When I wrote "OCaml from the Very Beginning", the intention was to have a book with no prerequisites – a bright individual, new to programming, could follow it. Because of this, and for length concerns, plenty of interesting material had to be omitted. This text, not being constrained in the same way, contains a variety of topics which require some existing experience with a functional language. Those who have read the previous text should have no problem with this one. Equally, it should be comprehensible to a functional programmer familiar with another language such as Standard ML or Haskell. The reader may need to make occasional reference to the OCaml manual.

There are, typically, two different activities when writing programs larger than a few dozen lines: firstly, dealing with the challenges of complexity inherent in the problem, by finding appropriate abstraction mechanisms and, secondly, finding and using the wide range of third-party libraries available for a given language. Most projects involve a combination of the two. In this text we concentrate wholly on the former, using nothing other than the OCaml Standard Library. Keeping up with the myriad third-party OCaml libraries is a task better suited to other media.

The book consists of sixteen short chapters falling broadly into three categories. Some introduce pieces of OCaml syntax with worked examples. Some survey practical topics such as input/output. Some cover little diversions or puzzles. The main matter of the book ends with a lengthy worked example: a program to build PDF files containing computer-generated drawings and text. There are full answers and hints for all questions in the book, and additional material in the online resources.

Acknowledgments

The quotation in Chapter 6 is taken from ISO-32000 © International Organization for Standardization. The tables of codes in the same chapter are taken from ITU-T T.30 © International Telecommunication Union. The presentation of the balancing operation for Red-Black trees in Chapter 11 and its functional implementation is due to Chris Okasaki, as described in the invaluable "Purely Functional Data Structures" (Cambridge University Press, ISBN 978-0521663502, 1998). Chapter 12 was inspired by a University of Cambridge Computer Science Tripos exam question set by Lawrence C. Paulson in 1999. Question 3 of that chapter is due to Peter D. Schumer in "Mathematical Journeys" (John Wiley & Sons, ISBN 0-471-22066-3, 2004).

I am grateful to the many colleagues and friends with whom I have been able to discuss OCaml style and substance, including Mark Shinwell, Leo White, Daniel Bünzli, Anil Madhavapeddy, Stephen Dolan and many others whom I have forgotten. Helpful comments on an earlier draft were provided by Stefan Schmiedl, Manuel Cornes, Jonas Bülow, Emmanuel Delaborde, Mario Alvarez Picallo, Giannis Tsaraias, Emmanuel Oga, and André Bjärby.