Foreword

IN THE FOREWORD to *The Cathedral and the Bazaar* (O'Reilly, 2001), Red Hat Chairman and CEO Bob Young identified two things that must happen if open-source software is going to make a lasting change: Open-source software must become widely used, and the users of the software must communicate and understand the benefits of the software development model.

I think that Jakarta Commons has succeeded on both fronts.

As one of the founders of the Jakarta Commons, I am utterly amazed at the success of the project as it nears the third anniversary of its founding. We knew what we did was going to be useful, but we didn't understand the extent to which the code and the community would grow. There are now 28 released components, 20 in progress in "the Sandbox," and lively, continuous discussion and debate on the mail lists—generally more mail traffic than a person can keep up with. Components have grown and "left the nest"—for example, Jakarta Cactus, the server-side testing framework, began as a Commons component. Most important, community participation has blossomed—from the 10 original committers, we have expanded to 82 as of this writing.

A bit of history: In early 2001, several of us working in various Jakarta subprojects noticed we had a problem. The subprojects had implemented a substantial variety of useful utility packages without any thought of reuse outside of each package's subproject. The result was that subprojects would reimplement (or copy outright) useful utility code from others. More important, we knew we had a trove of software to share but no way for people to find that software and obtain it in a simple and useful package. Driven by the motivation to make that software available and an open community debate on what the solution should be, Jakarta Commons was born—and since has become the place where Java programmers first look for help to solve common problems in server-side and client-side development.

The software is widely used throughout the Java world, both in commercial and open-source software. This familiar set of building blocks helps both developers and users: Developers have well-understood tools to work with, and users are familiar with the configuration and functionality of subsystems such as connection pools when they come from Jakarta Commons.

Equally as important, the growth of the community reflects the continued success of the software development model called *open source*. New components and improvements to existing components are driven by developers and users understanding that they can, to use the standard cliché, "scratch their itch." After showing up, all they need to do is contribute.

To that end, Harshad Oak, in *Pro Jakarta Commons*, brings what some may consider the rarest of contributions to open-source projects: comprehensive documentation. In this book, he covers 16 popular components. In each chapter,

you'll find not only background and motivation for the components but, for the working programmer, something even more valuable: code examples. With this book in hand, novice as well as experienced developers will be better able to take advantage of the treasure trove of useful utilities that is Jakarta Commons.

I encourage you to use this book, and the software it describes, to its fullest potential. And then if you have an improvement or an idea for something new, show up at "the Commons" and scratch that itch.

Geir Magnusson, Jr.
Wilton, Connecticut
February 2004