# Foreword

We live in rather interesting times.

The past ten years have seen so much technological change that anyone trying to actually get work done must devote a significant portion of his or her time to learning about all the latest technologies. Every Internet developer has had the experience of investing time and effort into a particular technology, only to run up against impenetrable roadblocks on the way to a final product.

The simple explanation for this is that the explosion of Internet technologies has produced a surfeit of incomplete, immature, or inflexible tools. All too often, the creators of these tools simply didn't anticipate our using them in the ways we must use them, and the tools prove rather flimsy or awkward under these conditions.

Not so with Perl and Apache. As the recipes in this book demonstrate, the marriage of the Perl programming language with the Apache Web server has brought unprecedented flexibility and power to Web developers. Most Web programmers stopped building simple, static Web sites ages ago—this is now the province of designers, not programmers. Today's complex Web sites demand custom user authentication procedures, dynamic server configuration, intelligent content negotiation, powerful server-side content-generation techniques, coordination with databases, and so on. mod_perl provides all this and more, without batting so much as an eyelash, for the low, low price of zero dollars.

With all the flexibility offered by mod_perl, many developers might have a hard time figuring out how to navigate the wide-open seas of mod_perl development. Many things are possible with mod_perl, but few examples exist of the right ways to do uncommon things. (There's always more than one way to do it with Perl, and although there's often no single right way to do something, some ways are usually better than others.) In addition, many developers can get stuck in a rut without knowing it, simply by repeating techniques that worked for them a long time ago but that they've never re-analyzed.

This book is quite helpful in both situations. Developers searching for solutions to specific problems can use this book as a collection of ready-made recipes to be understood and applied to their problems. Developers searching for continuing mod_perl education will find the recipes contained herein to be enlightening, well-researched, and broadly applicable.

Finally, although we programmers rarely admit this to our managers, we all know that mod_perl is fun. This characteristic is almost always a good thing for our managers anyway, because the main reason mod_perl is fun is that it makes accomplishing large goals easier than we would expect it to be, and as programmers we derive pleasure from that. The recipes in this book will appeal to that part of your programming personality. While editing the recipes in this book, I was continually delighted to find new solutions to problems I had never even thought of. After making your way through these recipes, you might find yourself fixing parts of your code you didn't know were broken, and attacking problems you previously thought you couldn't solve.

Have fun.

—*Ken Williams*
*ken@mathforum.org*