



Foreword

Flexibility

When we created the Apache project five years ago, our goal was to ensure that the server-side of the Web would never be dominated by the proprietary interests of any single company. To the Apache Group, the Web is more than just a network-based application; it is the means for people to communicate across geographical and political boundaries, to cooperate in the sharing of information, and to collaborate in the creation of new works of the imagination. Web servers are the printing presses of the Internet age.

In order to achieve our goal, we needed more than just another free Web server. We needed software that is, in every way, a commercial-grade implementation of the standards that define the Web. Any feature that might distinguish one Web server over another must be achievable in Apache, using standard protocols where others might use proprietary extensions, and with the robustness expected of a professional tool.

At the same time, we also knew that a web server must be a workhorse application — subject to the anarchic nature of the Internet, and yet expected to work 24 hours a day, 7 days a week, 52 weeks a year. Being webmasters for our own sites, we knew that the greater the performance requirements, the more emphasis there must be on maintaining a small server “footprint” — the size and complexity of the software executable that acts as the brains of the web server. High-performance sites needed the ability to remove any functionality from the server that was not needed for their own resources.

When Robert Thau designed the module framework that distinguishes the Apache architecture, its purpose was to provide webmasters with the ability to include almost any feature they might want in a web server, and yet do so in a way that avoided requiring the same features to be present on every server. While keeping the core server simple, the module framework allows each server to be tailored to the specific needs of the site it serves. Flexibility.

However, flexibility doesn't come without cost. In order to properly configure and run an Apache server, a webmaster needs to be familiar with the hundreds of feature modules that are available. Furthermore, each module can define its own set of configuration directives for controlling its behavior and that of the server as a whole. Without a guide, even us core server developers would get lost in the maze of optional features that make Apache work so well across so many different sites.

What Ralf has provided, in the form of this desktop reference, is a complete guide to the features and configuration information needed to run Apache as a robust, flexible, and high-performance web server. As one of the core developers, Ralf provides a level of insight regarding the inner-workings of Apache that you won't find in a typical user manual. This is the kind of book that you want located next to every server console.

As you work with the Apache software, remember that all of this has been accomplished by a volunteer community of software developers collaborating across the Internet. Open source is shared custom software — it only comes about when individuals have the foresight to share what they do with the rest of the world. The Apache Software Foundation supports a number of open-source software projects related to Web technology, including the HTTP server project, and welcomes anyone with a desire to contribute toward the future of Apache.

— Roy T. Fielding,
July 2000, Irvine, California