

Preface

You'd be hard-pressed to find a buzzword hotter than *web services*. Breathless articles promise that web services will revolutionize business, open new markets, and change the way the world works. Proponents call web services "The Third-Generation Internet," putting them on a par with email and the browseable web. And no protocol for implementing web services has received more attention than SOAP, the Simple Object Access Protocol.

This book will give you perspective to make sense of all the hype. When you finish this book, you will come away understanding three things: what web services are, how they are written with SOAP, and how to use other technologies with SOAP to build web services for the enterprise.

While this book is primarily a technical resource for software developers, its overview of the relevant technologies, development models, standardization efforts, and architectural fundamentals can be easily grasped by a nontechnical audience wishing to gain a better understanding of this emerging set of new technologies.

For the technical audience, this book has several things to offer:

- A detailed walk-through of the SOAP, WSDL, UDDI, and related specifications
- Source code and commentary for sample web services
- Insights on how to address issues such as security and reliability in enterprise environments

Web services represent a powerful new way to build software systems from distributed components. But because many of the technologies are immature or only address parts of the problem, it's not a simple matter to build a robust and secure web service. A web service solution today will either dodge tricky issues like security, or will be developed using many different technologies. We have endeavored to lay a roadmap to guide you through the many possible technologies and give you sound advice for developing web services.

Will web services revolutionize everything? Quite possibly, but it's not likely to be as glamorous or lucrative, or happen as quickly as the hype implies. At the most basic level, web services are plumbing, and plumbing is never glamorous. The applications they make possible may be significant in the future, and we discuss Microsoft Passport and Peer-to-Peer (P2P) systems built with web services, but the plumbing that enables these systems will never be sexy.

Part of the fundamental utility of web services is their language independence—we come back to this again and again in the book. We show how Java, Perl, C#, and Visual Basic code can be easily integrated using the web services architecture, and we describe the underlying principles of the web service technologies that transcend the particular programming language and toolkit you choose to use.