Preface

Introduction

The goal of this book is to introduce XML to a bioinformatics audience. It does so by introducing the fundamentals of XML, Document Type Definitions (DTDs), XML Namespaces, XML Schema, and XML parsing, and illustrating these concepts with specific bioinformatics case studies. The book does not assume any previous knowledge of XML and is geared toward those who want a solid introduction to fundamental XML concepts.

The book is divided into nine chapters:

- **Chapter 1: Introduction to XML for Bioinformatics.** This chapter provides an introduction to XML and describes the use of XML in biological data exchange. A bird's-eye view of our first case study, the Distributed Annotation System (DAS), is provided and we examine a sample DAS XML document. The chapter concludes with a discussion of the pros and cons of using XML in bioinformatic applications.
- **Chapter 2: Fundamentals of XML and BSML.** This chapter introduces the fundamental concepts of XML and the Bioinformatic Sequence Markup Language (BSML). We explore the origins of XML, define basic rules for XML document structure, and introduce XML Namespaces. We also explore several sample BSML documents and visualize these documents in the Rescentris Genomic Workspace™ Viewer.
- **Chapter 3: DTDs for Bioinformatics.** This chapter introduces XML Document Type Definitions (DTDs). With DTDs, you can define specific rules for XML document construction and validate XML instance documents against these rules. This chapter builds a DTD for representing protein sequences and does so in incremental stages—we therefore start out simply and add layers of complexity as new concepts are introduced. The chapter also includes an overview of XML data formats available from The National Center for Biotechnology Information (NCBI), and provides a complete description of the NCBI TinySeq DTD.
- **Chapter 4: XML Schemas for Bioinformatics.** XML Schema represents the successor to XML Document Type Definitions (DTDs). We begin by comparing the two specifications and describe some of the advantages of using XML Schema. To illustrate core concepts, we rebuild the protein sequence DTD from Chapter 3 as an XML Schema, enabling you to compare the two specifications directly. The chapter concludes with a discussion of the Proteomics Standards Initiative Molecular Interaction (PSI-MI) XML format, an XML exchange format used to encode protein–protein interactions.
- **Chapter 5: Parsing NCBI XML in Perl.** Perl remains the programming language of choice for many in bioinformatics. This chapter therefore explores several options for parsing XML in Perl,

Second, I want to thank everyone at the Computational Biology Center (cBio) at Memorial Sloan-Kettering Cancer Center, where I work. Chris Sander has created a unique and intellectually vibrant center, where I have been able to learn and thrive, and gain hands-on experience in many of the technologies described in this book. Alex Lash provided help in understanding the database resources at NCBI, and pointed me to the NCBI E-Fetch service described in Chapter 5. Anton Enright provided detailed feedback on Chapter 5. Gary Bader provided me with much-needed background information about specific biological databases and detailed background information about the PSI-MI XML format. Gary also provided feedback on Chapter 4.

Third, I want to thank Lorrie LeJeune, Simon St. Laurent, Tracey Cranston, and Brian Gilman who helped out with an earlier incarnation of this book, before it found a new home at Springer-Verlag.

Fourth, I want to thank several additional individuals who generously agreed to review specific chapters and provided scientific and technical feedback. Peter Covitz, Director of Bioinformatics Core Infrastructure at the National Cancer Institute (NCI) provided feedback on Chapter 9, and answered many of my questions regarding the NCI caBIO bioinformatics framework. Jeff Spitzner, Chief Science Officer of Rescentris, Ltd., reviewed Chapter 2, and provided valuable feedback regarding BSML. I also want to thank Paul Farrell, my editor at Springer, for ushering the book to completion and keeping me on schedule.

Finally, I want to thank my entire family for supporting me during the whole writing process associated with this book. Thanks to Dad, who hired me at the ripe age of twelve to complete my first bioinformatics programming project (really), and instilled in me a love of scientific ideas and ideals. Thanks to Mom for buying me my first computer (a Commodore Vic 20), and always reminding me to remain balanced. Special thanks to Nelli and Carla for their support and encouragement.

Lastly, I want to thank my wife, Amy. All authors thank their wives in the acknowledgments, but Amy has the distinction of supporting me in this fourth book endeavor while also being pregnant. I do not know how she puts up with me, but she has been my rock, my soulmate, my everything. I love you.

and focuses on two standard interfaces: the Simple API for XML (SAX) and the Document Object Model (DOM). We also explore the NCBI E-Fetch service, and retrieve nucleotide sequence records in XML in real time. This chapter assumes some prior knowledge of Perl.

- **Chapter 6: The Distributed Annotation System.** This chapter provides comprehensive coverage of the Distributed Annotation System (DAS), a distributed XML protocol used to exchange genome annotation data. To put DAS in perspective, we begin by exploring the process of genome annotation, and illustrate the DAS protocol from the end-user perspective. We then describe the DAS XML protocol in detail, and examine numerous sample XML documents from the Ensembl and UCSC DAS servers. The chapter includes a reference guide to all DAS commands, and a preview of anticipated features in the next version of DAS.
- **Chapter 7: Parsing DAS Data in SAX.** Despite the popularity of Perl, Java is becoming increasingly popular in bioinformatic applications. This chapter therefore describes the mechanics of parsing XML documents using the Java Simple API for XML (SAX). SAX is the de facto event-based XML parsing standard, and is widely implemented by many XML parsers, including several open source XML parsers. Several sample DAS applications are demonstrated, including one sample application which makes use of the open source BioJava library. This chapter assumes some prior knowledge of Java.
- **Chapter 8: Parsing DAS Data in JDOM.** This chapter focuses on the fundamentals of the JDOM API, a popular alternative to the SAX API. With JDOM, Java applications can easily navigate through XML document tree structures and extract elements, attributes, and character data. JDAS, an open source DAS client library, created by the author, is explored in detail. This chapter also assumes some prior knowledge of Java.
- **Chapter 9: Web Services for Bioinformatics.** Web services represent a new paradigm for building distributed web applications, and are currently being used extensively in bioinformatics. This chapter begins by presenting two broad approaches to building web services: the Representational State Transfer (REST) approach and the SOAP approach. We explore each approach in detail, and provide complete details on the latest SOAP specification. We also explore caBio, a comprehensive web service built by the National Cancer Institute (NCI).

Companion Web Site

This book includes a companion web site, available at: *http://www.xmlbio.org*. All the sample XML documents, and example Perl/Java programs described in the book are available for download.

Acknowledgments

Many people deserve special acknowledgments for making this book happen. First, I want to thank Lincoln Stein of Cold Spring Harbor Laboratory. Lincoln's presentation at the 2002 O'Reilly Open Bioinformatics Conference and his subsequent paper in *Nature* (described in Chapter 1), inspired me to write this book in the first place. Lincoln's vision of creating a "bioinformatics nation" is a compelling one; I hope this book provides readers with the nuts and bolts information to make Lincoln's dream a reality. Lincoln also provided answers to many of my questions regarding the DAS protocol, and a complete technical review of Chapter 6. His feedback and detailed explanations were invaluable.