LEGITIMATE
APPLICATIONS OF
PEER-TO-PEER
NETWORKS

DINESH C. VERMA
IBM T. J. Watson Research Center
LEGITIMATE APPLICATIONS OF PEER-TO-PEER NETWORKS
LEGITIMATE APPLICATIONS OF PEER-TO-PEER NETWORKS

DINESH C. VERMA
IBM T. J. Watson Research Center
Dedicated to
Paridhi, Archit, and Riya
CONTENTS

Preface xi
  Who Will Benefit from This Book? xii
  Who Is This Book Not For? xiii
  Organization of the Book xiii

1 The Peer-to-Peer Architecture 1
  1.1 Distributed Applications 1
    1.1.1 A Distributed Computing Example 2
    1.1.2 Client-Server Architecture 5
    1.1.3 Peer-to-Peer Architecture 7
  1.2 The Peer-to-Peer Software Structure 7
    1.2.1 Base Overlay Layer 8
    1.2.2 Middleware Functions 9
    1.2.3 Application Layer 10
  1.3 Comparison of Architectures 11
    1.3.1 Ease of Development 12
    1.3.2 Manageability 12
    1.3.3 Scalability 13
    1.3.4 Administrative Domains 15
    1.3.5 Security 15
    1.3.6 Reliability 16
2 Peer Discovery and Overlay Formation 19

2.1 Discovery 20
  2.1.1 Static Configuration 20
  2.1.2 Centralized Directory 21
  2.1.3 Using the Domain Name Service 23
  2.1.4 Member Propagation Techniques with Initial Member Discovery
     2.1.4.1 Member Propagation with Full Member List 25
     2.1.4.2 Member Propagation with Partial Member List 25
     2.1.4.3 Member Propagation with a Hint Server 26

2.2 Overlay Formation 27
  2.2.1 Creating an Overlay Link 28
     2.2.1.1 Communicating Across Firewalls 28
     2.2.1.2 Communicating Across Two Firewalls 29

2.3 Topology Selection 31
  2.3.1 Random Mesh Formation 32
  2.3.2 Tiered Formation 33
  2.3.3 Ordered Lattices 34

3 Application-Layer Multicast 37

3.1 General Multicast Techniques 38
  3.1.1 Group Addressing 39
  3.1.2 Group Maintenance 40
  3.1.3 Message Forwarding Scheme 42
  3.1.4 Multicast Routing 44
  3.1.5 Secure Multicast 45
  3.1.6 Reliable Multicast 46
  3.1.7 Multicast Flow and Congestion Control 47

3.2 Network-Layer Multicast—IP Multicast 49
  3.2.1 Problems with IP-Layer Multicast 50

3.3 Application-Layer Multicast 52
  3.3.1 Broadcast Mechanisms in Peer-to-Peer Networks 52
  3.3.2 Multicast in Peer-to-Peer Overlays 53

4 File-Sharing Applications 55

4.1 File-Sharing Overview 55
5 File Storage Service

5.1 Handle Management 78
5.2 Retrieving Files with Handles 80
   5.2.1 Circular Ring Routing 81
   5.2.2 Plaxton Scheme 82
   5.2.3 CAN Routing Algorithm 84
   5.2.4 Modified Network Routing Schemes 85
   5.2.5 Modified Broadcast 86
5.3 Miscellaneous Functions 87
   5.3.1 Access Control 87
   5.3.2 Availability and Reliability 89
5.4 Usage Scenarios 90
   5.4.1 Distributed File Systems 91
   5.4.2 Anonymous Publishing 92

6 Data Backup Service

6.1 The Traditional Data Management System 96
6.2 The Peer-to-Peer Data Management System 98
   6.2.1 The Backup/Restore Manager 99
   6.2.2 The Peer Searcher 100
   6.2.3 The File Searcher 101
   6.2.4 The Properties Manager 101
   6.2.5 The Data Manager 102
   6.2.6 The Schedule Manager 103
6.3 Security Issues 103
PREFACE

Peer-to-peer computing has gained a lot of prominence and publicity in the past years. Several start-up companies offer software packages that can be used for building peer-to-peer overlay networks for applications such as file sharing and music swapping. However, the legal and ethical questions associated with swapping copyrighted material on the Internet have clouded the entire peer-to-peer technology industry and given peer-to-peer networks a bad name. Many companies and universities have banned peer-to-peer networks because of copyright infringement issues.

The vitriolic legal debate over copyright violations has the potential danger of eclipsing many different legitimate applications that can be developed over peer-to-peer networks. Such networks offer decentralized domains of control and an increased level of parallelism and are extremely resilient to failures of machines and network. As a result, peer-to-peer technology can be used to develop many new applications that would benefit from the unique features offered by the peer-to-peer architecture. Some companies already offer peer-to-peer software for applications such as instant messaging, search engines, and data storage.

The goal of this book is to describe some of the applications that can exploit the unique features of peer-to-peer networks and to discuss why developing these applications on a peer-to-peer net-
work is better than developing them in the traditional manner. Peer-to-peer technology is not an appropriate choice for all applications, and the book also discusses the disadvantages that a peer-to-peer version of an application would have compared with a conventional client-server implementation of the same application.

**WHO WILL BENEFIT FROM THIS BOOK?**

This book is intended for enterprise IT operators, architects, software developers, and researchers who are interested in peer-to-peer networks. If you are a graduate student or network researcher who wants to learn about the different types of applications that could be developed on a peer-to-peer infrastructure, you will find this book to be very useful. The book contains a detailed description of many applications that can be developed on a peer-to-peer infrastructure and compares the relative merits and demerits of building those applications with a traditional client-server approach versus a peer-to-peer approach.

If you are a network operator who operates a corporate intranet, you will find the overview of peer-to-peer architecture contained in this book very helpful. This book will help you understand peer-to-peer systems and how you can monitor and control the peer-to-peer applications running on your intranet. You may want to pay special attention to the sections of Chapter 4 that describe the different measures and countermeasures to monitor and regulate your network bandwidth usage with peer-to-peer applications.

If you are an architect responsible for an enterprise network or an enterprise IT system and would like to harness peer-to-peer technology for use within your enterprise system, this book will help you understand the relative merits and demerits of peer-to-peer implementations of various applications compared with the traditional client-server implementations of the same application. This will help you decide among the different possible ways of implementing your own applications, as well as providing useful information to understand the trade-offs of different vendor products.

Finally, if you are a software developer working for a peer-to-peer software development company, or want to develop your own
peer-to-peer application, this book will give you a comprehensive view of how different researchers have tried to approach problems similar to the ones you may be encountering in your specific application.

WHO IS THIS BOOK NOT FOR?

This book is not intended to provide information about the specific implementation of peer-to-peer networks. It does not describe the architecture of a specific peer-to-peer protocol but discusses general techniques that can be used for peer-to-peer network construction and the applications that can run on such networks. Although it provides a broad overview of many common peer-to-peer networks, you will not find it appropriate if you are looking for an in-depth treatment of a specific instance.

This book is not intended as a discussion of the legal issues related to running peer-to-peer systems. If you are looking for a legal discussion on running applications on peer-to-peer networks, this book is not for you. This is a technical book that describes various applications that can take advantage of peer-to-peer technology in various manners and is not intended for a legal audience.

ORGANIZATION OF THE BOOK

The material in this book is organized into 10 chapters.

Chapter 1 discusses the architecture of a peer-to-peer network and compares the peer-to-peer approach to traditional client-server architecture. It examines the key benefits of a peer-to-peer architecture and discusses its relative merits vis-à-vis the traditional client-server architecture.

Chapter 2 discusses how the different nodes in a peer-to-peer network can be connected together to create an overlay network connecting the participants over the Internet. It also discusses how the nodes discover each other in order to create such an overlay.

Chapter 3 discusses a common building block that supports many of the applications that are run on peer-to-peer systems, namely, the ability to multicast messages to a number of partici-
pants without relying on any capabilities of the underlying network. This type of multicast communication performed at the application layer is essential for the operation of most peer-to-peer applications.

Chapter 4 provides a brief overview of the ubiquitous application driving many peer-to-peer systems: the sharing of files on the peer-to-peer network. It discusses the technology used for such applications and describes the different techniques that can be used to improve the efficiency and security of file-sharing applications. It also discusses the legal and resource usage issues that arise when peer-to-peer networks are used to illegally exchange copyrighted material. It describes the measures enterprise operators, ISPs, and copyright holders can use to prevent illegal exchange on peer-to-peer networks as well as the countermeasures that peer-to-peer developers can use to work around those controls.

Chapter 5 discusses the use of peer-to-peer networks to build file storage services on the Internet. Peer-to-peer computing can be used to provide storage systems that are capable of providing ubiquitous access to files of a user from any location. The techniques and mechanisms needed to build file storage systems are described in this chapter.

Chapter 6 discusses how peer-to-peer networks can be used to build a data backup service without requiring expensive backup servers or a large amount of disk-space. The assumptions under which such a service works well are also discussed in this chapter.

Chapter 7 discusses how a directory service can be improved by using a peer-to-peer infrastructure. The directory service on peer-to-peer infrastructure can provide more scalable solutions when distributed administration domains are involved in managing different portions of the directory.

Chapter 8 discusses how peer-to-peer systems can be used to build publish-subscribe middleware, which is commonly used in many enterprises to create different types of distributed applications. The advantages and disadvantages of a peer-to-peer implementation versus a traditional client-server implementation are discussed.

Chapter 9 discusses a variety of different applications that can be built on top of the different middleware and systems that are described in the earlier chapters. It includes examples of peer-to-peer instant messaging, peer-to-peer IP telephony, collaborative
databases, collaborative hosting of content, and anonymous web surfing, all applications that can be built in a scalable manner with peer-to-peer technology.

Finally, Chapter 10 discusses some of the topics that are related to peer-to-peer technology. It includes a discussion of peer-to-peer applications that were developed before file sharing propelled peer-to-peer technology into the evening news as well as grid computing, which has many similarities with peer-to-peer systems.
THE PEER-TO-PEER ARCHITECTURE

In this chapter, we look at the general architecture of a peer-to-peer system and contrast it with the traditional client-server architecture that is ubiquitous in current computing systems. We then compare the relative merits and demerits of each of these approaches toward building a distributed system.

We begin the chapter with a discussion of the client-server and peer-to-peer computing architectures. The subsequent subsections look at the base components that go into making a peer-to-peer application, finally concluding with a section that compares the relative strengths and weaknesses of the two approaches.

1.1 DISTRIBUTED APPLICATIONS

A distributed application is an application that contains two or more software modules that are located on different computers. The software modules interact with each other over a communication network connecting the different computers.

To build a distributed application, you would need to decide how many software modules to include in the application, how to place those software modules on the different computers in the network, and how each software module discovers the other modules it needs to communicate with. There are many other tasks