

Foundations of Computer Systems Research

计算机系统研究基础
(英文版)

施巍松



高等教育出版社
HIGHER EDUCATION PRESS

Foundations of Computer Systems Research

计算机系统研究基础(英文版)

施巍松



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

图书在版编目(CIP)数据

计算机系统研究基础 = Foundations of Computer Systems Research : 英文 / 施巍松主编. — 北京 : 高等教育出版社, 2010.10

ISBN 978-7-04-029063-9

I. ①计… II. ①施… III. ①计算机系统-英文
IV. ①TP30

中国版本图书馆CIP数据核字(2010)第195231号

策划编辑 刘 英 责任编辑 刘 英 封面设计 张 楠 责任印制 陈伟光

出版发行	高等教育出版社	购书热线	010-58581118
社 址	北京市西城区德外大街4号	咨询电话	400-810-0598
邮政编码	100120	网 址	http://www.hep.edu.cn
			http://www.hep.com.cn
经 销	蓝色畅想图书发行有限公司	网上订购	http://www.landracom.com
印 刷	涿州市星河印刷有限公司		http://www.landracom.com.cn
		畅想教育	http://www.widedu.com
开 本	787×1092 1/16	版 次	2010年10月第1版
印 张	17.75	印 次	2010年10月第1次印刷
字 数	430 000	定 价	59.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 29063-00

*To my parents, my wife Wei,
and my daughters, Ivy and Macy*

Preface

The field of computer systems is the core of computer science, and is concerned with how to design and implement a “good” system that is able to satisfy a variety of requirements and needs from end users and applications. The definition of “good” depends on to whom we are talking, and it could mean one or several of these basic requirements, to name a few: reliability, scalability, availability, usability, adaptability, agility, dependability, and performability. We envision that more and more new requirements will emerge as computing becomes more and more transparent and embedded in our daily life. This unstoppable trend, however, brings great challenge to future computer systems designers and practitioners. There is a gap between what the students learned in the classroom and what they are going to build in the real world. The goal of this book is to present the fundamental knowledge and techniques by laying out the foundations for the students who want to become researchers in the general area of computer systems. The book is intended for both senior undergraduate students and junior graduate students in the fields of computer science and computer engineering. Practitioners and systems designers in industry and research laboratories will find the book a very useful reference. One important principle guiding the writing of this book is that it should contain the material I would want my own students to learn before beginning their research.

The focus of systems research has changed considerably since the inception of computers. However, we have witnessed that several fundamental techniques and principles have been frequently used in systems research in the last fifty years. The goal of this book is to provide beginning computer systems researchers enough background for undertaking further systems research in their career path. The philosophy of this book is *foundation*, in other words, only the techniques that will at least survive for more than 10 years are selected.

In addition to general information for computer systems research, we have selected 15 specific topics that can be applied in the three phases of systems research, including *design*, *implementation* and *evaluation*. The 15 topics are selected based on the research experience of the authors in the past. It is by no means the complete list of techniques and probably has the bias of the authors. As time goes on, we will add more topics once we think a technique is fundamental enough to be included in the book. You are more than welcome to contribute new topics to this set by contacting the author directly by emailing at the following address csr.foundations@gmail.com. Supplemental materials of this book will be available at <http://www.cs.wayne.edu/~weisong>.

These 17 chapters of the book are organized into four parts, *General*, *Design*, *Implementation* and *Evaluation*, listed as follows:

1. Part I: General
 - Elements
 - Rules of Thumb

2. Part II: Design
 - Bloom Filters
 - Distributed Hash Tables
 - Locality Sensitive Hashing
 - XOR Operations
 - Adaptation
 - Optimistic Replication
 - Reputation and Trust
 - Moving Average
 - Machine Learning
3. Part III: Implementation
 - Asynchronous I/O
 - Multithreading
 - Virtualization
4. Part IV: Evaluation
 - Queueing Theory
 - Black Box Testing
 - Goodness-of-Fit

Following the first two chapters about the elements and rules of thumb of systems research, each topic will be covered by one chapter. To help readers understand these topics easily, we also provide two case studies of each topic at the end of each chapter (if possible), in addition to the basic description of the motivation and idea. These case studies are selected from different aspects of computer systems design and implementation.

Acknowledgement

This book would not be possible without the interactions with my past and current students at the Mobile and Internet Systems Laboratory at Wayne State University and several collaborators in China. I am greatly indebted to them: Hanping Lufei, Zhengqiang Liang, Kewei Sha, Safwan Al-Omari, Zhifeng Yu, Jayashree Ravi, Sharun Santhosh, Yonggeng Mao, Brandon Szeliga, Yong Xi, Tung Nguyen, Guoxing Zhan, Suhil Rawshdeh, Shinan Wang, Thanh Do Duc, Chenjia Wang, Kevin Monaghan, John Cavicchio, Hui Chen, Lingjun Fan, Huajia Mao, Ying Song, Yaqiong Li, Jianfeng Zhan, and Yongqiang He. The interactions with them directly motivated me to write this book. I also would like to thank the students who attended the distributed systems course I taught at the Institute of Computing Technology (ICT), Chinese Academy of Sciences, and the 2009 Dragon Star course entitled “Principles of Computer Systems Design” I taught at Tsinghua University in May 2009. The Dragon Star program was partially sponsored by the National Natural Science Foundation of China and ICT. Early feedbacks from them greatly helped reshape the content of this book. As a matter of fact, the majority of the book’s contents were written by the volunteer students from these two courses, under my guidance. Without their hard work, we would have not seen this print for at least another couple of years.

I am extremely fortunate to have the opportunity to work with the authors of Chapter 3 to Chapter 17 of the book, including Dajun Lu, Yanbin Liu, Wei Jiang, Dawen Xu, Mingdong Tang, Wei Lee, Dandan Tu, Mingwen Chen, Shinan Wang, Nan Yuan, Yongbin Zhou, Changlin Wan, Tung Nguyen, Yanan Li, Zhuhua Liao, Zhengqiang Liang, and Yong Zhao. All of them are dedicated and enthusiastic on their corresponding chapters. Without their tremendous effort, you would have not been able to see the book presented in such a coherent way.

I want to thank Professor Zhiwei Xu, the Chief Technology Officer of Institute of Computing

Technology, Chinese Academy of Sciences, who invited me to spend six months at ICT during my sabbatical leave. My appreciation also goes to ICT, including the Graduate Education Office and their staff, Key Laboratory of Network Science and Technology, Key Laboratory of Systems and Architecture, and so on. Without their support, I would not have had a chance to meet and interact with fellow students at ICT. It is these interactions that inspired me to complete the text as soon as possible. I would like to thank Professor Wenguang Chen from Tsinghua University, Professor Nong Xiao from National University of Defense Technology and Professor Limin Sun from Institute of Software, Chinese Academy of Sciences, who gave me a great opportunity to teach an early version of this book at their respective institutions. I also would like to thank many friends, especially Professor Yafei Dai from Peking University, Professor Yunquan Zhang from Institute of Software, Chinese Academy of Sciences, for their early interests to the book convinced me that writing this book is the right direction.

My deepest appreciation also goes to Ms. Ying Liu from Higher Education Press. It is her great enthusiasm and support that made this book be published so smoothly. I want to thank Scott D. Vance, Jacqueline D. Brown, who gave great comments on improving the presentation of this book.

I also would like to thank the support from Wayne State University and the colleagues at the Department of Computer Science. The generous support for my sabbatical leave (Winter 2009) and one course release as part of the 2009—2010 Wayne State University Career Development Chair Award (2009—2010) which afforded me enough time to focus on writing this book.

Last but not least, I am greatly indebted to my family: my parents, my wife, and my daughters. Without my parents' help with baby caring, I would have not been able to dedicate too much time to the book this year. My wife, as always, supported me in writing this book and pushed me to complete it as early as possible. Without her great support and understanding, I don't think this publication would have been possible. I also want to thank my two daughters, Ivy and Macy, who showed great understanding and lent cooperation. They are responsible for this book in more ways than even they realize. This book is dedicated to them.

Enjoy the book!

Weisong Shi
 Detroit
 August 2010

Contents

Part I General

1	Elements	3
1.1	Top Systems Conferences/Journals	3
1.2	How to Read a Research Paper	5
1.3	How to Write a Research Paper	6
1.3.1	Abstract	6
1.3.2	Introduction	6
1.3.3	Background Information/Problem Statement	6
1.3.4	Your Approach	6
1.3.5	Implementation	6
1.3.6	Performance Evaluation	7
1.3.7	Related Work	7
1.3.8	Conclusions	7
1.3.9	Acknowledgement	7
1.3.10	References	8
1.3.11	Most Common Mistakes in Paper Writing	8
1.4	How to Give a Presentation	9
1.4.1	General Approach	9
1.4.2	Understanding the Paper	10
1.4.3	Adapting the Paper for Presentation	10
1.4.4	Slides	11
1.4.5	The Dry-Run	12
1.4.6	To Memorize or not to Memorize?	13
1.4.7	You Are on the Stage	13
1.4.8	Interacting with the Audience and Dealing with Questions	14
1.5	Final Words: On Being a Scientist	14
	References	15
2	Rules of Thumb	16
2.1	Rules of Thumb	16
2.2	Further Readings	17
	References	17

Part II Design

3	Bloom Filters	21
3.1	Introduction	21
3.2	Standard Bloom Filters	22
3.2.1	Basic Idea of Bloom Filters	22
3.2.2	False Positive Rate Estimation	23
3.2.3	Optimal Number of Hash Functions	23
3.2.4	Another Method of Implementing	24
3.3	Counting Bloom Filters	25
3.4	Compressed Bloom Filters	27
3.5	<i>D</i> -left Counting Bloom Filters	28
3.5.1	<i>D</i> -left Hashing	28
3.5.2	<i>D</i> -left Counting Bloom Filters	29
3.5.3	Performance	30
3.6	Spectral Bloom Filters	31
3.6.1	Basic Principle of SBF	31
3.6.2	SBF Frequency Query Optimization	33
3.7	Dynamic Counting Bloom Filters	33
3.8	Case Studies	34
3.8.1	Case Study 1: Summary Cache	35
3.8.2	Case Study 2: IP Traceback	36
3.9	Conclusion	36
	References	37
4	Distributed Hash Tables	38
4.1	Introduction	38
4.2	An Overview of DHT	39
4.3	The Overlay Network of DHT	40
4.4	Chord: An Implementation of DHT	42
4.4.1	Topology of Chord	42
4.4.2	Key Lookup in Chord	42
4.4.3	Dynamic Updates and Failure Recovery	44
4.5	Case Study 1: Cooperative Domain Name System (CoDoNS)	46
4.5.1	Background and Motivation	46
4.5.2	Overview of the System	47
4.5.3	DHT in CoDoNS	47
4.5.4	Evaluation	48
4.6	Case Study 2: Cooperative File System (CFS)	48
4.6.1	Background and Motivation	48
4.6.2	Overview of the System	49
4.6.3	DHT in CFS	49
4.6.4	Evaluation	49

References	50
5 Locality Sensitive Hashing.....	52
5.1 Introduction	52
5.1.1 Basic Idea of LSH	52
5.1.2 The Origin of LSH	53
5.2 Overview	53
5.2.1 The Definition	53
5.2.2 Properties of LSH	54
5.2.3 Several LSH Families	54
5.2.4 Approximate Nearest Neighbor	58
5.3 Case Study 1: Large-Scale Sequence Comparison	60
5.3.1 Theory	60
5.3.2 Algorithm Complexity	61
5.3.3 Implementation Details	61
5.3.4 Results	62
5.4 Case Study 2: Image Retrieval	62
5.4.1 Motivation	62
5.4.2 The Problems of Existing Approaches	62
5.4.3 The System	63
5.4.4 Results	63
References	63
6 XOR Operations.....	65
6.1 Introduction	65
6.2 XOR Operation	65
6.2.1 Truth Table	66
6.2.2 Set Diagrams	66
6.3 XOR Properties	66
6.4 Compress with XOR	67
6.4.1 Case Study 1: XOR-linked list	67
6.4.2 Case Study 2: XOR swap algorithm	68
6.5 Fault Tolerance	68
6.5.1 Case Study 3: Hamming (7,4) code	68
6.5.2 Hamming Codes with Additional Parity	70
6.5.3 Case Study 4: RAID	70
6.6 Case Study 5: Feistel Cipher	71
6.7 Case Study 6: Kademlia	72
6.7.1 XOR Metric in Kademlia	73
6.7.2 Routing Table in Kademlia	74
6.7.3 Kademlia Protocol	74
6.8 Conclusion	75
References	76

- 7 Adaptation 77**
 - 7.1 Introduction 77
 - 7.2 How Adaptation Works and Key Issues 78
 - 7.2.1 How Does Adaptation Work? 78
 - 7.2.2 Classification of Adaptation 80
 - 7.3 Case Studies 83
 - 7.3.1 Case Study 1: Adaption in Internet Routing System 83
 - 7.3.2 Case Study 2: Adaptive Self-Configuration for Sensor Networks 87
 - References 90
- 8 Optimistic Replication 92**
 - 8.1 Introduction 92
 - 8.2 Topic Description 94
 - 8.2.1 Design Considerations 94
 - 8.2.2 Techniques and Algorithms 95
 - 8.3 Case Studies 100
 - 8.3.1 Case Study 1: The Notes System 100
 - 8.3.2 Case Study 2: The Bayou system 102
 - References 106
- 9 Reputation and Trust 107**
 - 9.1 Introduction 107
 - 9.2 Reputation Systems: Challenges and Models 108
 - 9.2.1 Challenges 108
 - 9.2.2 Reputation Models 109
 - 9.2.3 Threat Model 113
 - 9.3 Comparison of Representative Work 114
 - 9.4 Case Studies 116
 - 9.4.1 Case Study 1: EigenTrust 117
 - 9.4.2 Case Study 2: HOURS 117
 - 9.5 Conclusion 118
 - References 118
- 10 Moving Average 120**
 - 10.1 Introduction 120
 - 10.2 Topic Description 120
 - 10.2.1 Simple Moving Average 121
 - 10.2.2 Cumulative Moving Average 121
 - 10.2.3 Weighted Moving Average 121
 - 10.2.4 Exponential Weighted Moving Average 122
 - 10.3 Case Study 1: Attacks Detection 123
 - 10.3.1 Introduction of Denial of Service Attack 123
 - 10.3.2 Anomalies Detection 124

10.3.3	SYN Flooding Detection	126
10.3.4	Other Methods	126
10.4	Case Study 2: Machine Monitoring Technique	127
10.5	Case Study 3: Data Cleaning in Wireless Sensor Networks	130
10.6	Conclusion	132
	References	132
11	Machine Learning	134
11.1	Machine Learning Concepts	134
11.1.1	Concepts and History	135
11.2	Introduction of Machine Learning	136
11.2.1	A Typical Machine Learning Problem	136
11.2.2	Machine Learning in Computer Systems Research	139
11.3	Machine Learning Techniques	140
11.3.1	Category	140
11.3.2	Machine Learning Techniques and Algorithms	142
11.4	Case Studies	145
11.4.1	Case Study 1: Large-Scale System Problem Detection	145
11.4.2	Case Study 2: Snitch	146
11.5	Conclusion	147
	References	148
 Part III Implementation		
12	Asynchronous I/O	153
12.1	Motivation	153
12.2	I/O Multiplexing	155
12.3	Asynchronous I/O	158
12.3.1	Linux Asynchronous I/O	158
12.3.2	Windows Overlapped I/O	164
12.4	Conclusion	172
	References	172
13	Multithreading	173
13.1	Background	173
13.2	The Concept of Thread	174
13.3	Hardware Support for Multithreading	175
13.3.1	Block Multithreading	175
13.3.2	Interleaved Multithreading	176
13.3.3	Simultaneous Multithreading	176
13.4	Multithreading Programming	176
13.4.1	POSIX Threads (Pthreads)	177
13.4.2	JAVA Threads	178
13.4.3	WIN32 Threads	178

- 13.4.4 Common APIs180
 - 13.5 Multithreading Synchronization.....184
 - 13.5.1 Multithreading Synchronization Problems184
 - 13.5.2 Mutual Exclusion185
 - 13.5.3 Solutions of Mutual Exclusion186
 - 13.5.4 Mutual Exclusion Cases.....187
 - 13.6 Case Studies188
 - 13.7 Conclusion.....199
 - References189
- 14 Virtualization.....191**
 - 14.1 Virtualization Definitions.....191
 - 14.2 A Brief History of Virtualization192
 - 14.2.1 The Mainframe Virtualization193
 - 14.2.2 The x86 Virtualization.....193
 - 14.3 Why Virtualization?194
 - 14.4 Virtualization Capabilities.....196
 - 14.5 The Benefits of Virtualization196
 - 14.5.1 Increasing Utilization196
 - 14.5.2 Reducing Cost197
 - 14.5.3 Isolation.....197
 - 14.5.4 Improving Application Development Process.....197
 - 14.5.5 Business Continuity198
 - 14.5.6 Manageability, Scalability and Flexibility198
 - 14.6 Types of Virtualization.....199
 - 14.7 Virtualization Vendors and Products201
 - 14.8 Case Studies201
 - 14.8.1 Case Study 1: JVM202
 - 14.8.2 Case Study 2: VirtualPower.....204
 - 14.9 Issues of Virtualization.....205
 - 14.9.1 Issues of Adopting Virtualization205
 - 14.9.2 Issues of Providing Virtualization206
 - References208

Part IV Evaluation

- 15 Queueing Theory.....213**
 - 15.1 Introduction213
 - 15.1.1 Queueing Models.....214
 - 15.2 Fundamental Concepts.....216
 - 15.2.1 Useful Probability Distributions216
 - 15.2.2 Markov Chain.....218
 - 15.3 Queueing Systems219

15.3.1	Markovian Queues	220
15.3.2	Non-Markovian Queues	224
15.4	Queueing Networks	226
15.5	Case Studies	227
15.5.1	Case Study 1: Telephone Systems	227
15.5.2	Case Study 2: A Barber Shop	227
References		229
16	Black Box Testing	230
16.1	Introduction	230
16.2	Black Box Testing Techniques	231
16.2.1	Equivalence Partitioning	231
16.2.2	Boundary Value Analysis	232
16.2.3	Decision Table Testing	232
16.2.4	Pairwise Testing	233
16.2.5	State Transition Tables	233
16.2.6	Use Case Testing	234
16.3	Other Methods of Software Testing	234
16.4	Case Studies	235
16.4.1	Case Study 1: Web Services	235
16.4.2	Case Study 2: MobileTest	239
References		242
17	Goodness-of-Fit	244
17.1	Introduction	244
17.2	General Topics in Goodness-of-Fit	245
17.2.1	Hypothesis Testing	246
17.2.2	Definition	247
17.2.3	Common Problems in Goodness-of-Fit Tests	247
17.2.4	Quantitative Goodness-of-fit Techniques	249
17.3	Chi-Square Test	249
17.3.1	Meaning of the Chi-Square Test	250
17.3.2	Definition of Chi-Square	251
17.4	Kolmogorov-Smirnov test	254
17.4.1	How Does K-S Test Work?	255
17.4.2	Comparison of Chi-Square and Kolmogorov-Smirnov Tests	261
17.5	Case Studies	262
17.5.1	Case Study 1: Object Characteristics of Dynamic Web Content	262
17.5.2	Case Study 2: Failures in High-Performance Computing Systems	263
References		263
Index		265

Part I

General

Part I consists of two chapters. Chapter 1 describes the basic elements of computer systems research, including most common mistakes in English writing, and Chapter 2 lists 12 rules of thumb that are widely used in systems design.

Chapter 1

Elements

Weisong Shi

Abstract The content of this chapter lays the foundations for computer systems researchers. If you are an experienced researcher in this area, you can skip the chapter and move directly to the next chapter. The first thing for a beginner is to identify the top conferences/journals in his or her specific research area. Hence, the 20 top conferences and the 10 top journals in the computer systems area are listed in the first part of the chapter. You might want to customize them a little bit to fit your own interests. Next, three “how tos” are described respectively: *how to read a research paper*, *how to write a research paper*, and *how to present a research paper*. Finally, the chapter is concluded by pointing interested readers to an important book about the ethical foundations of scientific practices.

1.1 Top Systems Conferences/Journals

In this section, the 20 top systems or systems-related conferences and the 10 top journals in computer systems are listed based on the author’s personal research experience; and it is by no means a complete list. Note that among these 20 conferences, the first half is more systems-oriented, while the second half has its own specific interests, although they do have a couple of sections focusing on systems-related topics. Note that the list intends to give beginners a point to start, rather than an authoritative list. As a matter of fact, you are encouraged to work out a customized list based on your own research interests.

It is worth noting that at the time this book was written (2009), top conferences were more prestigious than journals in the computer systems community. Therefore, the 20 top systems conferences are listed first. In the long run, however, the systems community is called on to pay more attention to journals than conferences in the future because of the following two reasons. One is that the authors of a conference paper have little time (usually less than a month between the notification date and the camera ready date) to revise the paper according to review comments. Therefore, most of results published in conference proceedings are premature work. On the contrary, journal papers are archived documents, and authors have enough time to revise the paper and address the comments. The second reason is related to the development of computer science as a discipline. To the best of my knowledge, many other disciplines pay more attention to journals other than conferences. The differences in evaluation (i.e., conferences vs. journals) that exist between computer

Weisong Shi
Wayne State University, Detroit, MI 48202, USA. e-mail: weisong@wayne.edu