



Insight into

DBMS:
Design and Implementation

Chapter 1: Introduction



mlinking@126.com
+86 15010255486

Chapter 1: Introduction

- About me**
- Topics and Wishes**
- Examination & Resources**

□ 孔令波

□ You can find me here:

Yifu Building, West Wing **809**

School of Software Engineering,

Beijing JiaoTong University (BJTU),

Hai Dian District

Beijing 100044, China

□ Or email me:

➤ mlinking@126.com

□ Tel:

➤ 15010255486



BaiDu Pan now

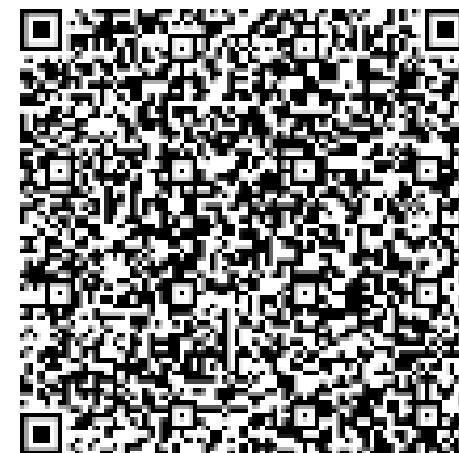


<input type="checkbox"/> 文件名	修改时间	类型	大小
<input type="checkbox"/> 03 Videos	2021-07-12 20:06	文件夹	-
<input type="checkbox"/> 01 PPTs(PDF)	2021-07-12 20:06	文件夹	-
<input type="checkbox"/> 02 Teaching Materials	2021-07-12 20:06	文件夹	-
<input type="checkbox"/> !!此处资料仅供内部使用,请勿外传,呵呵.txt	2021-09-07 15:32	txt文件	0B
<input type="checkbox"/> !!Materials here are only used internally, thanks!.txt	2021-09-07 15:32	txt文件	0B

数据库的那点事

(1/200)

暂无群简介



打开百度网盘，扫码加入该群



Chapter 1: Introduction

- About me**
- Topics and Wishes**
- Examination & Resources**

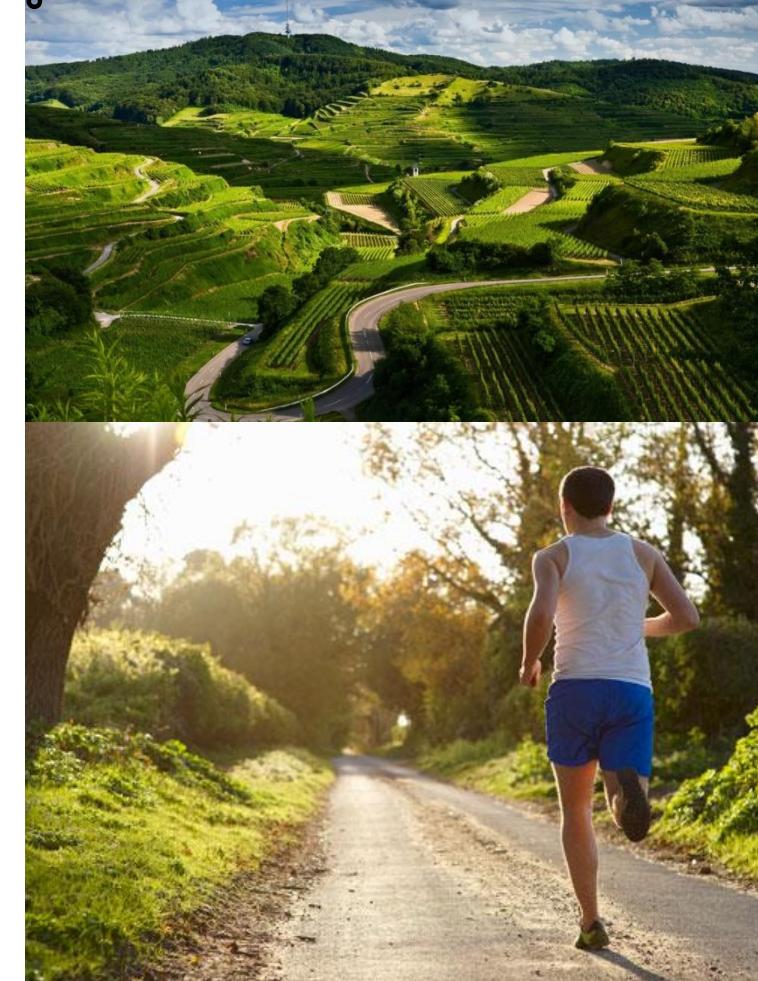
Do you know the top goals of this S.S.E for you?

□ BECOME MORE CIVILIZED!

- Polite, enjoying nature, good habits, . . .



你们知道我们对你们的期望吗?



Do you know the top goals of this S.S.E?

你们知道我们对你们的期望吗？

□ SHOWING YOU SOME LIFE-MAKING SKILLS (PROGRAMMING IS ONE)!

4th

Programming in IT companies

Intern practice
- Try yourself in companies

3rd

BI/Data Analytics, SE, BA, Frontier topics, ...

Traditionally, there are 3 classic softwares – OS, DBMS, Compiler

2nd

DSA, **OS**, **DBMS**, Web program

1st

Math, Programming (with C), OO (Java)

Fundamental programming

Learning from classics!

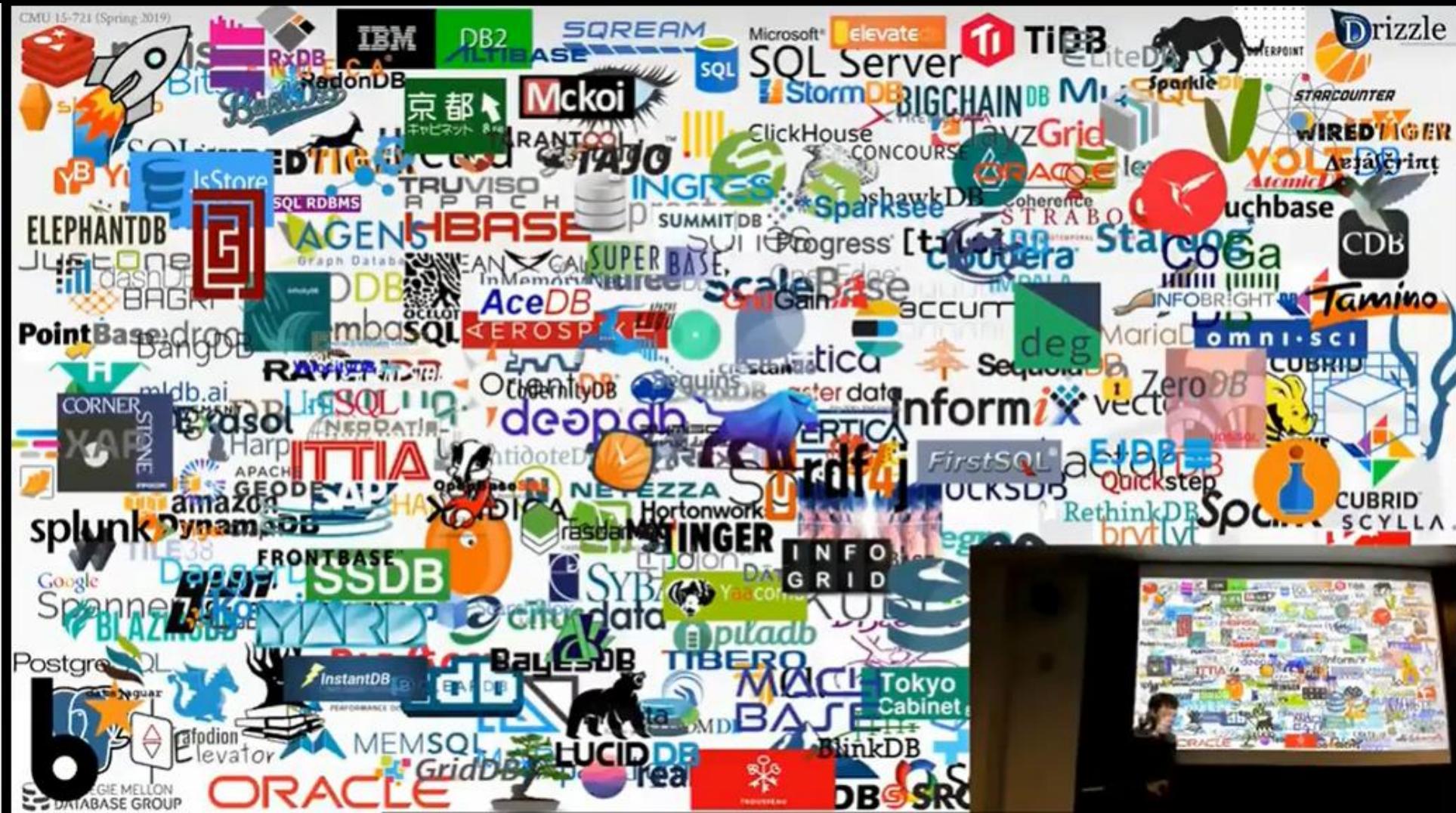
- OS is one of the three classic softwares/programs for Computer Science

- **DBMS** (Database Management System)

- An important software to ease the management of the mass data for users



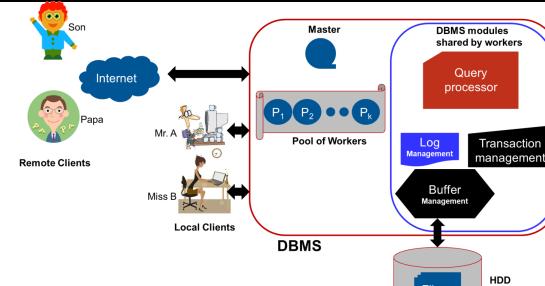
A lot of DBMS now



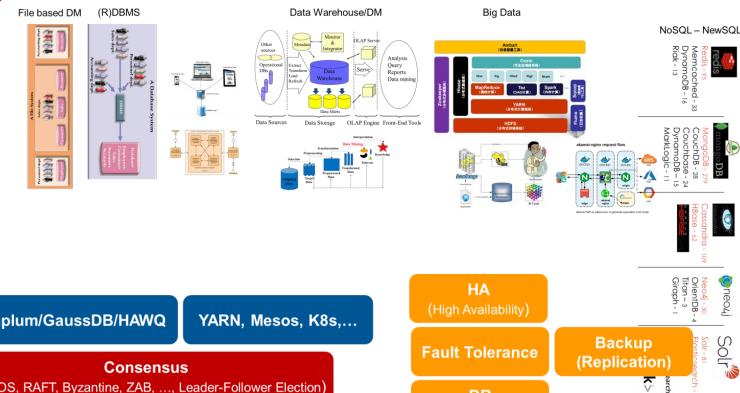
Topics

□ Topics

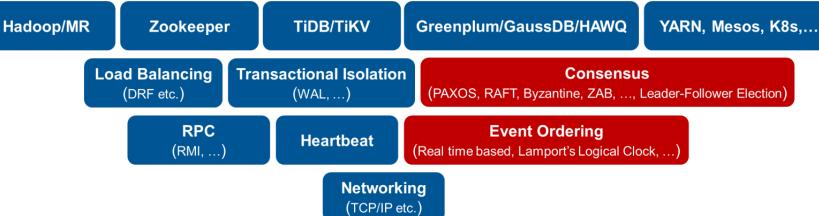
■ Know the Internals of (R)DBMS



■ Know the history of Data Management techniques



■ Know the Distributed Data System



□ Wishes

■ Know what they are talking about

➤ Not just the Web-based Development

✓ Front end – VUE, **Springs**, MVC ... or Micro Service, Clouding, DBA, ...

➤ But also Internals

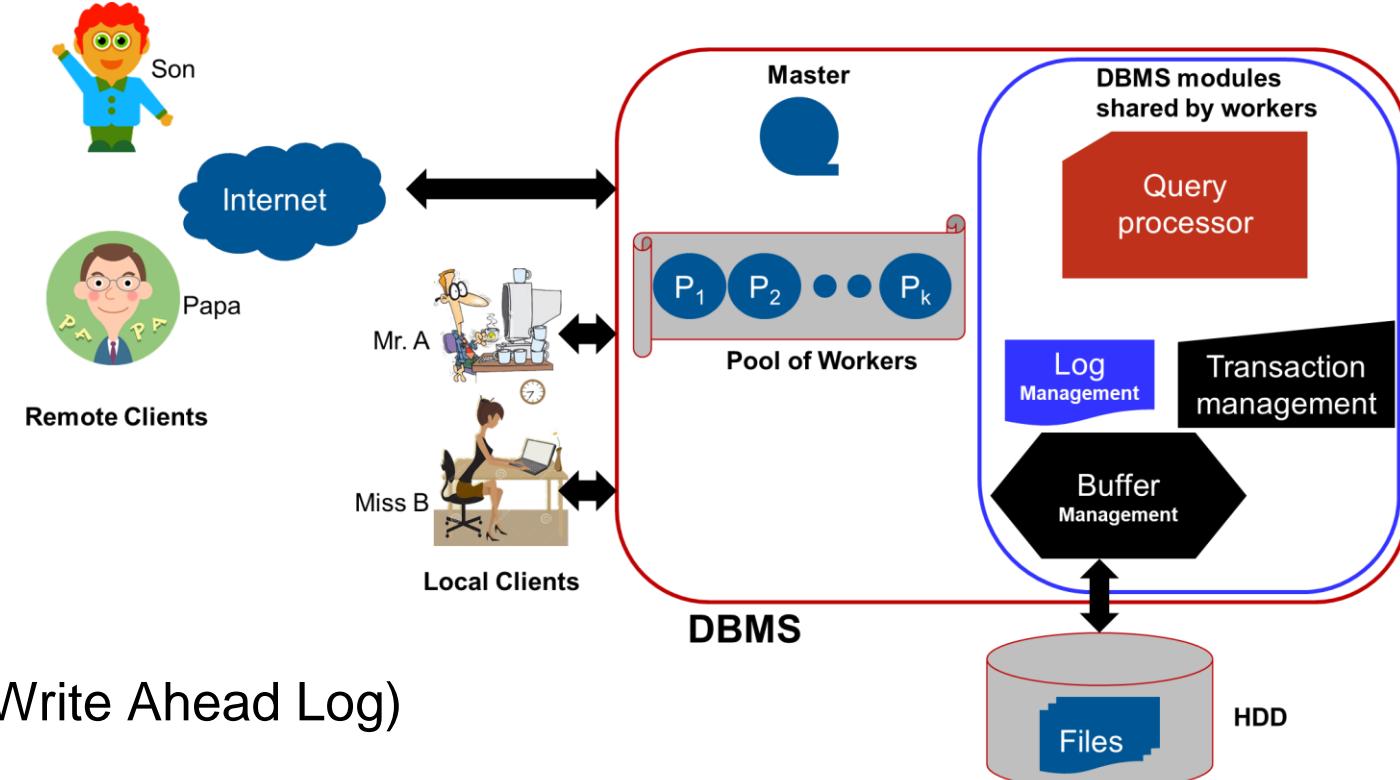
✓ IPC – Instruction per cycle, Pipelining, Hot backup, **PAXOS/RAFT/ZAB**, Logic Clock, ...

Topics

□ Topics

■ Know the Internals of (R)DBMS

- Respond many users
 - ✓ Multi-processed/Threaded
- Friendly interaction
 - ✓ SQL – Translate & Execute
- Data Consistency
 - ✓ Transaction Control + WAL (Write Ahead Log)

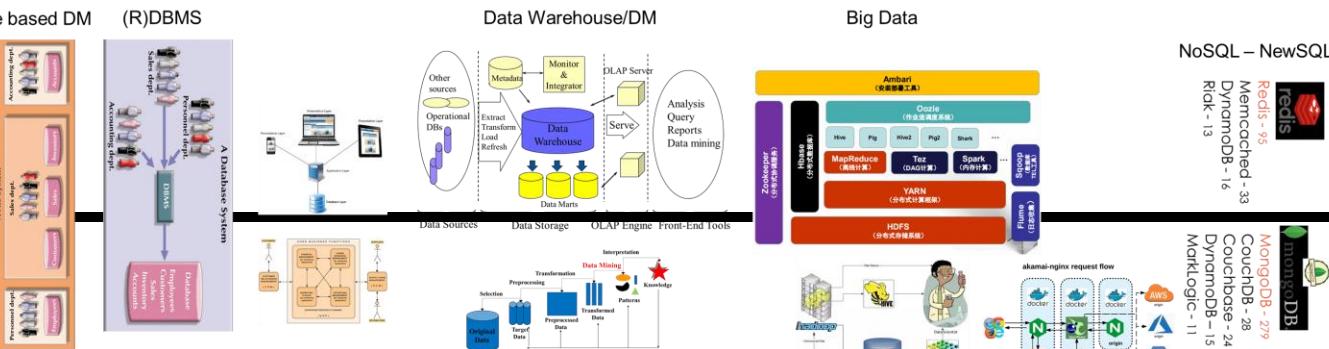
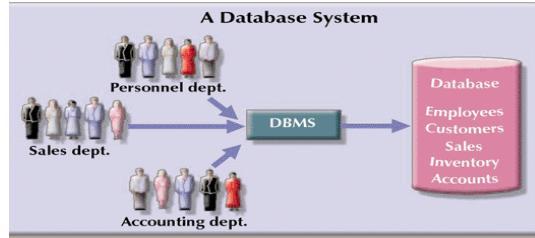


Topics

Topics

Know the history of Data Management techniques

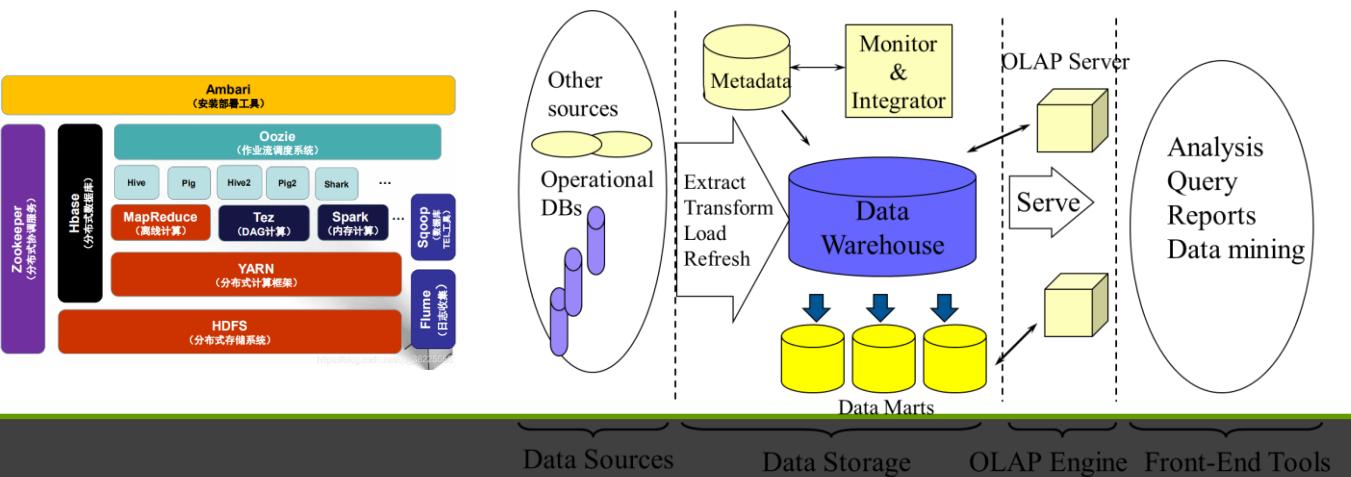
➤ File based data management



➤ RDBMS



➤ Data Warehouse + Data Mining (Statistics, Machine Learning, AI)

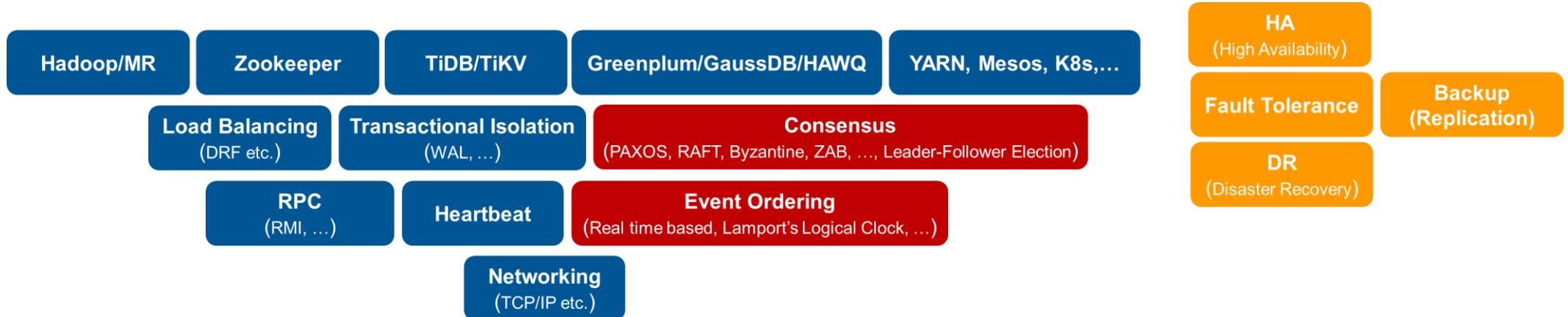


➤ Big Data (NoSQL, NewSQL)

Topics

□ Topics

■ Know the Distributed Database System



Topics

□ Wishes

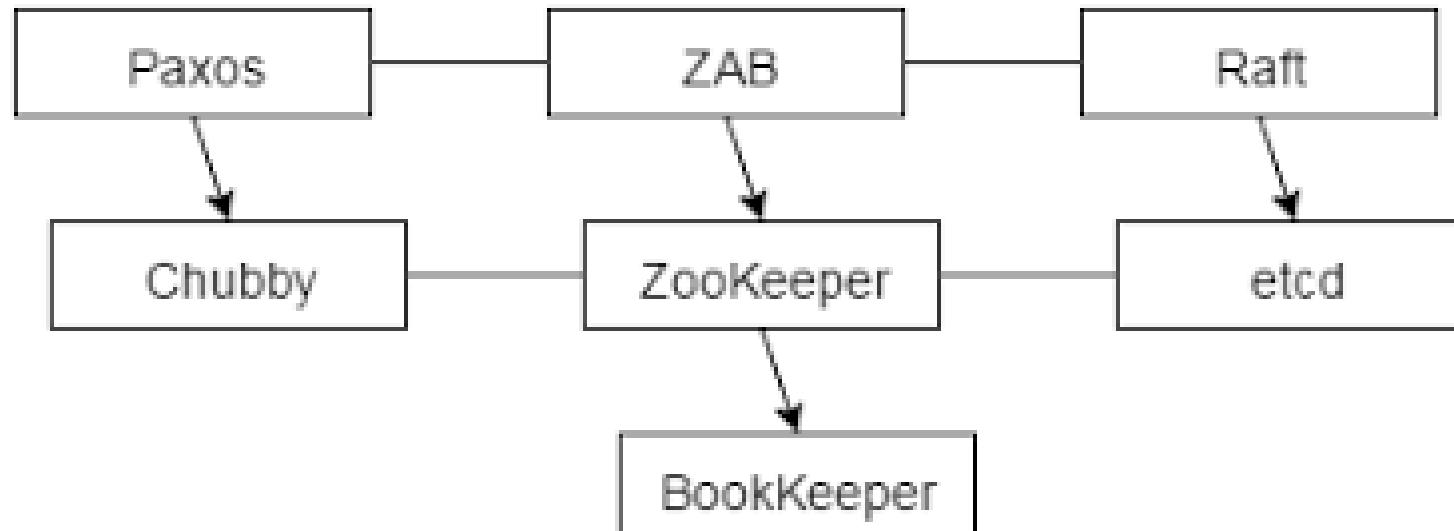
■ Know what they are talking about

➤ Not just the Web-based Development

✓ Front end – VUE, **Springs**, MVC ... or Micro Service, Clouding, DBA, ...

➤ But also Internals

✓ IPC – Instruction per cycle, Pipelining, Hot backup, **PAXOS/RAFT/ZAB**, Logic Clock, ...



□ 偶数科技主导研发的HAWQ获中日韩开源软件技术优胜奖



软件巨头纷收购，开源意义再彰显
上月，IBM宣布将计划斥资**340亿美元**收购开源
解决方案供应商红帽，如此大手笔的收购计划不
仅创下了历史新高，同时也成为了近年来开源公
司收购的里程碑事件。

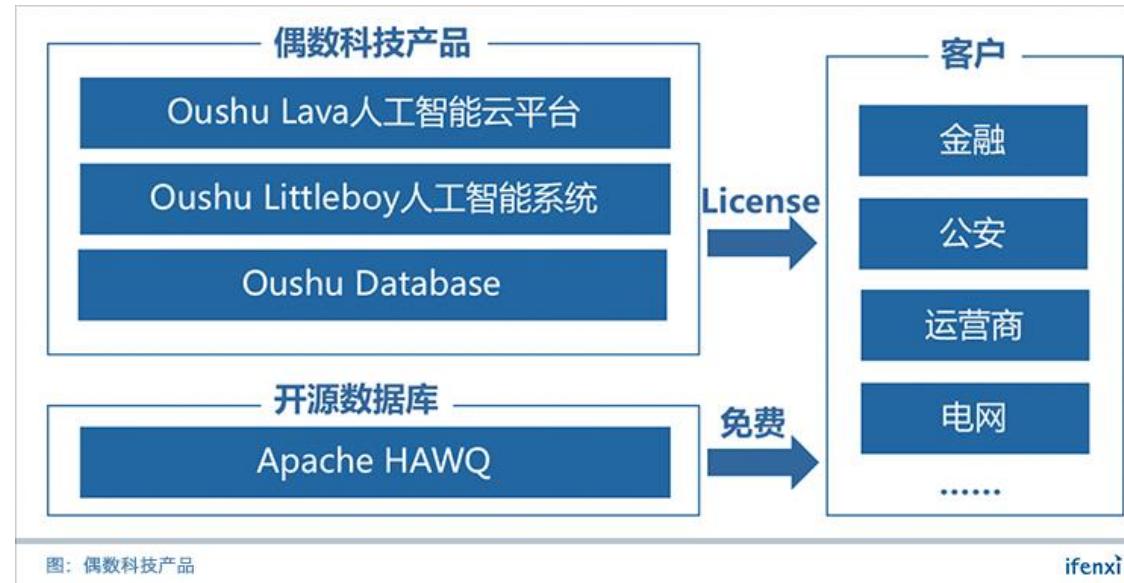
无独有偶，微软以**15亿美元**收购了GitHub、
Salesforce以**6.5亿美元**收购了**MuleSoft**、**EQT**
以**2.5亿美金**收购了**SUSE**，所有这些让越来越
多的人意识到开源不仅仅是全球互联网生态的重
要一部分，同时也隐藏着巨大的商业价值。

版权声明：本文为CSDN博主「偶数科技」的原
创文章，遵循CC 4.0 by-sa版权协议，转载请附
上原文出处链接及本声明。

原文链接：

<https://blog.csdn.net/oushukeji/article/details/84253580> 2018年11月19日 15:55:04 偶数科技

2018年，偶数科技推出了Oushu Littleboy人工智能系统和Oushu Lava人工智能云平台，开始切入公有云以及AI市场。



传统的数据仓库中通常会内置In-Database机器学习库，但对于使用者的AI知识水平要求较高。因此，为了降低使用门槛，偶数科技推出了Littleboy人工智能系统，由于使用了AutoML技术，可以帮助使用者自动选择模型并调参，完成AI模型的落地。

为方便企业使用，偶数科技还和微软合作推出了Lava人工智能云平台，集成了OushuDB和Littleboy两大模块，为企业提供更加易于管理的云数据仓库服务，减少了企业在大数据量下运营维护以及硬件的开支。



2017 杭州·云栖大会

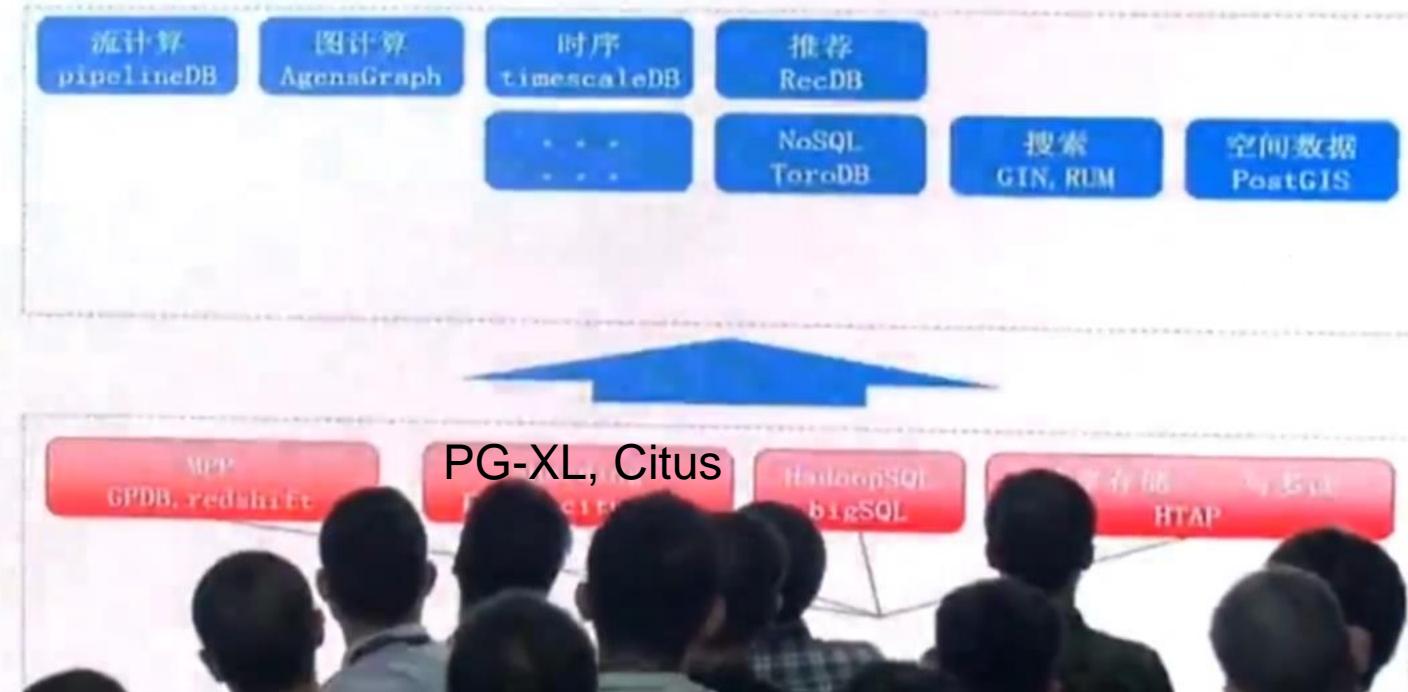
THE COMPUTING CONFERENCE

2017 杭州·云栖大会
THE COMPUTING CONFERENCE

飞天 / 智能
APSLA INTELLIGENCE

智能

TELLIGENCE





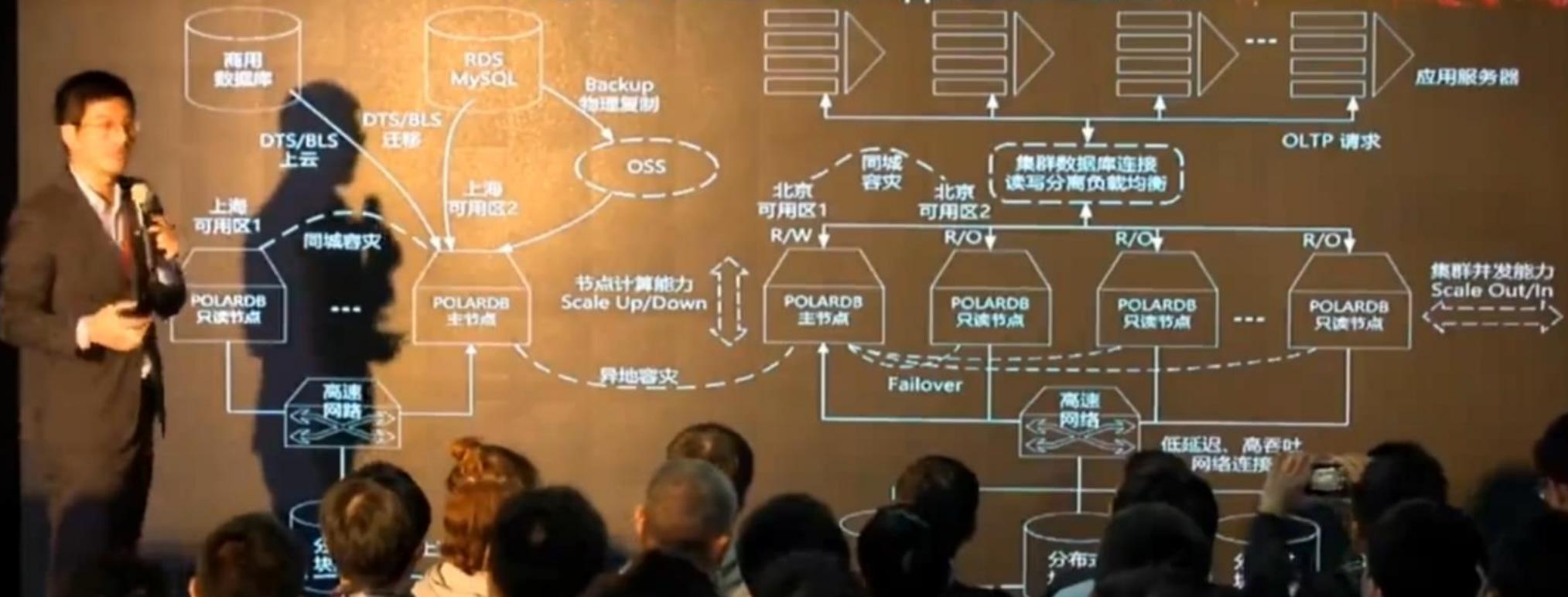
2017 杭州·云栖大会

THE COMPUTING CONFERENCE

飞天 智能
TAIJI INTELLIGENCE

POLARDB 赋能多场景应用

POLARDB Enables Multi-Application Scenarios



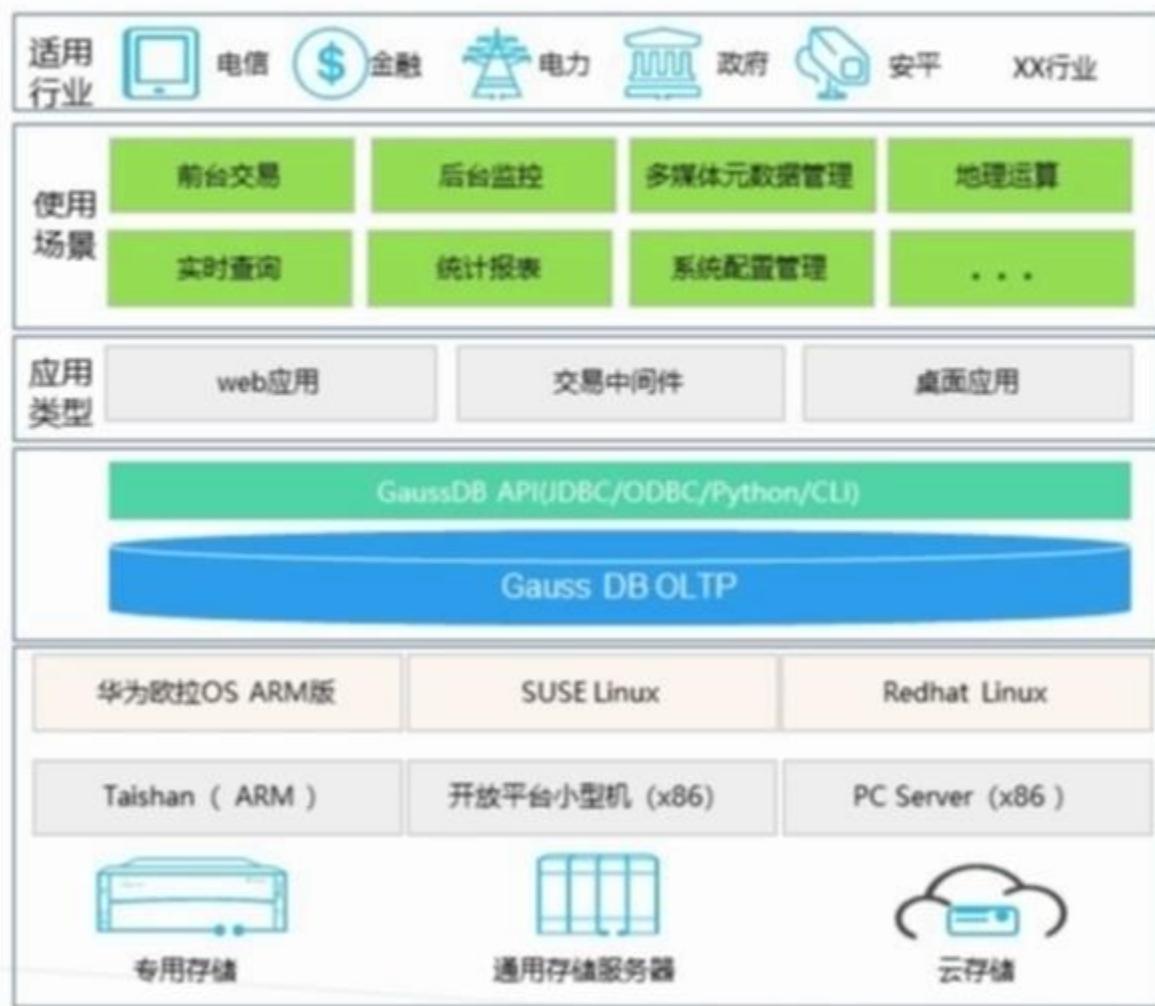
2017 杭州·云栖大会
THE COMPUTING CONFERENCE



Two valuable challenges – HUGE concurrency



GaussDB T逻辑架构



极致性能

- 高并发：单机百万tpmc, TPCC领先商业数据库 30%，是MySQL的5-8倍；
- 低延时：复杂查询能力，支持复杂企业负载；
- 高扩展：性能准线性扩展，扩展比大于0.8；



安全可靠

- 高可靠：支持双机冷热备份、两地三中心多种保护方式；
- 高安全：支持数据闪回和回收站；



简单易用

- 易开发：兼容SQL2003标准；
- 易运维：端到端异构数据迁移方案，图形化的开发调测工具Data Studio和性能调优工具集。

Chapter 1: Introduction

- About me**
- Topics and Wishes**
- Examination & Resources**

The way this semester to evaluate 100 pts based on

■ Basic (70)

1. Index File - 10
2. Simulate multiple users – 10 [multi-process] in PostgreSQL
3. Try a Syntax parsing example - 20
4. Lock table management - 20
5. Debug PostgreSQL/Greenplum/HAWQ - 10

■ 1 Web-based Application project (20)

- Java/JSP/PHP/Python/R + PostgreSQL/Greenplum/HAWQ - R.D.I.P

■ Advanced (10) – Understanding (?) in PostgreSQL/Greenplum/HAWQ

- [SQL execution][transaction mechanism(at least 2PL)]

Evaluation

□ Based on

■ Accomplishment degree [完成的程度]

- On time
- Code, Document
- Name your compressed file as “**yourID-Prjx.rar**” (containing at least code, PPT/document – or some valuable materials you find) and upload it into FTP
 - ✓ 0001-Prj01.rar

■ Presentation

- Q&A – average level (not just the best presenters)

■ Advanced skills

- If you use some interesting skills, grade goes higher

Scoring

80 pts if you could implement the required function points

- For example, if you are required to change the prompt symbol from “>” to “lbkong>”, I will check it in your code
- Document is clearly described
 - How do you carry out the practice? – development tools, configuration, etc.
 - Demonstration of Critical codes
 - ...
- Presentation is well organized and elaborated

More pts with better skills

- More efficient coding, clearer description, better presentation and documenting

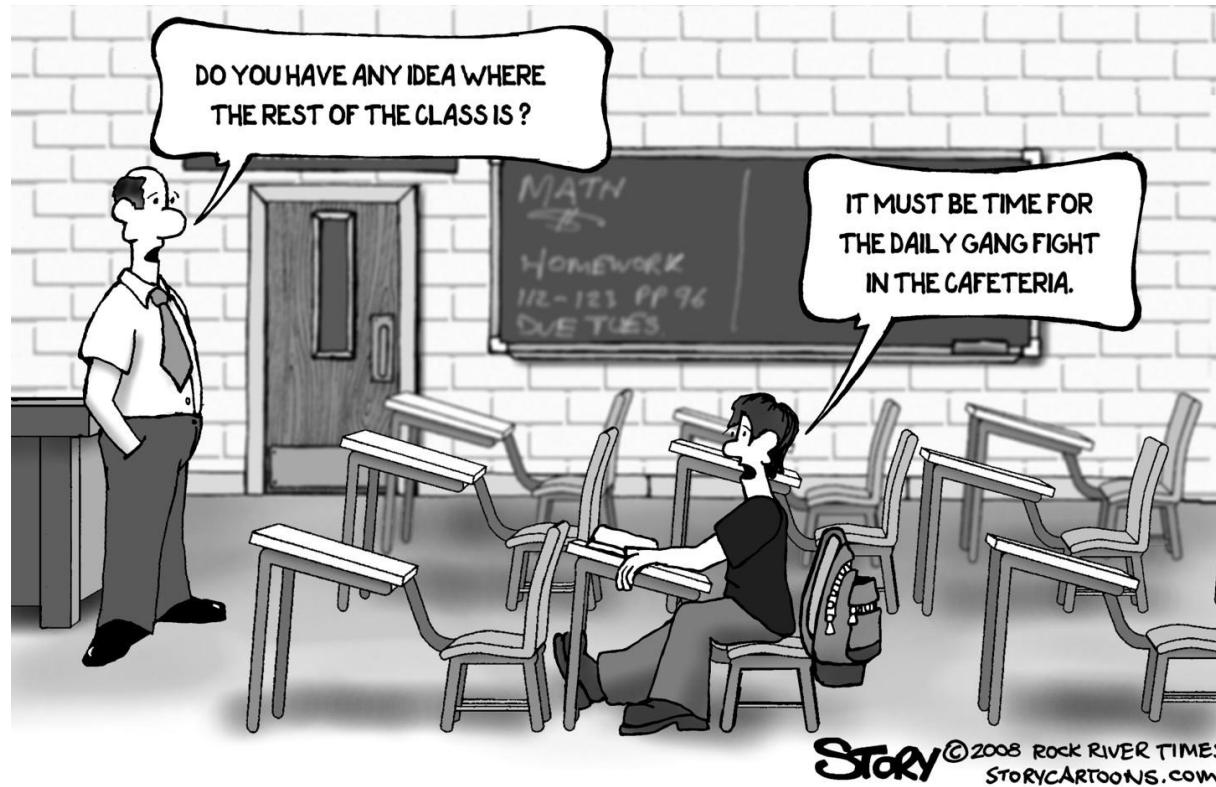
Outline of the chapters covered in this course

- **Introduction**
- **Overview of the projects**
- **Demonstration of Development environment**
 - Watch and practice by yourself
- **My understanding about (R)DBMS**
 - History and D&I
- **SQL translation with 2 conversions**
 - SQL → RA (Relational Algebra)
 - RA → Sequence of File operations
- **Transaction control**
- **Deeper**
 - File, (R)DBMS, ERP, DW, Big Data (No SQL, SQL again)
 - SQL on MPP and Hadoop (Greenplum, **HAWQ**)



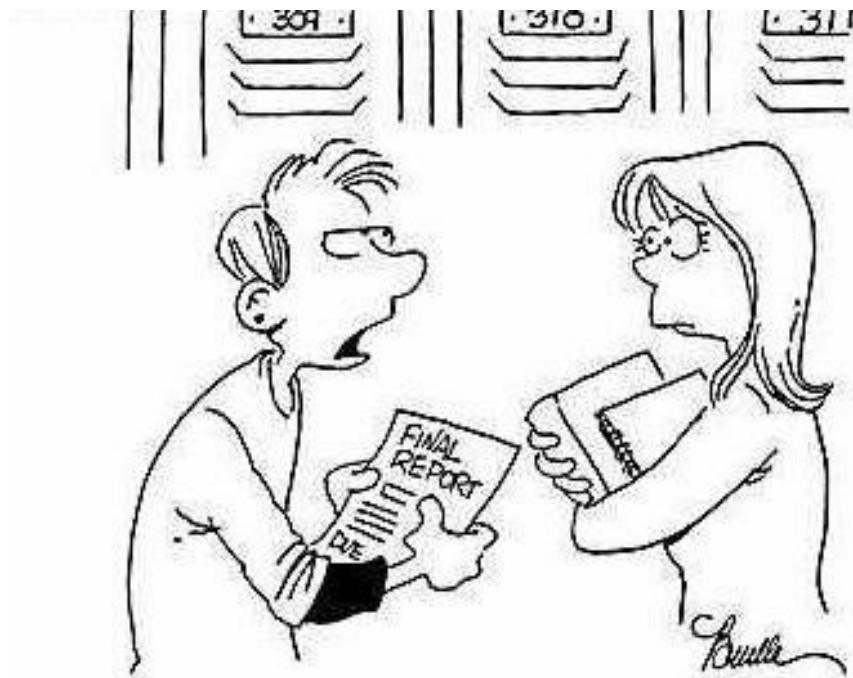
Two rules

Come to classes – mainly check your work in classes



YOUR RESPONSIBILITIES!

- Hand in your **assignments** on time
- Uphold** academic **honesty**



"I don't know what plagiarizing is, so I'm gonna take the easy way out and just copy something off the internet."

NO CHEATING!

Even I want to try **HELP-EACHOTHER** style

□ By “**HELP-EACHOTHER** style”

■ You'd better first try to cope with the projects by yourself!

➤ This is the only way you can improve yourself – no matter in study or life. “**勇敢面对, 才能真正成长!**”

➤ You can ask help from your classmates if you are in trouble with the projects.

■ It's even OK for you to “**borrow**” some codes from your classmates if you ADMIT “**Yes, I have tried everything by myself**”

■ But, **ATTENTION!** You should really understand the ideas and skills used to implement the projects!

➤ You should take the risk to persuade me by answering **MANY** questions!



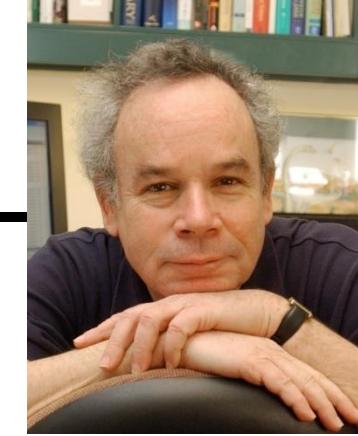
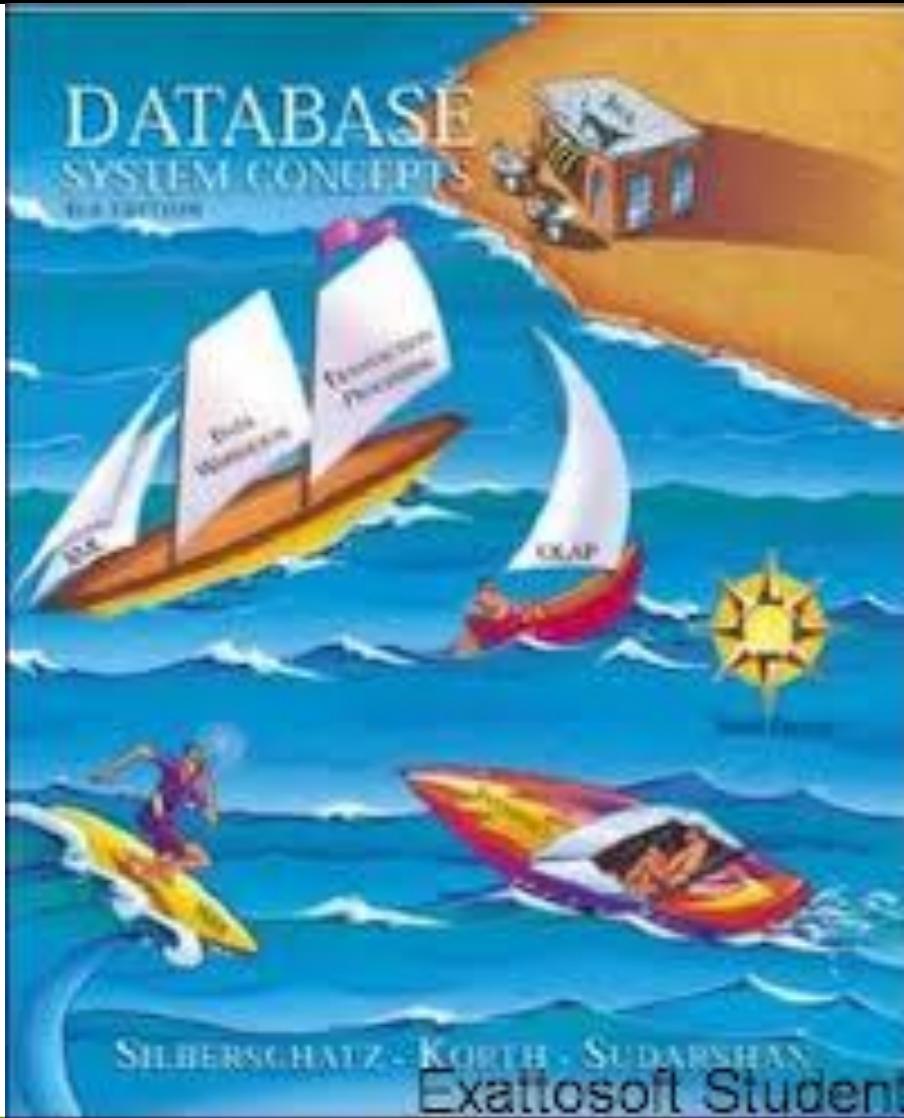
If you behave well, you could enjoy the study and life here ☺!



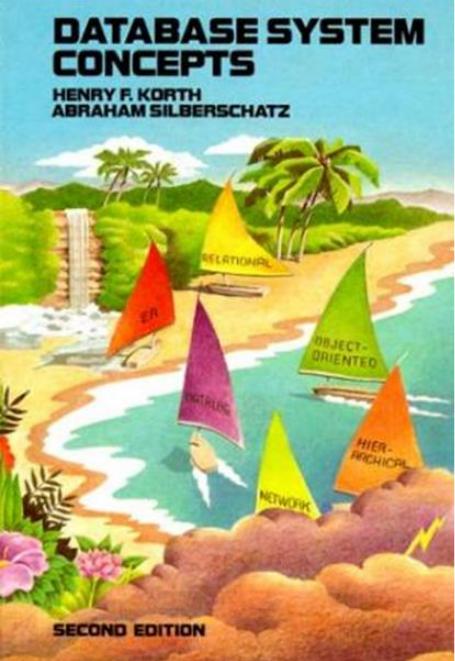
Chapter 1: Introduction

- About me**
- Topics and Wishes**
- Examination & Resources**

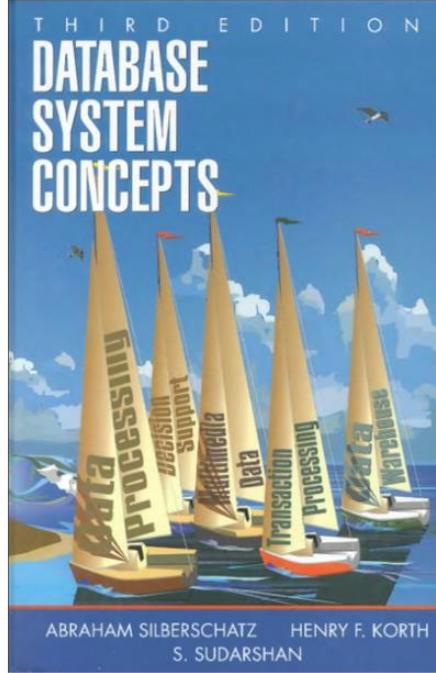
Silberschatz again!



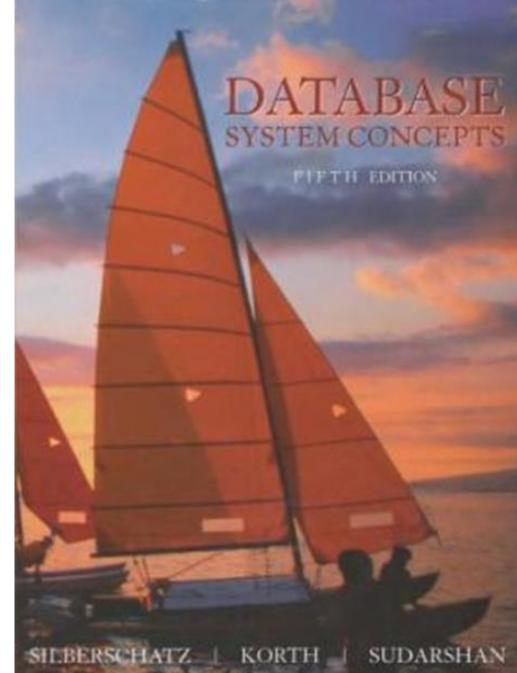
- Avi Silberschatz
- Henry F. Korth
- S. Sudarshan, “Database System Concepts”,
6th Edition,
- McGraw-Hill Education
- 2010



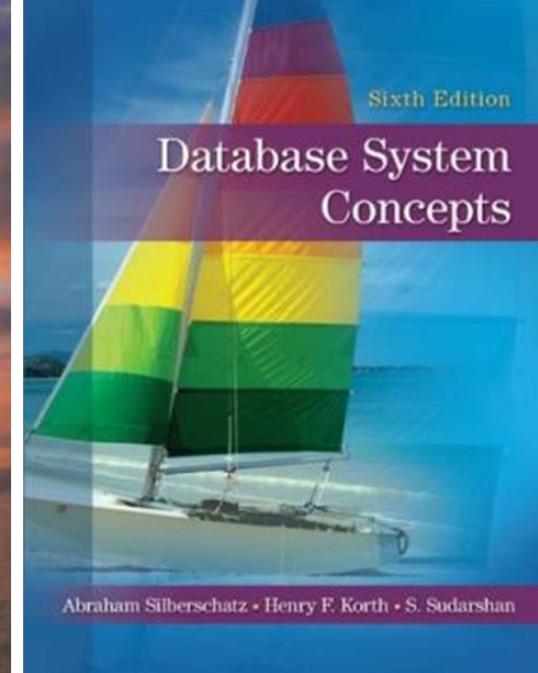
1991 2nd



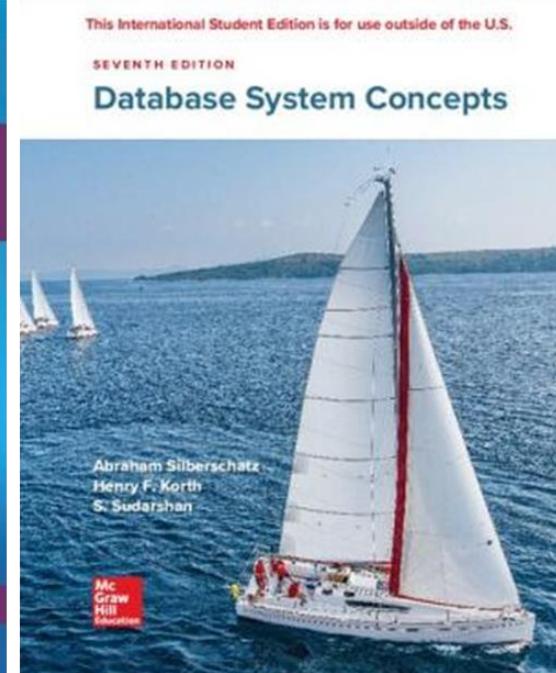
1999 3rd



2006 5th

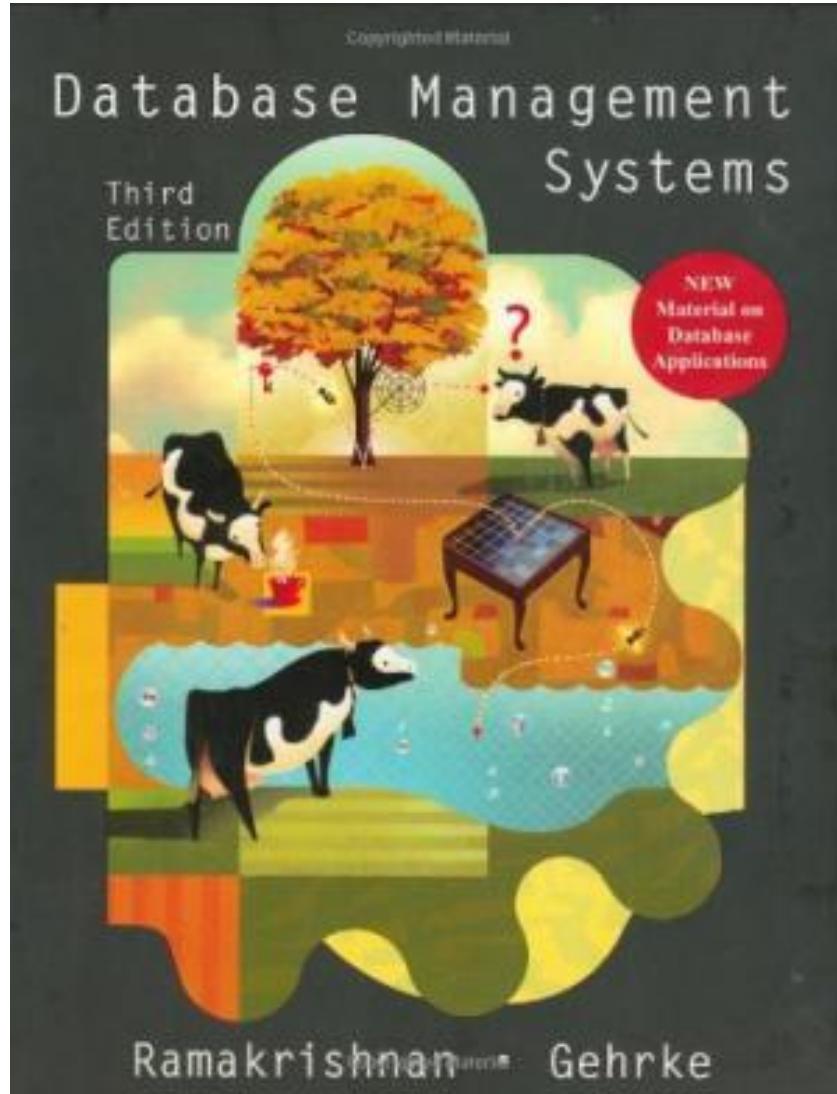


2010 6th

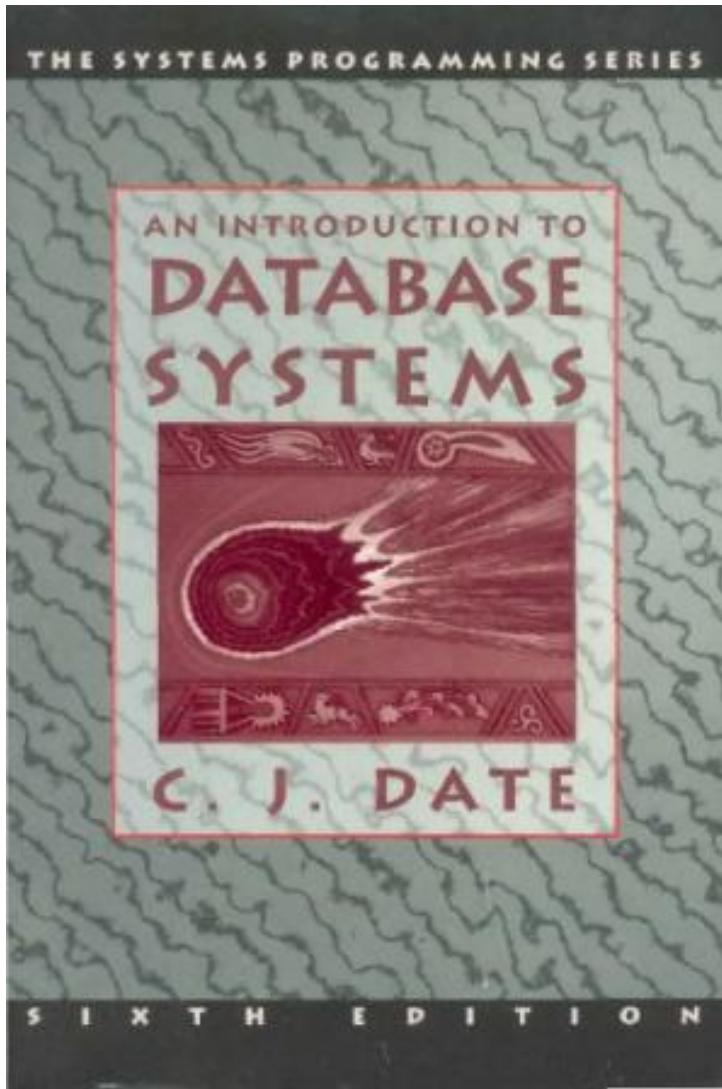


2020 7th

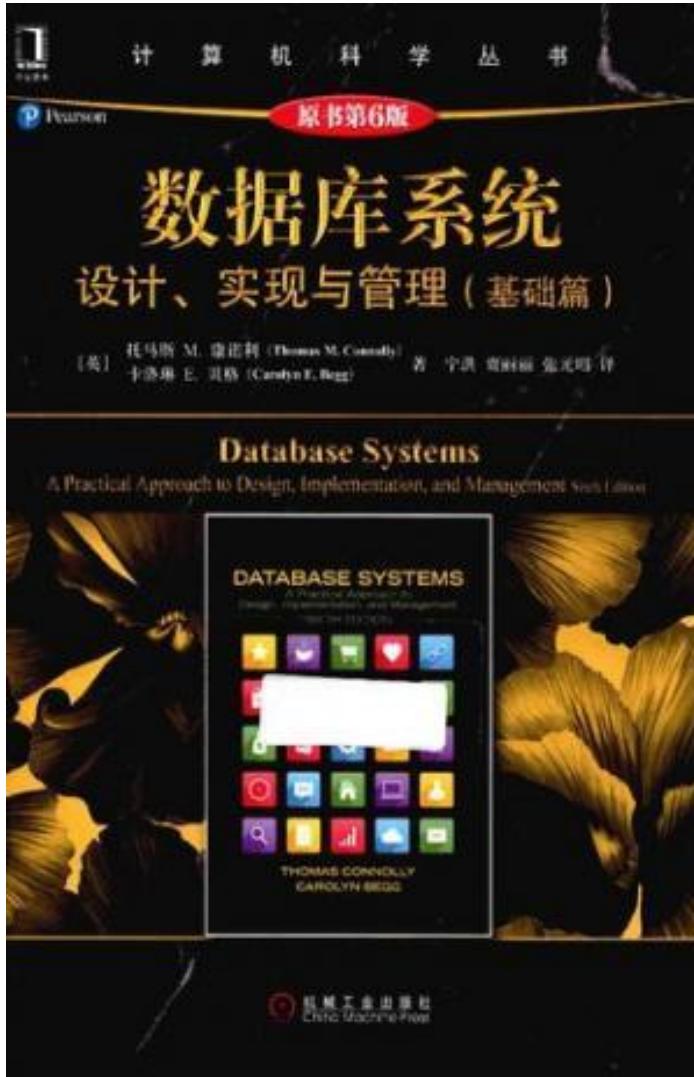
The COW book (数据库的奶牛书)



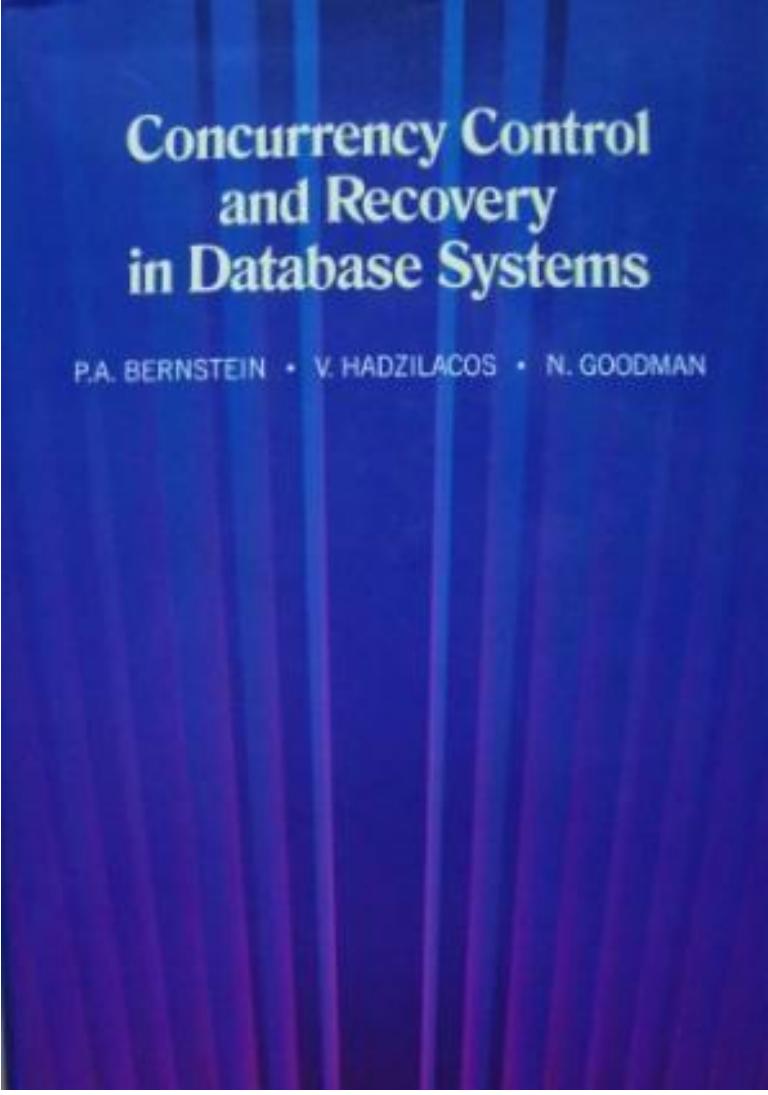
- Database Management Systems
- Raghu Ramakrishnan, Johannes Gehrke
- 2003
- McGraw-Hill Science/Engineering/Math



- An Introduction to Database Systems (6th)
- Date C.J.
- 1994



- 数据库系统: 设计、实现与管理
- 托马斯 M. 康诺利 (*Thomas M. Connolly*), 卡洛琳 E. 贝格 (*Carolyn E. Begg*)
- 2016



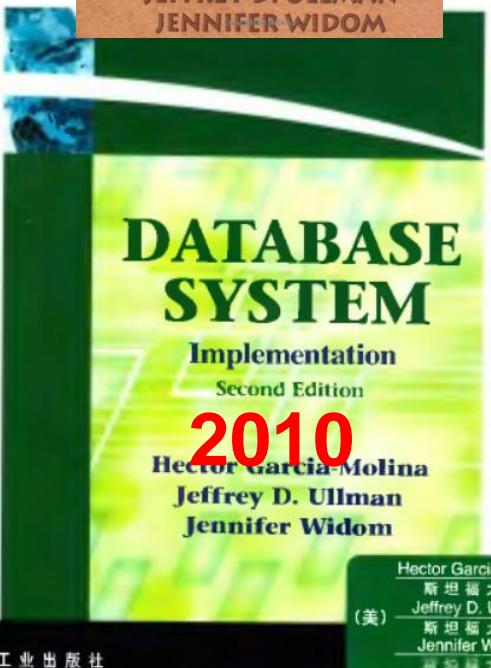
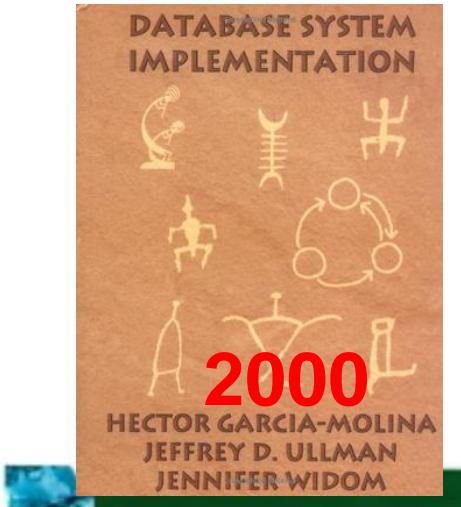
Concurrency Control and Recovery in Database Systems

P.A. BERNSTEIN • V. HADZILACOS • N. GOODMAN

- Concurrency Control and Recovery in Database Systems
- Philip A. Bernstein, Vassos Hadzilacos, Nathan Goodman
- 1987



This one is better!



□ Database System Implementation (2nd)

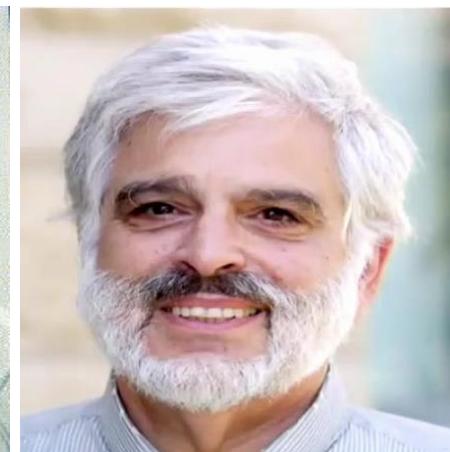
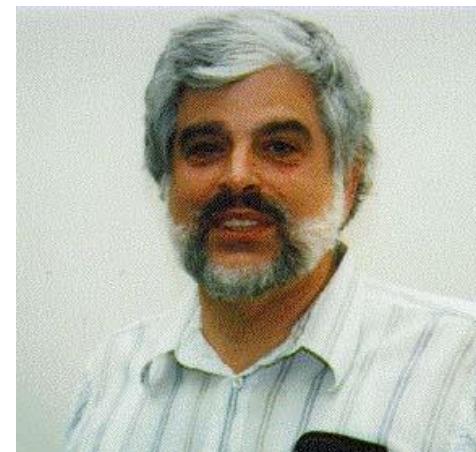
□ all of Stanford University

- Garcia-Molina, Jeffrey

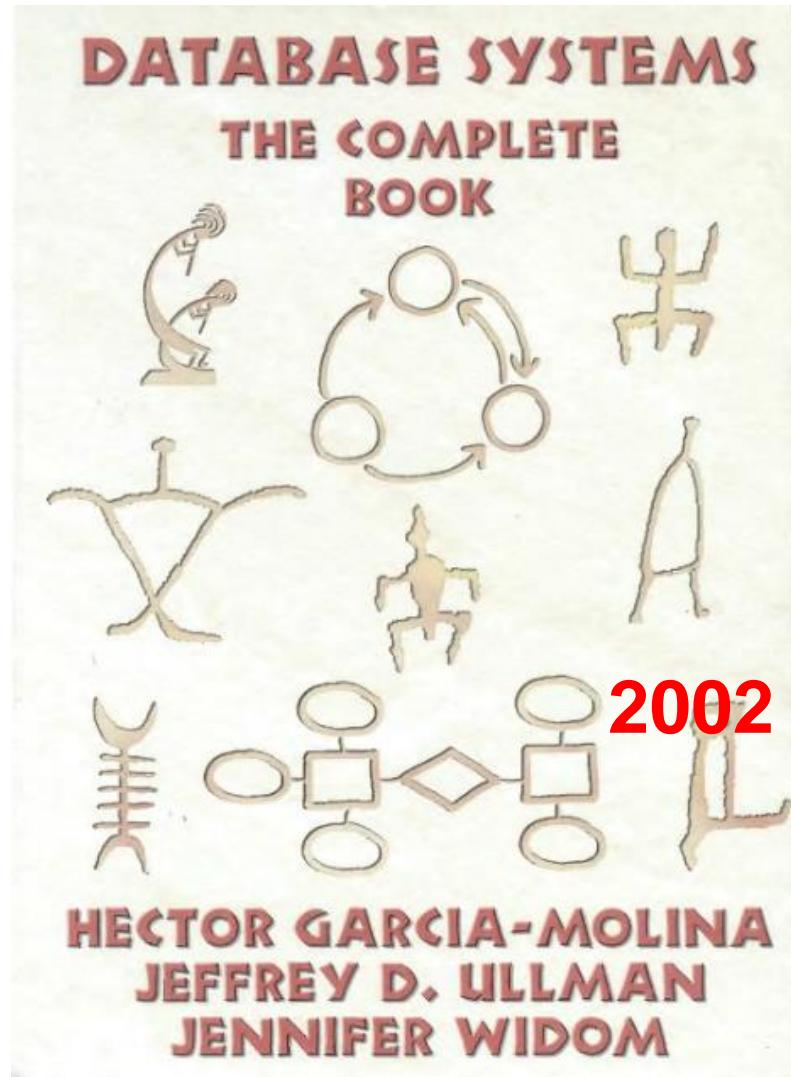
□ TOC

- Record and Block Organization
- Indexing & B trees + Hashing and other
- Query processing
- Crash Recovery
- Concurrency control
- Transactions
- Warehousing
- Distributed Databases
- BigTable, HBASE, Cassandra

It's interesting to know Ullman's book list, and you'll find it's valuable for you to know the D&I of DBMS

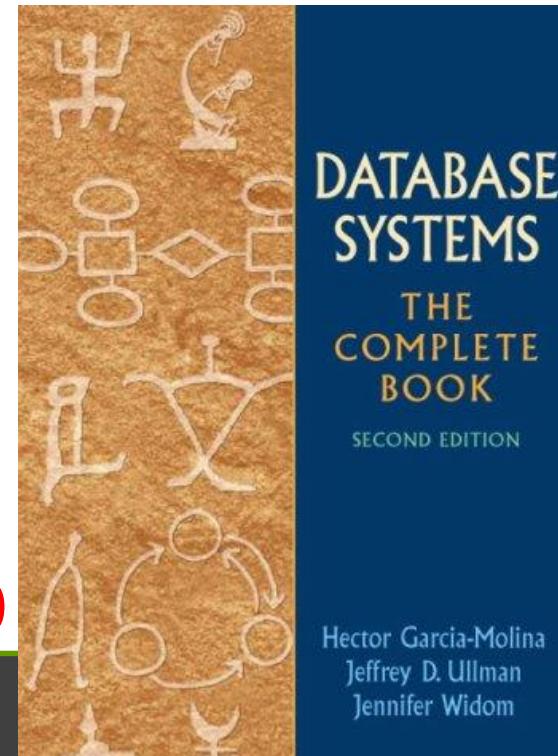


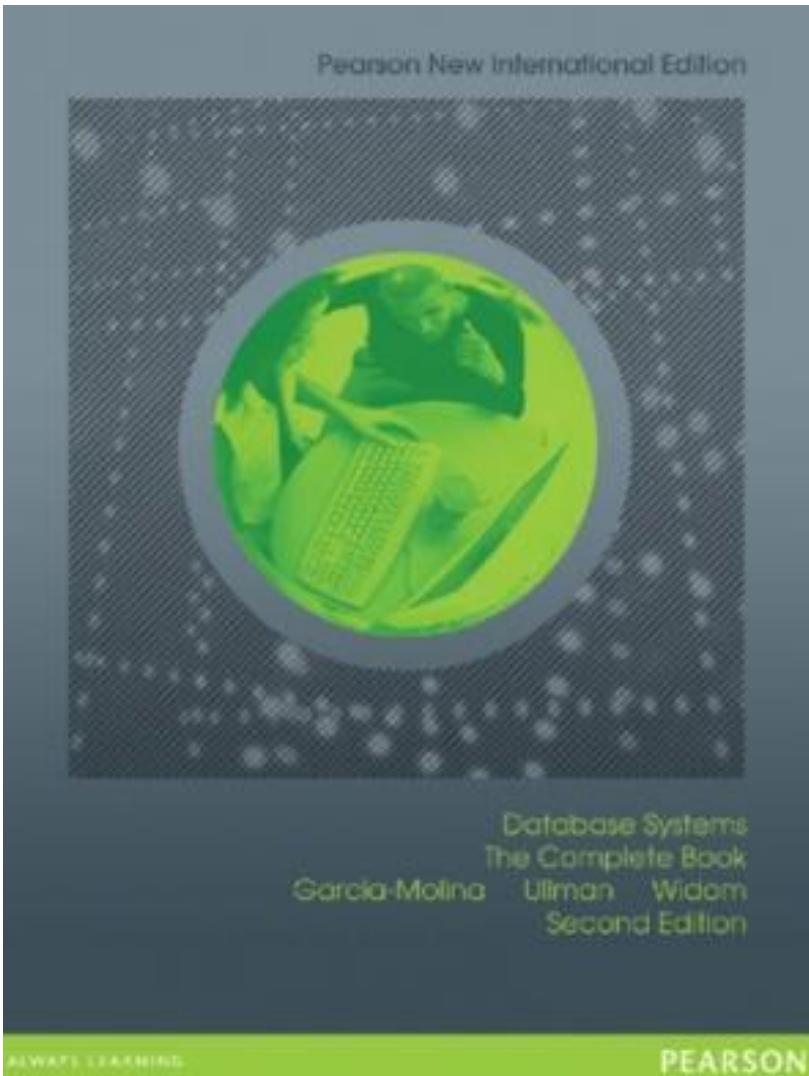
A more ambitious book



- Database System
The Complete Book
- all of Stanford University
 - Garcia-Molina, Jeffrey D.
Ullman, Widom

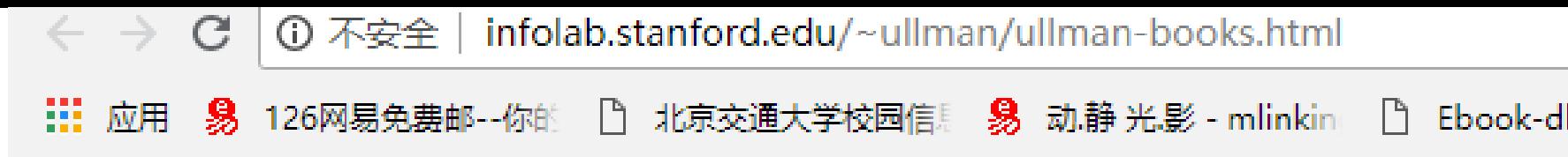
2009





- Database systems: the complete book
- (Int'l Edition)
- Pearson., Garcia-Molina, Hector, Ullman, Jeffrey D., Widom, Jennifer
- 2014

Interesting to know Ullman's book list



Contents

<http://infolab.stanford.edu/~ullman/ullman-books.html>

- [Introduction to Automata and Language Theory](#)
- [Database Systems: The Complete Book](#)
- [A First Course in Database Systems](#)
- [Database System Implementation](#)
- [Elements of ML Programming](#)
- [Foundations of CS/Pascal Edition](#)
- [Foundations of CS/C Edition](#)
- [Principles of Database and Knowledge-Base Systems](#)
- [Compilers: Principles, Techniques, and Tools \(aka "The Dragon Book"\)](#)
- [Future Book on Data Mining](#)

Google 厄尔曼 21st century compilers

全部 新闻 图片 视频 更多 设置 工具 已启用安全搜索

厄尔曼 › 图书 › 21st Century Compilers

The grid displays 14 book covers:

- 编译原理 (1986年)
- Introduction to Automata Theory, Languages, and Computation (1979年)
- 数据库系统全书 (2001年)
- 数据库系统基础教程 (1997年)
- Principles of Database and Knowledge-Base Systems (1980年)
- Principles of Compiler Design (1977年)
- The Design and Analysis of Computer Algorithms (1974年)
- Foundations of Computer Science (1992年)
- Elements of ML Programming (1994年)
- Mining of Massive Datasets (2011年)
- 大数据：互联网大规模数据挖掘 (2011年)
- 数据库系统实现 (2000年)

厄尔曼 › 图书 › 21st Century Compilers

The grid displays 5 book covers:

- Mining of Datasets (Second Edition)
- Database System Implementation (2000年)
- The Theory of Parsing, Translation and Optimization (1972年)
- Fundamental concepts of programming languages (1976年)
- Computational Aspects of VLSI (1984年)

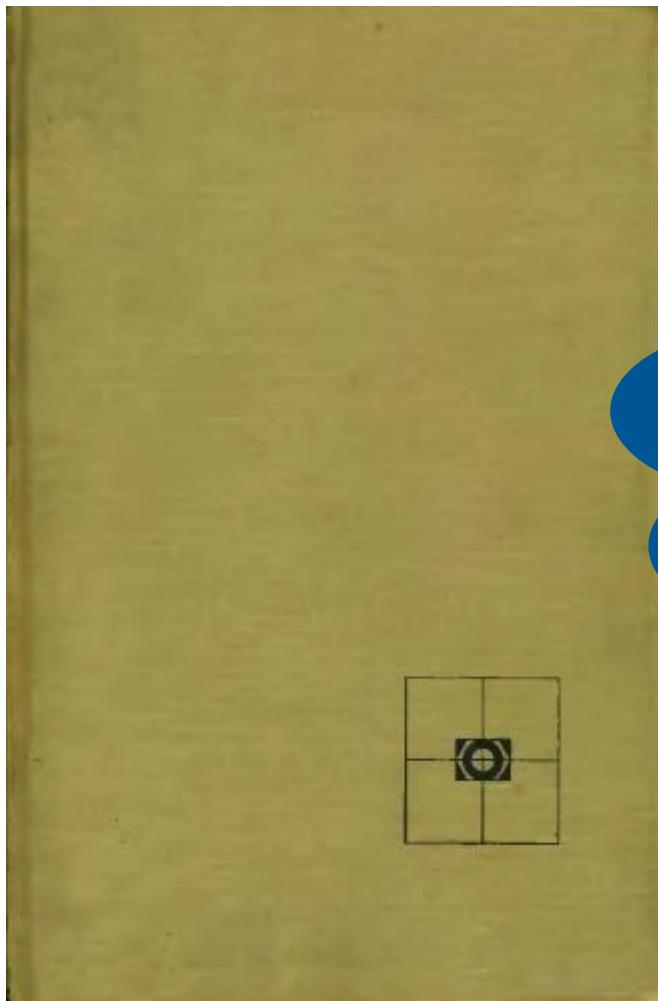
It's interesting to know
Ullman's book list, and you'll
find it's valuable for you to
know the D&I of DBMS

编译器是软件工程皇冠上的明珠

□ Books related to Compiler

- The core to D&I a (R)DBMS

Formal languages and their relation to automata



□ Hopcroft J.E., Ullman J.D.

SQL Translation and Execution is the core of DBMS. Ullman's first book may be this one – about Automata: Compiler's theoretical model

JOHN E. HOPCROFT

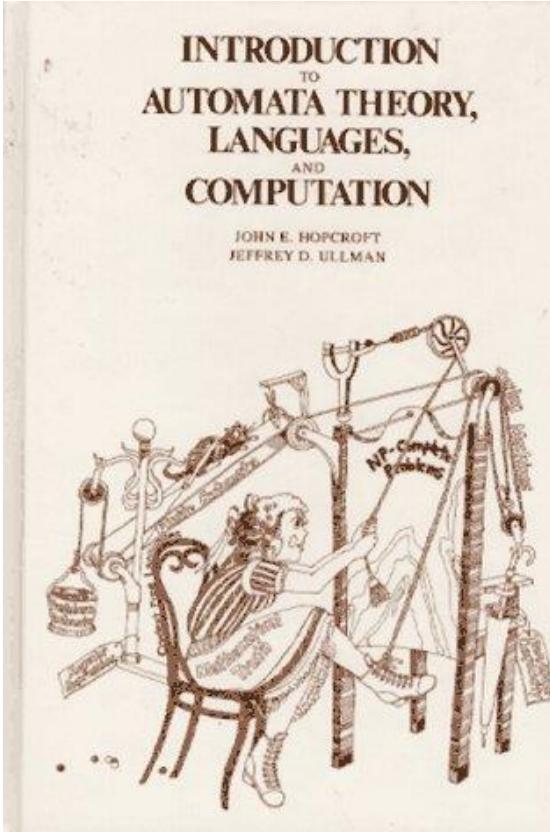
Cornell University, Ithaca, New York

JEFFREY D. ULLMAN

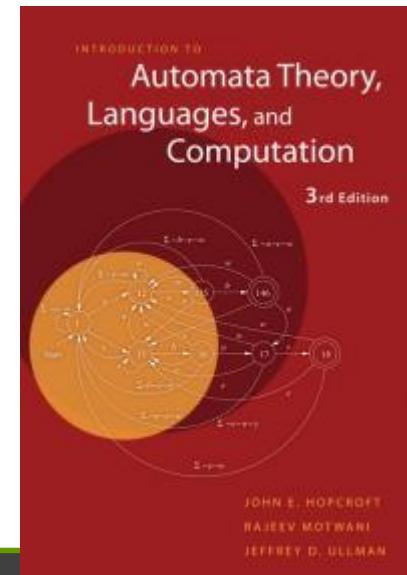
Bell Telephone Laboratories, Murray Hill, New Jersey

Introduction To Automata Theory, Languages, And Computation

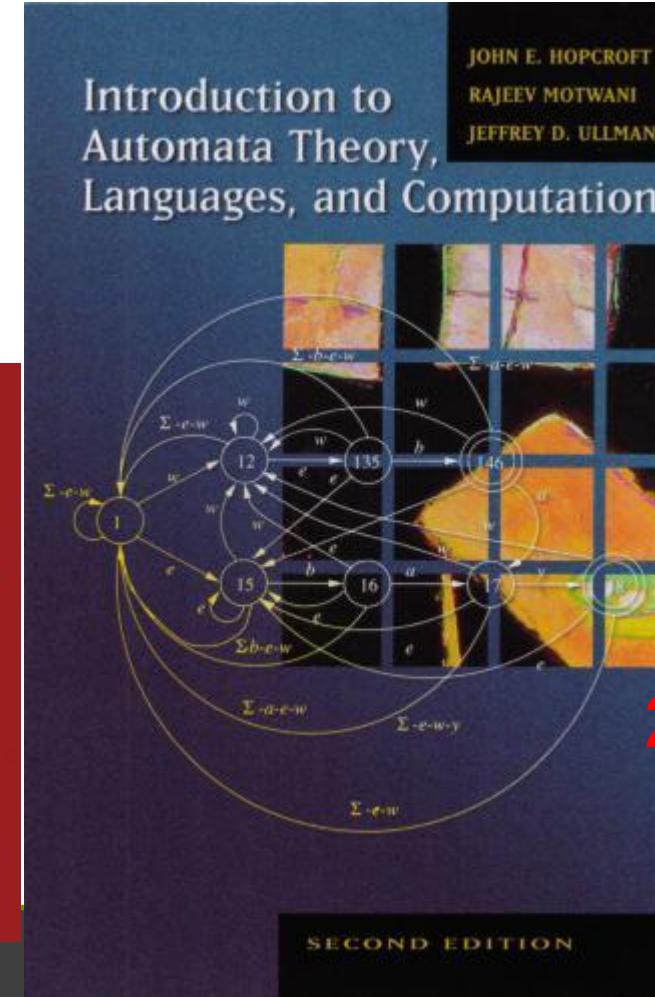
□ John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman



1979

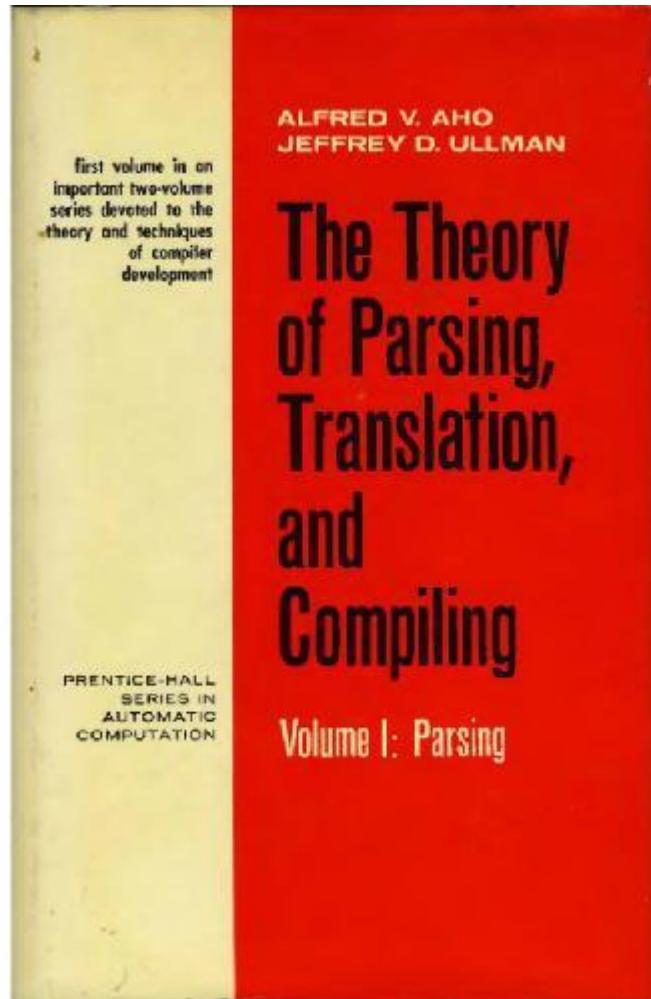


2006



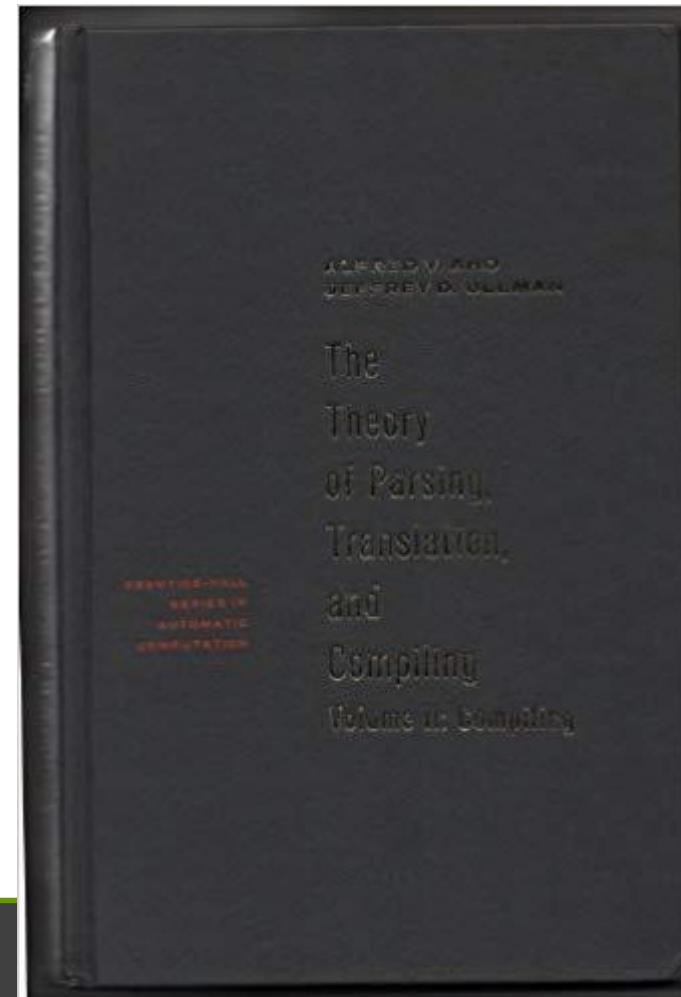
2001

The theory of parsing, translation, and compiling (Volume I - Parsing, II-Compiling)



1972

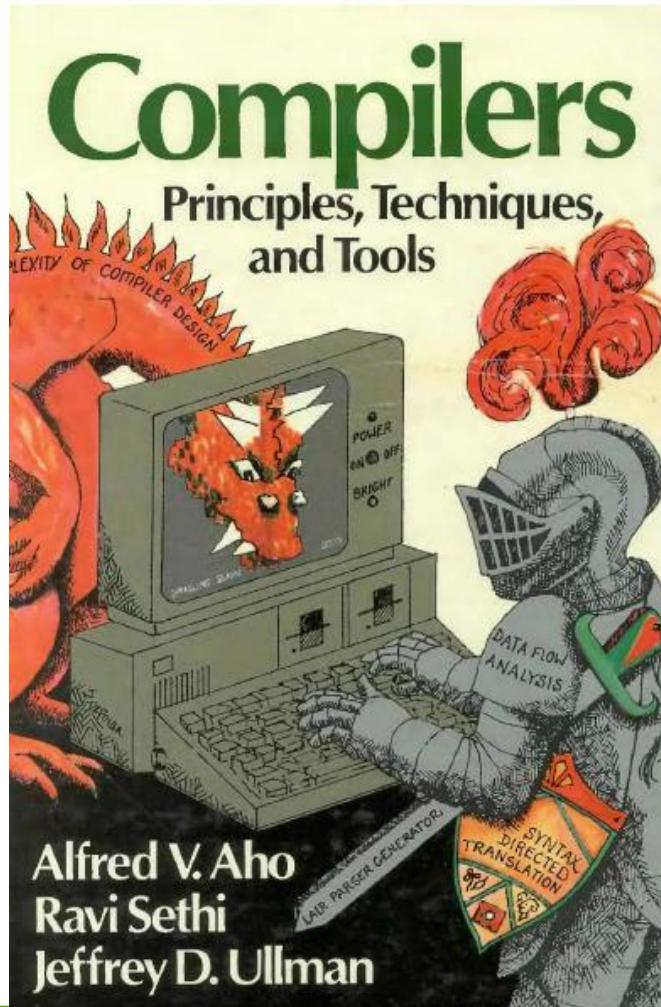
□ Alfred V. Aho
□ Jeffrey D. Ullman



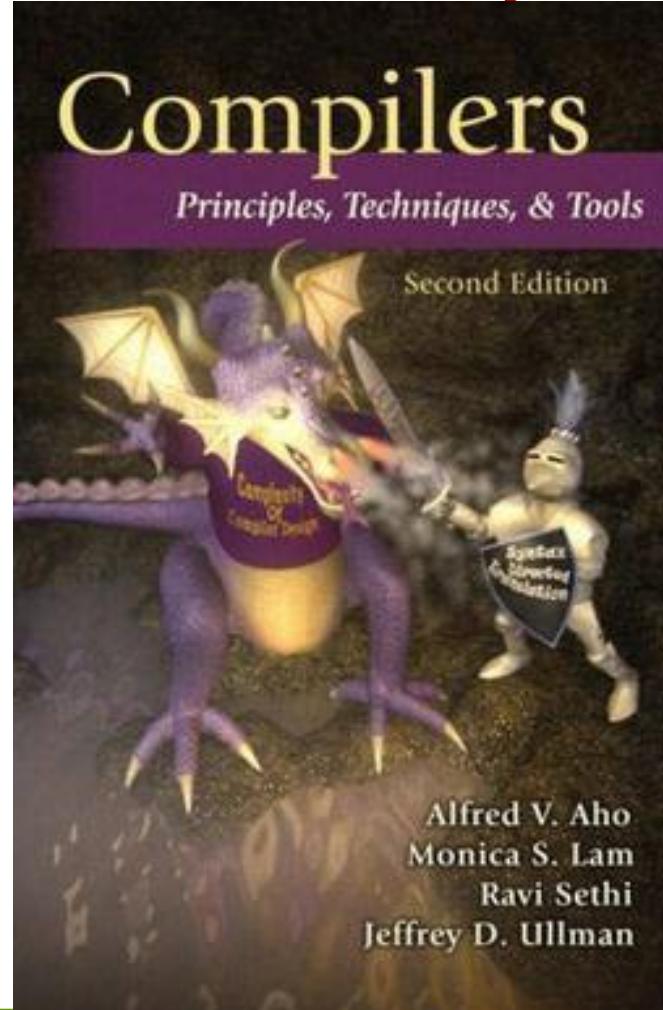
1973

Compilers: principles, techniques, & tools

□ Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman



1986



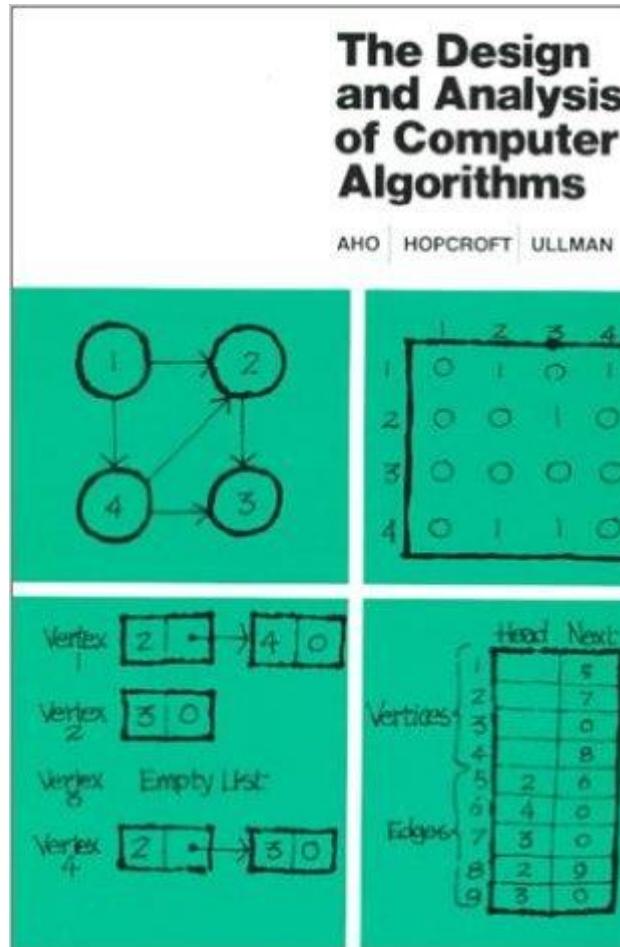
2007

□ Books related to Programming: DSA

- Data Structure and Algorithms
- To implement a (R)DBMS, you should be good at programming

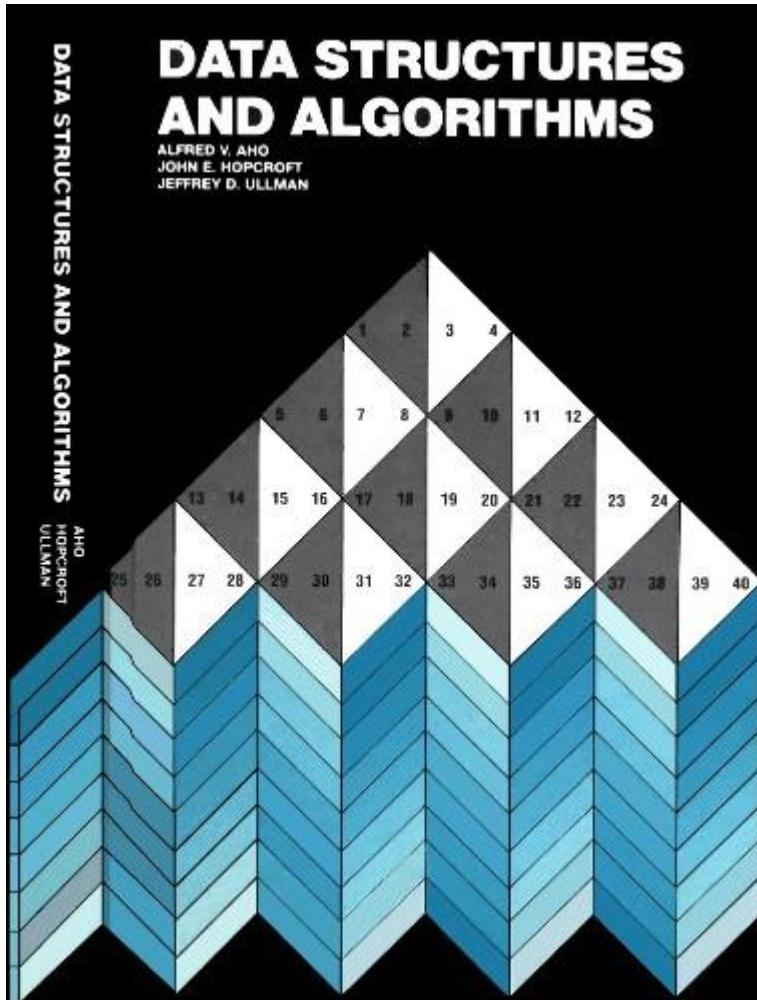
The Design and Analysis of Computer Algorithms

□ Alfred V. Aho, John E.
Hopcroft, Jeffrey D. Ullman



1974

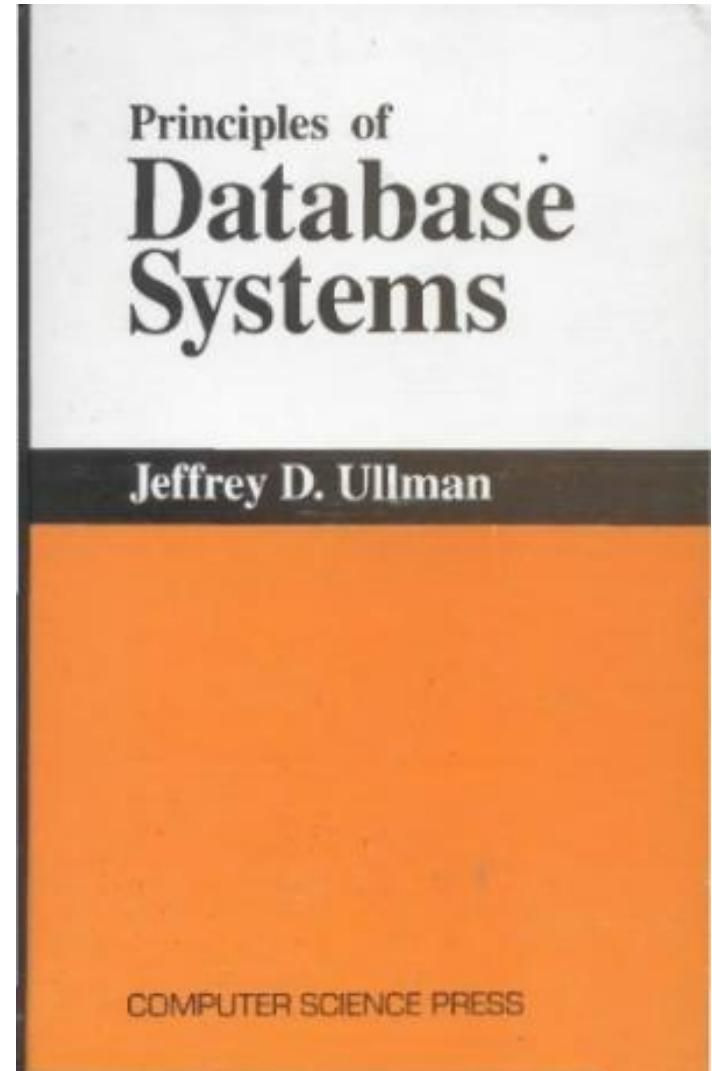
Data structures and algorithms



□ A. V: Aho, J. E. Hopcroft,
J. D. Ullman

1983

Books related to (R)DBMS



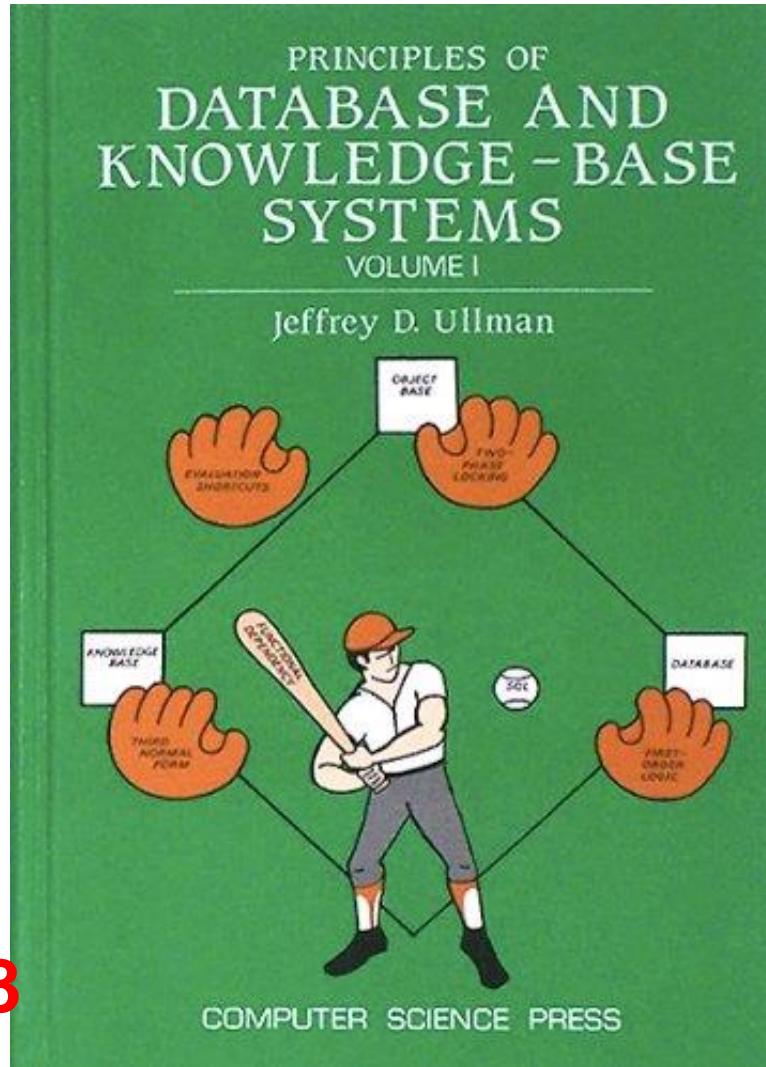
1980

- Principles of database systems
- Jeffrey D Ullman
- The comprehensive treatment of the theory and general principles relevant to database systems, as well as the many examples, exercises, and bibliographic notes included in each chapter, should make this book useful for database courses and for practising computer scientists.
- Computer Science Press
- 1980

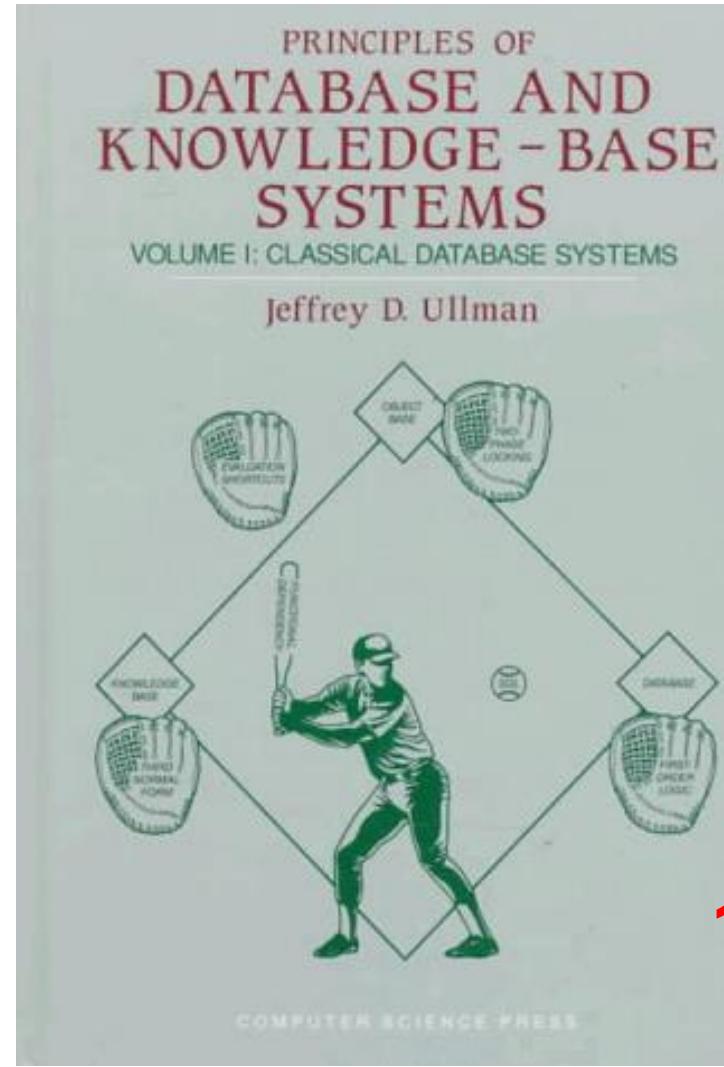


Database and Knowledge base systems (Volume I and II)

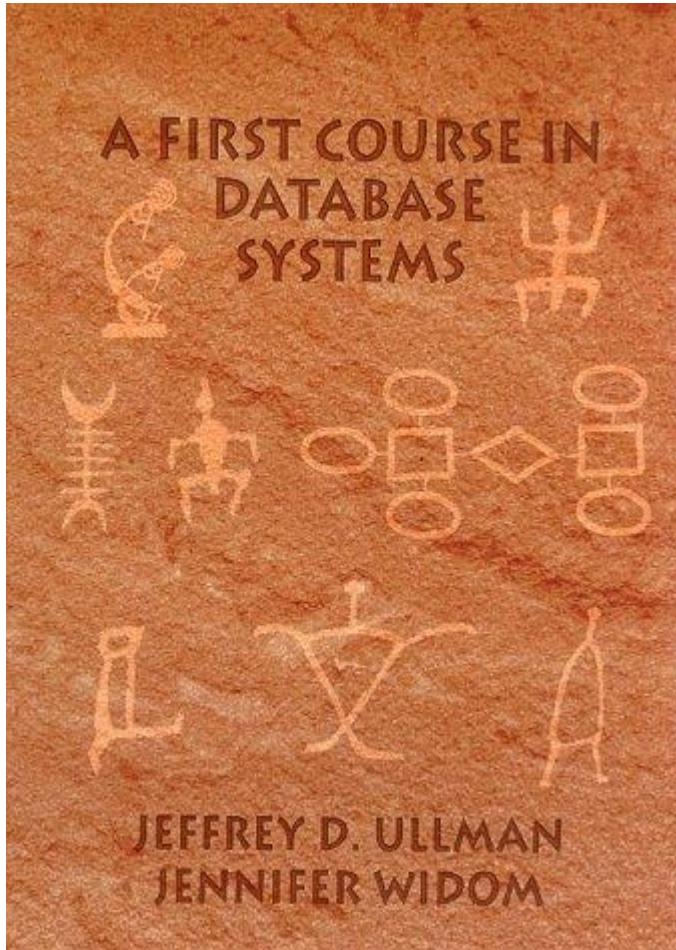
1988



1989



Like Silberschatz's - Concepts

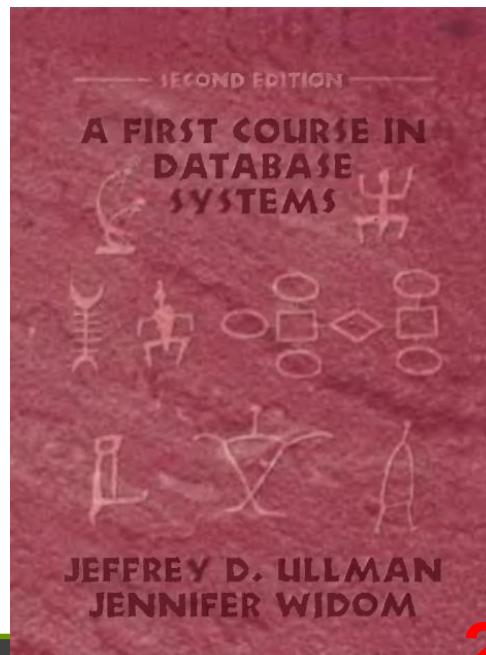


1997

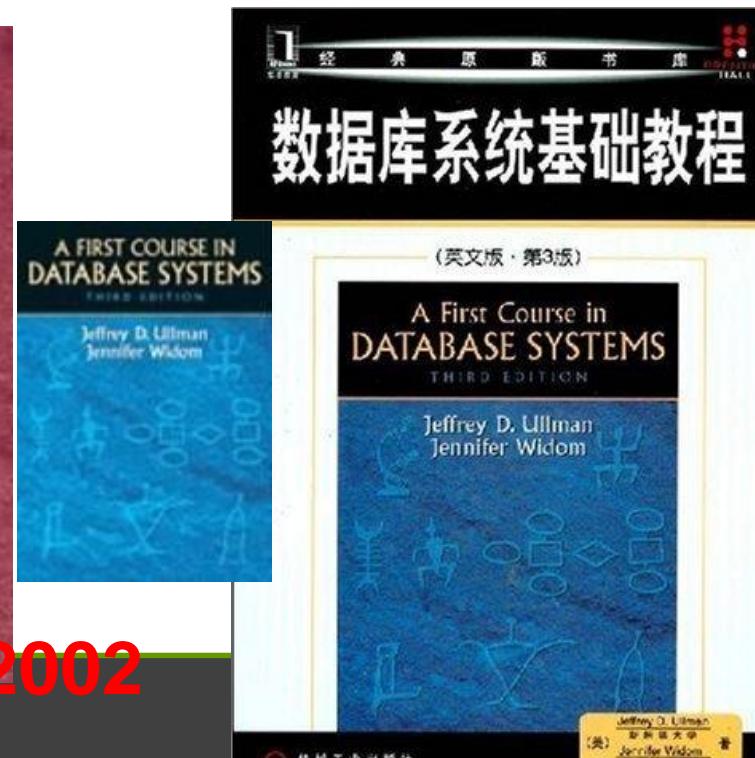
□ A First Course in Database Systems

□ Both of Stanford University

- Jeffrey D. Ullman
- Jennifer Widom



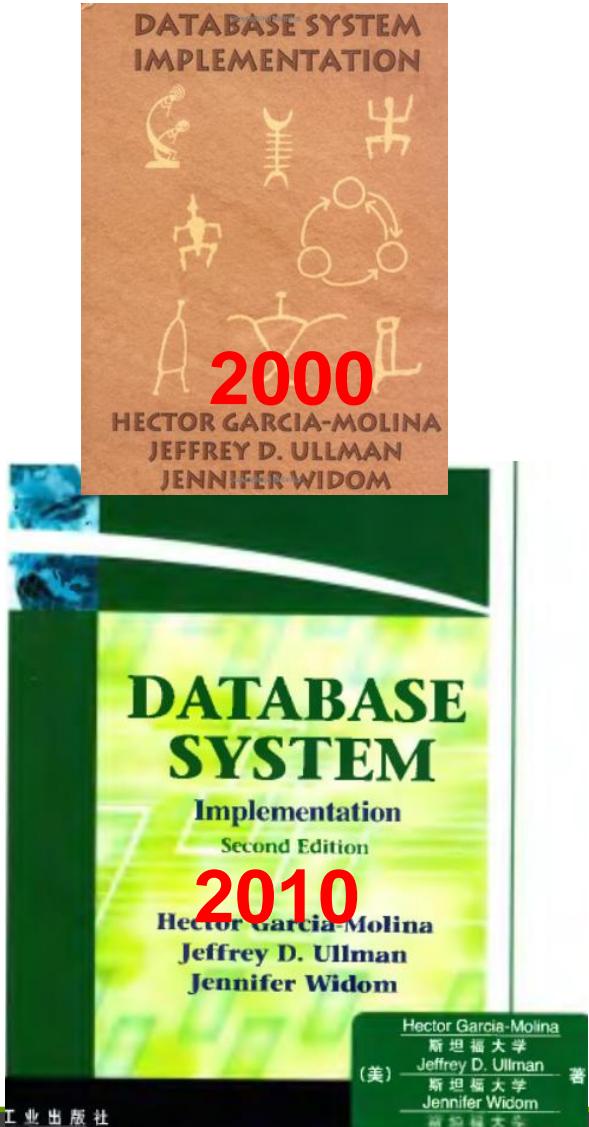
2002



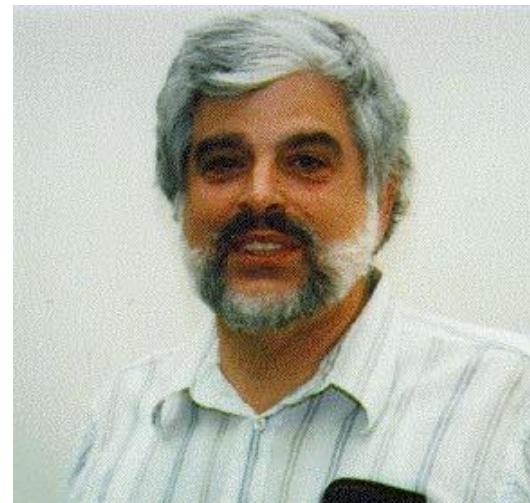
2008



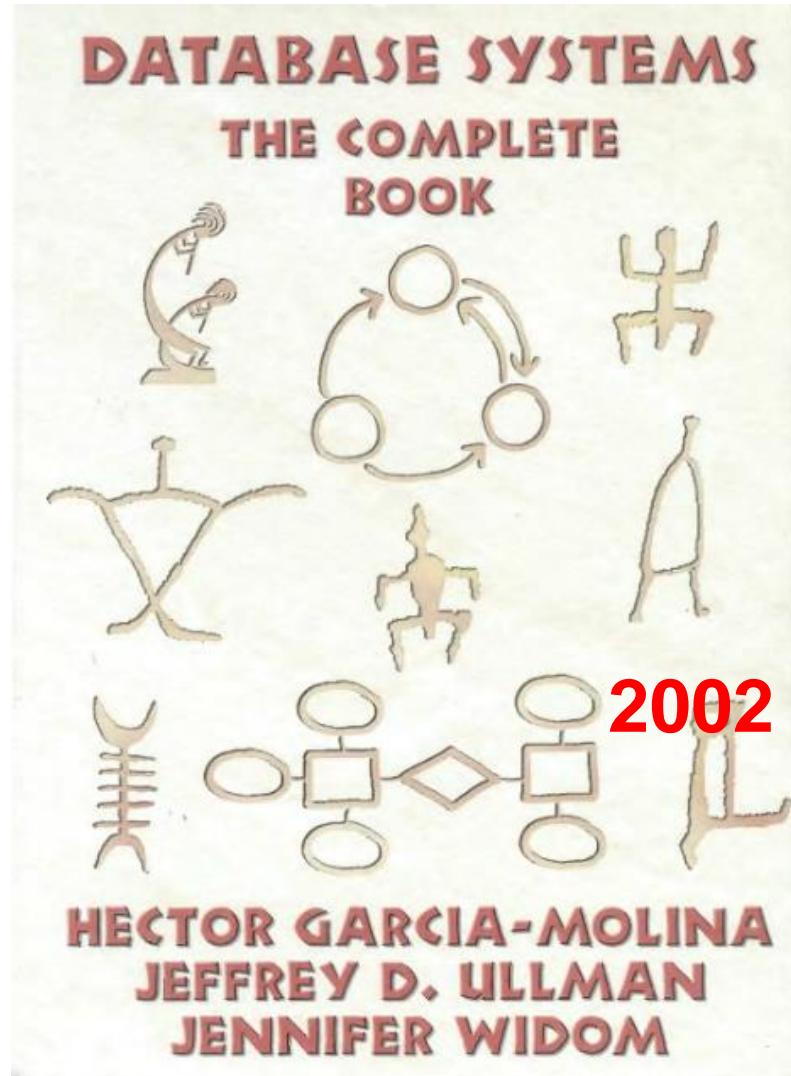
This one is better!



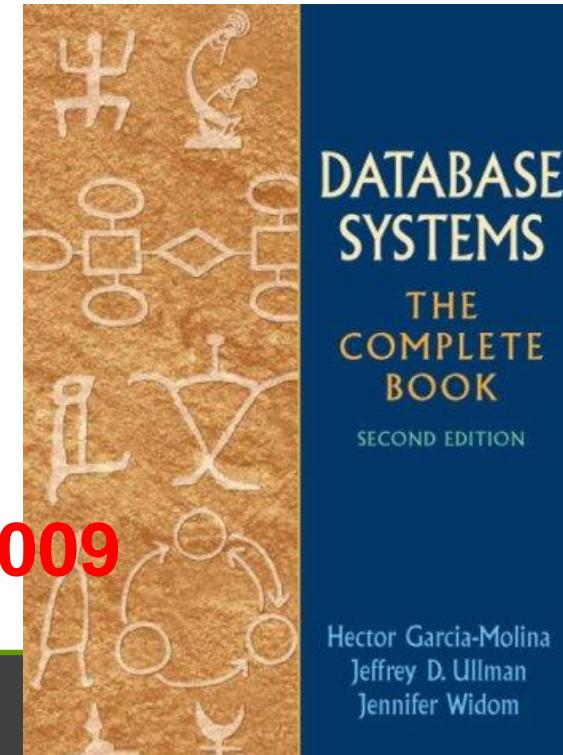
- Database System Implementation (2nd)
- all of Stanford University
 - Garcia-Molina, Jeffrey D. Ullman, Widom
- TOC
 - Record and Block Organization
 - Indexing & B trees + Hashing and other indexes
 - Query processing
 - Crash Recovery
 - Concurrency control
 - Transactions
 - Warehousing
 - Distributed Databases
 - BigTable, HBASE, Cassandra



A more ambitious book



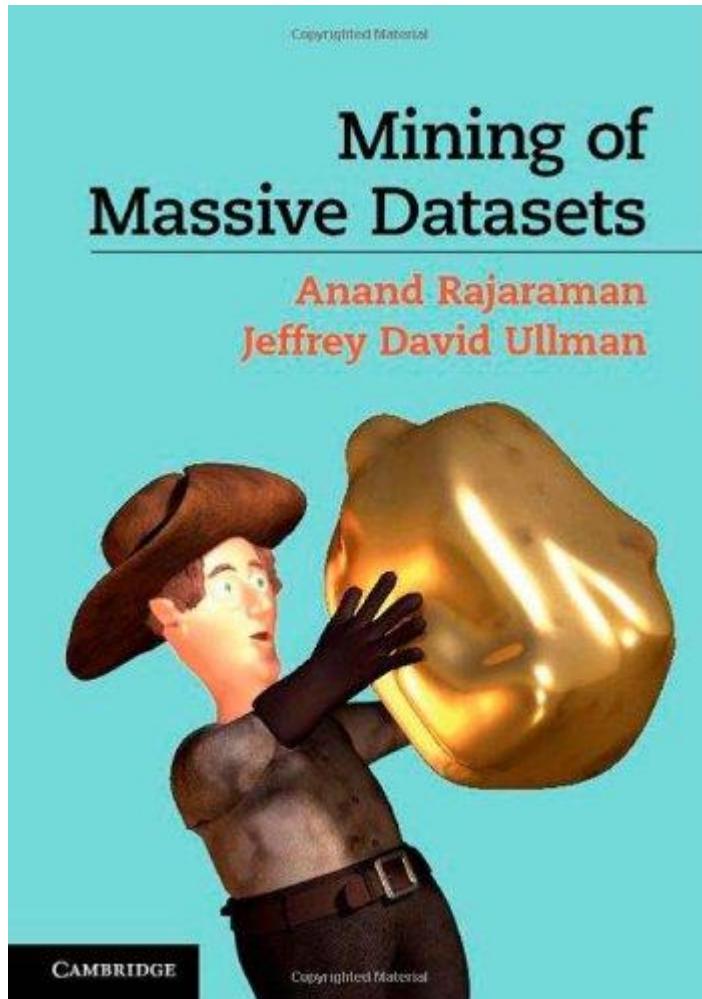
- Database System
The Complete Book
- all of Stanford University
 - Garcia-Molina, Jeffrey D.
Ullman, Widom



□ Books related to DM

- After (R)DBMS, DW/DM was once popular by integrating DBMS with ML

Mining of Massive Datasets



2011

- Anand Rajaraman,
- Jeffrey David Ullman



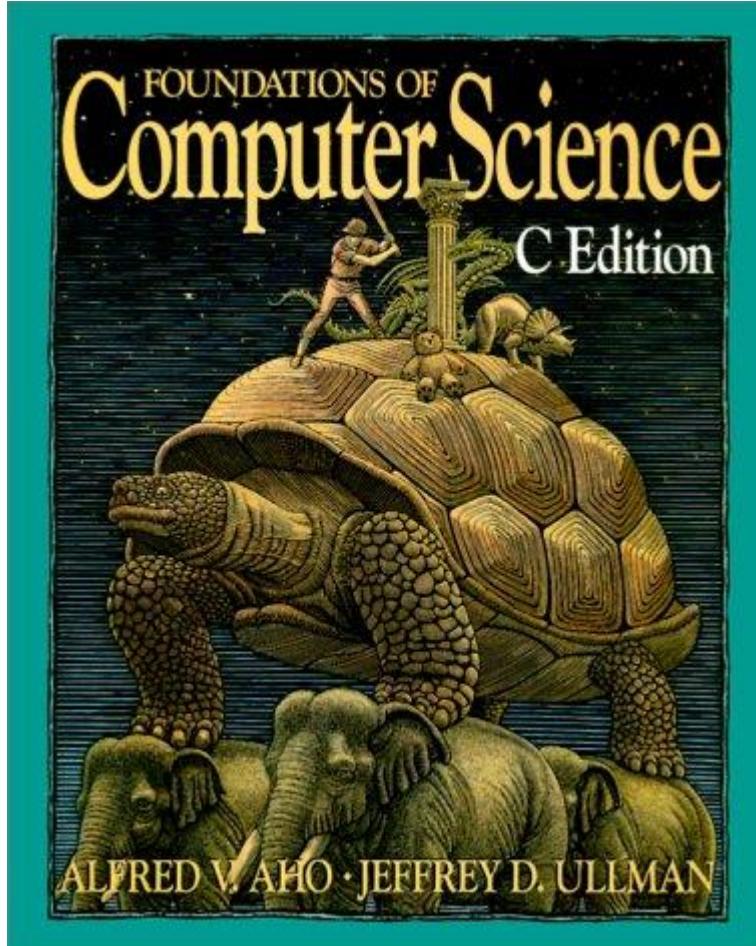
2015

Jure Leskovec,
Anand Rajaraman,
Jeffrey David Ullman

□ Other Books

- Ullman even once introduced a programming language – ML ("Meta Language")
[https://en.wikipedia.org/wiki/ML_\(programming_language\)](https://en.wikipedia.org/wiki/ML_(programming_language))
- Not Machine Learning

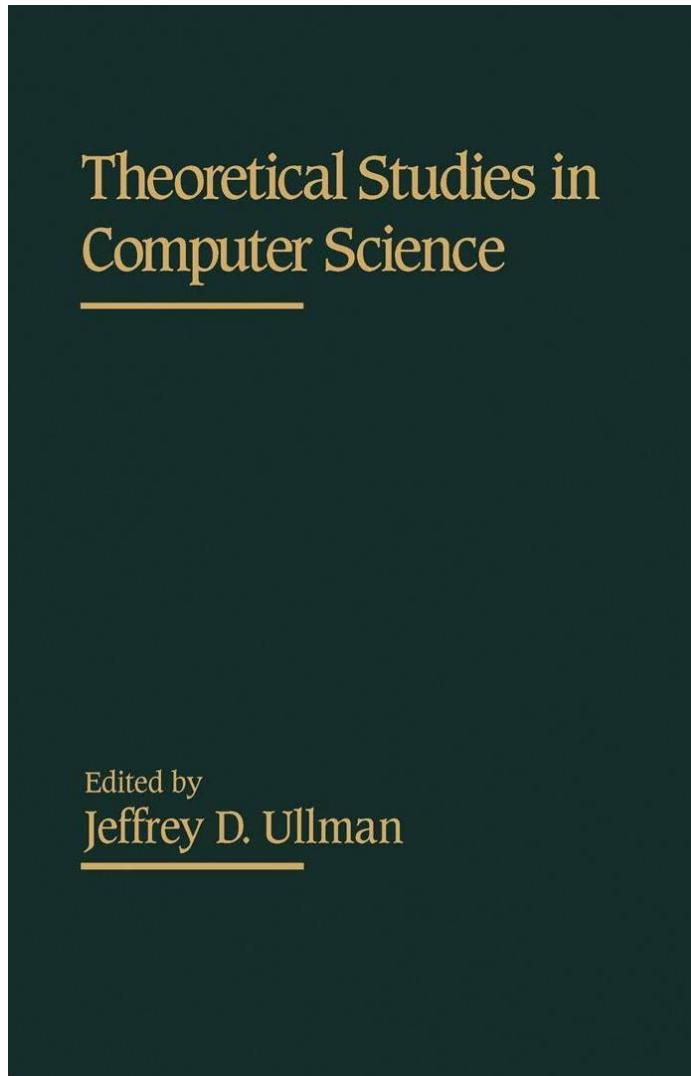
Foundations of computer science



- Alfred V. Aho
- Jeffrey D. Ullman

1995

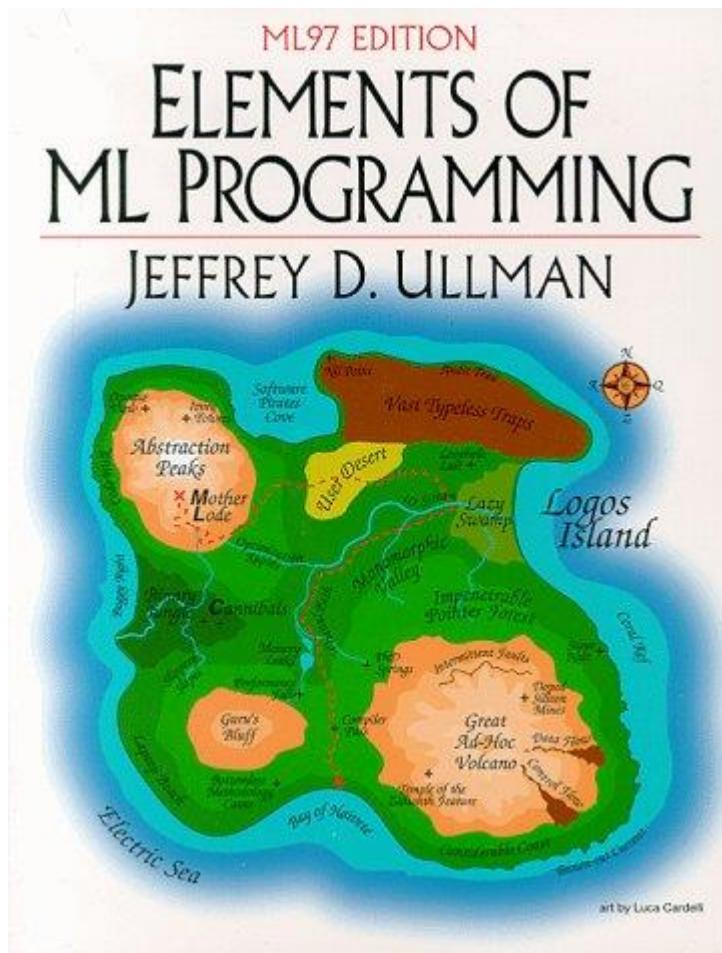
Theoretical Studies in Computer Science



□ Jeffrey D. Ullman

1992

Elements of ML Programming, ML97 Edition



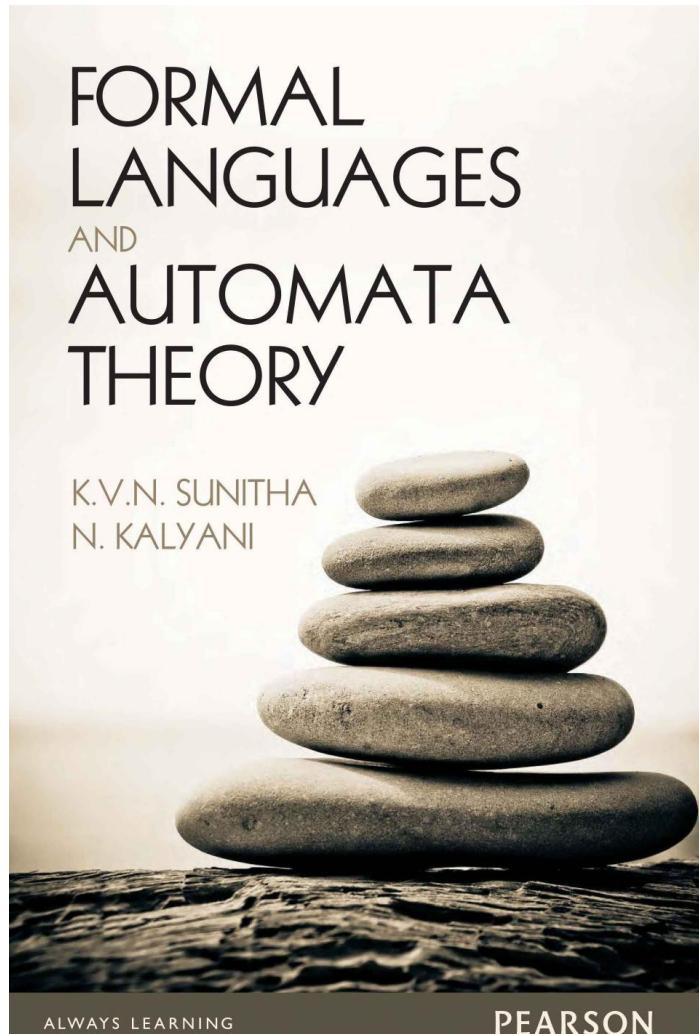
1997

<https://www.cs.unm.edu/~darko/smlnj/index.html>

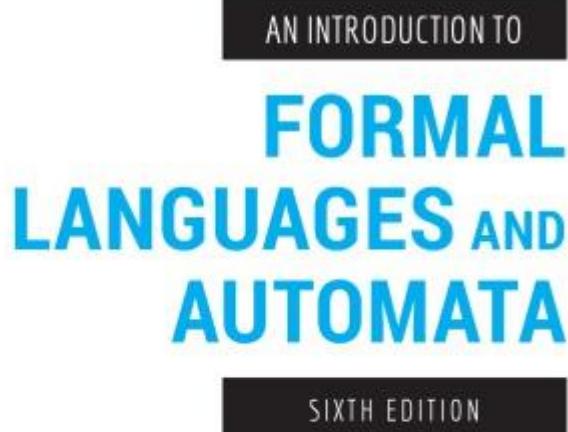


Standard ML of New Jersey

Standard ML is a safe, modular, strict, functional, polymorphic programming language with compile-time type checking and type inference, garbage collection, exception handling, immutable data types and updatable references, abstract data types, and parametric modules.



- Formal Languages and Automata Theory
- K.V.N. Sunitha

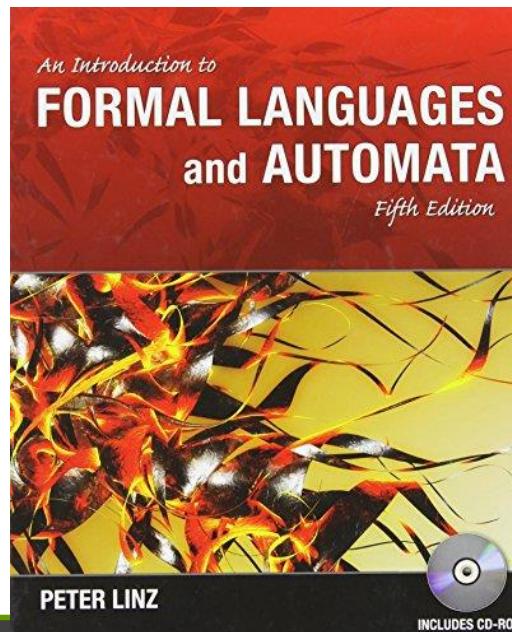


PETER LINZ
UNIVERSITY OF CALIFORNIA AT DAVIS



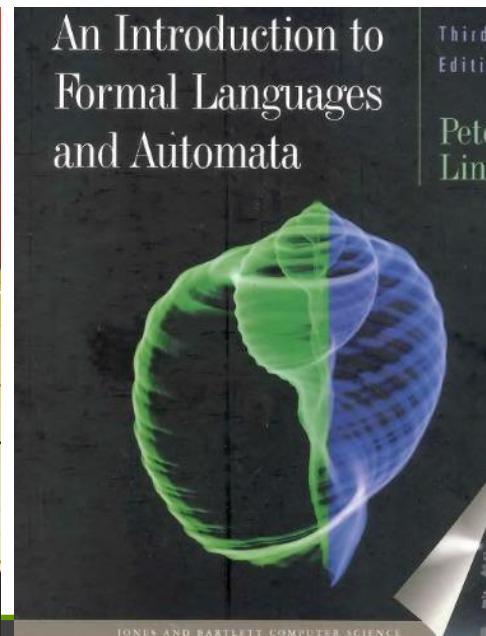
□ An Introduction to Formal Languages and Automata [6th ed.]

□ Peter Linz

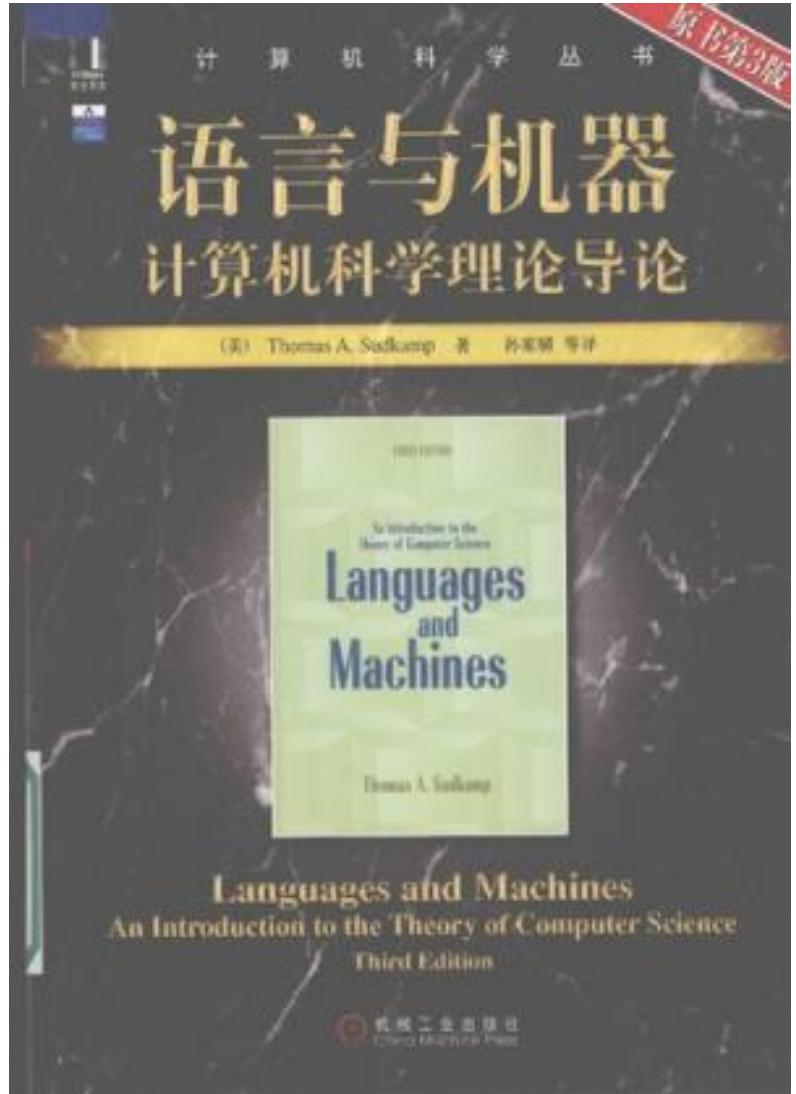


PETER LINZ

INCLUDES CD-ROM

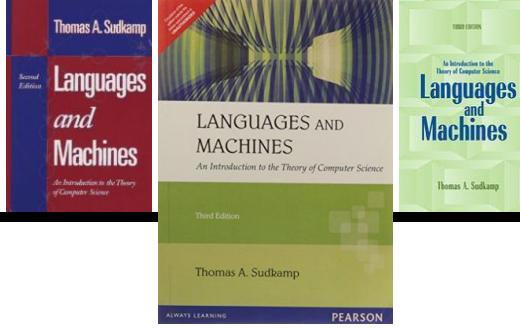


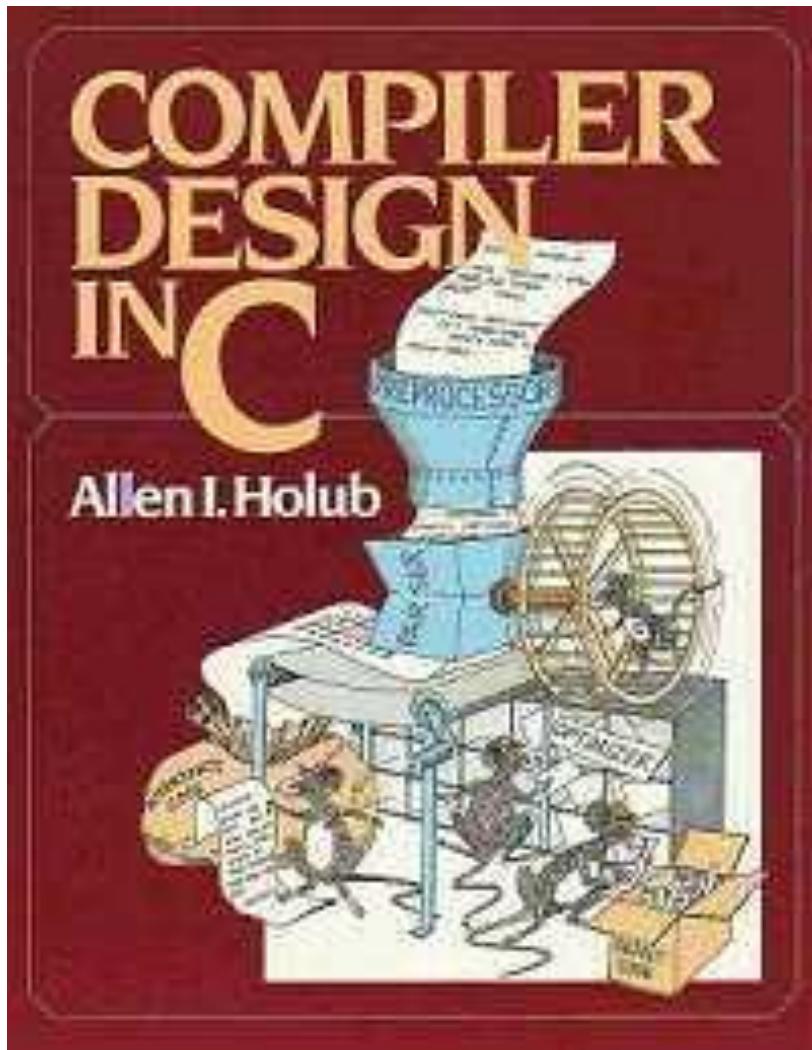
Peter Linz



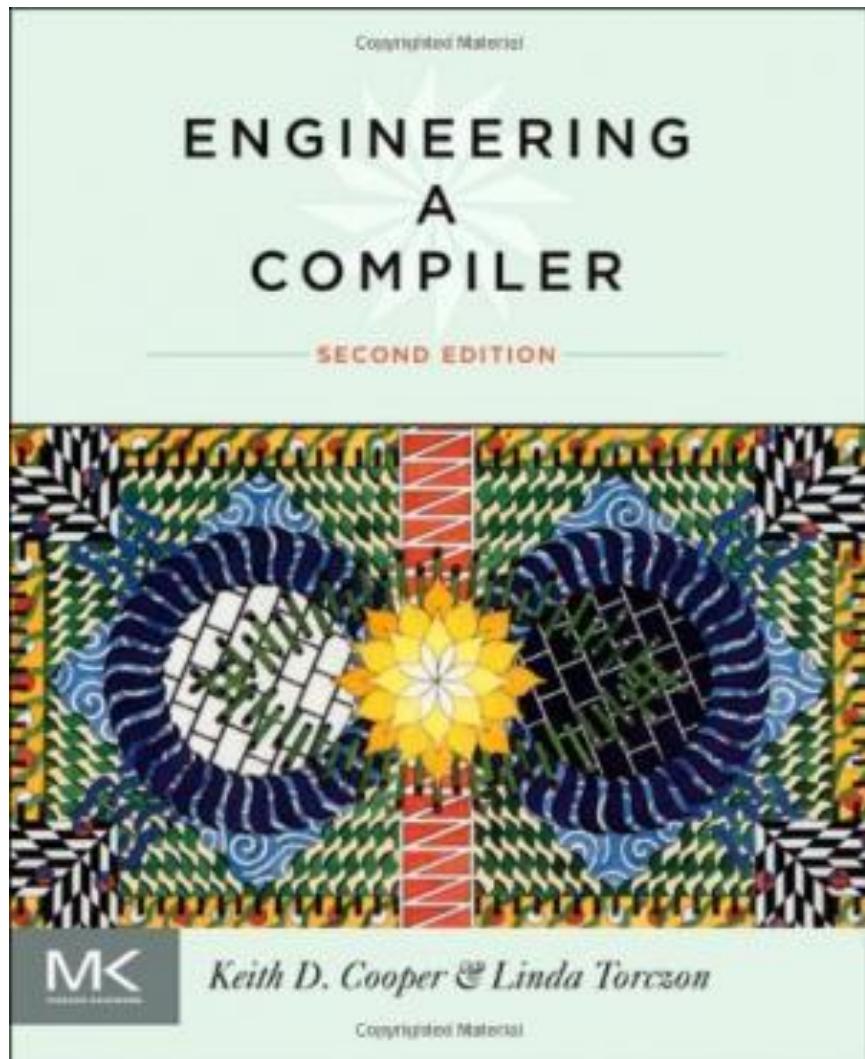
- 语言与机器: 计算机科学理论导论
- (美) 萨德坎普

■ 本书是计算理论方面的优秀教材之一，包括上下文无关文法、上下文无关文法范式、有限自动机、正则语言的性质、下推自动机和上下文无关语言、图灵机、图灵可计算函数、乔姆斯基层次、判定问题与丘奇图灵机、不可判定性、Mu-递归函数、时间复杂性、库克定理、NP-完全问题、LL(k)文法以及LR(k)文法等问题。本书不仅介绍了计算机科学的基础，而且通过概念的严格表述，以及使用通俗的例子来阐释定理，从而帮助学生提高数学论证能力以及对计算理论知识的全面深入的理解。书中每章后面都有附有大量习题，通过完成这些习题，学生可以加深对本章内容的理解。



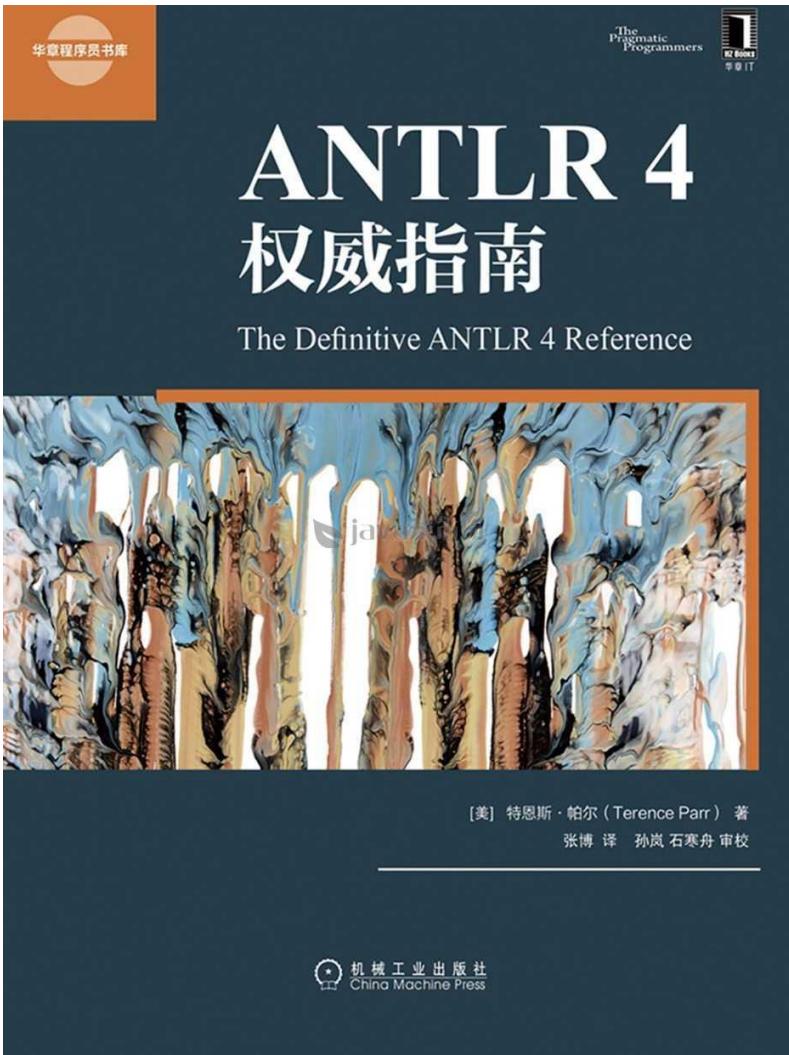


- Compiler Design in C
- Allen I. Holub



- Engineering a Compiler
- Keith Cooper, Linda Torczon

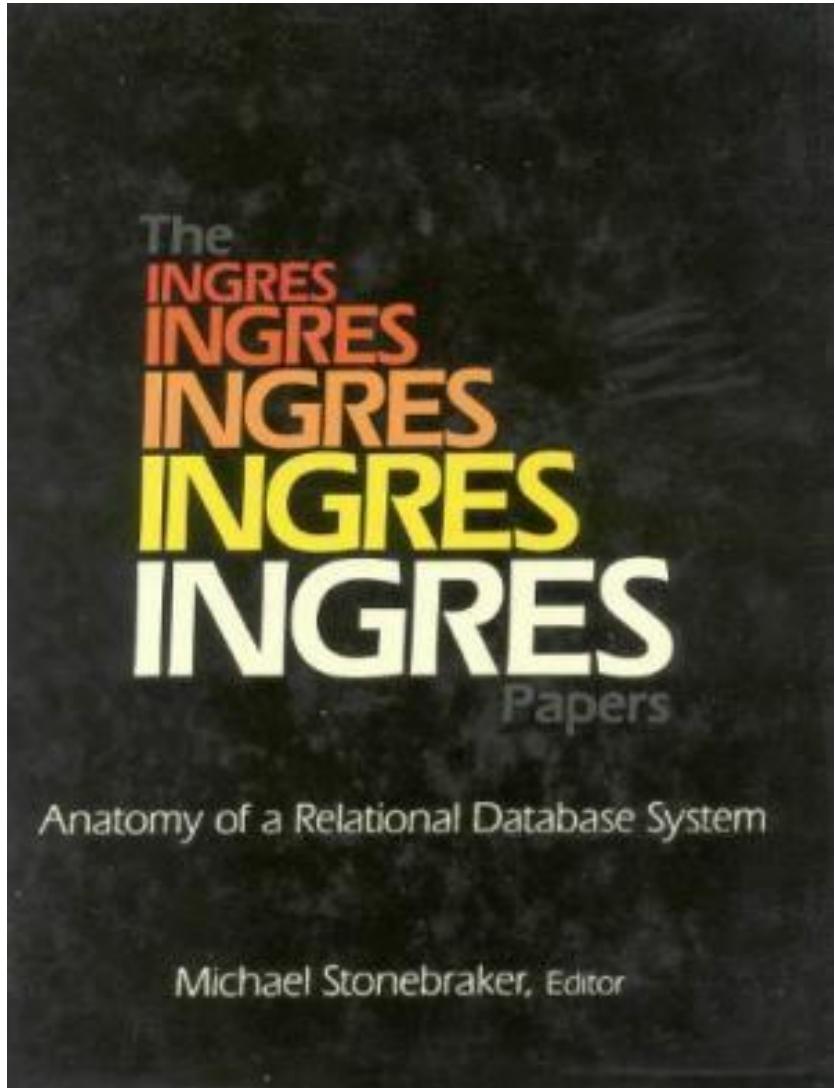
■ This entirely revised second edition of *Engineering a Compiler* is full of technical updates and new material covering the latest developments in compiler technology. In this comprehensive text you will learn important techniques for constructing a modern compiler. Leading educators and researchers Keith Cooper and Linda Torczon combine basic principles with pragmatic insights from their experience building state-of-the-art compilers. They will help you fully understand important techniques such as compilation of imperative and object-oriented languages.



- **ANTLR 4权威指南**
- **作者: Terence Parr**
- **出版社: 机械工业出版社**
- **译者: 张博**
- **出版年: 2017-5-1**
- **页数: 262**
- **定价: 69元**
- **丛书: 华章程序员书库**
- **ISBN: 9787111566489**



Michael (Mike) Stonebraker – PostgreSQL



□ 2014年图灵奖得主Michael Stonebraker



朱小虎XiaohuZhu [关注](#)

0.495 2015.03.26 18:00:34 字数 524 阅读 2,111



Michael Stonebraker 教授

今天在微博上看到这个消息，赶紧去[官方网站](#)看了看。

网站上首页新闻标题写着“Michael Stonebraker, Pioneer in Database Systems Architecture, Receives 2014 ACM Turing Award”。ACM将在 San Francisco 于今年6月份举办颁奖典礼。

看点一：

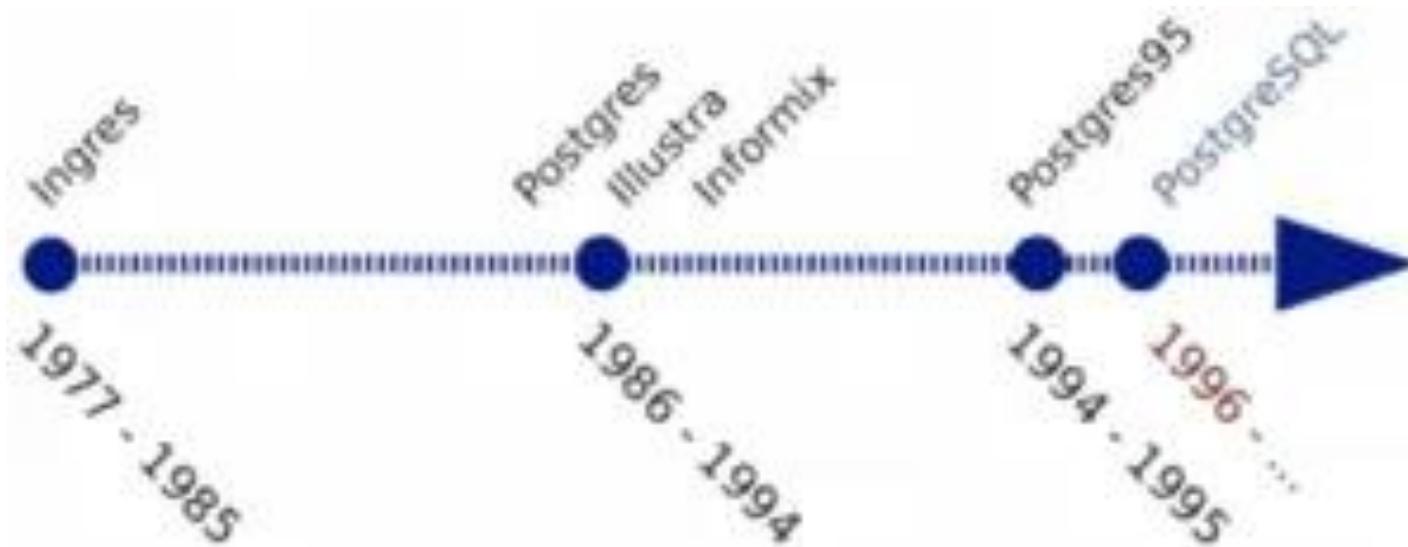
这是数据库领域获得的第四个图灵奖。前面几位是：1973年Bachman（数据库与网状数据库）、1981年Codd（关系数据库）、1998年Gray（数据库与事务处理）。

1973: Michael Stonebraker and Eugene Wong implement Codd's vision in INGRES

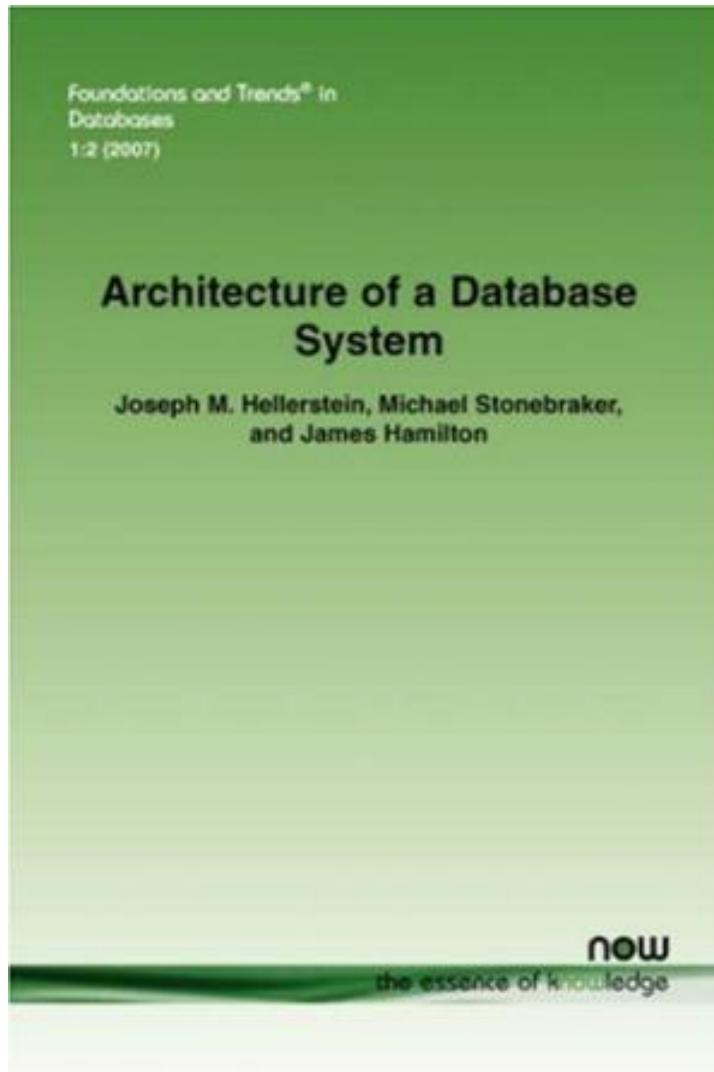
- Commercialized in 1983
- Evolved to Postgres (now PostgreSQL) in 1989



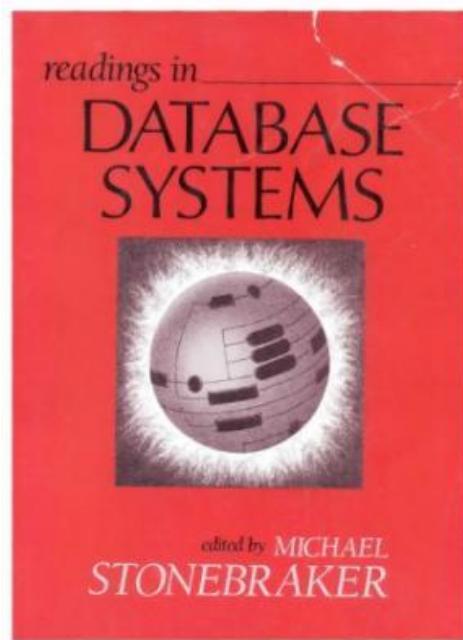
M. Stonebraker



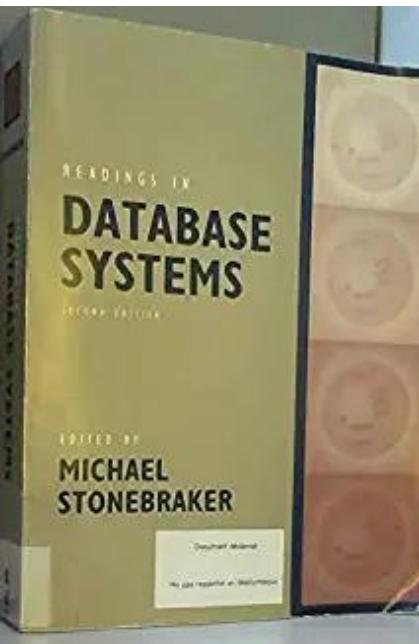
E. Wong



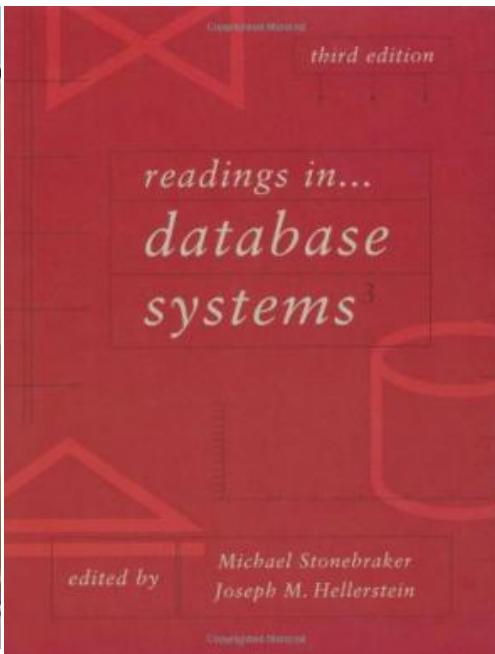
- Architecture of a Database System
- *Joseph M Hellerstein, Michael Stonebraker, James Hamilton*
 - Database Management Systems (DBMSs) are a ubiquitous and critical component of modern computing, and the result of decades of research and development in both academia and industry. *Architecture of a Database System* presents an architectural discussion of DBMS design principles, including process models, parallel architecture, storage system design, transaction system implementation, query processor and optimizer architectures, and typical shared components and utilities. Successful commercial and open-source systems are used as points of reference, particularly when multiple alternative designs have been adopted by different groups. Historically, DBMSs were among the earliest multi-user server systems to be developed, and thus pioneered many systems design techniques for scalability and reliability now in use in many other contexts. While many of the algorithms and abstractions used by a DBMS are textbook material, *Architecture of a Database System* addresses the systems design issues that make a DBMS work. *Architecture of a Database System* is an invaluable reference for database researchers and practitioners and for those in other areas of computing interested in the systems design techniques for scalability and reliability that originated in DBMS research and development.



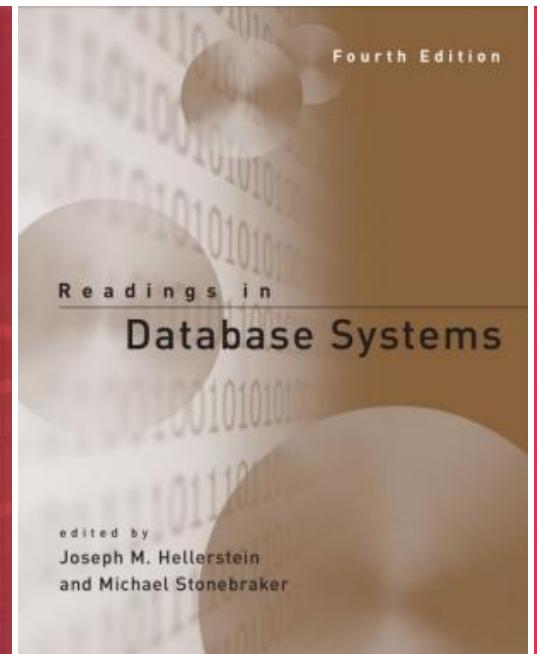
1988



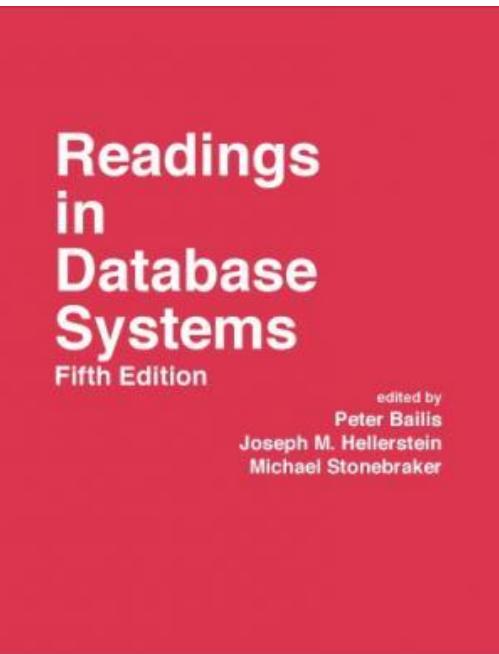
1994



1998



2005



2015

This book is dedicated to Michael Stonebraker, Jim Gray,
Ted Codd, and Charlie Bachman, recipients of the ACM
A.M. Turing Award for the management of data, one of the
world's most valuable resources, and to their many collaborators,
particularly the contributors to this volume.



Making Databases Work

The Pragmatic Wisdom of Michael Stonebraker

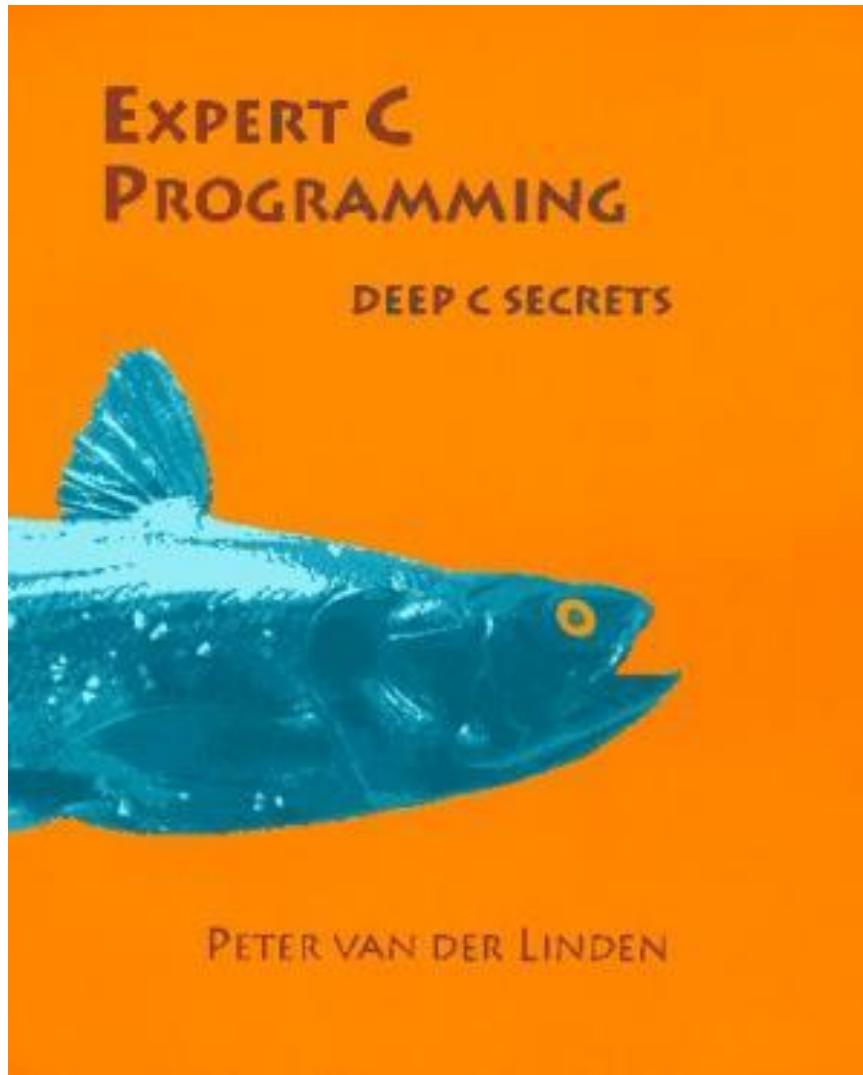
Michael L. Brodie
Editor



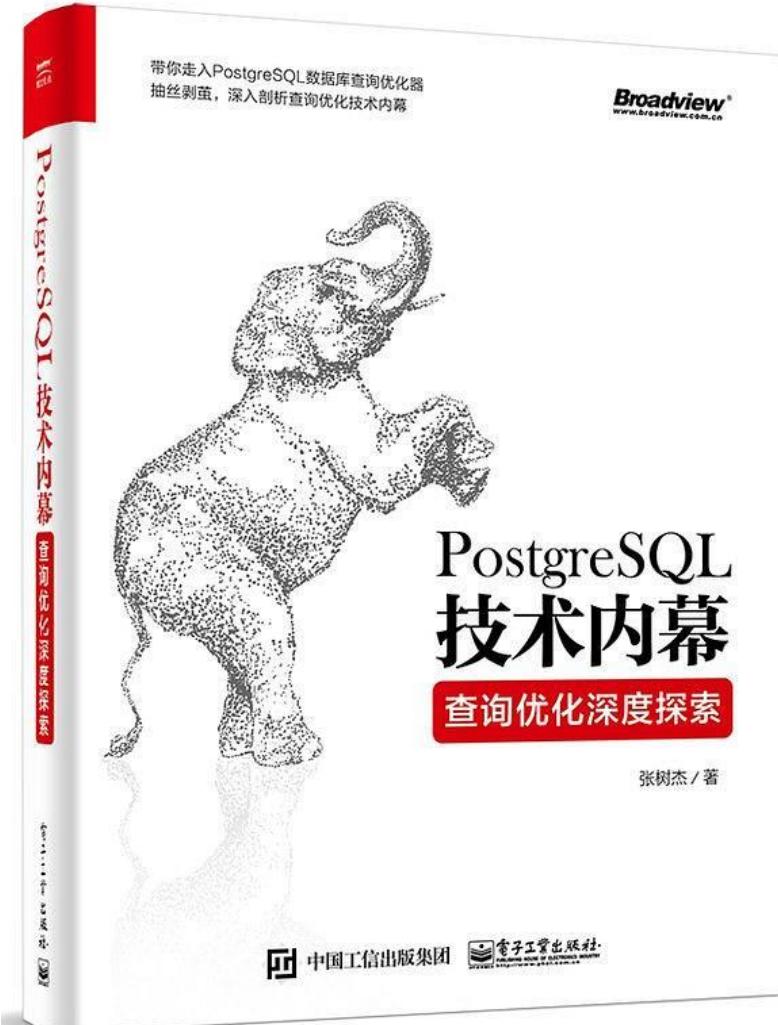
□ Making Databases Work: The Pragmatic Wisdom of Michael Stonebraker

□ Michael L. Brodie

■ This book celebrates Michael Stonebraker's accomplishments that led to his 2014 ACM A.M. Turing Award "for fundamental contributions to the concepts and practices underlying modern database systems." The book describes, for the broad computing community, the unique nature, significance, and impact of Mike's achievements in advancing modern database systems over more than forty years.



- Expert C Programming, Deep C Secrets
- Peter van der Linden
 - This is a very different book on the C language! In an easy, conversational style, Peter van der Linden, of Sun's compiler and OS kernel group, presents dozens of astonishing examples drawn from practical experience, including:
 - * Software that blew up the space probe to Venus
 - * The C bug that shut down the entire AT&T phone system
 - * C programmer job interview secrets
 - * Why programmers can't tell Halloween from Christmas day
 - * The C code for a complete BASIC interpreter
 - Expert C Programming reveals the coding techniques used by the best C programmers. It relates C to other languages, and includes an introduction to C++ that can be understood by an programmer without weeks of mind-bending study. Covering both the IBM PC and UNIX systems, it is an entertaining and educational romp through C showing how experts really use it. Expert C Programming is a must read for anyone who wants to learn more about the implementation, practical use, and folklore of C.



- **PostgreSQL技术内幕：查询优化深度探索**
- **作者: 张树杰**
- **出版社: 电子工业出版社**
- **出版年: 2018-6**
- **定价: 79**
- **ISBN: 9787121341489**

系统、深入地剖析开源数据库PostgreSQL的实现机制和工作原理
作者跟踪、研究PostgreSQL源代码十年的呕心之作

数据库技术

PostgreSQL 数据库内核分析



彭智勇 彭煜玮 ◎编著

机械工业出版社
China Machine Press

- PostgreSQL 数据库内核分析
- 作者: 彭智勇 / 彭煜玮
- 出版社: 机械工业出版社华章公司
- 出版年: 2012-1





□ 数据库查询优化器的艺术：原理解析与SQL性能优化

□ 李海翔

□ 2014

推荐序一	2.2.3 等价谓词重写 29
推荐序二	2.2.4 条件化谓 32
前言	2.2.5 外连接消除 33
	2.2.6 嵌套连接消除 37
	2.2.7 连接消除 38
	2.2.8 语义优化 40
	2.2.9 针对非 SPJ 的优化 41
第一篇 查询优化技术	2.3 启发式规则在逻辑优化阶段的运用 42
第 1 章 数据管理系统的查询优化 2	2.4 本章小结 43
1.1 数据库调优 3	第 3 章 物理查询优化 44
1.2 查询优化技术 5	3.1 查询代价估算 44
1.2.1 查询重用 5	3.1.1 代价模型 44
1.2.2 查询重写规则 6	3.1.2 选择率计算的常用方法 45
1.2.3 查询算法优化 6	3.2 单表扫描算法 45
1.2.4 并行查询优化 8	3.2.1 常用的单表扫描算法 45
1.2.5 分布式查询优化 9	3.2.2 单表扫描代价计算 47
1.2.6 其他优化 9	3.3 索引 47
1.3 本章小结 9	3.3.1 如何利用索引 47
第 2 章 逻辑查询优化 10	3.3.2 索引列的位置对使用索引的影响 50
2.1 查询优化技术的理论基础 10	3.3.3 联合索引对索引使用的影响 56
2.1.1 关系代数 11	3.3.4 多个索引对索引使用的影响 57
2.1.2 关系代数等价变换规则	
2.1.3 对优化的指导意义 13	
2.2 查询重写规则 17	
2.2.1 子查询的优化 18	
2.2.2 视图重写 28	

3.4 两表连接算法 59	5.5 本章小结 86
3.4.1 基本的两表连接算法 59	第 6 章 PostgreSQL 查询优化器
3.4.2 进一步认识两表连接算法 61	相关数据结构 88
3.4.3 连接操作代价计算 61	6.1 主要数据结构 88
3.5 多表连接算法 62	6.1.1 基本数据结构 88
3.5.1 多表连接顺序 62	6.1.2 查询树 91
3.5.2 常用的多表连接算法 63	6.1.3 各种对象的结构 95
3.5.3 多表连接算法的比较 68	6.1.4 连接操作相关的结构 99
3.6 本章小结 68	6.1.5 查询执行计划相关的结构 104
第 4 章 查询优化器与其他模块的关系 70	6.2 各个结构之间的关系 108
4.1 查询优化器整体介绍 70	6.3 各个阶段间和主要结构体间的关系 109
4.2 查询优化器与其他模块的关系 73	6.4 本章小结 110
4.3 本章小结 74	
第二篇 PostgreSQL 查询优化器原理解析	第 7 章 PostgreSQL 查询优化器实现原理解析 111
第 5 章 PostgreSQL 查询优化器概述 76	7.1 查询优化整体流程 111
5.1 PostgreSQL 查询执行过程 76	7.2 查询优化器实现原理解析 115
5.2 PostgreSQL 查询优化器的架构和设计思想 78	7.2.1 planner——入口函数 115
5.2.1 PostgreSQL 查询优化器架构 79	7.2.2 standard_planner——标准的查询优化器函数 116
5.2.2 PostgreSQL 查询优化器的层次 81	7.2.3 subquery_planner——生成(子)查询执行计划函数 117
5.2.3 PostgreSQL 查询优化器设计思想 81	7.2.4 grouping_planner——生成查询执行计划并对非 SPJ 优化 139
5.3 主要概念 81	7.2.5 build_minmax_path——聚集函数 MIN/MAX 的优化函数 141
5.4 代码层次结构 85	



PostgreSQL
the world's most advanced open source database

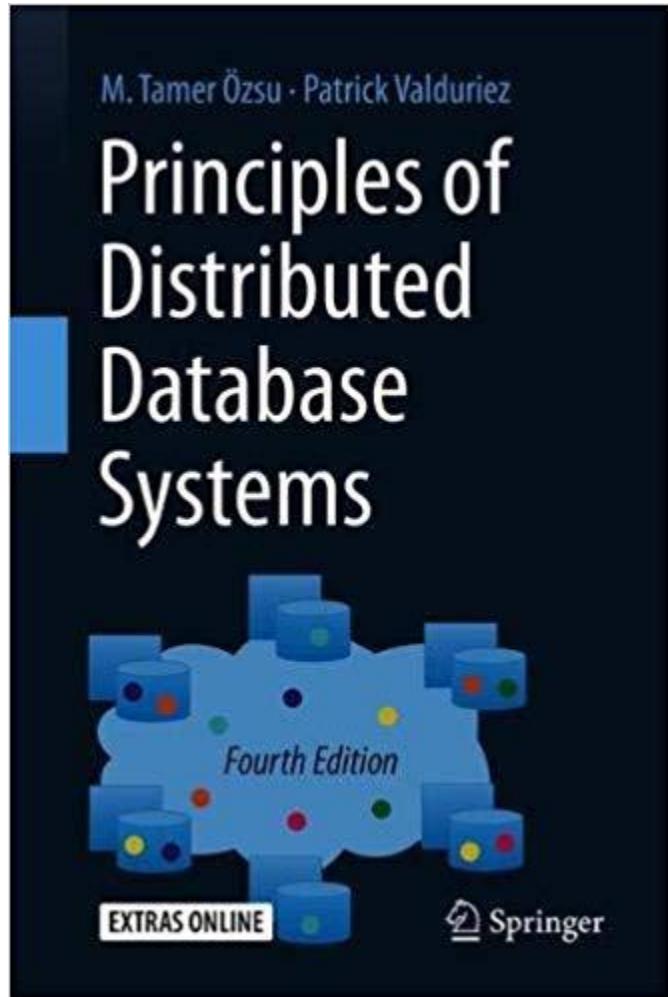


Pivotal
Greenplum®

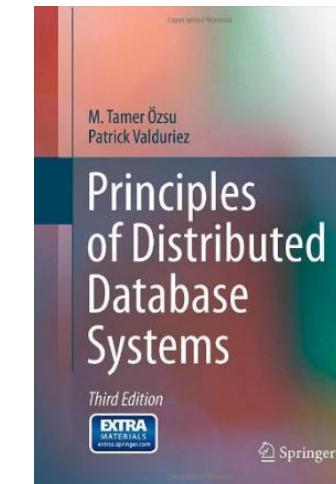
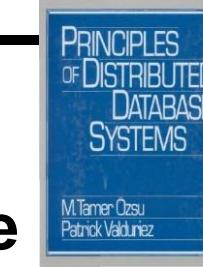


HAWQ

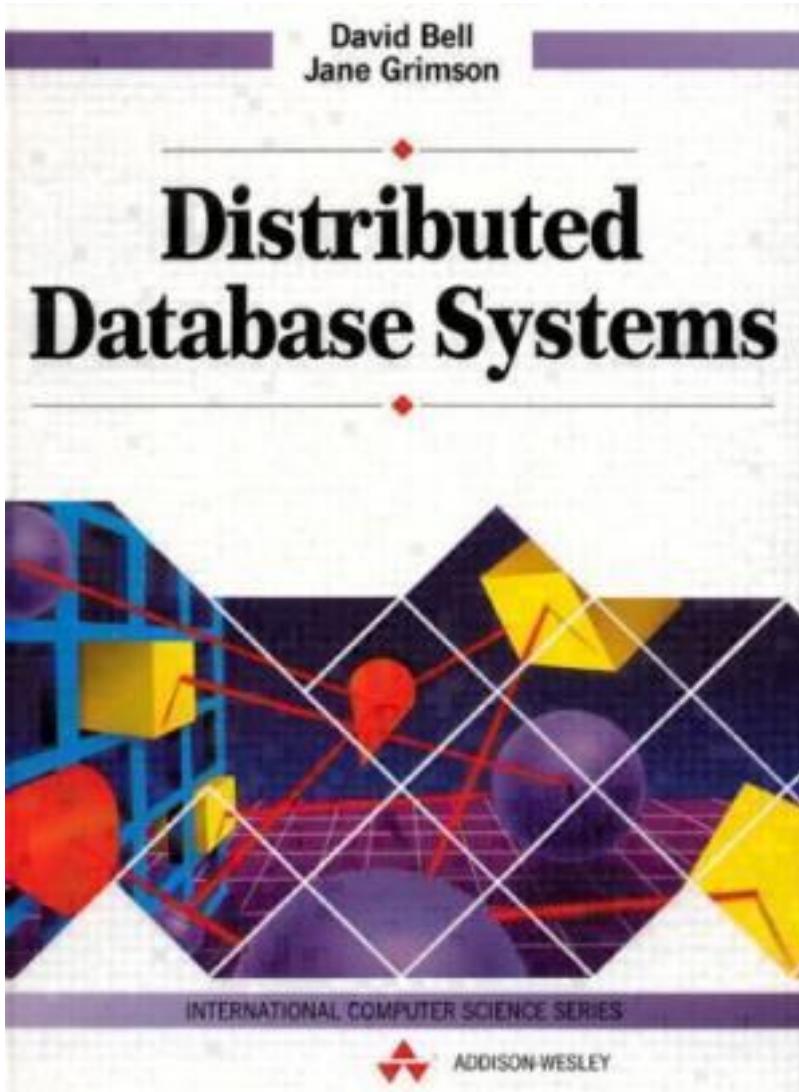
□ Ingres → [PostgreSQL](#) → Greenplum (EMC/Pivotal) → HAWQ (Oushu)



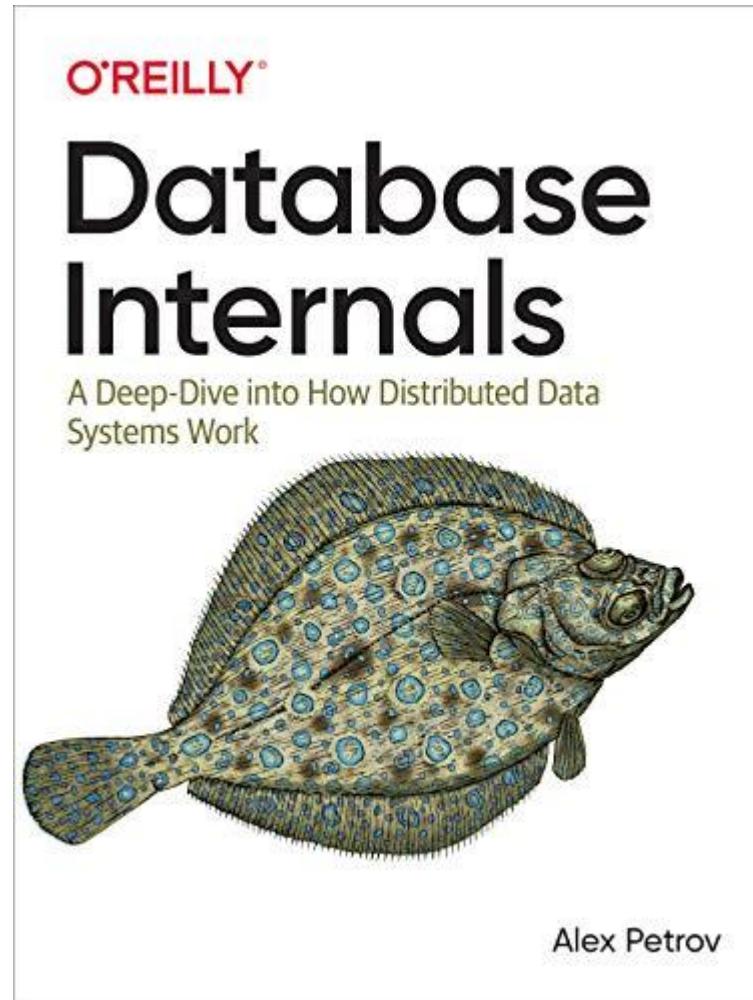
- **Principles of Distributed Database Systems**
- **作者:** M. Tamer Özsu / Patrick Valduriez
- **出版社:** Springer
- **出版年:** 2020-1-4
- **页数:** 672
- **定价:** USD 89.99
- **装帧:** Hardcover
- **ISBN:** 9783030262525



1992

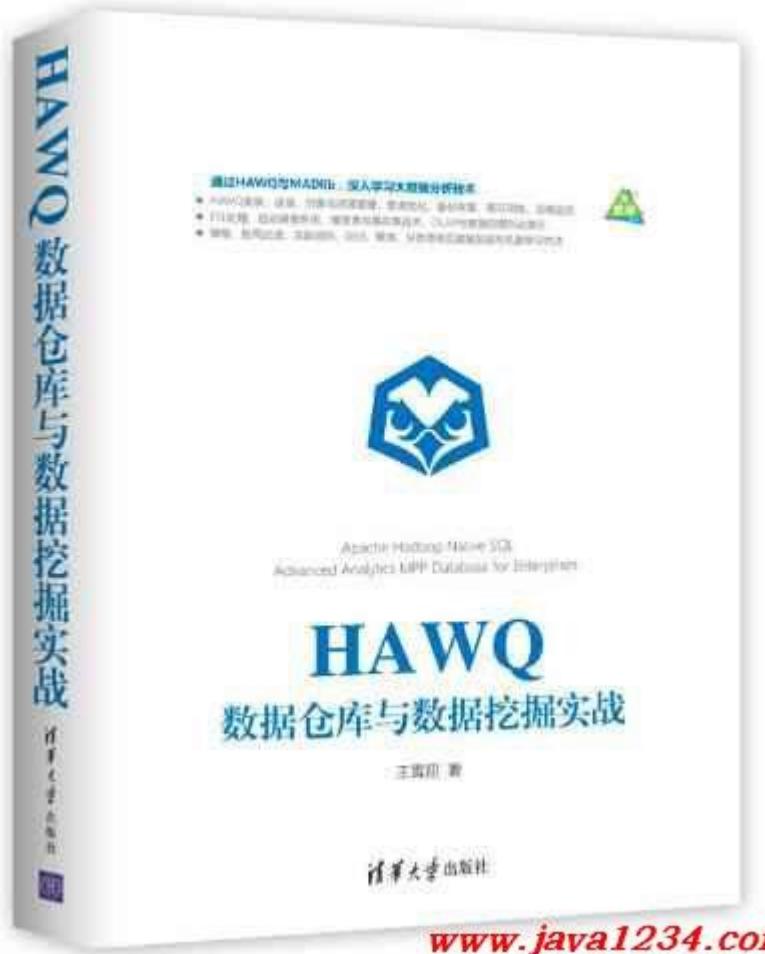


- Distributed Database Systems
- David Bell, Jane Grimson
 - Distributed database systems have now come of age with the announcement of several commercial products (Oracle, Ingres, Multi-Star) which meet the urgent need for integration of heterogeneous collections of data. This book adopts a practical approach, reviewing the fundamentals of database technology and developments in data communications (including standards) before reviewing the principles of distributed DB systems. It includes case studies of the leading products.



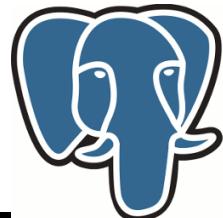
- **Database Internals: A Deep Dive into How Distributed Data Systems Work**
- **By 作者: Alex Petrov**
- **ISBN-10 书号: 1492040347**
- **ISBN-13 书号: 9781492040347**
- **Edition 版本: 1**
- **Release Finelybook 出版日期: 2019-10-15**
- **pages 页数: (376)**

Now DBMS → BD (Parallel/Distributed Data Processing)



Hadoop With Query

- **HAWQ数据仓库与数据挖掘实战**
- **王雪迎**
- **常雷做序**



PostgreSQL
the world's most advanced open source database



Pivotal
Greenplum®



HAWQ

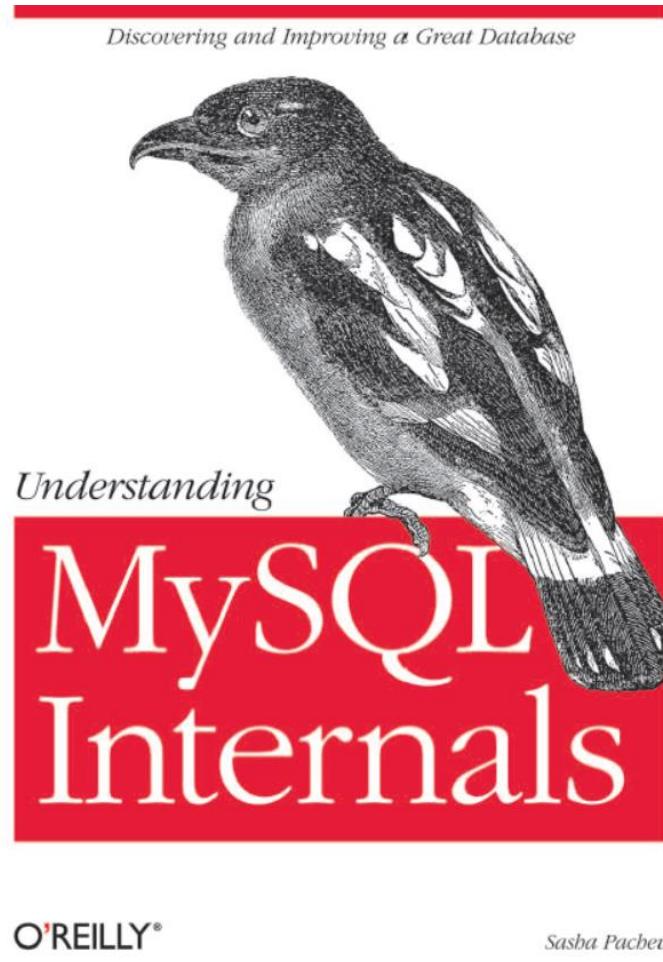
□ Ingres → PostgreSQL → Greenplum (EMC/Pivotal) → HAWQ (Oushu)

2018年8月，Apache HAWQ毕业为顶级项目，成为首个由国人主导研发的**Apache**数据库顶级项目。

HAWQ的创始人常雷博士，在2016年底创办偶数科技，推出了企业版新一代云数据仓库OushuDB，开始商业化运营。



2007



□ Understanding MySQL Internals

- 作者: Sasha Pachev
- 出版社: O'Reilly Media
- 出版年: 2007-4-17
- 页数: 258

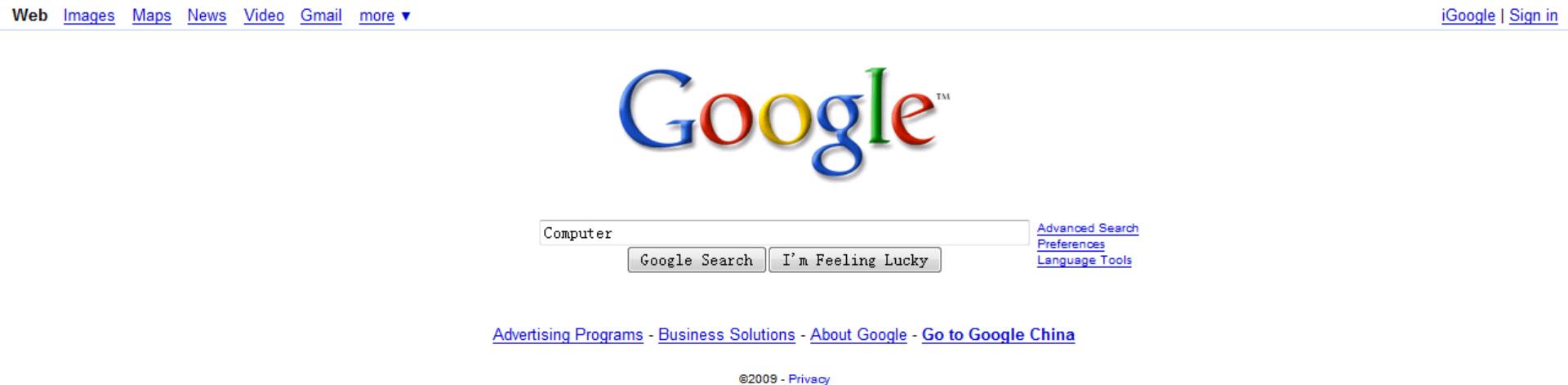
□ 深入理解MySQL核心技术

- 作者: Sasba Pacbev
- 出版社: 中国电力出版社
- 原作名: Understanding MySQL Internals
- 译者: 李芳 / 于红芸 / 邵健
- 出版年: 2009-9
- 页数: 246

Web resources

- Internet is always the source of rich information:
Google

- www.google.com, www.google.com.hk



Web resources

- Internet is always the source of rich information
- http://en.wikipedia.org/wiki/Main_Page

Make a donation to Wikipedia and give the gift of knowledge!

[article](#) [discussion](#) [view source](#) [history](#)

[Log in / create account](#)

Computer

From Wikipedia, the free encyclopedia

This article is about the machine. For other uses, see [Computer \(disambiguation\)](#).

"[Computer technology](#)" redirects here. For the company, see [Computer Technology Limited](#).

 This article **may be too long to comfortably read and navigate**. Please consider splitting content into sub-articles and using this article for a [summary](#) of the key points of the subject. (June 2009)

A computer is a machine that manipulates data according to a set of instructions.

Although mechanical examples of computers have existed through much of recorded human history, the first electronic computers were developed in the mid-20th century (1940–1945). These were the size of a large room, consuming as much power as several hundred modern personal computers (PCs).^[1] Modern computers based on integrated circuits are millions to billions of times more capable than the early machines, and occupy a fraction of the space.^[2] Simple computers are small enough to fit into a wristwatch, and can be powered by a watch battery. Personal computers in their various forms are icons of the Information Age and are what most people think of as "computers". The embedded computers found in many devices from MP3 players to fighter aircraft and from toys to industrial robots are however the most numerous.

The ability to store and execute lists of instructions called programs makes computers extremely versatile, distinguishing them from calculators. The Church–Turing thesis is a mathematical statement of this versatility: any computer with a certain minimum capability is, in principle, capable of performing the same tasks that any other computer can perform. Therefore computers ranging from a mobile phone to a supercomputer are all able to perform the same computational tasks, given enough time and storage capacity.

Contents [hide]

- 1 History of computing
- 2 Stored program architecture
 - 2.1 Programs
 - 2.2 Example
- 3 How computers work
 - 3.1 Control unit
 - 3.2 Arithmetic/logic unit (ALU)
 - 3.3 Memory
 - 3.4 Input/output (I/O)


The NASA Columbia Supercomputer.

□ 2020年2月28日11:50:51

□ 东南大学编译原理课程设计——基于python的Lex和Yacc实现

<https://github.com/seucs/compiler>

The screenshot shows the GitHub repository page for 'SEU CS Compiler Course Design: Python Implementation of LEX & YACC for Cminus'. The repository has 31 commits, 1 branch, 0 packages, 0 releases, and 2 contributors. The latest commit was made 2 years ago by user 'yanshengjia' on 5 Dec 2017. The commits listed include updates to README.md, config, graph, seulex, seuyacc, test, .gitattributes, .gitignore, README.md, and main.py.

Commit	Message	Date
Update README.md	yanshengjia	Latest commit 8fcba7e on 5 Dec 2017
refactor code		2 years ago
Add part of graph for display		2 years ago
refactor code		2 years ago
refactor code		2 years ago
refactor code		2 years ago
Added .gitattributes & .gitignore files		4 years ago
Added .gitattributes & .gitignore files		4 years ago
Update README.md		2 years ago
refactor code		2 years ago