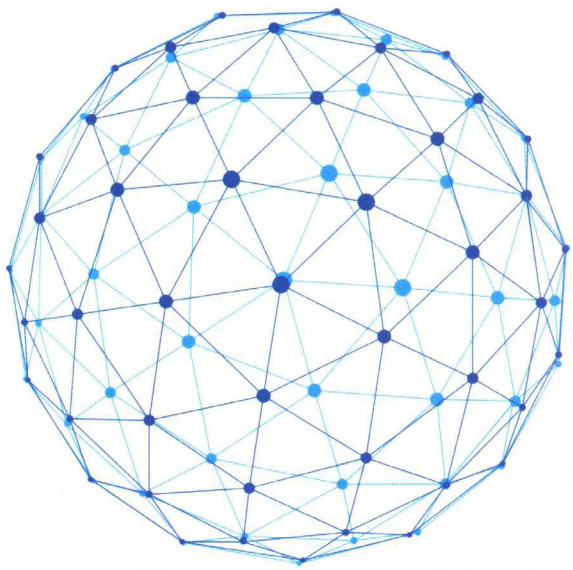
数据中国“百校工程”项目系列教材
数据科学与大数据技术专业系列规划教材

瑞翼教育

NoSQL
数据库原理与应用

王爱国 许桂秋 ◉ 主编
曾静 黄潮 张军 陈建强 刘军 ◉ 副主编



BIG DATA
Technology

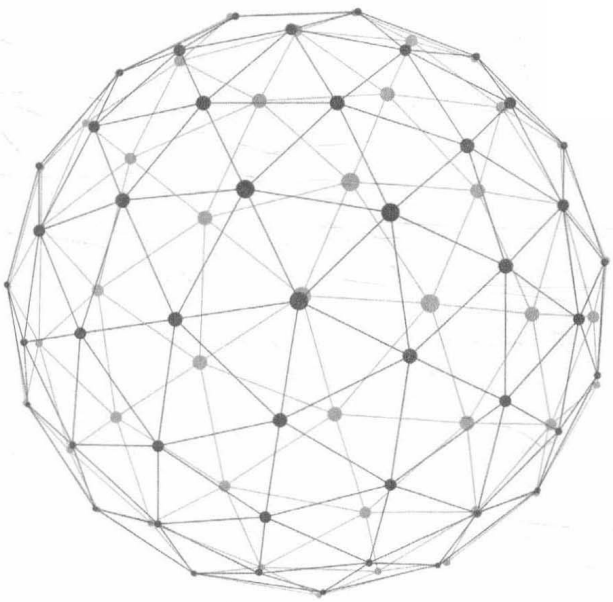
中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

数据中国“百校工程”项目系列教材
数据科学与大数据技术专业系列规划教材

瑞翼教育

NoSQL
数据库原理与应用

王爱国 许桂秋◉主编
曾静 黄潮 张军 陈建强 刘军◉副主编



BIG DATA
Technology

人 民 邮 电 出 版 社
北 京

图书在版编目（CIP）数据

NoSQL数据库原理与应用 / 王爱国，许桂秋主编. -- 北京 : 人民邮电出版社，2019.4
数据科学与大数据技术专业系列规划教材
ISBN 978-7-115-50350-3

Ⅰ. ①N… Ⅱ. ①王… ②许… Ⅲ. ①关系数据库系统－教材 Ⅳ. ①TP311.132.3

中国版本图书馆CIP数据核字(2019)第022495号

内 容 提 要

本书系统全面地介绍了 NoSQL 数据库的理论、技术与开发方法。

全书共 9 章，主要内容包括 NoSQL 数据库的发展历程以及它与传统关系型数据库相比所具有的优势、HBase 分布式数据库技术的原理与实践、MongoDB 分布式数据库技术的原理和实践、Memcached 和 Redis 技术、NewSQL 数据库技术，以及 MongoDB 和 HBase 数据库技术的综合实验。

本书适合作为高校 NoSQL 数据库技术课程的教材。

◆ 主　　编　王爱国　许桂秋
副 主 编　曾　静　黄　潮　张　军　陈建强　刘　军
责任编辑　张　斌
责任印制　陈　犇

◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路 11 号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
固安县铭成印刷有限公司印刷

◆ 开本：787×1092　1/16
印张：13　　2019 年 4 月第 1 版
字数：263 千字　　2019 年 4 月河北第 1 次印刷

定价：55.00 元

读者服务热线：(010)81055256　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广登字 20170147 号

前　言

数据是当今世界最有价值的资产之一。在大数据时代，人们生产、收集数据的能力大大提升。但是传统的关系型数据库在可扩展性、数据模型和可用性方面已远远不能满足当前的数据处理需求，因此，各种 NoSQL 数据库系统应运而生。NoSQL 数据库不像关系型数据库那样都有相同的特点，遵循相同的标准。NoSQL 数据库类型多样，可满足不同场景的应用需求，因此取得了巨大的成功。

NoSQL 数据库基本理念是以牺牲事务机制和强一致性机制，来获取更好的分布式部署能力和横向扩展能力，创造出新的数据模型，使其在不同的应用场景下，对特定业务数据具有更强的处理性能。

本书共 9 章，通过介绍 NoSQL 数据库的分布式架构、数据模型、数据原理等，让读者能够理解 NoSQL 数据库的特点，并在具体的应用场景中选择合适的数据库。通过学习 HBase 和 MongoDB 的数据操作方法以及编程方法，读者将学会在实际应用中运用这些原理。

本书建议安排教学课时为 64 学时，教师可根据学生的接受能力以及高校的培养方案选择教学内容。

本书中的部分实践内容由朱晨楠、王志鹏、张蓓蓓、熊福棚、李学博、陈冲、杨瑞和王永等同学实际操作验证通过，在此对他们表示感谢！

由于编者的能力和水平有限，写作时间仓促，书中难免存在疏漏或欠妥之处，恳请读者批评指正。

编　者

2019 年 1 月

目 录

第1章 绪论

数据管理经历了人工管理、文件系统、数据库系统三个阶段。人工管理阶段和文件系统阶段的数据共享性差，冗余度较高，数据库系统的出现解决了这两方面的问题。但是随着互联网技术的发展，数据库系统管理的数据及其应用环境发生了很大的变化，主要表现为应用领域越来越广泛，数据种类越来越复杂和多样，而且数据量剧增。在大数据时代的场景下，传统的关系型数据库已无法满足用户需求，NoSQL 数据库应运而生。本章首先介绍数据库的基本概念，然后分析关系型数据库在数据存储和管理上存在的问题，在此基础上引出 NoSQL 数据库，并重点将关系型数据库与 NoSQL 数据库的技术特点做对比，分析 NoSQL 数据库处理数据的优势，最后介绍几种比较常用的 NoSQL 数据库。

本章涉及一些数据库原理的知识，已具备这些知识的读者可有选择地学习。

本章的重点内容如下。

（1）数据库系统。

（2）分布式数据库的数据管理。

（3）ACID 与 BASE。

（4）NoSQL 数据库分类。

1.1 数据库系统

在信息化社会，充分有效地管理和利用各类信息资源，是进行科学研究和决策管理的前提条件。数据库技术是管理信息系统、办公自动化系统、决策支持系统等各类信息系统的核心部分，

是进行科学研究和决策管理的重要技术手段。

1.1.1 数据库系统的基本概念

数据库技术是研究数据库的结构、存储、设计、管理和使用的一门科学。数据库系统（Database System，DBS）是采用数据库技术的计算机系统，它是由计算机硬件、软件和数据资源组成的系统，能实现有组织地、动态地存储大量关联数据，并方便多用户访问。数据库系统由用户、数据库应用程序、数据库管理系统（DataBase Management System，DBMS）和数据库（Database，DB）组成，如图 1-1 所示。

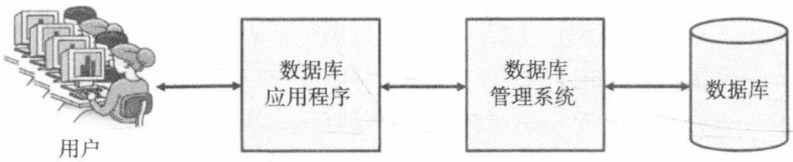


图 1-1　数据库系统

1. 数据库

数据库是长期存储在计算机内的、有组织的、统一管理的、可以表现为多种形式的、可共享的数据集合。这里“共享”是指数据库中的数据，可为多个不同的用户、使用多种不同的语言、为了不同的目的而同时存取，甚至同一数据也可以同时存取；“集合”是指某特定应用环境中的各种应用的数据及其之间的联系全部集中按照一定的结构形式进行存储。数据库中的数据按一定的数据模型组织、描述和存储，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户所共享。

在数据库技术中，用数据模型（Data Model）的概念描述数据库的结构和语义，对现实世界的数据进行抽象。数据库根据不同的逻辑模型可分成三种：层次型、网状型和关系型。

（1）层次型数据模型

早期的数据库多采用层次型数据模型，称为层次型数据库，如图 1-2 所示，它用树形（层次）结构表示实体类型及实体间的联系。在这种树形结构中，数据按自然的层次关系组织起来，以反映数据之间的隶属关系，树中的节点是记录类型，每个非根节点都只有一个父节点，而父节点可同时拥有多个子节点，父节点和子节点的联系是 1∶N 的联系。正因为层次型数据模型的构造简单，在多数的实际问题中，数据间关系如果简单地通过树形结构表示，则会造成数据冗余度过高，所以层次型数据模型逐渐被淘汰。

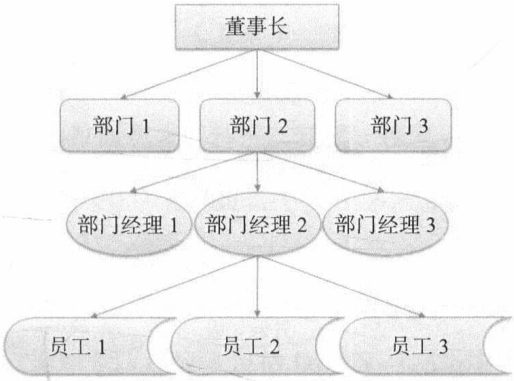


图 1-2　层次型数据库

（2）网状型数据模型

采用网状型数据模型的数据库称为网状型数据库，通过网络结构表示数据间联系，如图 1-3 所示。图中的节点代表数据记录，连线描述不同节点数据间的联系。这种数据模型的基本特征是，节点数据之间没有明确的从属关系，一个节点可与其他多个节点建立联系，即节点之间的联系是任意的；任何两个节点之间都能发生联系，可表示多对多的关系。在网状型数据模型中，数据节点之间的关系比较复杂，而且随着应用范围的扩展，数据库的结构变得越来越复杂，不利于用户掌握。

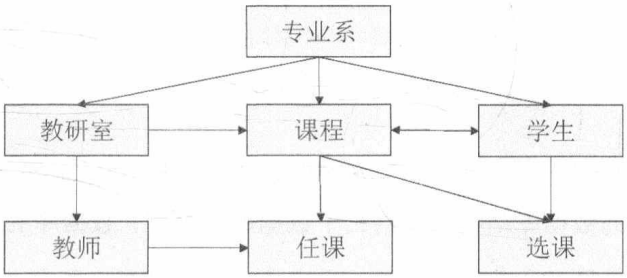


图 1-3　网状型数据库

（3）关系型数据模型

关系型数据模型开发较晚。1970 年，IBM 公司的研究员埃德加·弗兰克·科德（Edgar Frank Codd）在 *Communication of the ACM* 上发表了一篇名为 *A Relational Model of Data for Large Shared Data Banks* 的论文，提出了关系型数据模型的概念，奠定了关系型数据模型的理论基础。它是通过满足一定条件的二维表格来表示实体集合以及数据间联系的一种模型，如图 1-4 所示，学生、课程和教师是实体集合，选课和任课是实体间的联系，实体和实体间的联系均通过二维表格来描述。关系型数据模型具有坚实的数学基础与理论基础，使用灵活方便，适应面广，因此发展十分迅速。目前流行的一些数据库系统，如 Oracle、Sybase、Ingress、Informix 等都属于关系型数据库。

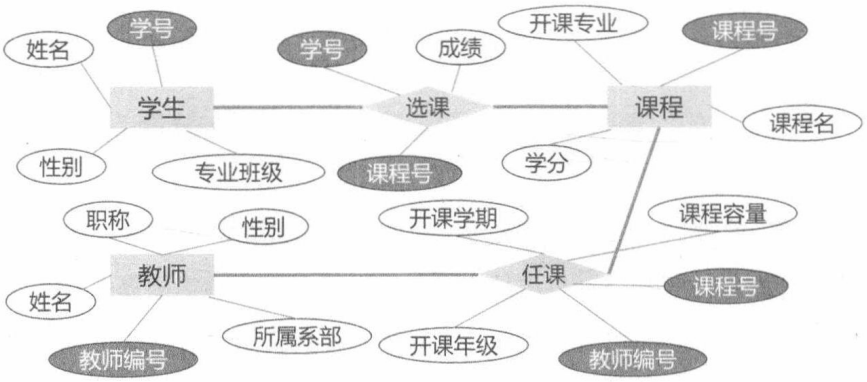


图 1-4　关系型数据库

2. 数据库管理系统

数据库管理系统（DBMS）是一种操纵和管理数据库的大型软件，用于建立、使用和维护数据库。DBMS 是一个庞大且复杂的产品，几乎都是由软件供应商授权提供的，如 Oracle 公司的 Oracle 和 MySQL、IBM 公司的 DB2、Microsoft 公司的 Access 和 SQL Server，这些 DBMS 占据了大部分的市场份额。

DBMS 对数据库进行统一管理和控制，以保证数据库的安全性和完整性。用户通过 DBMS 访问数据库中的数据，数据库管理员也通过 DBMS 进行数据库的维护工作。DBMS 允许多个应用程序或多个用户使用不同的方法，在同一时刻或不同时刻去建立、修改和询问数据库。DBMS 的主要功能如下。

（1）数据定义

DBMS 提供数据定义语言（Data Definition Language，DDL），供用户定义、创建和修改数据库的结构。DDL 所描述的数据库结构仅仅给出了数据库的框架，数据库的框架信息被存放在系统目录中。

（2）数据操纵

DBMS 提供数据操纵语言（Data Manipulation Language，DML），实现用户对数据的操纵功能，包括对数据库数据的插入、删除、更新等操作。

（3）数据库的运行管理

DBMS 提供数据库的运行控制和管理功能，包括多用户环境下的事务的管理和自动恢复、并发控制和死锁检测、安全性检查和存取控制、完整性检查和执行、运行日志的组织管理等。这些功能保证了数据库系统的正常运行。

（4）数据组织、存储与管理

DBMS 要分类组织、存储和管理各种数据，就需要确定以何种文件结构和存取方式来组织这些数据，实现数据之间的联系。数据组织和存储的基本目标是提高存储空间的利用率，选择合适

的存取方法提高存取效率。

（5）数据库的维护

数据库的维护包括数据库的数据载入、转换、转储、恢复，数据库的重组织和重构，以及性能监控分析等功能，这些功能分别由各个应用程序来完成。

（6）通信

DBMS 有接口负责处理数据的传送。这些接口与操作系统的联机处理以及分时系统和远程作业输入相关。网络环境下的数据库系统还应该包括 DBMS 与网络中其他软件系统的通信功能以及数据库之间的互操作功能。

DBMS 是数据库系统的核心，是管理数据库的软件。DBMS 是实现把用户视角下的、抽象的逻辑数据处理，转换成为计算机中具体的物理数据处理的软件。有了 DBMS，用户可以在抽象意义下处理数据，而不必考虑这些数据在计算机中的布局和物理位置。

3. 应用程序

数据库系统还包括数据库应用程序。应用程序最终是面向用户的，用户可以通过应用程序输入和处理数据库中的数据。例如，在学校选课系统中，管理员用户可以创建课程信息，学生用户可以修改课程信息，应用程序将这些操作提交给 DBMS，由 DBMS 将这种用户级别的操作转化成数据库能识别的 DDL。应用程序还能够处理用户的查询，比如学生查询星期一有哪些课程，应用程序首先生成一个课程查询请求，并发送给 DBMS，DBMS 从数据库中查询结果并格式化后返回给用户。

1.1.2　关系型数据库

关系型数据库建立在关系型数据模型的基础上，是借助于集合代数等数学概念和方法来处理数据的数据库。现实世界中的各种实体以及实体之间的各种联系均可用关系模型来表示，市场上占很大份额的 Oracle、MySQL、DB2 等都是面向关系模型的 DBMS。

1. 关系型数据库基本概念

在关系型数据库中，实体以及实体间的联系均由单一的结构类型来表示，这种逻辑结构是一张二维表。图 1-4 所示的学生选课系统中，实体和实体间联系在数据库中的逻辑结构可通过图 1-5 所示。

关系型数据库以行和列的形式存储数据，这一系列的行和列被称为表，一组表组成了数据库。图 1-6 所示的员工信息表就是关系型数据库。

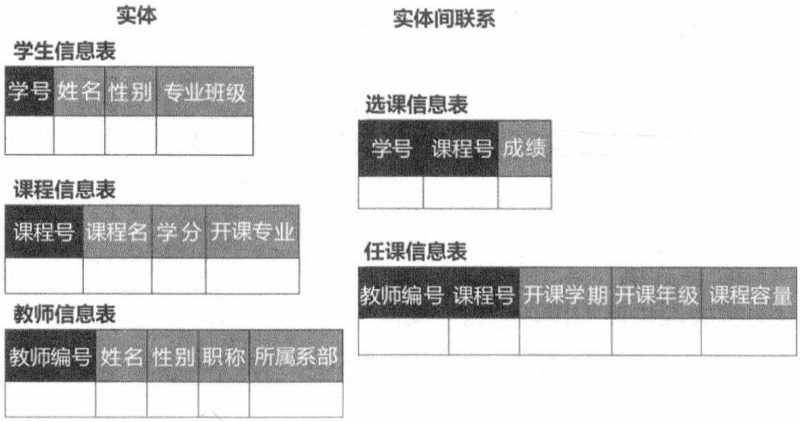


图 1-5　学生选课系统数据库逻辑结构

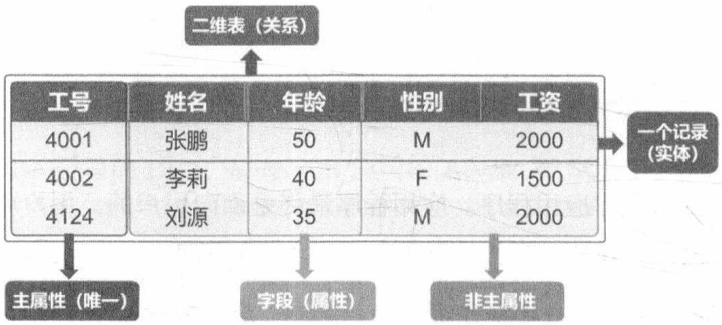


图 1-6　员工信息表

（1）二维表：也称为关系，它是一系列二维数组的集合，用来代表与存储数据对象之间的关系。它由纵向的列和横向的行组成。

（2）行：也叫元组或记录，在表中是一条横向的数据集合，代表一个实体。

（3）列：也叫字段或属性，在表中是一条纵行的数据集合。列也定义了表中的数据结构。

（4）主属性：关系中的某一属性组，若它们的值唯一地标识一个记录，则称该属性组为主属性或主键。主属性可以是一个属性，也可以由多个属性共同组成。在图 1-5 中，学号是学生信息表的主属性，但是课程信息表中，学号和课程号共同唯一地标识了一条记录，所以学号和课程号一起组成了课程信息表的主属性。

2. 结构化查询语言

关系型数据库的核心是其结构化的查询语言（Structured Query Language，SQL），SQL 涵盖了数据的查询、操纵、定义和控制，是一个综合的、通用的且简单易懂的数据库管理语言。同时 SQL 又是一种高度非过程化的语言，数据库管理者只需要指出做什么，而不需要指出该怎么做即可完成对数据库的管理。SQL 可以实现数据库全生命周期的所有操作，所以 SQL 自产生之日起就成了检验关系型数据库管理能力的“试金石”，SQL 标准的每一次变更和完善都引导着关系型数据库产品的发展方向。SQL 包含以下四个部分。

（1）数据定义语言（DDL）

DDL 包括 CREATE、DROP、ALTER 等动作。在数据库中使用 CREATE 来创建新表，DROP 来删除表，ALTER 负责数据库对象的修改。例如，创建学生信息表使用以下命令：

```
CREATE TABLE StuInfo(id int(10) NOT NULL,PRIMARY KEY(id),name
varchar(20),female bool,class varchar(20));
```

（2）数据查询语言（Data Query Language，DQL）

DQL 负责进行数据查询，但是不会对数据本身进行修改。DQL 的语法结构如下：

```
SELECT FROM 表名 1，表 2
where 查询条件 #可以组合 and、or、not、=、between、and、in、like 等;
group by 分组字段
having(分组后的过滤条件)
order by 排序字段和规则;
```

（3）数据操纵语言（Data Manipulation Language，DML）

DML 负责对数据库对象运行数据访问工作的指令集，以 INSERT、UPDATE、DELETE 三种指令为核心，分别代表插入、更新与删除。向表中插入数据命令如下：

```
INSERT 表名 (字段 1,字段 2,……,字段 n,) VALUES (字段 1 值,字段 2 值,……,字段 n 值)
 where 查询条件;
```

（4）数据控制语言（Data Control Language，DCL）

DCL 是一种可对数据访问权进行控制的指令。它可以控制特定用户账户对查看表、预存程序、用户自定义函数等数据库操作的权限，由 GRANT 和 REVOKE 两个指令组成。DCL 以控制用户的访问权限为主，GRANT 为授权语句，对应的 REVOKE 是撤销授权语句。

3. 关系型数据库的优缺点

关系型数据库已经发展了数十年，其理论知识、相关技术和产品都趋于完善，是目前世界上应用最广泛的数据库系统。

（1）关系型数据库的优点

① 容易理解：二维表结构非常贴近逻辑世界的概念，关系型数据模型相对层次型数据模型和网状型数据模型等其他模型来说更容易理解。

② 使用方便：通用的 SQL 使用户操作关系型数据库非常方便。

③ 易于维护：丰富的完整性大大减少了数据冗余和数据不一致的问题。关系型数据库提供对事务的支持，能保证系统中事务的正确执行，同时提供事务的恢复、回滚、并发控制和死锁问题的解决。

（2）关系型数据库的缺点

随着各类互联网业务的发展，关系型数据库难以满足对海量数据的处理需求，存在以下不足。

① 高并发读写能力差：网站类用户的并发性访问非常高，而一台数据库的最大连接数有限，且硬盘 I/O 有限，不能满足很多人同时连接。

② 对海量数据的读写效率低：若表中数据量太大，则每次的读写速率都将非常缓慢。

③ 扩展性差：在一般的关系型数据库系统中，通过升级数据库服务器的硬件配置可提高数据处理的能力，即纵向扩展。但纵向扩展终会达到硬件性能的瓶颈，无法应对互联网数据爆炸式增长的需求。还有一种扩展方式是横向扩展，即采用多台计算机组成集群，共同完成对数据的存储、管理和处理。这种横向扩展的集群对数据进行分散存储和统一管理，可满足对海量数据的存储和处理的需求。但是由于关系型数据库具有数据模型、完整性约束和事务的强一致性等特点，导致其难以实现高效率的、易横向扩展的分布式架构。

1.1.3 NoSQL 数据库的特点

NoSQL 数据库最初是为了满足互联网的业务需求而诞生的。互联网数据具有大量化、多样化、快速化等特点。在信息化时代背景下，互联网数据增长迅猛，数据集合规模已实现从 GB、PB 到 ZB 的飞跃。数据不仅仅是传统的结构化数据，还包含了大量的非结构化和半结构化数据，关系型数据库无法存储此类数据。因此，很多互联网公司着手研发新型的、非关系型的数据库，这类非关系型数据库统称为 NoSQL 数据库，其主要特点如下。

1. 灵活的数据模型

互联网数据如网站用户信息、地理位置数据、社交图谱、用户产生的内容、机器日志数据以及传感器数据等，正在快速改变着人们的通信、购物、广告、娱乐等日常生活，没有使用这些数据的应用很快就会被用户所遗忘。开发者希望使用非常灵活的数据库，容纳新的数据类型，并且不会被第三方数据提供商的数据结构变化所影响。关系型数据库的数据模型定义严格，无法快速容纳新的数据类型。例如，若要存储客户的电话号码、姓名、地址、城市等信息，则 SQL 数据库需要提前知晓要存储的是什么。这对于敏捷开发模式来说十分不方便，因为每次完成新特性时，通常都需要改变数据库的模式。NoSQL 数据库提供的数据模型则能很好地满足这种需求，各种应用可以通过这种灵活的数据模型存储数据而无须修改表；或者只需增加更多的列，无须进行数据的迁移。

2. 可伸缩性强

对企业来说，关系型数据库一开始是普遍的选择。然而，在使用关系型数据库的过程中却遇到了越来越多的问题，原因在于它们是中心化的，是纵向扩展而不是横向扩展的。这使得它们不适合那些需要简单且动态可伸缩性的应用。NoSQL 数据库从一开始就是分布式、横向扩展的，因

此非常适合互联网应用分布式的特性。在互联网应用中，当数据库服务器无法满足数据存储和数据访问的需求时，只需要增加多台服务器，将用户请求分散到多台服务器上，即可减少单台服务器的性能瓶颈出现的可能性。

3. 自动分片

由于关系型数据库存储的是结构化的数据，所以通常采用纵向扩展，即单台服务器要持有整个数据库来确保可靠性与数据的持续可用性。这样做的代价是非常昂贵的，而且扩展也会受到限制。针对这种问题的解决方案就是横向扩展，即添加服务器而不是扩展单台服务器的处理能力。NoSQL 数据库通常都支持自动分片，这意味着它们会自动地在多台服务器上分发数据，而不需要应用程序增加额外的操作。

4. 自动复制

NoSQL 数据库支持自动复制。在 NoSQL 数据库分布式集群中，服务器会自动对数据进行备份，即将一份数据复制存储在多台服务器上。因此，当多个用户访问同一数据时，可以将用户请求分散到多台服务器中。同时，当某台服务器出现故障时，其他服务器的数据可以提供备份，即 NoSQL 数据库的分布式集群具有高可用性与灾备恢复的能力。

1.2　分布式数据库的数据管理

大数据需要通过分布式的集群方式来解决存储和访问的问题。本节将从分布式的角度来介绍数据库的数据管理。分布式系统的核心理念是让多台服务器协同工作，完成单台服务器无法处理的任务，尤其是高并发或者大数据量的任务。分布式数据库是数据库技术与网络技术相结合的产物，它通过网络技术将物理上分开的数据库连接在一起，进行逻辑层面上的集中管理。在分布式数据库系统中，一个应用程序可以对数据库进行透明操作，数据库中的数据分别存储在不同的局部数据库中，由不同机器上不同的 DBMS 进行管理。分布式数据库的体系结构如图 1-7 所示。

1.2.1　分布式数据处理

分布式数据处理使用分而治之的办法来解决大规模数据管理问题。它处理数据的基本特点如下。

1. 分布的透明管理

在分布式系统中，数据不是存储在一个场地上，而是存储在计算机网络的多个场地上。但逻辑上是一个整体，它们被所有用户共享，并由一个 DBMS 统一管理。用户访问数据时无须指出数

据存放在哪里，也不需要知道由分布式系统中的哪台服务器来完成。

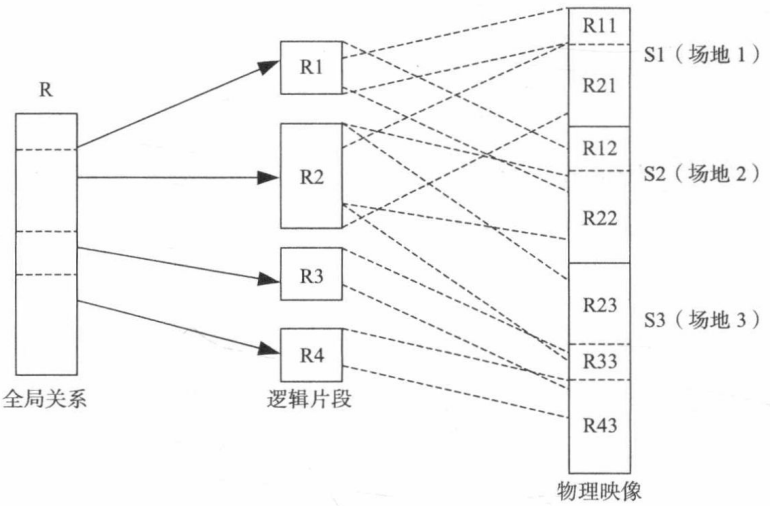


图 1-7　分布式数据处理体系结构

2. 复制数据的透明管理

分布式数据的复制有助于提高性能，更易于协调不同而又冲突的用户需求。同时，当某台服务器出现故障时，此服务器上的数据在其他服务器上还有备份，提高了系统的可用性。这种多副本的方式对用户来说是透明的，即用户不需要知道副本的存在，由系统统一管理、协调副本的调用。

3. 事务的可靠性

分布式数据处理具有重复的构成，因此消除了单点故障的问题，即系统中一个或多个服务器发送故障不会使整个系统瘫痪，从而提高了系统的可靠性。但是在分布式系统中，事务是并发的，即不同用户可能在同一时间对同一数据源进行访问，这就要求系统支持分布式的并发控制，保证系统中数据的一致。

分布式系统可以解决海量数据的存储和访问，但是在分布式环境下，数据库会遇到更为复杂的问题，举例如下。

（1）数据在分布式环境下以多副本方式进行存储，那么，在为用户提供数据访问时如何选择一个副本，或者用户修改了某一副本的数据，如何让系统中每个副本都得到更新。

（2）如果正在更新系统所有副本信息时，某个服务器由于网络或硬、软件功能出现问题导致其发生故障。在这种情况下，如何确保故障恢复时，此服务器上的副本与其他副本一致。

这些问题给分布式数据库管理系统带来了挑战，它们是分布式系统固有的复杂性，但更重要的是对分布数据的管理，控制数据之间的一致性以及数据访问的安全性。

1.2.2　CAP 理论

CAP 理论是指在一个分布式系统中，一致性（Consistency，C）、可用性（Availability，A）、分区容错性（Partition Tolerance，P）三者不可兼得。

1. 基本概念

（1）一致性（C）

一致性是指“all nodes see the same data at the same time”，即更新操作成功后，所有节点在同一时间的数据完全一致。一致性可以分为客户端和服务端两个不同的视角。从客户端角度来看，一致性主要指多个用户并发访问时更新的数据如何被其他用户获取的问题；从服务端来看，一致性则是用户进行数据更新时如何将数据复制到整个系统，以保证数据的一致。一致性是在并发读写时才会出现的问题，因此在理解一致性的问题时，一定要注意结合考虑并发读写的场景。

（2）可用性（A）

可用性是指“reads and writes always succeed”，即用户访问数据时，系统是否能在正常响应时间返回结果。好的可用性主要是指系统能够很好地为用户服务，不出现用户操作失败或者访问超时等用户体验不好的情况。在通常情况下，可用性与分布式数据冗余、负载均衡等有着很大的关联。

（3）分区容错性（P）

分区容错性是指“the system continues to operate despite arbitrary message loss or failure of part of the system”，即分布式系统在遇到某节点或网络分区故障的时候，仍然能够对外提供满足一致性和可用性的服务。

分区容错性和扩展性紧密相关。在分布式应用中，可能因为一些分布式的原因导致系统无法正常运转。分区容错性高指在部分节点故障或出现丢包的情况下，集群系统仍然能提供服务，完成数据的访问。分区容错可视为在系统中采用多副本策略。

2. 相互关系

CAP 理论认为分布式系统只能兼顾其中的两个特性，即出现 CA、CP、AP 三种情况，如图 1-8 所示。

（1）CA without P

如果不要求 Partition Tolerance，即不允许分区，则强一致性和可用性是可以保证的。其实分区是始终存在的问题，因此 CA 的分布式系统更多的是允许分区后各子系统依然保持 CA。

（2）CP without A

如果不要求可用性，相当于每个请求都需要在各服务器之间强一致，而分区容错性会导致同步时间无限延长，如此 CP 也是可以保证的。很多传统的数据库分布式事务都属于这种模式。