

数据类型

结构化数据 (Structured Data): 具备严格的模式, 可被组织进关系型数据库中。

非结构化数据 (Unstructured Data): 无需预先定义, 数据格式不固定。

文件系统与数据库系统的对比 (File Systems vs Database Systems)

文件系统缺点包括: 数据冗余、不一致性、数据隔离、完整性问题、并发访问问题。

1. 数据冗余 (Data Redundancy)

- 专业定义: 同样的数据在多个文件中重复存储
- 造成的问题: 存储空间浪费、数据更新困难、可能导致数据不一致

2. 数据不一致性(Data Inconsistency)

- 专业定义: 同一数据在不同位置的多个副本之间存在矛盾
- 例如: 同一客户信息在不同文件中的地址不同

3. 数据隔离(Data Isolation)

- 专业定义: 数据分散在不同文件中, 难以访问和组合查询
- 影响: 难以生成综合报表或进行复杂查询

4. 完整性问题(Data Integrity Problems)

- 专业定义: 难以在文件系统层面实施统一的完整性约束

- 包括:

- 实体完整性 (Entity Integrity)
- 参照完整性 (Referential Integrity)
- 域完整性 (Domain Integrity)

5. 并发访问问题(Concurrent Access Problems)

- 专业定义: 多用户同时访问和修改数据时的同步控制问题
- 英文表述: Concurrent Access Control
- 相关概念:
 - 并发控制 (Concurrency Control)
 - 死锁 (Deadlock)
 - 事务处理 (Transaction Processing)

这些问题在现代数据库管理系统(DBMS)中通过以下机制得到解决:

1. 数据字典(Data Dictionary): 控制冗余
2. 事务管理(Transaction Management): 保证一致性
3. 统一管理(Centralized Management): 解决隔离问题
4. 约束机制(Constraint Mechanism): 保证完整性
5. 并发控制(Concurrency Control): 处理并发访问

数据库管理系统的特性

数据独立性 (Data Independence)

高效的数据访问 (Efficient Data Access)

数据完整性与安全性 (Data Integrity and Security)

数据管理 (Data Administration)

并发访问与崩溃恢复 (Concurrent Access and Crash Recovery)

数据库设计 (Database Design)

概念设计 (Conceptual Design): 利用工具进行需求表示, 便于维护和转化为数据库实现。

逻辑设计 (Logical Design): 通过概念设计转换为数据模型并实现于 DBMS。

物理设计 (Physical Design): 进一步规范数据库的存储与访问。

数据库语言 (Database Languages)

数据定义语言 (DDL): 用于定义概念模式。

数据操作语言 (DML): 用于请求和操作数据, 有非过程性 DML (如 SQL) 和过程性 DML。

数据库用户类型:

数据库管理员 (Database Administrator, DBA): 负责安全、授权、数据恢复和数据库调优。应用程序员 (Application Programmer): 实现具体需求。最终用户 (End User): 实际使用数据库的用户。

数据库模型:

数据模型的级别包括: 高层次或概念模型 (如 ER 模型)、实现模型 (如关系模型) 和物理模型 (低层次)。

数据库管理系统类型:

主要包括关系型 (Relational)、键值型 (Key/Value)、图形型 (Graph)、文档型 (Document)、列族型 (Column-family)。

数据库历史 (History of Database Systems)

1960s: 通用数据库管理系统 (Integrated Data Store)。

1970s: 关系模型的提出 (Edgar Codd)。

1980s: SQL 标准化及事务概念提出 (Jim Gray)。

1990s 至今: 关系型数据库、NoSQL、大数据分布式处理的发展。

数据建模 (Data Modelling):

概念模型 (Conceptual Model): 抽象的高层数据模型, 例如 ER 模型和 ODL (Object Data Language), 用户友好。

逻辑模型 (Logical Model): 为具体的数据库管理系统(DBMS)实现的模型, 例如关系模型。

物理模型 (Physical Model): 在具体 DBMS 内部的文件存储方式。

设计阶段 (Design Stages): 概念设计 (Conceptual Design)、逻辑设计 (Logical Design)、物理设计 (Physical Design)