第5章 数据库存储

苏州大学 费子成

feizicheng@suda.edu.cn
https://web.suda.edu.cn/feizicheng/

本章内容

1. 数据库系统如何存储数据
2. 数据的存储介质特性比较
3. 数据库系统如何查找数据
4. 如何提高数据访问的效率
5. 如何防止数据库数据丢失
6. 数据的最小单位数据页面
7. 数据库如何来管理缓冲区
8. 如何选择行存储和列存储

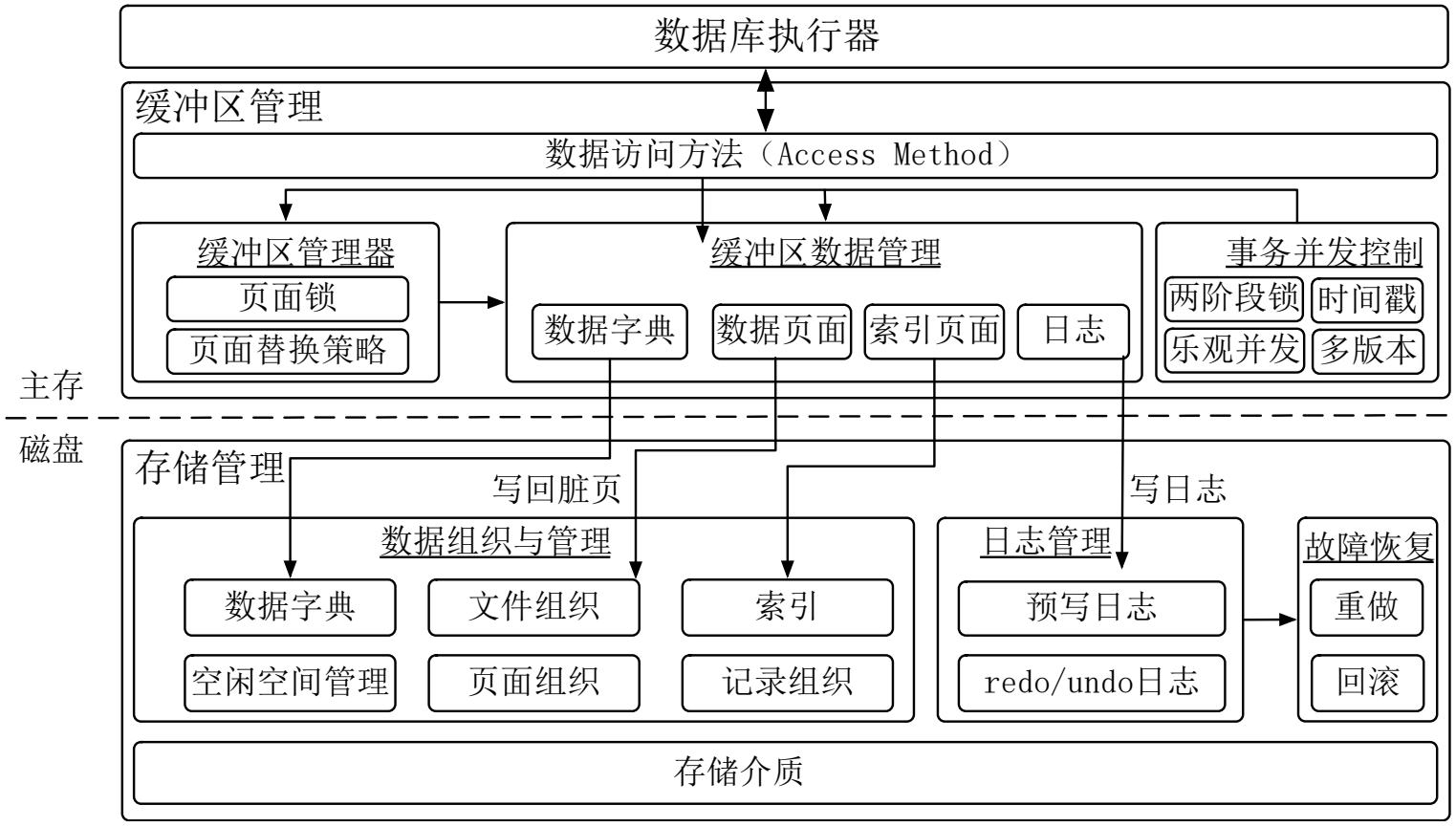
目录

目录

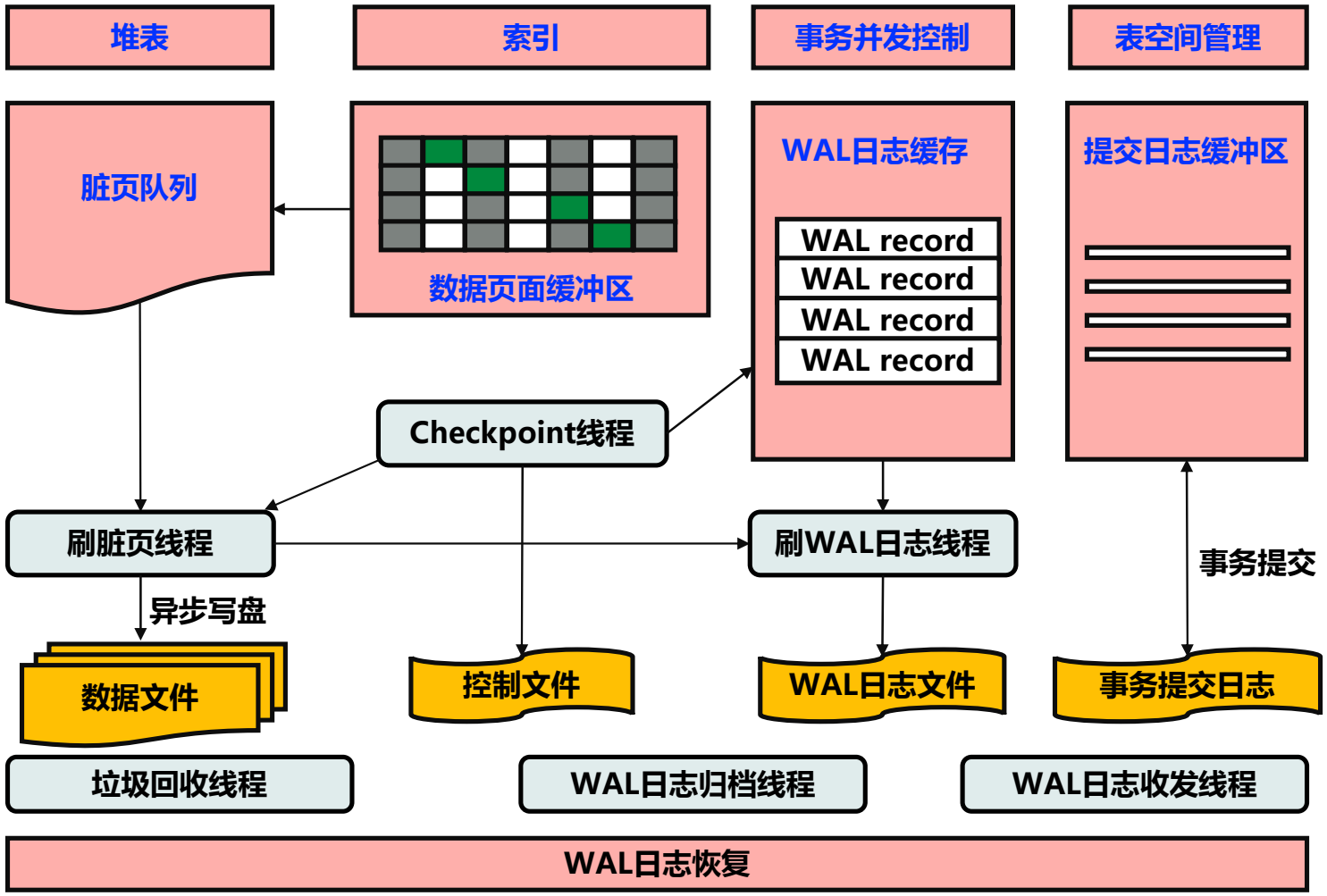
存储概览

SQL引擎
执行器
存储引擎 **文件组织 页面组织 元数据 日志 索引 缓冲区 故障恢复 并发控制**
存储介质

存储概览

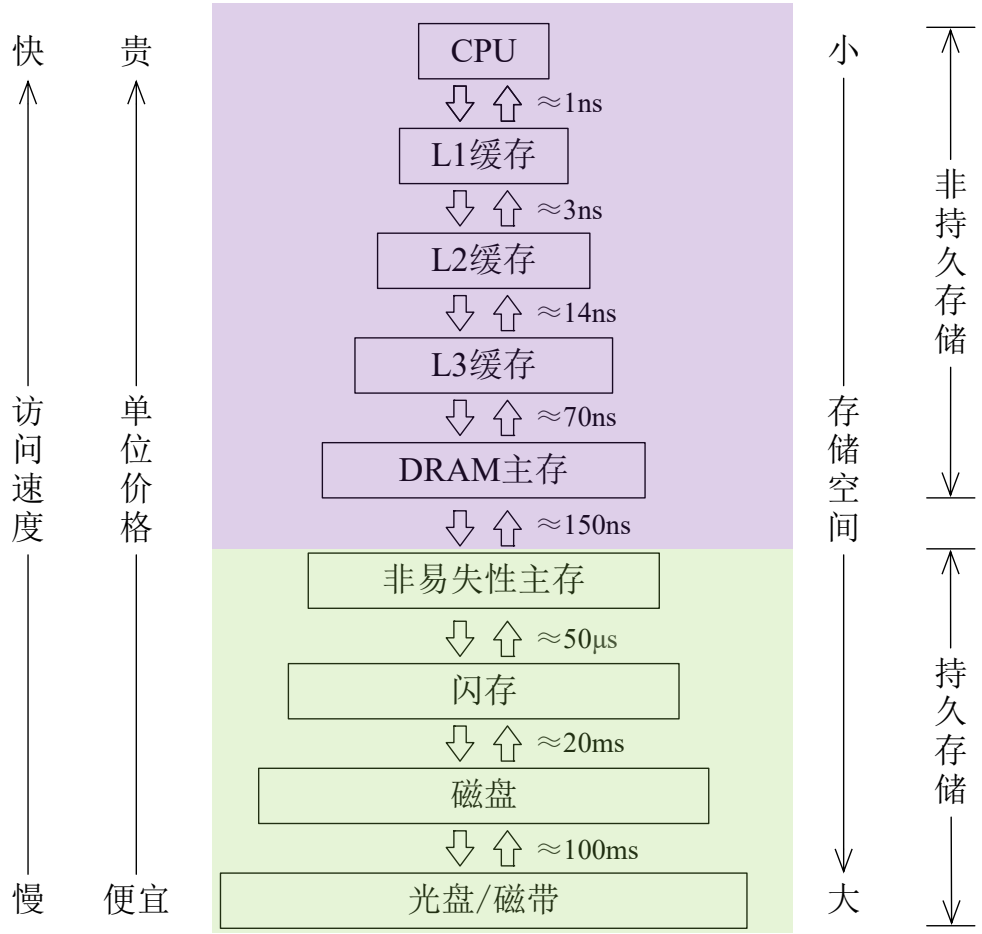

数据库执行器
缓冲区管理
数据访问方法（Access Method）
缓冲区管理器
页面锁
页面替换策略
缓冲区数据管理
数据字典
数据页面
索引页面
日志
事务并发控制
两阶段锁
时间戳
乐观并发
多版本
主存
磁盘
存储管理
写回脏页
写日志
数据组织与管理
数据字典
文件组织
索引
空闲空间管理
页面组织
记录组织
日志管理
预写日志
redo/undo日志
故障恢复
重做
回滚
存储介质

存储概览


堆表
索引
事务并发控制
表空间管理
脏页队列
数据页面缓冲区
WAL日志缓存
WAL record
WAL record
WAL record
WAL record
提交日志缓冲区
Checkpoint线程
刷脏页线程
刷WAL日志线程
事务提交
异步写盘
数据文件
控制文件
WAL日志文件
事务提交日志
垃圾回收线程
WAL日志归档线程
WAL日志收发线程
WAL日志恢复

目录

存储介质


快
贵
小
非持久存储
CPU
≈1ns
L1缓存
≈3ns
L2缓存
≈14ns
L3缓存
≈70ns
DRAM主存
≈150ns
非易失性主存
≈50μs
闪存
≈20ms
磁盘
≈100ms
光盘/磁带
访问速度
单位价格
存储空间
持久存储
慢
便宜
大

存储介质介绍

- **高速缓冲存储器**
  - 存取速度最快的存储介质，而单位价格却比较昂贵
- **DRAM主存(内存)**
  - 主要用于存放程序和其处理的数据。一般为8GB～32GB（2022年）
- **磁盘**
  - 容量大、数据持久性好，存储长期数据。一般为512GB～16TB（2022年）
- **闪存**
  - 又叫固态硬盘，个人计算机闪存大小一般为128GB～1TB（2022年）
- **非易失性主存**
  - 速度能DRAM相媲美，字节级寻址，在断电时不会丢失数据

存储介质特性比较

存储介质	访问速度	存储空间	单位价格	数据持久性
高速缓存	快	小	高	易失
DRAM	较快	中等	中等	易失
非易失性主存	中等	中等	较高	非易失
闪存	较慢	较大	较低	非易失
磁盘	慢	大	低	非易失

磁盘

口接口

- SATA（Serial ATA） 500 MB/s～600MB/s
- 存储区域网络（Storage Area Network， SAN）
- 网络附属存储（Network Attached Storage， NAS）
- 非易失性主存主机控制器接口规范（Non-Volatile Memory Express， NVMe）
- 磁盘阵列（Redundant Arrays of Independent Disks， RAID） ~ 3500MB/s

磁盘结构

盘片

- 覆盖磁性物质，通过改变磁性物质的磁场方向来存储二进制的0和1，高速旋转每分钟5400转至7200转

磁头

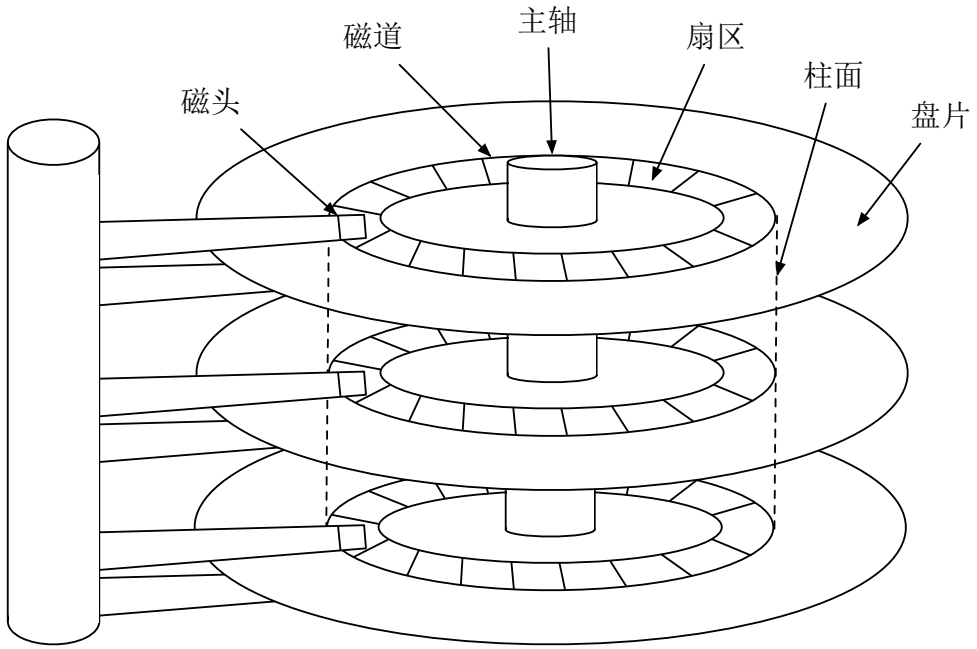
- 改变磁性物质状态并且读写数据

磁道

- 盘片上的圆环区域

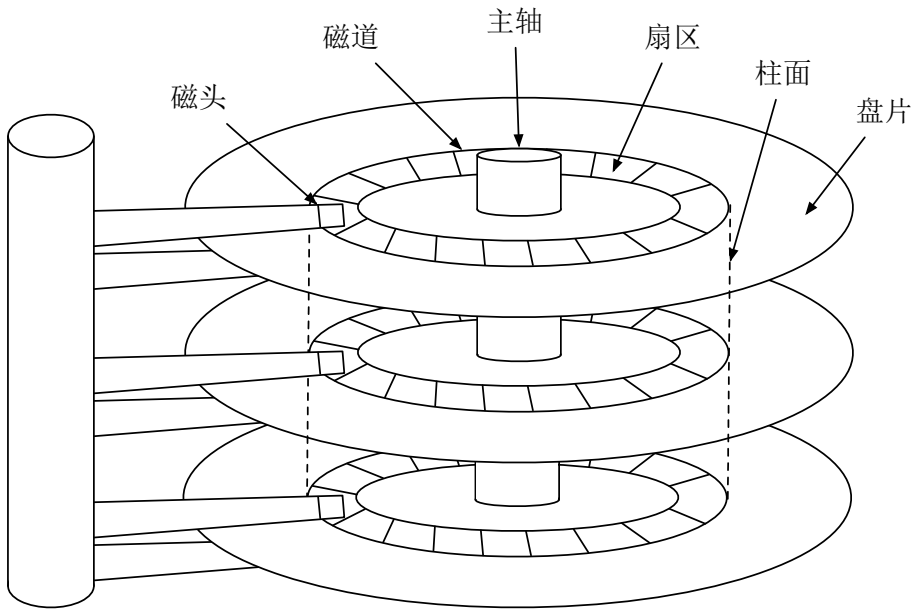
柱面

- 纵向对齐的多个磁道



磁盘结构

- **磁盘读写时延**
  - 寻道时延：磁头移动时延10ms
  - 旋转时延：盘片旋转时延
    - 5400转/分钟的磁盘平均时延是60*1000/5400*0.5=5.56毫秒
  - 数据读写时延：
    - 100MB/秒的磁盘
    - 读写4KB：4KB/100MB*1000=0.04毫秒
    - 读写1MB：1MB/100MB*1000=10毫秒
- **顺序读写：快（只有读写时延）**
- **随机读写：慢（寻道+旋转+读写）**



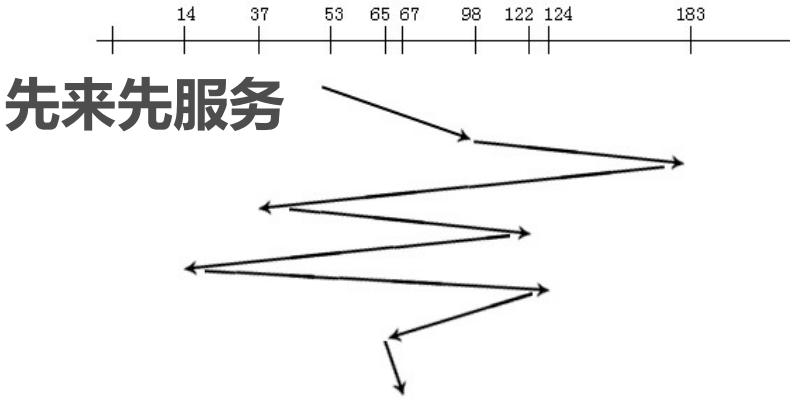
尽量顺序读写避免随机读写

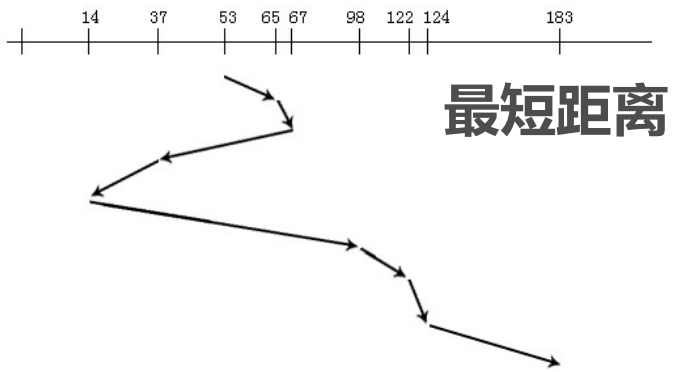
磁盘性能与优化

- **访问时间**
  - 系统发出读写指令后到磁盘开始返回数据的时间
- **数据传输率**
  - 每秒钟磁盘读写的数据量
- **磁盘访问优化技术**
  - 缓冲
  - 预读
  - 调度
  - 文件组织

磁盘调度技术：电梯算法

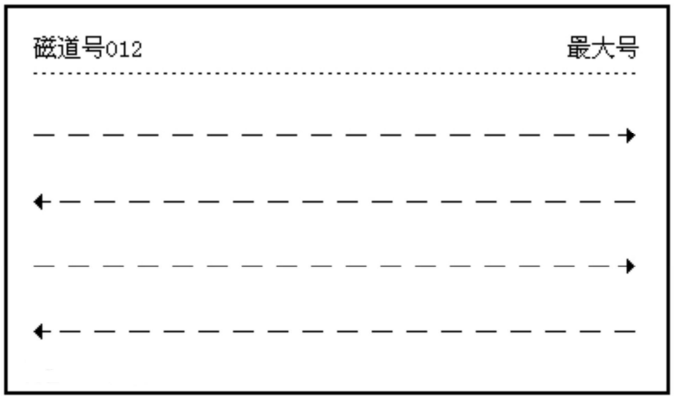
磁盘访问序列：98，183，37，122，14，124，65，67。读写头起始位置：53





口 电梯算法

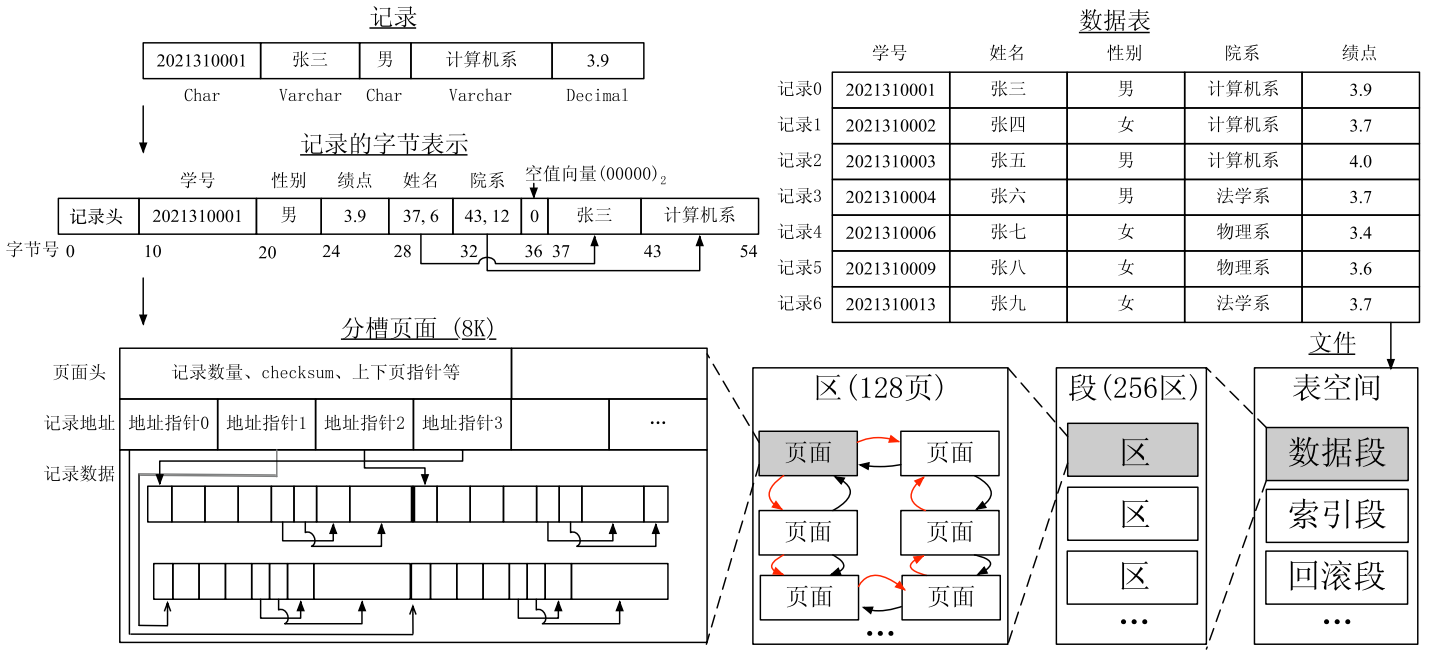
- 把磁头看作是在做横跨磁盘的扫描
- 从柱面最内圈到最外圈，然后再返回
- 正如电梯做垂直运动，从最底层再到顶层，然后再返回来。



目录

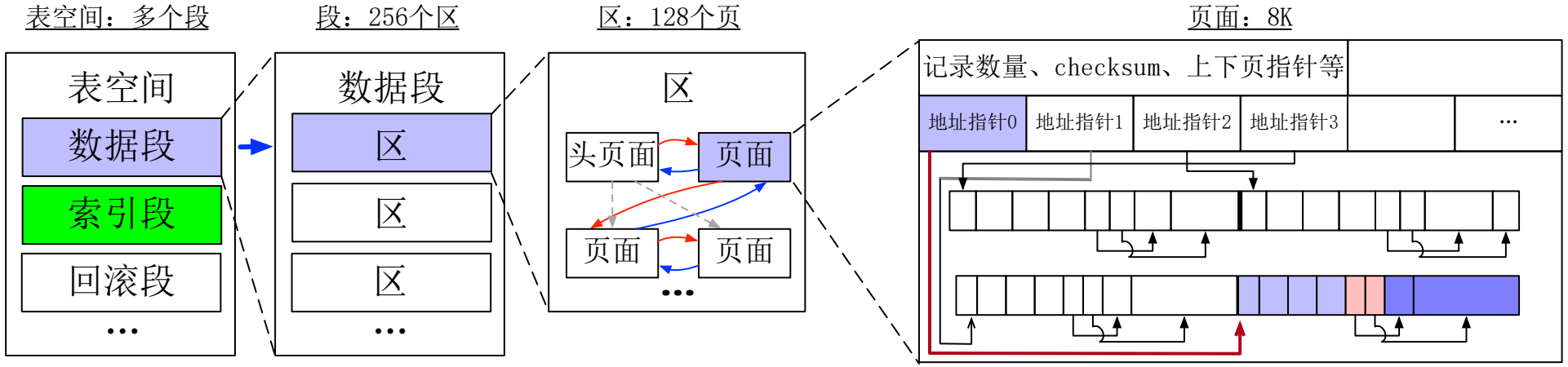
存储结构

- **数据块**
  - 存储介质上的数据最小存储单位
- **页面**
  - 统一大小来管理数据
  - 一般为8KB、16KB



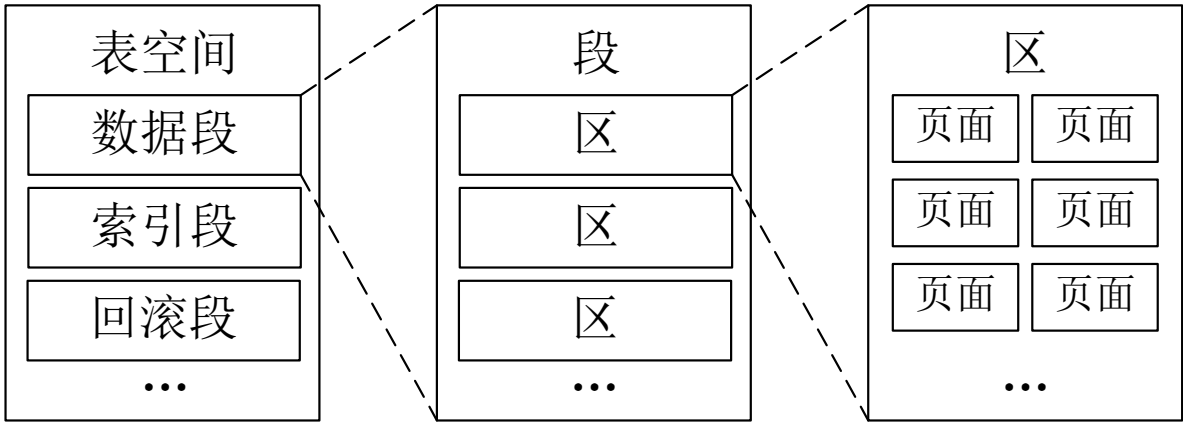
数据表空间管理

- **数据表空间由段（segment）、区（extent）、页（page）组成。**
  - 数据段、索引段、回滚段分别存储
  - 一次申请一个区（多个连续页），让相邻页的物理位置也相邻，方便实现顺序I/O。



申请新页面

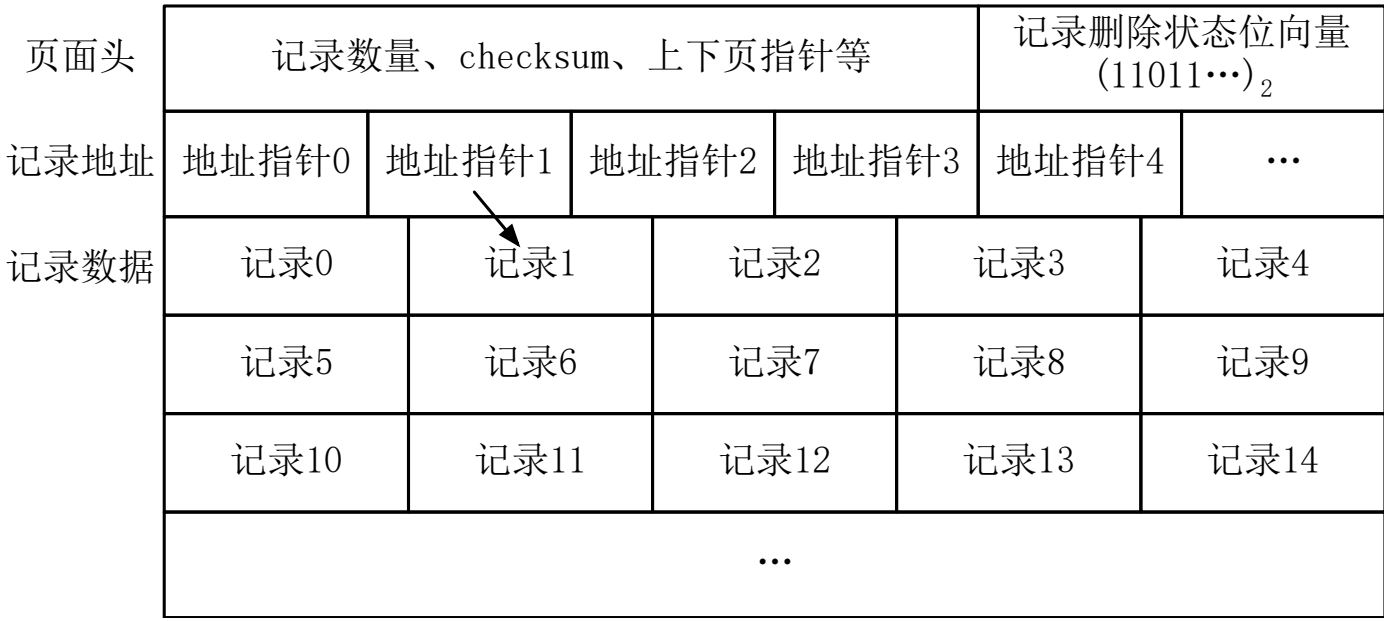
- 假设所有的页面都满时，要继续插入记录的话就需要申请新的页面
- 每次申请固定数量的区（extent）
  - 为了页面尽量连续



目录

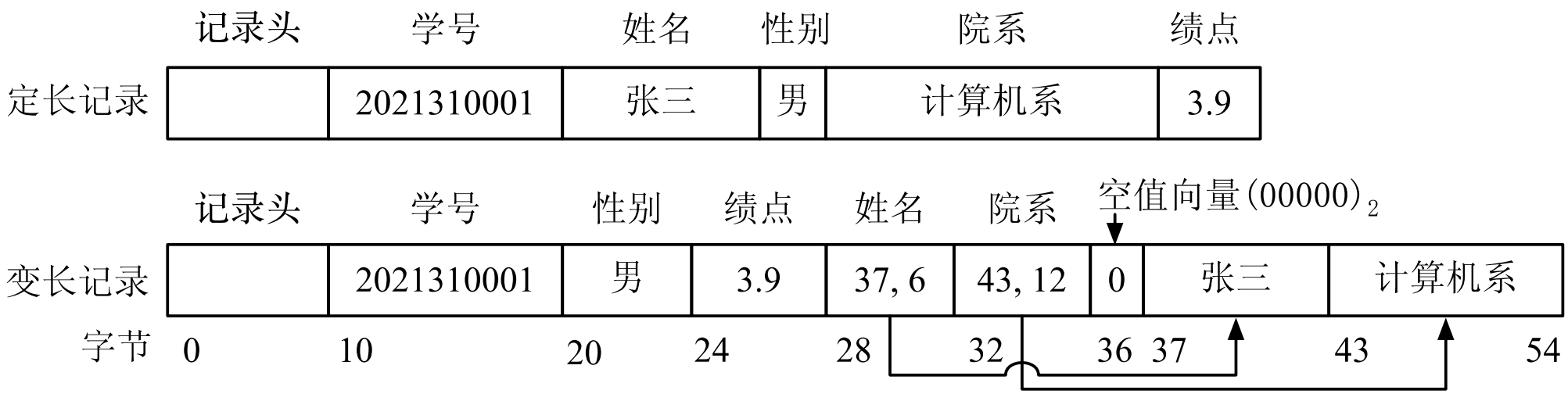
页面组织

- 页面大小为8KB或者16KB，一般为2的整数次幂
- 一个页面往往包含了多条记录



页面组织： 定长记录 vs 变长记录

- **定长记录：长度固定，页面中记录数目固定，易于定位一个页面的记录位置**
  - 例如年龄short，学号char（10）
- **变长记录：长度可变，页面中记录数目不固定，难以直接计算并定位记录位置**
  - 例如姓名varchar，院系varchar



定长记录

记录的长度（占用空间）固定

- 学号，char，10字节
- 姓名，char，20字节
- 姓别，char，4字节
- 院系，char，40字节
- 绩点，decimal，4字节

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

记录组织方式

- 记录头：事务信息、NULL值bitmap
- 每个属性值

	记录头	学号	姓名	性别	院系	绩点
定长记录		2021310001	张三	男	计算机系	3.9

页面头	记录数量、checksum、上下页指针等	记录删除状态位向量 $(11011\cdots)_2$

记录数据	记录0	记录1	记录2	记录3	记录4
	记录5	记录6	记录7	记录8	记录9
	记录10	记录11	记录12	记录13	记录14
	…				

记录添加

□如果页面可以容纳新纪录，添加到页面最后

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

记录删除

口删除记录，并依次将后续记录向前移动

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

记录删除

口将最后一条记录向前移动

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录6	2021310013	张九	女	法学系	3.7
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6

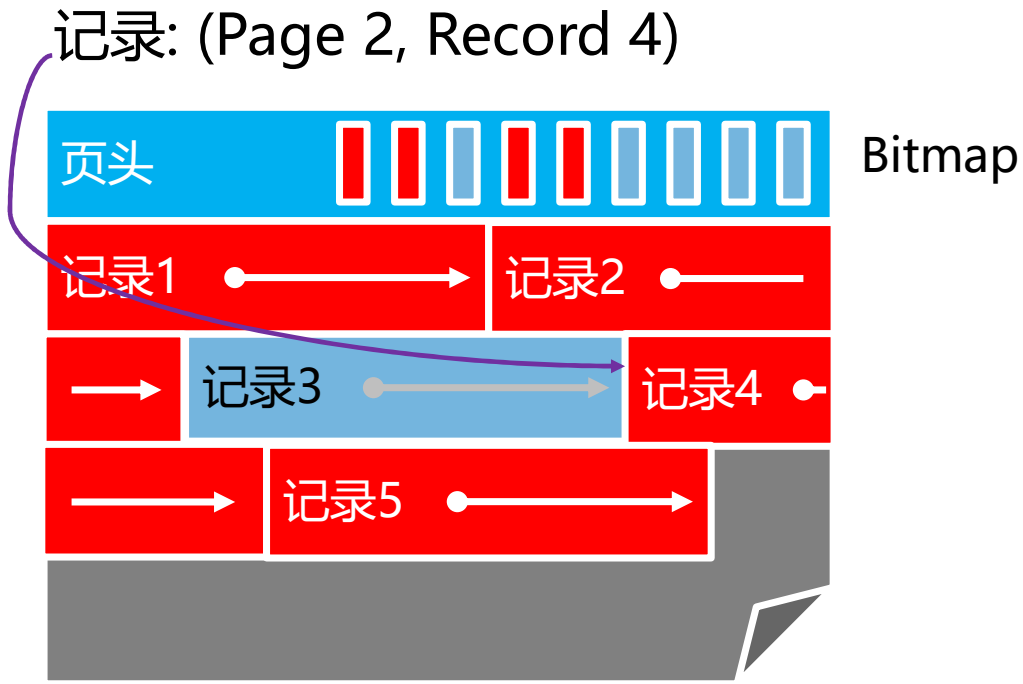
记录删除

- 可用空闲位置链表
- 插入记录时快读定位第一个空位

	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

定长记录（bitmap）

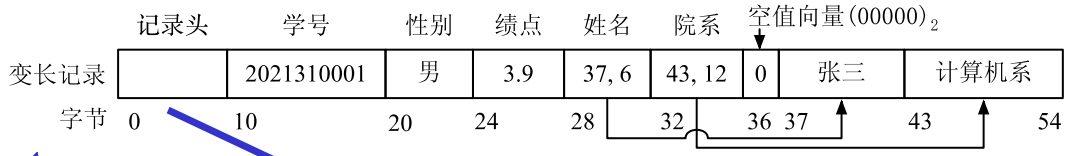
- Bitmap位图：标记记录是否存在
- **Insert插入：**找到第一个空槽
- **Delete删除：**清楚槽标志位
- Bitmap数目根据页面大小和记录长度计算



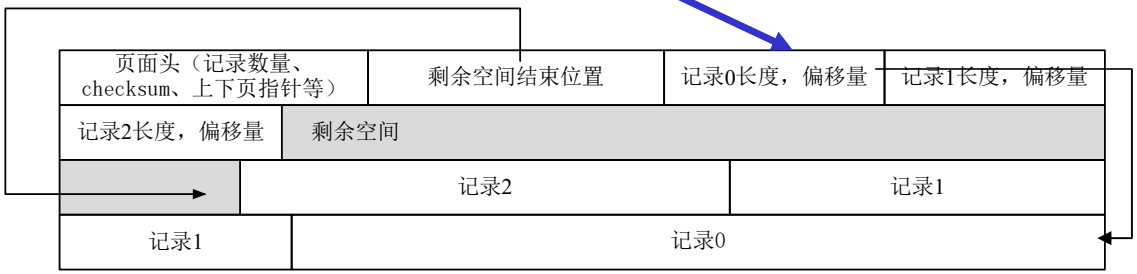
变长记录

□记录中的变长属性通过偏移量和长度定位

- 学号，char，10字节
- 姓名，varchar，变长
- 姓别，char，4字节
- 院系，varchar，变长
- 绩点，decimal，4字节



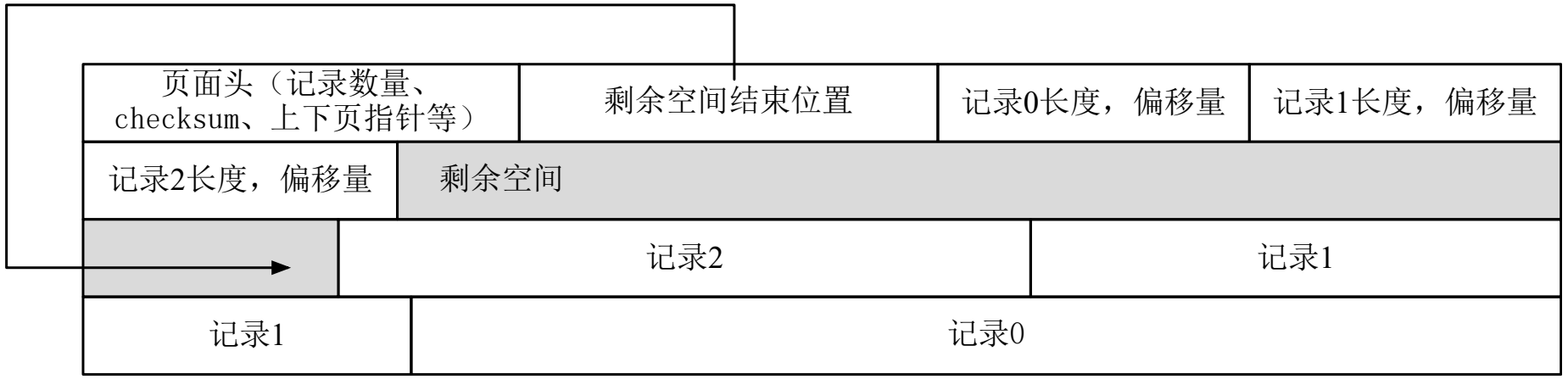
	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7



页面组织-变长记录

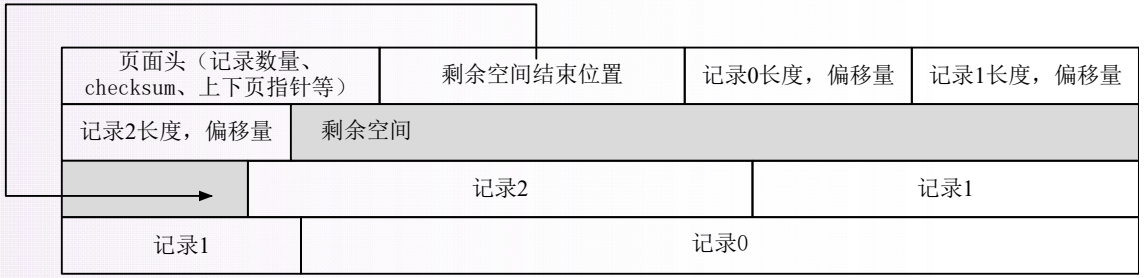
□页面内记录组织

- 页面头：页类型、状态
- 槽位：记录起始地址、长度（索引会指向槽位）
- 两头挤：页面、槽位等页面内从前往后；记录从后往前

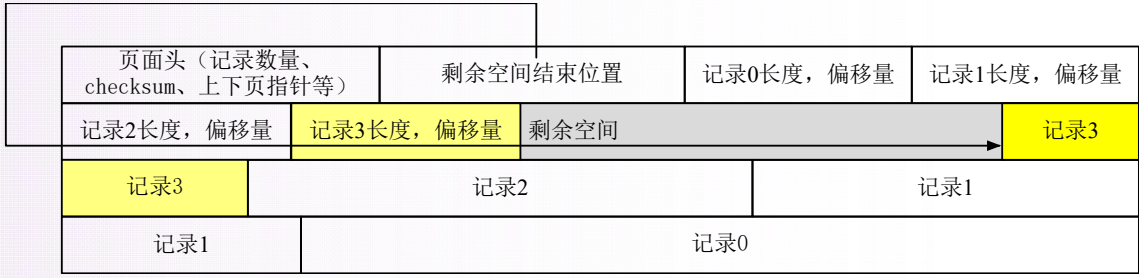


页面组织-变长记录

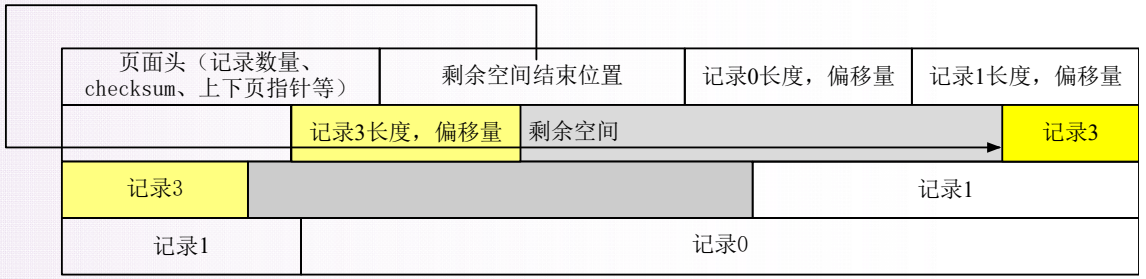
- 槽位→页面位置和长度
- 删除：删除槽位
- 插入：找到空闲槽位



- 插入记录3



- 删除记录2



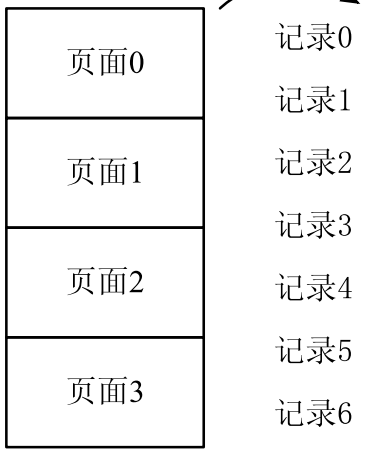
目录

文件组织

- 数据文件中页面之间的组织方式
- 目标：高效的数据页面访问（插入、删除、查找）
- 主要的文件组织方法
  - 堆表
  - 顺序表
  - 哈希表
  - B+树
  - 多表聚簇

堆表

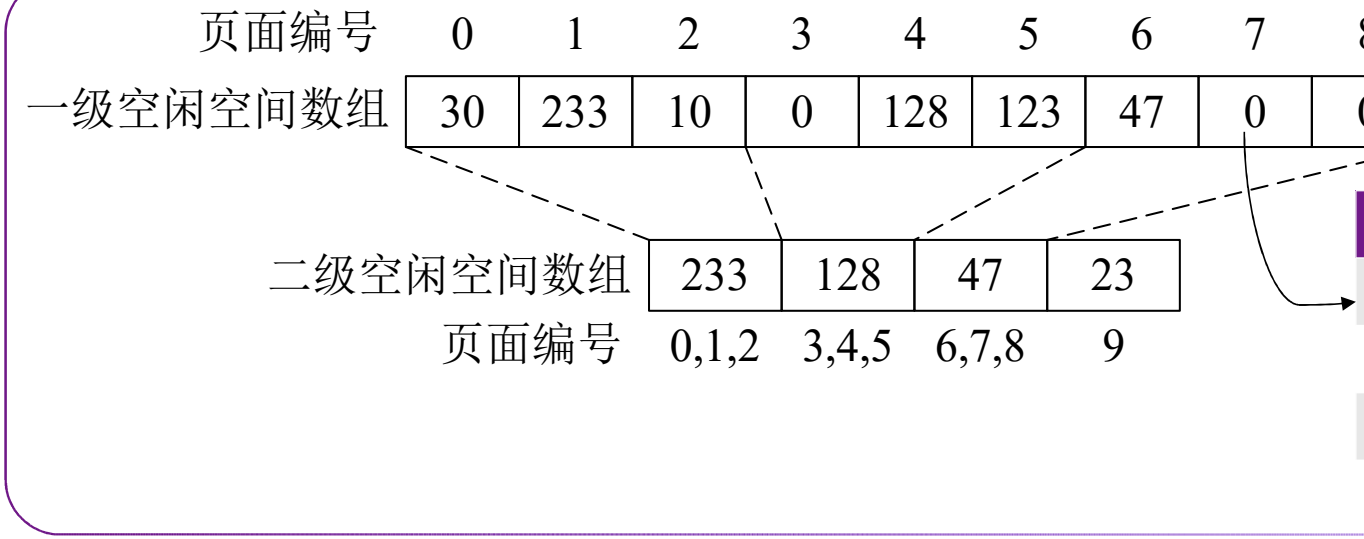
口记录的顺序没有限制，将记录简单排列在文件中



	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

堆表的多级空闲空间数组

- 快速找到空闲空间的位置
- 从高到低扫描各级空闲空间数组

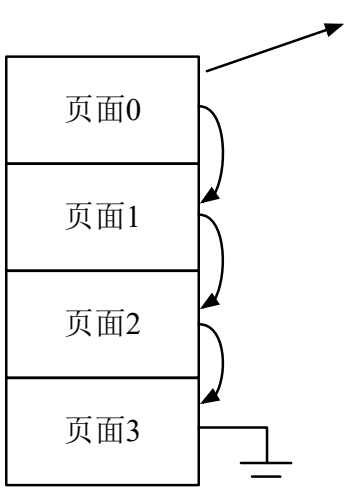


空闲空间编码	可用字节数
0	0~31
1	32~63
...	...
255	8164~8192

- 也可以用大根堆替代空闲空间数组，维护各页面空闲空间比例

顺序表

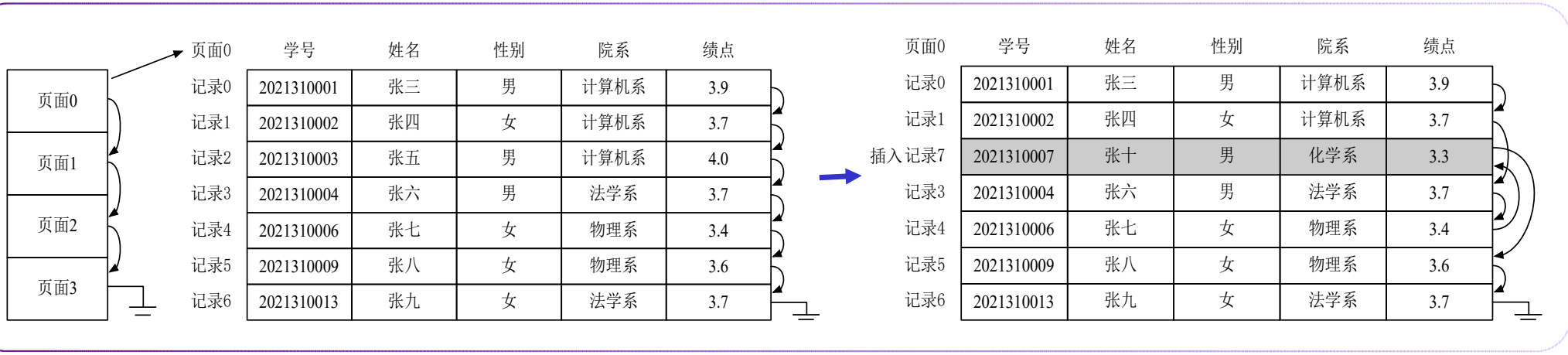
□ 记录按某个或某些字段的大小顺序存储



页面0	学号	姓名	性别	院系	绩点
记录0	2021310001	张三	男	计算机系	3.9
记录1	2021310002	张四	女	计算机系	3.7
记录2	2021310003	张五	男	计算机系	4.0
记录3	2021310004	张六	男	法学系	3.7
记录4	2021310006	张七	女	物理系	3.4
记录5	2021310009	张八	女	物理系	3.6
记录6	2021310013	张九	女	法学系	3.7

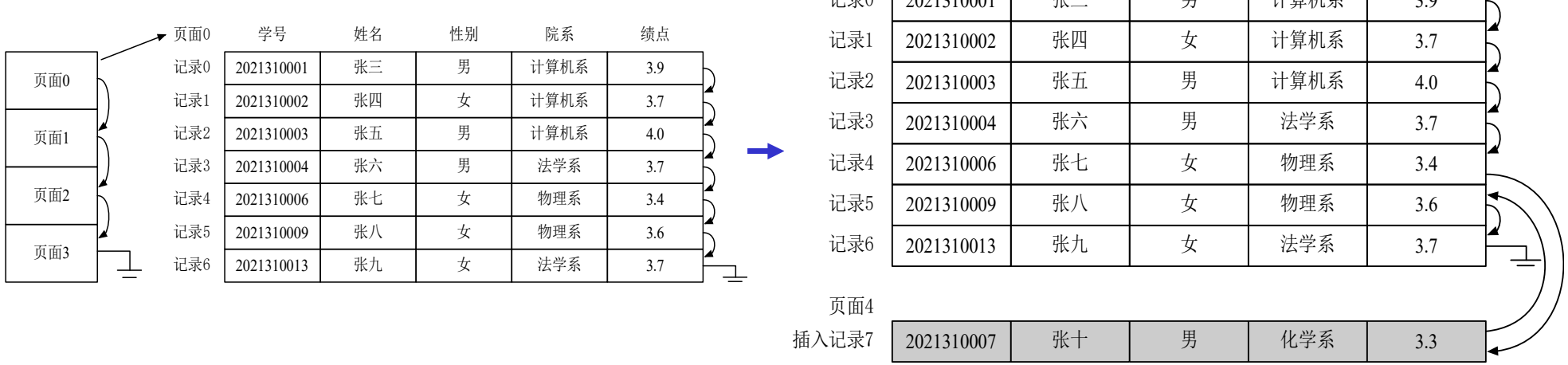
顺序表-插入

- 记录按某个或某些字段的大小顺序存储，使用链表连接文件中的记录
- 删除记录2后插入记录7结果如下



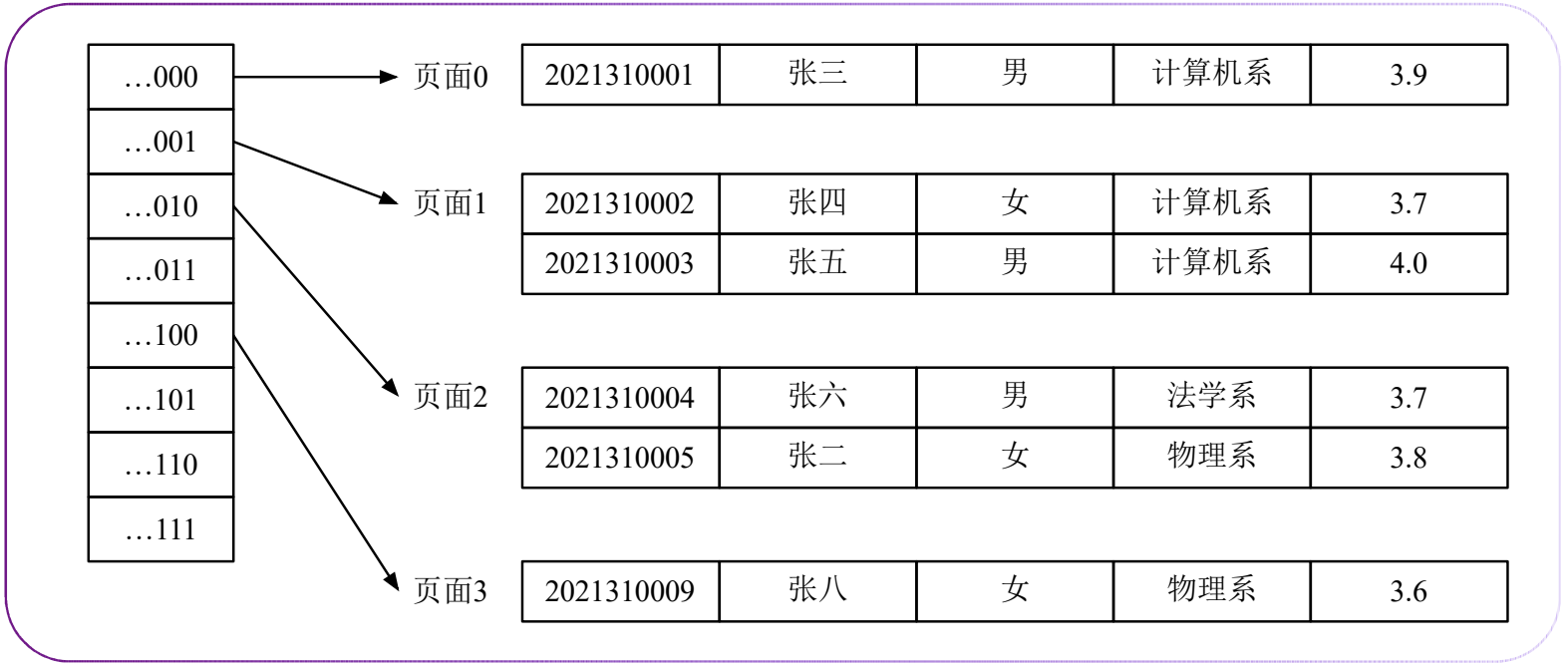
顺序表-插入到新页面

- 页面已满
- 直接插入记录7结果如下



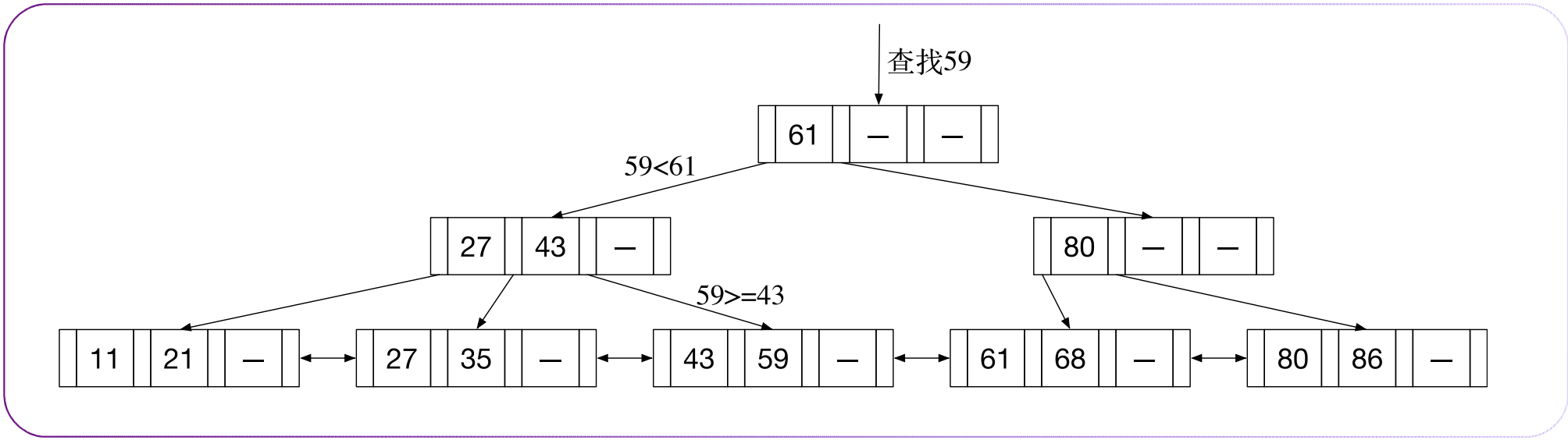
哈希表

口 使用哈希表将记录存储到不同的页面或者不同的页面集合



B+树

口平衡多叉树



文件组织方法性能对比

文件组织方法	复杂度	增	删	改	点查询	范围查询
堆表	简单	$O(1)$	$O(n)$	$O(n)$	$O(n)$	$O(n)$
堆表 + B+树	复杂	$O(log_m\, n)$	$O(log_m\, n)$	$O(log_m\, n)$	$O(log_m\, n)$	$O(log_m\, n + k)$
顺序表	中等	$O(log_2\, n)$	$O(log_2\, n)$	$O(log_2\, n)$	$O(log_2\, n)$	$O(log_2\, n + b)$
顺序表+ B+树	复杂	$O(log_m\, n)$	$O(log_m\, n)$	$O(log_m\, n)$	$O(log_m\, n)$	$O(log_m\, n + b)$
哈希表	中等	$O(l)$	$O(l)$	$O(l)$	$O(l)$	$O(n)$

n 页面数目　　m B树阶数　　k 结果记录数目　　b 结果页面数目　　l 哈希链长度

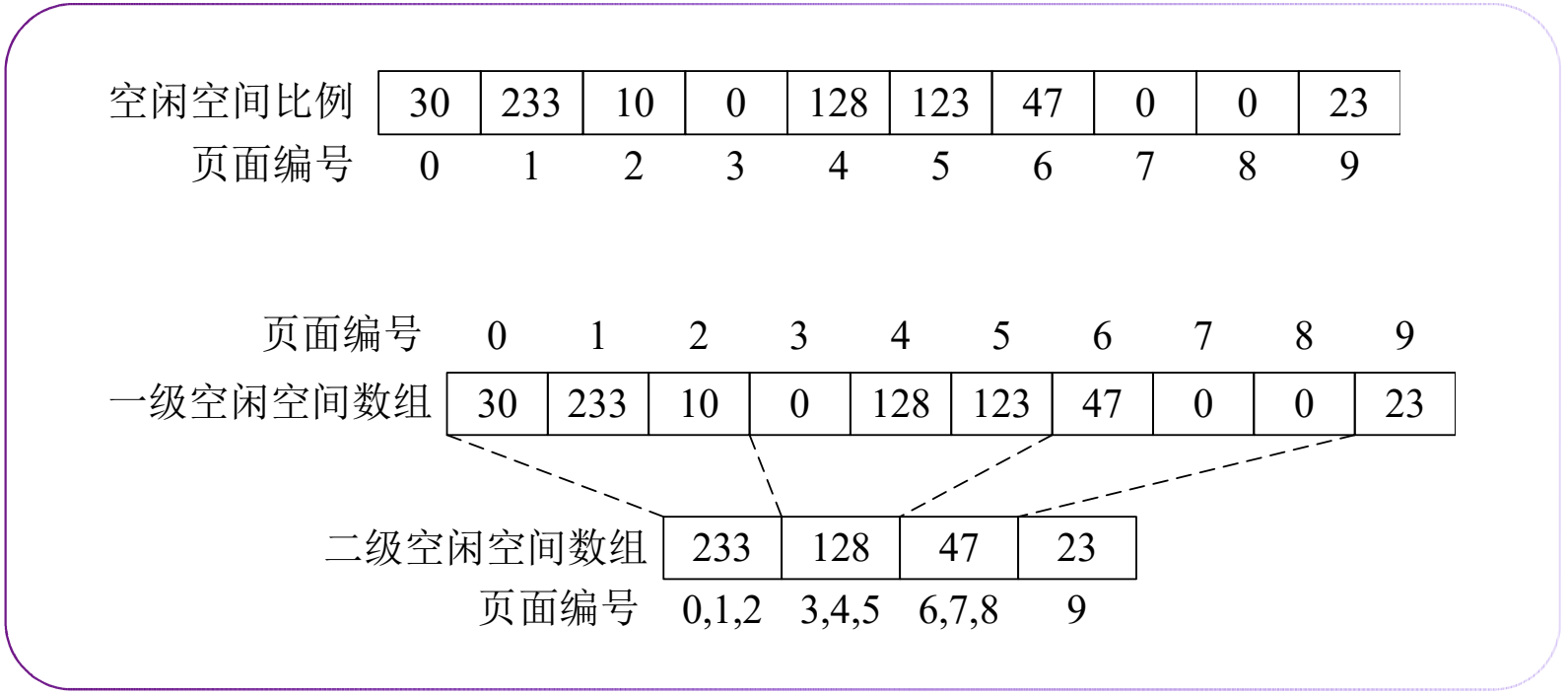
多表聚簇文件组织

口多个相关的表中的数据存储在同一个文件/页面中

院系记录0	计算机系	东主楼			
学生记录0	2021310001	张三	男	计算机系	3.9
学生记录1	2021310002	张四	女	计算机系	3.7
学生记录2	2021310003	张五	男	计算机系	4.0
院系记录1	物理系	西主楼			
学生记录3	2021310006	张七	女	物理系	3.4
学生记录4	2021310009	张八	女	物理系	3.6

空闲空间管理

- 插入一条长为x的记录，应该插入到哪个页面?
- 堆表需要找到合适的页面（空闲空间大于x的页面）
  - 1 记录每个页面空间空间大小

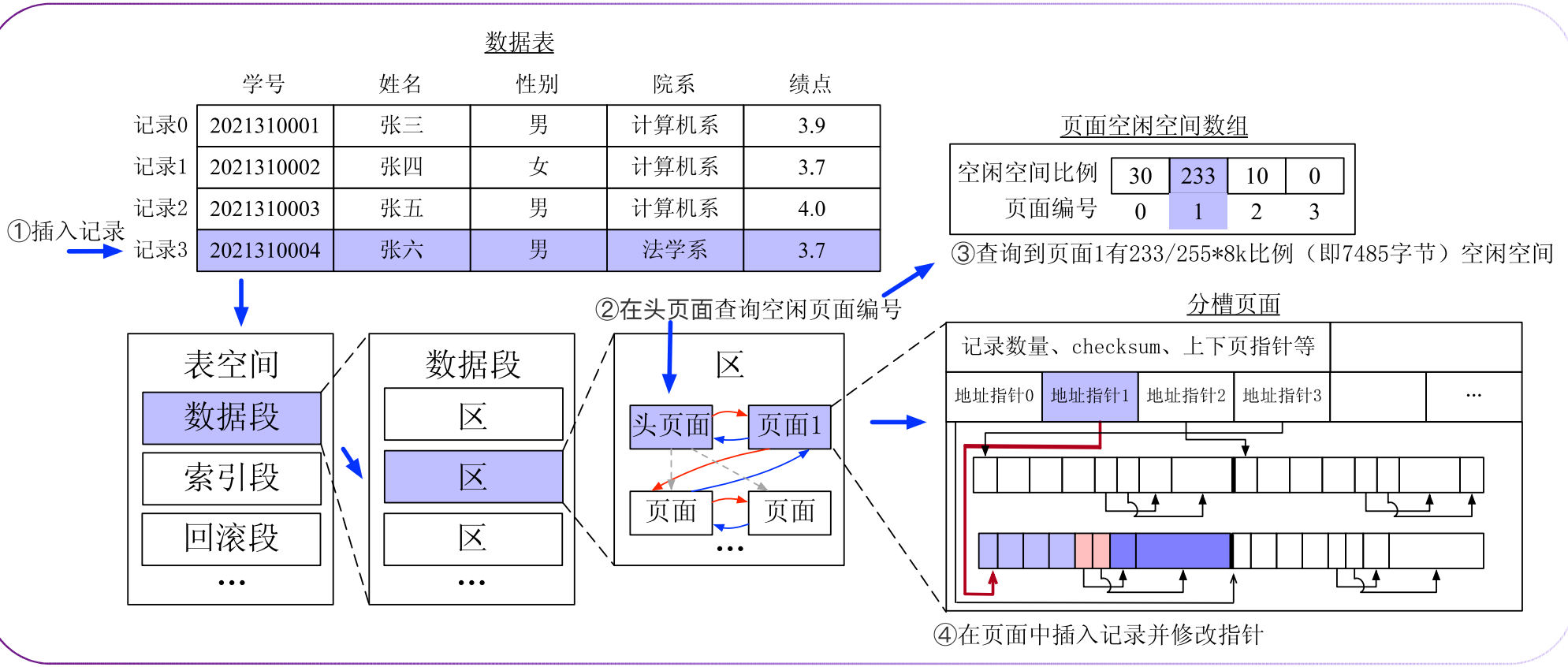


空闲空间管理

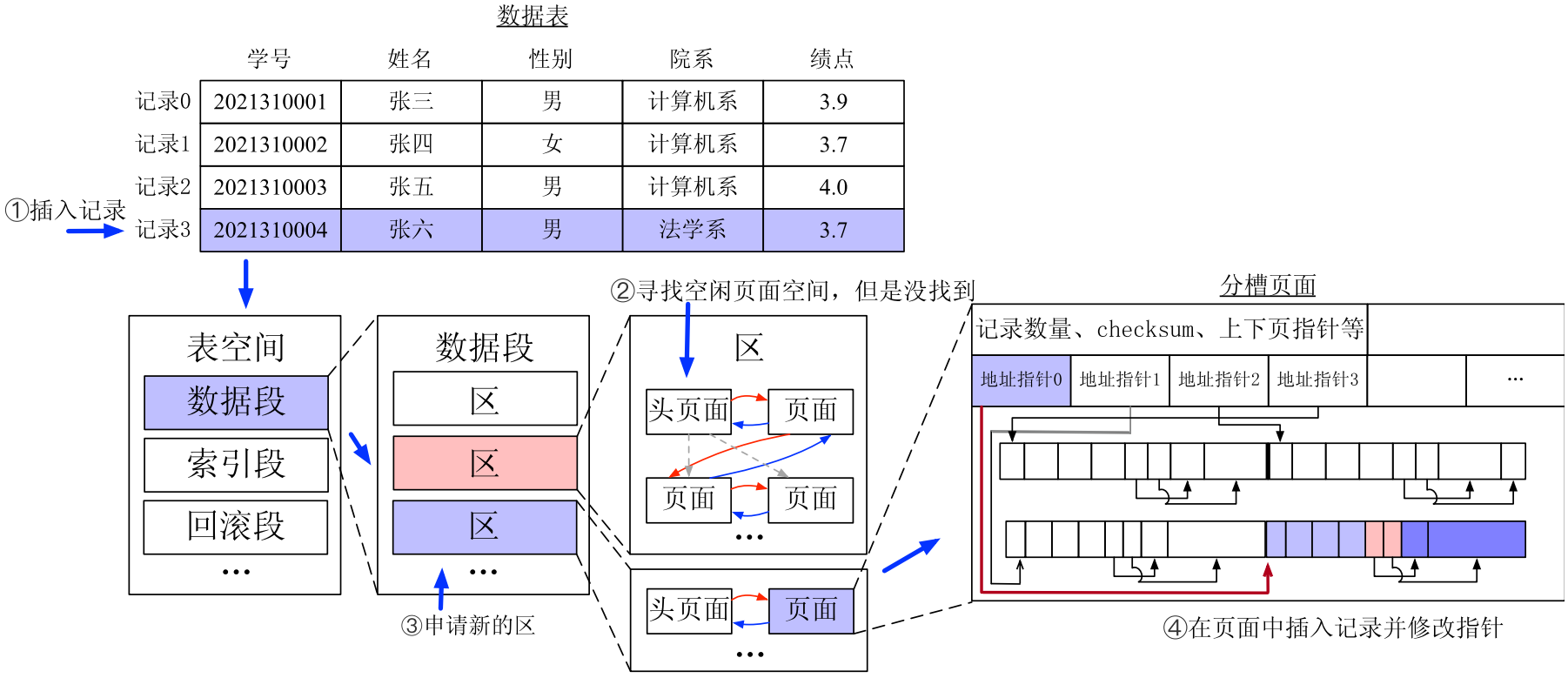
- 插入一条长为x的记录，应该插入到哪个页面?
- 堆表需要找到合适的页面（空闲空间大于x的页面）
  - 1 记录每个页面空间空间大小
  - 2 利用倒排列表

空闲空间编码	可用字节数	Page IDs
0	0~31	Page2, Page12
1	32~63	Page8, Page15
…	…	Page7, Page14
255	8164~8192	Page9, Page13

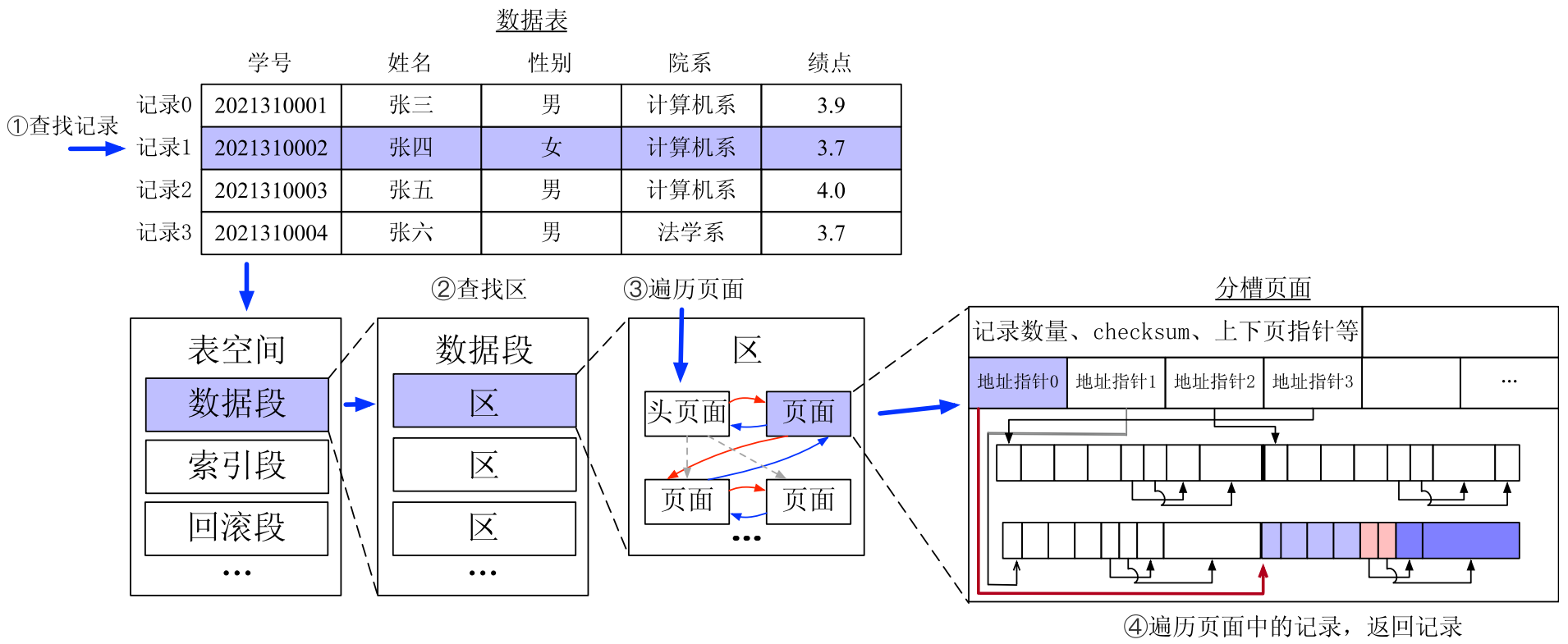
插入记录 – 有空闲页面


数据表
学号
姓名
性别
院系
绩点
记录0
2021310001
张三
男
计算机系
3.9
记录1
2021310002
张四
女
计算机系
3.7
记录2
2021310003
张五
男
计算机系
4.0
记录3
2021310004
张六
男
法学系
3.7
①插入记录
表空间
数据段
索引段
回滚段
…
数据段
区
区
区
…
区
头页面
页面1
页面
页面
…
②在头页面查询空闲页面编号
页面空闲空间数组
空闲空间比例
30
233
10
0
页面编号
0
1
2
3
③查询到页面1有233/255*8k比例（即7485字节）空闲空间
分槽页面
记录数量、checksum、上下页指针等
地址指针0
地址指针1
地址指针2
地址指针3
…
④在页面中插入记录并修改指针

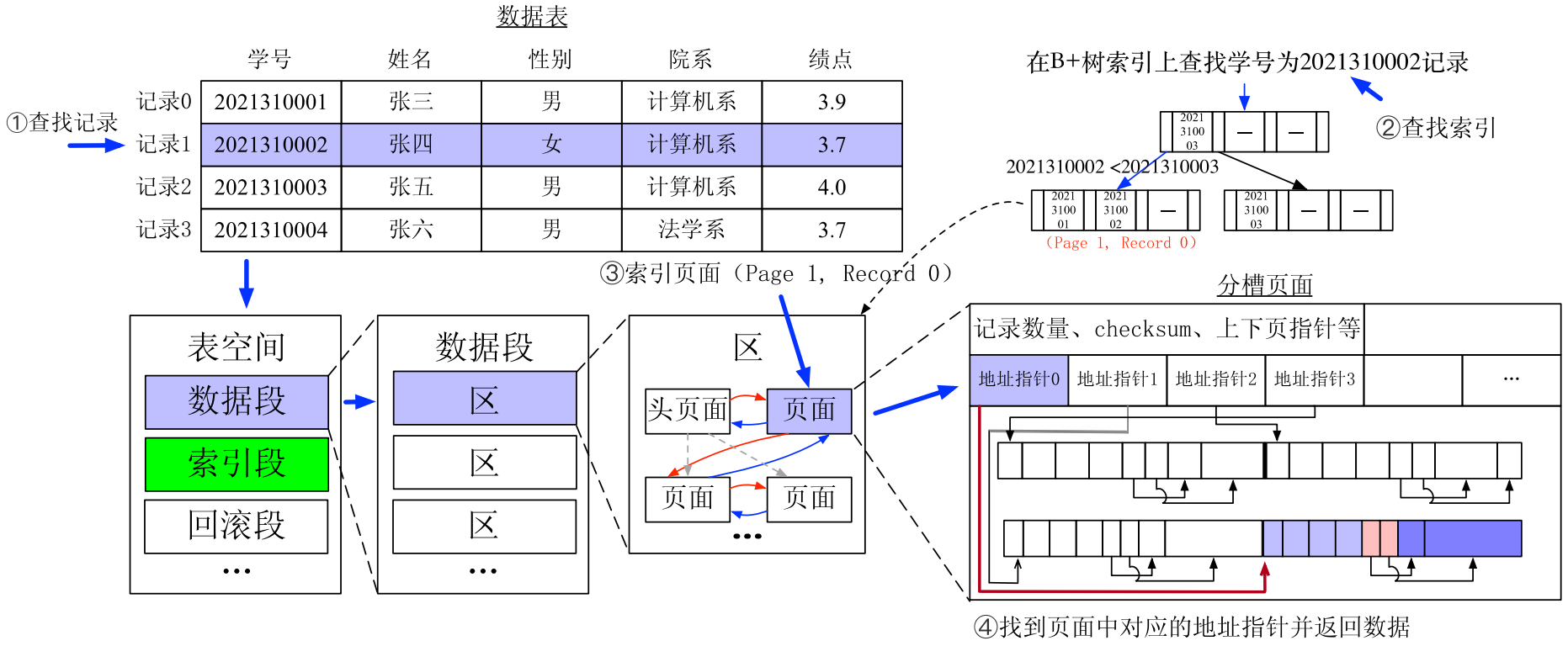
插入记录 – 无空闲页面


数据表
学号
姓名
性别
院系
绩点
记录0
2021310001
张三
男
计算机系
3.9
记录1
2021310002
张四
女
计算机系
3.7
记录2
2021310003
张五
男
计算机系
4.0
①插入记录
记录3
2021310004
张六
男
法学系
3.7
表空间
数据段
索引段
回滚段
…
数据段
区
区
区
…
③申请新的区
②寻找空闲页面空间，但是没找到
区
头页面
页面
页面
页面
…
头页面
页面
…
分槽页面
记录数量、checksum、上下页指针等
地址指针0
地址指针1
地址指针2
地址指针3
…
④在页面中插入记录并修改指针

查询记录 – 无索引


数据表
学号
姓名
性别
院系
绩点
①查找记录
记录0
2021310001
张三
男
计算机系
3.9
记录1
2021310002
张四
女
计算机系
3.7
记录2
2021310003
张五
男
计算机系
4.0
记录3
2021310004
张六
男
法学系
3.7
②查找区
③遍历页面
分槽页面
表空间
数据段
索引段
回滚段
数据段
区
区
区
区
头页面
页面
页面
页面
记录数量、checksum、上下页指针等
地址指针0
地址指针1
地址指针2
地址指针3
④遍历页面中的记录，返回记录

查询记录 – 有索引


数据表
学号
姓名
性别
院系
绩点
记录0
2021310001
张三
男
计算机系
3.9
记录1
2021310002
张四
女
计算机系
3.7
记录2
2021310003
张五
男
计算机系
4.0
记录3
2021310004
张六
男
法学系
3.7
①查找记录
在B+树索引上查找学号为2021310002记录
②查找索引
2021310002 <2021310003
(Page 1, Record 0)
③索引页面（Page 1, Record 0）
表空间
数据段
索引段
回滚段
数据段
区
区
区
区
头页面
页面
页面
页面
分槽页面
记录数量、checksum、上下页指针等
地址指针0
地址指针1
地址指针2
地址指针3
④找到页面中对应的地址指针并返回数据

删除和更新记录

- **删除一条记录：首先查找到该条记录，然后删除**
  - 定长：bitmap置为0
  - 变长：槽位清零
- **更新一条记录：首先查找到该条记录，然后更新**
  - 定长：直接可以原位更新
  - 变长：
    - 如果新纪录不长于老记录，原位更新，更新槽位
    - 如果新纪录长于老记录，槽位清零，插入新纪录
      - 本页面可以容纳，则插入本页面
      - 本页面不可以容纳，则插入其他页面

目录

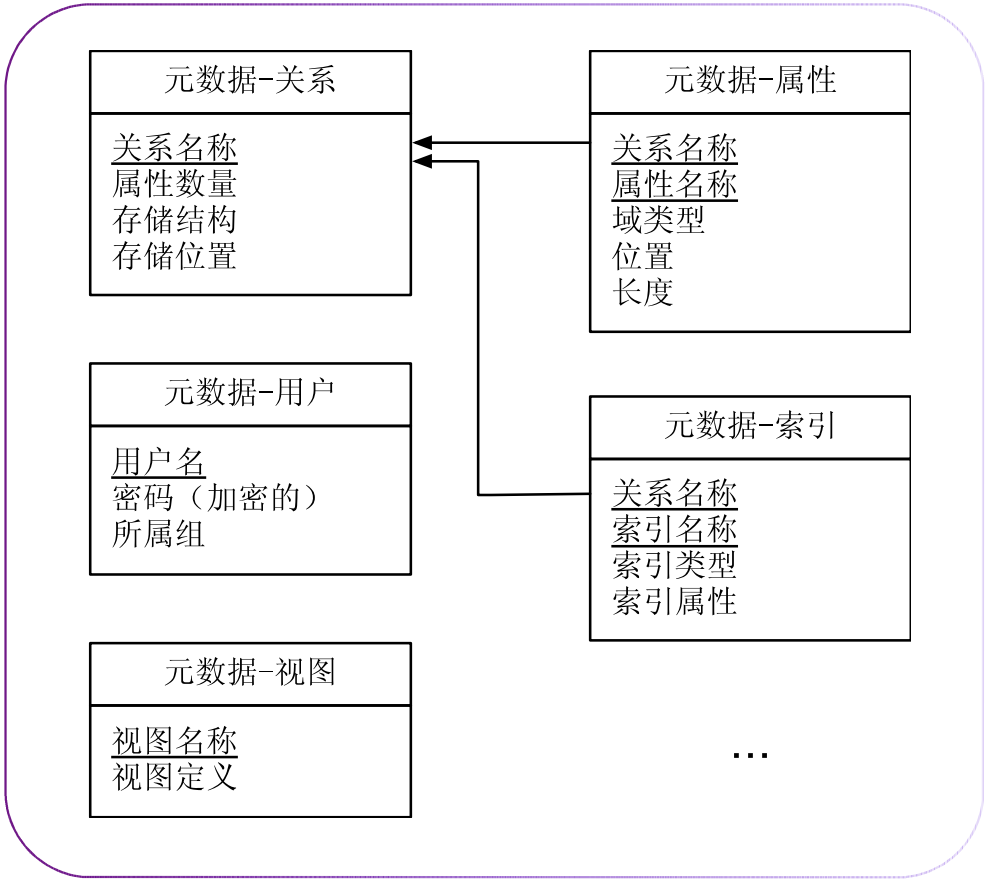
元数据

口关系表本身的属性

- 表名，即关系表名称
- 列名，即关系表的属性名
- 属性的域和长度
- 视图的名称和定义
- 完整性约束，例如外键约束
- 关系表的属性的统计信息，可用于辅助查询优化
- 索引信息

元数据的关系模式

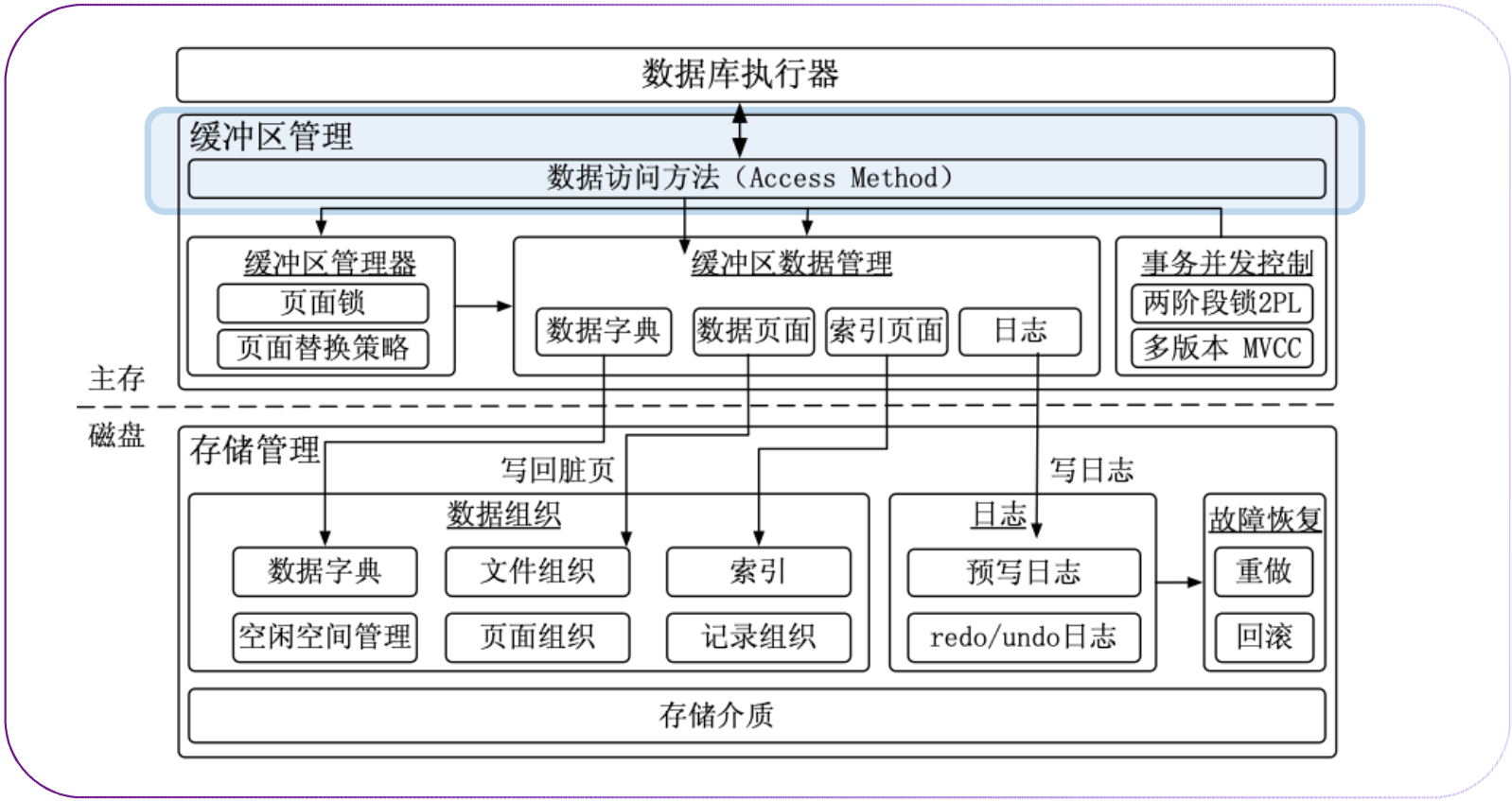
口像存储关系表一样来存储元数据



目录

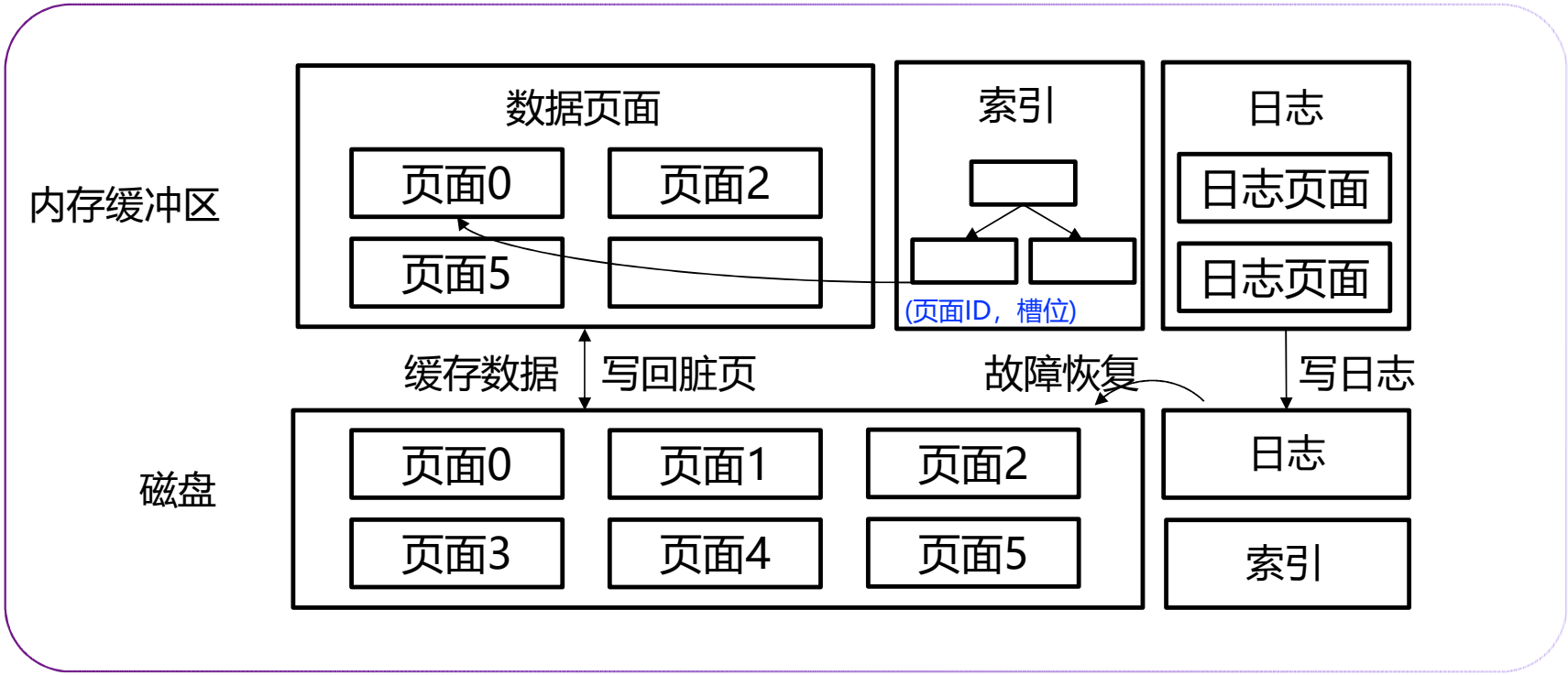
缓冲区

口将存储于磁盘的数据同时存储于主存，减少磁盘访问



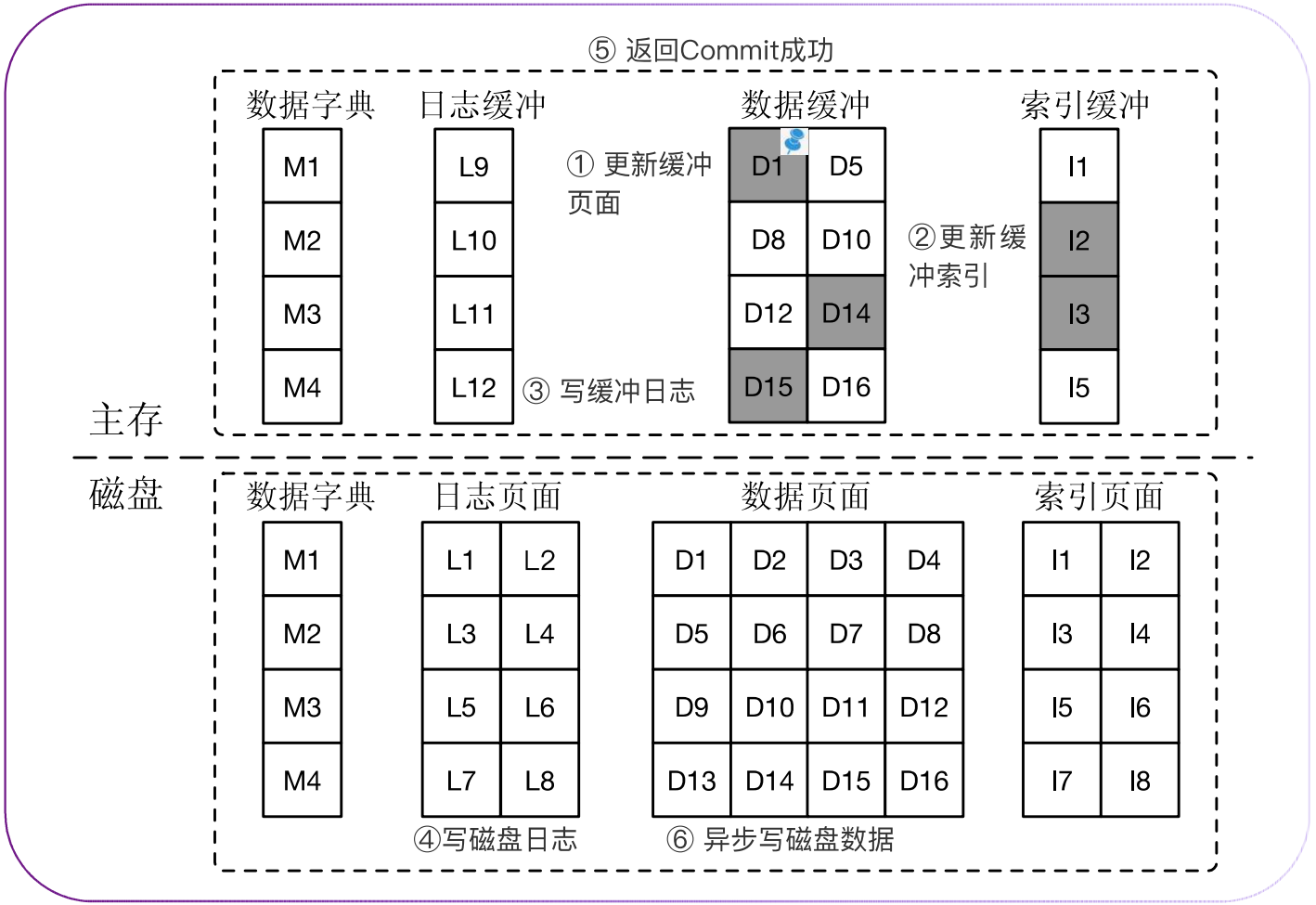
缓冲区组成

- 缓冲区数据主要包含缓存的磁盘数据页面、索引、日志等。日志可用于故障恢复。



缓冲区管理

1. 更新缓冲区页面
2. 更新缓冲区索引
3. 更新缓冲区日志
4. 更新磁盘日志
5. 返回提交成功
6. **异步刷新数据和索引页面**



缓冲区管理器

口为数据访问提供读取、修改和写入接口

口页面替换过程：

当程序请求访问磁盘页面时

1.如果该页面已经在缓冲区中，则返回缓冲区中的页面数据

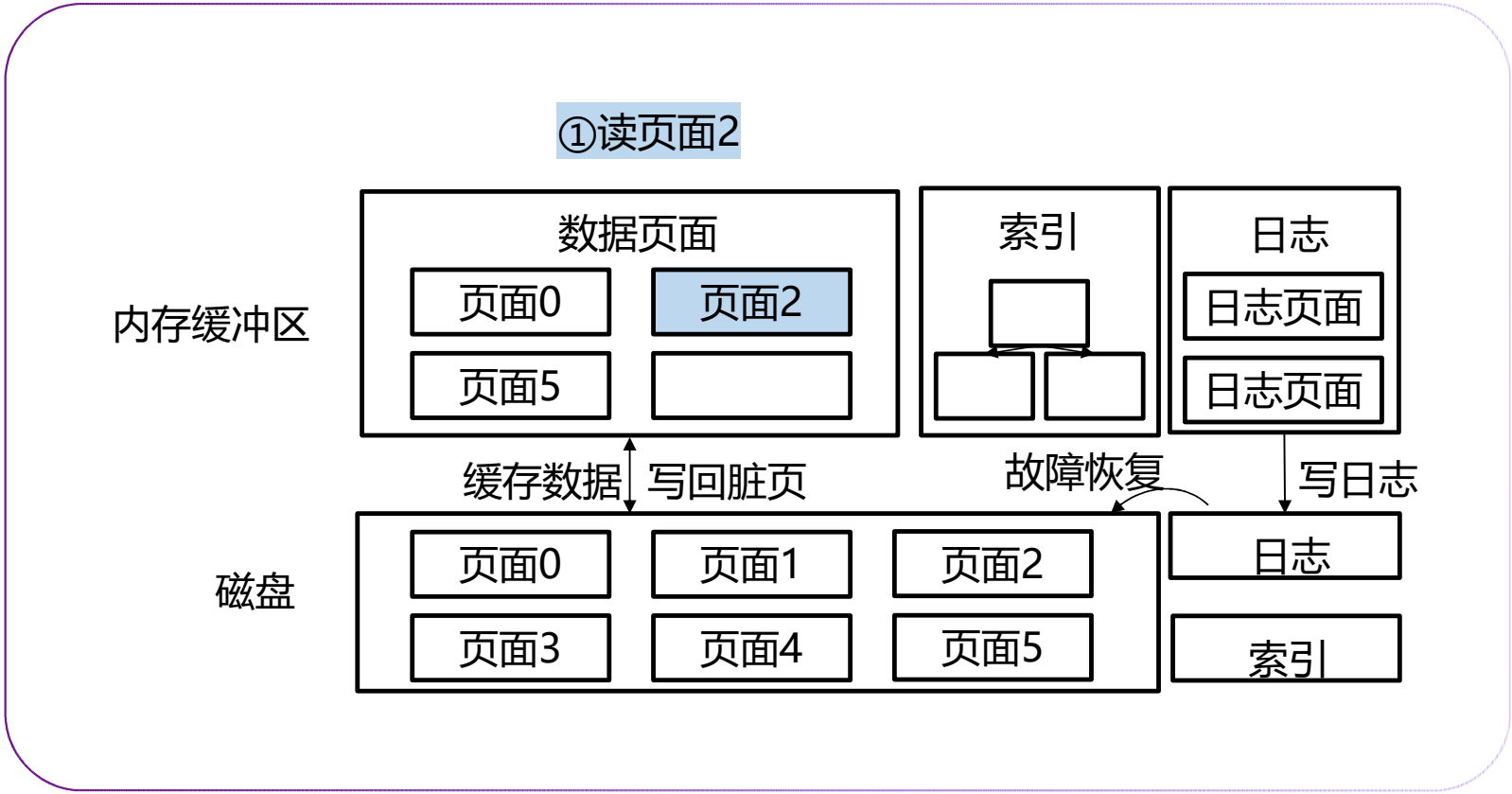
2.如果该页面不在缓冲区中，则从磁盘上读取该页面

(1)如果缓冲区未满，则将数据写入缓冲区

(2)如果缓冲区已满，则根据缓冲区页面替换策略挑选其他页面剔除或写回到磁盘上，再读入新页面

缓冲区——读取页面

1.如果该页面已经在缓冲区中，则返回缓冲区中的页面数据

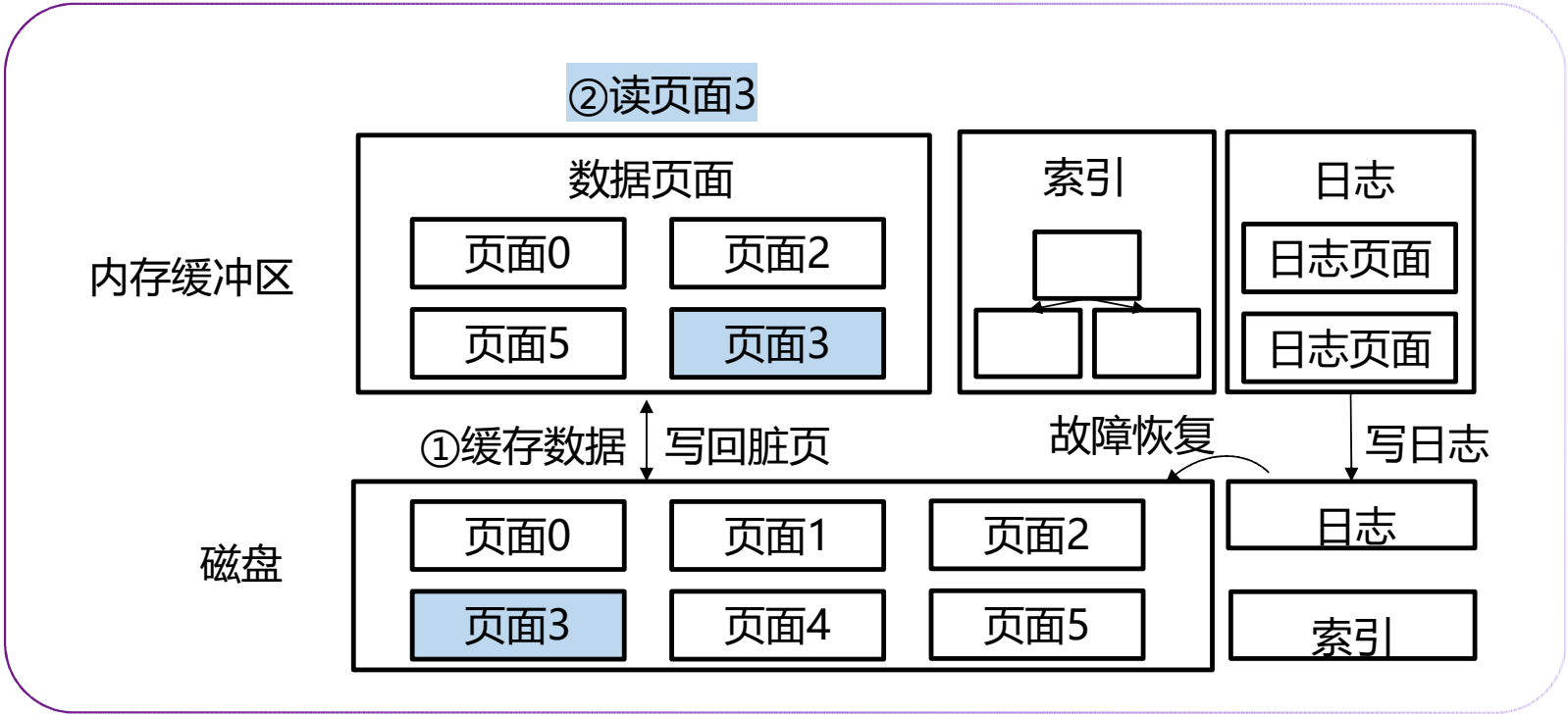


缓冲区——读取页面

2.如果该页面不在缓冲区中，则从磁盘上读取该页面

➡ (1)如果缓冲区未满，则将数据写入缓冲区

(2)如果缓冲区已满，则根据缓冲区页面替换策略挑选其他页面剔除或到磁盘上，再读入新页面

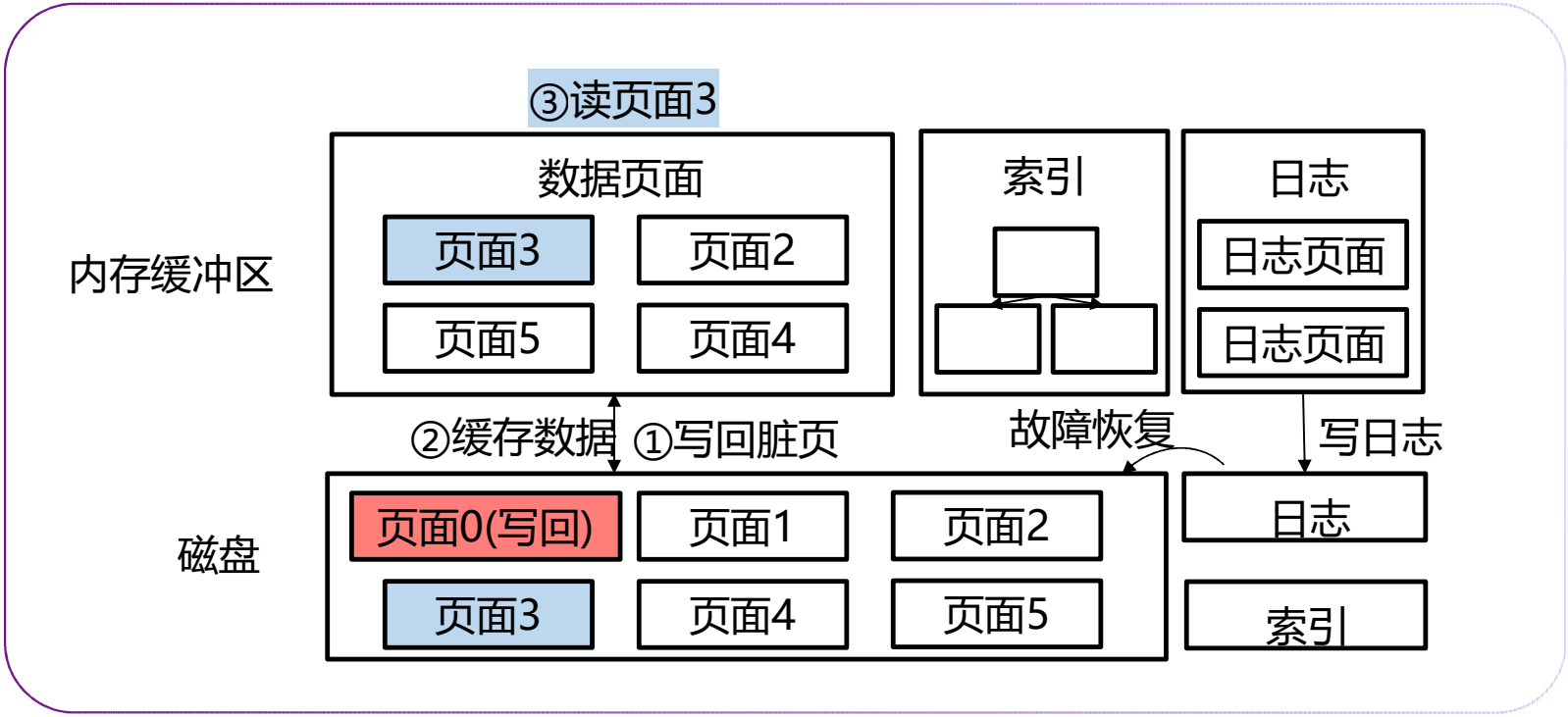


缓冲区——读取页面

2.如果该页面不在缓冲区中，则从磁盘上读取该页面

(1)如果缓冲区未满，则将数据写入缓冲区

➡ (2)如果缓冲区已满，则根据缓冲区页面替换策略挑选其他页面剔除或到磁盘上，再读入新页面

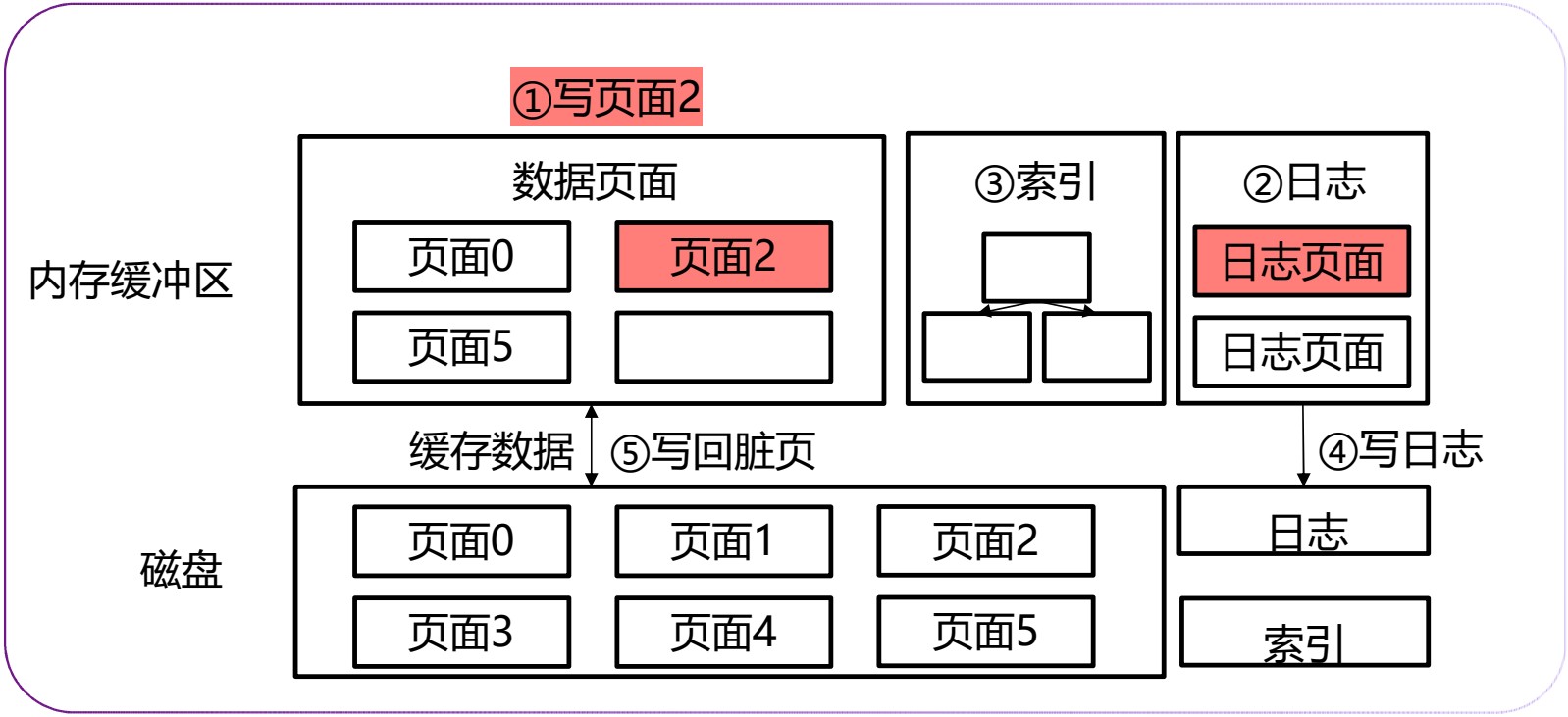


缓冲区——写入页面

2.写页面时

(1)如果该页面在缓冲区中，则写入缓存、日志和索引

(2)如果该页面不在缓冲区中，则从磁盘上读取该页面，再写入页面到缓存、日志和索引

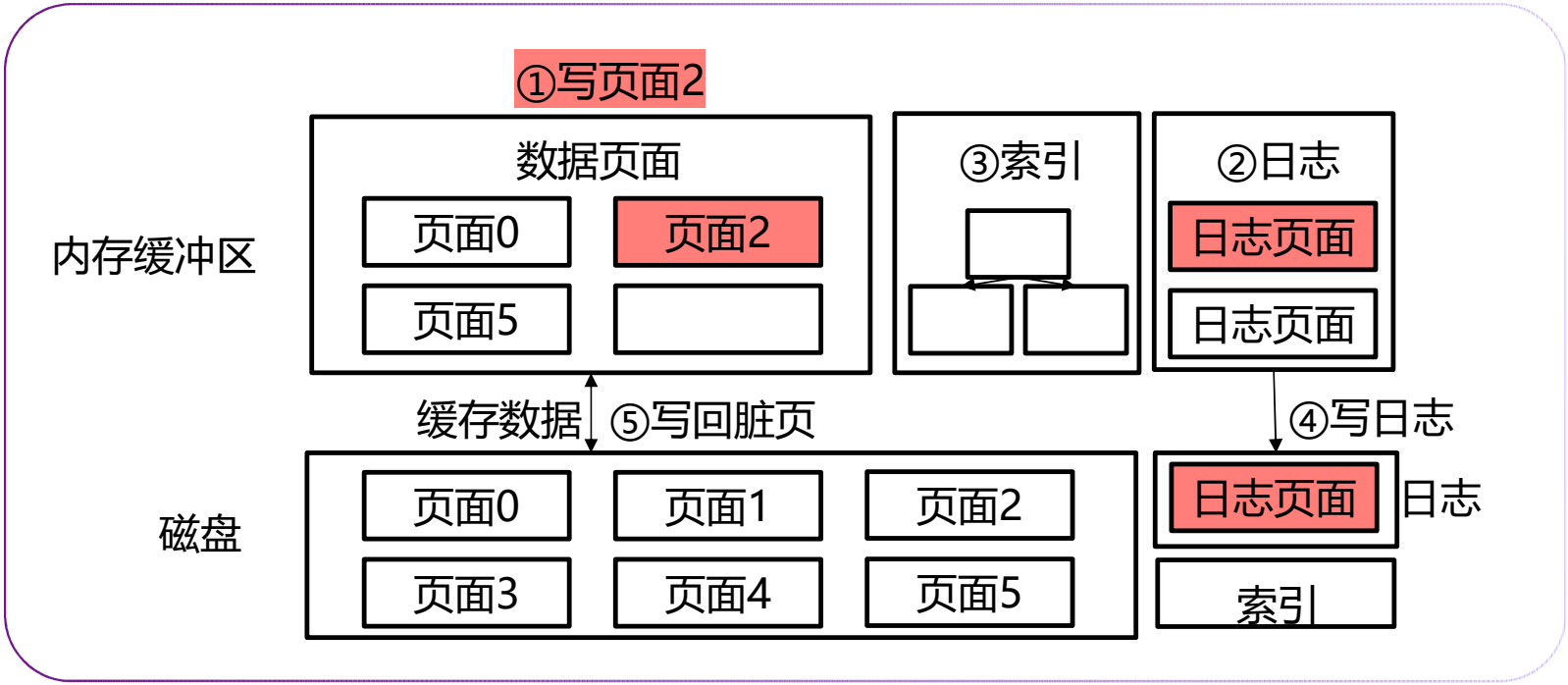


缓冲区——写入页面

2.写页面时

(1)如果该页面在缓冲区中，则写入缓存、日志和索引

(2)如果该页面不在缓冲区中，则从磁盘上读取该页面，再写入页面到缓存、日志和索引

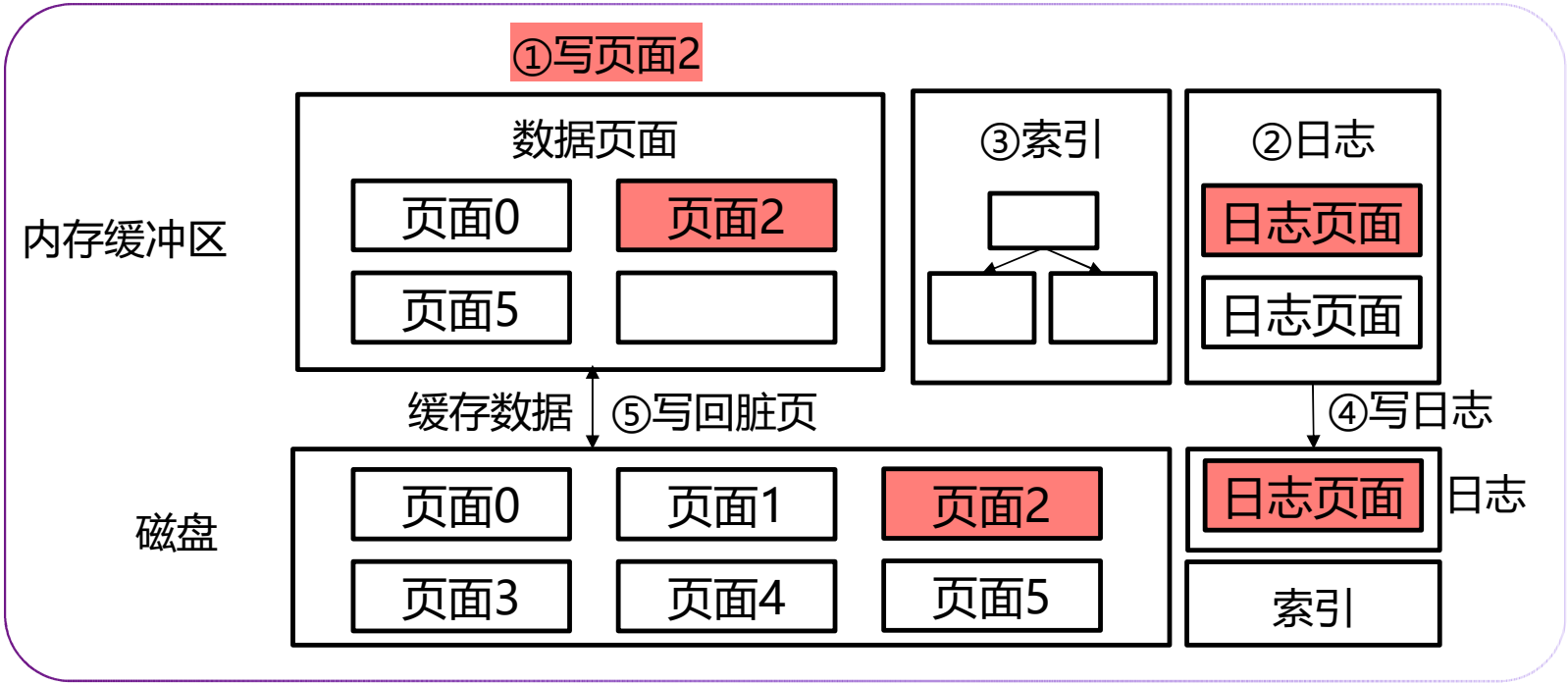


缓冲区——写入页面

2.写页面时

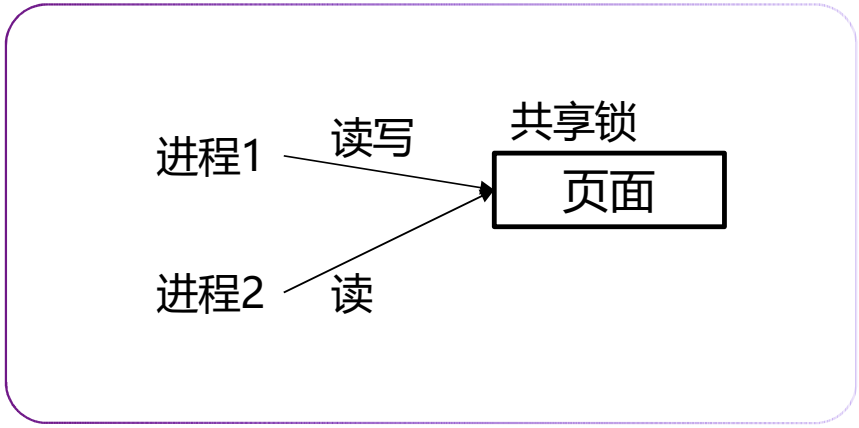
(1)如果该页面在缓冲区中，则写入缓存、日志和索引

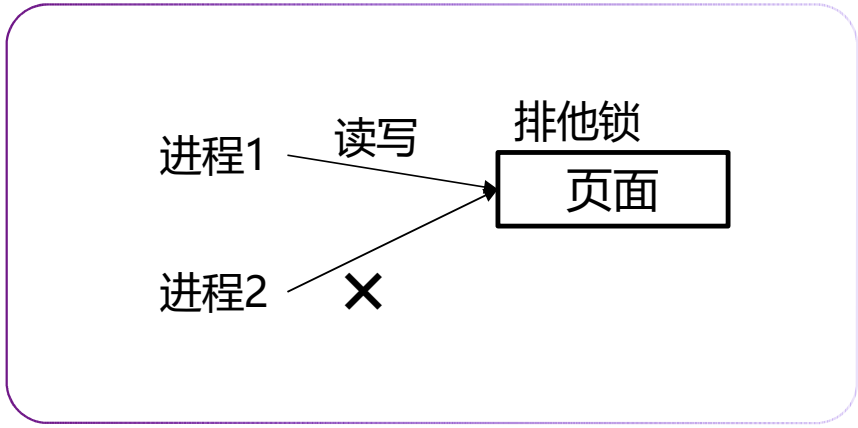
(2)如果该页面不在缓冲区中，则从磁盘上读取该页面，再写入页面到缓存、日志和索引



缓冲区页面固定和锁

- **当一个进程正在读取或写入某个页面时，该页面不应该被剔除，否则会读取或写入错误的数据**
  - 因此需要将使用中的页面暂时固定（pin）住
- **共享锁（shared locks）和排他锁（exclusive locks）**
  - 页面共享锁使多个进程可同时读取页面
  - 页面排他锁保证一个进程在修改页面时其他进程不可访问页面





缓冲区页面替换策略

□ 管理页面替换，最大化缓冲区命中率，最小化访问磁盘次数

- 最近最少使用策略LRU
  - 前1次使用时间越晚（越近期）的页面将来越有机会被使用，越不应该被替换
- 最不经常使用策略LFU
  - 使用频率越高的页面将来越有机会被使用，越不应该被替换
- LRU-K策略
  - 前第K次访问的时间越晚的页面将来越有机会被使用，越不应该被替换

缓冲区页面替换策略

口 最近最少使用策略LRU

– 前1次使用时间越晚（越近期）的页面将来越有机会被使用，越不应该被替换

访问	1	2	3	1	4	5	3	6
槽位1	1	1	1	1	1	1	1	6
槽位2		2	2	2	2	5	5	5
槽位3			3	3	3	3	3	3
槽位4					4	4	4	4

绿色为命中缓冲区不需要读取磁盘，蓝色格子为需要读取磁盘操作。

缓冲区页面替换策略

口 最近最少使用策略LRU

– 顺序扫描时的缺陷：顺序扫描的数据大于缓存大小时，LRU命中率降低。

访问	1	2	3	4	5	1	2	3
槽位1	1	1	1	1	5	5	5	5
槽位2		2	2	2	2	1	1	1
槽位3			3	3	3	3	2	2
槽位4				4	4	4	4	3

绿色为命中缓冲区不需要读取磁盘，蓝色格子为需要读取磁盘操作。

缓冲区页面替换策略

口 最不经常使用策略LFU

– 使用频率越高的页面将来越有机会被使用，越不应该被替换

访问	1	2	3	1	2	4	5	3	1	2
槽位1	1	1	1	1	1	1	1	1	1	1
槽位2		2	2	2	2	2	2	2	2	2
槽位3			3	3	3	3	5	5	5	5
槽位4						4	4	3	3	3

绿色为命中缓冲区不需要读取磁盘，蓝色格子为需要读取磁盘操作。

预写日志与故障恢复

口为什么使用日志：

- 如果缓冲区中的脏页面在写回到磁盘过程中发生故障，则无法保证写入的数据的正确性。
- 页面随机写→日志的顺序写

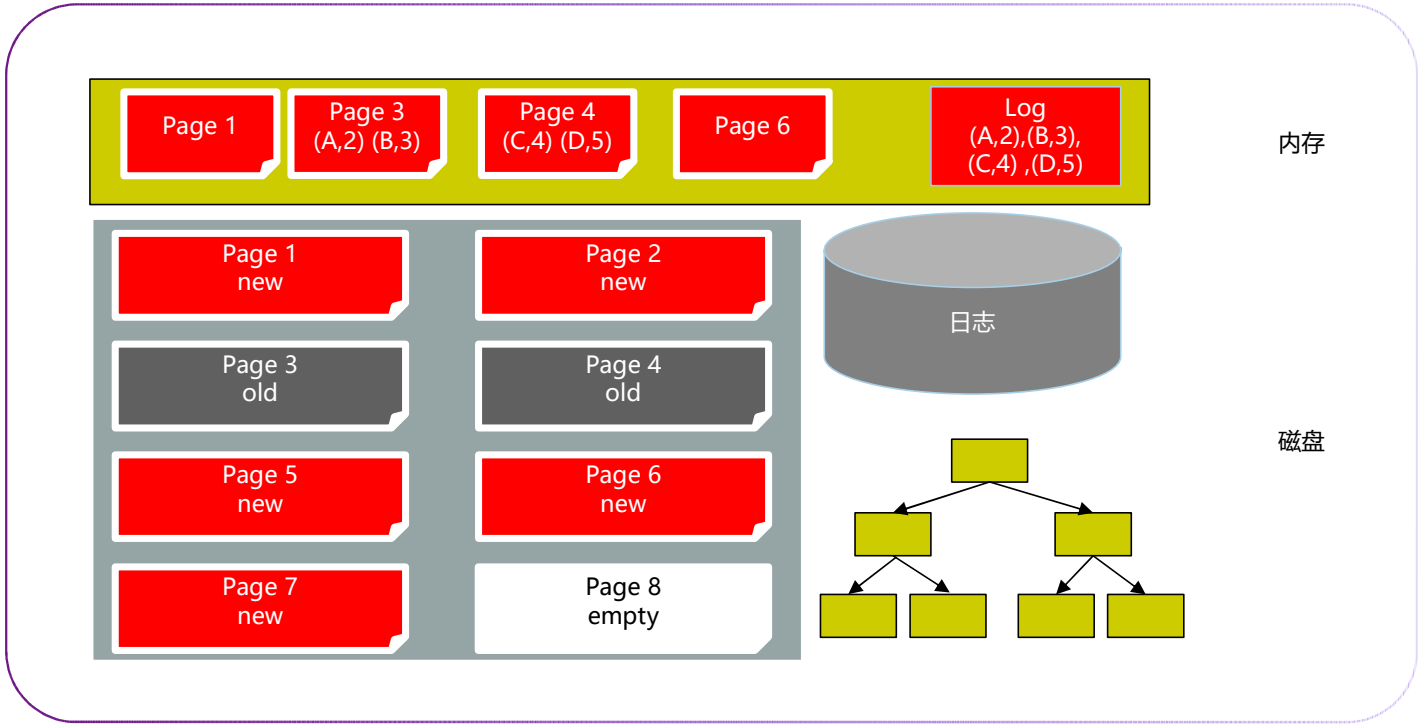
口数据库使用预写日志 （write-ahead log，WAL）来备份数据的修改以防丢失

- 写入数据前先写入日志，日志包含即将写入的页面编号和写入的内容，也可以包含原先页面中的内容作为备份以供恢复。
- 在日志写入到磁盘后再进行事务的提交，保证事务的持久性。
- 故障并重启时，扫描日志来判断每个操作的成功与失败，并决定撤销或者重做这些操作。

故障恢复

- 刷新日志后（写盘），但是还没有刷新数据页面，系统宕机如何恢复?
- 如何保障故障正确性?

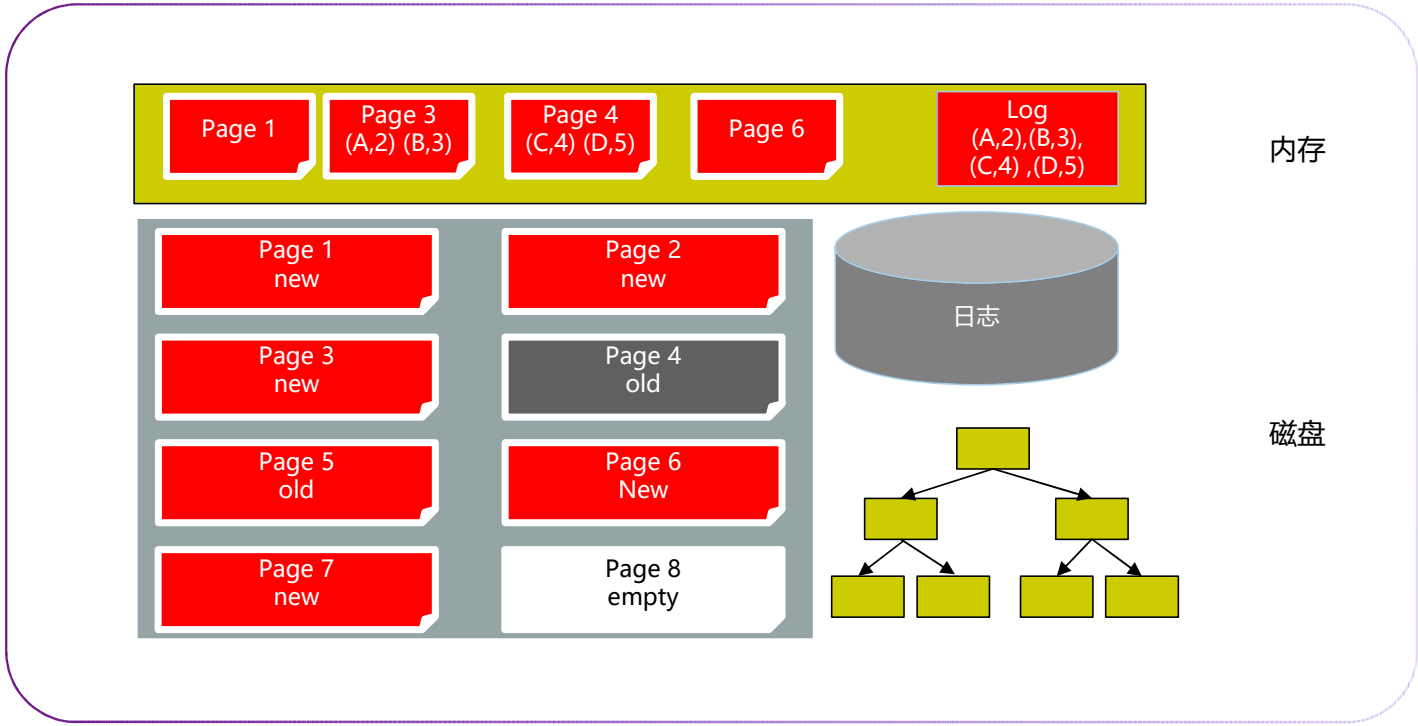
Insert (A, 2);
Insert (B, 3);
Insert (C, 4);
Insert (D, 5)
CRASH → Redo All



故障恢复

- 刷新日志后（写盘），但是还没有刷新数据页面，系统宕机如何恢复?
- 如何保障故障正确性?

Insert (A, 2);
Insert (B, 3);
刷盘
Insert (C, 4);
Insert (D, 5)
CRASH → Redo 4,5



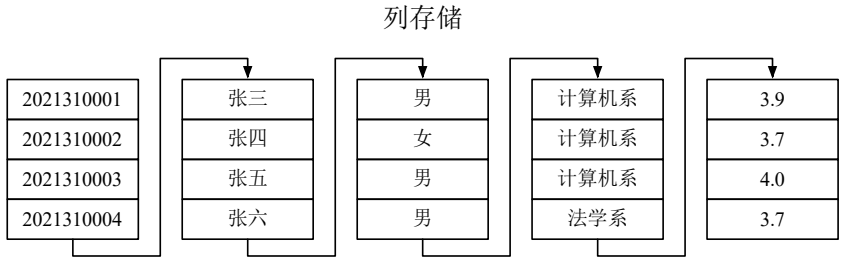
目录

行存储与列存储

- 行存储（row-oriented storage）：适合数据增删改
- 列存储（column-oriented storage）
  - 提高数据分析查询的速度；提升数据压缩率

学号	姓名	性别	院系	绩点
2021310001	张三	男	计算机系	3.9
2021310002	张四	女	计算机系	3.7
2021310003	张五	男	计算机系	4.0
2021310004	张六	男	法学系	3.7

学号	姓名	性别	院系	绩点
2021310001	张三	男	计算机系	3.9
2021310002	张四	女	计算机系	3.7
2021310003	张五	男	计算机系	4.0
2021310004	张六	男	法学系	3.7



行存储与列存储比较

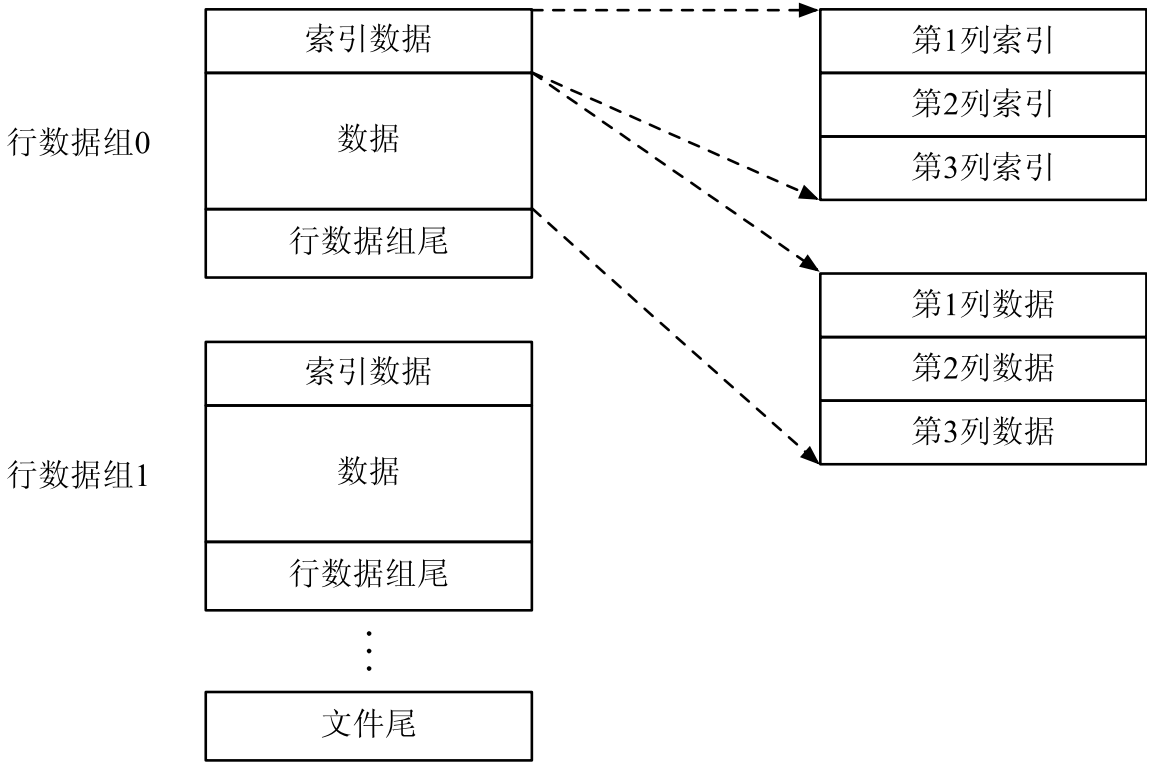
存储模型	优点	缺点
行存储	（1）数据被保存在一起 （2）插入和更新速度快	选择操作时即使只涉及某几列，所有数据也都会被读取
列存储	（1）查询操作时只有涉及的列会被读取 （2）投影操作很高效 （3）易于压缩	（1）选择操作完成时，被选择的列要重新组装 （2）插入和更新速度慢

行存储与列存储适用场景

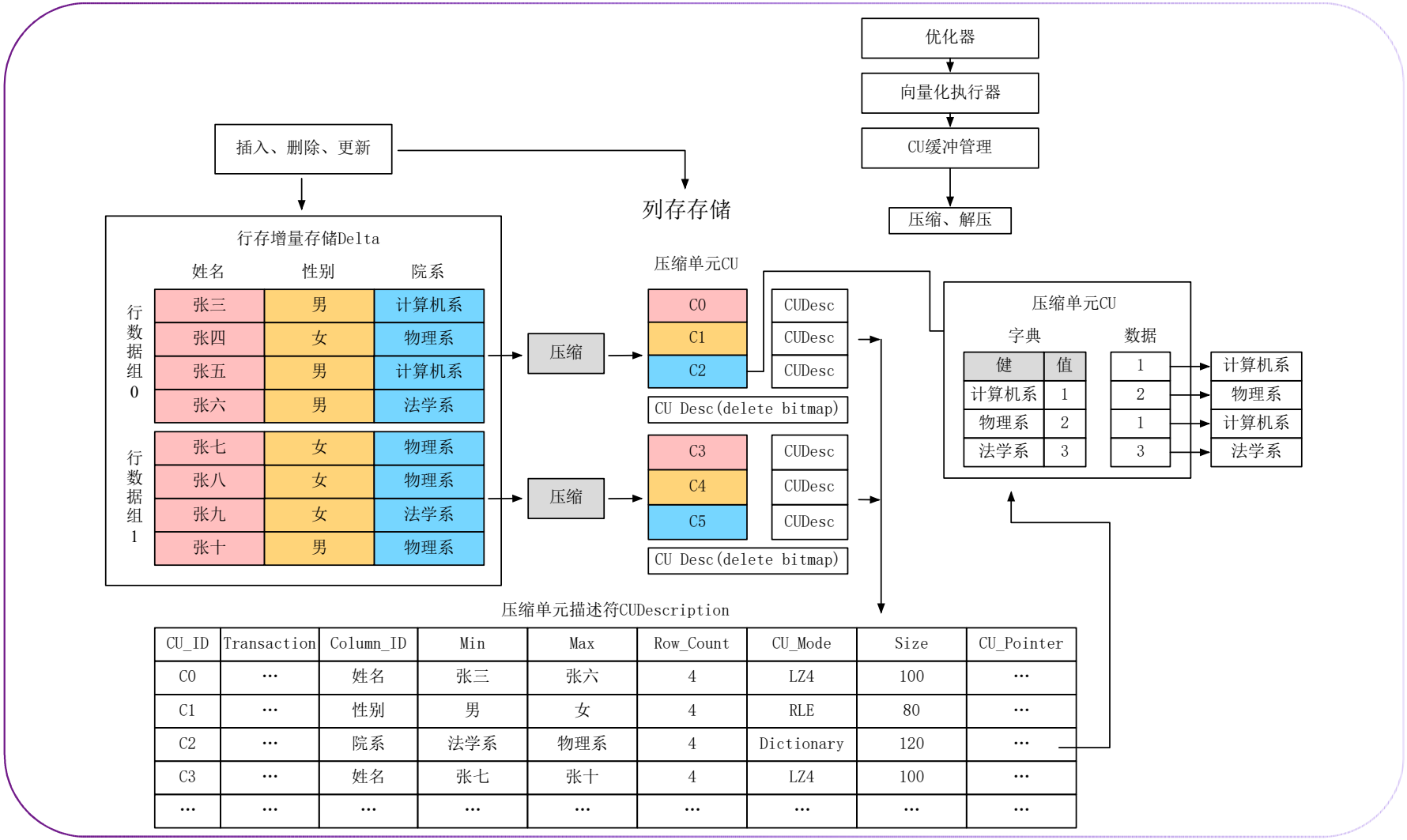
存储类型	适用场景
行存储	（1）点查询（返回记录少，基于索引的简单查询） （2）增、删、改操作较多的场景
列存储	（1）统计分析类查询（关联、分组操作较多的场景） （2）即席查询（用户灵活自定义条件，查询条件不确定，行存表扫描难以使用索引）

列存储文件组织

	学号	姓名
记录0	2021310001	张三
记录1	2021310002	张四
记录2	2021310003	张五
记录3	2021310004	张六
记录4	2021310006	张七
记录5	2021310009	张八
记录6	2021310013	张九
记录6	2021310013	张九
记录6	2021310013	张九
记录6	2021310013	张九
记录6	2021310013	张九



行列转换


优化器
向量化执行器
CU缓冲管理
压缩、解压
插入、删除、更新
列存存储
行存增量存储Delta
姓名 性别 院系
行数据组0
张三 男 计算机系
张四 女 物理系
张五 男 计算机系
张六 男 法学系
行数据组1
张七 女 物理系
张八 女 物理系
张九 女 法学系
张十 男 物理系
压缩
压缩
压缩单元CU
C0
C1
C2
CUDesc
CUDesc
CUDesc
CU Desc(delete bitmap)
C3
C4
C5
CUDesc
CUDesc
CUDesc
CU Desc(delete bitmap)
压缩单元CU
字典
健 值
计算机系 1
物理系 2
法学系 3
数据
1
2
1
3
计算机系
物理系
计算机系
法学系
压缩单元描述符CUDescription
CU_ID Transaction Column_ID Min Max Row_Count CU_Mode Size CU_Pointer
C0 … 姓名 张三 张六 4 LZ4 100 …
C1 … 性别 男 女 4 RLE 80 …
C2 … 院系 法学系 物理系 4 Dictionary 120 …
C3 … 姓名 张七 张十 4 LZ4 100 …
… … … … … … … … …

本章小结

- 数据库存储是数据库的核心模块，负责将用户的数据持久保存在存储介质中同时又要满足高并发访问、高I/O性能、故障恢复等要求。
- 数据库将关系表用一个或者多个文件存储，关系表中的记录按顺序排列在文件中。为了在海量的记录中高效的增、删、改、查，数据库设计了多种文件组织方法。
- 数据库将磁盘数据块统一用页面进行管理。页面包含多条记录，记录分为定长记录和变长记录。
- 缓冲区用于在主存中暂时保存部分磁盘页面的拷贝，可以大大减少数据库访问磁盘的次数，提高查询速度。缓冲区管理器根据页面替换策略管理页面的读入和换出。
- 行存储和列存储通过不同的存储顺序应对事务性场景和分析型场景。