

Hash-Based Indexes

SUN YAT-SEN UNIVERSITY

Review

索引技术概述

- 可以为关系建立索引，都是文件

- 索引文件由两部份组成

1. 数据项部分

- Data Entry**(数据项) \longleftrightarrow **data record** (数据记录)

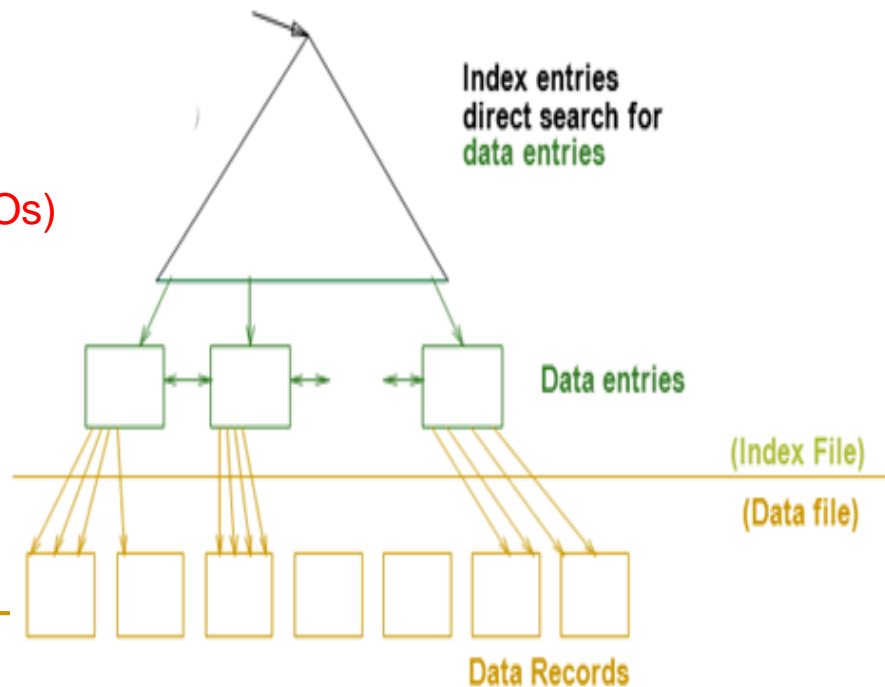
2. 引导部份

- 树索引技术

$\text{Cost} = \log_F N$ (2~3 I/Os)

- Hash索引

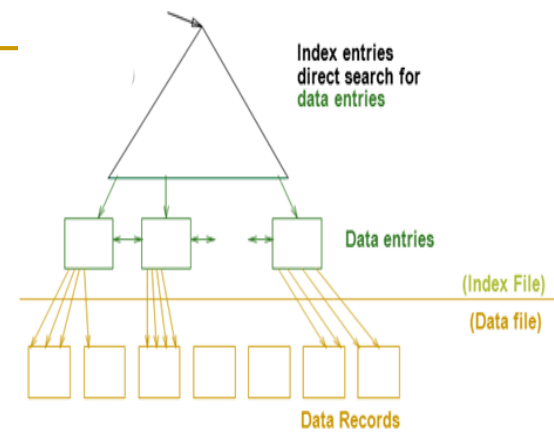
1~2 I/Os



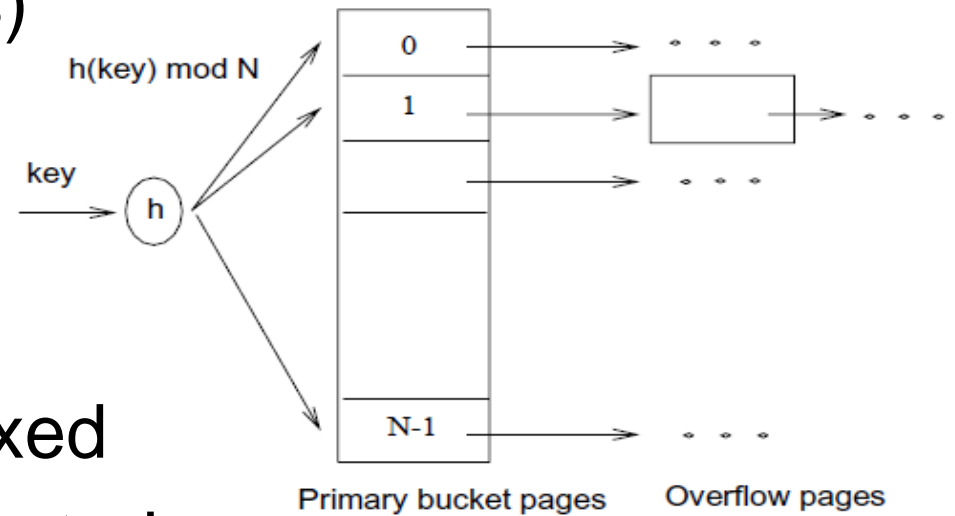
Introduction

- Hash-based indexes are best for equality selections. Cannot support range searches.
 - Static and dynamic hashing techniques exist; trade-offs similar to ISAM vs. B+ trees.
-

Static Hashing(静态哈希)

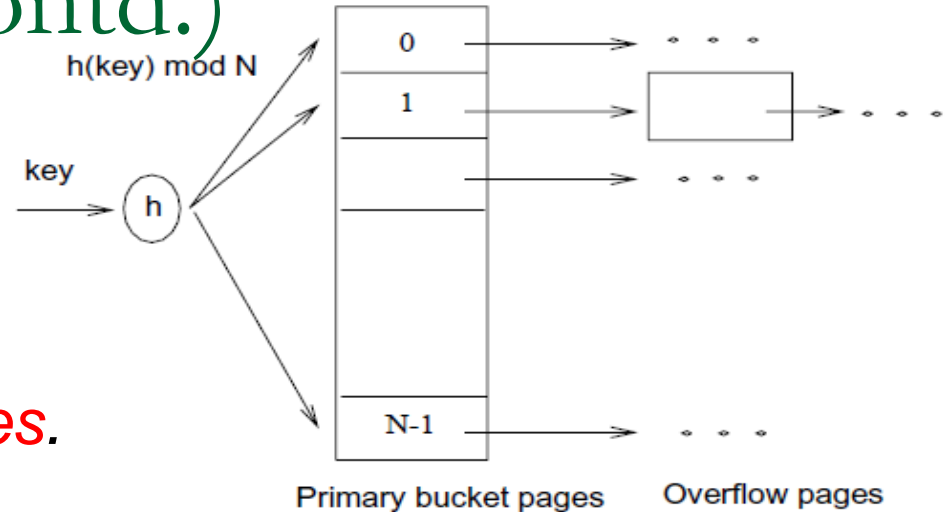
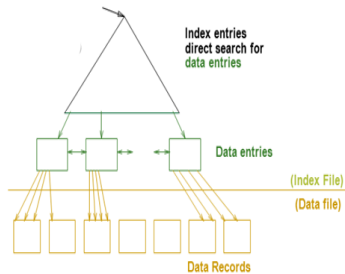


- 索引文件由一系列桶(bucket)组成
 - 每个桶有一个主页(primary page),也可能有一些溢出页(overflow pages)



- 静态性
 - Number of buckets is fixed
 - Primary pages are allocated sequentially, never deallocated;
 - overflow pages if needed.

Static Hashing (Contd.)

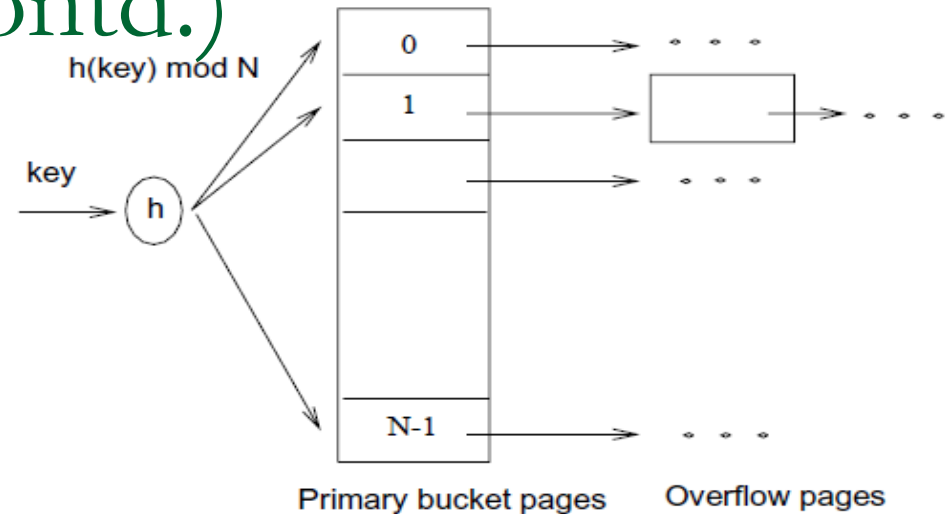


- Buckets contain *data entries*.
- 定位所属的桶?

$h(k) \bmod N$ = bucket to which data entry with key k belongs.

- N = number of buckets
- Hash function(哈希函数) works on *search key* field of record r . Must distribute values over range $0 \dots N-1$.
 - $h(\text{key}) = (a * \text{key} + b)$ usually works well.
 - a and b are constants;

Static Hashing (Contd.)



- Long overflow chains(长溢出链) can develop and degrade performance.
 - *Extendible and Linear Hashing: Dynamic techniques to fix this problem.*