CS 186 Discussion Section
Week 3

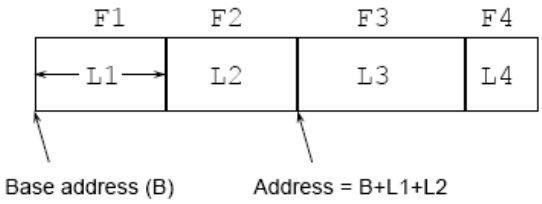
Peter Alvaro, Kuang Chen

February 9, 2009

Record Formats

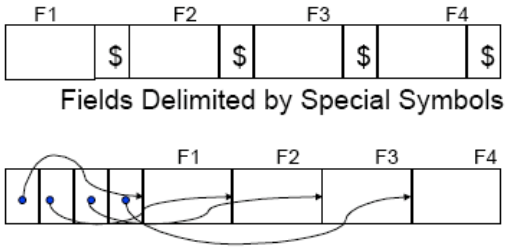
- Domain (type) of attributes (columns) help determine whether a record is fixed or variable length

Fixed length records



- Lengths of fields are stored in the *catalog*, thus finding fields is simple arithmetic

Two alternatives for variable Length records



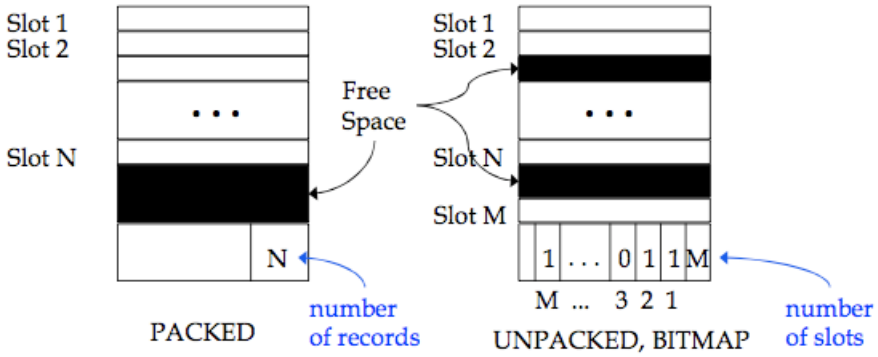
- 2nd alternative has direct access to fields; requires overhead for a directory (but we can put *null*s there)

Tradeoffs between fixed-length and variable-length records

- Fixed-length makes it easier to find a space to insert a record
- Fixed-length can have wasted space within its fixed-size fields in a record
- Variable-length can lead to fragmentation between records
- Variable-length doesn't waste space within a record

Page Layouts

Fixed length page layouts

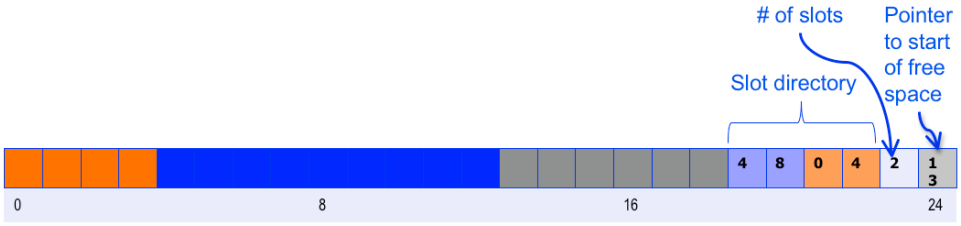


Tradeoffs between packed and unpacked layouts

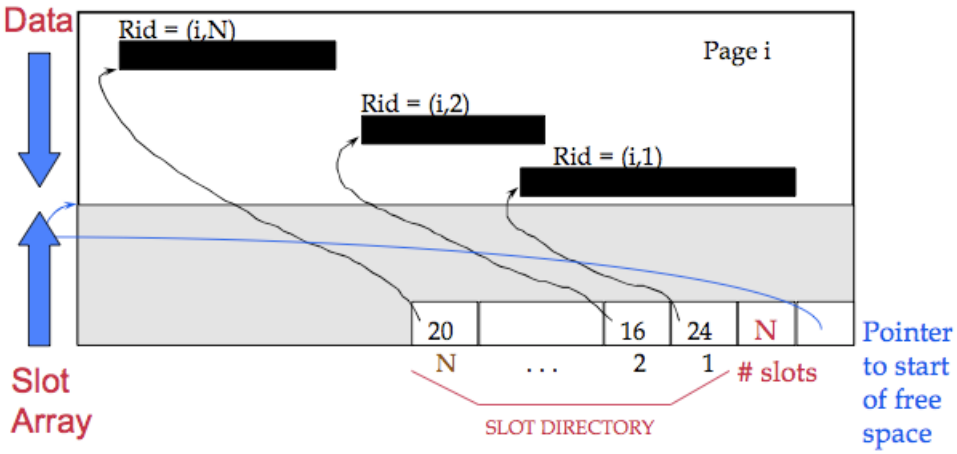
- Packed makes deletions more difficult because entries have to be moved
- With unpacked, deletion only involves clearing the bit
- Packed makes insertions easier because don't need to search for empty slot
- With unpacked, insertion involves first determining an empty slot

Question

Are pages holding short fixed-length record "really tall"?!? NO! Here's how it works in "real life", a page is an array of bytes and each slot is indexed *row-major* into the page array:



Variable length page layout

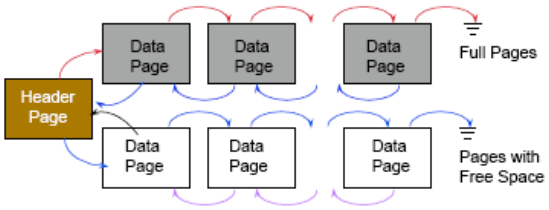


Heap Files

- A heap file is a collection of records in no particular order
- Access a record by specifying its *rid*, (page, offset/slot)
- Supports efficient *scan* operation
- Not good at value-based queries (we'll talk about Indexes!)

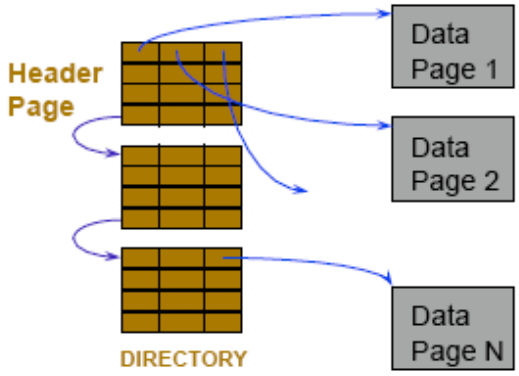
Two alternatives for organizing pages of Heap Files

The linked list approach



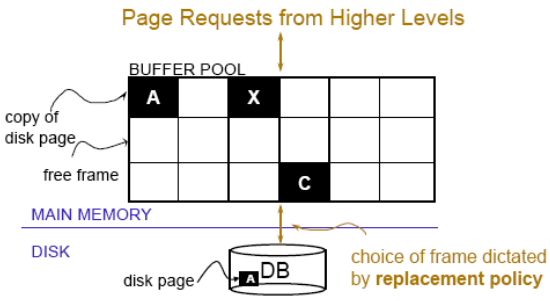
- Each page reserves space for two pointers
- Searching for free space could be expensive, why?

The directory approach



- A page of entries (pointers) at a time is much more efficient
- A directory entry can contain the size of a page's free space

Buffer Management



- Buffer manager copies blocks from disk, into frames of the buffer pool.
- Once in memory, pages (aka blocks in frames), can be pinned, marked dirty and later flushed to disk
- A page is a candidate for replacement if and only if its not pinned
- Strategy of replacing buffer pages in your first homework!