

Midterm Review

CPS 216
Advanced Database Systems

Announcements (March 8)

- ❖ Homework #2 has been graded
 - Sample solution is also available
 - Check your grades on Blackboard
- ❖ No reading assignment this week
- ❖ Project proposal due today
- ❖ Midterm exam on Thursday in class
 - Open book, open notes

Review: basics

- ❖ Relational model/algebra → physical data independence
- ❖ Design theory (FD's, BCNF) → help eliminate redundancy
- ❖ SQL
 - NULL and three-value logic → nifty feature, big mess
 - Bag versus set semantics
 - Subqueries, grouping and aggregation → which features add more expressiveness?
 - Views → logical data independence
 - Materialized views → reintroduce redundancy to improve performance
 - Constraints → the more you know the better you can do

Review: physical data organization

- ❖ Storage hierarchy (DC vs. Pluto)
 - Count I/O's
 - Get as much useful info as possible with each long trip
 - Do other things while waiting
- ❖ Disk performance → sequential beats random
- ❖ Data layout
 - Record layout (handling variable-length fields, NULL's)
 - Block layout (NSM, DSM, PAX)
 - Inter-/intra-record locality

Review: physical data organization (cont'd)

- ❖ Access paths
 - Primary versus secondary indexes
 - Tree-based indexes: ISAM, B⁺, B, R, R*, R⁺, GIST
 - Hash-based indexes: extensible, linear
 - Text indexes: inverted lists, signature files (and bit-sliced ones), suffix array, trie, suffix tree, Patricia trie, Pat tree
 - Variant indexes: value-list/bitmap, projection, bit-sliced indexes, join indexes
- Reintroduce redundancy to improve performance
- Fundamental trade-off: query versus update cost

Review: query processing

- ❖ Scan-based algorithms
- ❖ Sort- and hash-based algorithms (and their duality)
- ❖ Index-based algorithms

- ❖ Pipelined execution with iterators
 - Blocking and non-blocking operators
- ❖ Buffer management
 - Per-query, per-table policy is ideal
- The more you know the better you can do