ICS 421 Spring 2010
Performance Tuning

Asst. Prof. Lipyeow Lim
Information & Computer Science Department
University of Hawaii at Manoa
Performance Tuning

Given

• a database
  – Tables (schema etc)
  – Data

• a workload
  – Queries and their frequency
  – Updates and their frequency

• DBMS software running on some hardware

What knobs can you play with to improve performance?
Knobs & Factors

**Knobs**
- Indexes
- Query rewriting
- Table schema
- Locking
- Logging
- Hardware
- Memory

**Factors**
- Data Size
- Budget
- Purpose
- Workload
  - Read intensive vs write intensive
  - Types of queries
  - Frequencies
Query 100: Brute Force Cone Search

```sql
SELECT O.objID, O.ra, O.dec, O.hmid, O.zoneid
FROM Object O
WHERE ( SIN(RADIANS(O.dec))
  * SIN(RADIANS(+0.5))
+ COS(RADIANS(O.dec))
  * COS(RADIANS(+0.5))
  * COS(RADIANS((O.ra) - (67.5)))
) >= COS(RADIANS(1.0/60.0))
```
Query 101: Prefiltering using ZoneID

SELECT O.objID, O.ra, O.dec, O.htmid, O.zoneid
FROM Object O
WHERE (zoneid BETWEEN FLOOR((90.0 + 0.5 - (1.0/60.0))/0.008333) AND FLOOR((90.0 + 0.5 + (1.0/60.0))/0.008333))
AND ( SIN(RADIANS(O.dec)) * SIN(RADIANS( +0.5)) + COS(RADIANS(O.dec)) * COS(RADIANS( +0.5)) * COS(RADIANS((O.ra) - (67.5))) ) >= COS(RADIANS( 1.0/60.0))
Query 103: Prefiltering using a Pyramid

```
SELECT O.objID, O.ra, O.dec, O.htmid, O.zoneid
FROM Object O
WHERE O.ra BETWEEN ((67.5)-(1.0/60.0)) AND ((67.5)+(1.0/60.0))
AND O.dec BETWEEN ((+0.5)-(1.0/60.0)) AND ((+0.5)+(1.0/60.0))
AND (SIN(RADIANS(O.dec)) * SIN(RADIANS(+0.5))
+ COS(RADIANS(O.dec))* COS(RADIANS(+0.5))
* COS(RADIANS((O.ra) - (67.5))))
>= COS(RADIANS(1.0/60.0))
```
Query 110: Join with Detection

```
SELECT O.objID, O.ra, O.dec, O.htmid, O.zoneid, D.detectid
FROM Object O, Detection D
WHERE O.objid=D.objid
AND ( SIN(RADIANS(O.dec)) * SIN(RADIANS( +0.5))
+ COS(RADIANS(O.dec)) * COS(RADIANS( +0.5))
* COS(RADIANS(((O.ra) - (67.5)))) ) >=
COS(RADIANS( 1.0/60.0))
```
CREATE TABLE Object (  
  objID BIGINT,  
  htmID BIGINT,  
  zoneID INT,  
  ra DOUBLE,  
  dec DOUBLE,  
  cx DOUBLE,  
  cy DOUBLE,  
  cz DOUBLE,  
  lambda FLOAT,  
  beta FLOAT,  
  l FLOAT,  
  b FLOAT,  
  lsg FLOAT,  
  bsg FLOAT,  
  gMagBest REAL,  
  rMagBest REAL,  
  iMagBest REAL,  
  zMagBest REAL,  
  yMagBest REAL,  
  grColor REAL,  
  riColor REAL,  
  izColor REAL,  
  zyColor REAL,  
  sgSep REAL )  
CREATE TABLE Detection (  
  objID BIGINT,  
  detectID BIGINT,  
  filterID SMALLINT,  
  imageID BIGINT,  
  obsTime FLOAT,  
  raObs FLOAT,  
  decObs FLOAT,  
  mag REAL,  
  sky REAL,  
  sgSep REAL );
CREATE TABLE DETECTION201001(....)
CREATE TABLE DETECTION201002(....)
CREATE TABLE DETECTION201003(....)

ALTER TABLE DETECT201001 ADD CONSTRAINT CHK_JAN
    CHECK (MONTH(obsTime) =1);
...
INSERT INTO ...
...

CREATE VIEW DETECTION AS
    SELECT * FROM DETECTION201001
    UNION ALL
    SELECT * FROM DETECTION201002
    UNION ALL
    SELECT * FROM DETECTION201003
Performance Tuning Tools

• Explain
• Not getting the right plans? runstats
• Twisting the arm of the optimizer using selectivity clause
• Event Monitors
• Other smart tools
  – Index advisors
  – Schema advisors
  – Query patroller