

3

CMU SCS

## Overview

- Introduction
- Index selection and clustering
- Database tuning (de-normalization etc)
- Impact of concurrency

Faloutsos

CMU SCS 15-415

CMU SO

#### Introduction

- After ER design, schema refinement, and the definition of views, we have the *conceptual* and *external* schemas for our database.
- Next step?

Faloutsos



CMO SCS

## Introduction

- After ER design, schema refinement, and the definition of views, we have the *conceptual* and *external* schemas for our database.
- Next step?
- choose indexes, make clustering decisions, and to refine the conceptual and external schemas (if necessary) to meet performance goals.
- How to decide the above?

Faloutsos

MU SCS 15-41



#### Introduction

- We must begin by understanding the workload:
  - The most important queries and how often they arise.
  - The most important updates and how often they arise.
  - The desired performance for these queries and updates.

Faloutsos

CMU SCS 15-415



## Decisions to Make

• ??

Faloutsos



CMU SC

#### Decisions to Make

- What indexes should we create?
- For each index, what kind of an index should it be?
- Should we make changes to the conceptual schema?

Faloutsos

CMU SCS 15-415



CMU SCS

#### Decisions to Make

- What indexes should we create?
- Which relations should have indexes? What field(s) should be the search key? Should we build several indexes?
- For each index, what kind of an index should it be?
  - Clustered? Hash/tree?
- Should we make changes to the conceptual schema?
  - Consider alternative normalized schemas? (Remember, there are many choices in decomposing into BCNF, etc.)
  - Should we ``undo" some decomposition steps and settle for a lower normal form? (*Denormalization*.)
  - Horizontal partitioning, replication, views ...

Faloutsos

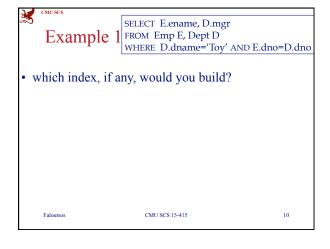
CMU SCS 15-415

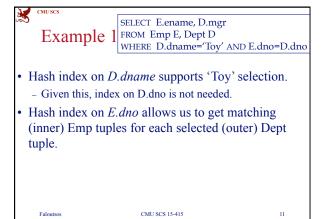


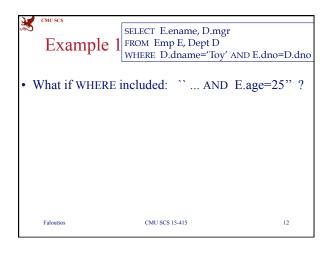
#### Overview

- Introduction
- **➡** Index selection and clustering
  - Database tuning (de-normalization etc)
  - Impact of concurrency

Faloutsos







CMU SCS 15-415 Faloutsos, Gibson



SELECT E.ename, D.mgr Example 1 FROM Emp E, Dept D WHERE D.dname='Toy' AND E.dno=D.dno

• What if WHERE included: `` ... AND E.age=25"?

- Could retrieve Emp tuples using index on *E.age*, then join with Dept tuples satisfying dname selection. Comparable to strategy that used *E.dno* index.
- So, if *E.age* index is already created, this query provides much less motivation for adding an E.dno index.

Faloutsos

CMU SCS 15-415

13



SELECT E.ename, D.mgr FROM Emp E, Dept D WHERE E.sal BETWEEN 10000 AND 20000 AND E.hobby='Stamps' AND E.dno=D.dno

Faloutsos

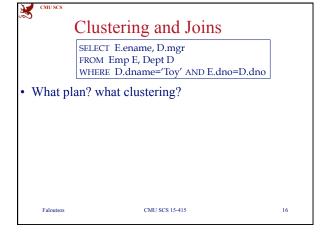
CMU SCS 15-415



SELECT E.ename, D.mgr FROM Emp E, Dept D WHERE E.sal BETWEEN 10000 AND 20000 AND E.hobby='Stamps' AND E.dno=D.dno

- Clearly, Emp should be the outer relation.
  - Suggests that we build a hash index on D.dno.
- What index should we build on Emp?
  - B+ tree on E.sal could be used, OR an index on E.hobby could be used. Only one of these is needed, and which is better depends upon the selectivity of the conditions.
    - · As a rule of thumb, equality selections more selective than range selections.
- · As both examples indicate, our choice of indexes is guided by the plan(s) that we expect an optimizer to consider for a query. Have to understand optimizers!

CMU SCS 15-415





## Clustering and Joins

SELECT E.ename, D.mgr FROM Emp E, Dept D WHERE D.dname="Toy" AND E.dno=D.dno

- Clustering is especially important when accessing inner tuples in INL.
  - Should make index on *E.dno* clustered.

Faloutsos CMU SCS 15-415 17



#### Overview

- Introduction
- Index selection and clustering
- Database tuning (de-normalization etc)
  - Impact of concurrency

Faloutsos CMU SCS 15-415 20



# Tuning the Conceptual Schema

- The choice of conceptual schema should be guided by the workload, in addition to redundancy issues:
  - We may settle for a 3NF schema rather than BCNF.
  - Workload may influence the choice we make in decomposing a relation into 3NF or BCNF.
  - We may further decompose a BCNF schema!
  - We might *denormalize* (i.e., undo a decomposition step), or we might add fields to a relation.
  - We might consider horizontal decompositions.

Faloutsos

CMU SCS 15-415

21



CMUSCS

# Tuning the Conceptual Schema

- If such changes are made after a database is in use: called *schema evolution*
- Q: How to mask these changes from applications?

Faloutsos

CMU SCS 15-415

22



CMU SCS

## Tuning the Conceptual Schema

- If such changes are made after a database is in use: called *schema evolution*
- Q: How to mask these changes from applications?
- A: Views!

Faloutsos

CMU SCS 15-415

CMU SCS	
Tuning the Conceptual Schema	
The choice of conceptual schema should be guided by the workload, in addition to redundancy issues:	
<ul> <li>We may settle for a 3NF schema rather than BCNF.</li> <li>Workload may influence the choice we make in decomposing a relation into 3NF or BCNF.</li> </ul>	
<ul><li>We may further decompose a BCNF schema!</li><li>We might <i>denormalize</i> (i.e., undo a decomposition step),</li></ul>	
or we might add fields to a relation.  - We might consider <i>horizontal decompositions</i> .	
Faloutsos CMU SCS 15-415 24	
CMU SCS	
Example?	
• Q: When would we choose 3NF instead of BCNF?	
Faloutsos CMU SCS 15-415 25	
CMU SCS	
S Carrier Carr	
Example?	
• Q: When would we choose 3NF instead of BCNF?	
• A: Student-Teacher-subJect (STJ) S J -> T	

T -> J

and queries ask for all three attributes ( <code>select \*</code> )

CMU SCS 15-415

CMU SCS 15-415 Faloutsos, Gibson



# Tuning the Conceptual Schema

- The choice of conceptual schema should be guided by the workload, in addition to redundancy issues:
  - We may settle for a 3NF schema rather than BCNF.
  - Workload may influence the choice we make in decomposing a relation into 3NF or BCNF.
- We may further decompose a BCNF schema!
- We might *denormalize* (i.e., undo a decomposition step), or we might add fields to a relation.
- We might consider *horizontal decompositions*.

CMU SCS 15-415

27



## Decomposition of a BCNF Relation

· Q: Scenario?

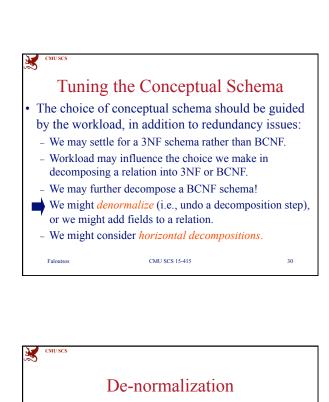
CMU SCS 15-415



## Decomposition of a BCNF Relation

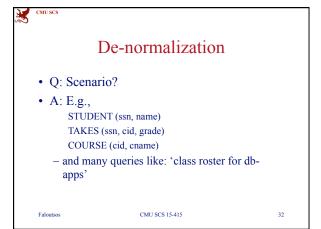
- Q: Scenario?
- A: eg., STUDENT(ssn, name, address, ph#, ...)
- with many queries like select ssn, name from student

Faloutsos



Faloutsos CMU SCS 15-415 31

• Q: Scenario?





CMU SCS

## Tuning the Conceptual Schema

- The choice of conceptual schema should be guided by the workload, in addition to redundancy issues:
  - We may settle for a 3NF schema rather than BCNF.
  - Workload may influence the choice we make in decomposing a relation into 3NF or BCNF.
  - We may further decompose a BCNF schema!
  - We might denormalize (i.e., undo a decomposition step), or we might add fields to a relation.
  - We might consider *horizontal decompositions*.

Faloutsos

CMU SCS 15-415

33



MUSCS

## **Horizontal Decompositions**

Sometimes, might want to replace relation by a collection of relations that are *selections*. Eg.,

STUDENT (ssn, name, status)

decomposed to

CurrentStudent (ssn, name, status)

Alumni (ssn, name, status)

Q: under what scenario would this help performance?

Faloutso

CMU SCS 15-415

34



# Masking Conceptual Schema

Changes

CREATE VIEW STUDENT(ssn, name, status)

AS SELECT '

FROM CurrentStudent

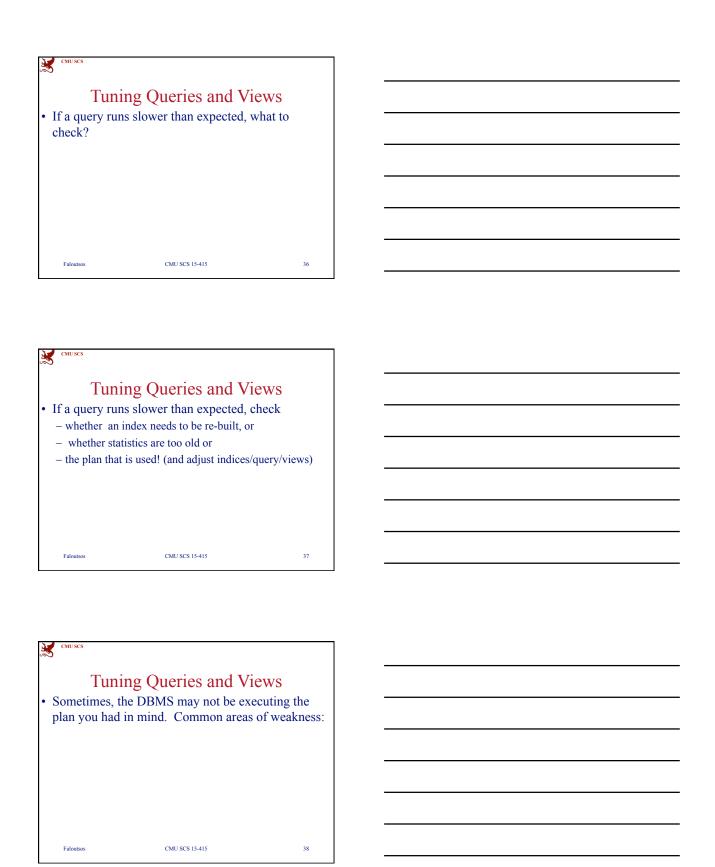
UNION

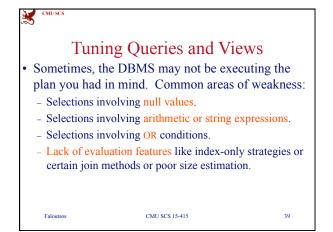
SELECT \* FROM Alumni

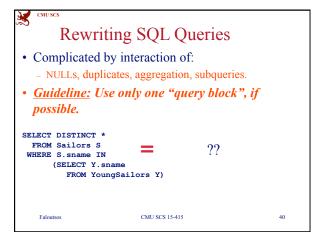
- · Masks change
- But performance-minded users should query the right table

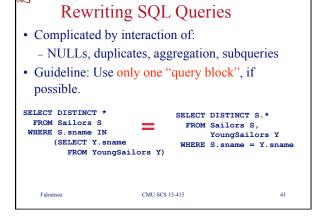
Faloutsos

CMU SCS 15-415









42



CMU SC

## More Guidelines for Query Tuning

 Minimize the use of DISTINCT: don't need it if duplicates are acceptable, or if answer contains a key.

Faloutsos

CMU SCS 15-415



CMU SCS

## More Guidelines for Query Tuning

- Consider DBMS use of index when writing arithmetic expressions:
- *E.age*=2\**D.age* will benefit from index on *E.age*, but might not benefit from index on *D.age*!

Faloutsos

CMU SCS 15-415

S 15-415



CMU SC

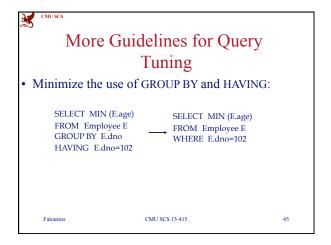
# More Guidelines for Query Tuning

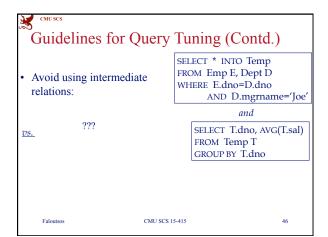
• Minimize the use of GROUP BY and HAVING:

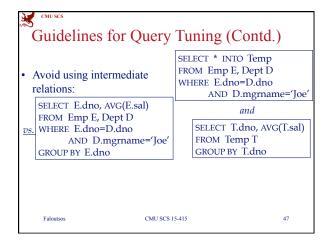
SELECT MIN (E.age) FROM Employee E GROUP BY E.dno HAVING E.dno=102

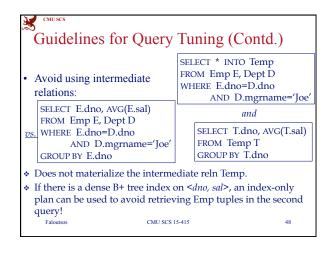
??

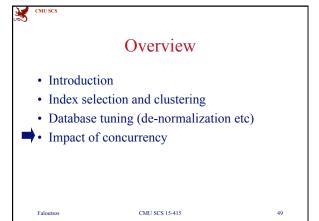
Faloutsos

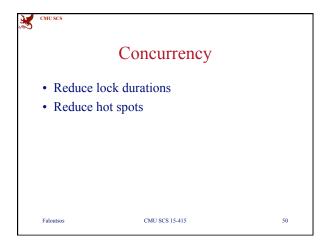


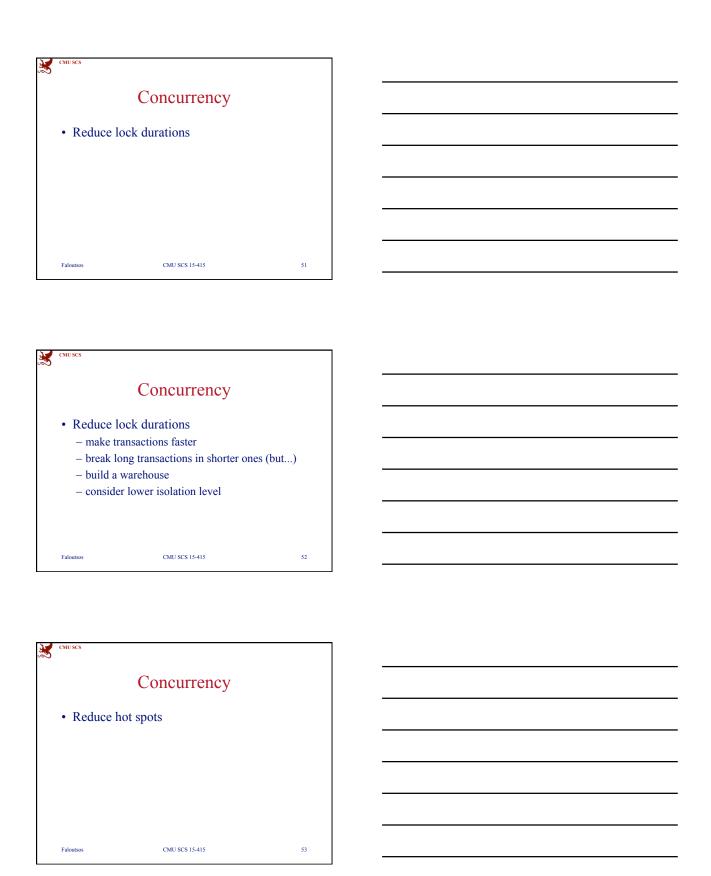














CMU SC

## Concurrency

- Reduce hot spots
  - delay operations on hot spots
  - optimize access patterns
  - partition (batch) operations on hot spots
  - choice of index (root of B-tree -> hot spot)

Faloutsos

CMU SCS 15-415

54



CMU SCS

## Summary

- Database design consists of several tasks: requirements analysis, conceptual design, schema refinement, physical design and tuning.
  - In general, have to go back and forth between these tasks to refine a database design, and decisions in one task can influence the choices in another task.

Also see the paper by Roussopoulos + Yeh (on the course web site)

Faloutsos

CMU SCS 15-415

55



CMU SCS

## Summary (cont'd)

- Understanding the nature of the *workload* is vital:
  - What are the important queries and updates? What attributes/relations are involved?
- · then:
  - refine conceptual schema and views
  - tune queries (indices, clustering, re-writing)

Faloutsos

CMU SCS 15-415



## Summary - schema refinement

• May choose 3NF or lower normal form over BCNF.

- May denormalize, or undo some decompositions.
- May decompose a BCNF relation further!
- May choose a *horizontal decomposition* of a relation.
- Importance of dependency-preservation based upon the dependency to be preserved, and the cost of the IC check (see text)

Faloutsos

CMU SCS 15-415

57



MUSCS

## Summary - Tuning

Tuning: on slow queries, check the chosen plan!

- Over time, indexes have to be fine-tuned (dropped, created, re-built, ...) for performance.
- System may still not find a good plan:
  - Only left-deep plans considered!
  - Null values, arithmetic conditions, string expressions, the use of ORs, etc. can confuse an optimizer.

Faloutso

CMU SCS 15-415

58



CMU SC

## Summary - Tuning

So, may have to rewrite the query/view: Avoid

- · nested queries,
- · temporary relations,
- · complex conditions, and
- operations like DISTINCT and GROUP BY.

Faloutsos