

CMPSCI 105: Lecture #12 Joins and Indexing

©2015 Dr. William T. Verts

INDEXING

- Indexes are additional hidden data structures associated with individual fields in a table.
- Indexes are maintained automatically by the database package.
- An index makes a table “look sorted” on the indexed field.
- Every field may have its own index.
- Indexes speed up both searches and joins.

©2015 Dr. William T. Verts

Original Tables, No Indexing

Neither Table has an Index

PEOPLE

	ID	NAME
1	12345	Fred
2	72401	Joe
3	22222	Mary
4	54321	Sam
5	20202	Martha
6	11111	Bob
7	47904	Tom

PAYMENTS

	ID	SALARY
1	60233	\$30,000
2	54321	\$20,000
3	11111	\$40,000
4	97330	\$50,000
5	72401	\$35,000
6	13333	\$45,000

©2015 Dr. William T. Verts

Tables without Indexes

- In a table without any indexes, data records may appear in any random order.
- A table might have its records in a sorted order without an index.
- Searching an unindexed table for matching records requires examining every record in the table. We call this a linear search, which runs in $O(N)$ “Order N” time for N records.

©2015 Dr. William T. Verts

Left Table Indexed on ID Field

Left Table has an Index

PEOPLE

	ID	NAME
1	11111	Bob
2	12345	Fred
3	20202	Martha
4	22222	Mary
5	47904	Tom
6	54321	Sam
7	72401	Joe

PAYMENTS

	ID	SALARY
1	60233	\$30,000
2	54321	\$20,000
3	11111	\$40,000
4	97330	\$50,000
5	72401	\$35,000
6	13333	\$45,000

©2015 Dr. William T. Verts

Left Table Indexed on ID Field

- The left table (People) *apparently* is indexed on the ID field.
- The right table (Payments) is still clearly not indexed.
- People might be unindexed, but if it is it would have to be on the ID field.

©2015 Dr. William T. Verts

Both Indexed on ID Field

Both Tables have Indexes

PEOPLE

	ID	NAME
1	11111	Bob
2	12345	Fred
3	20202	Martha
4	22222	Mary
5	47904	Tom
6	54321	Sam
7	72401	Joe

PAYMENTS

	ID	SALARY
1	11111	\$40,000
2	13333	\$45,000
3	54321	\$20,000
4	60233	\$30,000
5	72401	\$35,000
6	97330	\$50,000

©2015 Dr. William T. Verts

Both Indexed on ID Field

- Both tables are apparently sorted on the ID field.
- It could be possible that one or both aren't indexed.
- If one or both are indexed, they would have to be on the ID field.

©2015 Dr. William T. Verts

Searching Indexed Tables

- Typically, indexed fields contain unique information (no duplicates).
- Such fields can be searched with binary search in $O(\log_2(N))$ time. (Check middle item, discard half not containing search term, repeat until found or list becomes empty).

©2015 Dr. William T. Verts

JOINS

- Something spreadsheets can't do.
- Synthesize new tables from two or more source tables.
- Source tables must have a field in common (doesn't have to have the same name, but must have compatible data types).
- Result may contain more records than either source table, or may be empty.

©2015 Dr. William T. Verts

Types of Joins

- Inner Join (intersection of tables)
- Outer Join
 - Left Outer Join
 - Right Outer Join
 - Full Outer Join (union of tables)

©2015 Dr. William T. Verts

Here are the Source Tables again

Neither Table has an Index

PEOPLE

	ID	NAME
1	12345	Fred
2	72401	Joe
3	22222	Mary
4	54321	Sam
5	20202	Martha
6	11111	Bob
7	47904	Tom

PAYMENTS

	ID	SALARY
1	60233	\$30,000
2	54321	\$20,000
3	11111	\$40,000
4	97330	\$50,000
5	72401	\$35,000
6	13333	\$45,000

©2015 Dr. William T. Verts

Inner Join on ID Field (Intersection)

Inner Join

	ID	NAME	SALARY
1	11111	Bob	\$40,000
2	54321	Sam	\$20,000
3	72401	Joe	\$35,000

©2015 Dr. William T. Verts

Inner Join

- Contains only records where join-field in one table matches join-field in the other table.
- In this case, there are only three matches. Those three results will be filled in with data from both tables.

©2015 Dr. William T. Verts

Left Outer Join on ID Field

Left Outer Join

	ID	NAME	SALARY
1	11111	Bob	\$40,000
2	12345	Fred	
3	20202	Martha	
4	22222	Mary	
5	47904	Tom	
6	54321	Sam	\$20,000
7	72401	Joe	\$35,000

©2015 Dr. William T. Verts

Left Outer Join

- Contains all data from the left table.
- The matches with the right table will have their records completely filled in.
- Answer records from the left with no match in the right will have "holes".

©2015 Dr. William T. Verts

Right Outer Join on ID Field

Right Outer Join

	ID	NAME	SALARY
1	11111	Bob	\$40,000
2	13333		\$45,000
3	54321	Sam	\$20,000
4	60233		\$30,000
5	72401	Joe	\$35,000
6	97330		\$50,000

©2015 Dr. William T. Verts

Right Outer Join

- Contains all data from the right table.
- The matches with the left table will have their records completely filled in.
- Answer records from the right with no match in the left will have "holes".

©2015 Dr. William T. Verts

Full Outer Join on ID Field (Union)

Full Outer Join

	ID	NAME	SALARY
1	11111	Bob	\$40,000
2	12345	Fred	
3	13333		\$45,000
4	20202	Martha	
5	22222	Mary	
6	47904	Tom	
7	54321	Sam	\$20,000
8	60233		\$30,000
9	72401	Joe	\$35,000
10	97330		\$50,000

©2015 Dr. William T. Verts

Full Outer Join

- All records from both source tables will be in the answer.
- Those that match will be completely filled in (and appear only once).
- All other records will contain "holes".

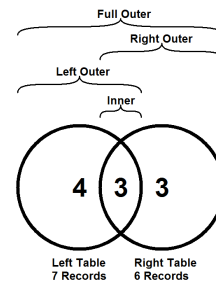
©2015 Dr. William T. Verts

For our example

- There are 7 records in the left table.
- There are 6 records in the right table.
- There are 3 matches (the inner join).
- The left outer join will contain 7 records, 3 filled in and 4 with holes.
- The right outer join will contain 6 records, 3 filled in and 3 with holes.
- The full outer join contains 4 (from left) + 3 (inner) + 3 (from right) = 10 records.

©2015 Dr. William T. Verts

Records in Result of Joins



©2015 Dr. William T. Verts

Relationships

- Many:Many
- 1:Many or Many:1
- 1:1

©2015 Dr. William T. Verts

Many:Many Relationships

- Neither table has an index on the join-field.
- Any record in either table may have multiple matches in the other table.
- For M records in one table and N records in the other, there will be $O(M \times N)$ comparisons.
- It is possible to have $M \times N$ records in the answer (all match), or none (no match).

©2015 Dr. William T. Verts

1:Many or Many:1 Relationships

- One table (M records) is without an index, the other (N records) is indexed on the join-field.
- The indexed table typically has unique information in the indexed field.
- A record in the unindexed table has at most one match in the indexed table.
- A record in the indexed table may have multiple matches in the unindexed table.
- There are $O(M \times \log_2(N))$ comparisons.

©2015 Dr. William T. Verts

1:1 Relationships

- Both tables have an index on the join-field.
- A record in either table has at most one match in the other table.
- Joining the tables requires only one pass each over the two tables.
- There are at most $O(M+N)$ comparisons.

©2015 Dr. William T. Verts

Suppose $M = N = 1024$ Records

- Many:Many
 - $M \times N = 1,048,576$ comparisons
 - Possibly 1,048,576 records in answer
- 1:Many or Many:1
 - $M \times \log_2(N) = 1024 \times 10 = 10,240$ comparisons
- 1:1
 - $M + N = 2048$ comparisons (maximum)

©2015 Dr. William T. Verts

Conclusion

- Indexing speeds up both queries and joins.
- In a join, having one index is faster than having none, having two indexes is even faster.
- For many:many relationships, neither table will have an index so joins will be slow (oh, well), and may generate many more records than are in either source table.

©2015 Dr. William T. Verts