Week 13 Assignment

Justin Hink

1. Join Query:

select \*   
from flights f  
inner join planes p  
on f.tailnum = p.tailnum

Non-indexed query time: 57676 ms (average of 5 runs)  
  
SQL to create index:

CREATE INDEX tailnum\_index\_flights ON flights (tailnum NULLS FIRST);

CREATE INDEX tailnum\_index\_planes ON planes (tailnum NULLS FIRST);

Indexed query time: 46123 ms (average of 5 runs)

We shouldn’t create an index for every possible join/other query as each one provides a certain amount of overhead. This becomes a particular concern when you insert or delete rows in the future. Each insert/delete must add/remove from ever nonclustered index. This takes time!

Also, indexes take up physical disk space. On systems bound by physical volume, this can be a concern for large data systems. Even if you have the room on disk, fitting them in cache (where we can see large performance increases) will be a challenge.

1. Generally speaking, PostgreSQL does not support actual calculated column like many other RDBMSs do. Instead, you need to create functions that can act as these columns within a query.   
   For example (from stack overflow):

CREATE TABLE tbl\_a (a\_id int, col1 int, col2 int);

INSERT INTO tbl\_a VALUES (1,1,1), (2,2,2), (3,3,3), (4,4,4);

CREATE TABLE tbl\_b (b\_id int, a\_id int, colx int);

INSERT INTO tbl\_b VALUES

(1,1,5), (2,1,5), (3,1,1)

,(4,2,8), (5,2,8), (6,2,6)

,(7,3,11), (8,3,11), (9,3,11);

CREATE FUNCTION col3(tbl\_a)

RETURNS int8 AS

$func$

SELECT sum(colx)

FROM tbl\_b b

WHERE b.a\_id = $1.a\_id

$func$ LANGUAGE SQL STABLE;

SELECT a\_id, col1, col2, tbl\_a.col3

FROM tbl\_a;

This same method can be used for any “pre-stored aggregation” column.

You can create similar functionality in a view.  
  
**Advantages of Views:**

* Enhanced Security (permissions on view, not underlying raw data)
* Simplified Queries
* Consistency (A view can present a consistent, unaltered representation of the structure of the underlying DB, even if the source tables are split or modified in some other way).

**Disadvantages of Views:**

* Possible performance degradation
* Update restrictions/complications
  + Query must be translated to transformation of underlying data tables.

1. http://rpubs.com/jhink/45912