V8.3 to V9.2 Upgrade Notes May 17, 2013

AWIPS 2 expects to upgrade postgres from Version 8.3.4 to Version 9.2.4 as part of the V13.4.1 build (deployment to begin Aug 2013). Below is a list of a few of the changes that will occur with this upgrade.

1) New migration tool pg_upgrade

a) utility for upgrading a database in-place for major releases

b) replaces the dump (using pg_dump) and restore operation

- dump/restore can still be done but is much slower

c) preserves table OIDs

d) cannot be used with db versions older than 8.3 !!!

2) Improved "vacuum –full"

a) faster than previous

b) before 9.0, VACUUM FULL required a REINDEX afterwards if you want to keep decent performance. It is no longer required because the new VACUUM FULL doesn't bloat the index anymore.

3) Other vacuum Improvements

a) addition of a "visibility map" will allow the vacuum process to skip pages that do not need vacuuming

b) automatically changes free space map (FSM) configuration

- max_fsm_pages and max_fsm_relations config parameters removed

c) substantial improvement in performance for large tables with few updates

d) skips pages which cannot be locked (reduces the possibility of vacuum getting stuck)

4) Use of ">=" (greater than or equal to) comparison operator is still valid

5) Database level collation and column level collation allowed

a) in 8.3 and previous, collation order (order in which things sort) was set for all databases on the server by initdb; now initdb sets the collation order for template1 which serves as the default for all databases

b) when OHD first switched from Informix to postgres, there were some problems dealing with lists of station identifiers with mixed case characters; problem was caused by the fact that Informix sorted the list differently than postgres due to a difference in the collation order

c) for column level collation use a collate clause as

```
CREATE TABLE  test ( a TEXT COLLATE “es_ES”

…);
```

d) when multiple collations are in force, there are a set of rules to determine which collation to use (See Section 22.2.1)

6) New table type = unlogged

a) lost if db crashes

b) not replicated

c) indexes of table also not logged

d) very fast read/write

e) temp tables are unlogged tables

7) Improved speed of SELECT COUNT (*)

8) Improved sort speed (approx 20%) through use of inline and faster comparison functions

9) PL/pgSQL

a) language installed by default

b) no longer allows certain variable names which match certain SQL commands; these variables need to be double quoted

c) CASE statement added

d) new syntax “EXECUTE command USING expression” (see Section 38.5.4 of V8.4 documentation)

10) PL/Python

a) added support for Python 3

11) SQL Changes

a) now allow statements with “IS NOT NULL” to use indexes

- useful in statements which use MAX() or MIN() on columns containing NULLs

12) DDL Changes

a) new statement “ALTER VIEW OWNER TO …”

- previously used an “ALTER TABLE …” statement for this

b) new statement “ALTER TABLE table_name SET WITH OIDS”

- adds an oid system column to the table

- has no effect if an oid column already exists

c) new statement “ALTER DATABASE SET TABLESPACE …”

- moves a db to a new tablespace

d) new statement “ALTER TABLE … RENAME CONSTRAINT …”

e) GRANT/REVOKE now allows column-level privileges

- example: GRANT SELECT (column1), INSERT (column2) ON salesactivity TO salesteam;

f) CREATE LANGUAGE statement should be replaced by CREATE EXTENSION statement

13) pg_dump

a) removed “-d” option because of confusion over its use

b) new option “--exclude-table-data”

- dump everything, EXCEPT this table's data

- pg_dump will still dump other tables' data

14) Triggers

a) added column trigger (trigger executes when column is updated) and “when” trigger (trigger executes when simple IF-THEN conditions are met)

b) triggers on views added

15) Config parameter standard_conforming_strings now set to ON as default

a) parameter needs to be ON to force postgres to treat a “\” as a literal

b) the statement in set_hydro_env

export PGOPTIONS='-c standard_conforming_strings=on'

was added previously to turn this option on

16) New functions

a) new functions pg_table_size and pg_indexes_size to make gathering size info easier

- pg_table_size(table_name) returns disk space used by table not including indexes

- pg_indexes_size(table_name) returns disk space used by the indexes attached to the specified table

- for example:

select pg_indexes_size('height'::regclass);

17) Descriptor Area

- now available (added in Version 9.0)

- Informix had them

- Descriptor Areas were used by the original db Code Generator

18) Schemas

a) Silently ignore nonexistent schemas specified in search_path