Converting Unsigned Bigints to Signed in PostgreSQL

By Roger Keays, 24 May 2008

Just upgrading Dspam and discovered that the new version uses (or recommends) a BIGINT column for tokens instead of NUMERIC(20) on postgresql for better performance. Dspam tokens are unsigned 64 bit values, but postgresql's BIGINTs are signed 64 bit values. The new dspam just marshals back and forth between signed and unsigned values which avoids the costly NUMERIC data type.

The problem is, of course that all our data is stored as unsigned NUMERIC values. Dspam has a program to convert the database, `dspam_pg2int8`, but it gave me a segmentation fault :(. Anyway, I figured out that it's not too hard to do yourself in SQL and came up with this script, which presumes postgres is using two's complement for negative values:

```
begin;
alter table dspam_token_data rename token to token2;
alter table dspam_token_data add column token bigint;
update dspam_token_data set token =
       case when token2 < (2 ^ 63)
            then token2
            else token2 - (2 ^ 64)
            end;
alter table dspam_token_data drop column token2;
commit;
```

Hope you find it helpful.

About Roger Keays



Roger Keays is an artist, an engineer, and a student of life. He has no fixed addressand has leftfootprints on 40-something different countries around the world. Roger is addicted to surfing. His other interests are music, psychology, languages, the proper use of semicolons, and finding good food.