

**Re: [postgis-users] postgis tuning parameters****From:** Kevin Neufeld <kneufeld@refrations.net>**To:** [PostGIS Users Discussion <postgis-users@postgis.refrations.net>](mailto:postgis-users@postgis.refrations.net)**Date:** 09/27/07 04:40 pm

Hi Dylan,

As a developer on a stand-alone postgresql box, I have the following suggestions:

**shared\_buffers** - This memory, of course as the name implies, is the amount of memory shared between concurrent database connections. Typically this should be about 50-75% of available RAM you have dedicated to postgresql on your box. So, a little math is involved... take your total memory, subtract that needed by the OS, subtract anything else needed by other applications, and use 75% of the remainder. This parameter is set on startup. I've seen web-server boxes with 16GB of RAM configured such that shared\_buffers consume about 12GB. The idea here is simple: disks are slow, memory is fast. The more data that will slowly float up into RAM the better.

**work-mem** - This memory is the amount of RAM used by PostgreSQL when performing sort or hash join operations before being forced to disk. This memory is a per-connection allocation, hence, it is tightly coupled with max\_connections (the number of concurrent connections on your database). The default value is 1MB. In my opinion, this is far, far too low to do anything useful in a database ... in a timely manner. For a web-service database with hundreds of concurrent connections, each using 1MB of ram, I suppose this makes sense. But for me, a developer on a small team hitting a development box with 4GB of ram, I have this set to about 150MB or 200MB. But since this variable can be set at run time, before I run a complex one-time query (and no one else is using the box) I often set this to about 1.4GB using "SET work\_mem TO 1400000;" before running my query.

**maintenance\_work\_mem** - This memory is the amount of RAM used by PostgreSQL when performing maintenance operations like VACUUM, ANALYZE, CREATE INDEX, etc... before being forced to disk. The default value of 32MB is again far too low. I often have this set to about 250MB on a 4GB box with 3 or 4 developers since it is unlikely that every developer will be creating an index all at the same time. Since this variable can also be set at run time, if the box is relatively idle from other developers, I often can create a large index quickly by issuing the command "SET maintenance\_work\_mem to 1400000;".

You will have to strike some sort of balance. shared\_buffers makes

sense with many concurrent connections, but not so much on a single developer box. I would say, for you, set `shared_buffers` to about 50% of available postgres RAM and juggle the rest between `work_mem` and `maintenance_work_mem`. Remember you can always set the last two on the fly!

Also, the others are right, you will have to play with your `SHMMAX` and `SHMMIN` settings to achieve these recommended settings.

Hope this helps,  
Kevin

-----  
Kevin Neufeld  
Software Developer  
Refractions Research Inc.  
300-1207 Douglas St.  
Victoria, B.C., V8W 2E7

Phone: (250) 383-3022  
Email: [kneufeld@refractions.net](mailto:kneufeld@refractions.net)

Dylan Beaudette wrote:

> Hi,  
>  
> I have looked over a couple documents on tuning a postgres server:  
>  
> 1.  
> <http://postgis.refractions.net/pipermail/postgis-users/2006-March/011539.html>  
>  
> 2. [http://www.powerpostgresql.com/Downloads/annotated\\_conf\\_80.html](http://www.powerpostgresql.com/Downloads/annotated_conf_80.html)  
>  
> When performing complex queries, with thousands of records and  
> geometries I am noticing that the postmaster process never exceeds 1%  
> of my system's memory (2Gb). I have tried setting the `shared_buffers`  
> parameter to values > 2000 ... but this only leads to an error when  
> the postgres server is restarted: cannot allocate this much mem, check  
> the `SHMMAX` kernel parameter.  
>  
> Perhaps some answers on are this page:  
> [http://www.redhat.com/docs/manuals/database/RHDB-2.1-Manual/admin\\_user/kernel-resources.html](http://www.redhat.com/docs/manuals/database/RHDB-2.1-Manual/admin_user/kernel-resources.html)  
>  
>  
> or in the minds of people on the list. The machine in question is  
> mostly used for such queries and I would like postgres to be able to  
> better utilize the resources available.

>  
> Cheers,  
>  
> Dylan  
>  
> --  
> Dylan Beaudette  
> Soils and Biogeochemistry Graduate Group  
> University of California at Davis  
> 530.754.7341  
>  
> \_\_\_\_\_  
> postgis-users mailing list  
> [postgis-users@postgis.refractions.net](mailto:postgis-users@postgis.refractions.net)  
> <http://postgis.refractions.net/mailman/listinfo/postgis-users>  
  
\_\_\_\_\_  
postgis-users mailing list  
[postgis-users@postgis.refractions.net](mailto:postgis-users@postgis.refractions.net)  
<http://postgis.refractions.net/mailman/listinfo/postgis-users>