# Installing PiAnoS from SVN (4.2.2 as of today) on Windows 7 x64

17.11.2014

Thibault Genessay

***Note:*** *this is a log of a complete development installation of PiAnoS on a fresh system. This is intended for people having SVN access to PiAnoS. If you have a .tar.gz release (SDK or regular download), you should follow the “normal” installation procedure found at*[*https://ips-labs.unil.ch/pianos/admin/install/index.html*](https://ips-labs.unil.ch/pianos/admin/install/index.html)*.*

Installed Notepad++ and 7-zip as life really hurts without them.

Downloaded PostgreSQL 9.3.5-3 x64 from EnterpriseDB, redirected from <http://www.postgresql.org/download/windows/>  
 Used all default options  
 Set a dummy password for the postgres system user (“postgres”)

Downloaded Apache 2.4.10 x64 from apachelounge.com (as the Apache Software Foundation no longer provides “official” binary downloads): <https://www.apachelounge.com/download/>  
 Since it is a zip file, installed, according to the ReadMe.txt found in the download, in c:\Apache24  
 Installed the Visual C++ Redistributable for Visual Studio 2012 Update 4 (http://www.microsoft.com/en-us/download/details.aspx?id=30679)  
 Ran the service installation step (as administrator): httpd.exe -k install

Downloaded PHP 5.6.3 x64 from <http://windows.php.net/downloads/releases/php-5.6.3-Win32-VC11-x64.zip>  
 Installed in C:\PHP  
 Added the following at the end of c:\Apache24\conf\httpd.conf:  
LoadFile "C:/PHP/libpq.dll"  
PHPIniDir "C:/PHP"  
LoadModule php5\_module "c:/PHP/php5apache2\_4.dll"  
AddType application/x-httpd-php .php  
DirectoryIndex index.php index.html  
 Copied c:\php\ php.ini-development to c:\php\php.ini

Started the Apache service from the Services Management console. Created a file c:\apache24\htdocs\index.php containing a call to phpinfo(), browsed to localhost/index.php and checked the output (which happened to be as expected).

Downloaded and installed TortoiseSVN to make my life easier.

Checked out <https://esc-svn.unil.ch/svn/projects/pianos/v4/trunk/> into c:\pianos\src

Changed the DocumentRoot of Apache to “c:/pianos/src” (adjusted the Directory directive just below as well)

Checked out <https://esc-svn.unil.ch/svn/users/tgenessa/php_classes> into c:\pianos\libs\php\_classes (you should see a directory named “unil” in “php\_classes”, and maybe some others such as “survey”, etc.)

In PHP.ini, uncommented and adjusted the PHP include dir to match:  
include\_path = ".;c:\pianos\libs\php\_classes"

Still in PHP.ini, made the required extensions active (as per https://ips-labs.unil.ch/pianos/admin/install/win7.html):

extension\_dir = "C:\PHP\ext"

extension=php\_curl.dll

extension=php\_fileinfo.dll

extension=php\_gd2.dll

extension=php\_gettext.dll

extension=php\_mbstring.dll

extension=php\_exif.dll

extension=php\_pgsql.dll

Restarted Apache. Checked phpinfo() (which I had to recreate in c:\pianos\src\phpinfo.php because we’ve changed the DocumentRoot). All extensions are active, functional, OK, green lights everywhere.

(Note that the LoadFile directive in httpd.conf serves the purpose of helping you, dear reader, spare 10 hours of googling to understand why f\*cking PHP won’t load the php\_pgsql.dll library otherwise. The Truth is that PHP is broken, and they somehow fiddle with paths loading the system DLLs sometimes, sometimes not. Sometimes it works. Sometimes it doesn’t. Forcing Apache to load the DLL before PHP even sees the light of day makes PHP happy.)

Browsed to localhost/main.php. Boom, saw the big exception with the backtrace. Not surprising, but we can already hear PiAnoS roaring. If you get anything else than a stack trace, you’ve done something wrong. As my grandma says: double-check everything.

Opened C:\Program Files\PostgreSQL\9.3\data\postgresql.conf and set the “standard\_conforming\_strings” setting to off. Restarted the PostgreSQL server. Note that this operation is needed because the PiAnoS code assumes a 8.x server whereas we’ve installed 9.x. This should really be fixed in the code rather than the other way around (global modification to fix a local problem, sigh…)

Opened pgAdmin III from the start menu. Connected to the local instance using the dummy password entered before. Created a new Login Role “pianos”, with “pianos” as password. Created a new database named “pianos”, owned by “pianos”.

Copied the file c:\pianos\src\backend\settings-local.php.template to settings-local.php. Edited the copy, adjusting the database name to “pianos” instead of “pianos4”  
 Application::set('db\_name', 'pianos');

Refreshed the page. The error now changes to something like “the relation language does not exist”. This is quite expected as we haven’t created any schema yet.

Checked out <https://esc-svn.unil.ch/svn/users/tgenessa/microdb> into c:\pianos\libs.

Installed PyGreSQL from c:\pianos\libs\microdb\pygresql. Note: this is a hack. Instructions can be found in the c:\pianos\libs\microdb\readme.txt file.

Downloaded and installed PIL from <http://www.lfd.uci.edu/~gohlke/pythonlibs/#pillow> (<http://www.lfd.uci.edu/~gohlke/pythonlibs/wyxyx8e9/Pillow-2.6.1.win32-py2.7.exe>)

Created the file c:\pianos\src\database\local-database.udb with the following content:

database  
 name pianos  
 host localhost  
 port 5432  
 user pianos  
 password pianos

import pianos4-te.udb

Opened a command shell. From the directory C:\pianos\src\database, ran the commands

C:\pianos\src\database>..\..\libs\microdb\dbadmin.py -s pianos4.udb createbackend  
C:\pianos\src\database>..\..\libs\microdb\dbadmin.py -s pianos4.udb sync

[Got the following output:

No module named dateutil.parser

WARNING: the python-dateutil module is not available. Using my built-in stupid p

arsing function instead

No module named chardet.universaldetector

WARNING: the chardet module is not available. No encoding detection will be perf

ormed (will assume UTF-8 by default)

Checking templates ...

Installing template template.sql

No current schema exist, creating ...

Running post-schema scripts ...

Running script pianos4-functions.sql

Running python module pianos4-install.py

Initializing PiAnoS 4

Registering administrator

Initializing PiAnoS 4 - Teaching Edition

Creating initial PiAnoS 4 dataset

Inserted image ../debug\_prints/print.jpg as ID 5

Inserted image ../debug\_prints/trace.jpg as ID 6

Created default folder as ID 7

Created default exercise as ID 8

Running script pianos4-te-functions.sql]

The warnings are expected.

Refreshed the page in the browser. The login form now appears, although it’s still not OK: we’re missing the CSS and JS dependencies.

Checked out <https://esc-svn.unil.ch/svn/users/tgenessa/cdn> into c:\pianos\cdn. This contains a lot of shared JS+CSS files I reuse across projects, this is why I keep this one out of c:\pianos\libs – although this is quite arbitrary since we’re putting everything under c:\pianos… Well, anyway.

Checked out <https://esc-svn.unil.ch/svn/users/tgenessa/svgcanvas> into c:\pianos\libs\svgcanvas. Made a symlink to the CDN as c:\pianos\cdn\svgcanvas, i.e. as Administrator, open a command prompt and type:  
C:\pianos\cdn>mklink /D svgcanvas c:\pianos\libs\svgcanvas

Made another symlink, while we’re at it:  
C:\pianos\cdn>mklink /D udb C:\pianos\libs\microdb\udb\res\webui\udb

Changed the PiAnoS configuration file to use these brand new files, adding the following:

Application::set('jslib\_jquery', '/cdn/jquery/js/jquery-1.8.2.min.js');

Application::set('jslib\_jquery\_ui', '/cdn/jquery/js/jquery-ui-1.9.1.custom.min.js');

Application::set('jslib\_md5', '/cdn/misc/md5-min.js');

Application::set('jslib\_jquery\_layout', '/cdn/jquery/js/jquery.layout-1.3.0-rc30.74.min.js');

Application::set('jslib\_jquery\_hashchange', '/cdn/jquery/js/jquery.ba-hashchange-1.3.min.js');

Application::set('jslib\_jquery\_cookie', '/cdn/jquery/js/jquery.cookie-1.4.0.js');

Application::set('jslib\_microdb', '/cdn/udb/udb.min.js');

Application::set('jslib\_ajaxupload', '/cdn/misc/ajaxupload.js');

Application::set('jslib\_jquery\_hotkeys', '/cdn/jquery/js/jquery.hotkeys-0.8.min.js');

Application::set('jslib\_jquery\_mousewheel', '/cdn/jquery/js/jquery.mousewheel-3.0.6.js');

Application::set('jslib\_d3', '/cdn/misc/d3.min.js');

Application::set('csslib\_jquery\_layout', '/cdn/jquery/css/layout.css');

Application::set('csslib\_microdb', '/cdn/udb/css/udb.css');

Application::set('csslib\_jquery\_ui', '/cdn/jquery/css/smoothness/jquery-ui-1.9.1.custom.css');

Application::set('microdb\_path', '/cdn/udb/');

Added the following to the c:\Apache24\conf\httpd.conf file:

Alias /cdn c:/pianos/cdn

<Directory c:/pianos/cdn>

Order allow,deny

Allow from all  
 Require all granted

</Directory>

And restarted the Apache service. Note the nasty “Require all granted” that is required from Apache 2.4.x onwards.

Refreshed the PiAnoS login page. Working. Logged in as ‘admin’, password ‘admin’.

Almost there, PHP 5.4.x introduced a deprecation which I am supposedly not depending upon (good news) but printing warnings in the JSON response from the server (bad news). In PHP.ini, uncommented the line:  
always\_populate\_raw\_post\_data = -1  
And restarted Apache.

Refreshed the PiAnoS login page *again*. Logged in.

*To be contined …*