

Work at SourceForge, help us to make it a better place! We have an immediate need for a [Support Technician](#) ([http://www.dice.com/common/seeker/docs/dice\\_jobs.jsp#job53](http://www.dice.com/common/seeker/docs/dice_jobs.jsp#job53)) in our San Francisco or Denver office.

Close ()

[Home \(/\)](#) / [Browse \(/directory/\)](#) / [Projects \(/p/\)](#) / [PAGC-Public Address GeoCoder \(/p/pagc/\)](#) / Code

# PAGC-Public Address GeoCoder (/p/pagc/)

Beta

Brought to you by: [drsampson \(/u/drsampson/\)](#), [putler \(/u/putler/\)](#), [sinclairw \(/u/sinclairw/\)](#)






















[Summary \(/projects/pagc/\)](#) | [Files \(/projects/pagc/files/\)](#) | [Reviews \(/projects/pagc/reviews/\)](#) | [Support \(/projects/pagc/support/\)](#) | [Wiki \(/p/pagc/wiki/\)](#) | [Mailing Lists \(/p/pagc/mailman/\)](#) | [Tickets \(/p/pagc/\\_list/tickets/\)](#) | [Code \(/p/pagc/code/\)](#)

























[Browse](#)  
[Commits](#)  
[\(/p/pagc/code/commit\)](#)

Tree ([360 \(/p/pagc/code/360/\)](#)) | [Branches \(/p/pagc/code/branches/\)](#) | [Tags \(/p/pagc/code/tags/\)](#) | [PostgreSQL \(/p/pagc/code/postgresql/\)](#) | [RSS \(/p/pagc/code/feed/\)](#)

[RO \(#\)](#) [HTTP \(#\)](#) Read Only access [svn checkout svn://svn.code.sf.net/p/pagc/code/trunk pagc-code](#)

File	Date	Author	Commit
 <a href="#">_COPYING (COPYING)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_Makefile (Makefile)</a>	2013-04-24	 woodbri	<a href="#">[r350] Rewrite to change mamory management and API. (D... (/p/pagc/code/350/)</a>
 <a href="#">_README.address_standardizer (README.address_standardizer)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_address_parser.c (address_parser.c)</a>	2013-04-16	 woodbri	<a href="#">[r344] Review code for memory management issues and fi... (/p/pagc/code/344/)</a>
 <a href="#">_address_standardizer-1.0.sql.in (address_standardizer-1.0.sql.in)</a>	2013-05-01	 woodbri	<a href="#">[r359] Added new function to pass argument address ins... (/p/pagc/code/359/)</a>
 <a href="#">_address_standardizer.c (address_standardizer.c)</a>	2013-05-01	 woodbri	<a href="#">[r359] Added new function to pass argument address ins... (/p/pagc/code/359/)</a>
 <a href="#">_address_standardizer.control (address_standardizer.control)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_analyze.c (analyze.c)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_err_param.c (err_param.c)</a>	2013-04-20	 woodbri	<a href="#">[r348] Add target to build a test_main for testing cod... (/p/pagc/code/348/)</a>
 <a href="#">_export.c (export.c)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_gamma.c (gamma.c)</a>	2013-04-17	 woodbri	<a href="#">[r346] Moving DBG def into header file. (/p/pagc/code/346/)</a>
			<a href="#">[r333] Initial checking of</a>

 <a href="#">_gamma.h (gamma.h)</a>	2013-04-09	 woodbri	<a href="#">postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_gazeteer.csv (gazeteer.csv)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_hash.c (hash.c)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_hash.h (hash.h)</a>	2013-04-17	 woodbri	<a href="#">[r346] Moving DBG def into header file. (/p/pagc/code/346/)</a>
 <a href="#">_khash.h (khash.h)</a>	2013-04-16	 woodbri	<a href="#">[r344] Review code for memory management issues and fi... (/p/pagc/code/344/)</a>
 <a href="#">_lexicon.c (lexicon.c)</a>	2013-04-18	 woodbri	<a href="#">[r347] More Debugging changes. (/p/pagc/code/347/)</a>
 <a href="#">_lexicon.csv (lexicon.csv)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_micro-macro.sql (micro-macro.sql)</a>	2013-04-25	 woodbri	<a href="#">[r352] Turn of debug output. added some additional tests. (/p/pagc/code/352/)</a>
 <a href="#">_mk-city-regex.pl (mk-city-regex.pl)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_mk-sql.pl (mk-sql.pl)</a>	2013-07-13	 woodbri	<a href="#">[r360] Fixed regex for PostgreSQL 9.3beta. (/p/pagc/code/360/)</a>
 <a href="#">_mk-st-regex.pl (mk-st-regex.pl)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_pagc-data-psql (pagc-data-psql)</a>	2013-04-12	 woodbri	<a href="#">[r339] Fixed up memory allocation calls and tested on .... (/p/pagc/code/339/)</a>
 <a href="#">_pagc_api.h (pagc_api.h)</a>	2013-04-17	 woodbri	<a href="#">[r346] Moving DBG def into header file. (/p/pagc/code/346/)</a>
 <a href="#">_pagc_common.h (pagc_common.h)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_pagc_std_api.h (pagc_std_api.h)</a>	2013-04-25	 woodbri	<a href="#">[r356] Changes to remove some warning messages when co... (/p/pagc/code/356/)</a>
 <a href="#">_pagc_tools.c (pagc_tools.c)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_pagc_tools.h (pagc_tools.h)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_parseaddress-api.c (parseaddress-api.c)</a>	2013-04-16	 woodbri	<a href="#">[r344] Review code for memory management issues and fi... (/p/pagc/code/344/)</a>
 <a href="#">_parseaddress-api.h (parseaddress-api.h)</a>	2013-04-12	 woodbri	<a href="#">[r338] Converting most of the memory management over t... (/p/pagc/code/338/)</a>

 <a href="#">_parseaddress-regex.h (parseaddress-regex.h)</a>	2013-04-11	 woodbri	<a href="#">[r334] Adding pre-built header files. Adding support f... (/p/pagc/code/334/)</a>
 <a href="#">_parseaddress-stcities.h (parseaddress-stcities.h)</a>	2013-04-11	 woodbri	<a href="#">[r334] Adding pre-built header files. Adding support f... (/p/pagc/code/334/)</a>
 <a href="#">_rules.txt (rules.txt)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_standard.c (standard.c)</a>	2013-04-20	 woodbri	<a href="#">[r349] Fix a windows specific crash. (/p/pagc/code/349/)</a>
 <a href="#">_std_pg_hash.c (std_pg_hash.c)</a>	2013-04-26	 woodbri	<a href="#">[r358] Added "order by id" to queries selecting lexico... (/p/pagc/code/358/)</a>
 <a href="#">_std_pg_hash.h (std_pg_hash.h)</a>	2013-04-24	 woodbri	<a href="#">[r350] Rewrite to change mamory management and API (P... (/p/pagc/code/350/)</a>
 <a href="#">_test.sql (test.sql)</a>	2013-04-25	 woodbri	<a href="#">[r352] Turn of debug output. added some additional tests. (/p/pagc/code/352/)</a>
 <a href="#">_test2.sql (test2.sql)</a>	2013-04-25	 woodbri	<a href="#">[r352] Turn of debug output. added some additional tests. (/p/pagc/code/352/)</a>
 <a href="#">_test_main.c (test_main.c)</a>	2013-04-20	 woodbri	<a href="#">[r348] Add target to build a test_main for testing cod... (/p/pagc/code/348/)</a>
 <a href="#">_tokenize.c (tokenize.c)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_usps-st-city-adds.txt (usps-st-city-adds.txt)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>
 <a href="#">_usps-st-city-orig.txt (usps-st-city-orig.txt)</a>	2013-04-09	 woodbri	<a href="#">[r333] Initial checking of postgresql address_standard... (/p/pagc/code/333/)</a>

## Read Me

This is a fork of the PAGC standardizer and a single line address parser.  
The code is built into a single postgresql extension library.

Portions of this code belong to their respective contributors.  
This code is released under an MIT-X license.

Copyright (c) 2006-2013 Stephen Woodbridge.  
Copyright (c) 2008 Walter Bruce Sinclair

woodbri@swoodbridge.com  
woodbr@imaptools.com

Also read files COPYING

```
-----
Makefile           - PGXS makefile
mk-city-regex.pl   - Perl script to create parseaddress-regex.h
mk-st-regex.pl     - Perl script to create parseaddress-stcities.h
README.address_standardizer - this file
COPYING           - License file
usps-st-city-adds.txt - add local additions of ST<tab>CITY NAME
```

```
usps-st-city-orig.txt    - Steve's extract of USPS city names

mk-city-regex           - created by make
mk-st-regexp            - created by make
usps-st-city-name.txt   - created by make
                        from usps-st-city-orig.txt and usps-st-city-adds.txt
parseaddress-regex.h    - created by make and mk-st-regexp
parseaddress-stcities.h - created by make and mk-city-regex
                        from usps-st-city-name.txt
```

---

#### PREREQUISITES:

- o Postgresql headers and PGXS tools
- o Perl 5 and Perl module Regexp::List which can be install with:  
    sudo perl -MCPAN -e "install Regexp::Assemble"
- o libpcre and headers  
    sudo apt-get install libpcre3-dev libpcre3

---

#### Build and Install:

```
make
sudo make install
```

For postgresql 9.1+ this will install all the files need for CREATE EXTENSION

```
createdb testdb
psql -c "create extension address_standardizer"
```

---

#### How the parser works

The parser works from right to left looking first at the macro elements for postcode, state/province, city, and then looks micro elements to determine if we are dealing with a house number street or intersection or landmark. It currently does not look for a country code or name, but that could be introduced in the future.

#### Country code

Assumed to be US or CA based on:

```
    postcode as US or Canada
    state/province as US or Canada
    else US
```

#### Postcode/zipcode

These are recognized using Perl compatible regular expressions. These regexs are currently in the parseaddress-api.c and are relatively simple to make changes to if needed.

#### State/province

These are recognized using Perl compatible regular expressions. These regexs are currently in the parseaddress-api.c but could get moved into includes in the future for easier maintenance.

#### City name

This part is rather complicated and there are lots of issues around ambiguities as to where to split a series of tokens when a token might belong to either the city or the street name. The current strategy follows something like this:

1. if we have a state, then get the city regex for that state
2. if we can match that to the end of our remaining address string then extract the city name and continue.
3. if we do not have a state or fail to match it then cycle through a series of regex patterns that try to separate the city from the street, stop and extract the city if we match

#### Number street name

1. check for a leading house number, and extract that
2. if there is an '@' then split the string on the '@' into street and street2 else put the rest into street

---

#### Managing the regexes

The regexes are used to recognize US states and Canadian provinces and USPS city names.

#### City regexes

---

usps-st-city-orig.txt - this file contains all the acceptable USPS city names by state. I periodically extract these from the USPS and generate this file. I do NOT recommend editing this file.

usps-st-city-adds.txt - this file you can add new definitions to if you need them. The format of both these files is:

```
<StateAbbrev><tab><CityName>
```

These files are assembled into usps-st-city-name.txt which is compiled by a perl script mk-city-regex.pl into parseaddress-stcities.h which is used to lookup the city regex for a specific state or province.

As I mentioned above is these fail to detect the city, then a secondary strategy is is deployed by cycling through a list of regex patterns. These patterns and regexes are generated by mk-st-regex.pl which creates the parseaddress-regex.h include. This is a perl script so you can view and edit it if that is needed.

I think that there might be some room for improved in the area if coordinating this process with PAGC's lexicon.csv and gazeteer.csv in the future.

---

Author: API: Stephen Woodbridge <woodbri@imaptools.com>  
PAGC: Walter Sinclar

This is a first attempt at extracting the PAGC standardizer code into a separate library. This was done so I could attempt to wrap it into a postgresql stored procedure. (see the directory psq1 for that).

This code is a part of PAGC and is release under an MIT-X license.

#### Assumptions:

Linux  
Sudo is installed and user has access to it.

PAGC does compile under Windows so you might get this to compile into a dll.

#### Build and installation:

Read the Makefile and change as appropriate.

```
make clean
make
make install
make test_main # build the CLI interactive test program
```

Author: Stephen Woodbridge <woodbri@imaptools.com>

These are postgresql wrappers for PAGC standardizer and address parser. These are released un an MIT-X style license.

#### Assumptions:

Linux  
sudo is installed and user has sudo access (see Makefile)  
postgresql 8.3 (make changes in the Makefile to change)

#### Build and Installation:

```
make
make install
```

```
# create a new database using a postgis template
```

```

createdb -T template_postgis -E LATIN1 mydb

# add the stored procedures
psql mydb -f /path/to/standardize_address.sql

Install PAGC lexicon.csv, gazeteer.csv or rules.txt using a perl script.

./pagc-data-psql lex lexicon.csv | psql mydb
./pagc-data-psql gaz gazeteer.csv | psql mydb
./pagc-data-psql rules rules.txt | psql mydb

Now you should be able to test some queries like:

psql mydb

select * from parse_address('2099 university ave w, saint paul, mn, 55104-3431');
select * from parse_address('university ave w @ main st, saint paul, mn, 55104-3431');

select * from parse_address('385 Landgrove Rd Landgrove VT 05148');
-- "385";"Landgrove Rd";"";"385 Landgrove Rd";"Landgrove";"VT";"05148";"";"US"

select * from standardize_address(
    'select seq, word::text, stdword::text, token from gaz union all select seq, word::text,
    stdword::text, token from lex ',
    'select seq, word::text, stdword::text, token from gaz order by id',
    'select * from rules order by id',
    'select 0::int4 as id, ''1071 B Ave''::text as micro, ''Loxley, AL 36551''::text as macro');

select * from standardize_address(
    'select seq, word::text, stdword::text, token from lex order by id',
    'select seq, word::text, stdword::text, token from gaz order by id',
    'select * from rules order by id',
    'select 0::int4 as id, ''116 commonwealth ave apt a''::text as micro, ''west concord,
    ma 01742''::text as macro');

\q

```

#### SourceForge

[About \(/about/\)](#)  
[Site Status \(/blog/category/sitestatus/\)](#)  
[@sfnet\\_ops \(http://twitter.com/sfnet\\_ops\)](#)  
[Powered by Allura \(https://forge-allura.apache.org/p/allura/\)](#)

#### Find and Develop Software

[Create a Project \(/create/\)](#)  
[Software Directory \(/directory/\)](#)  
[Top Downloaded Projects \(/top\)](#)

#### Community

[Blog \(/blog/\)](#)  
[@sourceforge \(http://twitter.com/sourceforge\)](#)  
[Job Board \(/jobs?source=footer\)](#)  
[Resources \(http://library.slashdotmedia.cc\)](#)  
[source=sfnet footer](#)

#### Help

[Site Documentation \(http://p.sf.net/sourceforge/doc\)](#)  
[Support Request \(/support\)](#)  
[Real-Time Support \(http://p.sf.net/sourceforge/irc\)](#)

Copyright © 2014 SourceForge. All Rights Reserved.  
 SourceForge is a Dice Holdings, Inc.  
<http://www.diceholdingsinc.com/phoenix.zhtml?>

[Terms \(http://slashdotmedia.com/terms-of-use\)](http://slashdotmedia.com/terms-of-use)
[Privacy \(http://slashdotmedia.com/privacy-statement/\)](http://slashdotmedia.com/privacy-statement/)
[Opt Out Choices \(http://slashdotmedia.com/opt-out-choices\)](#)
[Advertise \(http://slashdotmedia.com\)](http://slashdotmedia.com)

SourceForge ID: 844444 (sourceforge-inc)