
cjklb Documentation

Release 0.3.2

Christoph Burgmer

August 15, 2016

1	Downloading & Installing	3
1.1	Windows	3
1.2	Unix	3
1.3	Development version	4
1.4	Database	4
2	Command line tools	7
2.1	cjknife — Command Line Interface	7
2.2	installcjkdikt — Install dictionaries	9
2.3	buildcjkdikt — Build database	10
3	Reference	13
4	To do	15
5	Examples	17
6	Copyright & License	19
7	Contact	21
8	Indices and tables	23

Cjklb provides language routines related to Han characters (characters based on Chinese characters named Hanzi, Kanji, Hanja and chu Han respectively) used in writing of the Chinese, the Japanese, infrequently the Korean and formerly the Vietnamese language(s). Functionality is included for character pronunciations, radicals, glyph components, stroke decomposition and variant information.

This document is about version 0.3.2, see <http://cjklb.org/> for the newest and <http://cjklb.org/current> for the current development version. The project is hosted on <http://code.google.com/p/cjklb>. See <http://characterdb.cjklb.org/> for a collaborative effort on gathering language data for cjklb.

Contents:

Downloading & Installing

cjklb has the following dependencies:

- [Python](#) 2.4 or above (currently no support for Python3)
- [SQLite](#) 3+
- [SQLAlchemy](#) 0.4.8+
- [pysqlite2](#) (already ships with Python 2.5 and above)

Alternatively for MySQL as backend:

- [MySQL](#) 5+
- [MySQL-Python](#)

1.1 Windows

Download the .exe installer from the [Python package index](#) and run it.

Three scripts `cjknife.exe`, `buildcjkdb.exe`, and `installcjkdict.exe` will be added to the Python Scripts sub-directory. Make sure this directory is included in your `PATH` environment variable to access these programs from the command line.

CJK dictionaries are not included by default. If you want to install any of those run the following (with an Internet connection):

```
$ installcjkdict CEDICT
```

This will download CEDICT, create a SQLite database file and install it under the directory given by the `APPDATA` environment variable, e.g. `C:\windows\profiles\MY_USER\Application Data\cjklb`. Just substitute CEDICT for any other supported dictionary (i.e. EDICT, CEDICT, HanDeDict, CFDict, CEDICTGR).

1.2 Unix

Get the source package from the [Python package index](#) and deploy the library on your system:

```
$ sudo python setup.py install
```

CJK dictionaries are not included by default. If you want to install any of those run the following (with an Internet connection):

```
$ sudo installckdict CEDICT
```

This will download CEDICT, create a SQLite database file and install it to `/usr/local/share/cklib`. Just substitute CEDICT for any other supported dictionary (i.e. EDICT, CEDICT, HanDeDict, CFDict, CEDICTGR).

1.3 Development version

The development version is available from svn:

```
$ git clone git://github.com/cburgmer/cklib.git
```

You now need to generate the database. Download the UniHan database and call the build CLI (which is not yet installed as executable):

```
$ cd cklib
$ wget ftp://ftp.unicode.org/Public/UNIDATA/UniHan.zip
$ python -m cklib.build.cli build cklibData --attach= \
    --database=sqlite:///cklib/cklib.db
$ sqlite3 cklib/cklib.db "VACUUM"
```

The last step is optional but will help to optimize the database file.

Install by running:

```
$ sudo python setup.py install
```

1.4 Database

Packaged versions of the library will ship with a pre-built SQLite database file. You can however easily rebuild the database yourself.

First download the newest UniHan file:

```
$ wget ftp://ftp.unicode.org/Public/UNIDATA/UniHan.zip
```

Then start the build process:

```
$ sudo buildckdb -r build cklibData
```

1.4.1 SQLite

SQLite by default has no Unicode support for string operations. Optionally the ICU library can be compiled in for handling alphabetic non-ASCII characters. Cklib can register own Unicode functions if ICU support is missing. Queries with `LIKE` will then use function `lower()`. This compatibility mode has negative impact on performance and as it is not needed for dictionaries like EDICT or CEDICT it is disabled by default. See `cklib.conf` for enabling.

1.4.2 MySQL

With MySQL 5 the following `CREATE` command creates a database with `utf8` as character set using the general Unicode collation (MySQL from 5.5.3 on will support full Unicode given character set `utf8mb4` and collation `utf8mb4_bin`):


```
CREATE DATABASE cjklb DEFAULT CHARACTER SET utf8 COLLATE utf8_bin;
```

You might need to set access rights, too (substitute `user_name` and `host_name`):

```
GRANT ALL ON cjklb.* TO 'user_name'@'host_name';
```

Now update the settings in `cjklb.conf`.

MySQL < 5.5 doesn't support full UTF-8, and uses a version with max 3 bytes, so characters outside the Basic Multilingual Plane (BMP) can't be encoded. Building the UniHan database thus might result in warnings, characters above U+FFFF can't be built at all. You need to disable building the full character range by setting `wideBuild` to `False` in `cjklb.conf` before building. Alternatively pass `--wideBuild=False` to `buildcjldb`.

Command line tools

Contents:

2.1 cjkknife — Command Line Interface

cjkknife exposes most functions of the library to the command line.

2.1.1 Examples

Show character information:

```
$ cjkknife -i
Information for character (traditional locale, Unicode domain)
Unicode codepoint: 0x5468 (21608, character form)
Radical index: 30, radical form:
Stroke count: 8
Phonetic data (CantoneseYale): ju
Phonetic data (GR): jou
Phonetic data (Hangul):
Phonetic data (Jyutping): zau1
Phonetic data (MandarinBraille):
Phonetic data (MandarinIPA): tou
Phonetic data (Pinyin): zhu
Phonetic data (ShanghaineseIPA):
Phonetic data (WadeGiles): chou1
Semantic variants:
Glyph 0(*), stroke count: 8

Stroke order: (SP-HZG H-S-H S-HZ-H)
```

Search the EDICT dictionary:

```
$ cjkknife -w EDICT -x "knowledge"
/(n) knowledge/
/(n) knowledge/
/(n) knowledge/
/(n) learning/scholarship/erudition/knowledge/(P)/
/(n) scholarship/learning/knowledge/
/(n) scholarship/knowledge/literary ability/(P)/
/(n) knowledge/information/(P)/
/(n) human intellect/knowledge/
```

```
/(n) human intellect/knowledge/
/(n,vs) expertise/experience/knowledge/
/(n) knowledge/
/(n) knowledge/information/(P)/
/(n,vs) comprehension/knowledge/
/(n) sense/discretion/knowledge/
/(oK) (n) sense/discretion/knowledge/
```

See also:

Screenshots Examples on the project's wiki.

2.1.2 Options

- i** CHAR, **--information=CHAR**
print information about the given char
- a** READING, **--by-reading=READING**
prints a list of characters for the given reading
- r** CHARSTR, **--get-reading=CHARSTR**
prints the reading for a given character string (for characters with multiple readings these are grouped in square brackets; shows the character itself if no reading information available)
- f** CHARSTR, **--convert-form=CHARSTR**
converts the given characters from/to Chinese simplified/traditional form (if ambiguous multiple characters are grouped in brackets)
- q** CHARSTR
performs commands -r and -f in one step
- k** RADICALIDX, **--by-radicalidx=RADICALIDX**
get all characters for a radical given by its index
- p** CHARSTR, **--by-components=CHARSTR**
get all characters that include all the chars contained in the given list as component
- m** READING, **--convert-reading=READING**
converts the given reading from the input reading to the output reading (compatibility needed)
- s** SOURCE, **--source-reading=SOURCE**
set given reading as input reading
- t** TARGET, **--target-reading=TARGET**
set given reading as output reading
- l** LOCALE, **--locale=LOCALE**
set locale, i.e. one character out of TCJKV
- d** DOMAIN, **--domain=DOMAIN**
set character domain, e.g. 'GB2312'
- L, --list-options**
list available options for parameters
- V, --version**
print version number and exit
- h, --help**
display this help and exit

--database=DATABASEURL
database url

-x SEARCHSTR
searches the dictionary (wildcards ‘_’ and ‘%’)

-w DICTIONARY, --set-dictionary=DICTIONARY
set dictionary

2.2 installcjkdikt — Install dictionaries

installcjkdikt downloads and installs a dictionary.

2.2.1 Examples

Download and install CEDICT to \$HOME/cjklib/ (Windows), \$HOME/.cjklib/ (Unix) or \$HOME/Library/Application Support/ (Mac OS X):

```
$ installcjkdikt --local CEDICT
```

Download CFDICT:

```
$ installcjkdikt --download CFDICT
Getting download page http://www.chinaboard.de/cfdict.php?mode=dl... done
Found version 2009-11-30
Downloading http://www.chinaboard.de/cfdict/cfdict-20091130.tar.bz2...
100% |#####| Time: 00:00:00 193.85 B/s
Saved as cfdict-20091130.tar.bz2
```

2.2.2 Options

--version
show program’s version number and exit

-h, --help
show this help message and exit

-f, --forceUpdate
install dictionary even if the version is older or equal

--prefix=PREFIX
installation prefix

--local
install to user directory

--download
download only

--targetName=TARGETNAME
target name of downloaded file (only with --download)

--targetPath=TARGETPATH
target directory of downloaded file (only with --download)

-q, --quiet
don’t print anything on stdout

--database=URL
database url

--attach=URL
attachable databases

--registerUnicode=BOOL
register own Unicode functions if no ICU support available

Global builder options

--collation=VALUE
collation for dictionary entries

--enableFTS3=BOOL
enable SQLite full text search (FTS3)

--useCollation=BOOL
use collations for dictionary entries

2.3 buildckdb — Build database

buildckdb builds the database for the cklib library. Example: `buildckdb build allAvail`.

Builders can be given specific options with format `--BuilderName-option` or `--TableName-option`, e.g. `--Unihan-wideBuild=yes`.

2.3.1 Options

--version
show program's version number and exit

-h, --help
show this help message and exit

-r, --rebuild
build tables even if they already exist

-d, --keepDepending
don't rebuild build-depends tables that are not given

-p BUILDER, --prefer=BUILDER
builder preferred where several provide the same table

-q, --quiet
don't print anything on stdout

--database=URL
database url

--attach=URL
attachable databases

--registerUnicode=BOOL
register own Unicode functions if no ICU support available

--ignoreConfig
ignore settings from cklib.conf

Global builder options

--dataPath=VALUE
path to data files

--entrywise=BOOL
insert entries one at a time (for debugging)

--ignoreMissing=BOOL
ignore missing UniHan column and build empty table

--wideBuild=BOOL
include characters outside the Unicode BMP

--slimUniHanTable=BOOL
limit keys of UniHan table

--collation=VALUE
collation for dictionary entries

--enableFTS3=BOOL
enable SQLite full text search (FTS3)

--filePath=VALUE
file path including file name, overrides searching

--fileType=VALUE
file extension, overrides file type guessing

--useCollation=BOOL
use collations for dictionary entries

Reference

characterlookup
cjknife
build
build.builder
build.cli
dbconnector
dictionary
dictionary.entry
dictionary.format
dictionary.install
dictionary.search
exception
reading
reading.converter
reading.operator
test
test.build
test.characterlookup
test.dictionary
test.readingoperator
test.readingconverter
util

To do

Examples

Get characters by pronunciation (here: “” in Korean):

```
>>> from cjklib import characterlookup
>>> cjk = characterlookup.CharacterLookup('T')
>>> cjk.getCharactersForReading(u'', 'Hangul')
[u'', u'', u'', u'', u'', u'', u'', u'', u'', u'']
```

Get stroke order of characters:

```
>>> cjk.getStrokeOrder(u'')
[u'', u'', u'', u'', u'', u'', u'', u'', u'']
```

Convert pronunciation data (here from *Pinyin* to *IPA*):

```
>>> from cjklib.reading import ReadingFactory
>>> f = ReadingFactory()
>>> f.convert(u'losh', 'Pinyin', 'MandarinIPA')
u'lau.'
```

Access a dictionary (here using Jim Breen’s EDICT):

```
>>> from cjklib.dictionary import EDICT
>>> d = EDICT()
>>> d.getForTranslation('Tokyo')
[EntryTuple(Headword=u'', Reading=u'', Translation=u'/(n) Tokyo (current capital of Japan)/(P)')]
```

Copyright & License

Copyright (C) 2006-2012 cjklb developers

cjklb comes with absolutely no warranty; for details see License.

Parts of the data used by this library have their own copyright:

- Copyright © 1991-2009 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the “Data Files”) or Unicode software and any associated documentation (the “Software”) to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

- Decomposition data Copyright 2009 by Gavin Grover
- Shanghainese pronunciation data Copyright 2010 by Kellen Parker and Allan Simon, <http://www.sinoglot.com/wu/tools/data/>.

The library and all parts are distributed under the terms of the LGPL Version 3, 29 June 2007 (<http://www.gnu.org/licenses/lgpl.html>) if not otherwise noted.

Contact

For help or discussions on cjklb, join cjklb-devel@googlegroups.com.

Please report bugs to the [project's bug tracker](#).

Indices and tables

- `genindex`
- `modindex`
- `search`

Symbols

- attach=URL
 - buildcjkdbs command line option, 10
 - installcjkdbs command line option, 10
- collation=VALUE
 - buildcjkdbs command line option, 11
 - installcjkdbs command line option, 10
- dataPath=VALUE
 - buildcjkdbs command line option, 11
- database=DATABASEURL
 - cjknife command line option, 8
- database=URL
 - buildcjkdbs command line option, 10
 - installcjkdbs command line option, 10
- download
 - installcjkdbs command line option, 9
- enableFTS3=BOOL
 - buildcjkdbs command line option, 11
 - installcjkdbs command line option, 10
- entrywise=BOOL
 - buildcjkdbs command line option, 11
- filePath=VALUE
 - buildcjkdbs command line option, 11
- fileType=VALUE
 - buildcjkdbs command line option, 11
- ignoreConfig
 - buildcjkdbs command line option, 10
- ignoreMissing=BOOL
 - buildcjkdbs command line option, 11
- local
 - installcjkdbs command line option, 9
- prefix=PREFIX
 - installcjkdbs command line option, 9
- registerUnicode=BOOL
 - buildcjkdbs command line option, 10
 - installcjkdbs command line option, 10
- slimUniHanTable=BOOL
 - buildcjkdbs command line option, 11
- targetName=TARGETNAME
 - installcjkdbs command line option, 9
- targetPath=TARGETPATH
 - installcjkdbs command line option, 9
- useCollation=BOOL
 - buildcjkdbs command line option, 11
 - installcjkdbs command line option, 10
- version
 - buildcjkdbs command line option, 10
 - installcjkdbs command line option, 9
- wideBuild=BOOL
 - buildcjkdbs command line option, 11
- L, -list-options
 - cjknife command line option, 8
- V, -version
 - cjknife command line option, 8
- a READING, -by-reading=READING
 - cjknife command line option, 8
- d DOMAIN, -domain=DOMAIN
 - cjknife command line option, 8
- d, -keepDepending
 - buildcjkdbs command line option, 10
- f CHARSTR, -convert-form=CHARSTR
 - cjknife command line option, 8
- f, -forceUpdate
 - installcjkdbs command line option, 9
- h, -help
 - buildcjkdbs command line option, 10
 - cjknife command line option, 8
 - installcjkdbs command line option, 9
- i CHAR, -information=CHAR
 - cjknife command line option, 8
- k RADICALIDX, -by-radicalidx=RADICALIDX
 - cjknife command line option, 8
- l LOCALE, -locale=LOCALE
 - cjknife command line option, 8
- m READING, -convert-reading=READING
 - cjknife command line option, 8
- p BUILDER, -prefer=BUILDER
 - buildcjkdbs command line option, 10
- p CHARSTR, -by-components=CHARSTR
 - cjknife command line option, 8
- q CHARSTR

- ckknife command line option, 8
- q, -quiet
 - builcjkdb command line option, 10
 - installcjkdict command line option, 9
- r CHARSTR, -get-reading=CHARSTR
 - ckknife command line option, 8
- r, -rebuild
 - builcjkdb command line option, 10
- s SOURCE, -source-reading=SOURCE
 - ckknife command line option, 8
- t TARGET, -target-reading=TARGET
 - ckknife command line option, 8
- w DICTIONARY, -set-dictionary=DICTIONARY
 - ckknife command line option, 9
- x SEARCHSTR
 - ckknife command line option, 9

B

builcjkdb command line option

- attach=URL, 10
- collation=VALUE, 11
- dataPath=VALUE, 11
- database=URL, 10
- enableFTS3=BOOL, 11
- entrywise=BOOL, 11
- filePath=VALUE, 11
- fileType=VALUE, 11
- ignoreConfig, 10
- ignoreMissing=BOOL, 11
- registerUnicode=BOOL, 10
- slimUnihanTable=BOOL, 11
- useCollation=BOOL, 11
- version, 10
- wideBuild=BOOL, 11
- d, -keepDepending, 10
- h, -help, 10
- p BUILDER, -prefer=BUILDER, 10
- q, -quiet, 10
- r, -rebuild, 10

C

ckknife command line option

- database=DATABASEURL, 8
- L, -list-options, 8
- V, -version, 8
- a READING, -by-reading=READING, 8
- d DOMAIN, -domain=DOMAIN, 8
- f CHARSTR, -convert-form=CHARSTR, 8
- h, -help, 8
- i CHAR, -information=CHAR, 8
- k RADICALIDX, -by-radicalidx=RADICALIDX, 8
- l LOCALE, -locale=LOCALE, 8
- m READING, -convert-reading=READING, 8

- p CHARSTR, -by-components=CHARSTR, 8
- q CHARSTR, 8
- r CHARSTR, -get-reading=CHARSTR, 8
- s SOURCE, -source-reading=SOURCE, 8
- t TARGET, -target-reading=TARGET, 8
- w DICTIONARY, -set-dictionary=DICTIONARY, 9
- x SEARCHSTR, 9

I

installcjkdict command line option

- attach=URL, 10
- collation=VALUE, 10
- database=URL, 10
- download, 9
- enableFTS3=BOOL, 10
- local, 9
- prefix=PREFIX, 9
- registerUnicode=BOOL, 10
- targetName=TARGETNAME, 9
- targetPath=TARGETPATH, 9
- useCollation=BOOL, 10
- version, 9
- f, -forceUpdate, 9
- h, -help, 9
- q, -quiet, 9