PyCDDB

Roland Schäuble

PyCDDB

by Roland Schäuble

Table of Contents

Chapter 1. What is PyCDDB?

PyCDDB is a Python-Module (http://www.python.org) to access a CDDB-server to get information for audio compact discs like:

- Artist
- Disc title
- Track titles

and other information for digital audio compact discs.

PyCDDB requires a disc-ID, which is generated from the track starting positions. For this purpose, **discid** (http://discid.sourceforge.net can be used.

Chapter 2. PyCDDB

PyCDDB is a module, written in Python, to handle communication to the CDDB-server

2.1. Supported Platforms

The Python-Module should run on all platforms, where communication through http-connections is supported from **urllib** module.

Communication to the CDDB-server takes place via a http-connection. For this reason, the computer must be connected to the internet in some way, when making queries. Default CDDB-server is http://freedb.freedb.org but the name of the used server can be chosen by the client application.

2.2. Installation

There are several ways to install **PyCDDB** on your machine:

- Use the RPM-Package for RPM-based systems.
- Run (as a priviledged user) the Python-Setup script **setup.py** from the source-distribution with **install** parameter: **$ python setup.py install**
- Copy the file `PyCDDB.py` to the apropriate Python-Libarary directory of your Python installation. This is for example `/usr/lib/python-2.2` on a Unix-like platform

2.3. Usage

An example of how to use the PyCDDB-Module is given in the distribution (see file `TestPyCDDB.py`).

In general, 5 steps are required, to get information about a compact disc:

1. Get the disc-ID for the compact disc, you want to have the information about (line 1). You can use the discid-Program, which is available at http://discid.sourceforge.net or some other program, which generates the disc-ID in the appropriate form. The required format of the disc-ID for **PyCDDB** is as follows:

   ```
   8HexDigitID NumberOfTracks Track1StartFrame Track2StartFrame..TrackNStartFrame DiscLengt
   ```

   If you want to use another program to generate the disc-ID for **PyCDDB**, it has to use the same format. Another well known program to calculate the disc-ID is **cd-discid http://lly.org/~rcw/cd-discid/**

2. Create an instance of **PyCDDB** (line 3)
3. Send a query to the CDDB-server, using the disc-ID (line 4)
4. Check, if the disc is known by the CDDB-server. For some discs, more than one entry exist in the CDDB-Database. in this case, you have to choose, which one to read (lines 5...12)
5. Get the information about the specified disc from CDDB-server (line 14).
6. Use information for whatever you want to (lines 16..20)

```
 1: discid = ... # get disc-ID from somewhere
 2:
 3: db = PyCDDB.PyCDDB()
 4: items = db.query(discid)
 5: if len(items) > 0: # Items found?
 6:     if (len(items) > 1): # Multiple matches
 7:         print "Multiple matches found. Choose one of:"
 8:         for item in range(len(items)):
 9:             print "%d : %s %s" % (item, items[item]['category'], items[item]['title'])
10:         index = input("which item?")
11:     else: # single match
12:         index = 0
13:
14:     info = db.read(items[index])
15:     if len(info['TTITLE']) > 0: # read info
16:         print 40 * '-'
17:         print "Title: %s" % info['DTITLE']
18:         for track in range(len(info['TTITLE'])):
19:             print "Track: %02d %s" % (track, info['TTITLE'][track])
16:         print 40 * '-'
21:     else:
22:         print >> sys.stderr, "Read-Status %d: '%s.'" % (db.status(), db.message())
23: else:
24:     print >> sys.stderr, "Query-Status %d: '%s.'" % (db.status(), db.message())
```

2.3.1. Data structure of query-result

PyCDDB.query() returns an array of found matces. The array is empty, if no matches for the given disc-ID are found by the CDDB-server. Each match is a dictionary with the following entries:

- **'category':** Name of category in CDDB
- **'disc_id':** 8 hexdigit disc-ID
- **'title':** Disc title

A single item of read-output can be used, to feed **query**

```
[ { 'category': 'rock',
     'disc_id': 'e512640f',
     'title': 'Caf\xe9 Del Mar Vol.5 / Caf\xe9 Del Mar Vol.5'
  },
  { 'category': 'misc',
    'disc_id': 'e512640f',
    'title': 'Caf\xe9 Del Mar / Volumen Cinco'
  },
  { 'category': 'data',
    'disc_id': 'e512640f',
    'title': 'Various / Caf\xe9 del Mar - Volumen Cinco'
  }
]
```

2.3.2. Data structure of read-result

PyCDDB.query() returns a dictionary, consisting of the following items:

- **'DTITLE':** Disc title
- **'TTITLE':** Array of track titles
- **'EXTD':** Extra disc info
- **'EXTT':** Array of extra track infos
- **'DISCID':** Disc-ID
- **'PLAYORDER':** normally empty for non-local CDDBs
- **'DYEAR':** Year of release
- **'DGENGRE':** Genre-info for disc

```
{ 'TTITLE': ['Mumbai theme tune',
             'More than ever people',
             'Appreciation',
             :
             ...more track titles here...
             :
             'Close Cover'],
  'EXTD': 'Compiled with love by Jose Padilla \\n YEAR: 1998',
  'EXTT': ['', '', '', '', '', '', '', '', '', '', '', '', '', '', ''],
  'DISCID': 'e512640f',
  'PLAYORDER': '',
  'DTITLE': 'Caf\xe9 Del Mar Vol.5 / Caf\xe9 Del Mar Vol.5',
  'DYEAR': '1998',
  'DGENRE': 'Ambient'
}
```

Chapter 3. Download

PyCDDB is hosted at http://sourceforge.net

PyCDDB homepage is http://pycddb.sourceforge.net

The latest version of **PyCDDB** can always be found on http://www.sourceforge.net/projects/pycddb

The subversion repository can be browsed under http://pycddb.svn.sourceforge.net

The authors homepage can be found on http://homepage.sunrise.ch/mysunrise/rschaeuble/