
CBMPy Documentation

Release 0.7

Brett G. Olivier

November 19, 2014

| | | |
|-----------|---|-----------|
| 1 | CBMPy: Installation guide | 3 |
| 1.1 | Support | 3 |
| 1.2 | Python standard library modules | 3 |
| 1.3 | Required libraries (Python bindings) | 3 |
| 1.4 | Installation types: quick reference | 3 |
| 1.5 | Generic installation on Microsoft Windows (XP, 7, 2008) | 4 |
| 1.6 | Linux: Ubuntu | 7 |
| 1.7 | Linux: Ubuntu 14.04 | 8 |
| 1.8 | Apple Macintosh: OS X | 10 |
| 1.9 | Installing PySCeS-CBM Mariner (Microsoft Windows and Linux) | 11 |
| 2 | Introduction | 13 |
| 3 | CBMPy: CBCommon module | 15 |
| 4 | CBMPy: CBConfig module | 17 |
| 5 | CBMPy: CBCPLEX module | 19 |
| 6 | CBMPy: CBDDataStruct module | 27 |
| 7 | CBMPy: CBGLPK module | 31 |
| 8 | CBMPy: CBGUI module | 37 |
| 9 | CBMPy: CBModel module | 39 |
| 10 | CBMPy: CBModelTools module | 55 |
| 11 | CBMPy: CBMultiCore module | 57 |
| 12 | CBMPy: CBMultiEnv module | 59 |
| 13 | CBMPy: CBNetDB module | 61 |
| 14 | CBMPy: CBPlot module | 65 |
| 15 | CBMPy: CBQt4 module | 67 |
| 16 | CBMPy: CBRead module | 69 |
| 17 | CBMPy: CBReadtxt module | 71 |
| 18 | CBMPy: CBSolver module | 73 |
| 19 | CBMPy: CBTools module | 75 |
| 20 | CBMPy: CBWrite module | 81 |

CBMPY: INSTALLATION GUIDE

1.1 Support

CBMPy is Open Source software released under the GNU GPL 3 licence (included with the source code) and is in constant development. All the latest downloads, documentation and development information is available at **CBMPy on SourceForge**: <http://cbmpy.sourceforge.net>.

1.2 Python standard library modules

CBMPy is developed and tested against Python 2.7.x. The following Python Standard Library modules are used in CBMPy and should be available as part of any CPython distribution and not require additional installation:

```
'cPickle', 'cStringIO', 'cgi', 'copy', 'gc', 'itertools', 'locale', 'math',  
'multiprocessing', 'os', 'pprint', 'random', 're', 'shutil', 'subprocess',  
'time', 'urllib2', 'webbrowser', 'xml'
```

1.3 Required libraries (Python bindings)

Besides those mentioned above, the following packages are required for CBMPy's core functionality. Note that it is possible to install CBMPy using only *numpy* but that only very limited subset of functionality is then available. CBMPy is primarily developed on Microsoft Windows and Ubuntu Linux and where possible the package name is provided such that can be used with the software center or package manager `sudo apt-get install <package>` (please see the man pages for `sudo` and `apt-get` if you don't know what this). A comprehensive list of modules are listed at the end of this document however I consider these to be the minimum requirements. In the case of external C/C++ libraries the Python bindings should be installed as well (e.g. *libSBML*). Many of these are available in *batteries included* Python distributions.

1.4 Installation types: quick reference

1.4.1 Minimal

- **numpy** <http://numpy.scipy.org>
- **libsbml** (+ Python bindings) <http://sbml.org/download>
- **PyQT4** <http://www.riverbankcomputing.com/software/pyqt/download>
- Optimization libraries (one or more of): - CPLEX (LP, MILP): <http://www.ibm.com> - GLPK (LP): <http://tfinley.net/software/pyglpk/>

1.4.2 Full (highly recommended)

- **xlwt** <http://pypi.python.org/pypi/xlwt>
- **xlrd** <http://pypi.python.org/pypi/xlrd>
- wxPython
- Matplotlib
- Sympy

1.4.3 Complete

Web services and database:

- pysqlite2
- suds

Advanced functionality:

- SciPy
- H5Py
- NetworkX

User tools:

- iPython
- iPython-notebook
- SCiTE

1.5 Generic installation on Microsoft Windows (XP, 7, 2008)

For the modeller that does not want to customize his installation and install all of the above packages by hand there are some *batteries included* Python distributions which have many (if not most) of the packages listed above. An Open Source distributions is *Python(x,y)* available from <http://code.google.com/p/pythonxy> Alternatives include commercial distribution such as Anaconda <http://continuum.io> and the Enthought Python Distribution (EPD) <http://www.enthought.com>

Python(x,y) has a huge number of additional packages in addition to the base Python distribution, best of all it is Open Source and free for use. First of all download Python(x,y), I would recommend the latest *Python 2.7.x* distribution. In addition to the default packages automatically selected by the installer it is highly recommended to install either all the additional packages. If not at least select the following packages from the *Python* branch of the Python(x,y) installation directory:

- WxPython
- Sympy
- NetworkX
- xlrd
- xlwt
- h5py
- wxPython
- PyQT4

You should now have a working Python 2.7.x distribution. Try firing up an advanced shell like *iPython* and play around and get to grips with the fantastic, free text editor *SciTE*.

1.5.1 Installing CBMPy

There are two ways to install CBMPy either download the latest release as source bundle or binary from <http://cbmpy.sourceforge.net> and unzip or execute from a temporary directory (recommended). Or, if you want the latest (greatest and potentially broken) version grab the latest revision from the the CBMPy Subversion repository:

```
svn co http://sourceforge.net/p/cbmpy/code/HEAD/tree/trunk/cbmpy cbmpy
```

In both cases you should now have a directory that contains a file *setup.py* which can install by simply typing the following into a Windows shell (command line):

```
python setup.py build
python setup.py install
```

1.5.2 Installing libSBML with Python bindings

It is highly recommended to install libSBML which CBMPy uses to provide support for the Systems Biology Markup Language (SBML). First go to the libSBML download page <http://sbml.org/Software/libSBML> page follow the *Download libSBML* → *Stable* → *Windows* → *32bit* path and download libSBML (e.g. libSBML-5.10.0-win-x86.exe). The latest stable version can be found at <http://sbml.org/Software/libSBML>

<http://sourceforge.net/projects/sbml/files/libsbml/5.10.0/stable/Windows/32-bit/libSBML-5.10.0-win-x86.exe/download>

Run the installer and make sure you select the Python Bindings during installation or install the appropriate Python bindings that match your Python(x,y) version directly e.g. (libSBML-5.10.0-win-py2.7-x86.exe)

1.5.3 Optimization (1): IBM cplex optimization studio (Academic)

If you have access to the the IBM CPLEX solver. It is a a good idea to use the latest available version. Again choose the appropriate 32 or 64 bit version and an installation path that suites your setup.

- Run **cplex_studio126.win-x86-32.exe**
- Select English language and accept licence
- Set “Program” install directory to C:\ILOG\CPLEX_Studio126
- Allow default associations to be set and PATH update

Once installation is complete we need to install the Python bindings

- Open a terminal
- Execute `cd c:\ILOG\CPLEX_Studio126\cplex\python\x86_win32`
- Execute `python setup.py install`

1.5.4 Optimization (2): GLPK

CBMPy 0.7.0 includes support for the free, Open Source GLPK solver. This allows access to CBMPy’s LP functionality (MILP’s requires CPLEX). A port of PyGLPK 0.3 is maintained by the OpenCOBRA project which is mirrored here:

<https://sourceforge.net/projects/cbmpy/files/tools/glpk/>

Select the binary or source distribution you require and either execute the binary:

- Execute `glpk-0.3.win32-py2.7.exe`

1.5.5 Testing your new installation

If everything has gone according to plan you can test your installation:

- Open a terminal
- Execute `ipython`
- In `ipython` shell, execute `import numpy, h5py, xlrd, xlwt`

No import errors should occur.

- Execute `import libsbml`
- Execute `libsbml.LIBSBML_VERSION_STRING`

A successful test should return (for example):

```
In : libsbml.LIBSBML_VERSION_STRING
Out: '51000'
```

- Execute `import pyscscbm as cbm`

This should return:

```
In [1]: import pyscscbm as cbm

*****
Welcome to the PySCeS Constraint Based Modelling Toolkit (0.7.0 [r1147])
*****

GLPK not available

*****
Using CPLEX
*****

***
Multiple Environment Module (0.6.2 [r1147])
***
```

Exit `ipython` with CTRL-D

If you installed CPLEX then try:

- Open a terminal
- Execute `ipython`
- Execute `import cplex`
- Execute `lp = cplex.Cplex()`
- Execute `lp.solve()`

A succesful test should return:

```
In : lp.solve()
Tried aggregator 1 time.
No LP presolve or aggregator reductions.
Presolve time = 0.00 sec.
```

Exit `ipython` with CTRL-D

If you installed GLPK then try:

- Open a terminal
- Execute `ipython`
- Execute `import glpk`

- Execute `lp = glpk.LPX()`

A succesful test should return:

```
In : glpk.LPX()
<glpk.LPX 0-by-0 at 0x036C24C8>
```

Exit ipython with CTRL-D

1.5.6 Install CBMPy (<http://cbmpy.sourceforge.net>)

Download the latest version of CBMPy

- Run **pyscescbm-0.7.0.win32.exe** (or newer)

Test installation:

- Open a terminal
- Execute `ipython`
- Execute `import pyscescbm as cbm`

This should return:

```
In [1]: import pyscescbm as cbm

*****
Welcome to the PySCeS Constraint Based Modelling Toolkit (0.6.2 [r1147])
*****

GLPK not available

*****
Using CPLEX
*****

***
Multiple Environment Module (0.6.2 [r1147])
***
```

Exit ipython with CTRL-D

1.6 Linux: Ubuntu

On Linux many of the base dependencies are available as packages or from the Python Cheeseshop (<http://pypi.python.org/pypi>). For **libSBML**, **CPLEX** and/or **GLPK** please see the *Generic installation on Microsoft Windows (XP, 7, 2008)* for more details. For example using **Ubuntu** the base dependencies can be easily installed (depending on what functionality is required). If you don't know what these packages are please look them up before installing.

Required:

```
sudo apt-get install python-dev python-numpy
```

- **libSBML** for SBML support. Please see <http://sbml.org/Software/libSBML> or try the following. Depending on your configuration

you need to install `libxml2`, `bzip2` and their associated “dev” packages:

```
apt-get install libxml2 libxml2-dev
apt-get install zlib1g zlib1g-dev
apt-get install bzip2 libbz2-dev
```



```
easy_install pip
pip install python-libsbnl
```

- Optimization (at least one of):
 - IBM CPLEX: <http://www.ibm.com>
 - PyGLPK: <https://sourceforge.net/projects/cbmpy/files/tools/glpk/>

Please note that due to changes in the GLPK API the current version of PyGLPK (0.3) **only supports GLPK up until version 4.47**. If your system has a newer version of GLPK then the current workaround is to uninstall the newer version and compile 4.47 from source (also available from the above directory). Dependencies are standard Linux build tools and GMP etc:

```
tar xzf glpk-4.47.tar.gz
cd glpk-4.47
./configure --with-gmp
make
make check
sudo make install
```

Graphical interfaces (highly recommended):

```
sudo apt-get install python-wxgtk2.8 python-qt4 python-matplotlib
```

Extended IO (highly recommended):

```
sudo apt-get install python-xlrd python-xlwt python-sympy
```

Web services and database:

```
sudo apt-get install python-suds python-pysqlite2
```

Advanced functionality:

```
sudo apt-get install python-scipy python-h5py python-networkx
```

User tools (highly recommended):

```
sudo apt-get install ipython ipython-notebook scite
```

1.7 Linux: Ubuntu 14.04

1.7.1 Python2

First we create a scientific Python workbench:

```
sudo apt-get install python-dev python-numpy python-scipy python-matplotlib python-pip
sudo apt-get install python-sympy python-suds python-xlrd python-xlwt python-h5py
sudo apt-get install python-wxgtk2.8 python-qt4
sudo apt-get install ipython ipython-notebook
```

1.7.2 libSBML

Installing libSBML is now easy using Pip:

```
sudo apt-get install libxml2 libxml2-dev
sudo apt-get install zlib1g zlib1g-dev
sudo apt-get install bzip2 libbz2-dev

sudo pip install python-libsbnl
```

1.7.3 glpk/python-glpk

GLPK needs to be version 4.47 to work with glpk-0.3:

```
sudo apt-get install libgmp-dev
```

cd GLPK source (e.g. glpk-4.47):

```
./configure --with-gmp
make
make check
sudo make install
sudo ldconfig
```

cd to python-glpk source (glpk-0.3):

```
make
sudo make install
```

1.7.4 CBMPy

Finally, install CBMPy:

```
python setup.py build sdist
sudo python setup.py install
```

1.7.5 Installing PyscesMarinerCBM

This will install PySCeS Mariner that adds SOAP web-services capability to CBMPy. First unpack pyscesmariner-0.7.7.zip and install the cherrypy webserver:

```
sudo apt-get install python-cherrypy
```

1.7.6 Install soaplib

cd <pysces_cbm_mariner>/misc:

```
tar -xf soaplib-0.8.1.tar.gz
cd soaplib-0.8.1
python setup.py build sdist
sudo python setup.py install
```

1.7.7 Install Mariner

cd <pysces_cbm_mariner> and set mariner configuration (not needed for Ubuntu, Windows or if the server does not read SBML):

```
sudo nano /usr/local/lib/python2.7/dist-packages/pyscesmariner/MarinerConfig.py
PATH_LIBSBMLTHREAD = '/usr/local/lib/python2.7/dist-packages/pyscesmariner/libSBMLthread.pyc'
PATH_LIBSBML_CONVERTTHREAD = '/usr/local/lib/python2.7/dist-packages/pyscesmariner/libSBMLConvert'
```

cd to <pysces_cbm_mariner>:

```
python setup.py build sdist
sudo python setup.py install
```

1.7.8 Test installation

Open a new terminal window:

```
# cd <pysces_cbm_mariner>/demo
python cbm_server_demo.py
```

Open another terminal and run the client demo:

```
python cbm_client_demo.py
```

Kill the server by closing the terminal window.

1.7.9 Python3

Not all dependencies are available for Python3:

```
sudo apt-get install python3-dev python3-numpy python3-scipy python3-matplotlib python3-pip
sudo apt-get install python3-xlrd python3-h5py
```

```
# need to find out what is going on with Python3 and xlwt suds
# easy_install3 sympy ???
# wxPython and PyQt4 not in Ubuntu P3 builds yet
```

```
sudo apt-get install ipython3 ipython3-notebook
```

```
sudo apt-get install libxml2 libxml2-dev
sudo apt-get install zlib1g zlib1g-dev
sudo apt-get install bzip2 libbz2-dev
```

```
sudo pip3 install python-libsmb
```

```
sudo apt-get install python-qt4 python-qt4-dev python-sip python-sip-dev build-essential
```

1.8 Apple Macintosh: OS X

Installation is similar to Linux except packages are installed using distutils and pip. The first step is to install the Mac development tools xcode

Install Python packages:

```
sudo easy_install numpy ipython scipy matplotlib
sudo easy_install xlrd xlwt sympy suds pyparsing pip
```

Use pip to install advanced Ipython and libsmb:

```
sudo pip install ipython[notebook]
ARCHFLAGS=-Wno-error=unused-command-line-argument-hard-error-in-future pip install python-libsmb
```

For solvers, either install your own copy of CPLEX or build PyGLPK which requires building both the GMP and GLPK libraries.

GMP (<https://gmplib.org/>):

```
download gmp
./configure --prefix=/usr/local
make
make check
sudo make install
```

GLPK (<http://sourceforge.net/projects/cbmpy/files/tools/glpk>):

```
download glpk-4.47.tar.gz
./configure --prefix=/usr/local --with-gmp
make
sudo make install
```

PyGLPK (<http://sourceforge.net/projects/cbmpy/files/tools/glpk/>):

```
download python-glpk-0.3
python setup.py build
sudo python setup.py install
```

1.9 Installing PySCeS-CBM Mariner (Microsoft Windows and Linux)

The PySCeS Mariner module exposes the CBMPy functionality as SOAP web services (e.g. as a backend to FAME (<http://F-A-M-E.org>)). It is available for download from SourceForge:

- PySCeS-CBM Mariner: http://sourceforge.net/projects/cbmpy/files/release/pysces_mariner/

1.9.1 Dependencies: CherryPy, libXML and SOAPlib

PySCeS-CBM Mariner requires (pure python) soaplib 0.8.1 (supplied with it) or downloadable from:

<https://sourceforge.net/projects/cbmpy/files/tools/soaplib/>

Soaplib itself has two dependencies which should be installed first:

- LXML (<http://lxml.de>)
 - Windows: install with `easy_install lxml`
 - Linux (Ubuntu) use `sudo apt-get install python-lxml`
- CherryPy (<http://www.cherrypy.org>)
 - Windows: install with `easy_install cherrypy`
 - Linux (Ubuntu) use `sudo apt-get install python-cherrypy`
- SOAPLIB 0.8.1:
 - Windows: Execute `soaplib-0.8.1.win32.exe`
 - Linux: Unpack the zip archive and run `sudo python setup.py install`

Test installation:

- Open a terminal
- Execute “ipython”
- Execute “import cherrypy, lxml, soaplib” no errors or warnings should be generated
- Exit ipython with CTRL-D
- change directory to supplied soaplib tests e.g. “cd e:\cbmpy\tests\soaplib”
- Execute “python binary_test.py”
- Execute “python primitive_test.py”

All tests should pass.

1.9.2 PySCeS-CBM Mariner (<http://cbmpy.sourceforge.net>)

Download and install the latest version (0.7.4 or newer is required for CBMPy 0.7+):

- Windows: Execute `pyscesmariner-0.7.7.zip`
- Linux: unpack the archive and run `sudo python setup.py install`

To test installation, on Linux execute the commands in `run_server.bat` from the terminal directly.

- Open two terminals and in both
- Change directory to supplied PySCeS-CBM Mariner tests e.g. `cd e:\\cbmpy\\tests\\pyscesmariner`
- In terminal one Execute `run_server.bat`

Which should now display:

```
E:\\cbmpy\\tests\\pyscesmariner>python cbm_server_demo.py
Mariner using E:\\cbmpy\\tests\\pyscesmariner as a working directory
Mariner server name: 10.0.2.15
Mariner using port: 31313
```

```
Welcome to the PySCeS Constraint Based Modelling Toolkit (0.7.0)
```

```
<snipped>
```

```
Multiple Environment Module (0.6.2 [r1147])
```

```
PySCeSCBM/Mariner initialising ... this console is now blocked
```

In terminal two:

- Execute `python cbm_client_demo.py`

This should end without errors and display `done`. Congratulations you have successfully installed CBMPy and PySCeS-CBM Mariner!