

LetItB Installation Manual

Okinawa Institute of Science and Technology
Neural Computation Unit

Version 1 Dated April 21, 2008

Contents

1. Introduction.....	3
2. Installation Instructions for Linux and Mac OS.....	4
3. Xerces-C Library Installation	5
4. Graph Visualization Library Installation.....	6
5. Gnu Scientific Library Installation.....	6
6. Library for Numeric Calculation: BLAS/LAPACK Installation	7
7. SBML Library Installation.....	8
8. Sundials Library Installation	9
9. SBML ODE Solver Library Installation	10
10. Qt Installation	12
11. Qwt Installation	13
12. LetItB Installation	14

1. Introduction

This manual describes how to install the source codes or source programs of the LetItB software and its external libraries required to use the LetItB software.

You can install these source codes or source programs one by one in the step sequence shown in the contents of this manual.

- List of Required Libraries (* Please install the libraries one by one in the following step sequence)
 - Xerces-C Library (xerces-c-2.7.0 or later)
 - Graph Visualization Library (graphviz-2.16.1 or later)
 - Gnu Scientific Library (gsl-1.9 or later)
 - Library for numeric calculation: BLAS/LAPACK (mkl 8.1 or later)
 - Systems Biology Markup Language Library (libsbml-2.3.4)
 - SUite of Nonlinear and Differential/ALgebraic equation Solvers (sundials-2.1.1)
 - SBML ODE Solver Library (SBML-odeSolver-1.6.0)
 - Qt (qt-4.3.1 or later)
 - QWT (qwt-5.0.2 or later)

- LetItB Source Programs
 - LetItB_ver_alpha.tar.gz

[Recommended Environment]

* The following describes the operating environment for the current version of the LetItB software:

OS: Linux OS (32bit/64bit) or Mac OS X

CPU: Intel Pentium 1.30 GHz or higher

Memory: 1GB or more

Free Disk Space: 100MB or more

Resolution: SXGA(1280x1024) or higher

[Contact]

If you have any questions on the LetItB software, please contact the following e-mail address:

LetItB Support, Neural Computation Unit, Okinawa Institute of Science and Technology

e-mail: nc-admin@oist.jp

2. Installation Instructions for Linux and Mac OS

[Common Prerequisite for Both Environments]

Before LetItB installation, prepare an environment where you can compile such programs as gcc, g++ and make.

[Installation Instructions for Linux OS]

To install the LetItB software and required libraries, perform in sequence the install steps as shown from Section 3 “Xerces-C Library Installation” to Section 12 “LetItB Installation” in this manual.

[Installation Instructions for Mac OS X]

① Pre-install Step

Before LetItB installation, install X11 and Xcode on the target machine by using the Mac OS X installer CD.

② Install MacPorts

a. Install the MacPorts corresponding to your OS version by accessing the following URL:

URL: <http://svn.macports.org/repository/macports/downloads/MacPorts-1.5.0>

(Example: If you use Mac OS 10.4, download MacPorts-1.5.0-10.4.dmg)

b. Edit ~/.profile and ~/.bashrc, and define a path to the console.

Define the path as shown in the following URL:

URL: <http://guide.macports.org/>

c. Update the MacPorts package information.

Execute the following command:

```
# sudo port selfupdate
```

③ Install Required Libraries

Install the required libraries (Xerces-C, Expat, graphviz, grace) via MacPorts from the terminal or xterm.

More specifically, input the following commands from the terminal:

```
# sudo port install xercesc
```

```
# sudo port install expat
```

```
# sudo port install graphviz
```

```
# sudo port install grace
```

```
# sudo port install gsl
```

* Define the following environment variables whenever necessary:

```
export DYLD_FALLBACK_LIBRARY_PATH=/opt/local/lib:/opt/local/lib/graphviz
:$DYLD_FALLBACK_LIBRARY_PATH
```

- ④ To install the LetItB software and required libraries, perform in sequence the install steps as shown from Section 7 “SBML Library Installation” to Section 12 “LetItB Installation” in this manual.

3. Xerces-C Library Installation

* If you use Mac OS X, see Section 2.

- ① Get the source from the developer download site: <http://xml.apache.org/xerces-c/download.cgi>

- ② Extract the following compressed files on a working directory:

```
#cp xerces-c-current.tar.gz /opt
#tar zxvf ./xerces-c-current.tar.gz
```

- ③ Rename the following directory to an appropriate name:

```
#mv xerces-c-src_2_7_0 xersec-c-2.7.0
```

- ④ Define the environment variables:

```
# export XERCESCROOT=/opt/xerces-c-2.7.0
```

- ⑤ Install (run as root)

```
# cd /opt/xerces-c-2.7.0/src/xercesc
(Specify the following in one line)
# ./runConfigure -plinux -cgcc -xg++ -minmem -nsocket -tnative -rpthread
# gmake
# gmake install
```

[Additional] Build a sample (whenever necessary)

```
# cd /opt/xersec-c-2.7.0/samples
# ./runConfigure -plinux -cgcc -xg++
# gmake
```

* Define the following environment variables whenever necessary:

```
export XERCESCROOT=/opt/xerces-c-2.7.0
export LD_LIBRARY_PATH=/opt/xersec-c-2.7.0/lib:$ LD_LIBRARY_PATH
```

4. Graph Visualization Library Installation

* If you use Mac OS X, see Section 2.

① Get the source from the developer download site: <http://www.graphviz.org/>

② Extract the following compressed file on a working directory:

```
# tar zxvf graphviz-2.16.1.tar.gz
```

③ Install (run as root)

```
# cd /<destination>/graphviz-2.16.1
```

```
# ./configure --prefix=/opt/graphviz-2.16.1
```

```
# make
```

```
# make install
```

* Define the following environment variables whenever necessary:

```
export LD_LIBRARY_PATH=/opt/graphviz-2.16.1/lib:$LD_LIBRARY_PATH
```

5. Gnu Scientific Library Installation

* If you use Mac OS X, see Section 2.

① Get the source from the developer download site: <http://www.gnu.org/software/gsl/>

② Extract the following compressed file on a working directory:

```
# tar zxvf gsl-1.9.tar.gz
```

③ Move to the extracted directory

④ Install (run as root)

```
# cd /<destination>/gsl-1.9
```

```
# ./configure --prefix=/opt/gsl-1.9
```

```
# make
```

```
# make install
```

* Define the following environment variables whenever necessary:

```
export LD_LIBRARY_PATH=/opt/gsl-1.9/lib:$LD_LIBRARY_PATH
```

6. Library for Numeric Calculation: BLAS/LAPACK Installation

* If you use Mac OS X, no numeric calculation library installation is needed.

Math Kernel Library (Linux noncommercial use) should be downloaded independently from the Intel Web site to get the license file:

<http://www.intel.com/cd/software/products/asm-na/eng/219771.htm>

① Download the numeric calculation library

* Example: Use Intel's Math Kernel Library - l_mkl_p_8.1.014.tgz

② Extract the following compressed file on a working directory:

```
# tar zxvf l_mkl_p_8.1.014.tgz
```

③ Move to the extracted directory

④ Install (run as root)

```
# cd /<destination>/ l_mkl_p_8.1.014
```

```
# ./install.sh
```

* Perform the installation in accordance to its installer's guidance

* What follows describes the steps of the installer flow for your reference:

Step 1: Select 1

Step 2: Select 2

Step 3: Input the full path to the license file - XXX.lic

Step 4: Select 1

Step 5: Input /tmp/mkl

Step 6: Input /opt/intel/mkl/8.1

* Define the following environment variables whenever necessary:

```
export LD_LIBRARY_PATH=/opt/intel/mkl/8.1/lib/32:$LD_LIBRARY_PATH
```

7. SBML Library Installation

* Xerces-C needs to be installed beforehand

* For the SBML library, use libsbml-2.3.4 (version 2.3.4)

① Get the source from the developer download site

② Extract the following compressed file on a working directory:

```
#unzip libsbml-2.3.4.zip
```

③ Move to the extracted directory

④ Correct the following minor bugs:

```
/<destination>/libsbml-2.3.4/src/util/StringBuffer.c
```

Change size = 16 in Function "StringBuffer_appendNumber" in line 151 to size = 32

```
/<destination>/docs/src/Makefile.in
```

Change formatted=../formatted in line 79 to formatted=../formatted/formatted

* What follows should be modified if you encounter any error during make in g++:

```
/<destination>/libsbml-2.3.4/src/sbml/SBMLSchemaInputSource.h
```

In line 79 :

(Before modification) BinInputStream* SBMLSchemaInputSource::makeStream () const;

(After modification) BinInputStream* makeStream () const;

Related URL: <http://www.sbml.org/forums/index.php?t=tree&goto=3837&rid=0>

⑤ Install (run as root)

```
# cd /<destination>/libsbml-2.3.4
```

(Where using Linux OS)

```
# ./configure --prefix=/opt/libsbml-2.3.4 --with-xerces=/opt/xerces-c-2.7.0
```

(Where using Mac OS X)

```
# ./configure --prefix=/opt/libsbml-2.3.4 --with-xerces=/opt/local
```

```
# make
```

```
# make install
```

⑥ Install documents (you may skip this step)

```
# make docs
```

```
# make install-docs
```


* Define the following environment variables whenever necessary:

(Where using Linux OS)

```
export LD_LIBRARY_PATH=/opt/libsbml-2.3.4/lib:$LD_LIBRARY_PATH
```

(Where using Mac OS X)

```
export DYLD_FALLBACK_LIBRARY_PATH=/opt/libsbml-2.3.4/lib
:$DYLD_FALLBACK_LIBRARY_PATH
```

8. Sundials Library Installation

* For the Sundials library, use sundials-2.1.1 (version 2.1.1)

① Get the source from the developer download site: <http://www.llnl.gov/CASC/sundials/>

② Extract the following compressed file on a working directory:

```
# tar zxvf sundials-2.1.1.tar.gz
```

③ Create include and lib directories for sundials:

```
# mkdir -p /opt/sundials/include
```

```
# mkdir -p /opt/sundials/lib
```

④ Install (run as root)

```
# cd /<destination>/sundials
```

```
# ./configure --prefix=/opt/sundials --enable-shared
```

```
# make
```

```
# make install
```

* Define the following environment variables whenever necessary:

(Where using Linux OS)

```
export LD_LIBRARY_PATH=/opt/sundials/lib:$LD_LIBRARY_PATH
```

(Where using Mac OS X)

```
export DYLD_FALLBACK_LIBRARY_PATH=/opt/sundials/lib
:$DYLD_FALLBACK_LIBRARY_PATH
```

9. SBML ODE Solver Library Installation

* libsbml-2.3.4, sundials-2.1.1, graphviz need to be installed beforehand

① Get the source from the developer download site: <http://www.tbi.univie.ac.at/~raim/odeSolver/>

② Extract the following compressed file on a working directory:

```
#tar zxvf SBML_odeSolver-1.6.0.tar.gz
```

③ Move to the extracted directory

④ Correct the following minor bug:

```
/<destination>/src/sbmlsolver/sbml.h
```

In line 68

(Before modification) extern "C" {

(After modification) }

⑤ (Only for Mac OS X)

Apply the patch (isSetValue.patch):

```
# cp isSetValue.patch SBML_odeSolver-1.6.0/
```

```
# cd SBML_odeSolver-1.6.0/src/
```

```
# patch -p1 < ../isSetValue.patch
```

Define the environment variables:

```
# export LDFLAGS='-L/opt/local/lib -lgraph'
```

⑥ Install (run as root)

* For either OS, specify in one line:

```
# cd /<destination>/SBML_odeSolver-1.6.0
```

(Where using Linux OS)

```
# ./configure --prefix=/opt/SBML_odeSolver-1.6.0
```

```
--with-libsbml=/opt/libsbml-2.3.4 --with-sundials=/opt/sundials
```

```
--with-grace=/usr/X11R6/lib/xmgrace --with-graphviz=/opt/graphviz-2.16.1
```

(Where using Mac OS X)

```
# ./configure --prefix=/opt/SBML_odeSolver-1.6.0
```

```
--with-libsbml=/opt/libsbml-2.3.4 --with-sundials=/opt/sundials
```

```
--with-grace=/opt/local --with-graphviz=/opt/local/lib/graphviz
```

--- Continued for Installation ---

```
# make
# make install
```

* Define the following environment variables whenever necessary:

(Where using Linux OS)

```
export LD_LIBRARY_PATH=/opt/SBML_odeSolver-1.6.0/lib:$LD_LIBRARY_PATH
```

(Where using Mac OS X)

```
export DYLD_FALLBACK_LIBRARY_PATH =/opt/SBML_odeSolver-1.6.0/lib
:$ DYLD_FALLBACK_LIBRARY_PATH
```

* Actions to be taken for “make” errors

I. An error message appears telling malloc.h cannot be included

```
Execute # sudo ln -s /usr/include/malloc/malloc.h /usr/include/malloc.h.
```

II. An error message appears telling arith.h cannot be included

```
Execute # sudo emacs /opt/local/include/graphviz/geom.h
```

Comment out #include "arith.h" around line 24 in geom.h using an editor

III. An error message appears telling -Unwind_Resume cannot be resolved

```
Execute # emacs odeSolver/Makefile .
```

Modify the coding around line 67:

(Before modification) CCLD = \$(CC)

(After modification) CCLD = g++

10. Qt Installation

* Use the Open Source version

① Get the source from the developer download site:

<http://www.trolltech.com/developer/downloads/qt/x11>

② Extract the compressed file:

```
# tar zxvf ./qt-x11-opensource-src-4.3.1.tar.gz
```

③ Rename the following directory to an appropriate name:

```
# mv qt-x11-opensource-src-4.3.1 qt-4.3.1
```

④ Install (run as root)

```
# cd /<destination>/qt-4.3.1
# ./configure -prefix /opt/qt-4.3.1
# make
# make install
```

* It takes longer to complete make and make install operations.

⑤ Define the environment variables after the installation

(Where using Linux OS)

```
export QTDIR=/opt/qt-4.3.1
export PATH=$QTDIR/bin:$PATH
export LD_LIBRARY_PATH=$QTDIR/lib:$LD_LIBRARY_PATH
```

(Where using Mac OS X)

```
export QTDIR=/opt/qt-4.3.1
export PATH=$QTDIR/bin:$PATH
export DYLD_LIBRARY_PATH=$QTDIR/lib:$DYLD_LIBRARY_PATH
```

[Verification]

```
> qmake -version
```

When the version information is displayed, the operation is done successfully

[Basic Flow of Qt Compilation]

```
> qmake -project
> qmake
> make
```

11. Qwt Installation

* Qt needs to be installed beforehand

- ① Get the source from the developer download site: <http://qwt.sourceforge.net/>
- ② Extract the following compressed file on a working directory:
unzip ./qwt-5.0.2.tar.gz

* For default setting, the file is installed in /usr/local/qwt-5.0.2. If you want to change the default setting, change the file location as follows:

```
qwt-5.0.2/qwtconfig.pri
unix {
    INSTALLBASE = **** should be changed (example /opt/qwt-5.0.2)
}
```

- ③ Install (run as root)

```
# cd /<destination>/qwt-5.0.2
# qmake
# make
# make install
```

* Define the following environment variables whenever necessary:

(When using Linux OS)

```
export LD_LIBRARY_PATH=/usr/local/qwt/lib:$LD_LIBRARY_PATH
```

(When using Mac OS X)

```
export DYLD_LIBRARY_PATH=/opt/qwt-5.0.2/lib:$DYLD_LIBRARY_PATH
```

[Additional] “make” an example (whenever necessary)

```
> cd /<destination>/qwt-5.0.1/example
> qmake
> make
```

Reference URL : <http://qwt.sourceforge.net/qwtinstall.html>

12. LetItB Installation

* All required libraries need to be installed beforehand

① Get the source: LetItB_ver_alpha.tar.gz

② Extract the following compressed file on a working directory:

```
# tar zxvf LetItB_ver_alpha.tar.gz
```

③ Specify the path to the required library (edit a file):

```
LetItB_ver_alpha/src/create_makefile.sh
```

Enter the corresponding INCLUDEPATH, LIBS (library path) in the create_makefile.sh file

④ Compilation

```
# ./create_makefile.sh
```

```
# make
```

* The compilation is successfully done when the execution file of "LetItB" is created in the LetItB_ver_alpha directory

⑤ (Only for Mac OS X) Create link files whenever necessary

```
# ln -s /<destination>/LetItB_ver_alpha/LetItB.app/Contents/MacOS/LetItB
```

```
/<destination>/LetItB_ver_alpha/LetItB
```

* Enter in one line

[How to Execute]

```
> ./LetItB
```

* No attribute exist at run time

[How to Use]

Refer to the LetItB User's Manual separately provided.

End