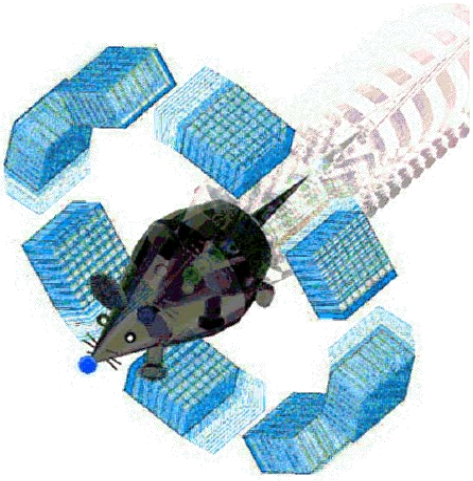

GATE Installation Guide

Version 5.0.0

for Windows XP and Vista

GATE

Geant4 **A**pplication for **T**omographic **E**mission:
a simulation toolkit for PET and SPECT

OpenGATE Collaboration
http://www.opengatecollaboration.org

Introduction

Currently the open-source package GATE is available for Mac OS X and Linux operating systems. In the following it will be described how GATE can be installed on a MS Windows XP or Vista platform. However, the installation is not straightforward, since a series of modifications on the source code level have to be applied.

This installation guide is separated into four parts:

- Installation of MS Visual C++ Compiler and Cygwin
- Installation of required tools for GATE
- Installation of GATE
- Results of PET and SPECT benchmarks

Installation of MS Visual C++ Compiler and Cygwin

For the Installation of GATE it is recommended to use the following freely available development environments:

- MS Visual C++ 2005 Express Edition with SP1
- MS Windows Server 2003 SP1 Platform SDK

1) Install MS Visual C++ 2005 Express Edition and start once to set environment variables
2) Install MS Windows Server 2003 SP1 Platform SDK
3) Start Visual C++ and make following changes under Extras/Options/Projects/VC++ directories:

   ”$(VCInstallDir)PlatformSDK\bin” → ”$(ProgramFiles)\Microsoft Platform SDK\Bin”
   ”$(VCInstallDir)PlatformSDK\include” → ”$(ProgramFiles)\Microsoft Platform SDK\include”
   ”$(VCInstallDir)PlatformSDK\lib” → ”$(ProgramFiles)\Microsoft Platform SDK\lib”

4) Open c:\Program Files\Microsoft Visual Studio 8\VC\VCProjectDefaults\ corewin_express.vsprops and replace everything with the following:

```
<?xml version="1.0"?>
<VisualStudioPropertySheet
   ProjectType="Visual C++"
   Version="8.00"
   Name="Core Windows Libraries">
   <Tool Name="VCLinkerTool"
        AdditionalDependencies="kernel32.lib user32.lib gdi32.lib winspool.lib
                                comdlg32.lib advapi32.lib shell32.lib
                                ole32.lib oleaut32.lib uuid.lib" />
</VisualStudioPropertySheet>
```

5) Open c:\Program Files\Microsoft Visual Studio 8\VC\VCWizards\AppWiz\Generic\Application\html\1031\AppSettings.htm and change the following part like this (line 478):

```
if (window.external.FindSymbol("IS_EXPRESS_SKU"))
    {
            SUPPORT_ATL.disabled = true;
            support_atl_title.disabled = true;
            SUPPORT_MFC.disabled = true;
            support_mfc_title.disabled = true;
    //      WIN_APP.disabled = true;
    //      WIN_APP_LABEL.disabled = true;
    //      DLL_APP.disabled = true;
    //      DLL_APP_LABEL.disabled = true;
    }
```

The installation process of Cygwin is described in the Geant4 installation guide:

http://geant4.slac.stanford.edu/installation

After installation change the cygwin.bat as follows (the standard installation directories were assumed):

```
@echo off

c:
chdir c:\cygwin\bin

call "c:\Program Files\Microsoft Visual Studio 8\VC\bin\vcvars32.bat"
call "c:\Program Files\Microsoft Platform SDK\SetEnv.Cmd"

bash --login -i
```

Add the following paths to c:\Program Files\Microsoft Visual Studio 8\VC\bin\vsvars32.bat:

```
@set PATH=c:\cygwin\bin;c:\Program Files\Microsoft Platform SDK\bin
@set INCLUDE= c:\cygwin\usr\include;c:\Program Files\Microsoft Platform SDK\include
@set LIB= c:\cygwin\lib;c:\Program Files\Microsoft Platform SDK\lib
```

Rename c:\cygwin\bin\link.exe to c:\cygwin/bin/link_cyg.exe, so that linking will be performed by the MS C++ Compiler and not by Cygwin. Be sure that c:\cygwin\bin\make.exe was replaced as described in the Geant4 installation guide.

Add the following to the bash.rc located in the home directory of the Cygwin directory (change the paths according to your own program directories):

```
export LD_LIBRARY_PATH=d:/Gate/CLHEP/lib/:$LD_LIBRARY_PATH
source d:/Gate/geant4/env.sh
export ROOTSYS=d:/Gate/root
export LD_LIBRARY_PATH=$ROOTSYS/lib:$LD_LIBRARY_PATH
export GATEHOME=d:/Gate/gate_v5.0.0
export G4WORKDIR=d:/Gate/gate_v5.0.0
export G4INSTALL=d:/Gate/geant4/
export G4VERSION=9.2
cd d:/Gate/
```

The MS C++ Compiler and Cygwin are only used for compilation and can be deinstalled afterwards.

Installation of required tools for GATE

- **CLHEP 2.0.4.2**
  Installation process and download page are provided in the Geant4 installation guide.
- **Geant4 9.2 patch-01**
  Installation process and download page are provided in the Geant4 installation guide.

- **ROOT 5.24/00**
  This software is available at: http://root.cern.ch/drupal
  Start cygwin.bat and switch to your ROOT directory. Then type:

  ```
  ./configure
  make
  ```

  After installation change \root\include\TString.h line 421 to:

  ```
  #ifdef NEED_STRCASECMP
  // extern int strcasecmp(const char *str1, const char *str2);
  extern int strncasecmp(const char *str1, const char *str2, Ssiz_t n);
  #endif
  ```

Installation of GATE

GATE 5.0.0 can be downloaded from http://opengatecollaboration.healthgrid.org/

Now comes the funny part ☺ Before GATE can be compiled successfully, a lot of modifications have to be carried out.

Delete \gate_v5.0.0\source\digitizer\src\GateSignalHandler.cc and \gate_v5.0.0\source\digitizer\include\GateSignalHandler.hh.

Copy the folder c:\cygwin\usr\include\libxml2\libxml as well as c:\cygwin\usr\include\iconv.h and c:\cygwin\usr\include\newlib.h and c:\cygwin\usr\include_ansi.h to c:\Program Files\Microsoft Platform SDK\Include.

Copy the folder c:\cygwin\usr\include\machine to c:\Program Files\Microsoft Platform SDK\Include and to c:\Program Files\Microsoft Platform SDK\Include\sys.

Then change c:\Program Files\Microsoft Platform SDK\Include\sys\times.h lines 10-12 to:

```
#ifndef _CLOCK_T_DEFINED
typedef long clock_t;
#define _CLOCK_T_DEFINED
```

In the following all changes are colored blue. The black colored lines should remain unchanged!

Files in \gate_v5.0.0

env_gate.sh

line 128 (change):

```
if [ ! -f xml2-config ] ; then
```

line 155 (change):

```
export G4ANALYSIS_USE_ROOT_PLOTTER=1
# unset G4ANALYSIS_USE_ROOT_PLOTTER
```

line 194 (change):

```
# export GATE_USE_LMF=1
```

gate.cc

line 40 (change):

```
#include "GateSourceMgr.hh"
// #include "GateSignalHandler.hh"
```

gate.cc

line 47 (insert):

```
#include "GatePulseProcessorChain.hh"
//ROOT
#include <TROOT.h>
#include <TPluginManager.h>
#include <TFile.h>
#include <TH1F.h>
#include <TMath.h>
```

line 112 (insert):

```
void DecodeArguments(int argc,char** argv)
{
/* magic line from Rene - for future reference! */
        gROOT->GetPluginManager()->AddHandler("TVirtualStreamerInfo",
                                              "*",
                                              "TStreamerInfo",
                                              "RIO",
                                              "TStreamerInfo()");
int nextArg = 1;
```

line 185 (change):

```
// Install the signal handler to handle interrupt calls
// GateSignalHandler::Install();
```

line 256 (change):

```
G4UIsession *session = (G4UIsession *)0;
// G4UItcsh* tcsh = new G4UItcsh;
```

line 263 (change):

```
#else
    session = new GateUIterminal;
#endif
```

variables.gmk

line 41 (insert):

```
# EXTRALIBS += `$(ROOTSYS)/bin/root-config --libs`
EXTRALIBS += -L$(ROOTSYS)/lib \
-lCore -lCint \
-lHist -lGraf -lGraf3d -lGpad \
-lTree -lRint -lPostscript -lRio \
-lMatrix -lPhysics -lMathCore -lNet \
-lm -ldl -lthread
endif
```

variables.gmk

line 51 (change):

```
#  XML, needed for optical transport
# ---------------------------------
ifdef GATE_USE_OPTICAL
CPPFLAGS  += -DGATE_USE_OPTICAL `xml2-config --cflags`
#EXTRALIBS += `xml2-config --libs`
EXTRALIBS += -L$(ROOTSYS)/lib \
-lxml2 -lm -liconv -lz
endif
```

delete everything after line 72 starting with:

```
# Deprecated or antiquated header
# and No overload warning
# These options are only usefull with GNU compiler
                    .
                    .
                    .
```

Files in \gate_v5.0.0\source\arf\src

GateARFTableMgr.cc 

line 353 (change):

Replace M_PI with 3.1415926535897

GateARFTable.cc 

line 353 (change):

Replace M_PI with 3.1415926535897

Files in \gate_v5.0.0\source\geometry\include

GateConeCreator.hh

line 39 (change):

Replace M_PI with 3.1415926535897

GateCylinderCreator.hh

line 58 (change):

Replace M_PI with 3.1415926535897

GateSphereCreator.hh

line 57, 58 (change):

Replace M_PI with 3.1415926535897

GatePolyconeCreator.hh

line 48 (change):

Replace M_PI with 3.1415926535897

Files in \gate_v5.0.0\source\geometry\src

GateCylinderCreator.cc

line 123, 124 (change):

Replace M_PI with 3.1415926535897

GateFreeAngularRepeater.cc

line 90, 92 (change):

Replace M_PI with 3.1415926535897

GateSphereCreator.cc

line 149, 150 (change):

Replace M_PI with 3.1415926535897

Files in \gate_v5.0.0\source\output\src

GateSinogramm.cc

line 202, 206, 210 (change):

if (fabs(G4double(crystal1ID - det1_c)) < fabs(G4double(crystal1ID - (det1_c + (G4int)m_crystalNb))))

if (fabs(G4double(crystal2ID - det1_c)) < fabs(G4double(crystal2ID - (det1_c + (G4int)m_crystalNb))))

if (fabs(G4double(diff1)) < fabs(G4double(diff2))) sigma = crystal1ID - crystal2ID;

GateToImageCT.cc

line 96, 103 (change):

original: if (fabs(fstepNumber-rint(fstepNumber)) >= 1.e-5) {

if (fabs(fstepNumber-floor(fstepNumber+0.5)) >= 1.e-5) {

original: m_frameNb = static_cast<size_t>(rint(timeStart/m_frameDuration));

m_frameNb = static_cast<size_t>(floor(timeStart/m_frameDuration+0.5));

GateToInterfile.cc

line 57 (insert):

```
#define BIG_ENDIAN      4321
#define LITTLE_ENDIAN   1234
/* All known win32 systems are little endian.  */
#define BYTE_ORDER      LITTLE_ENDIAN
```

GateToProjectionSet.cc

line 168, 174 (change):

original: if (fabs(fstepNumber-rint(fstepNumber)) >= 1.e-5) {

if (fabs(fstepNumber-floor(fstepNumber+0.5)) >= 1.e-5) {

original: m_projNb = static_cast<size_t>(rint(fstepNumber));

m_projNb = static_cast<size_t>(floor(fstepNumber+0.5));

GateToSinoAccel.cc

line 81, 87 (change):

original: if (fabs(fstepNumber-rint(fstepNumber)) >= 1.e-5) {

if (fabs(fstepNumber-floor(fstepNumber+0.5)) >= 1.e-5) {

original: m_frameNb = static_cast<size_t>(rint(fstepNumber));

m_frameNb = static_cast<size_t>(floor(fstepNumber+0.5));

line 293, 294, 342-345, 371 (change):

Replace M_PI with 3.1415926535897

GateToSinogramm.cc

line 123, 137 (change):

original: if (fabs(fstepNumber-rint(fstepNumber)) >= 1.e-5) {

if (fabs(fstepNumber-floor(fstepNumber+0.5)) >= 1.e-5) {

original: m_frameNb = static_cast<size_t>(rint(fstepNumber));

m_frameNb = static_cast<size_t>(floor(fstepNumber+0.5));

line 497, 498, 546-549, 623 (change):

Replace M_PI with 3.1415926535897

GateOutputMgr.cc

line 16 (insert):

#include <ctime>

line 206 (insert):

m_timer.Start();
startreal = clock();

line 228 (insert):

m_timer.Stop();
stopreal = clock();
double dResult = (static_cast<double>(stopreal - startreal)) / CLOCKS_PER_SEC;

line 231 (change):

G4cout << " Real simulation time (sec) := " << dResult << G4endl;

Files in \gate_v5.0.0\source\output\include

GateOutputMgr.hh

line 147 (insert):

G4Timer m_timer; //!< Timer
clock_t startreal, stopreal;

Files in \gate_v5.0.0\source\physics\include

GateSimplifiedDecay.hh

line 19 (delete):

Remove using namespace std;

Files in \gate_v5.0.0\source\mandatory_G4_classes\src

GateDetectorMessenger.cc

line 51 (change):

Replace M_PI with 3.1415926535897

Files in \gate_v5.0.0\source\management\src

GateExecutionTimeInformer.cc

line 36 (insert):

#define _SC_CLK_TCK 1
long sysconf(int name);

line 73 (change):

G4cout << "\r\nThe job was completed at " << asctime(localtime((const time_t*)realTime)) << "and needed " << (float)(time(NULL)-(*timeStamp)[firstTimeStamp])/float(sysconf(_SC_CLK_TCK)) << "s" <<

line 95 (change):

(*timeStamp)[s]=time(NULL);

line 97 (change):

G4cout << "Time stamp '" << s << "' set at " << asctime(localtime((const time_t*)realTime)) << flush;

line 111 (change):

G4cout << msg << " @ " << asctime(localtime((const time_t*)realTime)) << flush;

line 123 (change):

G4cout << "Time elapsed since last time stamp'" << lastTimeStamp << "': " << (float)(time(NULL)-(*timeStamp)[lastTimeStamp])/float(sysconf(_SC_CLK_TCK))

line 128 (change):

return(float)(time(NULL)-(*timeStamp)[lastTimeStamp])/float(sysconf(_SC_CLK_TCK));

GateExecutionTimeInformer.cc

line 139 (change):

G4cout << "Time elapsed since time stamp '" << s << "': " << (float)(time(NULL)-(*timeStamp)[s])/float(sysconf(_SC_CLK_TCK))

line 144 (change):

return(float)(time(NULL)-(*timeStamp)[s])/float(sysconf(_SC_CLK_TCK));

line 152 (change):

G4cout << "Processor cycles since last time stamp '" << lastTimeStamp << "': " << (float)(time(NULL)-(*timeStamp)[lastTimeStamp])

line 167 (change):

G4cout << "Processor cycles since time stamp '" << s << "': " << (float)(time(NULL)-(*timeStamp)[s])

GateOscTranslationMove.cc

line 74 (change):

Replace M_PI with 3.1415926535897

GateRandomEngine.cc

line 30 (insert):

#include <process.h>

line 124 (change):

seed = time(NULL)*_getpid();

For linking several dlls are required:

Copy following files from \root\bin\ to c:\Windows\System32\:

libCore.dll, libCint.dll, libHist.dll, libGraf.dll, libGraf3d.dll, libGpad.dll, libTree.dll, libRint.dll
libPostscript.dll, libRio.dll, libMatrix.dll, libPhysics.dll, libMathCore.dll, libNet.dll, libthread.dll

For linking several static libraries are required:

Copy following files from \root\lib\ to \gate_v5.0.0\tmp\WIN32-VC\Gate\

libCore.lib, libCint.lib, libHist.lib, libGraf.lib, libGraf3d.lib, libGpad.lib, libTree.lib, libRint.lib
libPostscript.lib, libRio.lib, libMatrix.lib, libPhysics.lib, libMathCore.lib, libNet.lib, libthread.lib

and rename them to:

libCore.a, libCint.a, libHist.a, libGraf.a, libGraf3d.a, libGpad.a, libTree.a, libRint.a
libPostscript.a, libRio.a, libMatrix.a, libPhysics.a, libMathCore.a, libNet.a, libthread.a

Download zlib.lib from here:

http://sourceforge.net/projects/gnuwin32/files/zlib/zlib-1.2.3-lib.zip/download

Unzip and rename \lib\zlib.lib to \lib\libz.a and copy to gate_v5.0.0\tmp\WIN32-VC\Gate\

Download zlib1.dll from here:

http://sourceforge.net/projects/gnuwin32/files/zlib/zlib-1.2.3-bin.zip/download

Unzip and rename \bin\zlib1.dll to \bin\libz.dll and copy to c:\Windows\System32\

Download libiconv.lib from here:

http://sourceforge.net/projects/gnuwin32/files/libiconv/libiconv-1.9.2-1-bin.zip/download

Unzip and rename \bin\libiconv.lib to \bin\libiconv.a and copy to \gate_v5.0.0\tmp\WIN32-VC\Gate\

Download libiconv.dll from here:

http://sourceforge.net/projects/gnuwin32/files/libiconv/libiconv-1.9.2-1-bin.zip/download

Unzip and rename \bin\libiconv2.dll to \bin\libiconv.dll and copy to c:\Windows\System32\

Download libxml2.lib from here:

http://sourceforge.net/projects/gnuwin32/files/libxml/libxml2-2.4.12-1-lib.zip/download

Unzip and rename \bin\libxml2.lib to \bin\libxml2.a and copy to \gate_v5.0.0\tmp\WIN32-VC\Gate\

Download libxml2.dll from here:

http://sourceforge.net/projects/gnuwin32/files/libiconv/libiconv-1.9.2-1-bin.zip/download

Unzip and copy \bin\libxml2.dll to c:\Windows\System32\

Copy \cygwin\lib\libdl.a to \gate_v5.0.0\tmp\WIN32-VC\Gate\

Copy \cygwin\lib\mingw\libm.a to \gate_v5.0.0\tmp\WIN32-VC\Gate\

That's all. Now Gate.exe can be created:

Start cygwin.bat and change to \gate_v5.0.0. Then type:

source env_gate.sh

make

After several minutes the following message should appear:

Using granular libraries …
Linking Gate …
… Done!

For the case that the compilation was successful, but a lot of errors occurred during linking with the following message at the end: ./bin/WIN32-VC/Gate.exe : warning LNK4088

then type again: make and normally after the second compilation linking works.

Setting of environmental variables

Go to Control Panel → System → Advanced tab and choose Environment variables.

Under System Variables double click on Path and add the paths to Gate.exe (e.g. d:\Gate\gate_v5.0.0\bin\WIN32-VC\) and to root.exe (e.g. d:\Gate\root\bin).

Under User Variables double click on New and add the following names and values:

Name	**Value (paths are user-specific)**
G4LEVELGAMMADATA	d:/Gate/geant4/data/PhotonEvaporation2.0
G4RADIOACTIVEDATA	d:/Gate/geant4/data/RadioactiveDecay3.2
G4LEDATA	d:/Gate/geant4/data/G4EMLOW6.2
G4NEUTRONHPDATA	d:/Gate/geant4/data/G4NDL3.13
G4ABLADATA	d:/Gate/geant4/data/G4ABLA3.0
GATEHOME	d:/Gate/gate_v5.0.0

For running GATE Gate.exe, Gate.exe.manifest (both in the same directory), the \geant4\data\ folder and in the GATEHOME directory the Materials.xml, GateMaterials.db and Surfaces.xml are required. GATE runs on all XP and Vista platforms with the Microsoft.NET Framework 2.0 installed.

To run a simulation open the DOS command prompt, switch to the folder containing the macro and then type gate.

For visualization the macro should contain the line: /vis/open OGLSWin32

Results of PET and SPECT benchmarks

The Windows version of GATE 5.0.0 was successfully validated with the PET and SPECT benchmarks provided in the GATE source code.

PET benchmark results

Variable type	Value
total decays during the acquisition	$3.6809 \cdot 10^{7}$
random coincidences	23666
unscattered coincidences	313492
scattered coincidences	372899
simulated O15 half-life	121.52
gamma acollinearity angle	0.58

SPECT benchmark results

Global information

Variable type	Value
number of emitted particles	17997870
detected counts between 20 and 190 keV	36041

Percentage of photons whose last scattered event occured in a specific medium

Medium	Value
Phantom	53.19
Table	2.89
Collimator	0.31
Crystal	6.24
Backcompartment	1.14

Percentage of scattered photons as a function of the scattering order

Scattering order	Value
order 1	48.75
order 2	25.64
order 3	12.77
order 4	6.52
order >4	6.32