



GuineaPig++ with XtremWeb-HEP

6eme réunion codes de calcul accélérateur (GCAO)

Lundi 31 janvier 2010

LAL

Simon Dadoun sdadoun@lal.in2p3.fr

Je suis au SI du LAL sur le projet européen DEGISCO (Oleg Lodyginsky)



GuineaPig+ with Xwhep
Author: S.Dadoun

GuineaPig++ avec XtremWeb-HEP

- 1) Presentation of XtremWeb-HEP
- 2) Submit a GuineaPig++ job and retrieve results
- 3) Example of multiple submission

4) Annex :

Used GuineaPig++ parameters file

Install and configure XtremWeb-HEP Client

Example of macro file

Presentation of Xtrem Web-HEP (Xwhep)

For intensive computation needs, there is the computer farms, the grids (e.g.: gLite middleware), and the desktop grids like Xwhep.

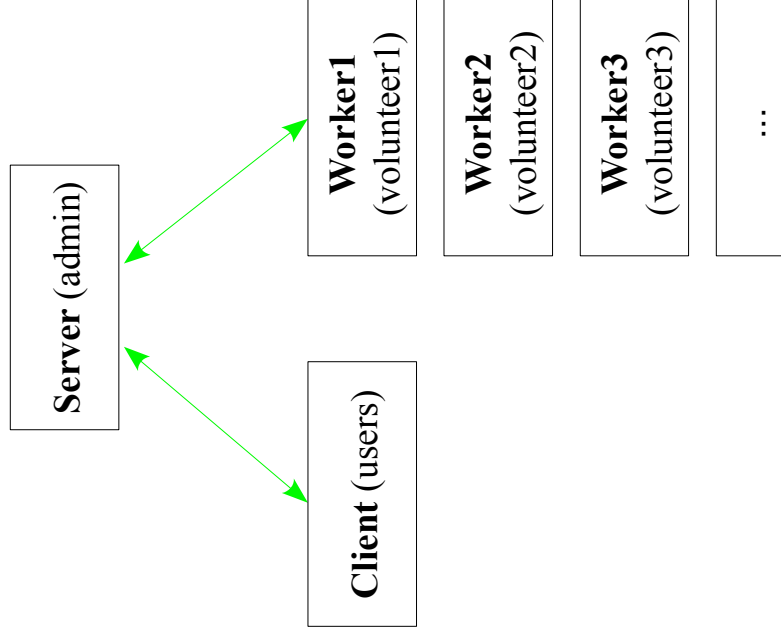
Xwhep is a desktop grid middleware like BOINC (SETI@Home), also known as volunteer computing.

Not for parallelized computing (because worker nodes cannot communicate with each other).

Three programs : server, worker and client.

Derived by Oleg Lodygensky from INRIA project Xtremweb.

Currently the LAL Xwhep deployment is composed by approximately 12 Linux-AMD64 and 25 Mac-Intel.



XtremWeb-HEP : admin

List of current available applications at LAL's installation :

```
UID='a87d6769-9694-46f7-9120-55b03da33095', NAME='DART'  
UID='3df0653c-b3f0-439e-b5f6-c28c9bd306e3', NAME='bnbss'  
UID='0cf21428-7c9a-42f3-a478-b684492cbbb9', NAME='cat'  
UID='e3fdd9a2-367e-4d22-87c1-b201deec6b9d', NAME='test'  
UID='afd5c6-12a7-4f54-9e34-e782f62d3991', NAME='VINA'  
UID='8f582f85-100e-4505-8bd7-22dc0c5a6dfd', NAME='sgmuons'  
UID='2e3ac23a-ed6b-4041-89bd-64af6a23be87', NAME='dsp'  
UID='de7d2350-c0e1-4229-8615-1939340c4bd9', NAME='isdep'  
UID='a737cd8f-204e-413b-a427-8aaf6c3c56a4', NAME='ls'  
UID='210eaaa-9e79-41a7-b35f-d0b442af0f52', NAME='FFMPEG'  
UID='0b7c51d9-717d-48e2-8be7-c3e315b0eec9', NAME='slinca'  
UID='7a0485cd-ab1d-491c-9a36-3c9aa5ab19db', NAME='scale'  
UID='6d185fdd-ab53-4ed4-b854-34338f9cab1d', NAME='autodock4_amd64'  
UID='6671debc-f2bc-49f1-ad2e-dc00d8240d19', NAME='guineapig'
```

To use one of those apps you need to install the client

You can install it from there :

<http://www.xtremweb-hep.org/spip.php?article52>

Submit a GuineaPig job and retrieve results

Verify GuineaPig++ is available

```
$> xwapps | grep guineapig
```

```
UID='6671debc-f2bc-49f1-ad2e-dc00d8240d19', NAME='guineapig'
```

Submit a job with guineapig application

```
$> xwsubmit guineapig nominal pairsprod output.dat --xwenv acc.dat  
xw://xwserv.lal.in2p3.fr:4321/e5f1049e-2078-4167-bcf5-9bd388a721ce
```

You can check the job status at any time

```
$> xwstatus
```

```
UID='7c5da2cf-1168-461b-b8fc-2a586599f175', STATUS='RUNNING',  
COMPLETEDDATE=NULL, LABEL=NULL, ERROR_MSG=NULL
```

Submit a GuineaPig job and retrieve results

When the job is completed, you retrieve its result

```
$> xwresults 7c5da2cf-1168-461b-b8fc-2a586599f175
```

This downloads the result zip file,

```
$> gunzip 823075a8-1de1-40c2-b7ae-71064330b186_ResultsOf_ecbba1f9-  
2d77-428a-a2a3-2001636d08cf.zip
```

which contains the following files :

```
beam1.dat      beam2.dat      output.dat      photon.dat  
rndm.save     secondaries.dat stderr.txt      stdout.txt
```

important acc.dat parameters :

```
n_x=32 ; n_y=64 ; n_z=32 ; n_t=5 ; n_m=30000 ;
```

Example of multiple submissions

This shell script submits 10 jobs :

```
#!/bin/sh
rm UIDS/*

for (( i=0; i<=9; i++ ))
do
    echo "xwsubmit guineapig nominal pairsprod output$i.txt --xwenv acc.dat >
UIDS/jobuid$i"

    xwsubmit guineapig nominal pairsprod output$i.txt --xwenv acc.dat > UIDS/jobuid$i

done
```

Example of multiple submissions

This shell script monitors the jobs (we can do it manually with `xwstatus`) :

```
#!/bin/sh

LOOPTEST=2
while [ $LOOPTEST -ne 1 ];
do
    xwstatus > xwstatus.tmp
    cat xwstatus.tmp | while read line; do echo "${line}"; done
    echo ""
    PENDING=`cat xwstatus.tmp | grep PENDING | wc -l`;
    echo "$PENDING pending jobs";
    RUNNING=`cat xwstatus.tmp | grep RUNNING | wc -l`;
    echo "$RUNNING running jobs";
    LOOPTEST=$(( $PENDING + $RUNNING ));
    sleep 30
done
echo "Finished!"
```


Example of multiple submissions

This shell script removes previously submitted jobs :

```
#!/bin/sh  
  
for job in `ls UIDS`  
do  
  
    echo "xwrm `cat UIDS/$job`"  
  
    xwrm `cat UIDS/$job`  
  
done
```

Time recorded

Time recorded by 2 configurations :

One run on a lx2 machine :

```
time ./guineapig++.exe nominal pairsprod output
n_m=30000 207 seconds ;
n_m=50000 393 seconds ;
```

10 jobs on Xwhep at LAL:

```
time ./guigogo.sh
n_m=30000 360 seconds ; benefit ~1600 seconds
n_m=50000 1000 seconds ; benefit ~3000 seconds
```

The longer the jobs are, the more benefit you have using Xwhep.

Conclusion

Xwhep is a good opportunity to use huge amount of CPUs very easily.

Volunteers are welcome to increase number of CPUs.

Prospect : we plan to provide a web user interface to run GuineaPig on a desktop grid using the LAL workers (futur accelerator simulation platform).

A bridge Xwhep to EGEE exists, under some security conditions, a job submitted on Xwhep can be computed by an EGEE ressource.

Annex : content of acc.dat

Used GuineaPig++ parameters file

```

$ACCELERATOR:: nominal
{
  energy=250.0;
  particles=2;
  beta_x=21.0;
  beta_y=0.4;
  offset_x=0.;
  offset_y=0.;
  angle_x = 0.0;
  emitt_x=10.0;
  emitt_y=0.04;
  sigma_z=300.0;
  charge_sign=-1;
}

$PARAMETERS:: pairsprod
{
  n_x=32 ;
  n_y=64 ;
  n_z=32 ;
  n_t=5 ;
  cut_x=3.0*sigma_x.1 ;
  cut_y=6.0*sigma_y.1 ;
  cut_z=3.0*sigma_z.1 ;
  n_m=30000 ;
  force_symmetric=0;
  integration_method=2 ;
  do_loss = 1 ;
  do_isr = 0;
  store_beam=1 ;
  electron_ratio=0.5 ;
  do_photons=1 ;
  photon_ratio=1. ;

  store_photons=1 ;
  do_pairs=1 ;
  track_secondaries=1;
  store_secondaries=2;
  do_compt = 0;
  grids=7 ;
  pair_ratio = 1.;
  pair_ecut = 0.005;
  pair_q2=2;
  beam_size=1;
  ext_field=0;
  do_compt_phot = 0;
  do_hadrons=0 ;
  store_hadrons = 0 ;
  do_jets=0 ;
  store_jets=0 ;
  rndm_seed =232341;
  rndm_load=0;
  rndm_save = 0;
  load_beam = 0;
}

```

Annex : XtremWeb-HEP, configure the client

To submit you need Xwhep client, you can install it from there :

<http://www.xtremweb-hep.org/spip.php?article52>

(It is available for Mac, Scientific-Linux (rpm), Debian (deb), it required Sun JRE 1.6).

Then copy the configuration template :

```
$> mkdir $HOME/.xtremweb  
$> cp /Applications/xwhep.client/conf/xtremweb.client.conf \  
$HOME/.xtremweb ( On Linux /opt/xwhep-client-{$VERSION} )
```

In \$HOME/.xtremweb/xtremweb.client.conf delete the login and password parameters.

Set the environment variable X509_USER_PROXY with the path and name of your proxy :

```
$> export X509_USER_PROXY=$HOME/.globus/myproxy
```

Or on Linux setenv X509_USER_PROXY "\$HOME/.globus/myproxy"

Indeed to use the Xwhep client you need to authenticate, either with login/password or with a X509 proxy, here we use X509 proxy.

To generate a proxy you need to have a X509 certificate and belong to a VO, both accepted by LAL.

More informations available on Xwhep svn, doc/DEGISCO_Tutorial.odp

Annex : Example of macro file

If you need to submit a lot of jobs (>50) quickly, you may consider using the xwmacro option. It allows in one java call to submit several commands.

This script generate a macro file and submit it :

```
#!/bin/sh
echo "--xswendwork guineapig ILC LC-GENERAL 000 --xwenv acc.dat" >> xwmacro.txt;
echo "--xswendwork guineapig ILC LC-GENERAL 000 --xwenv acc.dat" >> xwmacro.txt;
echo "--xswendwork guineapig ILC LC-GENERAL 000 --xwenv acc.dat" >> xwmacro.txt;
echo "Submit macro file...."
xwversion --xwmacro xwmacro.txt > jobuids
echo "Monitor the jobs...."
#add --xwget at the beginning of each line
sed -i 's/^/--xwget /' jobuids
#add --xwformat at the end of each line
sed -i 's/$/ --xwformat xml/g' jobuids
```

Then the file jobuids contains :

```
--xwdownload --xwget xw://xwserv.lal.in2p3.fr:4321/032...063ab0435 --xwformat xml
--xwdownload --xwget xw://xwserv.lal.in2p3.fr:4321/8a5...bde1ed6d3 --xwformat xml
--xwdownload --xwget xw://xwserv.lal.in2p3.fr:4321/b38...4a165632b --xwformat xml
```

Annex : Example of macro file

This part monitor the jobs status and stop when there is no more PENDING or RUNNING jobs :

```
xwversion --xwmacro jobuids > res
LOOPTEST=1
while [ $LOOPTEST -ne 0 ] ; do
  echo "Check jobs's status"
  PENDING=`cat res | grep PENDING | wc -l` ;
  RUNNING=`cat res | grep RUNNING | wc -l` ;
  echo "$PENDING pending jobs, $RUNNING running jobs";
  LOOPTEST=$(( $PENDING + $RUNNING )) ;
  sleep 3;
xwversion --xwmacro jobuids > res;
done
```

The file res contains the xml description of each jobs, e.g. :

```
<work uid="032a...0435" userproxy="xw://xwserv.lal.in2p3.fr:4321/260d...85f7"
isservice="false" isdeleted="false" appuid="6671...0d19" owneruid="972c...6eae"
accessrights="0x755" status="COMPLETED" returncode="0" cmdLine="IIC LC-GENERAL
ooo" dirinuri="xw://xwserv.lal.in2p3.fr:4321/e81c...b886"
resulturi="xw://xwserv.lal.in2p3.fr:4321/fd41...5d0d" arrivaldate="2011-01-26
16:25:16" completeddate="2011-01-26 16:25:58" sendtoclient="false" local="true"
active="true" replicated="false" maxretry="0" minmemory="0" mincpuspeed="0" />
```


Annex : Example of macro file

Finally this part retrieves the result zip file of each jobs :

```
echo "retrieve results of completed jobs and clean temporary files"  
#add --xdownload at the beginning of each line  
sed -i '' 's/^/--xdownload /' jobuids  
xwversion --xwmacro jobuids  
rm -f jobuids res xwmacro.txt
```