

1. Setting up the Simulator

1. The configuration file (cobalt.conf)
2. The simulator component

2. Starting the Simulator

1. Using the init.d script
2. Manually

3. Queue Simulator

Setting up the Simulator

Note: Cobalt version 0.98.x requires **python 2.5** and the **tlslite** python package

Checkout from svn, and set symlink

```
svn co https://svn.mcs.anl.gov/repos/cobalt/trunk cobalt
ln -s cobalt/src/lib cobalt/src/Cobalt
```

Add cobalt/src/ to \$PYTHONPATH

The configuration file (cobalt.conf)

Edit the DEFAULT_CONFIG_FILES variable in cobalt/src/lib/__init__.py to match the location of your cobalt.conf file,

or

Set the environment variable \$COBALT_CONFIG_FILE

Edit cobalt.conf as below (see cobalt/misc/cobalt.conf)

```
[bgpm]
mpirun: /path/to/cobalt/src/clients/brun.py

[cqm]
log_dir: /path/to/writable/dir

[bgsched]
utility_file: /path/to/empty/file

[statefiles]
location: /path/to/writable/dir

[components]
service-location=https://localhost:port
```

Generate cobalt.key

```
openssl req -x509 -nodes -days 1000 -newkey rsa:1024 -out cobalt.key -keyout cobalt.key
```

Edit the key variable in the [communication] section of cobalt.conf to point to cobalt.key

```
[communication]
key=/path/to/cobalt.key
password=password
```

The simulator component

The simulator component (brooklyn) reads in system information from an XML file. Use cobalt/misc/partitions.xml for reference.

```
ln -s cobalt/misc/partitions.xml cobalt/src/components/simulator.xml
```

You can also dump a valid xml machine representation from a running system using partadm.py

```
partadm.py --xml
```

Starting the Simulator

Starting the simulator means starting the components as usual, replacing bgsystem with brooklyn in the startup sequence.

Using the init.d script

```
sh cobalt/misc/cobalt start-debug
```

Manually

```
cd cobalt/src/components
./slp.py & ./brooklyn.py & ./cqm.py & ./scriptm.py & ./bgsched.py &
```

Queue Simulator

See [this page](#) for info on qsim.