Profiling a Job

The information in this document is targeted primarily towards the RFPK Software Team and associates and is specific to the computer systems and network installed in the RFPK Laboratory of the Department of Bioengineering of the University of Washington. RFPK is the Resource for Population Kinetics. Its work is supported, in part, by grant P41 EB-001975 of the National Institutes of Health (NIH) of the U.S. Department of Health and Human Services.

Table of Contents

Prerequisites

This document explains how to profile an existing job. In particular, it assumes there is at least an existing job with full source code in the ’/tmp’ directory on the CSPK server. For the rest of this documentation, I assume we are going to profile a job numbered 38. Thus, its full source code must be found in cspk::/tmp/spkruntest-job-38 like so:

```
ssh cspk
su
cd /tmp/spkruntest-job-38
ls -al
total 4336
drwx------    2 root     root         4096 Dec 20 16:34 .
drwxrwxrwt  422 root     root        12288 Dec 20 16:44 ..
-rw-rw-rw-    1 root     root        31072 Dec 20 16:33 DataSet.h
-r--r--r--    1 root     root         8891 Dec 20 16:33 data.xml
-rw-rw-rw-    1 root     root        42403 Dec 20 16:33 fitDriver.cpp
-rw-rw-rw-    1 root     root        17305 Dec 20 16:33 IndData.h
-rw-rw-rw-    1 root     root         1157 Dec 20 16:33 Makefile.SPK
-rw-rw-rw-    1 root     root         5002 Dec 20 16:33 NonmemPars.h
-rw-rw-rw-    1 root     root         8570 Dec 20 16:33 Pred.h
-rw-rw-rw-    1 root     root         2601 Dec 20 16:33 source.xml
```

There may be more files (ex. "driver", the binary executable) found in the directory, but the above list lists all necessary files.

Build SPK and PRED Libraries for Profiling

Both SPK and PRED libraries on the CSPK server need to be built with profiling instruments.

```
ssh cspk
cd r2/src/apps/spk/cspk/spk
make clean
./configure --enable-profiling --enable-release-build
```

```
cd spk
make
su
make install
exit


cd ../../pred
make clean
./configure --enable-profiling --enable-release-build
cd pred
make
su
make install
exit
```

Modify Makefile.SPK

The Makefile.SPK file found in the CSPK's /tmp/spkruntest-job-38/ directory contains commented lines that are meant to be activated when someone wants to profile the job.

```
su
cd /tmp/spkruntest-job-38/
```

Open Makefile.SPK with your favorite editor and locate the following lines in the file.

```
# C++ compiler flags to build a release version.
#CXX_FLAGS = -O3 -Dspk_release -DNDEBUG
# C++ compiler flags to build a debug version.
CXX_FLAGS = -g

# C++ compiler flags to turn on profiling
# CXX_FLAGS += -pg -Dspk_profiling
```

Uncomment the 2nd line and 7th line. Comment the 4th line. It should now look like this:

```
# C++ compiler flags to build a release version.
CXX_FLAGS = -O3 -Dspk_release -DNDEBUG
# C++ compiler flags to build a debug version.
#CXX_FLAGS = -g

# C++ compiler flags to turn on profiling
CXX_FLAGS += -pg -Dspk_profiling
```

Build the job and examine the profiling info

Rebuild the job and run it to generate a profiling info.

```
make -f Makefile.SPK -test
./driver
```

It should generated gmon.out file. Now you can examine the output via "gprof":

```
gprof driver | less
```

The info contains two types of outputs: flat and hierhachical call graphs. For more information, consult the man page of "gprof".

Copyright Notice

Notes

1. http://www.opencontent.org/openpub/