Creating your own Executable

Dynamic linking does not leave enough information to fully use symbolic debugging.

Create your own pigiRpc, ptrimRpc, or ptinyRpc that includes your stars.

An example is provided in $PTOLEMY/src/pigiExample. Copy the contents of this directory, replace your star with the example there (SDFMyStar.pl) and recompile:

```
make ptinyRpc.debug
```

You may optionally edit the file there "defpalettes.c" to change the default domain and the list of palettes you wish displayed. The make.template is extensively commented. READ IT.

More Extensive Debugging

Make a debug version of the executable (e.g. on Solaris):

```
cd $PTOLEMY/obj.sol2/pigiRpc
make ptinyRpc.debug
```

Make the smallest debug version that contains the code you are testing (ptinyRpc, ptrimRpc, or pigiRpc).

Running the new executable:

```
setenv PIGIRPC $PTOLEMY/obj.sol2/pigiRpc/ptinyRpc.debug
pigi -debug
```

Optional, but strongly recommended (before the above, or in your .cshrc file):

```
setenv PT_DEBUG ptgdb
```

This specifies to use emacs and gdb (a powerful combination).

Stack Traces

```
% gdb pigiRpc
    ...
Core was generated by `pigiRpc'.
Program terminated with signal 11, Segmentation fault.
#0 0x2eebb4 in end ()
(gdb) where
#0 0x2eebb4 in end ()
#1 0x1c774c in SimControl::doPreActions ()
#2 0x1c8538 in Star::run ()
#3 0x16ee74 in DataFlowStar::run ()
#4 0x16d888 in SDFScheduler::runOnce ()
#5 0x16d7c4 in SDFScheduler::run ()
#6 0x1cb4c0 in Target::run ()
#7 0x1cd4e4 in Runnable::run ()
#8 0x1bd810 in InterpUniverse::run ()
#9 0x197774 in KcRun ()
#10 0x18edac in RpcRun ()
#11 0x18eeb8 in RpcReRun ()
#12 0x177634 in RPCApplicationProcessEvents ()
#13 0x17675c in RPCMain ()
#14 0x2300 in main ()
```

Core Dumps

Recall that Ptolemy runs as two processes:

- **vem**
- **pigiRpc**

Vem rarely crashes in normal usage, but since Ptolemy is extensible, pigiRpc is fairly likely to crash as you develop code (or discover bugs in our code). You may get any of several messages, among them:

```
RPC Error: server: application exited without calling
RPCExit
Closing Application /home/ohm1/users/messer/ptolemy/lib/
pigiRpcShell on host foucault.berkeley.edu
Elapsed time is 1538 seconds
```

You may get the messages:

```
segmentation fault (core dumped)
```

The vem interface can no longer issue Ptolemy commands.

Debugging

- Dealing with core dumps
- Interpreting stack traces
- Running a symbolic debugger
- Creating custom executables

Note that if you are using Ptolemy as a software development environment (a reasonable use), you need to invest some effort in learning to use reasonable tools (make, gdb, emacs). Of these, only make is distributed with Ptolemy, but all are freely available on the net.