Debugger Update

David Robboy

Some major Bugs fixed in R2.3

- debug forgets that attached program was killed, no need to detach
- debug has an incorrect context when rerunning a program that terminated
- Debug won't load the newest version of pronto
- Debug: loading programs has a serious performance problem
- Debug doesn't display floating point registers correctly

More bugs fixed

- Debug sometimes fails to attach to running process on many nodes
- Attaching to nonexistent process hangs the debugger
- *whatis <class>.<member function>()* cannot find symbol
- Debug cannot find a function inherited from a parent C++ class
- If current scope is a class, "*whatis member*" gets a memory fault.
-
- There is a debug man page now.

Scalability of the ‘Print’ Command

- ‘Print’ takes a long time on many nodes if the value is different on each node
- Also applies to ‘where’ and other commands that can have a different value on each node
- This is a feature of the debugger design
- Workaround: specify a small context on the command line
  - print (0) x or print (0..32) x
  - But not just print x

Outstanding bugs
Big System bugs

- ‘where’ command seg. faults on 1025 nodes stopped at different places
- assign/set command hangs on many nodes
- step/next hangs on 2000 nodes
- To work on these requires lots of dedicated Janus time
  - And they’re hard
- How important? Should we work on them?

Outstanding bugs
C++

- Debug can’t find C++ member functions by indirection
- Debug can’t put breakpoints on member functions if they are local
- Specific to C++, relatively obscure, not fundamental to the working of the debugger
- Is this important to users?

Outstanding bugs
F90

- Debug can’t load f90 programs with assumed size/shape arrays
- Depends on correct DWARF information from PGI
- Plan to fix it when the compiler is fixed

What I want to know

- What current debugger problems are important to users?
- What should we work on?