Subject: GDB/MINGW interface Bug fix
Posted by omari on Thu, 02 Apr 2015 07:35:56 GMT
View Forum Message <> Reply to Message

Hi,

for the first time, I can use MINGW/GDB to compile and debug my application without crash.

there is two buges behind this crash
please, consider the application of this patch:

1 - file ide/Debuggers/Gdb_MI2Gdb.cpp line 173, in function MIValue Gdb_MI2::ReadGdb(bool wait)
 replace

```
  dbg->Read(s);
  StringStream ss(s);
```

 by

```
  String stmp;
  do
  {
   stmp = "";
   dbg->Read(stmp);
   s << stmp;
  } while(stmp.GetCount());
  StringStream ss(s);
```

 read all available output from gdb process, because one call to gdb->Read return 1024 character max.

2 - file ide/Debuggers/MIValue.h/.cpp

replace "rval_" by "const &"

when theide is compiled with COMPILER_MSC, "rval_" is evaluated as "&" (it is the case for the binary distribution of theide i think)
then the priority of constructors is changed:in a call like this MIValue v = MICmd("...");
MIValue(cons String&) is user by the compiler, instead of MIValue(MIValue&). (the MIValue is converted to String, then the constructor is called with the String as param)

NB:
this is the summary of the following threads

patch Gdb_MI parser   :http://www.ultimatepp.org/forums/index.php?t=msg&th=925 9&start=0&

[BUG?] MIValue :   http://www.ultimatepp.org/forums/index.php?t=msg&th=9260 &start=0&

Subject: Re: GDB/MINGW interface Bug fix
Posted by omari on Fri, 03 Apr 2015 14:55:11 GMT
View Forum Message <> Reply to Message

Hi,

the first issue can be resolved in LocalProcess.cpp LocalProcess::Read2(), line 448 and 452
replacing "if" by "while" :

replace:

```
#ifdef PLATFORM_WIN32
 LLOG("LocalProcess::Read");
 bool was_running = IsRunning();
 char buffer[1024];
 dword n;
---> if(hOutputRead && PeekNamedPipe(hOutputRead, NULL, 0, NULL, &n, NULL) && n &&
    ReadFile(hOutputRead, buffer, sizeof(buffer), &n, NULL) && n)
  reso.Cat(buffer, n);

---> if(hErrorRead && PeekNamedPipe(hErrorRead, NULL, 0, NULL, &n, NULL) && n &&
    ReadFile(hErrorRead, buffer, sizeof(buffer), &n, NULL) && n)
  rese.Cat(buffer, n);

 if(convertcharset) {
  reso = FromOEMCharset(reso);
  rese = FromOEMCharset(rese);
 }

 return reso.GetCount() || rese.GetCount() || was_running;
#endif
```

by

```
#ifdef PLATFORM_WIN32
 LLOG("LocalProcess::Read");
 bool was_running = IsRunning();
 char buffer[1024];
 dword n;
---> while(hOutputRead && PeekNamedPipe(hOutputRead, NULL, 0, NULL, &n, NULL) && n &&
    ReadFile(hOutputRead, buffer, sizeof(buffer), &n, NULL) && n)
  reso.Cat(buffer, n);
```

```
---> while(hErrorRead && PeekNamedPipe(hErrorRead, NULL, 0, NULL, &n, NULL) && n &&
    ReadFile(hErrorRead, buffer, sizeof(buffer), &n, NULL) && n)
  rese.Cat(buffer, n);

 if(convertcharset) {
  reso = FromOEMCharset(reso);
  rese = FromOEMCharset(rese);
 }

 return reso.GetCount() || rese.GetCount() || was_running;
#endif
```
