

---

Subject: Re: Considering different approach to Win32 release

Posted by [mirek](#) on Thu, 29 Oct 2015 12:22:24 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 29 October 2015 09:42Hi Mirek,

I have now tested 9105 on Windows 8.1 Professional x64 with the following results:

Thanks for testing! Definitely helpful.

Quote:

2. It still can't find the installed Visual studio 2015 community edition. Is this expected?

That is definitely unexpected and "wrong".

Well, instead of long and detailed request for investigation, could you please just look into `ide/InstantSetup.cpp` and put on line 180

```
vc = df.ScanForDir("/vc", "", "bin/link.exe;bin/cl.exe;bin/mspdb140.dll", "bin/1033");
bin = df.ScanForDir(x64 ? "bin/x64" : "bin/x86", "/windows kits/", "makecat.exe;accevent.exe", "");
inc = df.ScanForDir("", "/windows kits/", "um/adhoc.h", "um;ucrt;shared");
lib = df.ScanForDir("", "/windows kits/", "um/x86/kernel32.lib", "um;ucrt");
```

```
DUMP(vc); DUMP(bin); DUMP(inc); DUMP(lib); // <==== THIS
```

```
if(vc.GetCount() * bin.GetCount() * inc.GetCount() * lib.GetCount()) {
    bins.At(0) = vc + (x64 ? "/bin/amd64" : "/bin");
    bins.At(1) = bin;
    String& sslbin = bins.At(2);
```

and then run "Instant setup.." and look into .log?

(Explanation: Instead looking into registry (which proved error-prone with MSC12), theide now scans Program files directories and attempts to find appropriate directories.)

Quote:

3. MINGW Debug compiles UWord correctly and the resulting executable works.

4. MINGW Size compiles UWord correctly and the resulting executable works.

5. MINGW Optimal compiles UWord but the resulting executable crashes. It requires dropping the speed optimization flag from O3 to O2 in order to make it work.
6. MINGW Speed compiles UWord but the resulting executable crashes. It requires dropping the speed optimization flag from O3 to O2 in order to make it work.
7. MINGWx64 Debug compiles UWord correctly and the resulting executable works.
8. MINGWx64 Size compiles UWord but the resulting executable crashes. (Switching from Os to e.g. O0 in size optimization flags fixes the executable.)
9. MINGWx64 Optimal compiles UWord but the resulting executable crashes. (Switching from Os to O0, O1, O2 or O3 in size optimization flags fixes the executable.)
10. MINGWx64 Speed compiles UWord correctly and the resulting executable works.

Best regards,

Tom

Unfortunately, investigation revealed apparent bug in GCC optimizer (you can compile in release mode with full debug info and run in debug - bug in assembly is quite apparent).

I will try to downgrade gcc version. If that does not help (it should, in Linux it works fine), I will change -O levels...

Mirek

---