Subject: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Mon, 13 Dec 2021 09:04:21 GMT
View Forum Message <> Reply to Message

Hi,

I just set up a new computer with Windows 11 and Visual Studio Build Tools 2022. It seems that TheIDE is not yet capable of detecting the compilers and tools (2017, 2019 and 2022) that come with this release.

Could these be added to the automatic setup of build methods?

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by mirek on Sun, 19 Dec 2021 14:16:14 GMT
View Forum Message <> Reply to Message

Sure. Why do not you try yourself? I would gladly merged a patch...

It should be really easy, it might even be as easy as adding single line around ide/InstantSetup.cpp:237 (with new enum constant for 2022)

Mirek

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Lance on Sun, 19 Dec 2021 14:44:36 GMT
View Forum Message <> Reply to Message

I tried, somehow I could not figure it out.

```
void InstantSetup()
{
 bool dirty = false;
 String default_method;

 String bin = GetExeDirFile("bin");

 if(DirectoryExists(bin + "/clang"))
  for(int x64 = 0; x64 < 2; x64++) {
   String method = x64 ? "CLANGx64" : "CLANG";
  #ifdef INSTANT_TESTING
   method << "Test";
```

```
#endif
 VectorMap<String, String> bm = GetMethodVars(method);

 Vector<String> bins = Split(bm.Get("PATH", ""), ';');
 Vector<String> incs = Split(bm.Get("INCLUDE", ""), ';');
 Vector<String> libs = Split(bm.Get("LIB", ""), ';');
#ifdef INSTANT_TESTING
 if(CheckDirs(bins, 3) && CheckDirs(incs, 2) && CheckDirs(libs, 2)) {
  if(!x64)
   default_method = Nvl(default_method, method);
  continue;
 }
#endif

 bmSet(bm, "BUILDER", "CLANG");
 bmSet(bm, "COMPILER", x64 ? "" : "i686-w64-mingw32-c++");
 bmSet(bm, "COMMON_OPTIONS", "");
 bmSet(bm, "COMMON_CPP_OPTIONS", "");
 bmSet(bm, "COMMON_C_OPTIONS", "");
 bmSet(bm, "COMMON_LINK", "");
 bmSet(bm, "COMMON_FLAGS", "");
 bmSet(bm, "DEBUG_INFO", "2");
 bmSet(bm, "DEBUG_BLITZ", "1");
 bmSet(bm, "DEBUG_LINKMODE", "0");
 bmSet(bm, "DEBUG_OPTIONS", "");
 bmSet(bm, "DEBUG_FLAGS", "");
 bmSet(bm, "DEBUG_LINK", "-Wl,--stack,20000000");
 bmSet(bm, "RELEASE_BLITZ", "1");
 bmSet(bm, "RELEASE_LINKMODE", "0");
 bmSet(bm, "RELEASE_OPTIONS", "-O3 ");
 bmSet(bm, "RELEASE_FLAGS", "");
 bmSet(bm, "RELEASE_LINK", "-Wl,--stack,20000000");
 bmSet(bm, "DEBUGGER", "gdb");
 bmSet(bm, "ALLOW_PRECOMPILED_HEADERS", "1");
 bmSet(bm, "DISABLE_BLITZ", "");

//  bmSet(bm, "LINKMODE_LOCK", "0");

 String clang = bin + "/clang";

 bins.At(0) = clang + "/bin";
 bins.At(1) = clang + (x64 ? "/x86_64-w64-mingw32/bin" : "/i686-w64-mingw32/bin");
 bins.At(2) = GetExeDirFile(x64 ? "bin/SDL2/lib/x64" : "bin/SDL2/lib/x86");
 bins.At(3) = GetExeDirFile(x64 ? "bin/pgsql/x64/bin" : "bin/pgsql/x86/bin");
 bins.At(4) = GetExeDirFile(x64 ? "bin/mysql/lib64" : "bin/mysql/lib32");

 incs.At(0) = GetExeDirFile("bin/SDL2/include");
 incs.At(1) = GetExeDirFile(x64 ? "bin/pgsql/x64/include" : "bin/pgsql/x86/include");
```

```
  incs.At(2) = GetExeDirFile(x64 ? "bin/mysql/include" : "bin/mysql/include");

  libs.At(0) = GetExeDirFile(x64 ? "bin/SDL2/lib/x64" : "bin/SDL2/lib/x86");
  libs.At(1) = GetExeDirFile(x64 ? "bin/pgsql/x64/lib" : "bin/pgsql/x86/lib");
  libs.At(2) = GetExeDirFile(x64 ? "bin/mysql/lib64" : "bin/mysql/lib32");

  bm.GetAdd("PATH") = Join(bins, ";");
  bm.GetAdd("INCLUDE") = Join(incs, ";");
  bm.GetAdd("LIB") = Join(libs, ";");

  SaveVarFile(ConfigFile(method + ".bm"), bm);
  dirty = true;

  if(x64)
   default_method = Nvl(default_method, method);
 }

 enum { VS_2015, VS_2017, BT_2017, VS_2019, VSP_2019, BT_2019, VS_2022 };
 DirFinder df;

 for(int version = /*VS_2012*/VS_2022; version <= VS_2022; version++)
  for(int x64 = 0; x64 < 2; x64++) {
//   String x86method = decode(version, VS_2015, "MSVS15",
//                             VS_2017, "MSVS17", BT_2017, "MSBT17",
//                             VS_2019, "MSVS19", VSP_2019, "MSVC19P", BT_2019, "MSBT19",
//                             "MSBT");
   String x86method = decode(version, VS_2015, "MSVS15",
                             VS_2017, "MSVS17", BT_2017, "MSBT17",
                             VS_2019, "MSVS19", VSP_2019, "MSVC19P", BT_2019, "MSBT19",
                             VS_2022, "MSVS22", "MSVS22");
   String x64s = x64 ? "x64" : "";
   String method = x86method + x64s;
//   String builder = decode(version, VS_2015, "MSC15",
//                             VS_2017, "MSC17", BT_2017, "MSC17",
//                             VS_2019, "MSC19", VSP_2019, "MSC19", BT_2019, "MSC19",
//                             "MSC19"
   String builder = decode(version, VS_2015, "MSC15",
                           VS_2017, "MSC17", BT_2017, "MSC17",
                           VS_2019, "MSC19", VSP_2019, "MSC19", BT_2019, "MSC19",
                           VS_2022, "MSC22", "MSC22"
                ) + ToUpper(x64s);

  #ifdef INSTANT_TESTING
   method << "Test";
  #endif

   String vc, bin, inc, lib, kit81;
```

```
   VectorMap<String, String> bm = GetMethodVars(method);
   Vector<String> bins = Split(bm.Get("PATH", ""), ';');
   Vector<String> incs = Split(bm.Get("INCLUDE", ""), ';');
   Vector<String> libs = Split(bm.Get("LIB", ""), ';');
  #ifdef INSTANT_TESTING
   if(CheckDirs(bins, 2) && CheckDirs(incs, 4) && CheckDirs(libs, 3)) {
    if(x64)
     default_method = Nvl(default_method, x86method);

    continue;
   }
  #endif

   if(version == VS_2015)
    vc = df.Get("/microsoft visual studio 14.0/vc",
"bin/cl.exe;bin/lib.exe;bin/link.exe;bin/mspdb140.dll");
   else
//    vc = df.Get(decode(version, BT_2017, "/microsoft visual studio/2017/buildtools/vc/tools/msvc",
//                               VS_2017, "/microsoft visual studio/2017/community/vc/tools/msvc",
//                               BT_2019, "/microsoft visual studio/2019/buildtools/vc/tools/msvc",
//                               VS_2019, "/microsoft visual studio/2019/community/vc/tools/msvc",
//                               VSP_2019, "/microsoft visual studio/2019/professional/vc/tools/msvc",
//                               ""),
    vc = df.Get(decode(version, BT_2017, "/microsoft visual studio/2017/buildtools/vc/tools/msvc",
                               VS_2017, "/microsoft visual studio/2017/community/vc/tools/msvc",
                               BT_2019, "/microsoft visual studio/2019/buildtools/vc/tools/msvc",
                               VS_2019, "/microsoft visual studio/2019/community/vc/tools/msvc",
                               VSP_2019, "/microsoft visual studio/2019/professional/vc/tools/msvc",
                               VS_2022, "/Microsoft Visual Studio/2022/Community/VC/Tools/MSVC",
                               //VS_2022, "/microsoft visual studio/2022/community/vc/tools/msvc",
                               ""),
               x64 ? "bin/hostx64/x64/cl.exe;bin/hostx64/x64/mspdb140.dll"
                   : "bin/hostx86/x86/cl.exe;bin/hostx86/x86/mspdb140.dll");

   bin = df.Get("/windows kits/10/bin", "x86/makecat.exe;x86/accevent.exe");
   inc = df.Get("/windows kits/10", "um/adhoc.h");
   lib = df.Get("/windows kits/10", "um/x86/kernel32.lib");
```

for unknown reason, vc is empty even when I am sure I have vs community 2022 installed in default location.

Also, can we make it smarter? the directory structure is quite stable across versions: they seems to differ by the year number. Ideally we can assume future version will not break the tradition until it does. That way we don't need to modify source code every time VS has a new version.

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Lance on Sun, 19 Dec 2021 14:54:49 GMT
View Forum Message <> Reply to Message

Removed wide picture that's disturbing.

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Mon, 20 Dec 2021 08:08:37 GMT
View Forum Message <> Reply to Message

mirek wrote on Sun, 19 December 2021 16:16Sure. Why do not you try yourself? I would gladly merged a patch...

It should be really easy, it might even be as easy as adding single line around ide/InstantSetup.cpp:237 (with new enum constant for 2022)

Mirek

Hi,

Yes, why not... now that you pointed out the correct spot within 400k lines of code.

Best regards,

Tom

File Attachments
1) SupportForMSBT22.7z, downloaded 282 times

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Mon, 20 Dec 2021 08:26:45 GMT
View Forum Message <> Reply to Message

Lance wrote on Sun, 19 December 2021 16:54Pasted code is a bit messy. here is screenshots of changes I made. Not that I think what I have done will be helpful, I am just curious why it doesn't work.

Lance,

I had to tamper with the builder too to add MSC22 and MSC22X64 build methods. Please merge the files I posted above, add your VS_2022 code just ahead of BT_2022 and see if it works for VS.

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by mirek on Mon, 20 Dec 2021 08:40:34 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Mon, 20 December 2021 09:08mirek wrote on Sun, 19 December 2021 16:16Sure. Why do not you try yourself? I would gladly merged a patch...

It should be really easy, it might even be as easy as adding single line around ide/InstantSetup.cpp:237 (with new enum constant for 2022)

Mirek

Hi,

Yes, why not... now that you pointed out the correct spot within 400k lines of code.

Best regards,

Tom

Most appreciated. Applying.

Is there a reason to exclude BT_2019 here:

ide/InstantSetup.cpp:357

```
bmSet(bm, "DEBUGGER", findarg(version, BT_2017, BT_2019, BT_2022) >= 0 ? String()
```

?

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Mon, 20 Dec 2021 09:17:53 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 20 December 2021 10:40Tom1 wrote on Mon, 20 December 2021 09:08mirek wrote on Sun, 19 December 2021 16:16Sure. Why do not you try yourself? I would gladly merged a patch...

It should be really easy, it might even be as easy as adding single line around ide/InstantSetup.cpp:237 (with new enum constant for 2022)

Mirek

Hi,

Yes, why not... now that you pointed out the correct spot within 400k lines of code.

Best regards,

Tom

Most appreciated. Applying.

Is there a reason to exclude BT_2019 here:

ide/InstantSetup.cpp:357

```
    bmSet(bm, "DEBUGGER", findarg(version, BT_2017, BT_2019, BT_2022) >= 0 ? String()
```

?

No reason. I have just misunderstood the findarg. Please fix my mistakes...

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Lance on Mon, 20 Dec 2021 12:35:56 GMT
View Forum Message <> Reply to Message

Hi Tom:

I could not get it work. But it's good to notice that MscBuilder.cpp also need to be revised.

Following is the lines that I have changed.

```
 enum { VS_2015, VS_2017, BT_2017, VS_2019, VSP_2019, BT_2019, VS_2022, BT_2022 };
 DirFinder df;

 for(int version = VS_2015; version <= BT_2022; version++)
  for(int x64 = 0; x64 < 2; x64++) {
   String x86method = decode(version, VS_2015, "MSVS15",
                                 VS_2017, "MSVS17", BT_2017, "MSBT17",
                                 VS_2019, "MSVS19", VSP_2019, "MSVC19P",
                                 BT_2019, "MSBT19",
                                 VS_2022, "MSVS22",
```

```
                                BT_2022, "MSBT22",
                                "MSBT");
   String x64s = x64 ? "x64" : "";
   String method = x86method + x64s;
   String builder = decode(version, VS_2015, "MSC15",
                                VS_2017, "MSC17", BT_2017, "MSC17",
                                VS_2019, "MSC19", VSP_2019, "MSC19", BT_2019, "MSC19",
                                VS_2022, "MSC22",
                                BT_2022, "MSC22",
                                "MSC22"
                    ) + ToUpper(x64s);
.....

   if(version == VS_2015)
    vc = df.Get("/microsoft visual studio 14.0/vc",
"bin/cl.exe;bin/lib.exe;bin/link.exe;bin/mspdb140.dll");
   else
    vc = df.Get(decode(version, BT_2017, "/microsoft visual studio/2017/buildtools/vc/tools/msvc",
                                VS_2017, "/microsoft visual studio/2017/community/vc/tools/msvc",
                                BT_2019, "/microsoft visual studio/2019/buildtools/vc/tools/msvc",
                                VS_2019, "/microsoft visual studio/2019/community/vc/tools/msvc",
                                VSP_2019, "/microsoft visual studio/2019/professional/vc/tools/msvc",
                                VS_2022, "/microsoft visual studio/2022/community/vc/tools/msvc",
                                BT_2022, "/microsoft visual studio/2022/buildtools/vc/tools/msvc",
                                ""),
                x64 ? "bin/hostx64/x64/cl.exe;bin/hostx64/x64/mspdb140.dll"
                    : "bin/hostx86/x86/cl.exe;bin/hostx86/x86/mspdb140.dll");
```

Not sure what I have missed.

BTW, how do you develop IDE? I have to build it and overwrite my current \upp\theide.exe with newly generated one.

I installed VS Build Tools 2022, and could not get it work either. One interesting thing I noticed is that VS community is installed under "program files" while VS build tools 2022 is installed "program files (x86)". Not that it has anything to do with the problem I am having.

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Mon, 20 Dec 2021 13:17:49 GMT
View Forum Message <> Reply to Message

Hi Lance,

I might use a parallel upp installation in e.g. c:\upp2 for tinkering with theide. (However, I felt lucky enough to do it directly on my live copy...)

Did you remember to run menu "Setup > Automatic build methods setup" after restarting theide? Nothing else really comes to mind...

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Lance on Mon, 20 Dec 2021 16:02:32 GMT
View Forum Message <> Reply to Message

Hi Tom:

I get MSBT2022 set up properly after applying your patch. Still no luck with VS 2022 community. But never mind, Build Tools is just what I need.

Thanks!

Lance

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Mon, 20 Dec 2021 16:58:00 GMT
View Forum Message <> Reply to Message

Hi Lance,

I'm glad to hear it worked out.

Please note that Mirek already merged my patch and also fixed a mistake I had made. So, just use the git head instead of my patch.

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Tue, 21 Dec 2021 08:09:12 GMT
View Forum Message <> Reply to Message

Hi,

I took a look at the VS_2022 community issue and found out that it was indeed the installation location in Program Files -- not (x86) -- that prevented VS_2022 from being found. Anyway, please find attached a patch to support VS_2022 community.

Maybe it's best that Mirek checks first if this is the right way to fix it...

Best regards,

Tom

File Attachments
1) InstantSetup.zip, downloaded 257 times

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by mirek on Tue, 21 Dec 2021 08:26:50 GMT
View Forum Message <> Reply to Message

Looks fine, applied, thank you. You have saved me a lot of time.

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Tue, 21 Dec 2021 08:44:40 GMT
View Forum Message <> Reply to Message

Well, I'm glad I could help.

BR,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Klugier on Tue, 21 Dec 2021 11:56:02 GMT
View Forum Message <> Reply to Message

Hello Tom,

Could you add support for Visual Studio Professional? The only difference is the path (professional):
VSP_2022, "/microsoft visual studio/2022/professional/vc/tools/msvc",

It should be placed after VS_2022:
VS_2019, "MSC19", VSP_2019, "MSC19", BT_2019, "MSC19",
VS_2022, "MSC22", VSP_2022, "MSC22", BT_2022, "MSC22",

Klugier

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Tue, 21 Dec 2021 12:15:04 GMT
View Forum Message <> Reply to Message

Klugier wrote on Tue, 21 December 2021 13:56Hello Tom,

Could you add support for Visual Studio Professional? The only difference is the path (professional):
VSP_2022, "/microsoft visual studio/2022/professional/vc/tools/msvc",

It should be placed after VS_2022:
VS_2019, "MSC19", VSP_2019, "MSC19", BT_2019, "MSC19",
VS_2022, "MSC22", VSP_2022, "MSC22", BT_2022, "MSC22",

Klugier
Hi Klugier,

I'm afraid there is not much I can do in this case... I don't have the VS 2022 professional. Nor do I have commit rights, so it's probably best you add the lines, test it and either commit it yourself, if you have the rights or give it to Mirek to deal with, if you don't.

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by mirek on Thu, 23 Dec 2021 07:25:56 GMT
View Forum Message <> Reply to Message

Klugier wrote on Tue, 21 December 2021 12:56Hello Tom,

Could you add support for Visual Studio Professional? The only difference is the path (professional):
VSP_2022, "/microsoft visual studio/2022/professional/vc/tools/msvc",

It should be placed after VS_2022:
VS_2019, "MSC19", VSP_2019, "MSC19", BT_2019, "MSC19",
VS_2022, "MSC22", VSP_2022, "MSC22", BT_2022, "MSC22",

Klugier

If you have it tested, why not PR?

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Klugier on Thu, 23 Dec 2021 10:42:40 GMT
View Forum Message <> Reply to Message

Hello Mirek,

Before I provide patch for VS Professional 2022, please apply Tom's changes related to community edition. Right now on master we have only build tools.

Klugier

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Thu, 23 Dec 2021 11:58:37 GMT
View Forum Message <> Reply to Message

Hi Klugier,

I thought VS_2022 (i.e. community edition support) was committed by Mirek already before, but at least when I just checked, it is there now.

Best regards,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Thu, 23 Dec 2021 12:08:23 GMT
View Forum Message <> Reply to Message

I guess this is just about what you were looking for. Please test.

BR,

Tom

File Attachments
1) InstantSetup.7z, downloaded 260 times

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Klugier on Thu, 23 Dec 2021 14:20:09 GMT
View Forum Message <> Reply to Message

Hello Tom and Mirek,

Thank you for patch, however I created it by myself The PR is available here. I also added one

fix. I had installer installed in x86 profiles directory (inside Microsoft Visual Studio directory) and my Visual Studio lands in x64 program files. In order to fix that I changed DirFinder constructor to detect both x86 and normal program files directories.

I tested with Visual Studio Professional 2022 and everything works as expected.

Klugier

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Tom1 on Thu, 23 Dec 2021 16:07:54 GMT
View Forum Message <> Reply to Message

Well, that's strange. I mean, I already added support for x64 program files before... So it sounds you have not been able to update to head before doing your additions.

BR,

Tom

Subject: Re: TheIDE support for Visual Studio Build Tools 2022?
Posted by Klugier on Thu, 23 Dec 2021 16:35:58 GMT
View Forum Message <> Reply to Message

Hello,

That's true, however your code assumes that if there is x86 directory, then ignore x64. On my setup - when I have "Microsoft Visual Studio" in both "Program Files (x86)" and "Program Files" the detection doesn't work.

The original code was:

```
 Array<FileSystemInfo::FileInfo> root = StdFileSystemInfo().Find(Null);
 for(int i = 0; i < root.GetCount(); i++) {
  if(root[i].root_style == FileSystemInfo::ROOT_FIXED) {
   int drive = *root[i].filename;
   String pf = GetProgramsFolderX86();
   pf.Set(0, drive);
   pf = AppendFileName(pf, "Microsoft Visual Studio");
   if(DirectoryExists(pf)) // <- if x86 "Microsoft Visual Studio" exists skip x64 processing (this is
wrong!). You could have Microsoft Visual Studio Installer in x86 and Visual Studio in x64.
    GatherDirs(path, pf);
   else {
    pf = GetProgramsFolder();
    pf.Set(0, drive);
```

```
    pf = AppendFileName(pf, "Microsoft Visual Studio");
    if(DirectoryExists(pf))
     GatherDirs(path, pf);

   }
  }
 }
```

The code in my PR is:

```
 Array<FileSystemInfo::FileInfo> root = StdFileSystemInfo().Find(Null);
 for(int i = 0; i < root.GetCount(); i++) {
  if(root[i].root_style == FileSystemInfo::ROOT_FIXED) {
   int drive = *root[i].filename;

   // Process both x86 and x64 directories
   String x86pf = GetVisualStudioFolder(drive, GetProgramsFolderX86());
   if(DirectoryExists(x86pf)) {
    GatherDirs(path, x86pf);
   }
   String pf = GetVisualStudioFolder(drive, GetProgramsFolder());
   if(DirectoryExists(pf)) {
    GatherDirs(path, pf);
   }
  }
 }
```

Klugier