

---

Subject: [FEATURE] PCH support  
Posted by [Shire](#) on Fri, 29 Jul 2011 09:43:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Currently, TheIDE have no support for PCH compiling. PCH technology significantly reduces partial rebuild time. I wrote small patch for MSC and GCC compilers.

Patch changes:

- remove build time bm\* macros such bmYEAR and bmSECOND
- add flag PCH and corresponding checkbox in Output mode configuration dialog
- remove old automatic PCH switch in MSC builder
- change .pdb generation behavior. Now builder generates one .pdb for MSC >= 8 and multiple .pdbs (per thread) for MSC < 8. When compiling by MSC < 8 with PCH, compilation performs in one thread and generates one .pdb file.
- PCH compilation runs in parallel with source files. When PCH becomes ready then compilation performs using PCH
- BLITZ can also use PCH
- tested with MSC (7.1, 8.0, 9SP1, 10SP1), MinGW 3.4.5, TDI MinGW 4.5.1, GCC 4.4.5, GCC 4.7.1

How to enable PCH? Enter Package organizer, select header file, add compiler flag "PCH". Now open "Output mode" dialog, and set checkbox "PCH" for this package.

UPD:

Currently PCH can be used only for C++ files.

When PCH is enabled, it will be force included in every suitable C++ file.

If you want to disable PCH inclusion, set attribute "Optimize for speed" or compiler option (like "-DNOPCH") to C++ file.

UPD2:

Updated due to changes in ide/Builders package.

Unusable manifest embedding and output binary date correction (see here) removed.

UPD3:

Updated due to changes in ide/Builders package.

### File Attachments

1) [upp.theide.pch.patch](#), downloaded 335 times

---

---

Subject: Re: [FEATURE] PCH support  
Posted by [dolik.rce](#) on Fri, 29 Jul 2011 11:15:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Shire!

One curious question: How much does it actually affect the build times? I believe that with BLITZ

minimizing the count of header inclusions, the difference should be very small. Also, if I understand the theory behind precompiled headers correctly (I never used it), it is not very useful, unless your code is designed properly. Which most of the U++ probably isn't. So it will only help you on your own packages at this moment, right?

Best regards,  
Honza

PS: What is the format of the patch? I'd recommend you to use 'svn diff' (assuming you use SVN), as it is best portable... Or just post all changed files in zip archive, it is often easier than deciphering weird diff format

---

Subject: Re: [FEATURE] PCH support  
Posted by [Shire](#) on Fri, 29 Jul 2011 12:53:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Weird diff format? It human readable! Use "patch" tool, Luke. See attachment.  
I have no access to external SVN repository here.

Build time reducing in 1.5-3 times per single C++ file depend on compiler.  
Benchmarks performed on uppsrc/ide package in full debug.  
First build was full, with all files included. For include all files in blitz batch their dates made equal.  
Second build performed after touching 4 files (Assist.cpp, AutoSetup.cpp, idewin.cpp, idebar.cpp).  
Third build (touched same files) executed only for Blitz.

Core 2 Duo 2.4GHz, 2Gb RAM, WXP Pro, MSC 10SP1

No options (compile everything)

Toggle Spoiler

Full

ide: 45 file(s) built in (1:14.14), 1647 msecs / file, duration = 79610 msecs, parallelization 100%

Partial

ide: 4 file(s) built in (0:06.98), 1745 msecs / file, duration = 7515 msecs, parallelization 100%

PCH

Toggle Spoiler

Full PCH

ide: 46 file(s) built in (0:17.82), 387 msecs / file, duration = 19250 msecs, parallelization 100%

Partial PCH

ide: 4 file(s) built in (0:01.07), 268 msecs / file, duration = 1172 msecs, parallelization 95%

Blitz

Toggle Spoiler

Full Blitz

ide: 45 file(s) built in (0:07.44), 165 msecs / file, duration = 8312 msecs, parallelization 4%

Partial Blitz-2

ide: 44 file(s) built in (0:10.70), 243 msecs / file, duration = 11437 msecs, parallelization 96%

Partial Blitz-3

ide: 4 file(s) built in (0:06.88), 1721 msec / file, duration = 7485 msec, parallelization 100%

Blitz PCH

Toggle Spoiler

Full Blitz PCH

ide: 46 file(s) built in (0:07.90), 171 msec / file, duration = 8391 msec, parallelization 48%

Partial Blitz PCH-2

ide: 44 file(s) built in (0:04.50), 102 msec / file, duration = 4672 msec, parallelization 58%

Partial Blitz PCH-3

ide: 4 file(s) built in (0:01.02), 256 msec / file, duration = 1156 msec, parallelization 94%

TDI-mingw 4.5.1

No options (compile everything)

Toggle Spoiler

Full

ide: 45 file(s) built in (1:33.89), 2086 msec / file, duration = 100406 msec, parallelization 100%

Partial

ide: 4 file(s) built in (0:11.66), 2917 msec / file, duration = 12359 msec, parallelization 95%

PCH

Toggle Spoiler

Full PCH

ide: 46 file(s) built in (0:51.11), 1111 msec / file, duration = 54797 msec, parallelization 100%

Partial PCH

ide: 4 file(s) built in (0:05.16), 1291 msec / file, duration = 5547 msec, parallelization 98%

Blitz

Toggle Spoiler

Full Blitz

ide: 45 file(s) built in (0:28.86), 641 msec / file, duration = 29438 msec, parallelization 2%

Partial Blitz-2

ide: 44 file(s) built in (0:31.03), 705 msec / file, duration = 31907 msec, parallelization 73%

Partial Blitz-3

ide: 4 file(s) built in (0:10.46), 2616 msec / file, duration = 11062 msec, parallelization 94%

Blitz PCH

Toggle Spoiler

Full Blitz PCH

ide: 46 file(s) built in (0:29.18), 634 msec / file, duration = 29672 msec, parallelization 38%

Partial Blitz PCH-2

ide: 44 file(s) built in (0:25.95), 589 msec / file, duration = 26407 msec, parallelization 54%

Partial Blitz PCH-3

ide: 4 file(s) built in (0:04.82), 1205 msec / file, duration = 5188 msec, parallelization 100%

---

Subject: Re: [FEATURE] PCH support

Posted by [Shire](#) on Fri, 29 Jul 2011 13:19:59 GMT

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

---

Quote:I believe that with BLITZ minimizing the count of header inclusions, the difference should be very small

Blitz reduces build time by pipelining all .cpp files in one, invoking compiler one time.

This is very good for library packages without changes.

But when you develop and change some few files in some few packages, blitz often make full rebuild of these packages to exclude new changed files from its batch and build all changed files independently. This takes too many time for partial build and can be accelerated by using precompiled header (and, in some cases, turn off blitz).

PCH can be useful on currently changing packages - main and dependent developing packages.

---

---

Subject: Re: [FEATURE] PCH support

Posted by [dolik.rce](#) on Fri, 29 Jul 2011 13:29:48 GMT

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

---

Shire wrote on Fri, 29 July 2011 14:53Weird diff format? It human readable! Use "patch" tool, Luke. See attachment.

I have no access to external SVN repository here.I know it is human readable, I read it I'm just not used to see the "new-style context format". Also, I use 'patch' when I want to apply a patch, but I know for sure that some devs don't, since they don't have the GNU userland on windows. Most notably Mirek, who IIRC applies the patches in TortoiseSVN, which only understands unified diff format ('diff -u').

Shire wrote on Fri, 29 July 2011 14:53Build time reducing in 1.5-3 times per single C++ file depend on compiler.

Benchmarks performed on uppsrc/ide package in full debug.

First build was full, with all files included. For include all files in blitz batch their dates made equal.

Second build performed after touching 4 files (Assist.cpp, AutoSetup.cpp, idewin.cpp, idebar.cpp).

Third build (touched same files) executed only for Blitz.

Well, those numbers seem quite interesting. I must admit that they surprised me I will give it a closer look as soon as I have a little free time, hopefully over the weekend.

Honza

---

---

Subject: Re: [FEATURE] PCH support

Posted by [Alboni](#) on Mon, 29 Jul 2013 19:20:28 GMT

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

---

TortoiseSVN is very good for seeing what has changed.

---