Subject: 2024rc
Posted by mirek on Thu, 07 Nov 2024 17:48:59 GMT
View Forum Message <> Reply to Message

https://sourceforge.net/projects/upp/files/upp/2024rc3/

Subject: Re: 2024rc3
Posted by Tom1 on Fri, 08 Nov 2024 11:24:58 GMT
View Forum Message <> Reply to Message

Thanks!

Testing...

Best regards,

Tom

Subject: Re: 2024rc3
Posted by cbpporter on Mon, 11 Nov 2024 09:08:30 GMT
View Forum Message <> Reply to Message

Tested on fresh install on another PC, and one common problem is back: CLANG marks my source file with error, with a red exlamation mark in the top elft corner, with the text "Errors".

It has no tooltip or click to tell you what the errors are.

This is extremely common and comes and goes, and is very annoying because first it has a big red exclamation mark on your file, then it kills "IntelliSense".

This time it was triggered by a code like this (and many other cases):

 String fullName = name;
 fullName << ".";

I attached the error.

File Attachments
1) clangerror.png, downloaded 617 times

186: error: use of overloaded operator '<<' is ambiguous (with operand types 'String' and 'con
String.h 488: candidate function
String.h 532: candidate function

Subject: Re: 2024rc3
Posted by mirek on Mon, 11 Nov 2024 09:15:45 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 11 November 2024 10:08Tested on fresh install on another PC, and one common problem is back: CLANG marks my source file with error, with a red exlamation mark in the top elft corner, with the text "Errors".

It has no tooltip or click to tell you what the errors are.

You should see it in scrollbar.

Quote:
This is extremely common and comes and goes, and is very annoying because first it has a big red exclamation mark on your file, then it kills "IntelliSense".

This time it was triggered by a code like this (and many other cases):

```
 String fullName = name;
 fullName << ".";
```

I attached the error.

This probably means some problem of libclang installation. A good way to check is to activate "Assist diagnostics" and then Assist / "Current file parse errors" (post it here).

Subject: Re: 2024rc3
Posted by mirek on Sun, 17 Nov 2024 15:40:38 GMT
View Forum Message <> Reply to Message

It is RC4 now...

Subject: Re: 2024rc3
Posted by cbpporter on Mon, 18 Nov 2024 12:47:28 GMT

mirek wrote on Mon, 11 November 2024 11:15cbpporter wrote on Mon, 11 November 2024 10:08Tested on fresh install on another PC, and one common problem is back: CLANG marks my source file with error, with a red exlamation mark in the top elft corner, with the text "Errors".

It has no tooltip or click to tell you what the errors are.

You should see it in scrollbar.

Quote:
This is extremely common and comes and goes, and is very annoying because first it has a big red exclamation mark on your file, then it kills "IntelliSense".

This time it was triggered by a code like this (and many other cases):

```
String fullName = name;
fullName << ".";
```


I attached the error.

This probably means some problem of libclang installation. A good way to check is to activate "Assist diagnostics" and then Assist / "Current file parse errors" (post it here).

Sorry for being late, I had to locate the non main dev PCs where this problems occurs since on my mains (where VS is not detected, unrelated) it does not.

Errors do appear in the scrollbar but there is no tooltip. Nor on the exclamation bar. But they do appear in code.

Anyway, the one main problem that appears in all the assist diagnostic logs is:

ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (35): use of undeclared identifier 'char32_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (58): use of undeclared identifier 'char16_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (59): use of undeclared identifier 'char32_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (77): use of undeclared identifier 'char16_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (78): use of undeclared identifier 'char32_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (92): unknown type name 'u16string'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (93): unknown type name

'u32string'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (102): no type named 'u16string' in namespace 'std::pmr'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (103): no type named 'u32string' in namespace 'std::pmr'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__functional\hash.h (371): use of undeclared identifier 'char16_t'

The rest seems to be related to String, so they might disappear if this one does.

This is on RC3 and RC4.

But for a more testable solution, in AddressBook.cpp, I've attached the output.

File Attachments
1) parse_errors, downloaded 155 times

Subject: Re: 2024rc3
Posted by cbpporter on Mon, 18 Nov 2024 12:50:23 GMT
View Forum Message <> Reply to Message

Also, maybe the first debug session shouldn't start with half the screen taken up by asm debug . This has been the case for years... Is ASM debugging so important for U+++ users?

File Attachments
1) asm.png, downloaded 521 times

```
7FF63DE613F0    rex.w
7FF63DE613F1    sub esp,0x258
7FF63DE613F7    mov [rsp+0x248],rcx
7FF63DE613FF    mov [rsp+0x240],rdx
7FF63DE61407    mov [rsp+0x238],r8
7FF63DE6140F    lea rcx,[rsp+0x228]
7FF63DE61417 ➪  call dword 0x7ff63de77a20
7FF63DE6141C    mov rdx,[rsp+0x248]
7FF63DE61424    lea r8,[rel 0x3e0760a8]
7FF63DE6142B    lea rcx,[rsp+0x208]
7FF63DE61433    call dword 0x7ff63de77b30
7FF63DE61438    jmp dword 0x7ff63de6143d
7FF63DE6143D    lea rcx,[rsp+0x208]
7FF63DE61445    call dword 0x7ff63de77bd0
```

Subject: Re: 2024rc3
Posted by mirek on Mon, 18 Nov 2024 13:00:59 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 18 November 2024 13:50Also, maybe the first debug session shouldn't start with half the screen taken up by asm debug . This has been the case for years... Is ASM debugging so important for U+++ users?

It is important for me...

If it has been like that for years, it can wait till the next release.

Subject: Re: 2024rc3
Posted by mirek on Mon, 18 Nov 2024 13:02:31 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 18 November 2024 13:47mirek wrote on Mon, 11 November 2024 11:15cbpporter wrote on Mon, 11 November 2024 10:08Tested on fresh install on another PC, and one common problem is back: CLANG marks my source file with error, with a red exlamation mark in the top elft corner, with the text "Errors".

It has no tooltip or click to tell you what the errors are.

You should see it in scrollbar.

Quote:
This is extremely common and comes and goes, and is very annoying because first it has a big red exclamation mark on your file, then it kills "IntelliSense".

This time it was triggered by a code like this (and many other cases):

```
 String fullName = name;
 fullName << ".";
```


I attached the error.

This probably means some problem of libclang installation. A good way to check is to activate "Assist diagnostics" and then Assist / "Current file parse errors" (post it here).

Sorry for being late, I had to locate the non main dev PCs where this problems occurs since on my mains (where VS is not detected, unrelated) it does not.

Errors do appear in the scrollbar but there is no tooltip. Nor on the exclamation bar. But they do appear in code.

Anyway, the one main problem that appears in all the assist diagnostic logs is:

ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (35): use of undeclared identifier 'char32_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (58): use of undeclared identifier 'char16_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (59): use of undeclared identifier 'char32_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (77): use of undeclared identifier 'char16_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (78): use of undeclared identifier 'char32_t'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (92): unknown type name 'u16string'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (93): unknown type name 'u32string'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (102): no type named 'u16string' in namespace 'std::pmr'
ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__fwd\string.h (103): no type named 'u32string' in namespace 'std::pmr'

ERROR: C:\Dev\upprc4\bin\clang\include\c++\v1__functional\hash.h (371): use of undeclared identifier 'char16_t'

The rest seems to be related to String, so they might disappear if this one does.

This is on RC3 and RC4.

But for a more testable solution, in AddressBook.cpp, I've attached the output.

Well, going to libclang comes with price - I do not already have control over bad installations. This is clearly one of those case where something happened to include path.

Mirek

Subject: Re: 2024rc3
Posted by cbpporter on Mon, 18 Nov 2024 17:58:14 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 18 November 2024 15:02

Well, going to libclang comes with price - I do not already have control over bad installations. This is clearly one of those case where something happened to include path.

Mirek
Hmm, but why would it be a bad install? It is a fresh U++ RC install.

I will keep an eye on it...

Subject: Re: 2024rc3
Posted by mirek on Mon, 18 Nov 2024 18:30:53 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 18 November 2024 18:58mirek wrote on Mon, 18 November 2024 15:02

Well, going to libclang comes with price - I do not already have control over bad installations. This is clearly one of those case where something happened to include path.

Mirek
Hmm, but why would it be a bad install? It is a fresh U++ RC install.

I will keep an eye on it...

Well, it is a conjecture at this point, but in the past these kinds of bugs were invariably caused by libc++ headers being different version than libclang or some similar sort of mismatch / incomatibility in linux packages. So I do not mean bad U++ install, but bad LLVM install.

Obviously if you look at errors, all starts with libclang not knowing some types it definitely should know...

Subject: Re: 2024rc3
Posted by Novo on Wed, 20 Nov 2024 19:42:59 GMT
View Forum Message <> Reply to Message

Latest src.
$ cat /etc/os-release
NAME="Alpine Linux"
ID=alpine
VERSION_ID=3.12.12
PRETTY_NAME="Alpine Linux v3.12"
HOME_URL="https://alpinelinux.org/"
BUG_REPORT_URL="https://bugs.alpinelinux.org/"

$ clang --version
Alpine clang version 10.0.0 (https://gitlab.alpinelinux.org/alpine/aports.git
7445adce501f8473efdb93b17b5eaf2f1445ed4c)
Target: x86_64-alpine-linux-musl
Thread model: posix
InstalledDir: /usr/bin

Application: CParser (.cache/upp.out/CLANG.Debug.Debug_Full.Shared/CParser)
$ ./CParser
Segmentation fault

$ gdb ./CParser
#0 0x00005555555f34c8 in std::atomic<bool>::load (this=<error reading variable: Cannot access memory at address 0x7ffff7feff0>,
    __m=<error reading variable: Cannot access memory at address 0x7ffff7fefec>)
   at /usr/bin/../lib/gcc/x86_64-alpine-linux-musl/9.3.0/../../../../include/c++/9.3.0/atomic:113
#1 0x000055555560fcbf in Upp::StaticPrimitive_<Upp::Mutex>::Get (this=0x555555779a80
<Upp::Heap::mutex>) at Mt.h:326
#2 0x00005555556476d8 in Upp::StaticMutex::Enter (this=0x555555779a80
<Upp::Heap::mutex>) at Mt.h:333
#3 0x000055555564e4f0 in Upp::EnterHeapMutex () at sheap.cpp:377
#4 0x000055555564f9a4 in Upp::CurrentHeap::CurrentHeap (this=0x7ffff7ff0f8) at
sheap.cpp:421
#5 0x000055555564e7a9 in Upp::MemoryAllocSz_ (sz=@0x7ffff7ff190: 60) at sheap.cpp:458

#6 0x00005555556534ad in Upp::MemoryAllocSz (size=@0x7ffff7ff190: 60) at heapdbg.cpp:142
#7 0x0000555555653572 in Upp::MemoryAlloc (size=60) at heapdbg.cpp:155
#8 0x0000555555652d39 in operator new (size=24) at heap.cpp:295
#9 0x00007ffff7e71440 in __cxa_thread_atexit () from /usr/lib/libstdc++.so.6
#10 0x000055555564e57b in Upp::MakeHeap () at sheap.cpp:403
#11 0x000055555564f9b0 in Upp::CurrentHeap::CurrentHeap (this=0x7ffff7ff288) at sheap.cpp:423
#12 0x000055555564e7a9 in Upp::MemoryAllocSz_ (sz=@0x7ffff7ff320: 60) at sheap.cpp:458
#13 0x00005555556534ad in Upp::MemoryAllocSz (size=@0x7ffff7ff320: 60) at heapdbg.cpp:142
#14 0x0000555555653572 in Upp::MemoryAlloc (size=60) at heapdbg.cpp:155
#15 0x0000555555652d39 in operator new (size=24) at heap.cpp:295
#16 0x00007ffff7e71440 in __cxa_thread_atexit () from /usr/lib/libstdc++.so.6
#17 0x000055555564e57b in Upp::MakeHeap () at sheap.cpp:403
#18 0x000055555564f9b0 in Upp::CurrentHeap::CurrentHeap (this=0x7ffff7ff418) at sheap.cpp:423
#19 0x000055555564e7a9 in Upp::MemoryAllocSz_ (sz=@0x7ffff7ff4b0: 60) at sheap.cpp:458
#20 0x00005555556534ad in Upp::MemoryAllocSz (size=@0x7ffff7ff4b0: 60) at heapdbg.cpp:142
#21 0x0000555555653572 in Upp::MemoryAlloc (size=60) at heapdbg.cpp:155
#22 0x0000555555652d39 in operator new (size=24) at heap.cpp:295
#23 0x00007ffff7e71440 in __cxa_thread_atexit () from /usr/lib/libstdc++.so.6
#24 0x000055555564e57b in Upp::MakeHeap () at sheap.cpp:403
#25 0x000055555564f9b0 in Upp::CurrentHeap::CurrentHeap (this=0x7ffff7ff5a8) at sheap.cpp:423
#26 0x000055555564e7a9 in Upp::MemoryAllocSz_ (sz=@0x7ffff7ff640: 60) at sheap.cpp:458
#27 0x00005555556534ad in Upp::MemoryAllocSz (size=@0x7ffff7ff640: 60) at heapdbg.cpp:142
#28 0x0000555555653572 in Upp::MemoryAlloc (size=60) at heapdbg.cpp:155
#29 0x0000555555652d39 in operator new (size=24) at heap.cpp:295
#30 0x00007ffff7e71440 in __cxa_thread_atexit () from /usr/lib/libstdc++.so.6
#31 0x000055555564e57b in Upp::MakeHeap () at sheap.cpp:403
#32 0x000055555564f9b0 in Upp::CurrentHeap::CurrentHeap (this=0x7ffff7ff738) at sheap.cpp:423
#33 0x000055555564e7a9 in Upp::MemoryAllocSz_ (sz=@0x7ffff7ff7d0: 60) at sheap.cpp:458
#34 0x00005555556534ad in Upp::MemoryAllocSz (size=@0x7ffff7ff7d0: 60) at heapdbg.cpp:142
#35 0x0000555555653572 in Upp::MemoryAlloc (size=60) at heapdbg.cpp:155
#36 0x0000555555652d39 in operator new (size=24) at heap.cpp:295
#37 0x00007ffff7e71440 in __cxa_thread_atexit () from /usr/lib/libstdc++.so.6
#38 0x000055555564e57b in Upp::MakeHeap () at sheap.cpp:403
#39 0x000055555564f9b0 in Upp::CurrentHeap::CurrentHeap (this=0x7ffff7ff8c8) at sheap.cpp:423
#40 0x000055555564e7a9 in Upp::MemoryAllocSz_ (sz=@0x7ffff7ff960: 60) at sheap.cpp:458
#41 0x00005555556534ad in Upp::MemoryAllocSz (size=@0x7ffff7ff960: 60) at heapdbg.cpp:142
#42 0x0000555555653572 in Upp::MemoryAlloc (size=60) at heapdbg.cpp:155
#43 0x0000555555652d39 in operator new (size=24) at heap.cpp:295

#44 0x00007ffff7e71440 in __cxa_thread_atexit () from /usr/lib/libstdc++.so.6

I understand that this is an old version of a compiler. But if I roll back Upp source code approximately four years back, then everything works just fine.

Subject: Re: 2024rc3
Posted by Novo on Thu, 21 Nov 2024 06:08:24 GMT
View Forum Message <> Reply to Message

Alpine Linux is musl-based.

Subject: Re: 2024rc3
Posted by mirek on Thu, 21 Nov 2024 09:31:29 GMT
View Forum Message <> Reply to Message

Looks like a problem with thread_local implementation.

I bet this is the commit: https://github.com/ultimatepp/ultimatepp/commit/c38e4ba32248 8292108a716b369fe70953add6d7

I have tried a workround, it is now commited on master, or you can just


```
Heap *MakeHeap()
{
 if(heap_closed__)
  return &Heap::aux;

 if(heap_tls__)
  return heap_tls__;

 static thread_local byte sHeap__[sizeof(Heap)]; // zero initialization is fine for us
 heap_tls__ = (Heap *)sHeap__;

 static thread_local HeapExitThreadGuard __;
 __.Used(); // "odr-used", register allocator to be shutdown at thread exit

 return heap_tls__;
}
```


Mirek

Subject: Re: 2024rc3
Posted by Novo on Thu, 21 Nov 2024 17:48:33 GMT

mirek wrote on Thu, 21 November 2024 04:31Looks like a problem with thread_local implementation.

Mirek

Everything is fine now.

Thanks!

Subject: Re: 2024rc3
Posted by Novo on Fri, 22 Nov 2024 01:07:05 GMT

Ide on Linux was crashing like crazy in "Open main package" till I compiled a Debug version and ran it with valgrind.
I've attached a valgring log file.
I guess the problem is related to reading of an outdated config file.

```
File Attachments
```
1) vg.log, downloaded 147 times

Subject: Re: 2024rc3
Posted by mirek on Fri, 22 Nov 2024 14:07:44 GMT

Novo wrote on Fri, 22 November 2024 02:07Ide on Linux was crashing like crazy in "Open main package" till I compiled a Debug version and ran it with valgrind.
I've attached a valgring log file.
I guess the problem is related to reading of an outdated config file.

Uhm, tried to fix valgring complains (master or

Image StreamRaster::LoadFileAny(const char *fn, Gate<int, int> progress)
{
 FileIn in(fn);
 if(in)
  return LoadAny(in, progress);
 return Null;
}

but nothing there indicating what kind of error is going on. Is it possible to create plain old gdb bt?

Mirek

Subject: Re: 2024rc3
Posted by mirek on Fri, 22 Nov 2024 14:51:40 GMT
View Forum Message <> Reply to Message

Also https://www.ultimatepp.org/forums/index.php?t=msg&th=122 73&prevloaded=1&&start=0

Subject: Re: 2024rc3
Posted by mirek on Sat, 23 Nov 2024 11:32:04 GMT
View Forum Message <> Reply to Message

After digging deeper, I have fixed a bunch of errors related to handling non-existent files which in theory might fix the issue. Thanks Novo! (And please test master)

Subject: Re: 2024rc3
Posted by Novo on Sat, 23 Nov 2024 19:32:33 GMT
View Forum Message <> Reply to Message

Thank you. It looks like it is still crashing. It looks like TheIde creates missing/malformed files after one successful launch.
I still was getting crashes on one machine. After I ran TheIde with valgrind TheIde stopped crashing.

I've attached a valgrind log. There is still a problem.

```
File Attachments
```
1) vg2.log, downloaded 145 times

Subject: Re: 2024rc3
Posted by mirek on Sun, 24 Nov 2024 10:09:26 GMT
View Forum Message <> Reply to Message

Novo wrote on Sat, 23 November 2024 20:32Thank you. It looks like it is still crashing. It looks like TheIde creates missing/malformed files after one successful launch.

I still was getting crashes on one machine. After I ran TheIde with valgrind TheIde stopped crashing.

I've attached a valgrind log. There is still a problem.

Thanks. I believe I have fixed valgrind problem (in master). I do not think it should have caused crashing, but hard to say.

Please check.

Mirek

Subject: Re: 2024rc3
Posted by Novo on Mon, 25 Nov 2024 04:02:05 GMT
View Forum Message <> Reply to Message

Thank you!
Everything seems to work fine.
I'm running out of machines with outdated Upp config files

Subject: Re: 2024rc3
Posted by cbpporter on Mon, 25 Nov 2024 08:36:01 GMT
View Forum Message <> Reply to Message

Well I have no new comments. Things like Editor and other annoyances could be fixed later, the purpose of the rc seems to be different.

But I do have several PCs and each shows a different combination of problems:
 - not detecting VS. While I'm working on a fix, my solution is just to ignore it and use Clang.
 - Clang not doing any assist: for now I'll consider this a me issue and schedule that PC for windows reinstall, since it is quite ancient.
 - lately my main PC, after the RC update, has started showing weird number in code navigator panel, between []. It is often something small like [21], but for some symbols is over [1000]. This in on RC4 and only one PC .

Subject: Re: 2024rc3
Posted by mirek on Mon, 25 Nov 2024 09:31:59 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 25 November 2024 09:36
 - lately my main PC, after the RC update, has started showing weird number in code navigator panel, between []. It is often something small like [21], but for some symbols is over [1000]. This in on RC4 and only one PC .

```cpp
int PaintCpp(Draw& w, const Rect& r, int kind, const String& name, const String& pretty, Color ink,
bool focuscursor, bool retval_last)
{
 int x = r.left;

 Image img = CxxIcon(kind);
 Size isz = img.GetSize();
 w.DrawImage(x + (DPI(16) - isz.cx) / 2, r.top + (r.GetHeight() - isz.cy) / 2, img);

 x += DPI(16);
 int y = r.top + (r.GetHeight() - Draw::GetStdFontCy()) / 2;

#ifdef _DEBUG
 String ks = " [" + AsString(kind) + "] ";
 w.DrawText(x, y, ks);
 x += GetTextSize(ks, StdFont()).cx;
#endif
```

That is debug feature. Compile it in release...