Subject: libclang-15 seems to produce incomplete Assist data (may be installation issue)
Posted by mr_ped on Wed, 01 Feb 2023 21:12:21 GMT
View Forum Message <> Reply to Message

Can't recall if this was already discussed, so I will post my experience:

OS: KDE Neon = linux distro based on Ubuntu 22.04 LTS

Yesterday the clang-15 packages appeared in the repository (I think Ubuntu one, not Neon specific), so I installed them all, but the default non-version package is not updated and still points to v14. (mentioning it just in case this is the real issue)

When I launched IDE, it starts using libclang-15 (somehow finds it and gives it more priority than 14), and partially-fails on the small CLI-Core source with errors like method String::IsEmpty does not exist and similar.

Digging a bit into assist dumps, there is difference how String forward declaration affects index, with libclang-15 the all-index contains:

/home/ped/upp/uppsrc/Core/String.h
25: Upp::String -> String, bases:
33: Upp::AString -> template <B> AString : public B, bases: B
42: Upp::AString::IsEmpty()const -> bool IsEmpty() const, bases:
345: Upp::String -> String, bases:

The String bases is empty upon forward declaration, and stays empty even with actual declaration. Then Assist has no idea the Astring::IsEmpty is also part of String interface.

After uninstalling clang-*-15 packages the IDE reverted to libclang14 automatically and there the dump of full index is as expected:

/home/ped/upp/uppsrc/Core/String.h
25: Upp::String -> String, bases:
33: Upp::AString -> template <B> AString : public B, bases: B
42: Upp::AString::IsEmpty()const -> bool IsEmpty() const, bases:
345: Upp::String -> String : public Moveable<String, AString<String0>>, bases: Moveable<Upp::String, AString<Upp::String0>>

Also with v15 the very first error from parsing current file is:

ERROR: /../lib/gcc/x86_64-linux-gnu/11/../../../../include/c++/11/cstdint (52): no member named 'int_fast8_t' in the global namespace

(this seems related to include paths)

With v14 the error log is empty.

So I guess it's either the mismatch of runtime config (IDE using v15 libclang but asking v14 default binaries for include paths/etc?), or v15 has regressions and can't parse Upp sources correctly, or the results are returned differently and v15 is not compatible with current integration? Or something completely different.

For the moment I will keep v15 uninstalled, don't require it for anything particular, but in case the v15 should work (like works in some rolling distro with v15) and this is just issue with my config, I may try again and maybe force it somehow to be default (v15 instead of v14), or figure out what exactly is wrong about config. If the integration is broken or the libclang itself is broken, then I can probably stay away from 15 for some time. (or figure out how to force IDE to use 14 even when 15 packages are installed)

---

Subject: Re: libclang-15 seems to produce incomplete Assist data (may be installation issue)
Posted by mirek on Sun, 05 Feb 2023 12:25:40 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Wed, 01 February 2023 22:12Can't recall if this was already discussed, so I will post my experience:

OS: KDE Neon = linux distro based on Ubuntu 22.04 LTS

Yesterday the clang-15 packages appeared in the repository (I think Ubuntu one, not Neon specific), so I installed them all, but the default non-version package is not updated and still points to v14. (mentioning it just in case this is the real issue)

When I launched IDE, it starts using libclang-15 (somehow finds it and gives it more priority than 14), and partially-fails on the small CLI-Core source with errors like method String::IsEmpty does not exist and similar.

Digging a bit into assist dumps, there is difference how String forward declaration affects index, with libclang-15 the all-index contains:

/home/ped/upp/uppsrc/Core/String.h
  25: Upp::String -> String, bases:
  33: Upp::AString -> template <B> AString : public B, bases: B
  42: Upp::AString::IsEmpty()const -> bool IsEmpty() const, bases:
  345: Upp::String -> String, bases:

The String bases is empty upon forward declaration, and stays empty even with actual declaration. Then Assist has no idea the Astring::IsEmpty is also part of String interface.

After uninstalling clang-*-15 packages the IDE reverted to libclang14 automatically and there the dump of full index is as expected:

/home/ped/upp/uppsrc/Core/String.h
25: Upp::String -> String, bases:
33: Upp::AString -> template <B> AString : public B, bases: B
42: Upp::AString::IsEmpty()const -> bool IsEmpty() const, bases:
345: Upp::String -> String : public Moveable<String, AString<String0>>, bases: Moveable<Upp::String, AString<Upp::String0>>

Also with v15 the very first error from parsing current file is:

ERROR: /../lib/gcc/x86_64-linux-gnu/11/../../../../include/c++/11/cstdint (52): no member named 'int_fast8_t' in the global namespace

(this seems related to include paths)

With v14 the error log is empty.

So I guess it's either the mismatch of runtime config (IDE using v15 libclang but asking v14 default binaries for include paths/etc?), or v15 has regressions and can't parse Upp sources correctly, or the results are returned differently and v15 is not compatible with current integration? Or something completely different.

For the moment I will keep v15 uninstalled, don't require it for anything particular, but in case the v15 should work (like works in some rolling distro with v15) and this is just issue with my config, I may try again and maybe force it somehow to be default (v15 instead of v14), or figure out what exactly is wrong about config. If the integration is broken or the libclang itself is broken, then I can probably stay away from 15 for some time. (or figure out how to force IDE to use 14 even when 15 packages are installed)

9 out of 10 cases with clang problems it is bad include path...

Note there is "Assis diagnostics" option in Setup/Assist, maybe it can tell more?