

24

What's New in Visual Studio for C++ Developers

MICHAEL PRICE & MRYAM GIRMAY



Cppcon
The C++ Conference

20
24



September 15 - 20

Meet the Speakers



Michael Price

Senior Product Manager
Visual Studio

michael.price@microsoft.com



Mryam Girmay

Productivity Product Manager
Visual Studio

mryamgirmay@microsoft.com



Welcome to CppCon 2024!

Join #visual_studio channel on CppCon Discord <https://aka.ms/cppcon/discord>

- Ask any questions
- Discuss the latest announcements



Take our survey
~~Win prizes~~
<https://aka.ms/cppcon/vs>



Mission of the C++ product team at Microsoft

Our
agenda
today



- by participating in the creation of the C++ Standards
- by investing in the Microsoft Visual C++ (MSVC) Compiler & Libraries
- by simplifying acquisition in C++ via vcpkg
- by improving the Visual Studio IDE
- by continuing to enhance the C++ experience for Visual Studio Code

Empower every C++ developer and their teams to achieve more



Visual Studio Code



Thu 09/19 – 16:15pm

What's New for Visual Studio Code

Alexandra Kemper & Sinem Akinci

```
helloworld > helloworld.cpp > main()
1 #include <iostream>
2 #include <vector>
3 #include <string>
4
5 using namespace std;
6
7 int main()
8 {
9     vector<string> msg{"C++", "World", "from", "VS Code", "and the C++ extension!"};
10    msg.insert(msg.begin(), "Hello! ");
11    for (const string &word : msg)
12    {
13        cout << word << " ";
14    }
15    cout << endl;
16 }
```

<https://aka.ms/cpp/code>

#1 most used code editor

[StackOverflow Developer Surveys]

Free, open-source code editor

Runs on Windows, macOS, and Linux

C++ IntelliSense, debug & code browsing

CMake, vcpkg and git integrations

SSH/container/WSL remote experiences

and much, much more...

10 Years of Microsoft @ CppCon



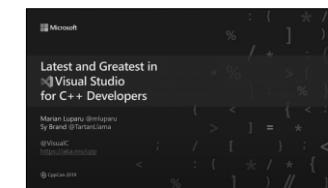
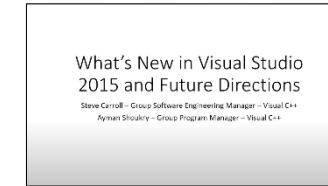
Many thanks to...

You all, the C++ community, for helping us make our products better

And CppCon organizers, staff, and volunteers for giving us so many opportunities to connect with users

<https://aka.ms/vsatcppcon>

<https://aka.ms/msatcppcon>





Visual Studio



Agenda

1. Productivity
2. Game Development
3. MSVC Toolchain
4. Debugging, Cross-Platform & Source Control

<https://aka.ms/Visual-Studio>



Visual Studio



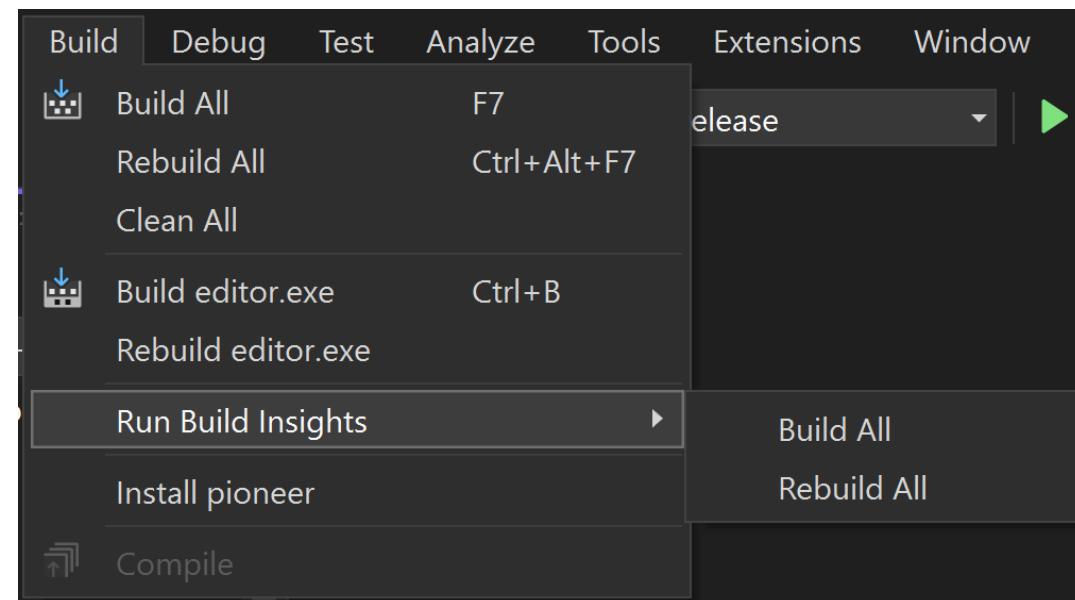
Agenda

1. Productivity
2. Game Development
3. MSVC Toolchain
4. Debugging, Cross-Platform & Source Control

Build Insights

Analyze and optimize your build

- Detailed analytics about your C++ builds
- Integrated into Visual Studio
- Visualize your include tree
- Identify “expensive” included files
- Find inlined functions that bloat your binaries



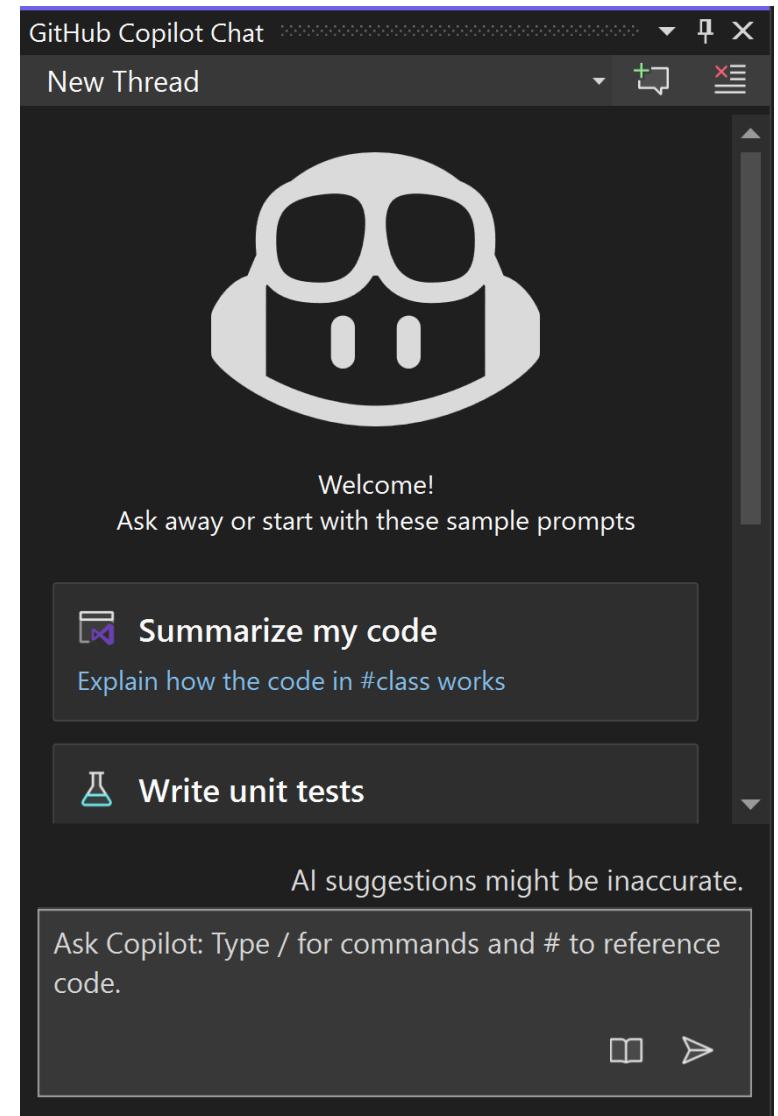
Announcing



GitHub Copilot is GA and is bundled with Visual Studio

- Activate within Visual Studio  GitHub Copilot
- Includes GitHub Copilot Chat and editor suggestions
- Enables GitHub Copilot-powered features. **Look for:** ✨
- Works with any repository, even those not hosted on GitHub

<https://aka.ms/github-copilot-features>



Improving GitHub Copilot for C++

GitHub Copilot is *more than an LLM*

Provide **useful context** from your project while enabling you to **stay in control of your content** and data



Find development tasks where GitHub Copilot is **naturally suited** to help the user and **refining that experience**

Demo





Updated Build Insights

Functions view

- Identify functions that significantly impact build times

Template view

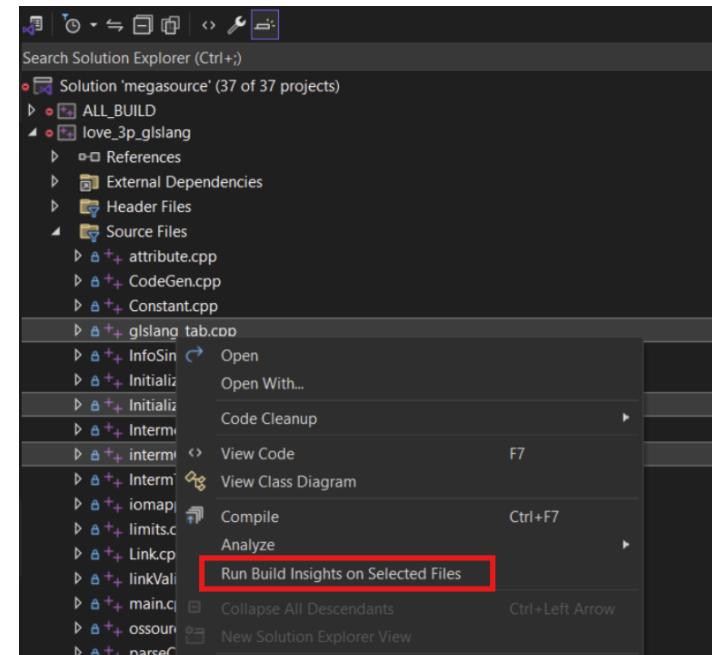
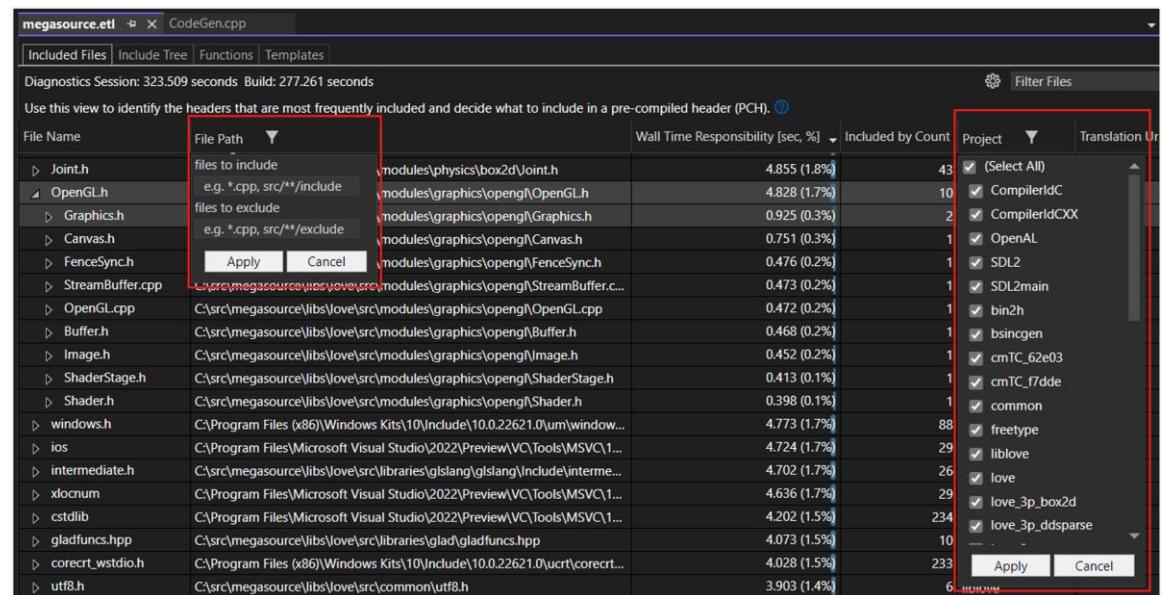
- Identify the compilation time of each template in your project

File Path and Project Filter

- Filter down your analysis to specific projects and directories.

File selection compilation

- Run Build Insights on the selected files directly from the solution explorer.
 - Analyze specific files without running Build Insights on the entire project.



Boost your Productivity

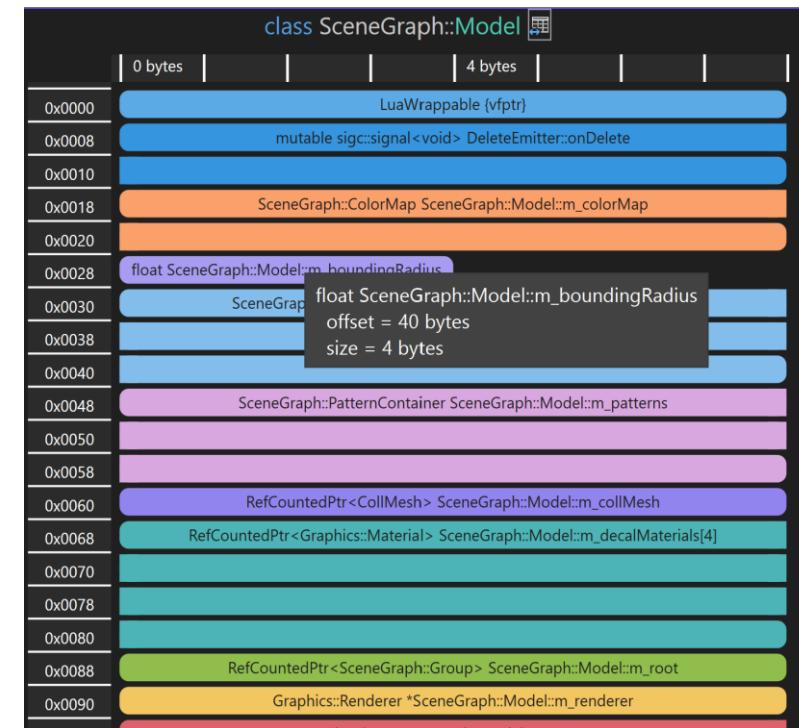
- Include Diagnostics
 - View #include references
 - View #include build time
- Memory layout
 - Visualize sizes, offsets, and padding of data members
 - Interactive Visualization

8 references | Build: 0.0570s

```
#include "GeoSphere.h"

# Sphere3D (1)
# 84 : m_atmos.reset(new Drawables::Sphere3D(renderer, 4, 1.0f, ATTRIB_POSITION));
# Draw (1)
# 85 : m_atmos->Draw(renderer, mat.Get());
# Collapse All
# include "graphics/Drawables.h"
2 references | Build: 0.0200s
```

#include "graphics/Drawables.h"



Boost your Productivity

Text formatting options

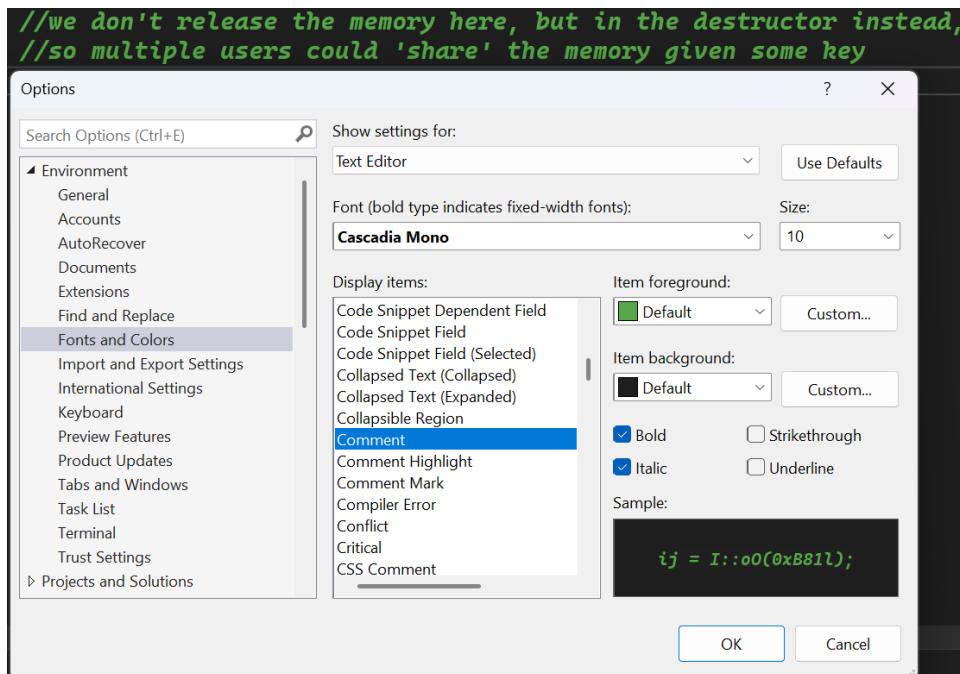
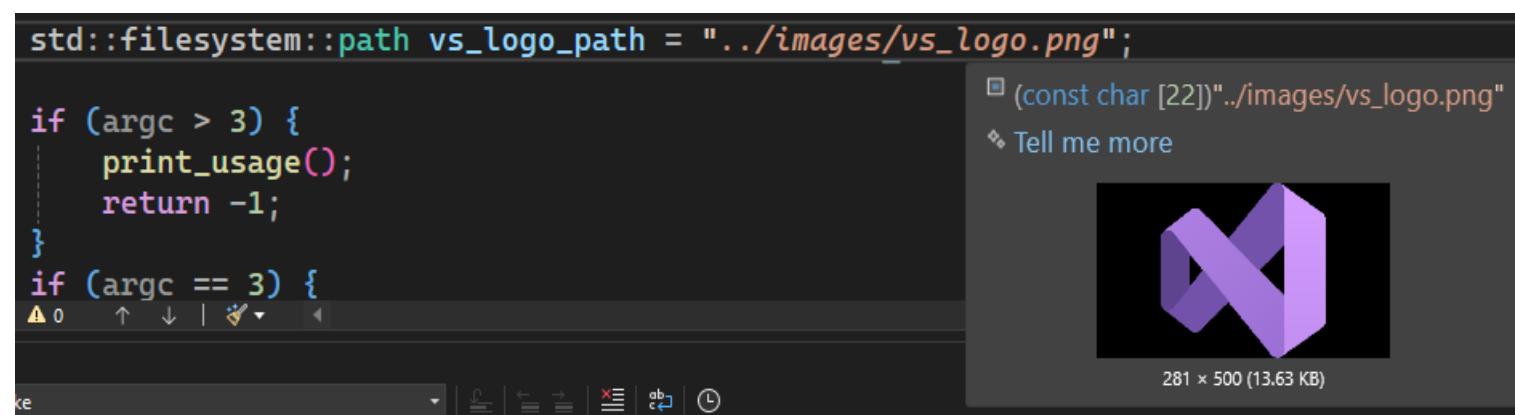
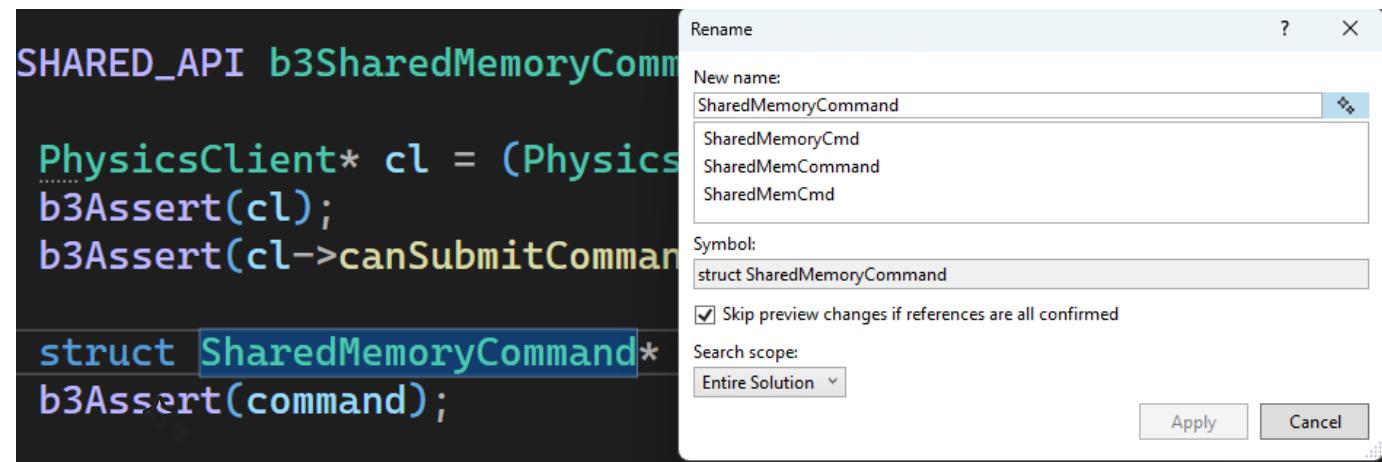


Image hover preview



Boost your Productivity

❖ AI-Powered Rename Suggestions



Case-preserving find and replace

```
void AddItem(int catalogItemId, double unitPrice)
    auto it = std::find_if(items.begin(), items.end(),
        return item.CatalogItemId == catalogItemId);
    );
```

```
void AddItem(int productId, double unitPrice)
    auto it = std::find_if(items.begin(), items.end(),
        return item.ProductId == productId);
    );
```

Announcing

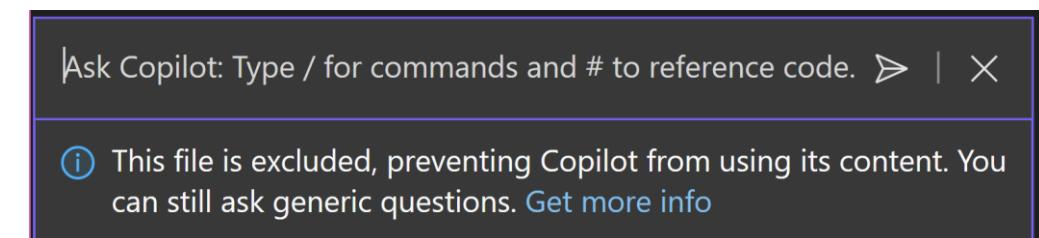
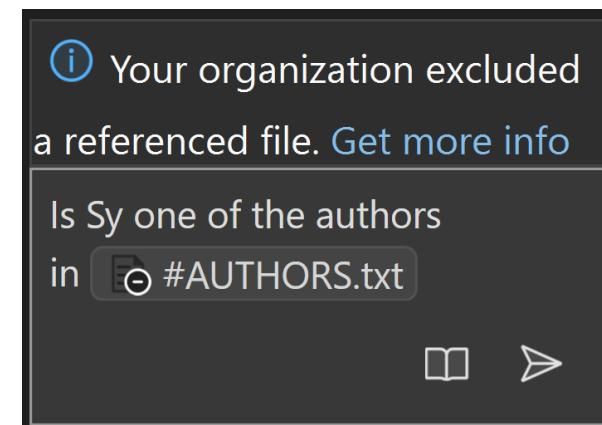
 GitHub Copilot content exclusions is in public beta

Paths to exclude in this repository:

```
1 # Exclude anything under /c  
2 - "/contrib/**"  
  
3  
4 # Exclude anything under a  
5 - "./**"  
  
6  
7 # Exclude AUTHORS.txt  
8 - "/AUTHORS.txt"
```



- Contents of excluded files are not brought in as context for any GitHub Copilot powered feature
- GitHub Copilot code completion suggestions disabled for excluded files

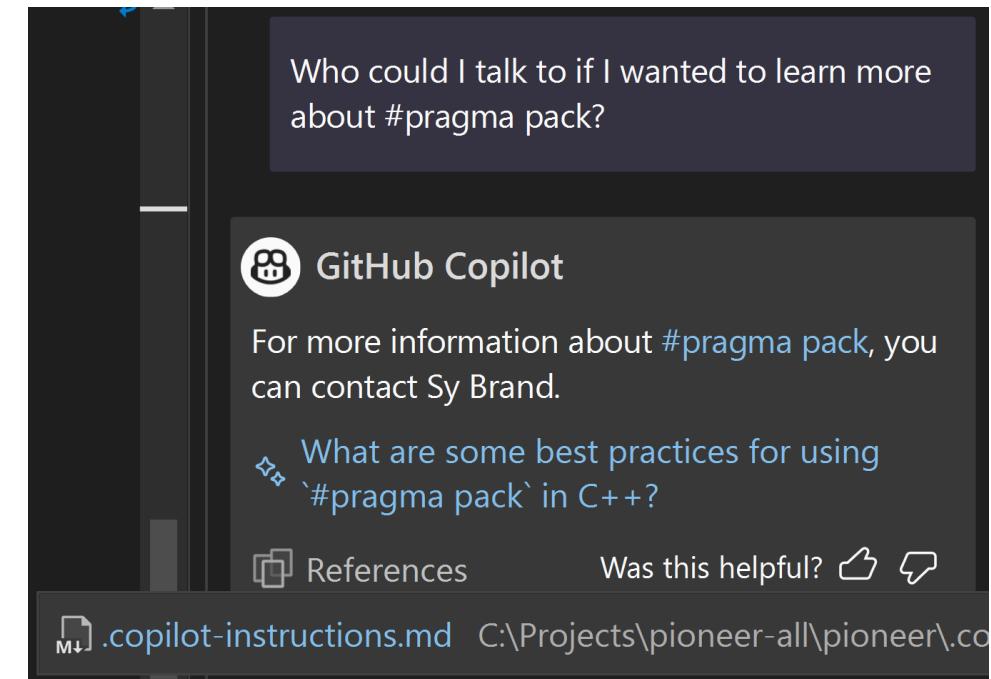


https://aka.ms/copilot/exclude_content

Experiment

GitHub Copilot custom instructions

- If the prompt asks for information regarding a specific topic, in addition to your normal response, direct them to contact the following topic owners for more information:
 - ****C++ language features**:** Sy Brand
 - ****C++ standard library**:** Michael Price
 - ****MSVC**:** Marian Luparu
 - ****Visual Studio**:** Mryam Girmay

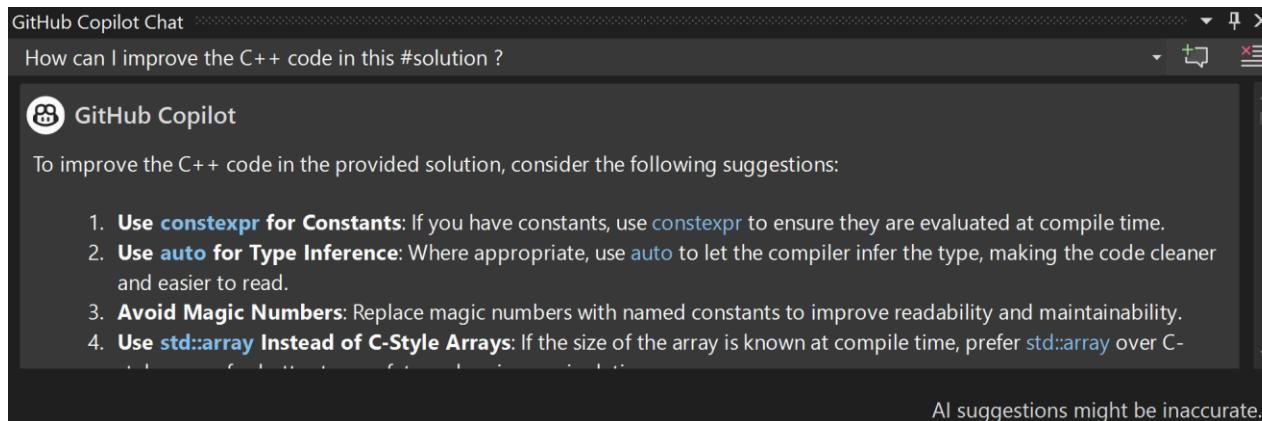


The screenshot shows the GitHub Copilot interface. A user prompt at the top asks, "Who could I talk to if I wanted to learn more about #pragma pack?" Below it, GitHub Copilot provides a response: "For more information about #pragma pack, you can contact Sy Brand." It also includes a follow-up question: "What are some best practices for using `#pragma pack` in C++?". At the bottom, there are "References" and "Was this helpful? ⤵ ⤷" buttons, and the file path ".copilot-instructions.md" is visible.

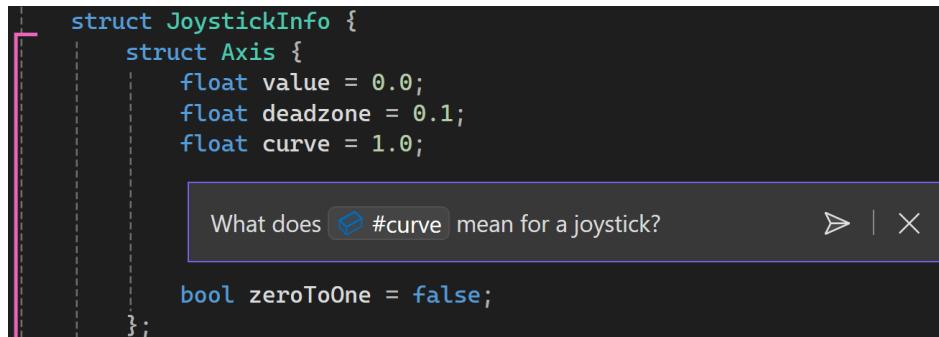
GitHub Copilot

More Later! ✨

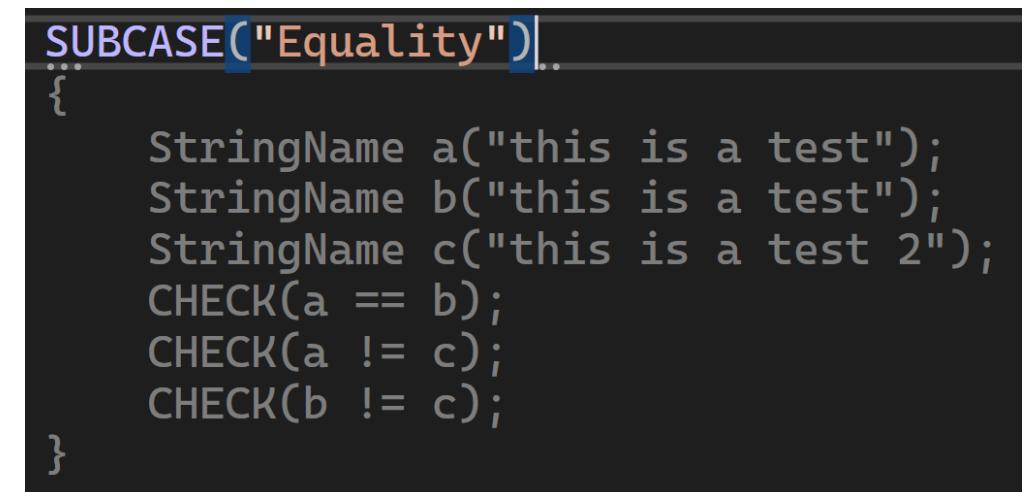
GitHub Copilot Chat Window (**Ctrl+Shift+C**) or  GitHub Copilot



Inline GitHub Copilot Chat (**Alt+T**)



GitHub Copilot Editor Suggestions (**Táč** to accept)



- Generate Copilot summary of C++ symbols
- Error List – Ask Copilot for an explanation and fix
- Slash commands - `/explain`, `/fix`, `/tests` and more



Visual Studio



Agenda

1. Productivity
2. Game Development
3. MSVC Toolchain
4. Debugging, Cross-Platform & Source Control



Unreal Engine Integrations



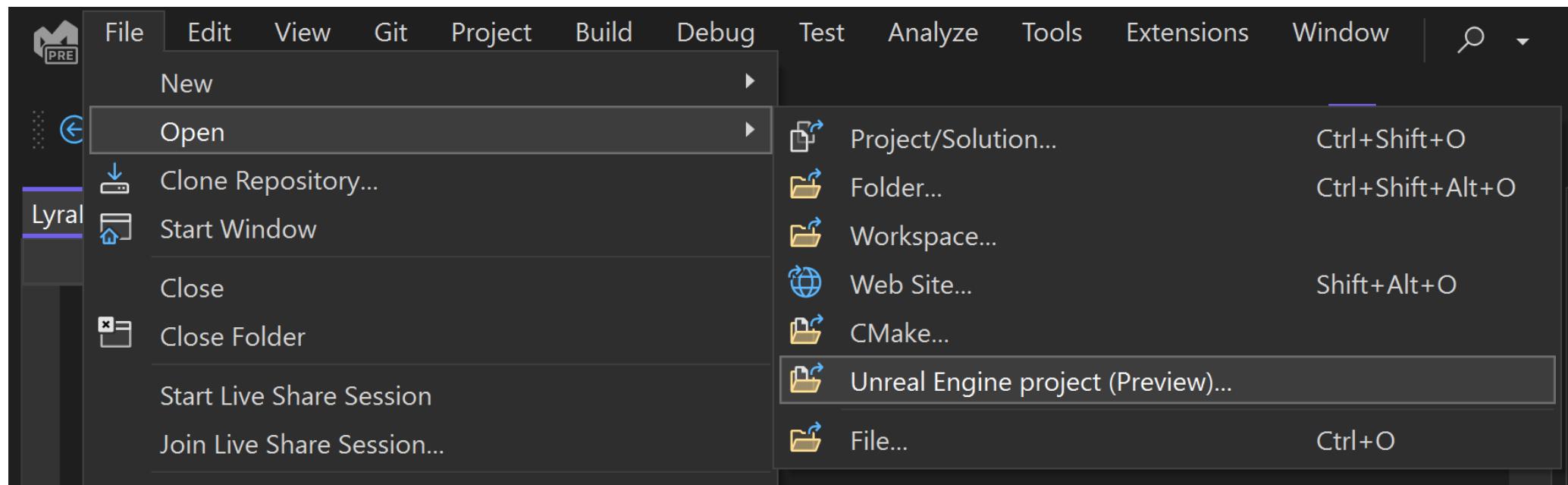
UNREAL ENGINE



Microsoft Visual Studio

Direct Support for Unreal Engine Projects

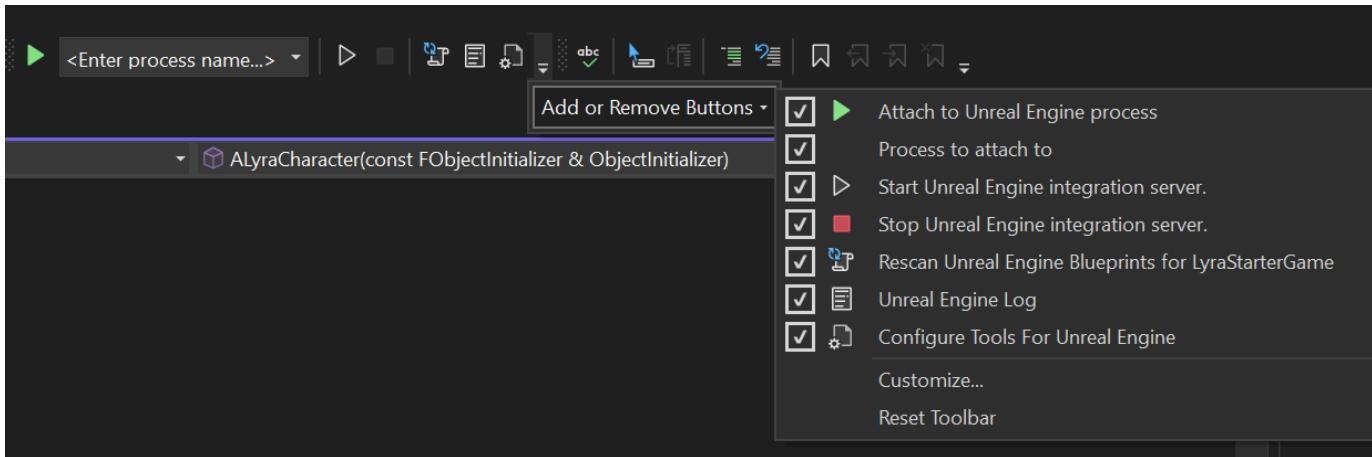
- You no longer need to generate a Visual Studio solution
- Open .uproject directly in Visual Studio
- Unreal Engine 5.4 and later works automatically
 - 5.0-5.3 requires additional patching



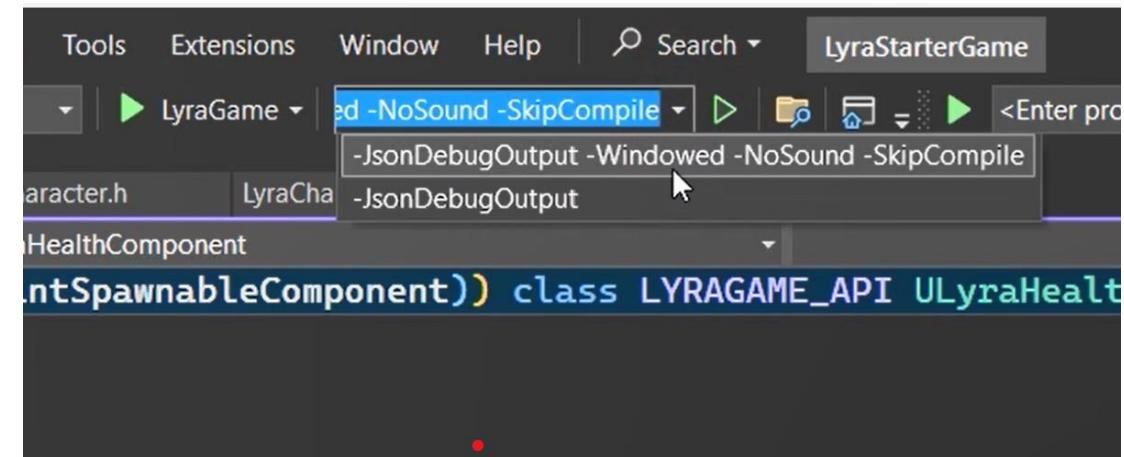
Unreal Engine Integration



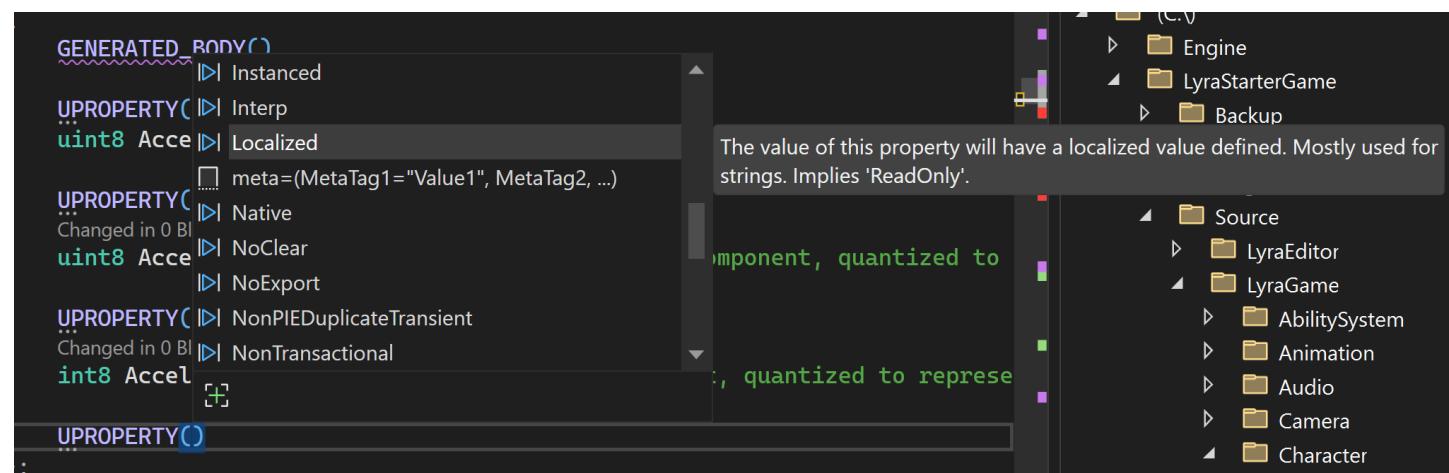
Easily Attached Unreal Engine Toolbar



Quickly Set Command Line Args



Macro Specifier Suggestions Along their description



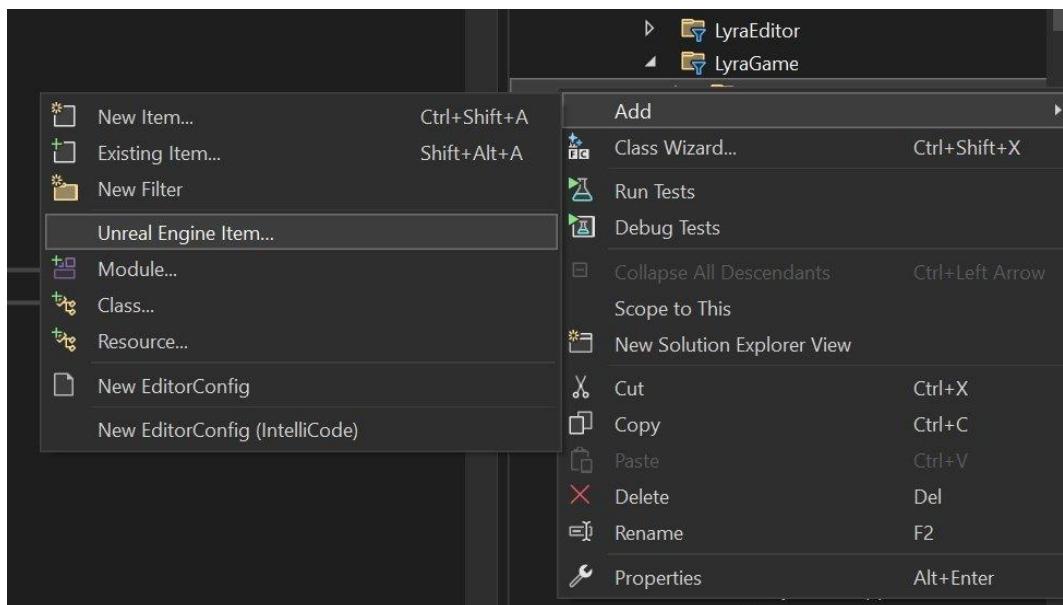
Unreal Engine Toolbar

Set Command Line Args

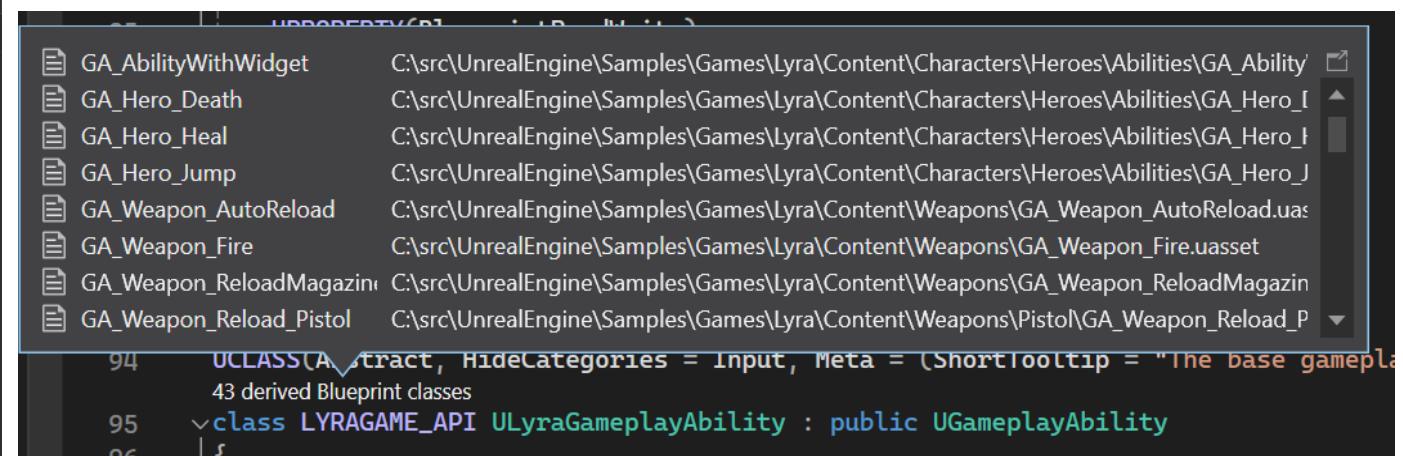
Unreal Engine Macro Specifier Suggestions

Unreal Engine Integration

Add Unreal Engine Classes, Modules, Plugins



Blueprints references works without integration plugin



Modernizing Game Development with Visual Studio and AI
David Li, Game Dev PM @ Visual Studio
Thursday, October 3, 2024
Seattle Convention Center



Visual Studio

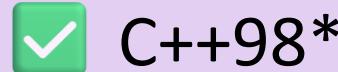


Agenda

1. Productivity
2. Game Development
3. MSVC Toolchain
4. Debugging, Cross-Platform & Source Control



Visual Studio



C++98*

* with /permissive-



C++11



C++14



C++17



C++20



C++23

with /std:c++latest



C++26

with /std:c++latest



Feature complete

- `std::optional` monadic interface
- `std::mdspan`
- `std::string::contains`
- `std::string(nullptr_t)` forbidden
- `std::format` improvements
- More `constexpr`
- Many ranges improvements



Track remaining STL work

- <https://aka.ms/STL-cpp23>



Visual Studio



C++98*

* with /permissive-

C++11

C++14

C++17

C++20

C++23

with /std:c++latest

C++26

with /std:c++latest



Feature complete

- Improving `std::format` for pointers
- `std::println()` with 0 arguments



Track remaining STL work

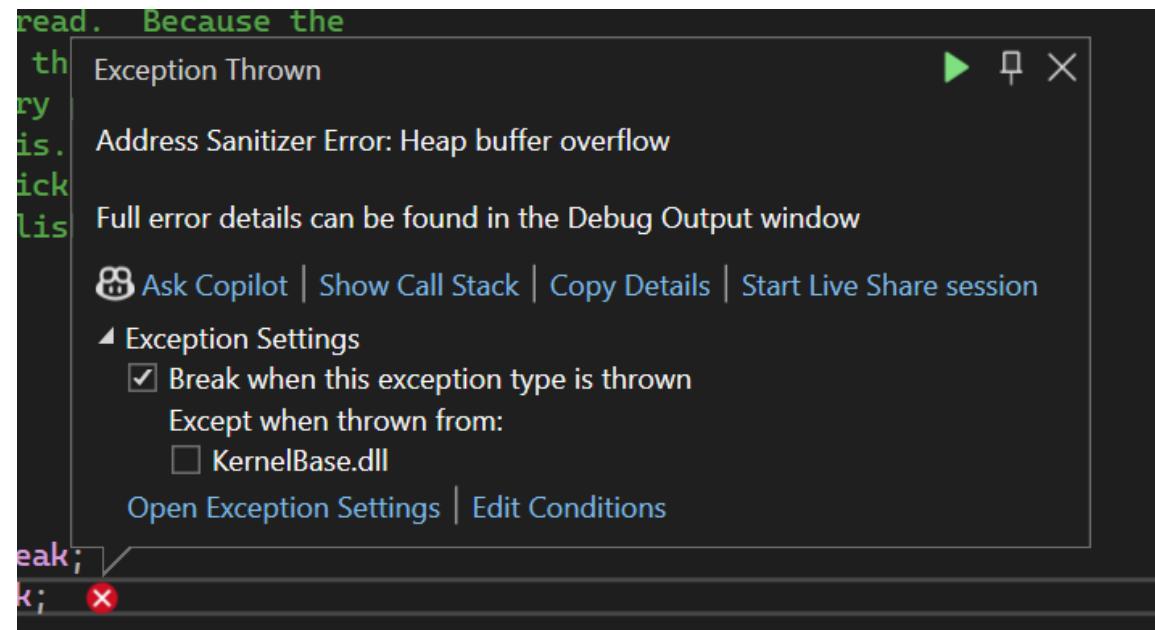
- <https://aka.ms/STL-cpp26>

Address Sanitizer

Diagnose and report hidden memory safety errors, with zero false positives, as your app runs

Used internally during testing for many products

- Windows
- Azure Storage
- Office (opt-in for individual devs)
- C++ Standard Library



Find out how to get started today at:
<https://aka.ms/Cpp-AddressSanitizer>

Backend improvements

Productivity

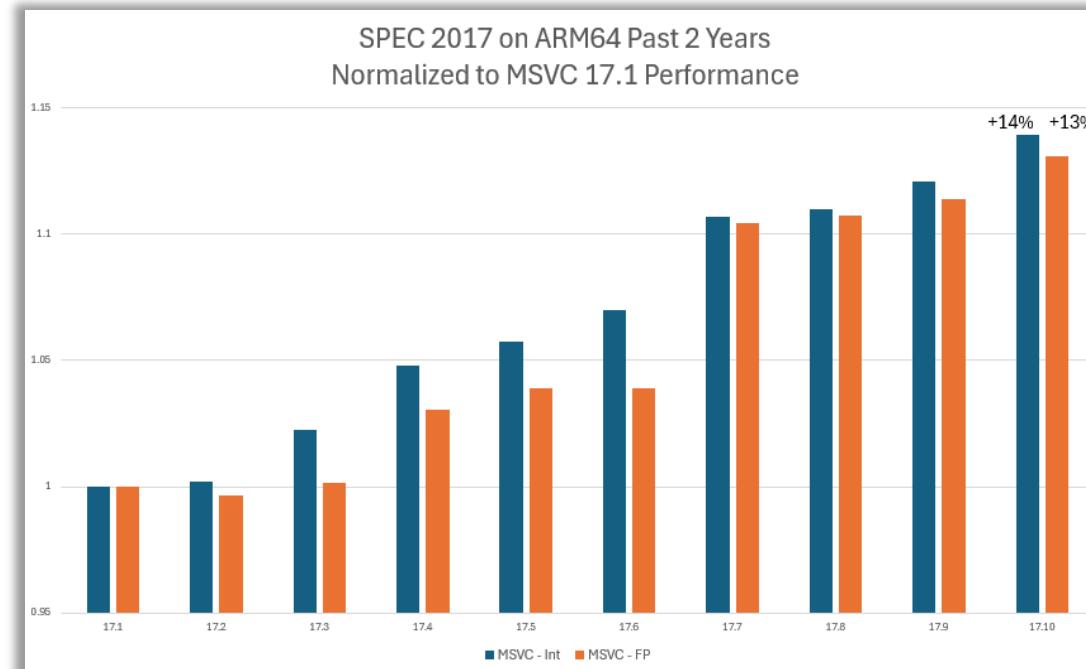
- Higher throughput for /LTCG via multiple threads
- Visibility of block-scoped variables in debugger Locals window

Performance

- New SLP vectorizer to vectorize non-loop code
 - <https://aka.ms/cpp/slp-vec-example>
- Better utilization of special HW addressing modes

Safety and Security

- Safer signed arithmetic with `__declspec(guard(overflow))`
- Address Sanitizer compatibility with C++ modules
- Fewer missed bugs (more checks for inlined calls) with Address Sanitizer



<https://aka.ms/cppblog/backend>

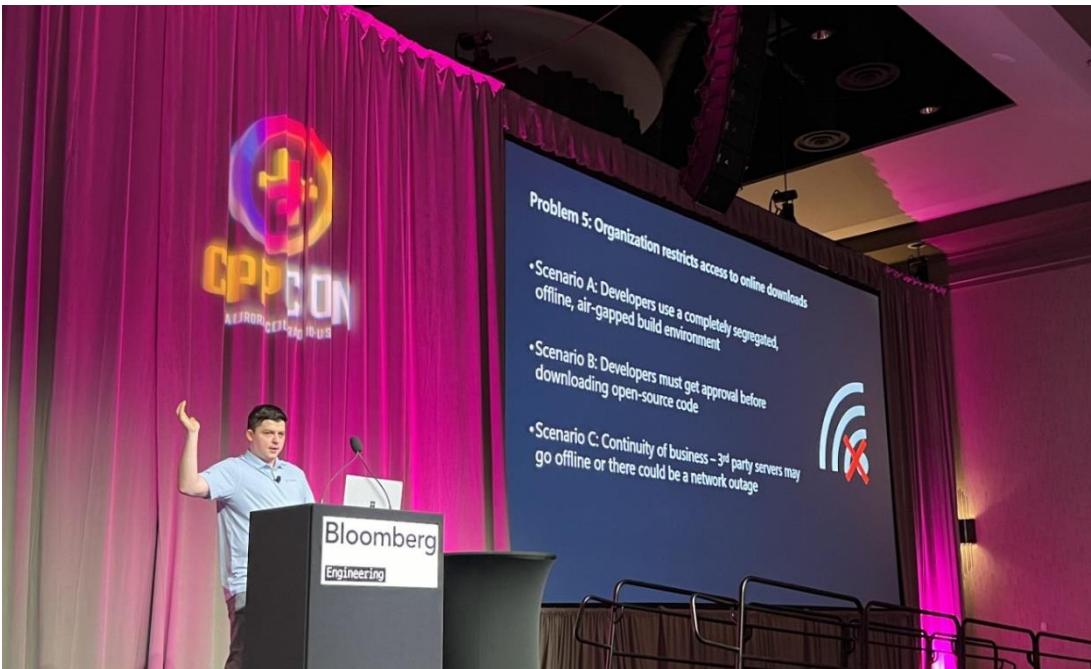
Dependency Management



Tue 09/17 – 16:45pm

10 Problems Large Companies Have with Managing C++ Dependencies and How to Solve Them

Augustin Popa



Dependency management is consistently cited as a top pain point in ISO C++ developer surveys

Learn about the challenges that large software teams have with managing C++ dependencies and how they overcame them



vcpkg for professional C++ development

Manifests

Express your dependencies declaratively and lock them down in your source control system.

Versioning

Choose your preferred package versions and lock them for reproducible builds

Registries

Create your own private library catalogs

Triplets

Target over 80 different pre-determined environments or define your own custom one

Asset Caching (experimental)

Continue operating your development environment even if the original source changes or disappears

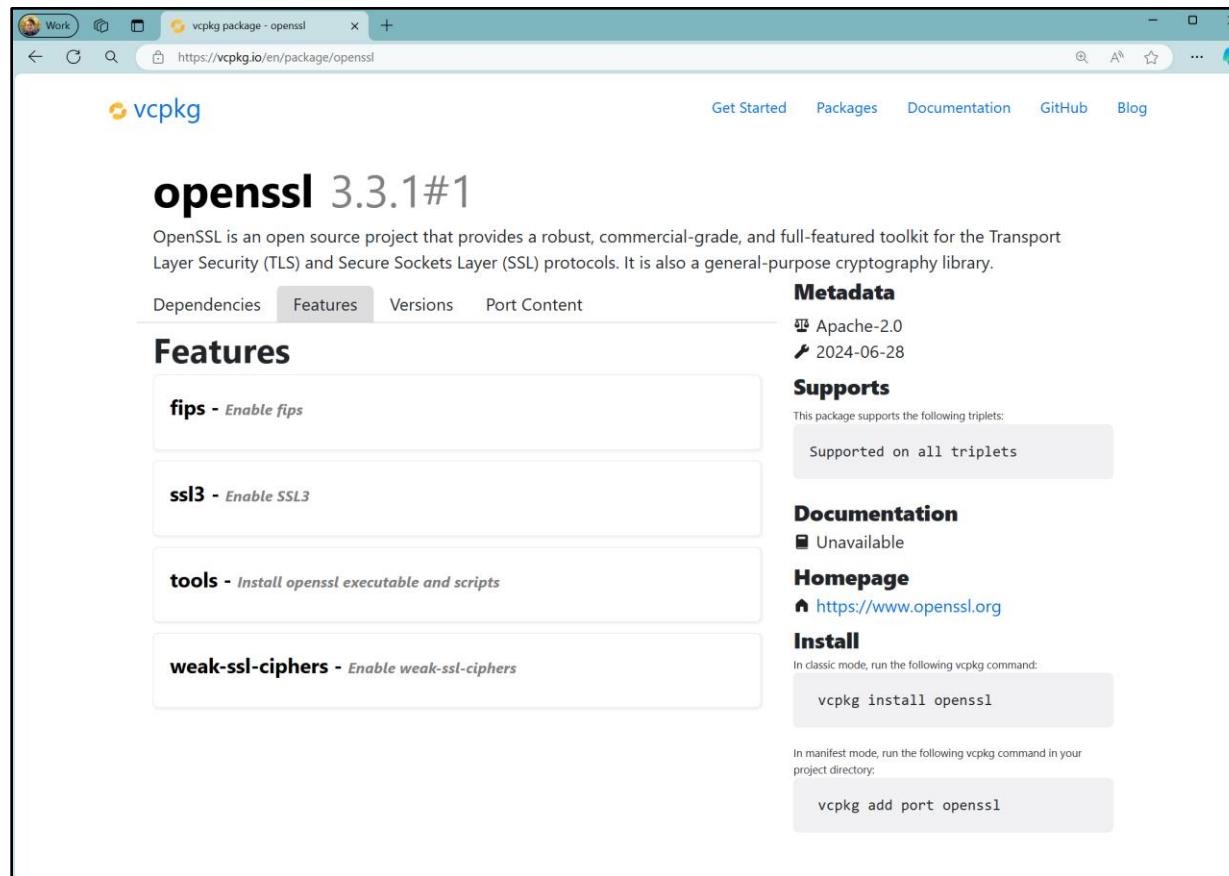
Binary Caching

Share the compiled libraries you consume locally with your development team and continuous integration system

Visit <https://aka.ms/vcpkg> for more details

Updated <https://vcpkg.io>

Browse and search from over 2300 open-source C and C++ libraries



Find transitive dependencies

Discover configurable features

Identify available port and library versions

Explore important metadata and documentation

Easy-to-copy commands to install and begin using a library



Visual Studio



Agenda

1. Productivity
2. Game Development
3. MSVC Toolchain
4. Debugging, Cross-Platform & Source Control



Cross Platform in Visual Studio

Build and debug applications on both Windows and Linux from a single IDE

CMake Debugger in Visual Studio

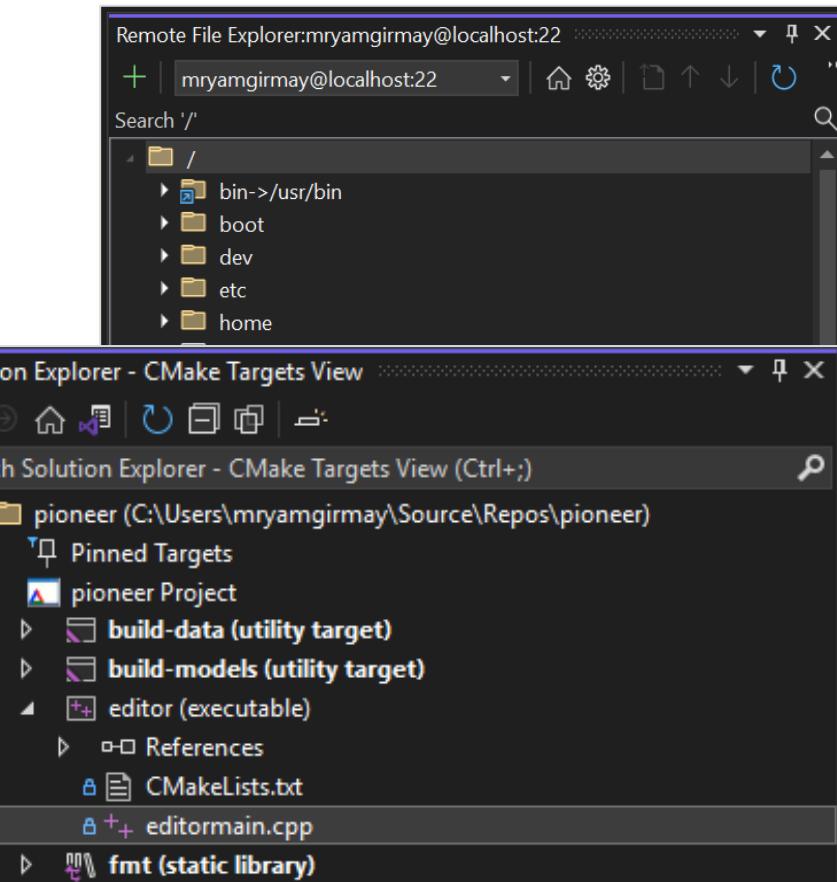
Remote File Explorer for Linux

Target View Improvements

Automatically Install WSL from Visual Studio

Debug Linux Console Apps in Integrated Terminal

<https://aka.ms/cpp-cross-platform>



WSL Installation Guide

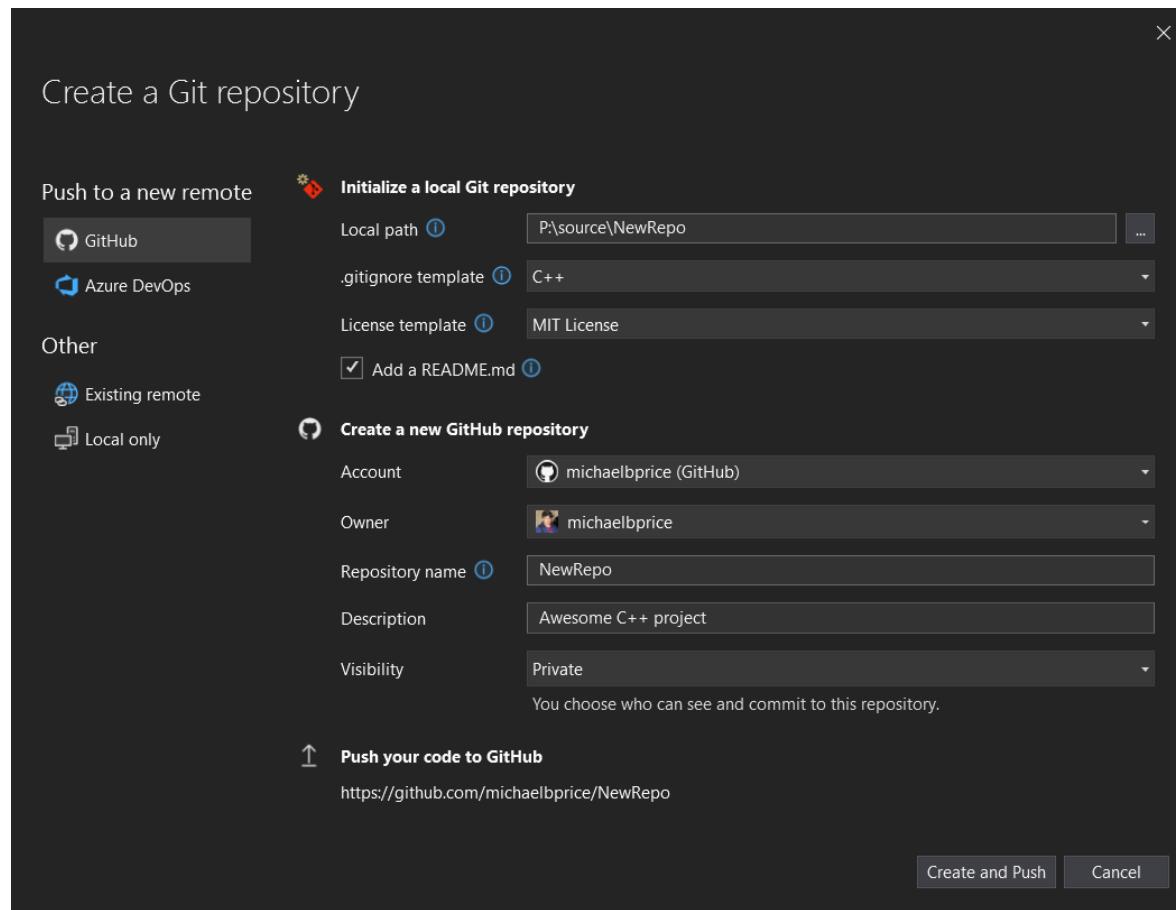
Status: Your WSL instance is ready to use. Click "Finish" to return to your project.

[Refresh System Check Values](#)

Check System Prerequisites

- Supported CPU architecture
- Supported CPU manufacturer
- Supported processor model

Source Control



Integration with **popular repository hosting** platforms

Graphical UI for source control operations

Advanced file comparison and **conflict resolution** tools

View **commit history** and annotate source with commit information

Support for **multiple branches** and remotes

<https://aka.ms/vs-collab>

Demo



Source Control + GitHub Copilot

Say goodbye to the Üřđáťjēđ ηăiňç čřř commit message



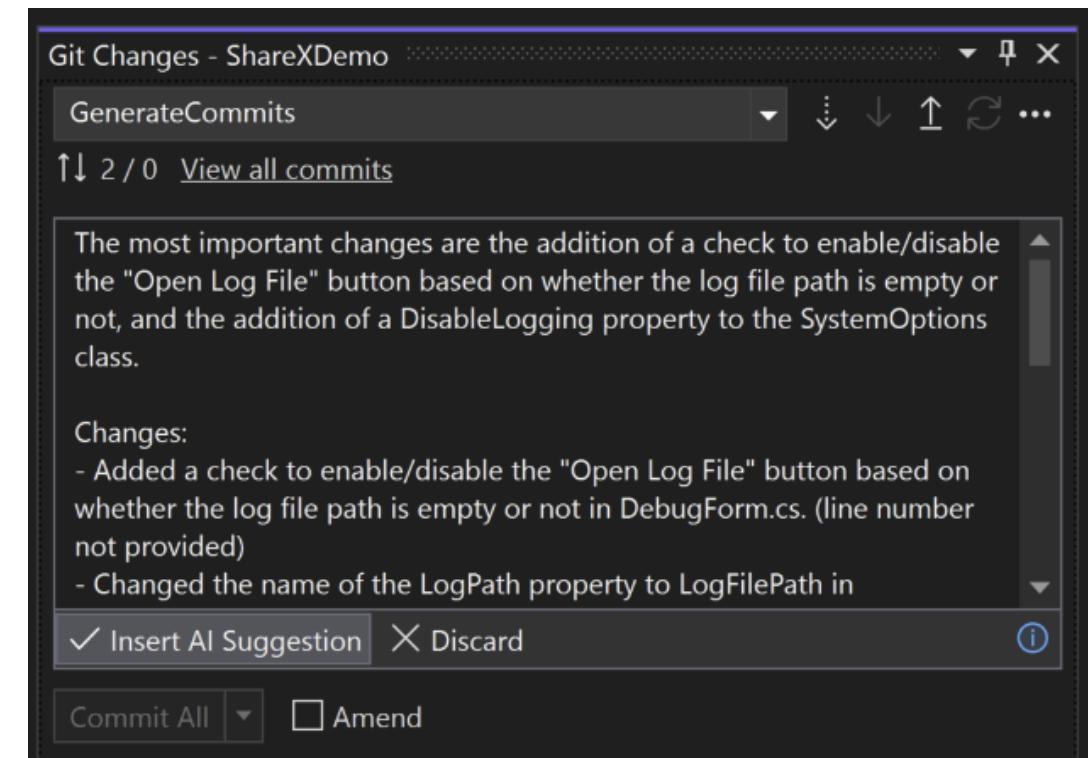
GitHub Copilot
Commit Messages



GitHub Copilot
Pull Request Descriptions



GitHub Copilot
Explain Commit



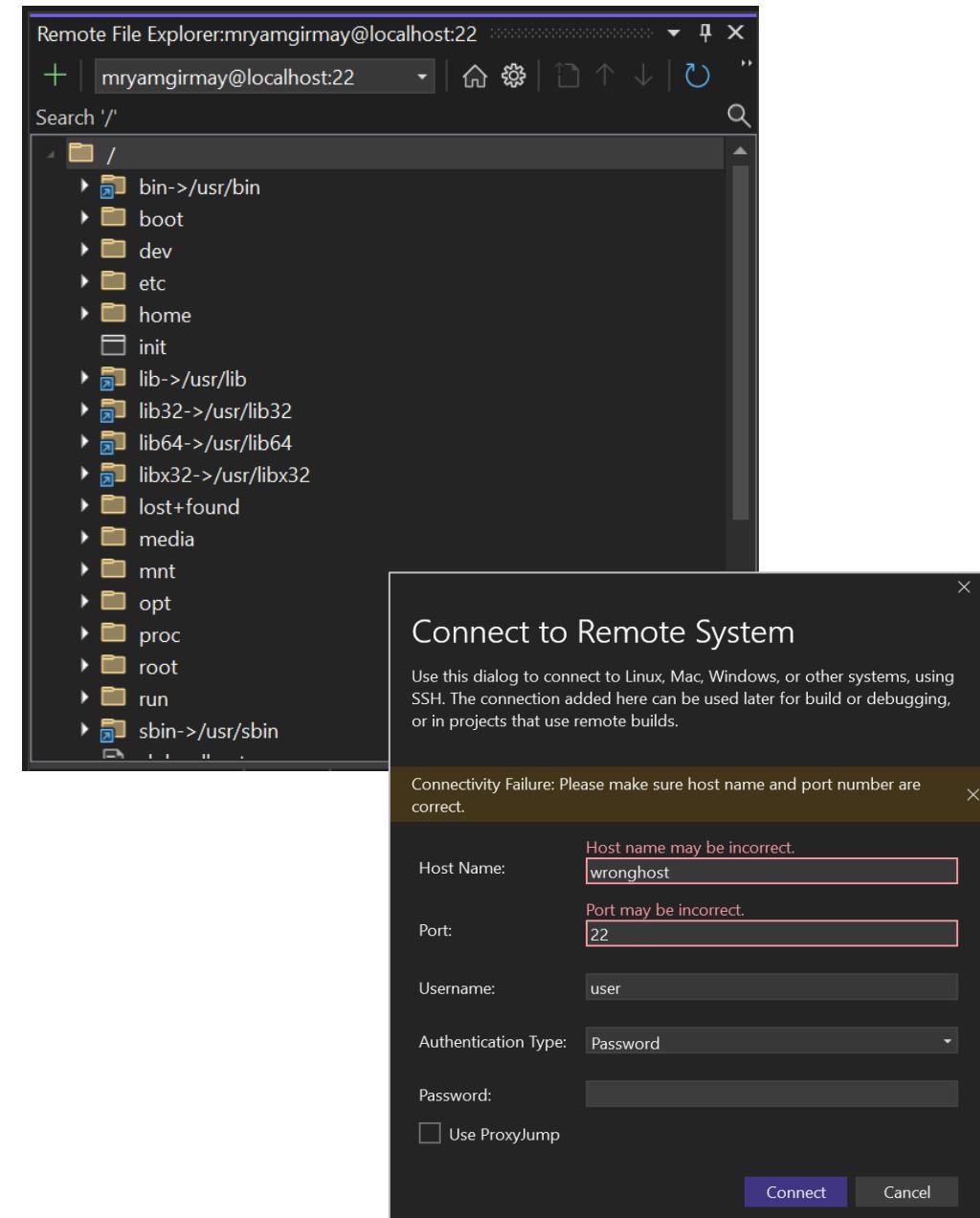
Cross-Platform

Remote file explorer improvements

- You can now search for files and folders on your remote machine
- The Remote File Explorer now supports viewing and editing files directly within Visual Studio
- Automatically detects remote file change

Connection manager improvements

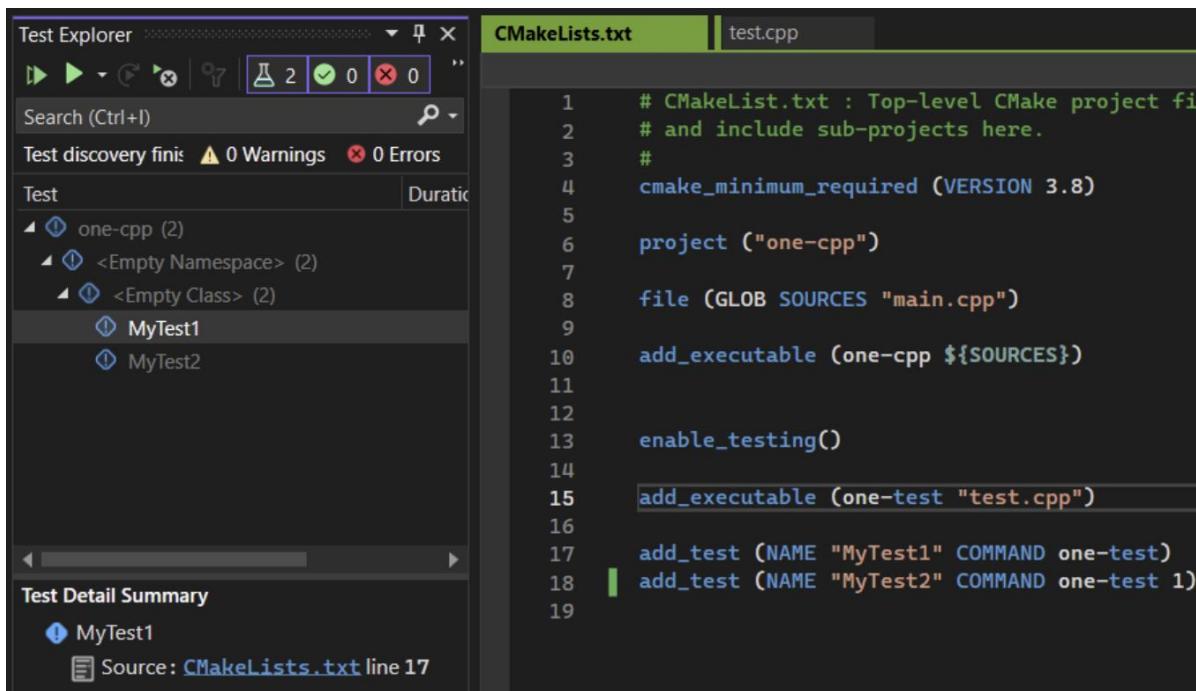
- Clearly indicate required fields
- Error messages are now displayed in an error bar



Cross-Platform

Remote Linux Testing

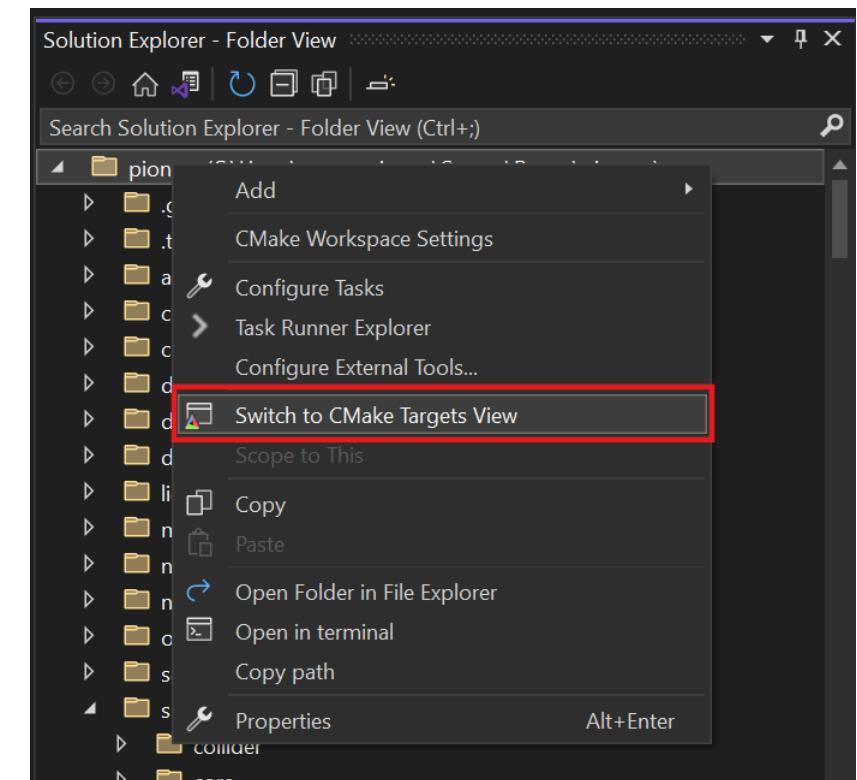
- Remote Linux users can now use the Test Explorer in Visual Studio
- Allows debugging individual tests instead of running all tests.



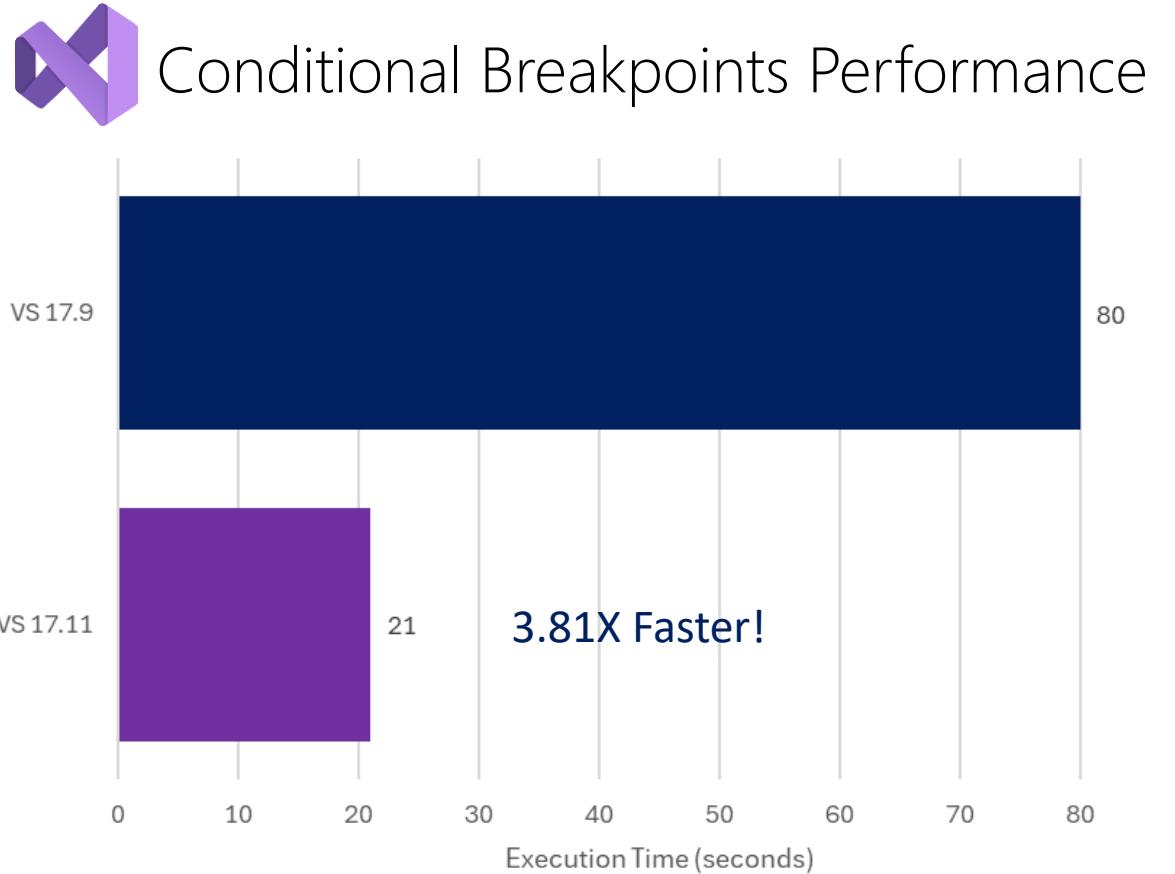
Learn more at <https://aka.ms/vslinux>

CMake Targets View Improvements

- Visualize projects by buildable targets instead of the traditional folder structure.
- Easily switch to this view



Debugging



Breakpoint groups

- Organize and simplify the management of large collections of breakpoints by grouping them.

Conditional Breakpoints with GitHub Copilot ✨

- AI-generated expressions provides expression suggestions based on your code.



Mission of the C++ product team at Microsoft

Our
agenda
today



- by participating in the creation of the C++ Standards
- by investing in the Microsoft Visual C++ (MSVC) Compiler & Libraries
- by simplifying acquisition in C++ via vcpkg
- by improving the Visual Studio IDE
- by continuing to enhance the C++ experience for Visual Studio Code

Empower every C++ developer and their teams to achieve more



Goodbye to CppCon 2024!

Join #visual_studio channel on CppCon Discord <https://aka.ms/cppcon/discord>

- Ask any questions
- Discuss the latest announcements



Take our survey
~~Win prizes~~
<https://aka.ms/cppcon/vs>



Happy Coding!