What's New in Visual Studio:

CMake Debugger, Diagnostics Improvements, Video Games, and More

DAVID LI & MRYAM GIRMAY

Cppcon
The C++ Conference

2023
October 01 - 06

Microsoft

Meet the Speakers

David Li

Game Developer Product Manager
Visual Studio
𝕏 @thecpppm

Mryam Girmay

Productivity Product Manager
Visual Studio
𝕏 @MryamGirmay

Agenda

1. Conformance & Safety
2. Performance
3. Productivity

Welcome to CppCon 2023!

Come by our booth and join #visual_studio channel on CppCon Discord https://aka.ms/cppcon/discord

- Meet the Microsoft C++ team
- Ask any questions
- Discuss the latest announcements

Take our survey

Win prizes

https://aka.ms/cppcon/vs

Microsoft

Mission of the C++ product team at Microsoft

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

1. by participating in the creation of the **C++ Standards**

Our agenda today

2. by investing in the Microsoft Visual C++ (**MSVC**) Compiler & Libraries
3. by simplifying acquisition in C++ via **vcpkg**
4. by improving the **Visual Studio IDE**

5. by continuing to enhance the C++ experience for **Visual Studio Code**

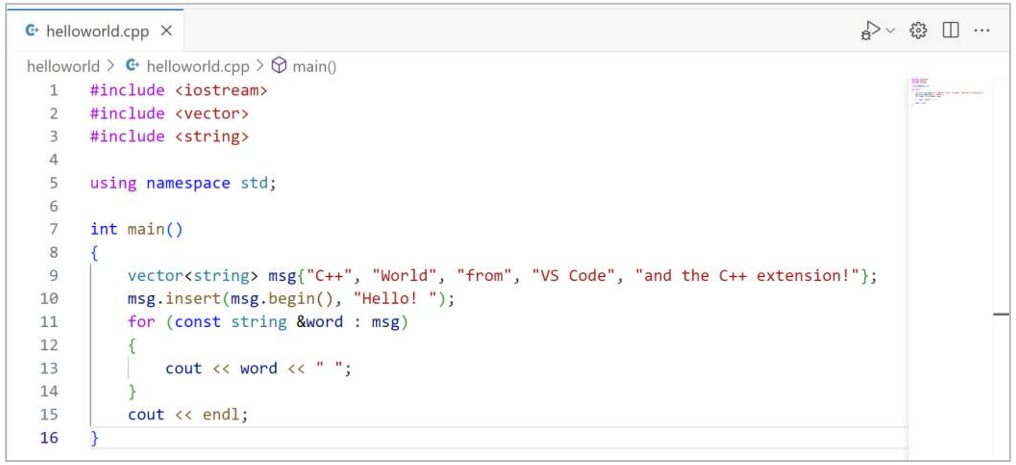
Visual Studio Code

Microsoft

Thu 10/05 – 3:15pm

How Visual Studio Code Can Help You Develop More Efficiently in C++

Alexandra Kemper & Sinem Akinci



```cpp
#include <iostream>
#include <vector>
#include <string>

using namespace std;

int main()
{
    vector<string> msg{"C++", "World", "from", "VS Code", "and the C++ extension!"};
    msg.insert(msg.begin(), "Hello! ");
    for (const string &word : msg)
    {
        cout << word << " ";
    }
    cout << endl;
}
```

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…

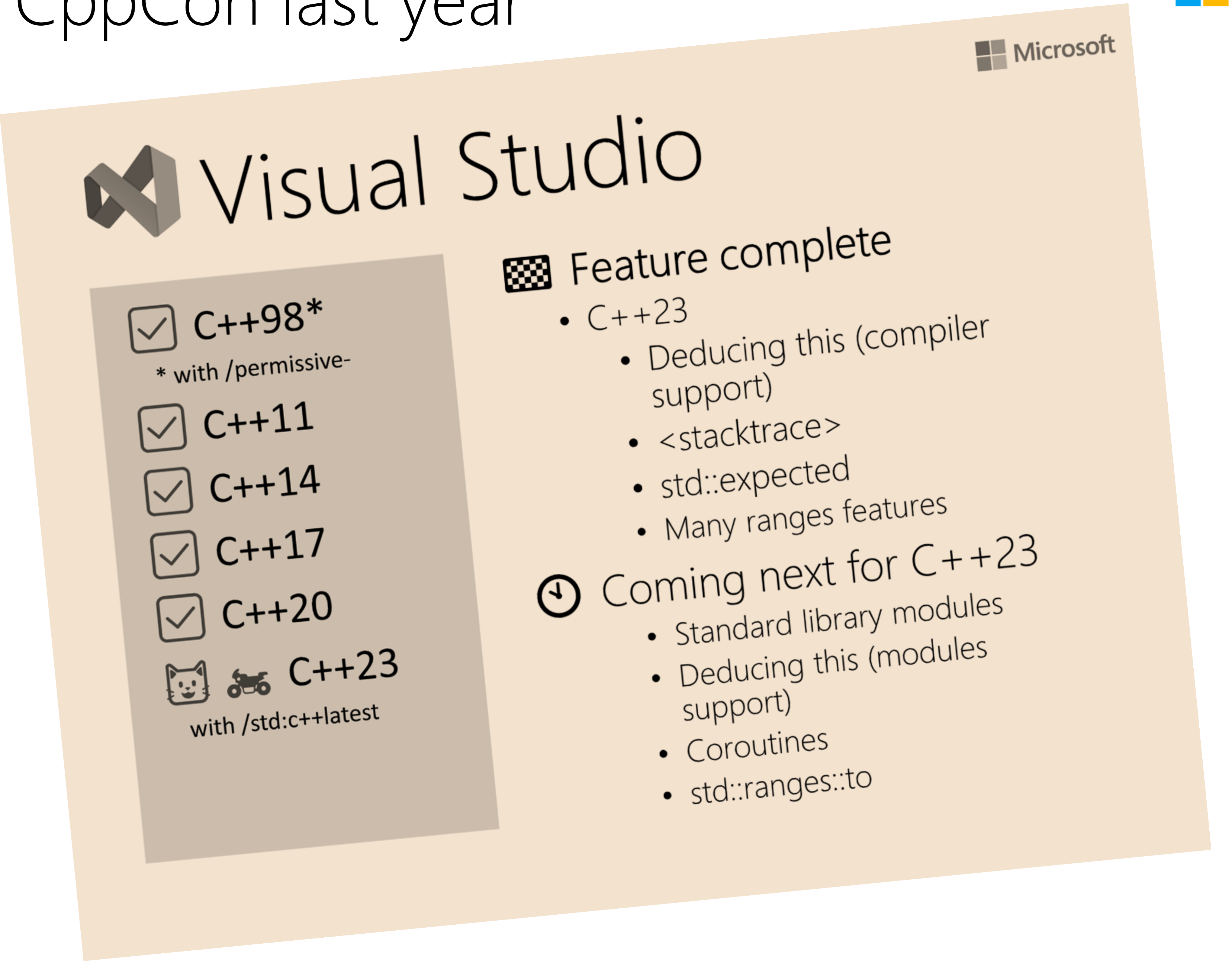
Visual Studio

Microsoft

Agenda

1. Conformance & Safety
2. Performance
3. Productivity

At CppCon last year



Visual Studio

☑ C++98*

* with /permissive-

☑ C++11

☑ C++14

☑ C++17

☑ C++20

🐱 🏍 C++23

with /std:c++latest

🏁 Feature complete

- C++23
  - Deducing this (compiler support)
  - <stacktrace>
  - std::expected
  - Many ranges features

🕘 Coming next for C++23

- Standard library modules
- Deducing this (modules support)
- Coroutines
- std::ranges::to

Visual Studio

Microsoft

☑ C++98*

* with /permissive-

☑ C++11

☑ C++14

☑ C++17

☑ C++20

😺 🏍 C++23

with /std:c++latest

🏁 Feature complete

- C++23
  - Standard library modules
  - std::print
  - Ranges:
    - to
    - zip and enumerate
    - cartesian_product

🕘 Coming next for C++23

- Deducing this (modules support)
- 9 remaining C++ STL features

Announcing

In Visual Studio 2022 version 17.6,

C++20 for C++/CLI

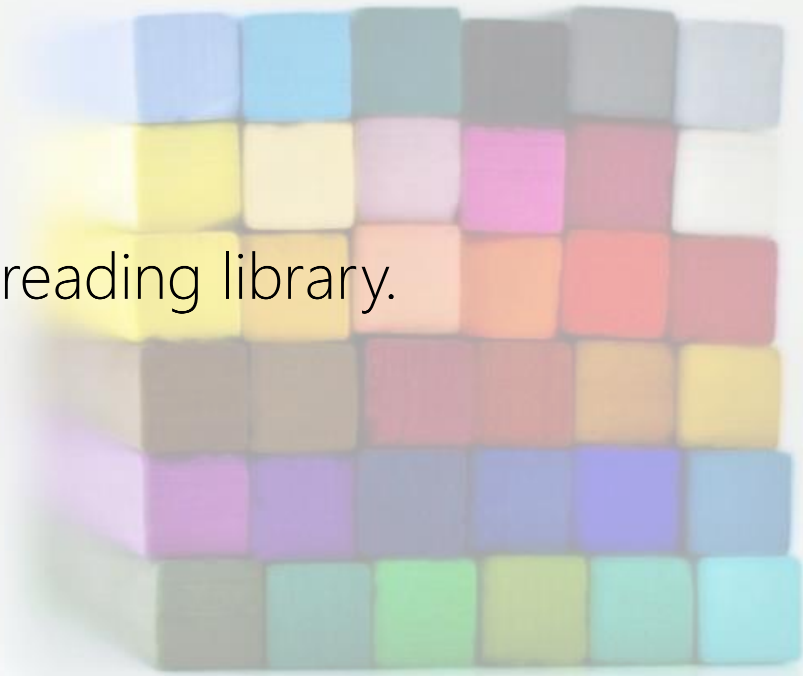
- Use /std:c++20 in your C++/CLI projects.
- Most C++20 features are supported except for support for module import under /clr.

Visit https://aka.ms/cpp20-cppcli for more details

Announcing

In Visual Studio 2022 version 17.8 Preview 2,

C11 Threads Support



- Implementation of the optional C11 threading library.
- Includes:
    - Mutexes
    - Condition variables
    - Thread-specific storage

Visit https://aka.ms/c11-threads for more details

Static Analysis

- 10+ new checks:
  - High-confidence lifetime checks.
  - Date calculations not considering leap years.
  - Evaluation order related problems.
  - And more…
- Added "**Key Events**" to more analyses to provide reasoning behind warnings.
- Performance improvements (up to 20%) and significantly decreased memory usage.

Earlier this week

Lifetime Safety in C++

Gabor Horvath

Announcing

In Visual Studio 2022 version 17.7,

Address Sanitizer continue_on_error Mode Now Available

- A new “checked build” for C and C++
- Diagnoses and reports hidden memory safety errors, with zero false positives, as your app runs
- Does not stop on first error: you can run your entire test suite collecting all memory safety issues.
- Enable developers to implement a simple new gate for shipping C++ on Windows.

Visit https://aka.ms/asan-coe for more details

Visual Studio

Microsoft

Agenda

1. Conformance & Safety
2. Performance
3. Productivity

Microsoft

Announcing

In Visual Studio 2022 version 17.4,

Native ARM64 Toolchain

- Develop for ARM64 on ARM64 with no emulation
- Includes ARM64 versions of Ninja and CMake
- Available with the C++ Desktop and Game Development Workloads

Visit https://aka.ms/ARM64-native for more details

Announcing

In Visual Studio 2022 version 17.7,

New MSVC Backend Optimizations

- Host of new backend improvements
- Both machine-independent and ARM64-specific
- ARM64 improvements cover both scalar and vector (NEON) instructions

Visit https://aka.ms/cpp/back-mi and https://aka.ms/cpp/back-arm64 for more details

```c
void absolute_difference(
           int * __restrict a, int * __restrict b,
           int * __restrict c, int n) {
    for (int i = 0; i < n; i++) {
        a[i] = (b[i] - c[i]) > 0
             ? (b[i] - c[i])
             : (c[i] - b[i]);
    }
}
```

```
; Before

movdqu    xmm0, XMMWORD PTR
[r11+rax]
add       ecx, 4
movdqu    xmm1, XMMWORD PTR [rax]
lea       rax, QWORD PTR [rax+16]
movdqa    xmm3, xmm0
psubd     xmm3, xmm1
psubd     xmm1, xmm0
movdqa    xmm2, xmm3
pcmpgtd   xmm2, xmm4
movdqa    xmm0, xmm2
andps     xmm2, xmm3
andnps    xmm0, xmm1
orps      xmm0, xmm2
movdqu    XMMWORD PTR [r10+rax-16],
xmm0
cmp       ecx, edx
jl        SHORT $LL4@s32_1
```

```
; Before

movdqu      xmm0, XMMWORD PTR [r11+rax]
add         ecx, 4
movdqu      xmm1, XMMWORD PTR [rax]
lea         rax, QWORD PTR [rax+16]
movdqa      xmm3, xmm0
psubd       xmm3, xmm1
psubd       xmm1, xmm0
movdqa      xmm2, xmm3
pcmpgtd     xmm2, xmm4
movdqa      xmm0, xmm2
andps       xmm2, xmm3
andnps      xmm0, xmm1
orps        xmm0, xmm2
movdqu      XMMWORD PTR [r10+rax-16], xmm0
cmp         ecx, edx
jl          SHORT $LL4@s32_1
```

```
; After

vmovdqu     xmm1, XMMWORD PTR [r10+rax]
vpsubd      xmm1, xmm1, XMMWORD PTR [rax]
vpabsd      xmm2, xmm1
add         edx, 4
vmovdqu     XMMWORD PTR [rbx+rax], xmm2
lea         rax, QWORD PTR [rax+16]
cmp         edx, ecx
jl          SHORT $LL4@s32_1
```

```
bool did_meow();
bool did_not_meow() {
    return !did_meow();
}
```

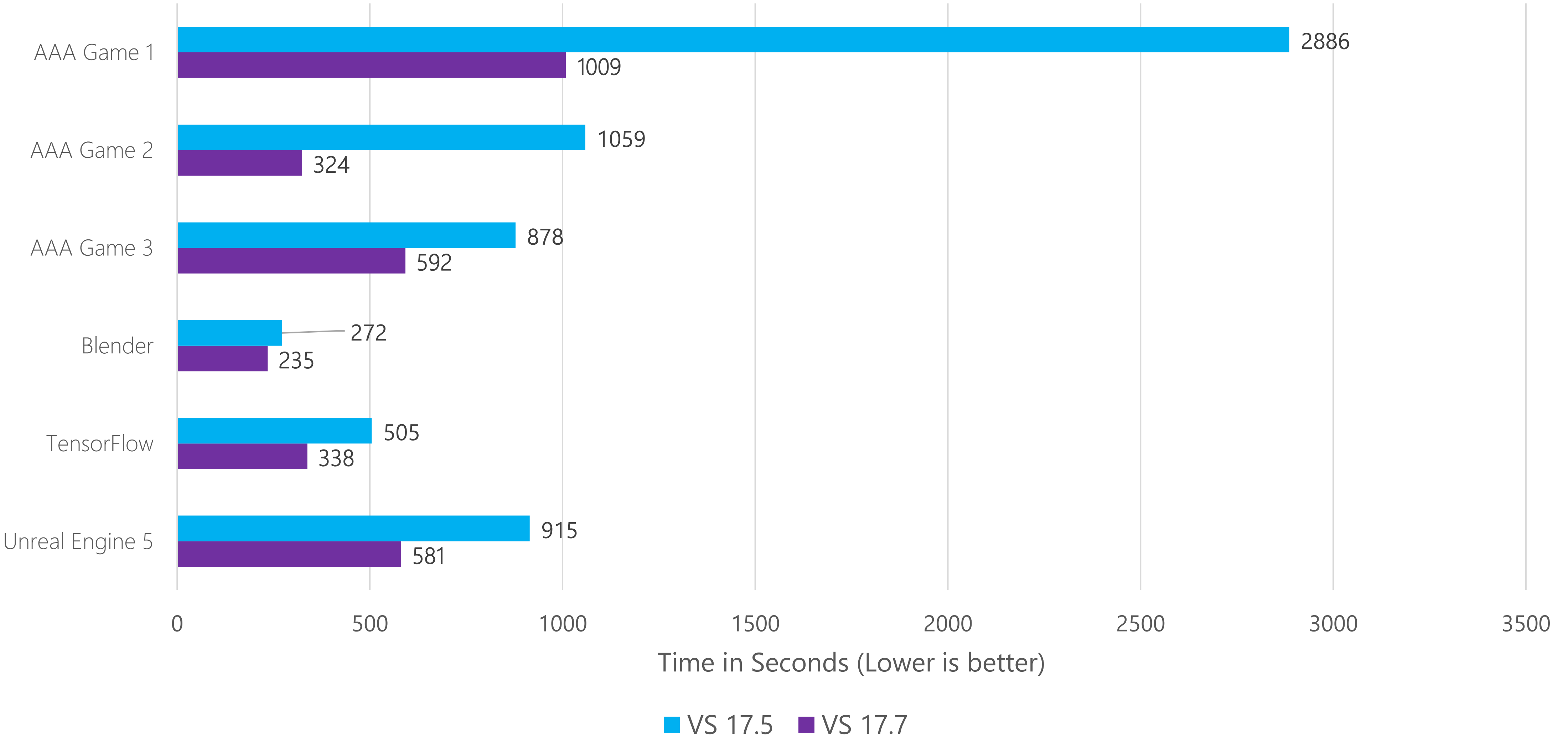
```
; Before

bl          |?did_meow|
uxtb        w8,w0
cmp         w8,#0
cseteq      w0
ldr         lr,[sp],#0x10
```

```
; Before

bl              |?did_meow|
uxtb            w8,w0
cmp             w8,#0
cseteq          w0
ldr             lr,[sp],#0x10
```

```
; After

bl              |?did_meow|
eor             w8,w0,#1
uxtb            w0,w8
ldr             lr,[sp],#0x10
```

MSVC LTCG Build Time Speed Up

Microsoft

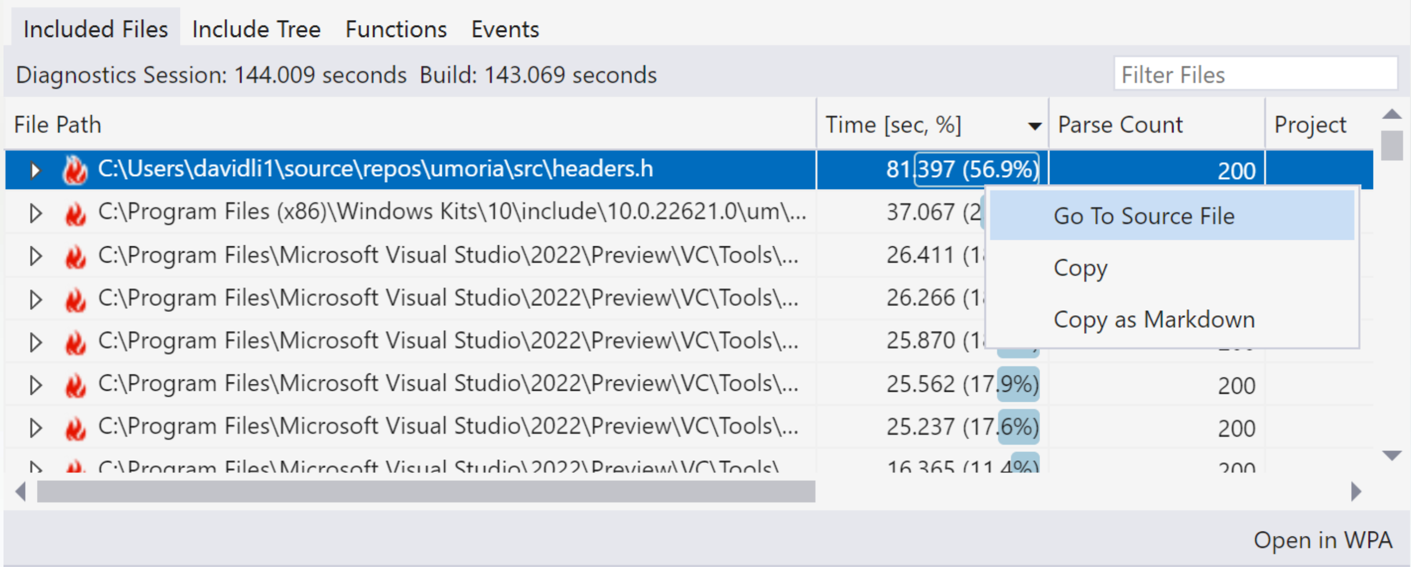


Announcing

In Visual Studio 2022 version 17.7,

Build Insights now integrated with Visual Studio 2022

- Capture .etl Trace from Visual Studio
- Analyze **MSBuild** and **CMake** projects
- Discover
  - **Expensive #includes**
  - Visualized #include tree
  - Function Generation Time
  - Force Inlinees
- Navigate to files
- Open .etl trace with Windows Performance Analyzer



Diagnostics Session: 144.009 seconds Build: 143.069 seconds

File Path	Time [sec, %]	Parse Count	Project
C:\Users\davidli1\source\repos\umoria\src\headers.h	81.397 (56.9%)	200	
C:\Program Files (x86)\Windows Kits\10\include\10.0.22621.0\um\...	37.067 (2		
C:\Program Files\Microsoft Visual Studio\2022\Preview\VC\Tools\...	26.411 (1		
C:\Program Files\Microsoft Visual Studio\2022\Preview\VC\Tools\...	26.266 (1		
C:\Program Files\Microsoft Visual Studio\2022\Preview\VC\Tools\...	25.870 (1		
C:\Program Files\Microsoft Visual Studio\2022\Preview\VC\Tools\...	25.562 (17.9%)	200	
C:\Program Files\Microsoft Visual Studio\2022\Preview\VC\Tools\...	25.237 (17.6%)	200	
C:\Program Files\Microsoft Visual Studio\2022\Preview\VC\Tools\	16.365 (11.4%)	200	

Visit https://aka.ms/BuildInsightsIntegration for more details

Microsoft

Get to Code Faster

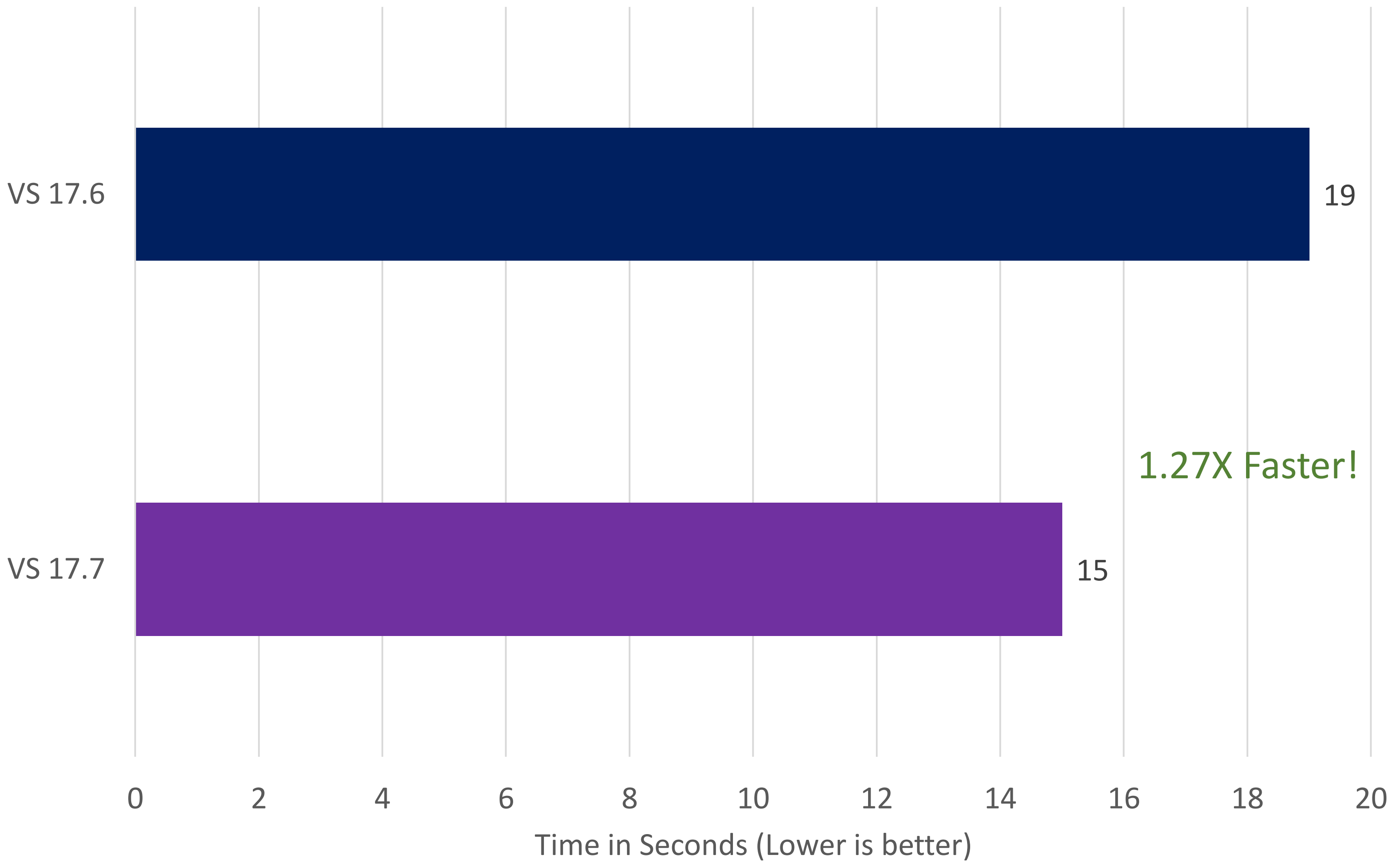
Getting to Code Scenarios	VS2022 17.4.12 (time in seconds)		VS2022 17.8 Preview 2 (time in seconds)
Load Solution (1st Time)	17.5s	1.3X →	13.4s
Colorize (Subsequent)	3.5s	1.8X →	1.9s
Config Switch (Subsequent)	13.7s	1.8X →	7.4s

Benchmark performed using:
12 Core Threadripper, 128GB RAM
Win11 Machine
UE 5.1.1

Visit https://aka.ms/cpp/VSPerformance for more details

Debug Launch Performance for All Projects

Microsoft



Visit https://aka.ms/cpp/VSPerformance for more details

Measured on Unreal Engine 5

Announcing

In Visual Studio 2022 version 17.5,

Improved Debug Build Performance with [[msvc::intrinsic]]

- A community-driven compiler improvement sparked by Vittorio Romeo's blog: [the sad state of debug performance in C++.]
- STL named casts like std::move and std::forward will be inlined at **all** optimization levels.
- Can be added to user-defined functions and function templates.

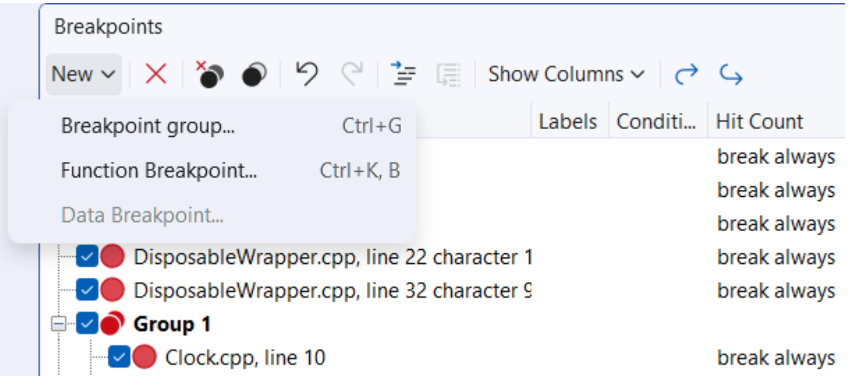
Visit https://aka.ms/msvc-intrinsic for more details

Agenda

1. Conformance & Safety
2. Performance
3. Productivity

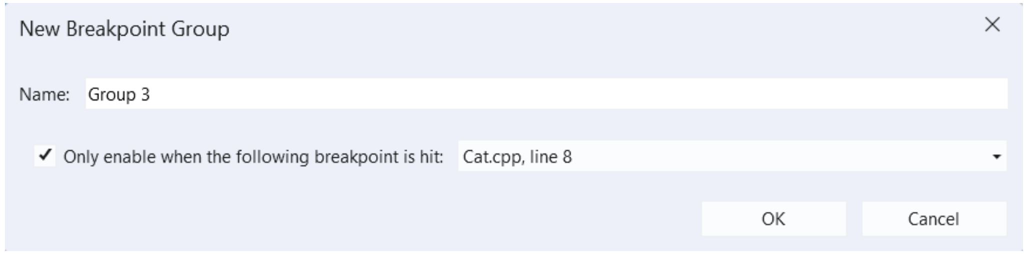
VS Debugger Improvements

Breakpoint Groups



	Labels	Conditi...	Hit Count
			break always
			break always
			break always
☑ DisposableWrapper.cpp, line 22 character 1			break always
☑ DisposableWrapper.cpp, line 32 character 9			break always
☑ **Group 1**			
☑ Clock.cpp, line 10			break always

Dependent Breakpoint Groups



Name: Group 3

✓ Only enable when the following breakpoint is hit: Cat.cpp, line 8

.natjmc & .natstepfilter now supported within individual solutions for better collaboration

Coming Soon: View *Instruction Pointers* for functions in modules view.

Visit https://aka.ms/BreakpointGroups for more details

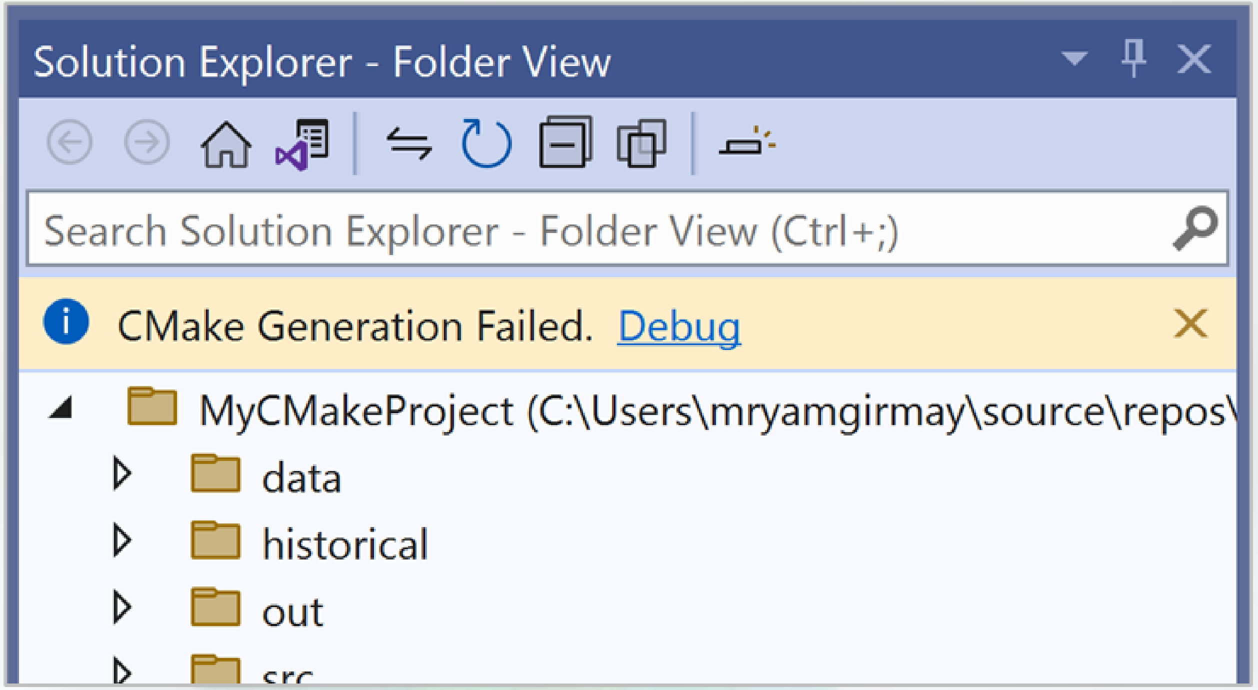
Microsoft

Announcing

In Visual Studio 2022 version 17.5,

CMake Debugger in Visual Studio

- Debug your CMake scripts and CMakeList.txt
- View call stacks of file names
- Open-source



Solution Explorer - Folder View

Search Solution Explorer - Folder View (Ctrl+;)

CMake Generation Failed. Debug

MyCMakeProject (C:\Users\mryamgirmay\source\repos\

data

historical

out

src

Visit https://aka.ms/CMake-debugger for more details

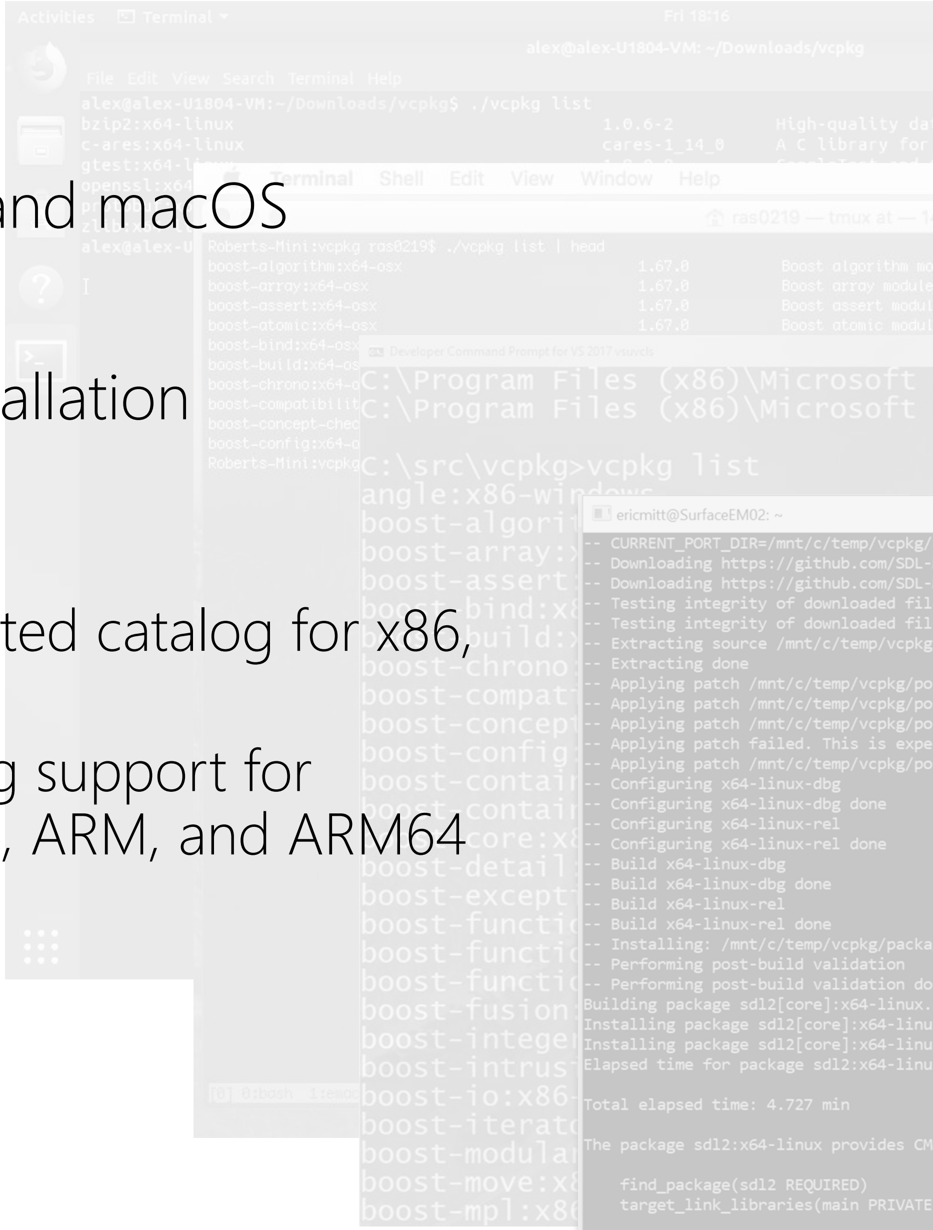
Simplify C++ dependency management with vcpkg

Open-source package manager for Windows, Linux, and macOS

2200+ popular open-source libraries available for installation

- Largest open-source catalog of any package manager
- Built from source, on-demand
- Avoid diamond dependency conflicts with curated, tested catalog for x86, x64, Arm64, and UWP targets
- 50+ unique build targets available by default, including support for Windows, Linux, macOS, iOS, Android, wasm, x86, x64, ARM, and ARM64
  - Or add your own custom targets

https://vcpkg.io



Activities
Terminal
Fri 18:16
alex@alex-U1804-VM: ~/Downloads/vcpkg
File Edit View Search Terminal Help

```
alex@alex-U1804-VM:~/Downloads/vcpkg$ ./vcpkg list
bzip2:x64-linux                                   1.0.6-2          High-quality da
c-ares:x64-linux                                  cares-1_14_0     A C library for
gtest:x64-l
openssl:x64
alex@alex-U
```

Terminal Shell Edit View Window Help
ras0219 — tmux at

```
Roberts-Mini:vcpkg ras0219$ ./vcpkg list | head
boost-algorithm:x64-osx                           1.67.0          Boost algorithm mo
boost-array:x64-osx                               1.67.0          Boost array module
boost-assert:x64-osx                              1.67.0          Boost assert modul
boost-atomic:x64-osx                              1.67.0          Boost atomic modul
boost-bind:x64-osx
boost-build:x64-os
boost-chrono:x64-o
boost-compatibilit
boost-concept-chec
boost-config:x64-o
Roberts-Mini:vcpkg
```

```
C:\Program Files (x86)\Microsoft
C:\Program Files (x86)\Microsoft

C:\src\vcpkg>vcpkg list
angle:x86-wi
boost-algori
boost-array:
boost-assert
boost-bind:x
boost-build:
boost-chrono
boost-compat
boost-concep
boost-config
boost-contai
boost-contai
boost-core:x
boost-detail
boost-except
boost-functi
boost-functi
boost-functi
boost-fusion
boost-intege
boost-intrus
boost-io:x86
boost-iterat
boost-modula
boost-move:x
boost-mpl:x8
```

```
-- CURRENT_PORT_DIR=/mnt/c/temp/vcpkg/
-- Downloading https://github.com/SDL-
-- Downloading https://github.com/SDL-
-- Testing integrity of downloaded fil
-- Testing integrity of downloaded fil
-- Extracting source /mnt/c/temp/vcpkg
-- Extracting done
-- Applying patch /mnt/c/temp/vcpkg/po
-- Applying patch /mnt/c/temp/vcpkg/po
-- Applying patch /mnt/c/temp/vcpkg/po
-- Applying patch failed. This is expe
-- Applying patch /mnt/c/temp/vcpkg/po
-- Configuring x64-linux-dbg
-- Configuring x64-linux-dbg done
-- Configuring x64-linux-rel
-- Configuring x64-linux-rel done
-- Build x64-linux-dbg
-- Build x64-linux-dbg done
-- Build x64-linux-rel
-- Build x64-linux-rel done
-- Installing: /mnt/c/temp/vcpkg/packa
-- Performing post-build validation
-- Performing post-build validation do
Building package sdl2[core]:x64-linux.
Installing package sdl2[core]:x64-linu
Installing package sdl2[core]:x64-linu
Elapsed time for package sdl2:x64-linu

Total elapsed time: 4.727 min

The package sdl2:x64-linux provides CM

    find_package(sdl2 REQUIRED)
    target_link_libraries(main PRIVATE
```

vcpkg for professional C++ development

Manifests

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

Registries

Create your own private library catalogs

Asset Caching (experimental)

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

Versioning

Choose your preferred package versions and lock them for reproducible builds

Triplets

Target over 50 different pre-determined environments or define your own custom one

Binary Caching

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

Visit https://aka.ms/vcpkgdocs for more details

Announcing

In Visual Studio 2022 version 17.7,

vcpkg now shipping with Visual Studio

- Installable component for the **C++ Desktop** and **C++ Gaming** workloads.
- No need to bootstrap.
- Supports manifest mode only.

Visit https://aka.ms/vcpkg-in-box for more details

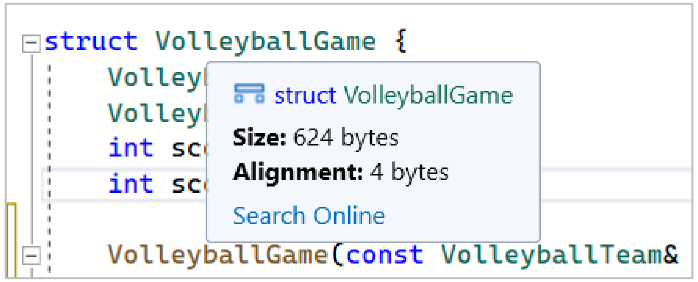
Demo

- vcpkg.json
- CMakePresets.json
- CMake Debugger
- Problem Details Window
- Static Analysis
- Address Sanitizer (ASAN)
- All-In-One-Search
- Sticky Scroll
- Brace Pair Colorization
- Macro Expansion
- Spell Checker
- Build Insights
- #include Cleanup

Microsoft

Level-up C++ Productivity

Size and Alignment



```
struct VolleyballGame {
    Volley
    Volley
    int sc
    int sc

    VolleyballGame(const VolleyballTeam&
```

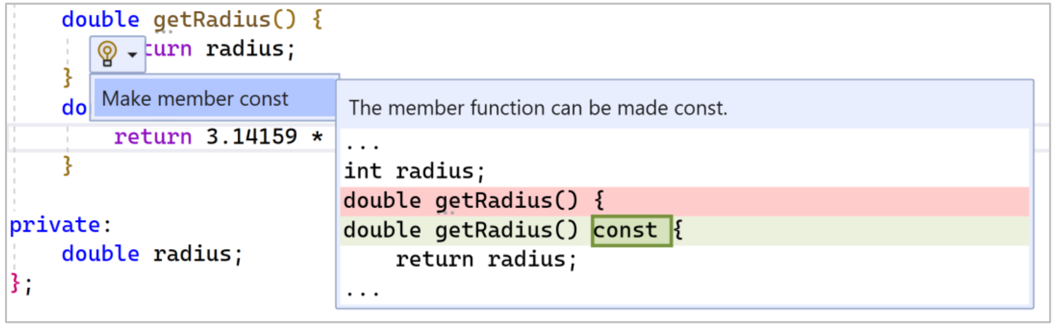
struct VolleyballGame

Size: 624 bytes

Alignment: 4 bytes

Search Online

Make Member Function const



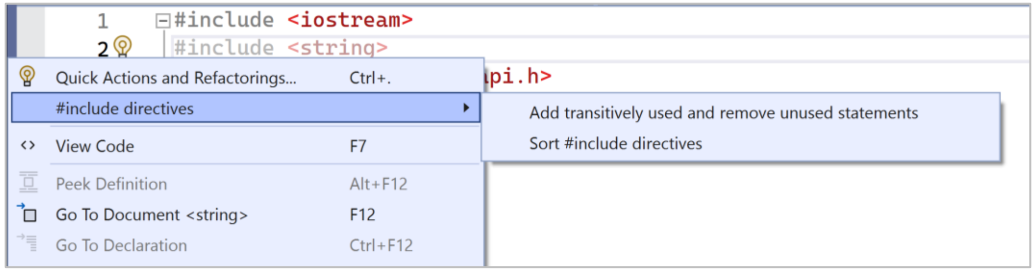
```
    double getRadius() {
        turn radius;
    }
    do
        return 3.14159 *
    }

private:
    double radius;
};
```

The member function can be made const.

```
...
int radius;
double getRadius() {
double getRadius() const {
    return radius;
...
```

#include Cleanup

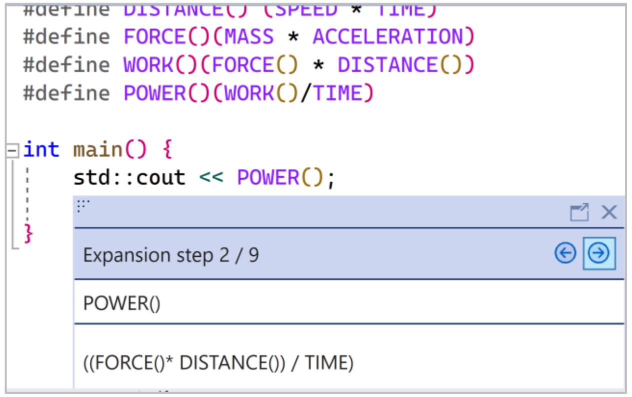
(Preview Feature)



```
#include <iostream>
#include <string>
pi.h>
```

Quick Actions and Refactorings... Ctrl+.
#include directives
View Code F7
Peek Definition Alt+F12
Go To Document <string> F12
Go To Declaration Ctrl+F12
Add transitively used and remove unused statements
Sort #include directives

Visualize Macro Expansion



```
#define DISTANCE() (SPEED * TIME)
#define FORCE()(MASS * ACCELERATION)
#define WORK()(FORCE() * DISTANCE())
#define POWER()(WORK()/TIME)

int main() {
    std::cout << POWER();
}
```

Expansion step 2 / 9

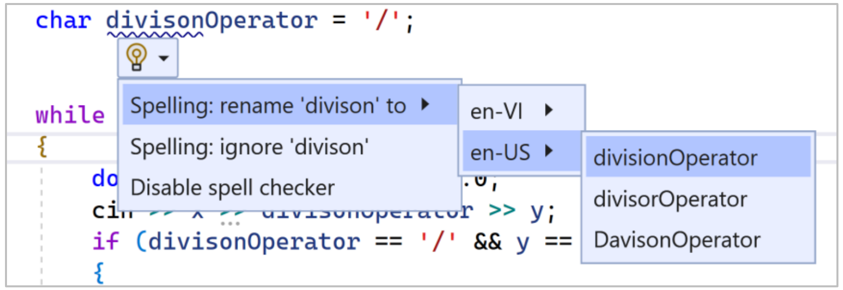
POWER()

((FORCE()* DISTANCE()) / TIME)

Core Editor Improvements

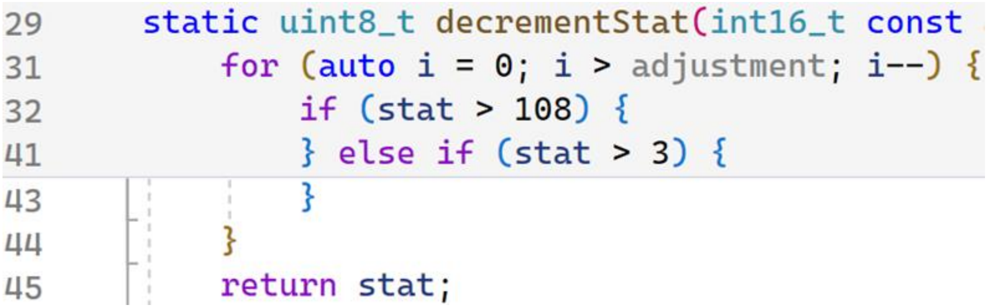
Microsoft

Spell Checker



```
char divisonOperator = '/';

while
{
    do                                   .0;
    ci                                    >> y;
    if (divisonOperator == '/' && y ==
    {
```

Sticky Scroll & Brace Pair Colorization



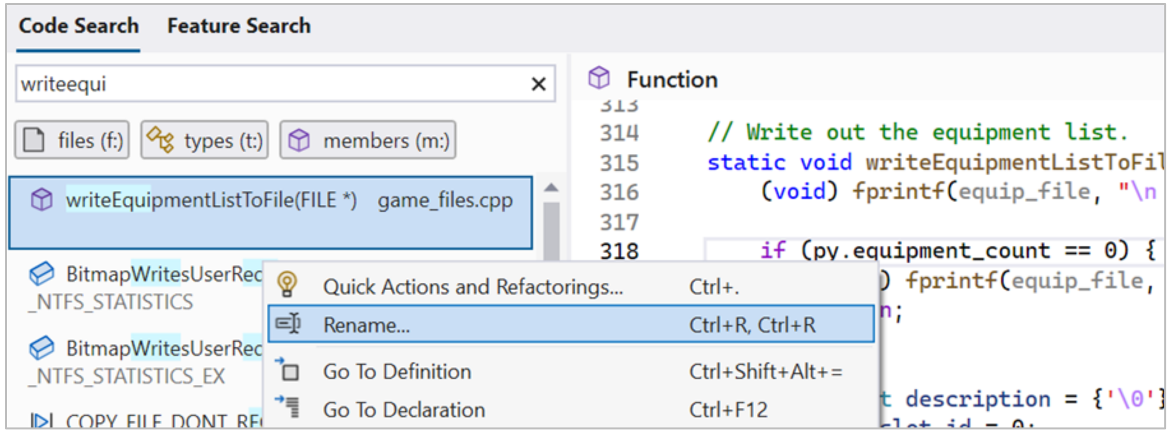
```
    static uint8_t decrementStat(int16_t const
        for (auto i = 0; i > adjustment; i--) {
            if (stat > 108) {
            } else if (stat > 3) {
            }
        }
        return stat;
```

All-in-One Search

Preview Code

Refactorings

Navigation



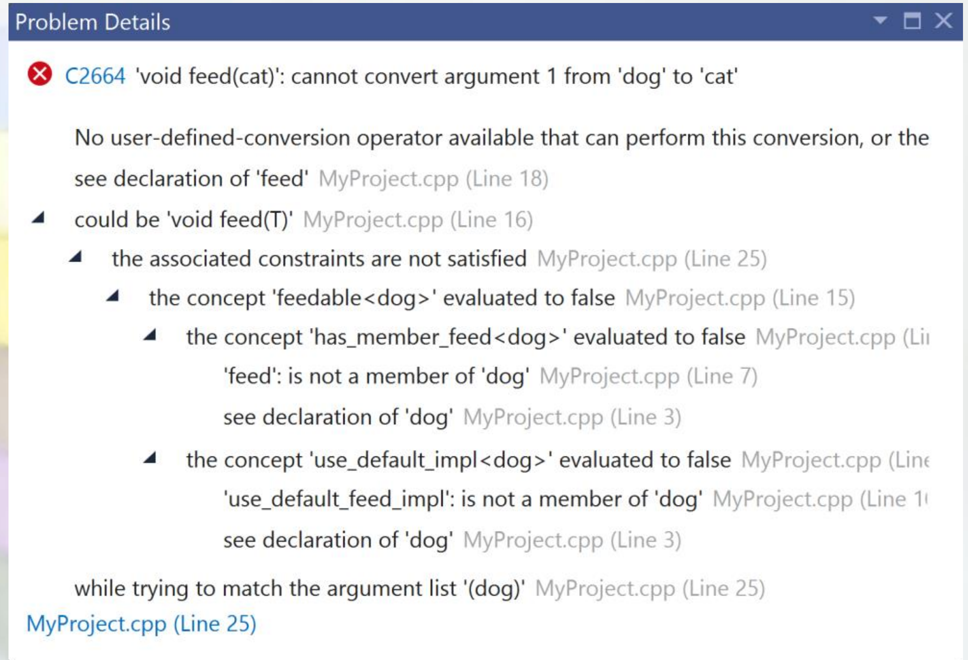
```
    // Write out the equipment list.
    static void writeEquipmentListToFil
        (void) fprintf(equip_file, "\n

        if (py.equipment_count == 0) {
            ) fprintf(equip_file,
            n;

            t description = {'\0'}
```

Announcing

In Visual Studio 2022 version 17.8 Preview 3,

Structured Diagnostics

- Visualize the hierarchical structure of compiler errors
- Expand and collapse to see more/fewer details
- Achieved using the SARIF standard format for diagnostics



C2664 'void feed(cat)': cannot convert argument 1 from 'dog' to 'cat'

No user-defined-conversion operator available that can perform this conversion, or the

see declaration of 'feed' MyProject.cpp (Line 18)

- could be 'void feed(T)' MyProject.cpp (Line 16)
  - the associated constraints are not satisfied MyProject.cpp (Line 25)
    - the concept 'feedable<dog>' evaluated to false MyProject.cpp (Line 15)
      - the concept 'has_member_feed<dog>' evaluated to false MyProject.cpp (Li
        - 'feed': is not a member of 'dog' MyProject.cpp (Line 7)
        - see declaration of 'dog' MyProject.cpp (Line 3)
      - the concept 'use_default_impl<dog>' evaluated to false MyProject.cpp (Line
        - 'use_default_feed_impl': is not a member of 'dog' MyProject.cpp (Line 1
        - see declaration of 'dog' MyProject.cpp (Line 3)

while trying to match the argument list '(dog)' MyProject.cpp (Line 25)

MyProject.cpp (Line 25)

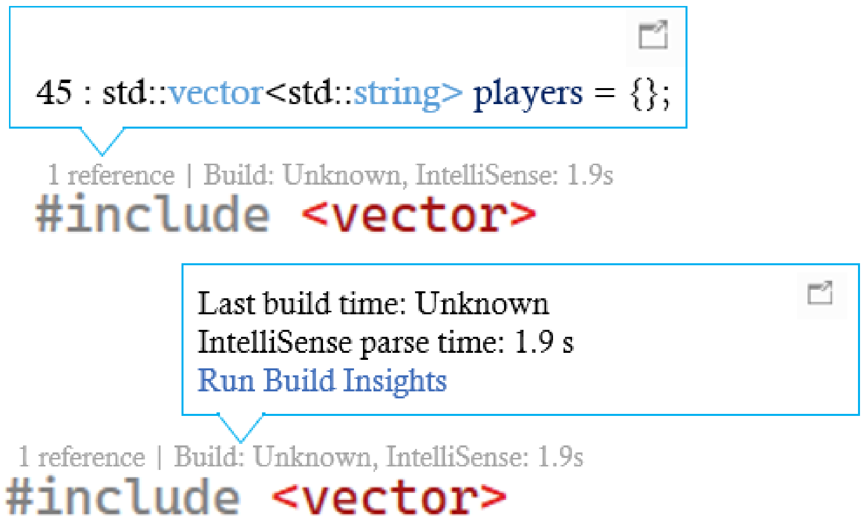
Demo

Structured Diagnostics

Coming Soon

#include Diagnostics

- Build and IntelliSense parse time
- Usage reference



45 : std::vector<std::string> players = {};

1 reference | Build: Unknown, IntelliSense: 1.9s

`#include <vector>`

Last build time: Unknown
IntelliSense parse time: 1.9 s
Run Build Insights

1 reference | Build: Unknown, IntelliSense: 1.9s

`#include <vector>`

Unreal Engine Integrations

Microsoft

Microsoft

Announcing

First-class **Unreal Engine** support

- Superior editing and debugging experience
- Simplifies code management
- Stay focused on your code
- Unreal Engine tools now ever closer to the editing experience

Visit https://aka.ms/cpp/gamedevblogs for more details

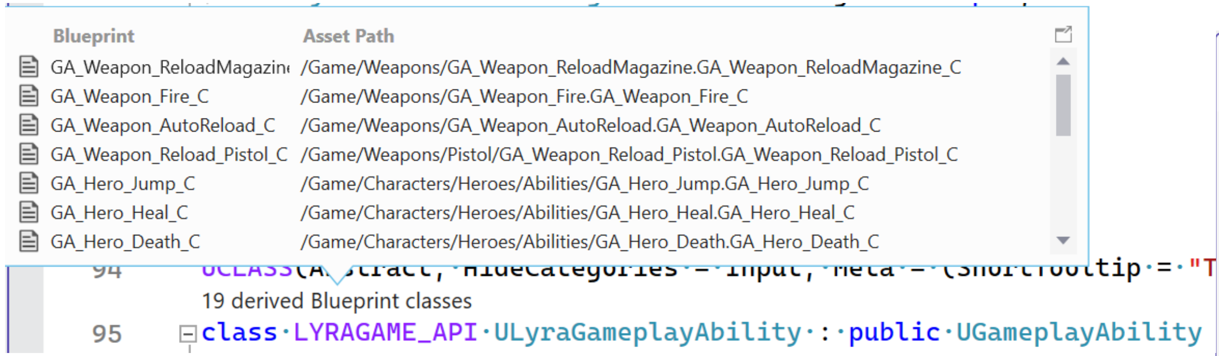
Microsoft

Unreal Engine Integration

*Get Started: aka.ms/**cpp/unreal***

+

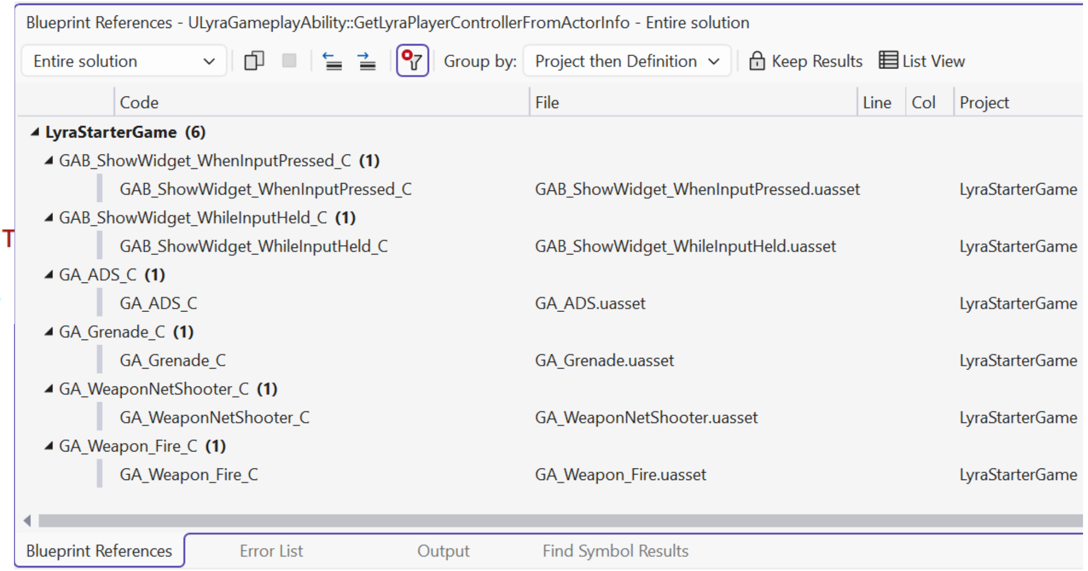
See Blueprint References



Blueprint	Asset Path
GA_Weapon_ReloadMagazin	/Game/Weapons/GA_Weapon_ReloadMagazine.GA_Weapon_ReloadMagazine_C
GA_Weapon_Fire_C	/Game/Weapons/GA_Weapon_Fire.GA_Weapon_Fire_C
GA_Weapon_AutoReload_C	/Game/Weapons/GA_Weapon_AutoReload.GA_Weapon_AutoReload_C
GA_Weapon_Reload_Pistol_C	/Game/Weapons/Pistol/GA_Weapon_Reload_Pistol.GA_Weapon_Reload_Pistol_C
GA_Hero_Jump_C	/Game/Characters/Heroes/Abilities/GA_Hero_Jump.GA_Hero_Jump_C
GA_Hero_Heal_C	/Game/Characters/Heroes/Abilities/GA_Hero_Heal.GA_Hero_Heal_C
GA_Hero_Death_C	/Game/Characters/Heroes/Abilities/GA_Hero_Death.GA_Hero_Death_C

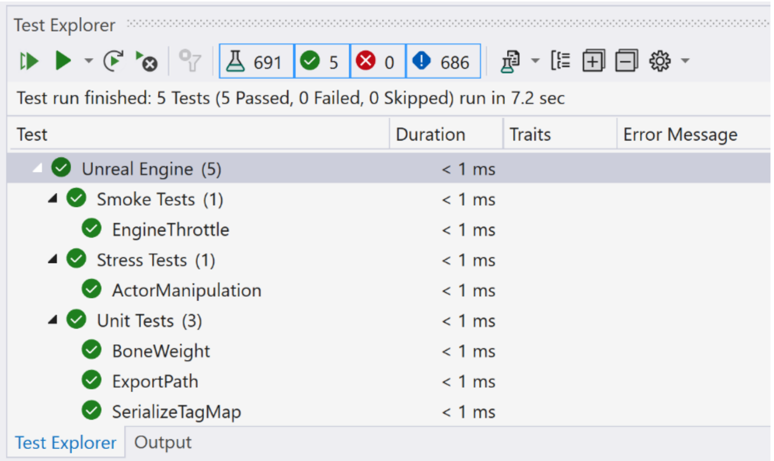
```
UCLASS(Abstract, HideCategories = Input, Meta = (ShortTooltip = "T
19 derived Blueprint classes
class LYRAGAME_API ULyraGameplayAbility : public UGameplayAbility
```

Search Blueprint References



Blueprint References - ULyraGameplayAbility::GetLyraPlayerControllerFromActorInfo - Entire solution

Code	File	Line	Col	Project
LyraStarterGame (6)				
GAB_ShowWidget_WhenInputPressed_C **(1)**				
GAB_ShowWidget_WhenInputPressed_C	GAB_ShowWidget_WhenInputPressed.uasset			LyraStarterGame
GAB_ShowWidget_WhileInputHeld_C **(1)**				
GAB_ShowWidget_WhileInputHeld_C	GAB_ShowWidget_WhileInputHeld.uasset			LyraStarterGame
GA_ADS_C **(1)**				
GA_ADS_C	GA_ADS.uasset			LyraStarterGame
GA_Grenade_C **(1)**				
GA_Grenade_C	GA_Grenade.uasset			LyraStarterGame
GA_WeaponNetShooter_C **(1)**				
GA_WeaponNetShooter_C	GA_WeaponNetShooter.uasset			LyraStarterGame
GA_Weapon_Fire_C **(1)**				
GA_Weapon_Fire_C	GA_Weapon_Fire.uasset			LyraStarterGame



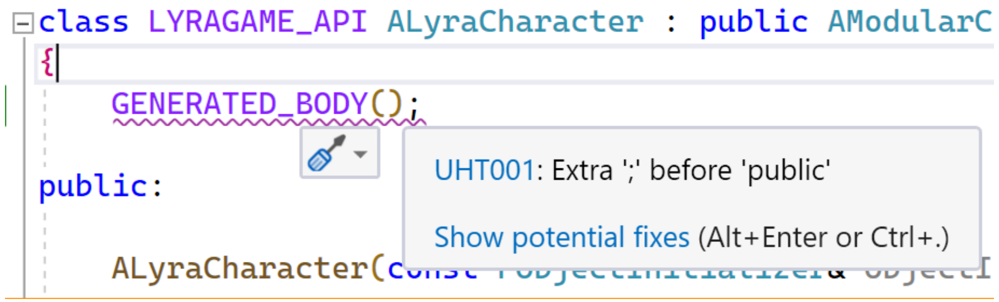
Test run finished: 5 Tests (5 Passed, 0 Failed, 0 Skipped) run in 7.2 sec

Test	Duration	Traits	Error Message
Unreal Engine (5)	< 1 ms		
Smoke Tests (1)	< 1 ms		
EngineThrottle	< 1 ms		
Stress Tests (1)	< 1 ms		
ActorManipulation	< 1 ms		
Unit Tests (3)	< 1 ms		
BoneWeight	< 1 ms		
ExportPath	< 1 ms		
SerializeTagMap	< 1 ms		

Run **UE Tests** in Visual Studio

Unreal Engine Integration

Unreal Header Tool Warnings

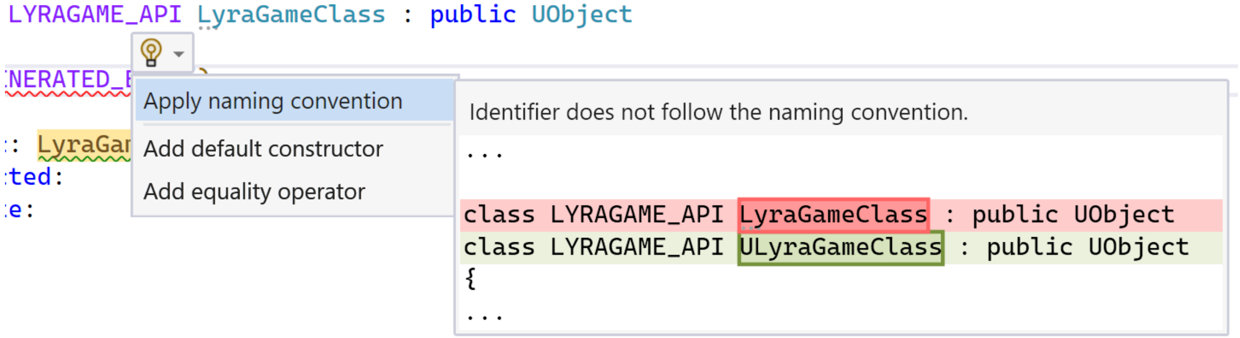


```
class LYRAGAME_API ALyraCharacter : public AModularC
{
    GENERATED_BODY();

public:

    ALyraCharacter(const FObjectInitializer& ObjectI
```

UHT001: Extra ';' before 'public'

Show potential fixes (Alt+Enter or Ctrl+.)

UE Naming Convention Checker

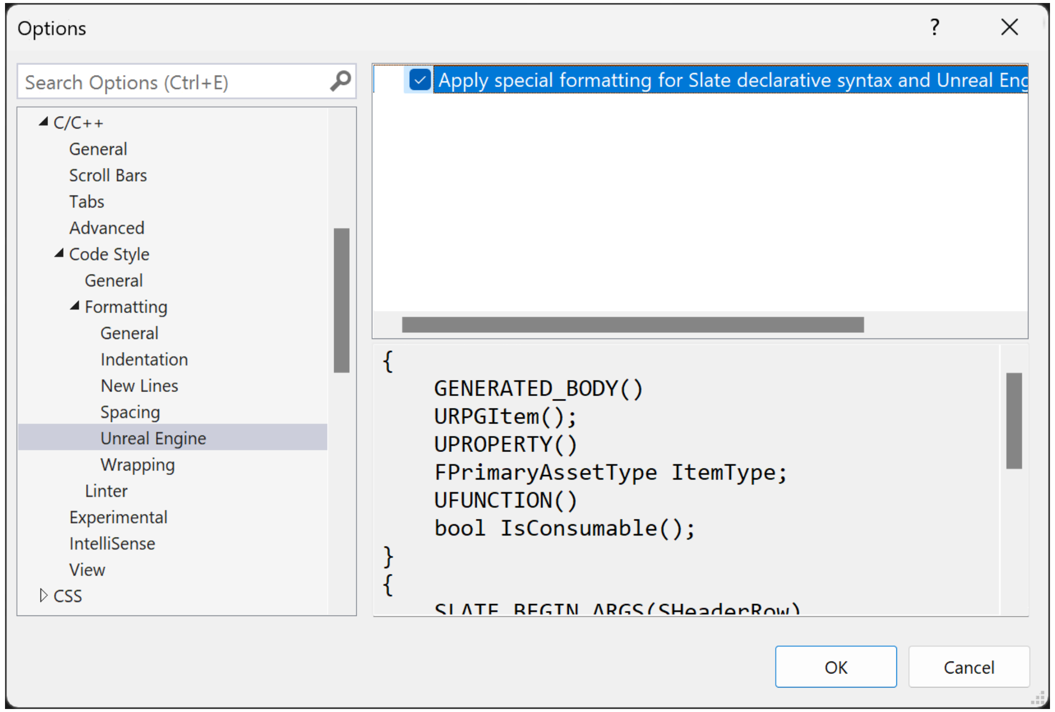


```
LYRAGAME_API LyraGameClass : public UObject

NERATED_E

: LyraGar
ted:
e:
```

Identifier does not follow the naming convention.

```
...

class LYRAGAME_API LyraGameClass : public UObject
class LYRAGAME_API ULyraGameClass : public UObject
{
...
```

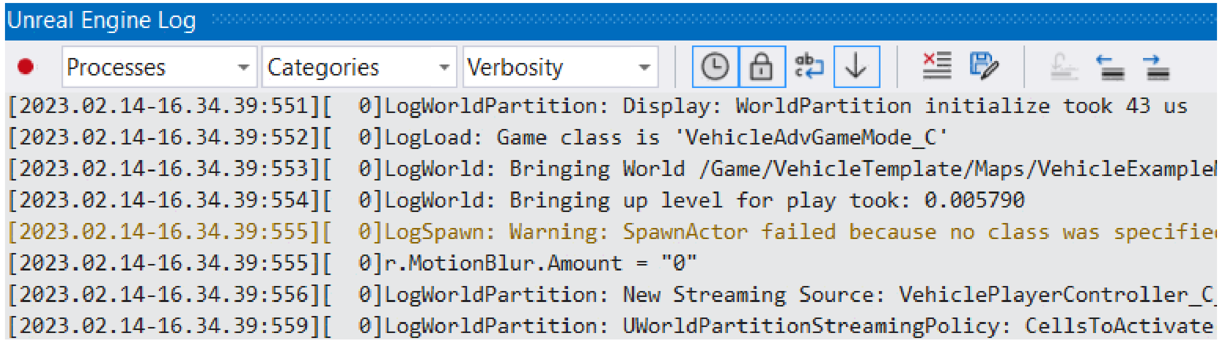
UE Macro Indentation Rules



☑ Apply special formatting for Slate declarative syntax and Unreal Eng

```
{
    GENERATED_BODY()
    URPGItem();
    UPROPERTY()
    FPrimaryAssetType ItemType;
    UFUNCTION()
    bool IsConsumable();
}
{
    SLATE_BEGIN_ARGS(SHeaderRow)
```

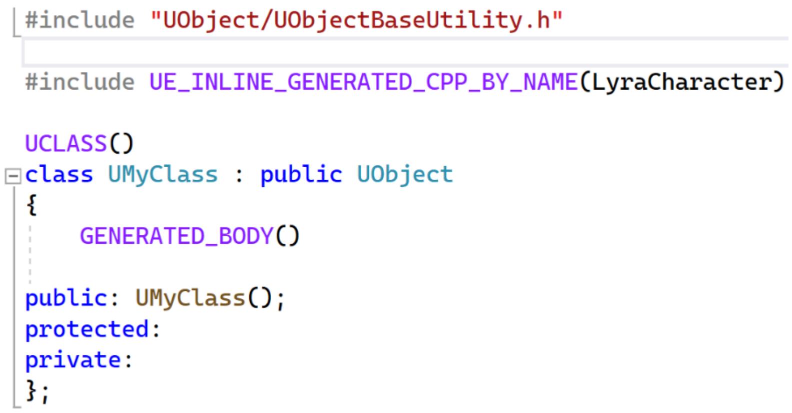
Unreal Engine Integration



```
[2023.02.14-16.34.39:551][  0]LogWorldPartition: Display: WorldPartition initialize took 43 us
[2023.02.14-16.34.39:552][  0]LogLoad: Game class is 'VehicleAdvGameMode_C'
[2023.02.14-16.34.39:553][  0]LogWorld: Bringing World /Game/VehicleTemplate/Maps/VehicleExample
[2023.02.14-16.34.39:554][  0]LogWorld: Bringing up level for play took: 0.005790
[2023.02.14-16.34.39:555][  0]LogSpawn: Warning: SpawnActor failed because no class was specifie
[2023.02.14-16.34.39:555][  0]r.MotionBlur.Amount = "0"
[2023.02.14-16.34.39:556][  0]LogWorldPartition: New Streaming Source: VehiclePlayerController_C
[2023.02.14-16.34.39:559][  0]LogWorldPartition: UWorldPartitionStreamingPolicy: CellsToActivate
```

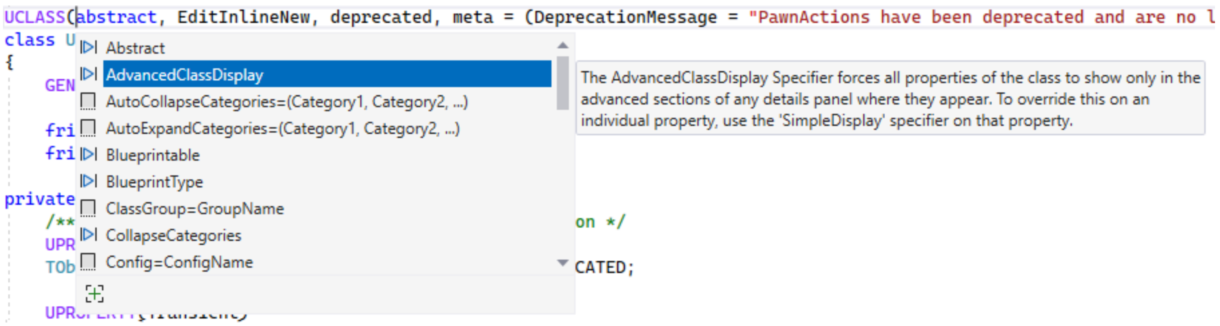
Stream UE Logs

UE Code Snippets



```
#include "UObject/UObjectBaseUtility.h"

#include UE_INLINE_GENERATED_CPP_BY_NAME(LyraCharacter)

UCLASS()
class UMyClass : public UObject
{
    GENERATED_BODY()

public: UMyClass();
protected:
private:
};
```

Coming Soon: Macro Specifier Suggestions



```
UCLASS(abstract, EditInlineNew, deprecated, meta = (DeprecationMessage = "PawnActions have been deprecated and are no l
class U
{
    GEN

    fri
    fri

private
    /**                                                                                  on */
    UPR
    TOb                                                                                  CATED;
```

The AdvancedClassDisplay Specifier forces all properties of the class to show only in the advanced sections of any details panel where they appear. To override this on an individual property, use the 'SimpleDisplay' specifier on that property.

CMake Integration

CMake is a first-class project system in Visual Studio

Familiar edit-build-debug inner-loop
- including MSVC and Clang-tidy squiggles & quick actions
- Doxygen integration

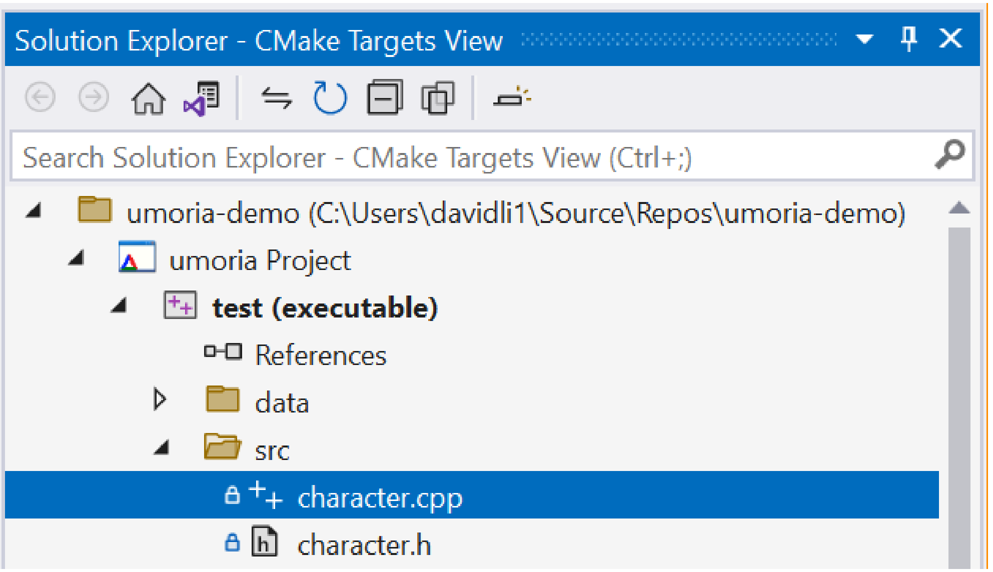
Target Windows, Linux, macOS, MinGW and more

Seamless onramp to CI with CMakePresets.json

C++ Modules support for Visual Studio generator projects

New: **CMake Debugger, Target View Improvements**

Visit https://aka.ms/cmakevs for more details


Solution Explorer - CMake Targets View
Search Solution Explorer - CMake Targets View (Ctrl+;)
umoria-demo (C:\Users\davidli1\Source\Repos\umoria-demo)
umoria Project
test (executable)
References
data
src
character.cpp
character.h

🐧 Linux Targeting

Target any Linux distro, including WSL-2

Interactive SSH Integrated Terminal for Linux

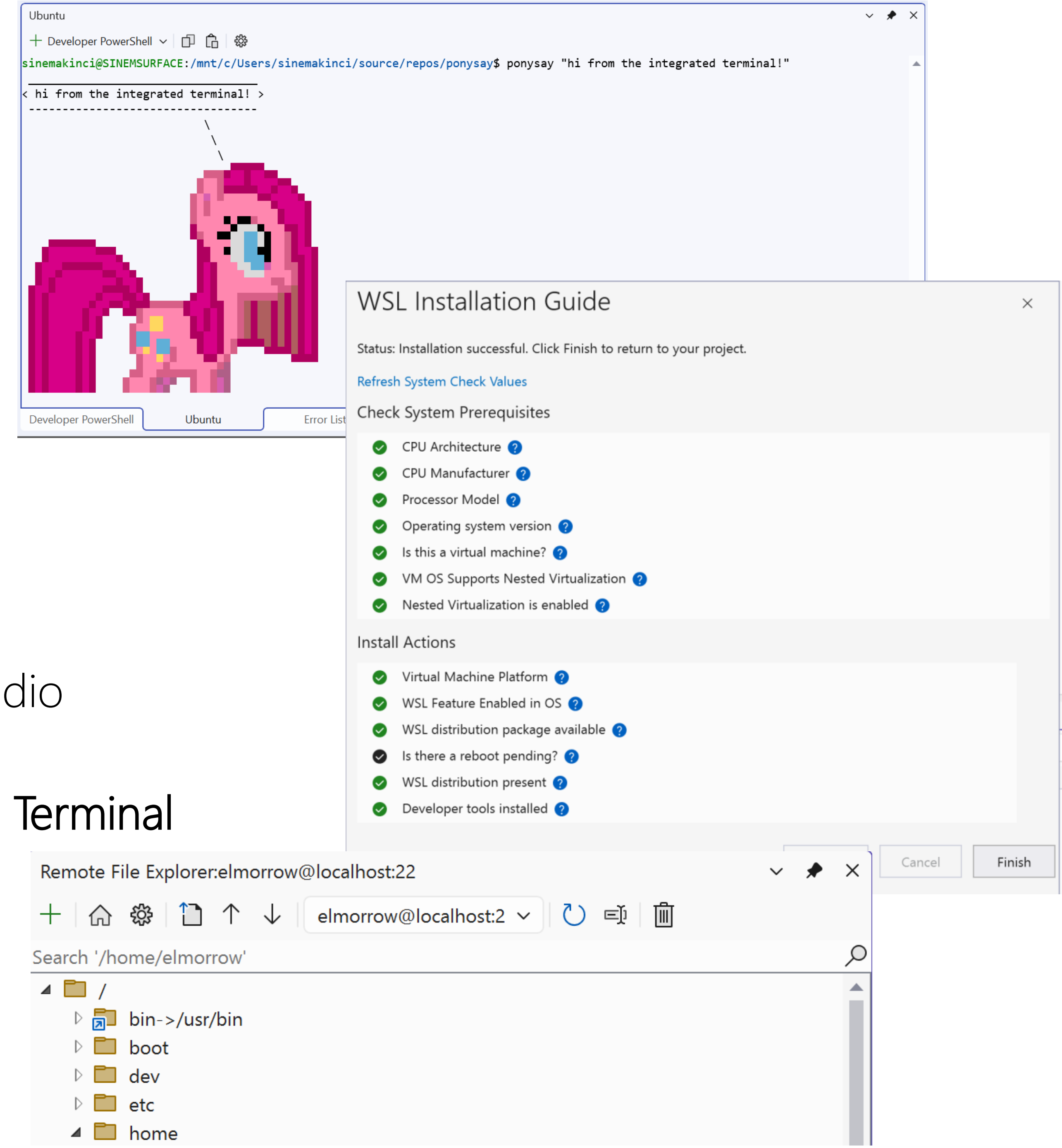
CMake Presets for Linux Targeting

New: **Automatically Install WSL** from Visual Studio

New: Debug Linux Console Apps in **Integrated Terminal**

New: **Remote File Explorer** for Linux

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



```
sinemakinci@SINEMSURFACE:/mnt/c/Users/sinemakinci/source/repos/ponysay$ ponysay "hi from the integrated terminal!"
 _________________________________
< hi from the integrated terminal! >
 ---------------------------------
```

WSL Installation Guide

Status: Installation successful. Click Finish to return to your project.

Refresh System Check Values

Check System Prerequisites

- ✓ CPU Architecture
- ✓ CPU Manufacturer
- ✓ Processor Model
- ✓ Operating system version
- ✓ Is this a virtual machine?
- ✓ VM OS Supports Nested Virtualization
- ✓ Nested Virtualization is enabled

Install Actions

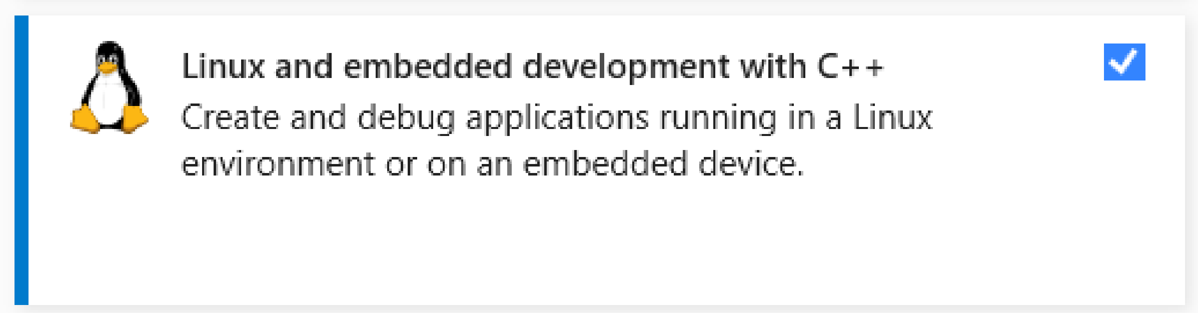
- ✓ Virtual Machine Platform
- ✓ WSL Feature Enabled in OS
- ✓ WSL distribution package available
- ✓ Is there a reboot pending?
- ✓ WSL distribution present
- ✓ Developer tools installed

Remote File Explorer:elmorrow@localhost:22

Microsoft

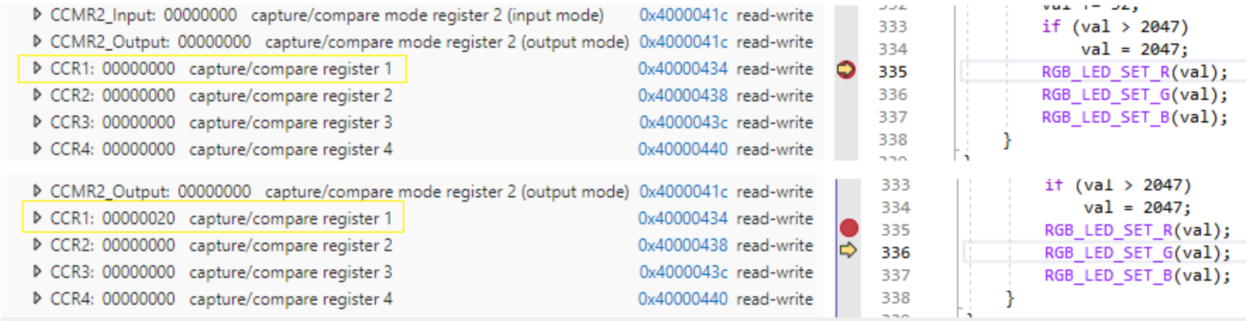
Embedded

- On Chip Debugging with hardware breakpoint limits
- Peripheral and RTOS object views
- Serial monitor
- Available in Visual Studio 2022
  - Add Linux and embedded workload



Linux and embedded development with C++ ☑

Create and debug applications running in a Linux environment or on an embedded device.

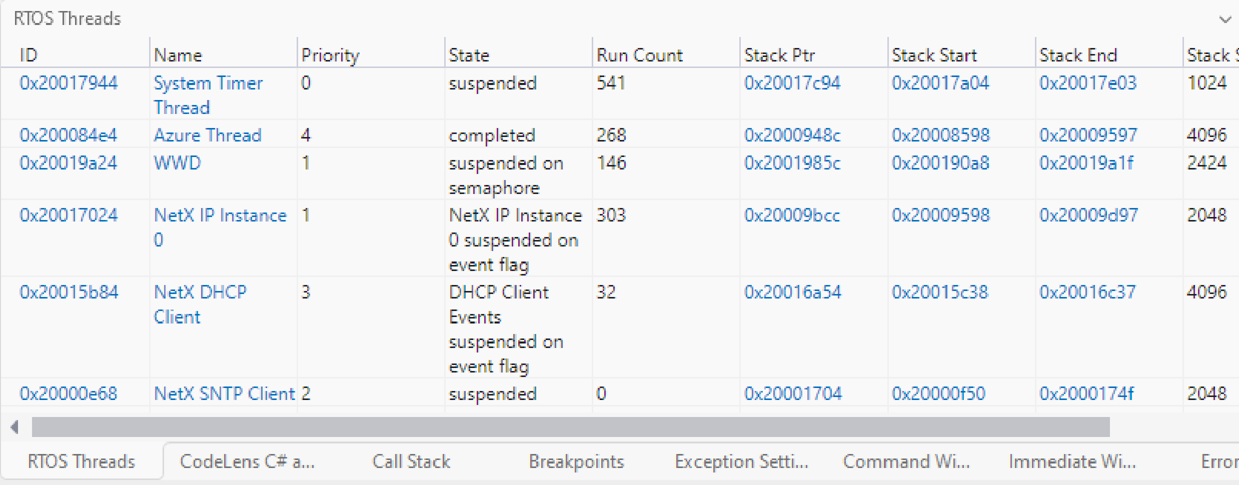


```
▷ CCMR2_Input:  00000000   capture/compare mode register 2 (input mode)    0x4000041c  read-write
▷ CCMR2_Output: 00000000   capture/compare mode register 2 (output mode)   0x4000041c  read-write
▷ CCR1: 00000000  capture/compare register 1                               0x40000434  read-write
▷ CCR2: 00000000  capture/compare register 2                               0x40000438  read-write
▷ CCR3: 00000000  capture/compare register 3                               0x4000043c  read-write
▷ CCR4: 00000000  capture/compare register 4                               0x40000440  read-write
```

```
        if (val > 2047)
            val = 2047;
        RGB_LED_SET_R(val);
        RGB_LED_SET_G(val);
        RGB_LED_SET_B(val);
    }
```

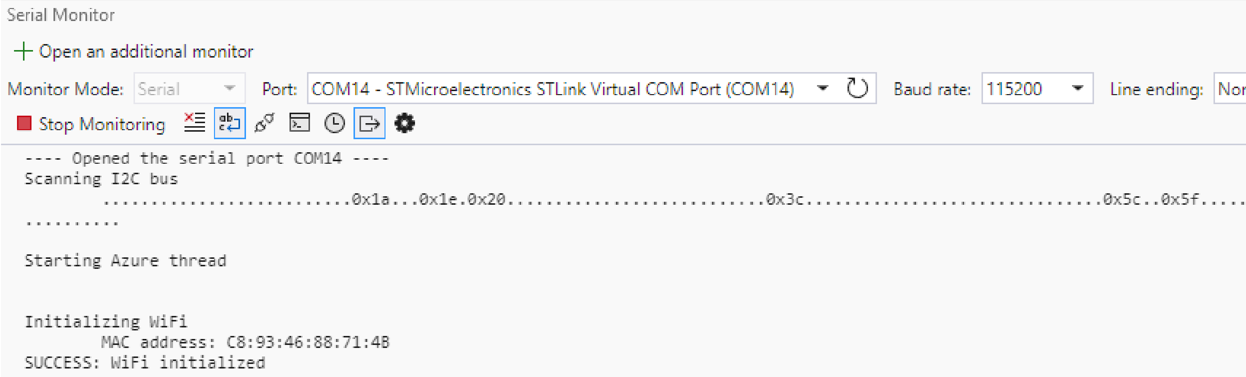
```
▷ CCMR2_Output: 00000000   capture/compare mode register 2 (output mode)   0x4000041c  read-write
▷ CCR1: 00000020  capture/compare register 1                               0x40000434  read-write
▷ CCR2: 00000000  capture/compare register 2                               0x40000438  read-write
▷ CCR3: 00000000  capture/compare register 3                               0x4000043c  read-write
▷ CCR4: 00000000  capture/compare register 4                               0x40000440  read-write
```

```
        if (val > 2047)
            val = 2047;
        RGB_LED_SET_R(val);
        RGB_LED_SET_G(val);
        RGB_LED_SET_B(val);
    }
```



RTOS Threads

ID	Name	Priority	State	Run Count	Stack Ptr	Stack Start	Stack End	Stack
0x20017944	System Timer Thread	0	suspended	541	0x20017c94	0x20017a04	0x20017e03	1024
0x200084e4	Azure Thread	4	completed	268	0x2000948c	0x20008598	0x20009597	4096
0x20019a24	WWD	1	suspended on semaphore	146	0x2001985c	0x200190a8	0x20019a1f	2424
0x20017024	NetX IP Instance 0	1	NetX IP Instance 0 suspended on event flag	303	0x20009bcc	0x20009598	0x20009d97	2048
0x20015b84	NetX DHCP Client	3	DHCP Client Events suspended on event flag	32	0x20016a54	0x20015c38	0x20016c37	4096
0x20000e68	NetX SNTP Client	2	suspended	0	0x20001704	0x20000f50	0x2000174f	2048



Monitor Mode: Serial

Port: COM14 - STMicroelectronics STLink Virtual COM Port (COM14)

Baud rate: 115200

Line ending: Nor

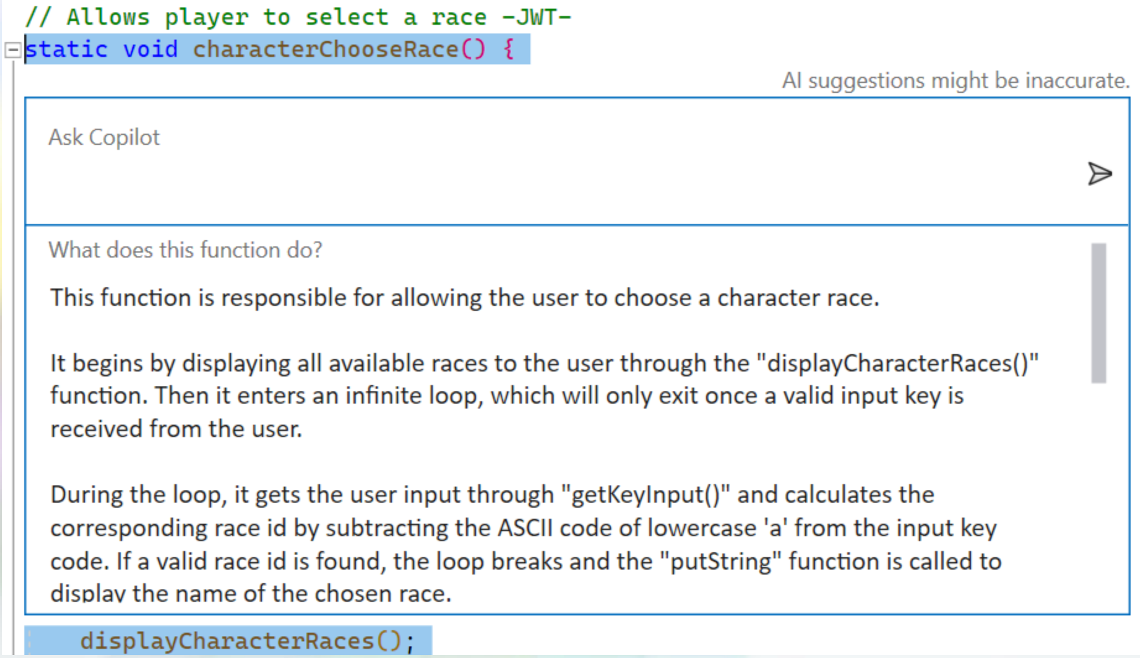
```
---- Opened the serial port COM14 ----
Scanning I2C bus
        .........................0x1a...0x1e.0x20..........................0x3c............................0x5c..0x5f.....
..........

Starting Azure thread


Initializing WiFi
        MAC address: C8:93:46:88:71:48
SUCCESS: WiFi initialized
```

Microsoft

Announcing

GitHub Copilot Chat now in public beta

- **Code Generation**: Autocomplete new lines, comments, and functions
- **Fix Exceptions**: Copilot understands errors and suggest corrections
- **Learning Tool**: Provides you explanations to quickly understand a new code base



```
// Allows player to select a race -JWT-
static void characterChooseRace() {
```

AI suggestions might be inaccurate.

What does this function do?

This function is responsible for allowing the user to choose a character race.

It begins by displaying all available races to the user through the "displayCharacterRaces()" function. Then it enters an infinite loop, which will only exit once a valid input key is received from the user.

During the loop, it gets the user input through "getKeyInput()" and calculates the corresponding race id by subtracting the ASCII code of lowercase 'a' from the input key code. If a valid race id is found, the loop breaks and the "putString" function is called to display the name of the chosen race.

```
    displayCharacterRaces();
```

GitHub Copilot Chat VS Extension Required

GitHub Copilot Chat
GitHub Copilot Chat brings fully integrated AI-powered Copilot chat experiences to Visual Studio. Requires a GitHub ID which has been admit...

Visit https://aka.ms/CopilotChatVS for more details

Make Your Voice Heard!

🏆 Top **Developer Community** Suggestion Tickets

Completed

- C++20 support for C++/CLI
- \`std::move\` (and similar functions) result in poor debug performance and worse debugging experience
- C7602 does not say which constraint is unsatisfied, or why
- Add stdatomic.h support for C11
- "Sync with active document" not working in CMake Target View
- "Linux Console Window" buffer size not configurable
- Integer overflow detection
- Arm64 support for Linux and embedded development with C++

In-progress

- LeakSanitizer
- Multi-argument operator[]
- Full integration of Build Insights

Send us your suggestions: https://aka.ms/cpp/suggest

Microsoft

🚀 Mission of the C++ product team at Microsoft

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

1. by participating in the creation of the **C++ Standards**
2. by investing in the Microsoft Visual C++ (**MSVC**) Compiler & Libraries
3. by simplifying acquisition in C++ via **vcpkg**
4. by improving the **Visual Studio IDE**
5. by continuing to enhance the C++ experience for **Visual Studio Code**

One Last Announcement…

Announcing

IFC SDK for C++ Programs

- Open-source under Apache 2-with-LLVM-exception license
  - Builds on MSVC’s implementation of the IFC Spec
  - IFC file is a persistent representation of C++ Module interface semantics
- Aims to help accelerate C++ Modules in the community and tools ecosystem
- Available on GitHub today: https://github.com/microsoft/ifc

Visit https://aka.ms/IFCSDK for more details

Enjoy the rest of the conference!

Come by our booth and join #visual_studio channel on CppCon Discord **https://aka.ms/cppcon/discord**

- Meet the Microsoft C++ team
- Ask any questions
- Discuss the latest announcements

Take our survey

Win prizes https://aka.ms/cppcon/vs

Our sessions

Monday 2nd

- Lifetime Safety in C++ – Gabor Horvath
- Informal Birds of a Feather for Cpp2/cppfront – Herb Sutter

Tuesday 3rd

- What's New in Visual Studio – David Li & Mryam Girmay

Thursday 5th

- Cooperative C++ Evolution: Towards a Typescript for C++ – Herb Sutter (Keynote)
- How Visual Studio Code Can Help You Develop More Efficiently in C++ – Alexandra Kemper & Sinem Akinci
- Regular, Revisited – Victor Ciura

Friday 6th

- Getting Started with C++ – Michael Price

Microsoft

Happy Coding!

Thank You