Latest and Greatest in Visual Studio for C++ developers

Steve.Carroll & Daniel.Moth

@microsoft.com

https://blogs.msdn.microsoft.com/vcblog/

Win an Xbox One S Forza Horizon 3 + Destiny 2 Bundle!

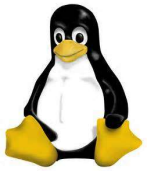
Take this survey: https://aka.ms/cppcon

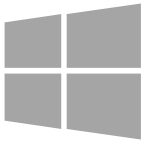
Mission of C++ Product Team at Microsoft (DevDiv)

Make the lives of all C++ developers on the planet better

1. by participating with the **C++ Standards** committee
2. by investing in the Microsoft Visual C++ (**MSVC**) Compiler
3. by improving the **Visual Studio IDE**
4. by continuing to enhance the C++ extension for **Visual Studio Code**

Visual Studio Code: C/C++ Extension





- Lightweight, keyboard focused
- Git integration
- Code Editing
  - IntelliSense, Code Browsing, Switch header/source, Code formatting (clang-format)
- Debugging
  - Core-dump debugging, launch, attach, breakpoints (incl. conditional and function), stepping, threads, call stack, watch, GDB and MI commands
- Easily run, build, test, and run external tasks

https://code.visualstudio.com/docs/languages/cpp

AGENDA

- Microsoft Visual C++ (MSVC) Compiler and Libraries
  - Conformance
  - Compiler Diagnostics
  - Code Analysis
  - Code Gen Quality
  - Build Throughput

- Visual Studio 2017
  - Faster installation and your disk will thank you
  - Pain-free upgrade
  - Just point Visual Studio to your code
  - Use Visual Studio for all your projects and target platforms
  - Be more productive than ever

MSVC Conformance

- Completed in VS2017 RTM or earlier
- Completed in VS2017 15.3
- Targeting in VS2017 15.5
- Targeting in VS2017 15.6

Two-phase Name Lookup (*partial support*) under /permissive-

C++11 ✓

default/deleted func, Inline namespaces, User-defined literals, noexcept, char16_t/char32_t, alignas/alignof, __func__, Extended sizeof, Inheriting constructors, Unicode string literals, Magic statics, thread_local, Unrestricted unions, Attributes, Universal char names in literals, Data-dep ordering attributes, constexpr, Expression SFINAE (via Boost), Expression SFINAE (via more libraries), STL: All C++11 Features

C++14 ✓

auto, decltype(auto) return types, Generic lambdas, Generalized lambda captures, Binary literals, Sized Deallocators, Deprecated attribute, Digit separator, Variable Templates, NSDMI for Aggregates, STL: All C++14 Features, Extended Constexpr

C++17

Auto with braced-init list, u8 char literal, Attributes on namespace and enum, Removing trigraphs, typename in template template-param, Nested Namespace, Ignoring unrecognized attributes, Terse static_assert, Attribute [[fallthrough]], Generalized range-based for-loops,
STL:<any>, STL:<optional>, STL:<string_view>, STL:<variant>, STL:<algorithm> sample(), STL:<tuple> apply()

Structured Bindings | Selection statements with initializers | Construction rules for enum class values | Constexpr lambdas | if constexpr

__has_include | Capturing *this by value | Using attribute namespaces without repetition | [[maybe_unused]] | [[nodiscard]]

Removing operator++ for bool | Remove register keyword | STL:<string_view> UDLs

Over-aligned dynamic memory allocation | Allowing more non-type template args | Fold expressions | Removing some empty unary folds

Adding noexcept to the type system | Inline variables | Removing dynamic-exception-specifications | Hexfloat literals

Matching template template-parameters to compatible arguments

Refining expression evaluation order | Declaring non-type template parameters with auto | Rewording inheriting constructors

Fixing qualification conversions | Guaranteed copy elision | Pack expansions in using-declarations | Extended aggregate initialization

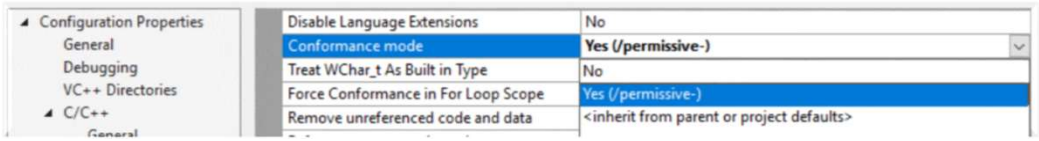
Template argument deduction for class templates

C++TS

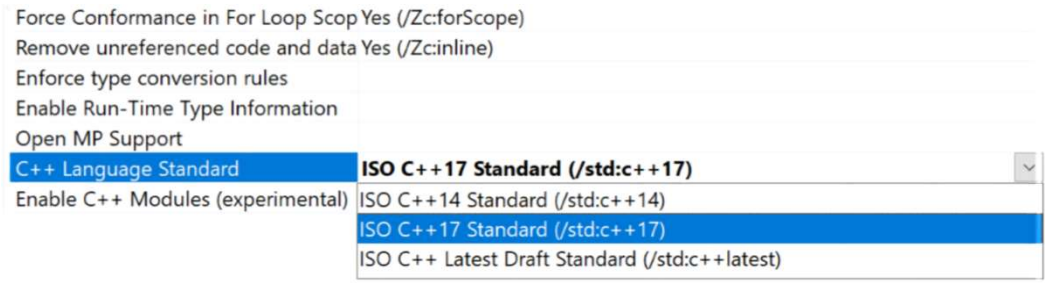
(experimental)

C++ Coroutines | Filesystem* | C++ Modules | C++ Concepts

Compiler Switches



- /permissive-
  - A “less permissive”, more conforming mode in the compiler
  - Off by default now… on by default at some point in the future
  - /Zc conformance switches available for fine-tuning /permissive-
- /std:c++14
  - On by default, for now
  - The latest ISO-blessed C++ standard
    - i.e. standards features up to C++14



- /std:c++17
- /std:c++latest (tracks the latest C++ draft)
  - Off by default, forever
  - Every implemented feature that is not part of a finalized standard yet

Conformance Testing with ~60 OSS Libraries from GitHub

- Testing with GitHub master branches and compiler development trunk
  - MSVC default mode – 58 projects
  - MSVC /std:c++17 mode – 58 projects
  - MSVC /permissive- mode – 55 projects

No.	Source
1	CoreCLR
2	Chakra
3	ClangLLVM
4	OpenSSL
5	Chrome
6	OpenCV
7	RxCpp
8	Boost
9	UnrealEngine
10	Electron
11	QTCreator
12	QT
13	Cocos2dx
14	OSQuery
15	FLAC
16	WinRT
17	Z3
18	PDFium
19	X265
20	RocksDB
21	VCPKG
22	PostgreSQL
23	CryEngine
24	APPLE_LZFSE
25	Blender
26	Dolphin
27	Facebook_ZSTD
28	Glslang
29	Google_Brotli
30	Google_LiquidFun
31	Google_MathFu
32	Google_Protobuf
33	Google_RE2
34	Google_Snappy
35	Google_VP9
36	Google_SwiftShader
37	Irrlicht
38	LAME
39	ITK
40	VTK
41	Sprout
42	LibGIT2
43	LibJPEG
44	LibJPEG_Turbo
45	LUA
46	LUAJIT
47	LZ4
48	Serious_Engine
49	Python3
50	PHP7
51	MySQL
52	Mesos
53	SDL
54	Azure_iot_sdk_c
55	Dlib
56	Bond
57	KTL
58	Outcome

Compiler Diagnostics Improvements

- Column Information and Source Context : /diagnostics:caret

```
2     int foo(int, int);
3
4   int main()
5   {
6       return foo(foo(foo(1, 2), foo(3, 4), 5), foo(6, 7));
7   }
```

Before:

```
test.cpp(6): error C2660: 'foo': function does not take 3 arguments
```

After:

```
test.cpp(6,40): error C2660: 'foo': function does not take 3 arguments
    return foo(foo(foo(1, 2), foo(3, 4), 5), foo(6, 7));
                                             ^
```

- Special Member Function errors

```
1   struct A
2   {
3       A(); // non-trivial constructor
4   };
5
6   struct B {}; // trivial constructor
7
8   union variant
9   {
10      A a;
11      B b;
12  };
13
14  int main()
15  {
16      variant var;
17  }
18
```

Before:

```
variant.cpp
variant.cpp(16): error C2280: 'variant::variant(void)': attempting to reference a deleted function
variant.cpp(12): note: compiler has generated 'variant::variant' here
```

After:

```
variant.cpp
variant.cpp(13,1): error C2280: 'variant::variant(void)': attempting to reference a deleted function
  variant var;
      ^
variant.cpp(10): note: compiler has generated 'variant::variant' here
};
variant.cpp(10,1): note: function was deleted because a variant member has a non-trivial constructor 'A::A(void)'
};
^
variant.cpp(2): note: see declaration of 'A::A'
  A(); // non-trivial constructor
```

Continued Compiler Diagnostics Improvements

- Template dependent name diagnostics

Example	Current diagnostic (under /permissive-)	New diagnostic under /permissive-

template<typename T>
void bar()
{
    T::foo<int>(); // Should be T::template foo<T>();
}
	test.cpp(5,17): error C2187: syntax error: ')' was unexpected here
	    T::foo<int>();
	                ^
		test.cpp(5,8): error C7510: 'foo': use of dependent template name requires 'template' keyword
		    T::foo<int>();
		       ^

template <typename T>
void bar() {
    T::Type x; // Should be typename T::Type x
}
	test.cpp(5,13): error C2760: syntax error: unexpected token 'identifier', expected ';'
	    T::Type x;
	            ^
		test.cpp(5,5): error C7511: 'Type': use of dependent type name must be prefixed with 'typename'
		    T::Type x;
		    ^

struct X {};
template <class T>
void bar() {
    typename X x; // Should be "X x"
}
	test.cpp(5,11): error C2760: syntax error: unexpected token 'identifier', expected 'id-expression'
	        typename X x;
	                  ^
		test.cpp(5,11): error C7511: 'X': 'typename' keyword must be followed by a qualified name
		        typename X x;
		                  ^

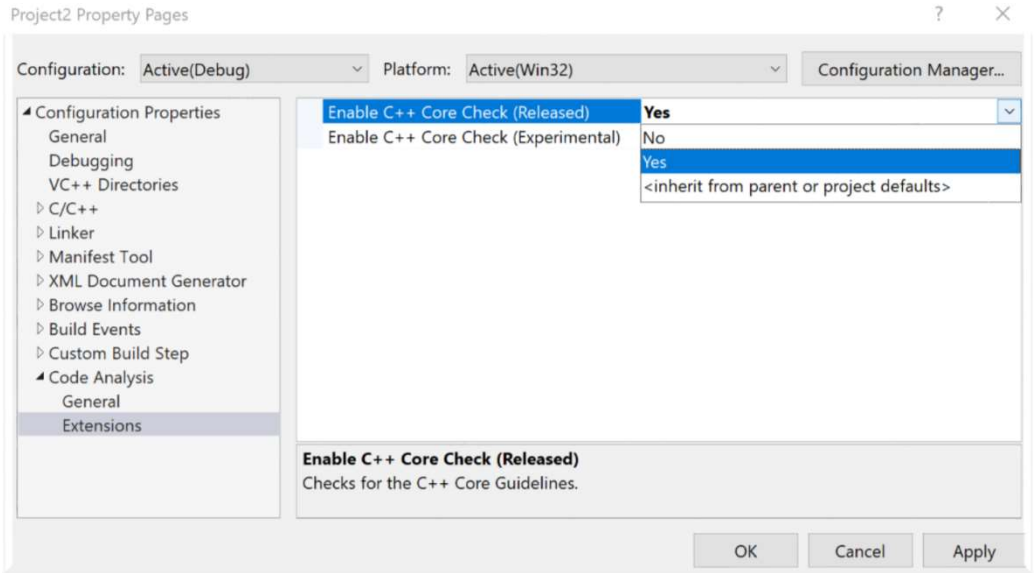
- Member initialization order

Example	Current diagnostic	New diagnostic

struct C
{
    C (int a): y(a), x(y) {}
    int x;
    int y;
};
	No diagnostics	warning C5038: data member 'C::y' will be initialized after data member 'C::x'

C++ Core Check to enforce the C++ Core Guidelines

- Resource Management
  - Help with low-level resource management
  - Identify missing resource cleanup
  - Help with smart-pointer usage
- Interfaces; Expressions and Statements
  - Guide better class declarations
  - Prevent use of dangerous constructs
  - Catch simple but hard-to-spot mistakes
- Constants and Immutability



https://aka.ms/CppCoreCheck

Generated Code Quality: 8.9% better VS2015 -> VS2017

- Taking advantage of the new SSA-based optimizer
  - Partial redundancy elimination
  - Common subexpression elimination overhaul, focus on eliminating loads
  - New control flow graph optimization module
- Many inliner improvements
  - More aggressive and precise inline heuristic in the presence of C++ EH
  - Better leveraged information of single call-site and nested loops
  - Improved inlining for very small functions with and without PGO
  - Better cooperation with the _restrict keyword
- Improved loop optimizations
  - Significantly improved loop unswitching
  - Significantly improved complete loop unrolling
  - Improved conditional vectorization heuristics
  - Improved speculative memset generation
- New optimizations
  - SLP vectorizer & vectorization of min/max sequence reductions involving scalars
  - Scalar replacement to sink stores out of loops

SPEC 2017 Benchmark	VS 2015 Update 3	VS 2017 15.5
602.gcc_s	521	443
605.mcf_s	572	546
620.omnetpp_s	402	393
623.xalancbmk_s	163	157
625.x264_s	269	204
631.deepsjeng_s	317	302
641.leela_s	450	431
657.xz_s	2247	2101
619.lbm_s	943	869
638.imagick_s	5721	4891
644.nab_s	1907	1637
508.namd_r	226	222
510.parest_r	287	280
511.povray_r	336	337
526.blender_r	278	239

Measurements in seconds
Lower is better

Build Throughput

- Building SPEC 2017 is 20% faster end-to-end

- When using /debug:fastlink, 2-4x faster
  - E.g, Fable, Forza, Chrome, Bing Maps
  - On by default in VS 2017, and mspdbcmf.exe integrated as part of Visual Studio build experience

SPEC 2017 Benchmark	VS 2015 Update 3	VS 2017 15.5
602.gcc_s	145	132
605.mcf_s	2.5	2
620.omnetpp_s	85	65
623.xalancbmk_s	189	147
625.x264_s	28	24
631.deepsjeng_s	5	5
641.leela_s	11	9
657.xz_s	10	6
619.lbm_s	2	2
638.imagick_s	54	46
644.nab_s	6	6
508.namd_r	15	14
510.parest_r	205	158
511.povray_r	27	22
526.blender_r	221	189
Total buildtime:	**1005.5**	**827**

Measurements in seconds
Lower is better

- IncrediBuild included free with VS 2017
  - Generates an improved build plan which breaks down false dependencies
  - Intelligent dynamic resource management (up to 8 cores with the FREE extension)
    - E.g., Qt from 942 to 844 seconds
    - E.g., ACE from 392 to 282 seconds

  - Build visualization and diagnostic tools to find build bottlenecks

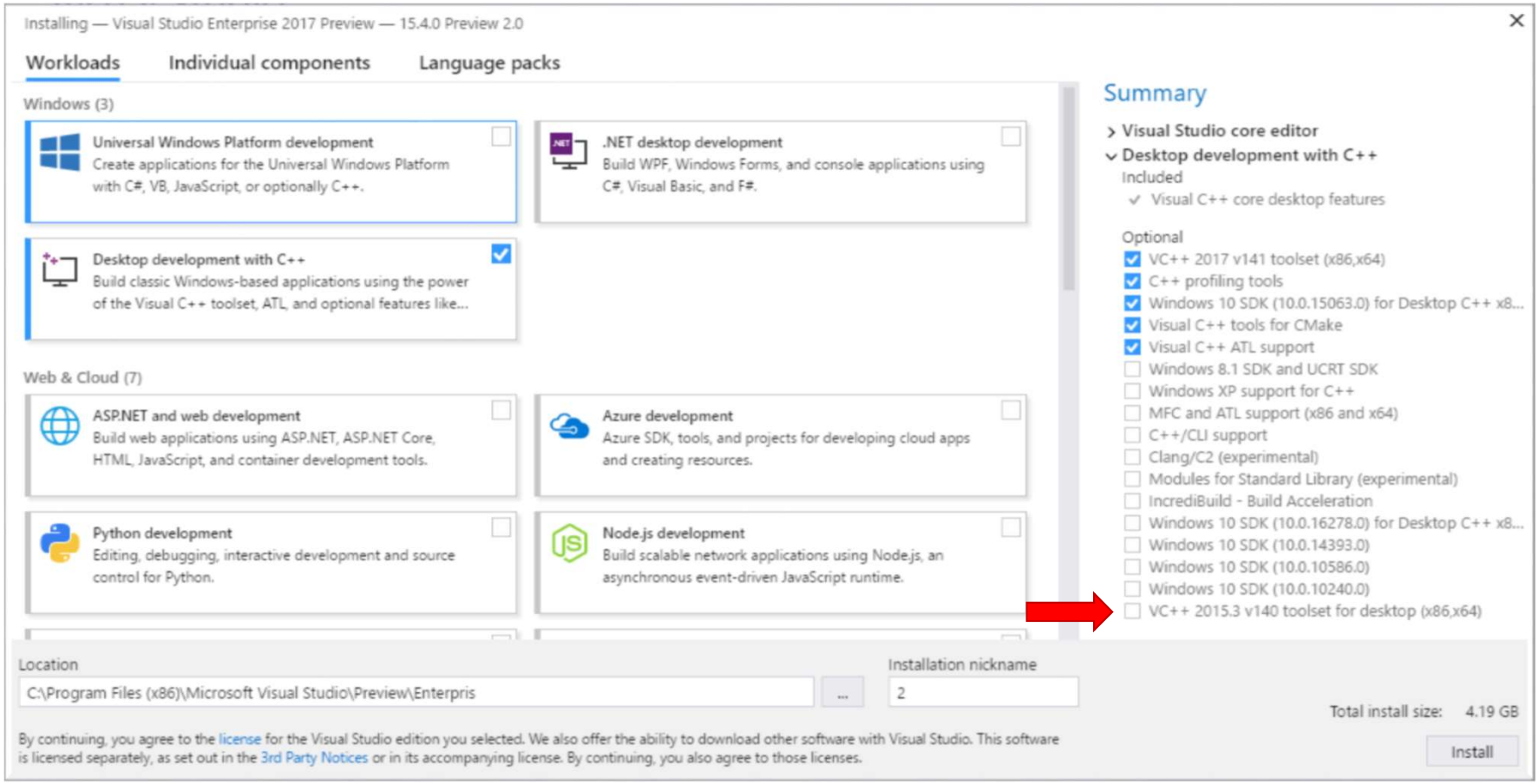
AGENDA

- Microsoft Visual C++ (MSVC) Compiler and Libraries
  - Conformance
  - Compiler Diagnostics
  - Code Analysis
  - Code Gen Quality
  - Build Throughput

- Visual Studio 2017
  - Faster installation and your disk will thank you
  - Pain-free upgrade
  - Just point Visual Studio to your code
  - Use Visual Studio for all your projects and target platforms
  - Be more productive than ever

Visual Studio 2017

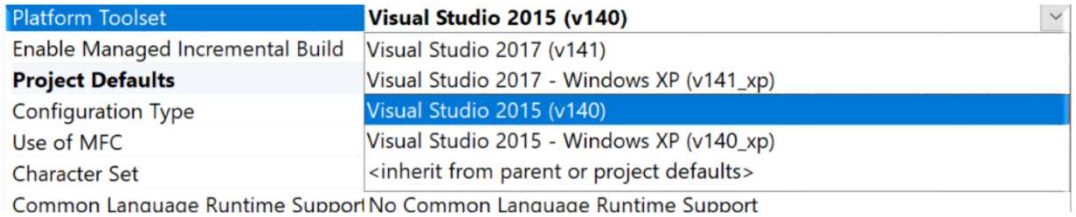
RTM 15.0	15.3	15.5	15.6
• Mar 2017	• Aug 2017	• Nov 2017	• ??? 201?



https://aka.ms/vcinstaller

Pain-Free Upgrade to VS 2017

- Install VS2015 toolset with VS2017 (without needing the VS2015 IDE)



- Compiler Switches – “pay for play”
- Binary compatibility between the VS2015 and VS2017 runtimes
- Vcpkg for getting the latest version of open source libraries

Porting and Upgrading Guide: https://msdn.microsoft.com/library/dn986839.aspx

Vcpkg - Libraries Acquisition

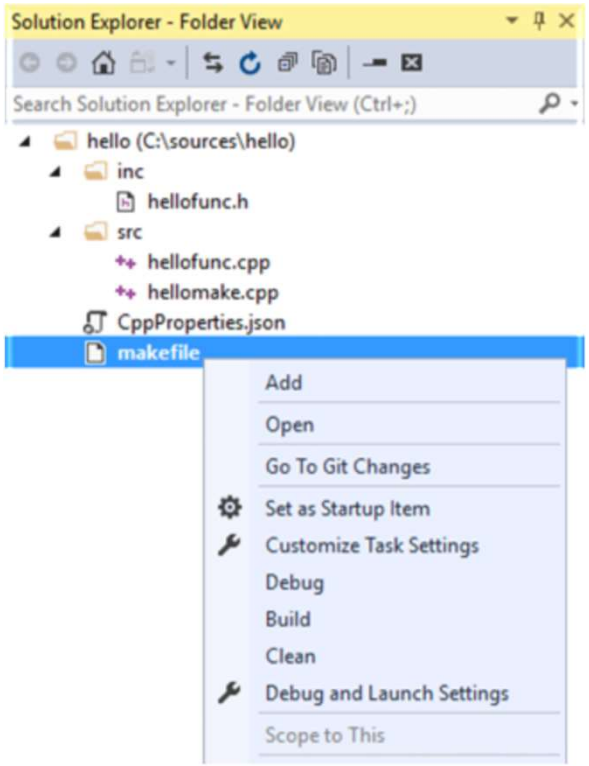
- 80% of C++ projects use 2 or more 3rd party libs
  - A majority of them use open source libraries
- Open source tool based on a port tree approach (Vcpkg)
  - Usage: vcpkg install boost
  - Installs the .h, .lib and binaries in a “lib folder” ready to use
  - 350+ libraries in the catalog, added by 150+ contributors

Tuesday 8AM “Package management for C++ OSS libraries on Windows with vcpkg”

https://github.com/Microsoft/vcpkg

Open Folder, CMake etc

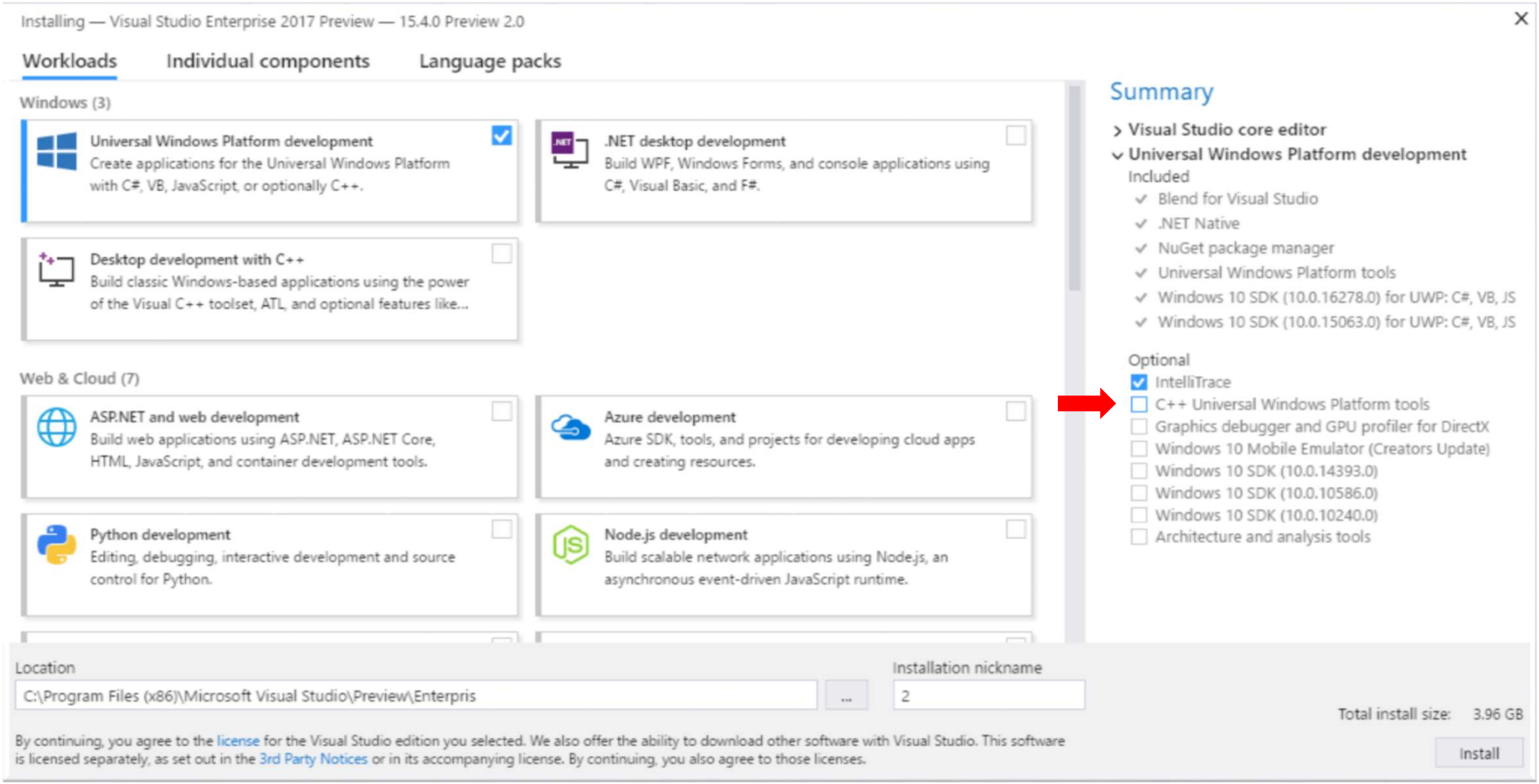
- Ideal for non-MSBuild projects
  - Work with any project, e.g. CMake, make, and other C++ build systems
  - Cross-platform development with MinGW and Cygwin
- Easy to get started
  - devenv.exe <directory>
  - “File > Open > Folder…” (Ctrl+Alt+Shift+O)
- Read, Navigate, Edit
  - All C++ navigation and IntelliSense features
- Build, Debug
  - Flexible integration of external build processes
  - Familiar Visual Studio debugging experience



AGENDA

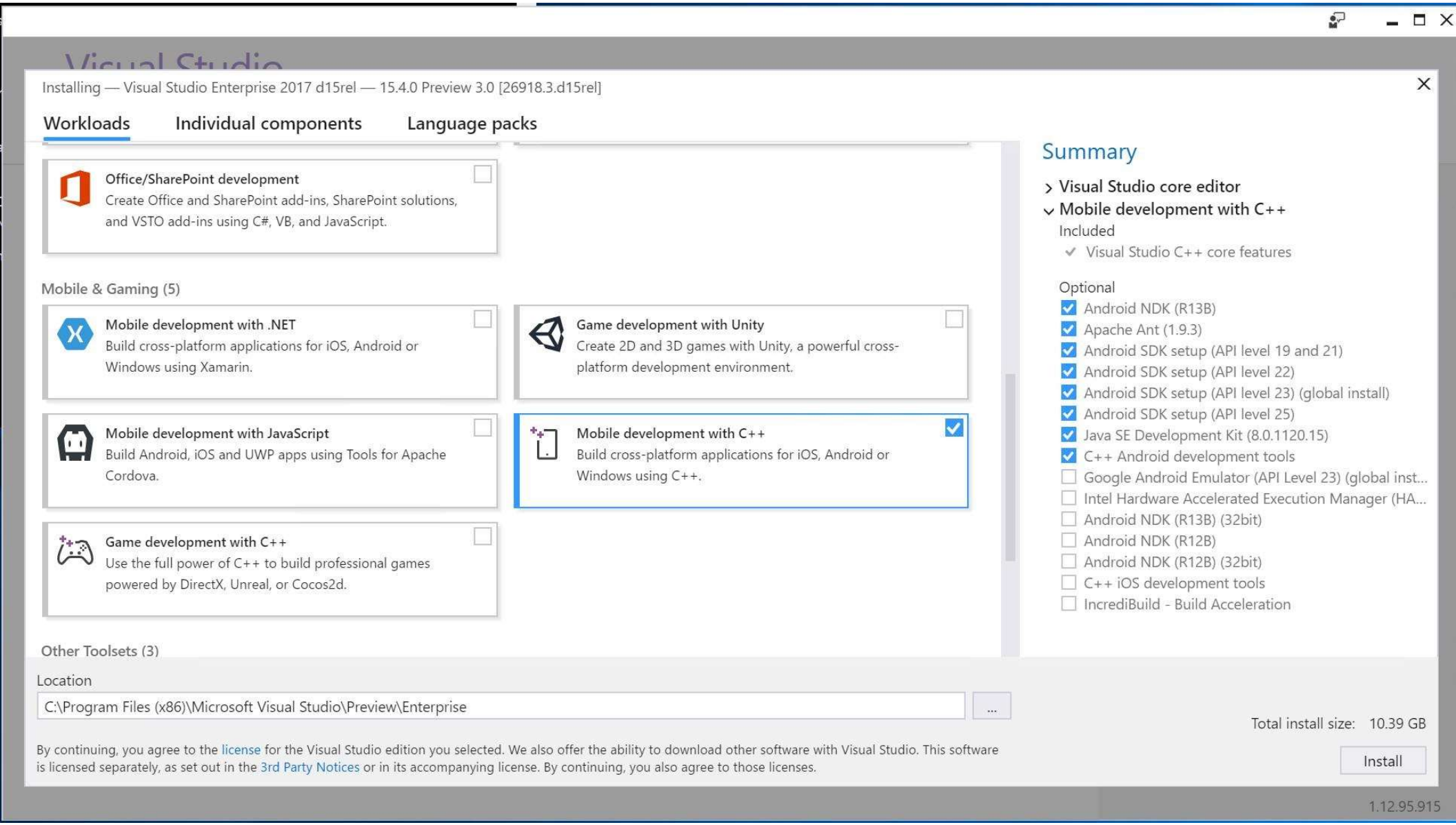
- Microsoft Visual C++ (MSVC) Compiler and Libraries
  - Conformance
  - Compiler Diagnostics
  - Code Analysis
  - Code Gen Quality
  - Build Throughput

- Visual Studio 2017
  - Faster installation and your disk will thank you
  - Pain-free upgrade
  - Just point Visual Studio to your code
  - Use Visual Studio for all your projects and target platforms
  - Be more productive than ever


Installing — Visual Studio Enterprise 2017 Preview — 15.4.0 Preview 2.0
Workloads
Individual components
Language packs
Windows (3)
Universal Windows Platform development
Create applications for the Universal Windows Platform with C#, VB, JavaScript, or optionally C++.
.NET desktop development
Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F#.
Desktop development with C++
Build classic Windows-based applications using the power of the Visual C++ toolset, ATL, and optional features like...
Web & Cloud (7)
ASP.NET and web development
Build web applications using ASP.NET, ASP.NET Core, HTML, JavaScript, and container development tools.
Azure development
Azure SDK, tools, and projects for developing cloud apps and creating resources.
Python development
Editing, debugging, interactive development and source control for Python.
Node.js development
Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.
Summary
Visual Studio core editor
Universal Windows Platform development
Included
Blend for Visual Studio
.NET Native
NuGet package manager
Universal Windows Platform tools
Windows 10 SDK (10.0.16278.0) for UWP: C#, VB, JS
Windows 10 SDK (10.0.15063.0) for UWP: C#, VB, JS
Optional
IntelliTrace
C++ Universal Windows Platform tools
Graphics debugger and GPU profiler for DirectX
Windows 10 Mobile Emulator (Creators Update)
Windows 10 SDK (10.0.14393.0)
Windows 10 SDK (10.0.10586.0)
Windows 10 SDK (10.0.10240.0)
Architecture and analysis tools
Location
C:\Program Files (x86)\Microsoft Visual Studio\Preview\Enterpris
...
Installation nickname
2
Total install size: 3.96 GB
By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.
Install

Windows, Windows Store

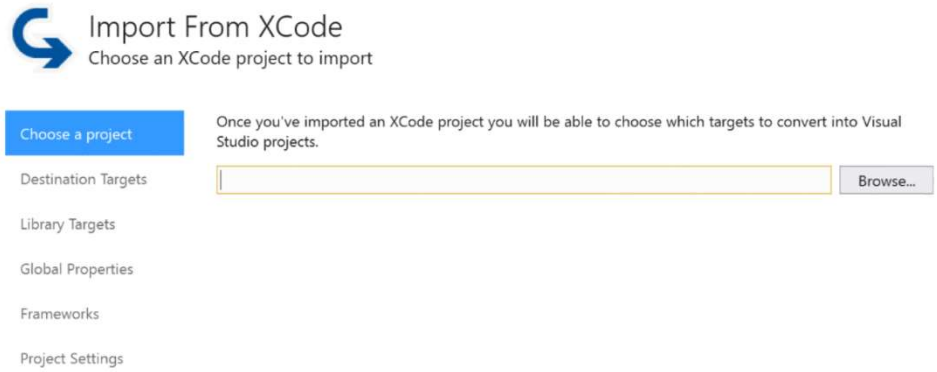
- Universal Windows Platform (UWP)
  - New way to target Win10 Desktop, Mobile, Xbox, and HoloLens from a single binary
  - Accelerated OpenGLES 1, 2 and 3 support through Angle OSS library
  - Large game deployments to Xbox
  - C++/WinRT - a language-extension-free WinRT projection for C++ (https://github.com/Microsoft/cppwinrt)
- Bridges to Windows Store
  - Desktop Bridge “Centennial” - Packaging your desktop apps in the Store
  - Windows Bridge for iOS “Islandwood” (https://github.com/Microsoft/WinObjC)
- Share source code across Windows Desktop, UWP, Android, iOS, *and* Linux


Visual Studio
Installing — Visual Studio Enterprise 2017 d15rel — 15.4.0 Preview 3.0 [26918.3.d15rel]
Workloads
Individual components
Language packs
Office/SharePoint development
Create Office and SharePoint add-ins, SharePoint solutions, and VSTO add-ins using C#, VB, and JavaScript.
Mobile & Gaming (5)
Mobile development with .NET
Build cross-platform applications for iOS, Android or Windows using Xamarin.
Game development with Unity
Create 2D and 3D games with Unity, a powerful cross-platform development environment.
Mobile development with JavaScript
Build Android, iOS and UWP apps using Tools for Apache Cordova.
Mobile development with C++
Build cross-platform applications for iOS, Android or Windows using C++.
Game development with C++
Use the full power of C++ to build professional games powered by DirectX, Unreal, or Cocos2d.
Other Toolsets (3)
Summary
Visual Studio core editor
Mobile development with C++
Included
Visual Studio C++ core features
Optional
Android NDK (R13B)
Apache Ant (1.9.3)
Android SDK setup (API level 19 and 21)
Android SDK setup (API level 22)
Android SDK setup (API level 23) (global install)
Android SDK setup (API level 25)
Java SE Development Kit (8.0.1120.15)
C++ Android development tools
Google Android Emulator (API Level 23) (global inst...
Intel Hardware Accelerated Execution Manager (HA...
Android NDK (R13B) (32bit)
Android NDK (R12B)
Android NDK (R12B) (32bit)
C++ iOS development tools
IncrediBuild - Build Acceleration
Location
C:\Program Files (x86)\Microsoft Visual Studio\Preview\Enterprise
...
Total install size: 10.39 GB
By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.
Install
1.12.95.915

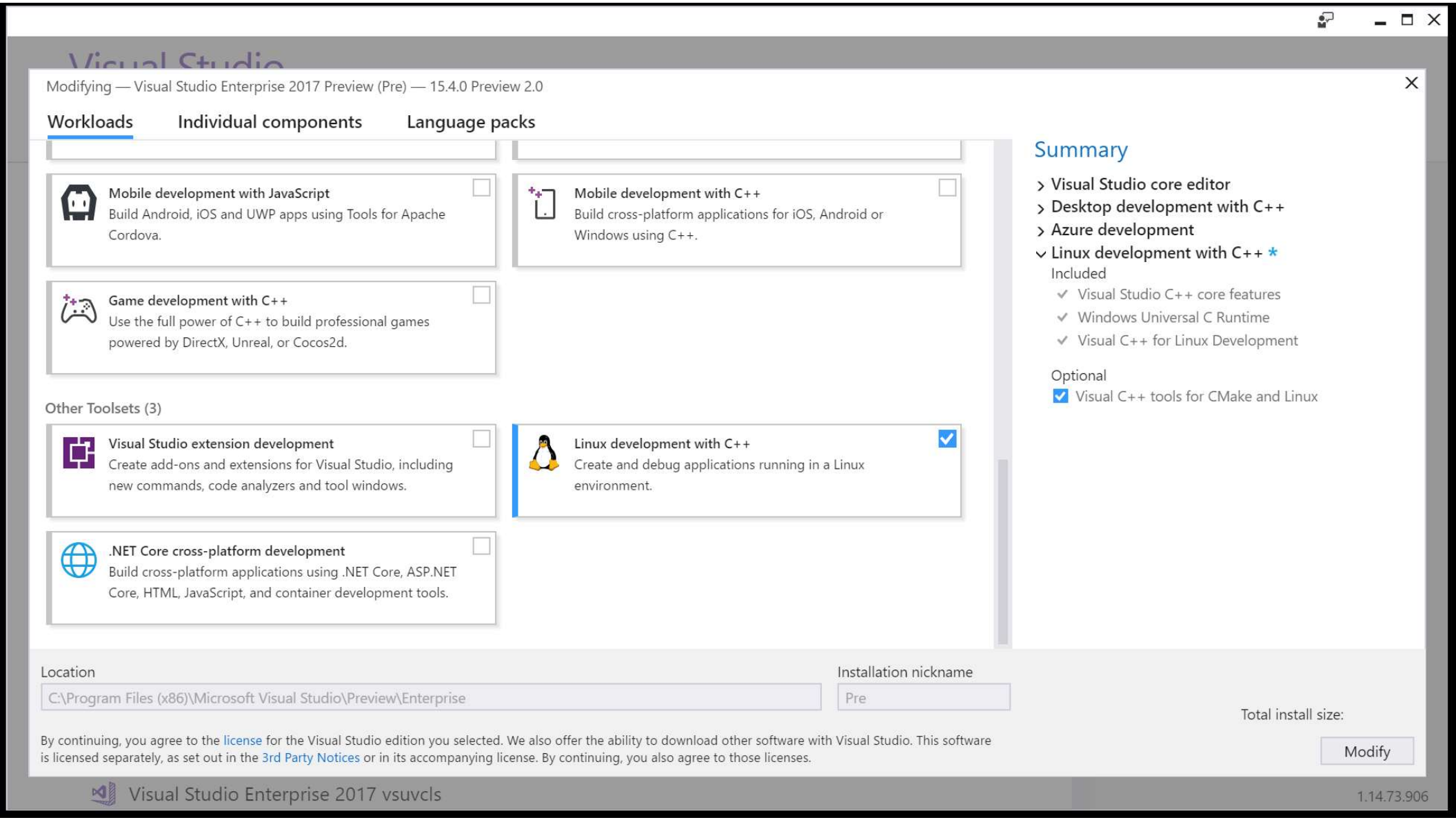
Android and iOS

- Android
  - C++ IntelliSense and debugging
  - Java IntelliSense and debugging
  - Support for Android Gradle build system
  - Built-in support for Android API level 25 and NDK r15c in upcoming VS updates

- iOS
  - Easily import (and roundtrip) your Xcode project into Visual Studio

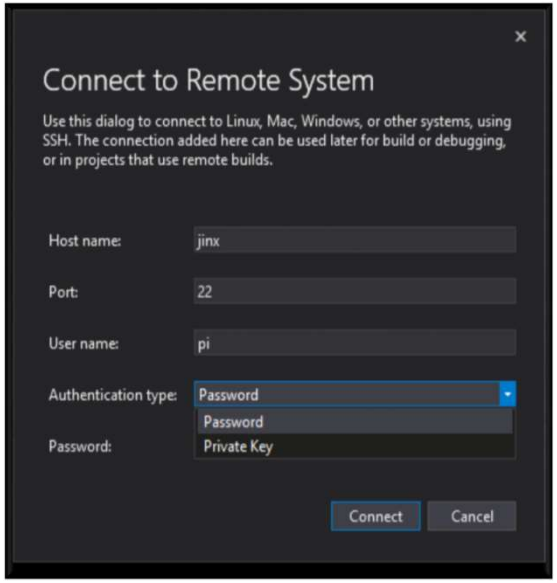


https://visualstudio.com/vs/cplusplus-mdd/


Visual Studio
Modifying — Visual Studio Enterprise 2017 Preview (Pre) — 15.4.0 Preview 2.0
Workloads
Individual components
Language packs
Mobile development with JavaScript
Build Android, iOS and UWP apps using Tools for Apache Cordova.
Mobile development with C++
Build cross-platform applications for iOS, Android or Windows using C++.
Game development with C++
Use the full power of C++ to build professional games powered by DirectX, Unreal, or Cocos2d.
Other Toolsets (3)
Visual Studio extension development
Create add-ons and extensions for Visual Studio, including new commands, code analyzers and tool windows.
Linux development with C++
Create and debug applications running in a Linux environment.
.NET Core cross-platform development
Build cross-platform applications using .NET Core, ASP.NET Core, HTML, JavaScript, and container development tools.
Summary
Visual Studio core editor
Desktop development with C++
Azure development
Linux development with C++ *
Included
Visual Studio C++ core features
Windows Universal C Runtime
Visual C++ for Linux Development
Optional
Visual C++ tools for CMake and Linux
Location
C:\Program Files (x86)\Microsoft Visual Studio\Preview\Enterprise
Installation nickname
Pre
Total install size:
By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.
Modify
Visual Studio Enterprise 2017 vsuvcls
1.14.73.906

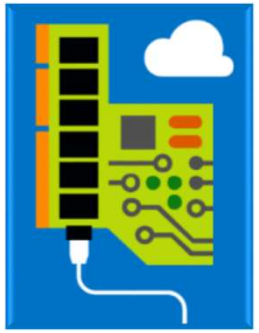
Linux

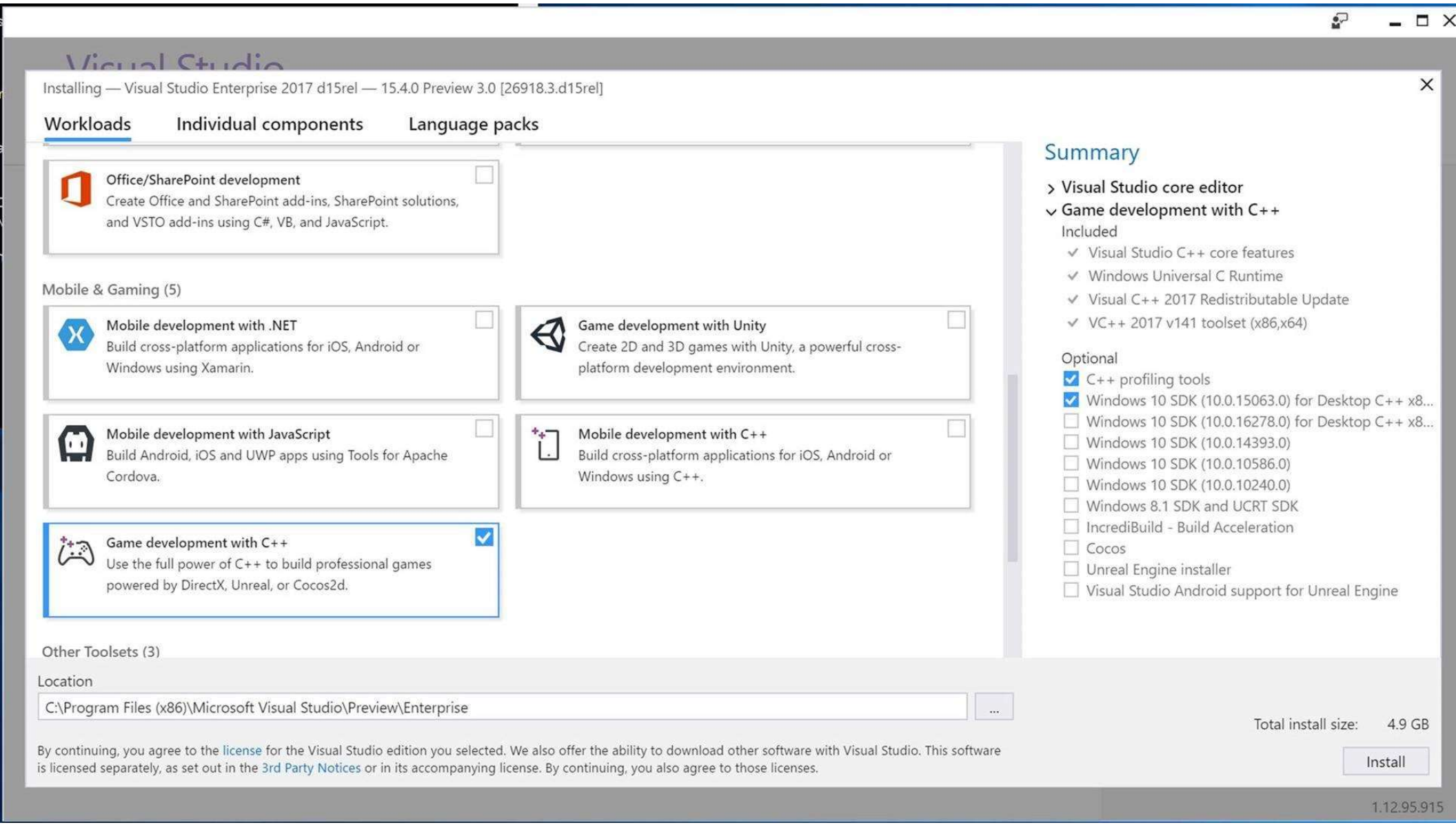
- Use Visual Studio with any Linux distro or Windows Subsystem for Linux (WSL)
  - Remote system needs SSH, GDB, and GCC for compile
  - Connect using user/password or private key
  - Project templates enable control of GCC/GDB on remote target
  - IntelliSense supports GCC with standard Linux libraries out of the box
  - Debug from your projects or attach to remote process
    - Use either gdb or gdbserver on the remote
    - Python pretty printer type visualizers supported in gdb mode
  - Support for CMake > 3.8 added in 15.4
- Resources
  - Documentation: https://aka.ms/vslinux
  - Issues, discussion: https://github.com/microsoft/vslinux



IoT – Internet of Things

- C++ for Linux Development works with IoT devices running Linux
  - E.g. Raspberry Pi, Beaglebone
  - Yocto SDKs can be used by overriding project defaults
- Visual Studio 2017 version 15.5 introduces ARM GCC support
  - Local cross compile use ARM compiler
  - mbed folder based project support
  - Launch templates provided to illustrate how to debug devices
- The Azure IoT SDK supports Linux and MCU devices
  - Portable to very small devices
  - Provides message processing and device management capabilities
  - Any device with network capability can get messages to/from Azure
  - Devices without network capability can communicate through gateways
  - https://azure.microsoft.com/develop/iot/




Visual Studio
Installing — Visual Studio Enterprise 2017 d15rel — 15.4.0 Preview 3.0 [26918.3.d15rel]
Workloads
Individual components
Language packs
Office/SharePoint development
Create Office and SharePoint add-ins, SharePoint solutions, and VSTO add-ins using C#, VB, and JavaScript.
Mobile & Gaming (5)
Mobile development with .NET
Build cross-platform applications for iOS, Android or Windows using Xamarin.
Game development with Unity
Create 2D and 3D games with Unity, a powerful cross-platform development environment.
Mobile development with JavaScript
Build Android, iOS and UWP apps using Tools for Apache Cordova.
Mobile development with C++
Build cross-platform applications for iOS, Android or Windows using C++.
Game development with C++
Use the full power of C++ to build professional games powered by DirectX, Unreal, or Cocos2d.
Other Toolsets (3)
Summary
Visual Studio core editor
Game development with C++
Included
Visual Studio C++ core features
Windows Universal C Runtime
Visual C++ 2017 Redistributable Update
VC++ 2017 v141 toolset (x86,x64)
Optional
C++ profiling tools
Windows 10 SDK (10.0.15063.0) for Desktop C++ x8...
Windows 10 SDK (10.0.16278.0) for Desktop C++ x8...
Windows 10 SDK (10.0.14393.0)
Windows 10 SDK (10.0.10586.0)
Windows 10 SDK (10.0.10240.0)
Windows 8.1 SDK and UCRT SDK
IncrediBuild - Build Acceleration
Cocos
Unreal Engine installer
Visual Studio Android support for Unreal Engine
Location
C:\Program Files (x86)\Microsoft Visual Studio\Preview\Enterprise
...
Total install size: 4.9 GB
By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.
Install
1.12.95.915

Games

- Use the cloud to build, launch, and scale out your games
  - https://azure.microsoft.com/solutions/gaming/
- Built-in graphics debugger and profiler for DirectX 10, 11, 12 for diagnosing graphics issues and performance bottlenecks
- Use Visual Studio to build cross-platform games with popular game engines
  - Unity, Unreal engine, and Cocos

https://www.visualstudio.com/features/game-development-vs

AGENDA

- Microsoft Visual C++ (MSVC) Compiler and Libraries
  - Conformance
  - Compiler Diagnostics
  - Code Analysis
  - Code Gen Quality
  - Build Throughput

- Visual Studio 2017
  - Faster installation and your disk will thank you
  - Pain-free upgrade
  - Just point Visual Studio to your code
  - Use Visual Studio for all your projects and target platforms
  - Be more productive than ever

Productivity

- Code Editing
  - Predictive IntelliSense, IntelliSense filtering, Code formatting enforcement with editorconfig
  - Coming soon: Ctrl + Click to Go To Definition, Structure Visualizer in-box (download the extension in the meantime)
- Code Navigation
  - Manage a large list more easily with Find All References (better perf coming in 15.5)
  - Navigate To -> Go To with filtering
  - Improved Error List results
- Source Control Management
  - Force push your changes, SSH support for remotes, View Commit Diff
- Debugging/Diagnostics
  - Run to Click, Reattach to Process, Improved Exception Helper, Break-on-exception conditions, Improved Memory and CPU profiler
- Unit Testing
  - Test Adapter for Google Test
  - Test Adapter for Boost.Test
  - …

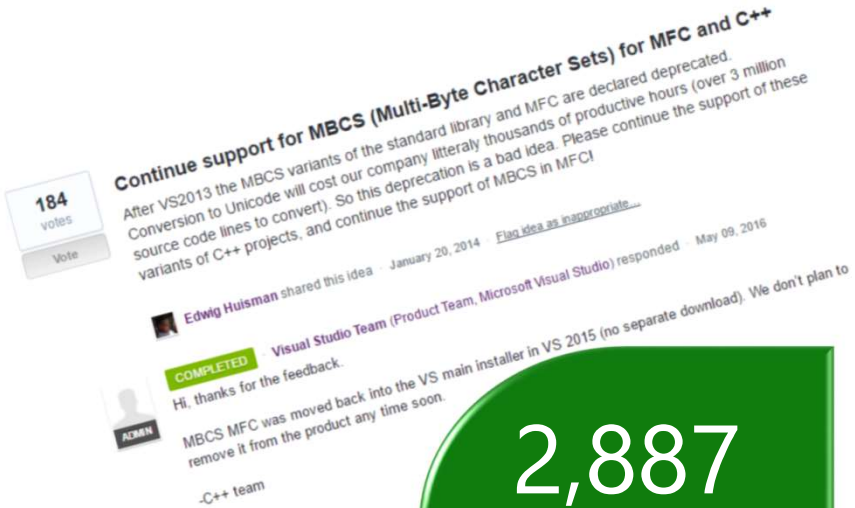
Performance

- Auto-precompiled headers for IntelliSense perf (when no PCH)
- Memory usage during debugging significantly decreased
  - E.g. debugging a specific problematic project down to 1.9G from… crashing
- Solution load dramatic improvements

Chromium - 4600 Solution Items	VS 2015 Update 3	VS 2017	Improvement
Time - First Solution Open (s)	1,213	182	6.6x
Time – Subsequent Solution Open (s)	1,211	68	17.8x
Private Working Set (MB)	2,293	804	2.8x
Virtual Memory (MB)	3,066	1,302	2.3x

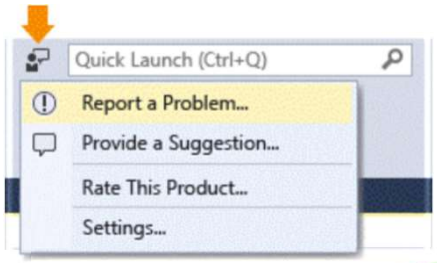
You Helped Us Build It!

Visual Studio User Voice
https://visualstudio.uservoice.com/



2,887
Votes
on UserVoice

"Report a Problem…" tool
https://developercommunity.visualstudio.com



1,376
fixed bugs
publicly reported through
Connect & IDE

Summary

- Visual Studio Code
  - Code editor redefined, optimized for editing and debugging your C/C++ code
- MSVC - Visual C++ toolset (compiler and libs)
  - The obvious choice on Windows
- Visual Studio 2017
  - Fast and easy workload installation, Pain-Free Upgrade, Open Folder
  - Performance you can feel
  - Most productive IDE for your editing, building, debugging
- Any C++ developer, building any type of app
  - No matter what platform you are targeting
- Microsoft
  - We are listening and participating, tell us what you want to see next