Contents

- Visual Studio 2017
  - Visual Studio Roadmap
  - Current Release Notes
  - Preview Release Notes
  - Release Notes History
  - Release Rhythm
  - Distributable Code
  - Platform Compatibility
  - System Requirements
  - License Terms
- Visual Studio 2017 for Mac
  - Current Release Notes
  - Preview Release Notes
  - Distributable Code
  - Platform Compatibility
  - System Requirements
  - Introducing Visual Studio for Mac
  - Servicing
- Visual Studio 2015
- Visual Studio 2013
- Visual Studio 2012
- Visual Studio 2010
- Visual Studio Support Resources
  - Support Page
  - Installation Guide
  - Troubleshooting Install Issues
  - Product Lifecycle and Servicing
  - Documentation
- Team Foundation Server 2018

- Update 3 Release Notes
- Update 2 Release Notes
- Update 1 Release Notes
- RTW Release Notes
- Release Notes History
- SHA-1 Values

Team Foundation Server 2017

- Update 3 Release Notes
- Update 2 Release Notes
- Update 1 Release Notes
- RTW Release Notes
- Release Notes History
- SHA-1 Values

Team Foundation Server 2015

- Update 4 Release Notes
- Update 3 Release Notes
- Update 2 Release Notes
- Udpate 1 Release Notes
- RTW Release Notes
- Release Notes History

10/2/2018 • 43 minutes to read

Visual Studio 2017 version 15.8 Release Notes

NOTE

If you are accessing this page from a non-English language version, and want to see the most up-to-date content, please visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

Click a button to download the latest version of Visual Studio 2017. For instructions on installing and updating Visual Studio 2017, see Update Visual Studio 2017 to the most recent release. Also, see instructions on how to install offline.

Download Community 2017 | Download Professional 2017 | Download Enterprise 2017

Visit the visualstudio.com/downloads page to download other Visual Studio 2017 products.

What's New in 15.8

The Visual Studio Blog is the official source of product insight from the Visual Studio Engineering Team. You can find in-depth information about the Visual Studio version 15.8 releases in the following posts:

- Visual Studio 2017 version 15.8 -- *Read the latest blog post!*

Visual Studio 2017 version 15.8 Releases

- October 02, 2018 -- Visual Studio 2017 version 15.8.6 Servicing Update New
- September 20, 2018 -- Visual Studio 2017 version 15.8.5 Servicing Update
- September 11, 2018 -- Visual Studio 2017 version 15.8.4 Servicing Update
- September 6, 2018 -- Visual Studio 2017 version 15.8.3 Servicing Update
- August 28, 2018 -- Visual Studio 2017 version 15.8.2 Servicing Update
- August 17, 2018 -- Visual Studio 2017 version 15.8.1 Servicing Update
- August 14, 2018 -- Visual Studio 2017 version 15.8 Minor Release

IMPORTANT

Visual Studio 2017 version 15.8 Security Advisory Notices

- September 11, 2018 -- Visual Studio 2017 version 15.8.4 Servicing Update
- August 14, 2018 -- Visual Studio 2017 version 15.8
- July 10, 2018 -- Visual Studio 2017 version 15.8 Preview 4
- June 26, 2018 -- Visual Studio 2017 version 15.8 Preview 3
- May 08, 2018 -- Visual Studio 2017 version 15.8 Preview 1

Summary of Notable New Features in 15.8

- The installer now offers the option to download all files before starting installation.
- We improved performance during project unload/reload and branch switching.
- Solution load times can be improved by disabling automatic document restore.
- We made significant test performance improvements while running large solutions with multiple test projects.
- Visual Basic now provides you with a significant performance improvement.
- Performance Profiling now offers the ability to pause/resume data collection and a new .NET Object Allocation Tracking tool has been added.
- We made improvements to the CPU Usage tool for Performance Profiling.
- We added new productivity features, such as code cleanup, invert-if refactoring, Go to Enclosing Block, Multi-Caret support, and new keyboard profiles.
- You can choose your target instance when debugging extensions.
- Launch the Snapshot Debugger directly from the publish summary page.
- We introduced F# 4.5, a new language version that fully supports `Span<'T>` and contains significant improvements to stack traces for async code.
- We improved performance and added new features in F# tools, such as **Ctrl+Click** to **Go to Definition**.
- Enhancements to C++ toolset include improvements to the SSA optimizer and linker.
- C++ cross-platform development additions include ClangFormat updates, and configuration templates in CMake and Open Folder for MinGW, Linux, and Windows.
- Changes to C++ productivity include Template IntelliSense, Quick Info tooltips on macros, convert to constexpr lightbulbs, in-editor code analysis squiggles, and more.
- C++ debugging improvements include **Just My Code** updates and new data breakpoints.
- There are numerous JavaScript and TypeScript improvements and features.
- You can now manage client-side libraries in your web projects.
- We added a new single project Docker container experience for ASP.NET Core web projects.
- You can now customize the tag for a Docker image with the publish improvements made in this release.
- Xamarin improvements include support for Xcode 9.4 and smarter Android incremental builds.
- You can use Google's Android emulator side-by-side with Hyper-V on Windows 10 April 2018 Update.
- We added a split-view editor for the Xamarin.Android designer.
- The Xamarin.Forms Previewer now has toolbox support, and we moved it to a split-view editor.
- You can now use cross-language debugging with Python 3.7.0rc1.
- The Migrate local Azure Function settings to Azure now displays new values.
- With added support for Azure Functions, you now have a new target host in the Configure Continuous Delivery to Azure dialog.
- Test Explorer summary pane improvement now provides more informative test status.
- .NET Test Adapter Extensions: breaking change and deprecation.
- We added native support for the Language Server Protocol.
- You now have secrets support for ASP.NET .NET Framework projects.

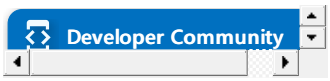
- Visual Studio now offers .NET Framework 4.7.2 development tools to supported platforms with 4.7.2 runtime included.
- Async autoload packages are delayed until startup and solution load complete.
- The Build Tools Visual Studio 2017 now support Workflow and includes the VSSDK.
- Git and TFS status now update properly for external file changes in .NET Core projects.
- .NET Core SDK 2.1.400 is included.

Top Issues Fixed in 15.8

These are the customer-reported issues addressed in 15.8:

- SSDT not present in Visual Studio Build Tools 2017.
- Anaconda fails to uninstall during upgrade (Error 87).
- Upgrade to 15.7 generates error: *PackageId:Anaconda3.Exe.x64;PackageAction:Uninstall;ReturnCode:87;*.
- Visual Studio update generates *UWP BuildError at MakePri 0x80073b0f and 0xdef00532* errors at my Solution.
- VS 2017 has problems with the ASP.NET Core export template feature.
- Test Explorer (SpecFlow tests): goes to the feature class file not the feature file.
- Project template exported from a C++ project only contains a vcxproj and vcxproj.filters, no source or headers.
- Bad codegen on x64.
- Right-click context menu on a Solution Folder does nothing in Enterprise 15.8 Preview 2.
- VS2017 doesn't remember screen position and size for use on 3 monitors.
- UTF8-no-BOM causes diff window to display erroneous encoding warning.
- Double-clicking a word only selects to the start of the word, not the entire word.
- VS2015 and VS2017 create an empty Debug folder in the path defined by $(SolutionDir) despite both $(OutDir) and $(IntDir) using completely different settings.
- If you create an MFC dialog-based project in Japanese environment with VS2017, editing the dialog will cause errors in the build.
- Download Progress Text display on Visual Studio Installer.
- Right click executable project in Solution Explorer -> Debug -> Start new instance picks random project from solution to start.
- DEVENV Command line build fails with: *This operation may only take place in the UI thread*.
- Find and replace remembers file types.
- Error messages are unreadable in Package manager console because of font color and background color.
- Visual Studio Community 2017 15.4.1 Web publish preview fails with not authorized.

See all customer-reported issues fixed in Visual Studio 2017 version 15.8.



Details of What's New in 15.8

Visual Studio 2017 version 15.8.0

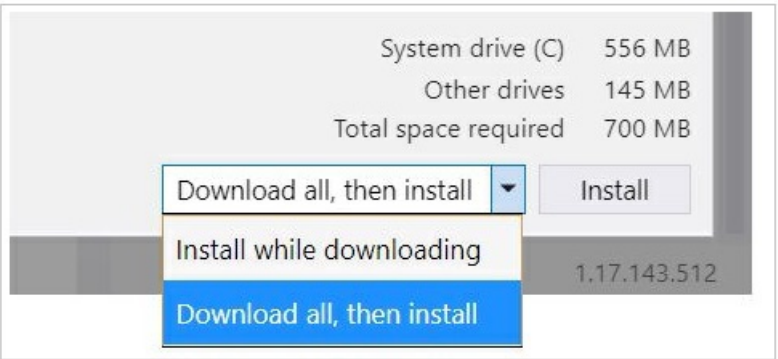
released on August 14, 2018

New Features in 15.8

Install

You now have the option to download all installation files before starting your installation.

- To use this new option, select the "Download all, then install" option in the installer *(Figure 1)*. We suggest this option if you have a slower internet connection.
- The default option remains "Install while downloading", which downloads and installs in parallel.



(Figure 1) Download Option

Performance

This release contains the following performance improvements:

- Branch switching for C#, VB, and C++ projects is much faster for large solutions since solution reload is no longer required.
- We reduced the time to unload and reload a small set of C# and VB projects in large solutions from minutes to several seconds.
- We added an option to disable reopening documents that were open in the previous session, as reopening certain types of files or designers can delay solution load.
  - Toggle this option in **Tools > Options > Projects > Solutions > General**.

Test Performance

We significantly improved performance when running a few tests in a large solution with multiple test projects. In our labs, a solution with over 10,000 MSTests executed a single test up to 82% faster!

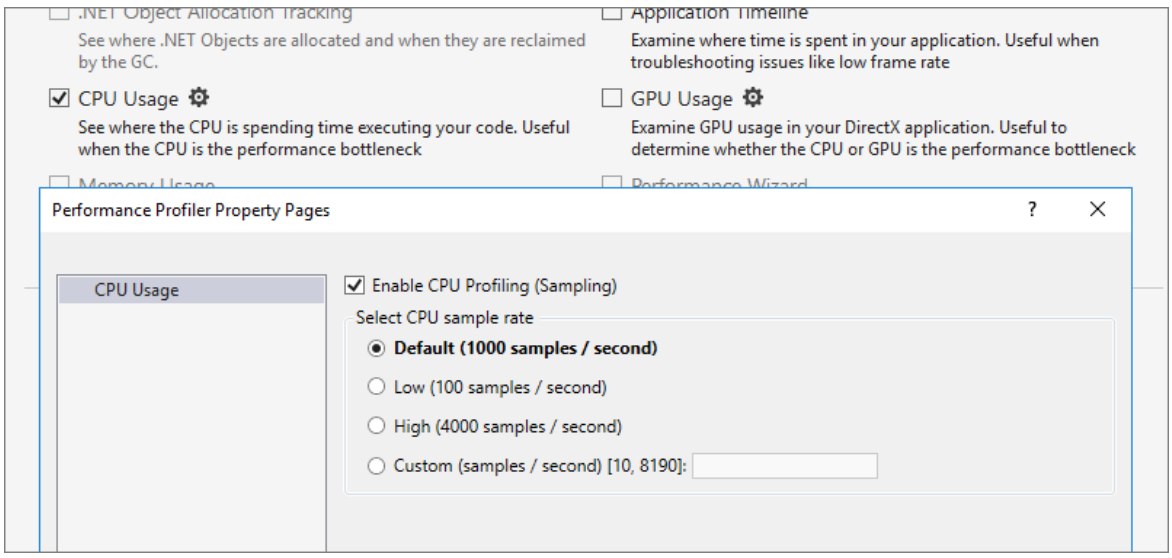
Visual Basic Performance Improvement

Visual Basic now provides a significant performance improvement when using the pattern `CInt(Fix(number))` to convert from non-integer types to integers.

Performance Profiling

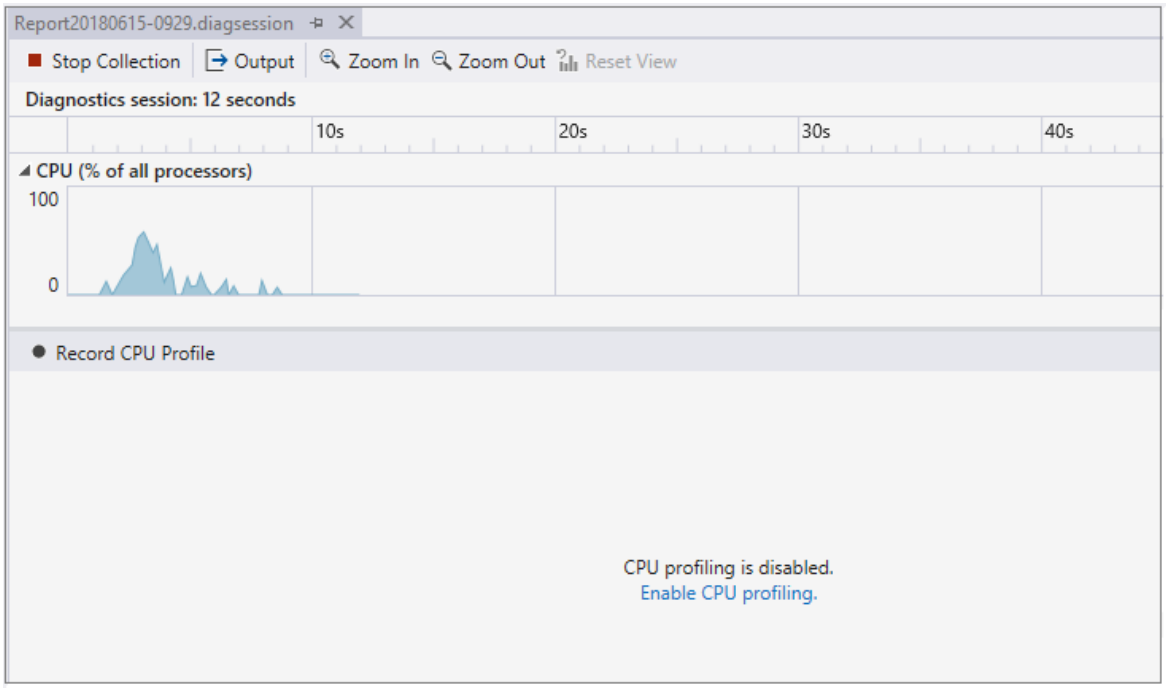
This release contains the following performance profiling enhancements:

- We added the capability to start profiling in a paused state:
  - The CPU Usage tool in the Performance Profiler (**ALT-F2**) can now be started in a paused state *(Figure 2)*. This can be useful while waiting for a scenario worthy of a CPU Usage investigation. If CPU usage collection is disabled at startup, the Performance Profiler will not collect any CPU usage sample stack data until it is specifically enabled. This results in a smaller amount of data to collect and analyze, thus making your performance investigations more efficient.
  - To start a session with CPU Usage sample collection disabled, click the gear icon next to **CPU Usage** on the Performance Profiler launch page. On the CPU Usage property page, uncheck **Enable CPU Profiling (Sampling)** and click **OK** to save the settings.



(Figure 2) CPU Usage Tool Settings

- Once you start the target application (click **Start** on the Performance Profiler launch page), you will see the CPU utilization graph *(Figure 3)* which allows you to control CPU profiling. To enable CPU usage sample stack data collection, select **Enable CPU profiling** in the middle of the view, or click **Record CPU Profile** just below the CPU utilization graph. You can click **Record CPU Profile** to enable/disable sample data collection as many times as you like. The CPU utilization graph color changes to indicate whether sample collection is enabled/disabled at that point in time.

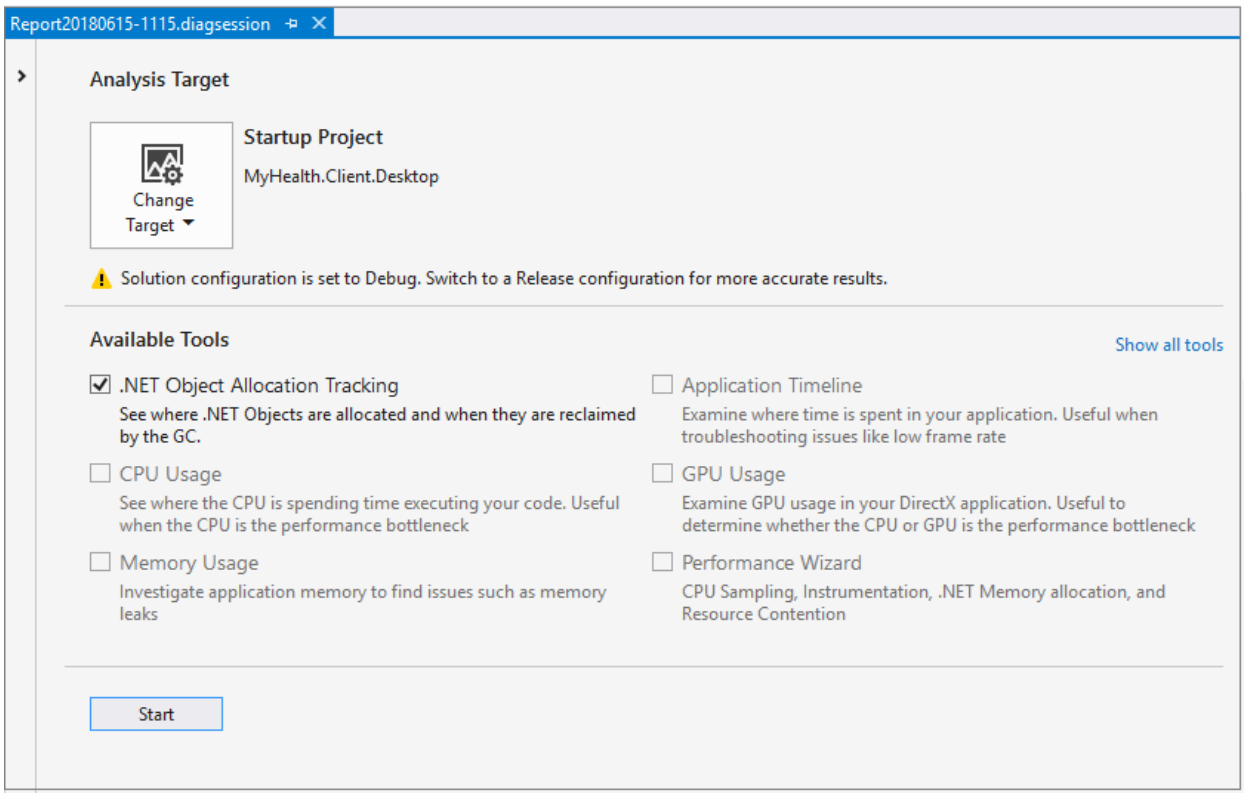


(Figure 3) CPU Usage Utilization Graph

- We added a .NET Object Allocation Tracking tool:
  - The .NET Object Allocation Tracking Tool joins the family of tools available from the Performance Profiler. Invoking this tool for a performance profiler session initiates the collection of a stack trace for every .NET object allocation that occurs in the target application. This stack data is analyzed along with object type and size information to reveal details of the memory activity of your application. You can quickly determine the allocation patterns in your code and identify anomalies. In addition, for Garbage Collection (GC) events, you can easily determine which objects were collected and retained and quickly identify object types that dominate the memory usage of the application.
  - This is especially useful for API writers to help minimize allocations. Many applications will overrun the

buffer limits involved in the diagnostics data collection, but small test applications exercising an API's key scenarios can be diagnosed quite well. While your test application is executing, the Performance Profiler displays a line graph of Live Objects (count), as well as an Object Delta (% change) bar graph.

- To use the .NET Object Allocation Tracking Tool, display the Performance Profiler launch page *(Figure 4)*, select a target to profile (the default target is the startup project in the solution), check **.NET Object Allocation Tracking** under Available Tools, then click **Start**.



(Figure 4) Performance Profiler Launch Page

Performance Profiling (CPU Usage)

This release contains the following improvements to the CPU Usage tool of the Performance Profiler (available through **ALT-F2**):

- The Call Tree view now displays asynchronous execution by logical call stack by default. You can turn off this behavior by unchecking the option **Stitch Async Code** in the Filter dropdown of the CPU Usage main view.
- We added a Modules/Functions view that displays performance information by module (dll) and by function within a module. One can display the Modules/Functions view from the context menu available when selecting a function in the CPU Usage main view, or from the View dropdown in the Call Tree or Caller/Callee views.
- Instance indication has been added to the CPU Usage graph in the CPU Usage tool's main view. You can view the instances when a function is executing (for instance, on the stack) by double-clicking a function listed in any of the CPU Usage views.

Productivity

This release contains the following productivity enhancements:

- You can perform additional code cleanup with **Format Document** (**Ctrl + K, D** or **Ctrl + E, D**) for C# development. Configure cleanup by going to **Tools** > **Options** > **Text Editor** > **C#** > **Code Style** > **Formatting** > **General**.
- We added more refactorings and quick actions using **Ctrl + .** or **Alt + Enter**:
  - *Invert If* enables you to invert your logic in if-else statements. Place your cursor in the `if` keyword to trigger this refactoring.
  - *Add parameter from method callsite* allows you to add a parameter to a method by adding an argument to a method callsite and triggering *Quick Actions and Refactorings*.

- *Remove unnecessary parentheses* removes parentheses around binary operators that are not essential for compilation. You can configure this style rule through **Tools** > **Options** > **Text Editor** > **C#** > **Code Style** > **General** or .editorconfig:
  - `dotnet_style_parentheses_in_arithmetic_binary_operators`
  - `dotnet_style_parentheses_in_relational_binary_operators`
  - `dotnet_style_parentheses_in_other_binary_operators`
  - `dotnet_style_parentheses_in_other_operators`
- *Use ternary conditionals in assignments and return statements* can also be configured as a style rule in **Tools** > **Options** > **...** or through .editorconfig:
  - `dotnet_style_prefer_conditional_expression_over_assignment`
  - `dotnet_style_prefer_conditional_expression_over_return`

- We added new commands and improvements to the **Go to All** window:
  - Go to Enclosing Block (**Ctrl + Alt + UpArrow**) allows you to quickly navigate up to the beginning of the enclosing code block.
  - Go to Next/Previous Issue (**Alt + PgUp/PgDn**) allows you to skip to the next/previous issue (error, squiggle, lightbulb).
  - Go to Member (**Ctrl + T, M**) is now scoped to the file by default. You can change the default back to solution by toggling the Scope to Current Document (**Ctrl + Alt + C**).
- You can now create insertion points and selections at multiple, arbitrary locations in a file with multiple caret support. This allows you to add, edit, or delete text in multiple places simultaneously.
  - Insert carets with **Ctrl + Alt + LeftMouseClick**.
  - Add a selection and caret at next location that matches current selection with **Shift + Alt + Ins**.
  - See **Edit > Multiple Carets** for full list of actions.
- Access a contextual navigation menu with **Alt + `**.
- Keep your keybindings consistent with two new keyboard profiles: *Visual Studio Code* and *ReSharper (Visual Studio)*. You can find these schemes under **Tools > Options > Environment > Keyboard** and the top drop-down menu.

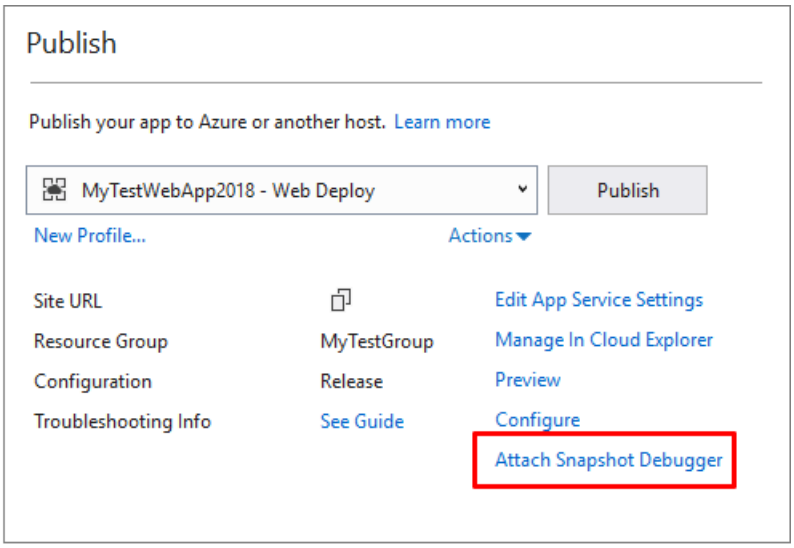
Debugging

This release contains the following debugging enhancements:

- When you have more than one instance of Visual Studio 2017 installed, you can now select which instance to deploy your extension to when debugging *(Figure 5)*. That way you can, for example, develop in the Visual Studio release channel while debugging in the preview channel.

(Figure 5) Select Debug Instance

- You can now attach the Snapshot Debugger directly from the Publish summary page *(Figure 6)*.



(Figure 6) Attach Snapshot Debugger from Publish Summary

- We now support viewing Managed Tasks through the Tasks window while debugging minidumps with heap.

Tools for Universal Windows Platform Developers

We made many important improvements to the XAML designer for projects targeting the Universal Windows Platform with a target platform version of the Windows 10 Fall Creators Update (build 16299) or later. These improvements include:

- You can now edit collections in the Property Inspector.
- The designer now allows Templates and Styles to be edited, including when the definitions for those entities are defined in other documents.
- Properties of type IconElement (such as Icon in an AppBarButton) now have a custom editor in the Property Inspector making these properties easier to set.
- The designer, editor, and Edit and Continue should all now work correctly with x:DefaultBindMode.
- The Visual State Manager experience in Blend now supports AdaptiveTrigger.

F# 4.5 and F# Tools for Visual Studio

We introduced the F# language version 4.5 with this release. This also corresponds with the new 4.5.x family of FSharp.Core (the F# core library). You can read the specs for each of these changes in the F# RFC repository. There are also many improvements to F# tools for Visual Studio with this release.

F# 4.5

Here are the some of the highlights with the F# language version 4.5:

Span<'T> support

We implemented `Span<'T>` and related features so that effective consumption and production of APIs using `Span`, `Memory`, and `ref`-like constructs are possible with F#. The features for this include:

- New `voidptr` type.
- New `NativePtr.ofVoidPtr` and `NativePtr.toVoidPtr` functions in FSharp.Core.
- New types `inref<'T>` and `outref<'T>`, which are read-only byrefs and write-only byrefs, respectively. This correspond to `in ref` and `out ref` in C#.
- The ability to produce `ByRefLike` structs (such as `Span` and `ReadOnlySpan`).
- The ability to produce `IsReadOnly` structs.
- Implicit dereference of `byref`- and `inref`-returns from methods.
- The ability to produce extension members on `byref` / `inref` / `outref`.

This feature set fixes a bug in the initial design of `byref`-returns in F# 4.1, where functions, methods, and properties returning `byref`'s were not implicitly dereferencing the return value. We make this change to bring the feature in-line with how C# handles `ref`-returns. An error message is used when a type annotation indicates that an implicit dereference of a `ref`- return is now used.

Additionally, this feature set also fixes a bug in the F# compiler where "Evil struct replacement" was possible; i.e., calling a method on an F# struct could replace the actual struct that was called with a different one. The `this` parameter on a struct is now considered an `inref<MyStruct>`, with an error that suggests you add a mutable field if you wish to modify the struct.

You can learn more about `span` and `ref`-like constructs in the RFC for this feature set.

Match! in computation expressions

F# 4.5 introduces `match!`, a new keyword for use within computation expressions, contributed entirely by John Wostenberg. This syntactic sugar is equivalent to a `let!` followed by a `match` on the result. You can learn more in the RFC for `match!`.

Relaxing the need to upcast with `yield` in sequence, list, and array expressions

F# 4.5 now relaxes some cases where an upcast when using `yield` was required to convert a subtype into a supertype. This restriction was already not necessary for these expressions since F# 3.1 when *not* using `yield`, so this makes things more consistent with existing behavior. You can learn more in the RFC for this feature.

Permitting indentation on list and array brackets

F# 4.5 now relaxes the indentation rule for list and array brackets that required them to be indented one scope forward when on their own line. This previous requirement has always been quite confusing, especially for beginners to F#. Additionally, it is not required for F# sequence expressions. This now brings array and list expressions to the same consistent state as sequence expressions. You can learn more in the RFC for this feature.

Enum cases emitted as public

F# 4.5 now emits enumeration cases as public under all circumstances, to align with how C# emits enumeration cases. This also makes it easier for profiling tools to analyze logs from F# code, where the value rather than the label name was emitted. You can learn more in the RFC for this feature.

F# Compiler Improvements

Improvements to the F# compiler in addition to the previously-mentioned language features are in F# 4.5. These include:

- We improved compiler performance work by removing up to 2.2% of all allocations in the F# compiler (under various scenarios).
- We fixed a bug that results in an `AccessViolatioNException` when using `yield!` with struct-based enumerables.
- It is now possible to inherit from `FSharpFunc` again.
- Tail calls are disabled by default for debug builds of F# for .NET Core. They are enabled for release, thus matching the desktop F# compiler.
- F# reference normalization has been fixed to allow you to control transitive assembly references written to an output file. This allows you to perform the equivalent of assembly redirection on .NET Core.
- We fixed a bug where the error message used when attempting to use dynamic invocation on inline functions was ignored. The error message now propagates.
- F# now respects the `WarningsNotAsErrors` flag that you can set in project files.
- When branches of a pattern match do not return the same error message type condition, has been updated to be friendlier by Isaac Abraham.
- An internal error bug when a compiling interface implementation that lacks an overloaded method implementation has been fixed by Steffen Forkmann.

- Some unnecessary array copying in the lexing phase of the compiler has been removed by Gauthier Segay.
- Incomplete pattern matches on F# enumerations now produces a detailed warning that gives an example of a case not covered, contributed by John Wostenberg.
- `#nowarn "2003" is now respected, contributed by Matthias Diitrich.
- A bug where consuming C# extensions methods could fail in F# overload resolution has been fixed by Steffen Forkmann.
- An internal QueueList data structure bug was fixed by Steffen Forkmann.
- Various smaller optimizations and code cleanup efforts were contributed by Steffen Forkmann, Eugene Auduchinok, and ncave.

F# Core Library 4.5.x

The following additions to the F# Core Library are now available:

- Significant work has been done to improve stack traces for `async { }` computation expressions. You should now be able to see user code and user line numbers in stack traces. You can learn more in the RFC for this feature.
- `FuncConvert.FromFunc` and `FuncConvert.FromAction` APIs that type `System.Func` and `System.Action` overloads, to help in interoperation with C#. You can learn more in the RFC for this feature.
- `ValueOption` is a new type available, which is the first in a set of future features ultimately aimed at better performance for Active Patterns. You can learn more in the RFC for this feature.
- `TryGetValue` is now a new member on the F# Map type. You can learn more in the RFC for this feature.
- We fixed High CPU usage upon the first invocation of `MailboxProcessor.TryReceive`.
- Comparison for `bool` now uses fast generic comparison, contributed by Vasily Kirichenko.
- The summary text for `Array.allPairs` has been updated to be correct, contributed by Patrick McDonald

F# Tooling Improvements

Significant improvements in the F# tools, such as performance enhancements and some new editor features are included this release. As always, with a large number of contributions from the F# open source community. Here are the highlights:

- We improved IntelliSense performance for .NET SDK-style projects of all forms, including those that use multi-targeting.
- A community-driven effort to analyze and improve IntelliSense performance for very large files was contributed by Vasily Kirichenko, Steffen Forkmann, and Gauthier Segay. IntelliSense in very large files (10k+ lines of code) is roughly twice as fast now.
- The warning for an outdated FSharp.Core (despite the package being installed) is no longer present in .NET SDK-style projects.
- The description tooltip that displays XML documentation for a member after `.` in IntelliSense no longer times out after 10 seconds.
- A bug where you could not set breakpoints in object constructor arguments has been fixed.
- A bug where a renamed symbol would be duplicated when it is a generic parameter has been fixed.
- Templates for .NET Framework (classic F# templates) now consume FSharp.Core from a NuGet package, to align with .NET SDK F# templates.
- Automatic, transactional brace completion is now available for `()`, `[]`, `{}`, `[||]`, and `[<>]` brace pairs. We did this work in collaboration with Gibran Rosa.
- You can now go to definition with **Ctrl + Click** on an F# symbol. The settings for this gesture are also respected in the **Tools > Options** window.
- The IntelliSense performance UI has been modified to allow configuration of stale typecheck information for various IDE features. Explanations for each option are now present in tooltips for the settings.
- Brace match highlighting now correctly highlights braces, completed in collaboration with Vasily Kirichenko.
- Go to definition now navigates correctly when a type is defined recursively, contributed by Vasily Kirichenko.

- A bug where an auto-imported namespace wasn't opened when the top of a file was empty has been fixed by Vasily Kirichenko.
- A bug where `printf` specifiers that contained dots were miscolored has been fixed by Vasily Kirichenko.
- A bug where all opens were considered unused inside of a recursive module has been fixed by Vasily Kirichenko.
- The performance of the Unused Opens analyzer has been improved significantly by Vasily Kirichenko.
- Autocompletion for attributes now only suggests options that are actually attributes, contributed by Vasily Kirichenko.
- Signature Help tooltips are now generated for Type Provider static parameters at the constructor call site, contributed by Vasily Kirichenko.
- A bug where value types used as units of measure were colored as reference types has been fixed by Vasily Kirichenko.
- A bug where semantic colorization could disappear for some files while scrolling has been fixed by Vasily Kirichenko.
- There is now an experimental CodeLens implementation, contributed by Victor Peter Rouven Müller. You can turn it on in **Options > Text Editor > F# > Code Lens**.
- A bug where the F# compiler service would incorrectly elide the module names in XML documentation has been fixed by Sebastian Urban.
- Code that uses `Dictionary` with `ContainsKey` and subsequent `Item` calls has been changed to use `TryGetValue`, by Eugene Auduchinok.
- Jakob Majoka also contributed in the process of consuming a different API for Tooltips.

Infrastructure, Packaging, and Open Source Improvements

We made the following enhancements to infrastructure, packaging, and our open source contribution experience:

- The F# compiler distributed with Visual Studio no longer installs as a singleton in the F# Compiler SDK location. It is now fully side-by-side with Visual Studio, meaning that side-by-side installations of Visual Studio will finally have truly side-by-side F# tooling and language experiences.
- The FSharp.Core NuGet package is now signed.
- ETW logging has been added to the F# tools and compiler.
- The very large `control.fs` / `control.fsi` files in FSharp.Core have been split into `async.fs` / `async.fsi`, `event.fs` / `event.fsi`, `eventmodule.fs` / `eventmodule.fsi`, `mailbox.fs` / `mailbox.fsi`, and `observable.fs` / `observable.fsi`.
- We added .NET SDK-style versions of our project performance stress test artifacts.
- We removed Newtonsoft.json from our codebase, and you now have one less package downloaded for OSS contributors.
- We now use the latest versions of System.Collections.Immutable and System.Reflection.Metadata.

C++ Conformance and Toolset Improvements

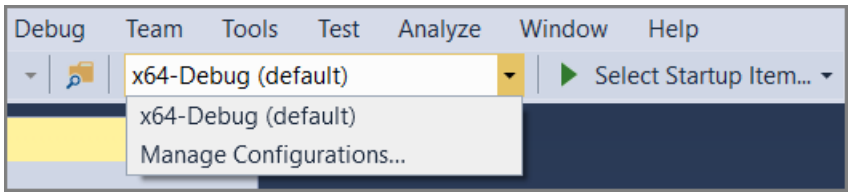
This release contains the following C++ conformance and toolset improvements:

- A new, experimental, token-based preprocessor that conforms to C++11 standards (including C99 preprocessor features), enabled with the /experimental:preprocessor switch. This is controlled with macro `_MSVC_TRADITIONAL`, which is defined 1 when using the traditional preprocessor and 0 when using the new experimental standards conformant preprocessor.
- The Visual Studio Developer Command Prompt supports enabling the Visual C++ Spectre variant 1 mitigated runtimes (`-vcvars_spectre_libs = spectre`). More information about Spectre mitigations is available on the Visual C++ Team Blog.
- Two new additions to the SSA Optimizer focused on modern C++ code generation: redundant store elimination and folding of redundant branches.
- Optimized memory mapped I/O performance in the linker to reduce link times.

C++ Cross-Platform Development

We added, improved, and added the following to C++ cross-platform development for this release:

- Added **Add** > **New Item** template for generating a .clang-format file following the coding convention specified for ClangFormat in **Tools** > **Options**. If the Visual Studio convention is selected, the generated file tries to match the user's current Visual Studio formatting configuration from **Tools > Options**.
- Updated the shipped clang-format.exe version to 6.0.0.
- Templates to simplify adding configurations to CppProperties.json.
- Added templates to simplify adding configurations to CMakeSettings.json *(Figure 7)*.

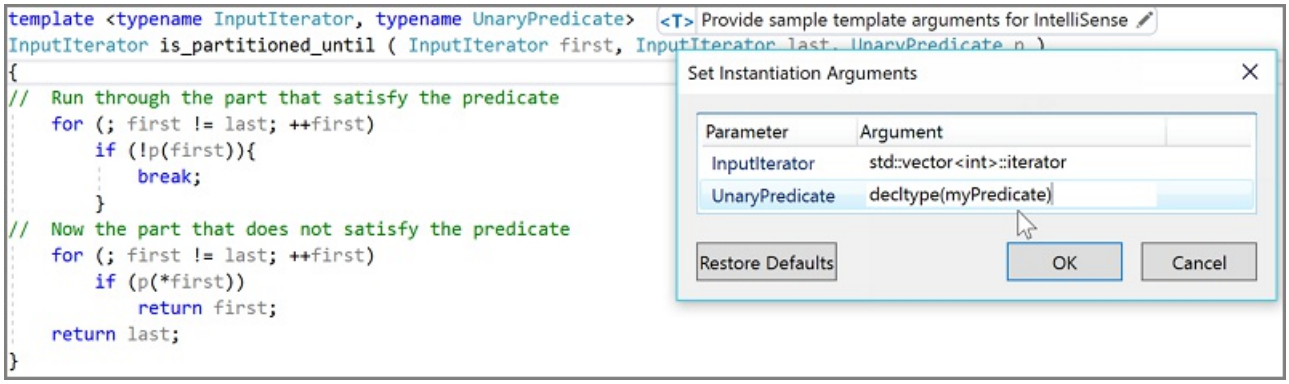


(Figure 7) Configuration Templates for CMake

C++ Productivity

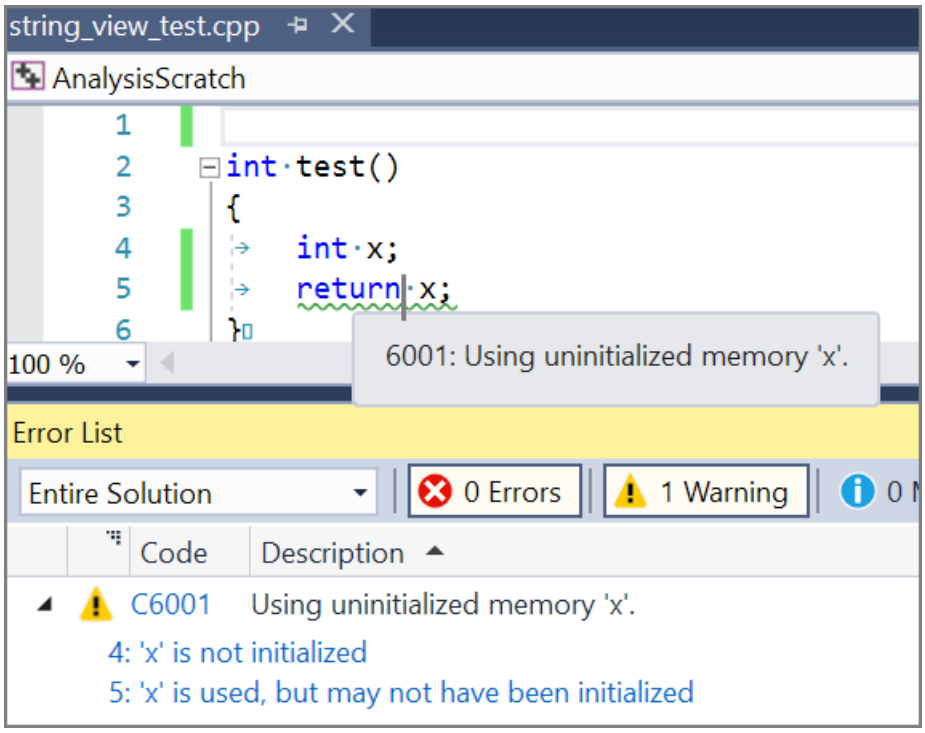
We made the following enhancements and improvements to C++ productivity:

- C++ Quick Info tooltips on macros now show what they expand to, rather than just their definition. This is particularly useful for complex macros that reference other macros as it clarifies what the macro identifier is replaced with by the preprocessor.
- Added a new quick-fix lightbulb to convert basic macros to constexpr as a new tool to modernize code.
- IntelliSense for templates provide you more details about template arguments to take full advantage of IntelliSense within your template body *(Figure 8)*.



(Figure 8) Template IntelliSense

- We are working to refresh our code analysis experience. You can now enable the new, in-progress features under **Tools** > **Options** > **Text Editor** > **C++** > **Experimental** > **Code Analysis**. Code analysis can run in the background when files are opened or saved, and results are displayed in the error list and as green squiggles in the editor *(Figure 9)*.

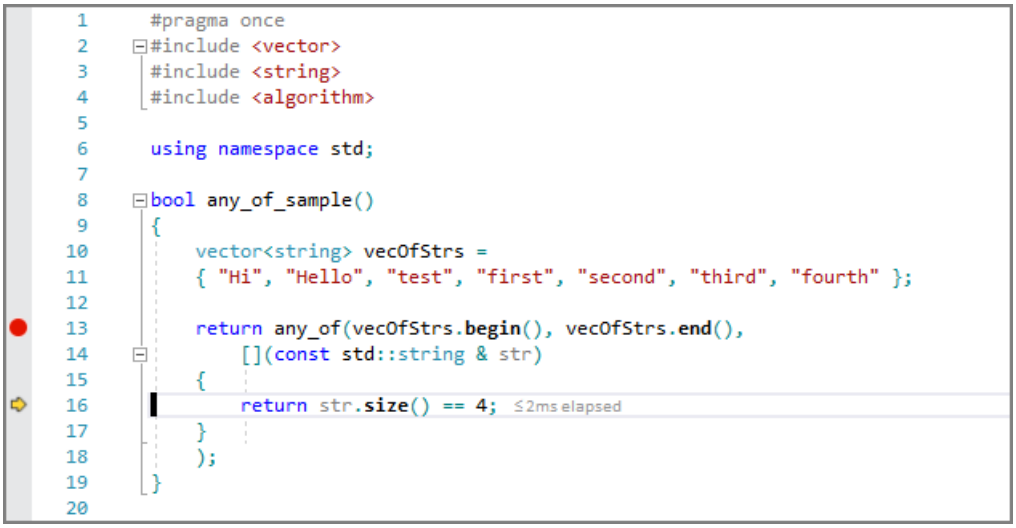


(Figure 9) In-editor Code Analysis

C++ Debugging Improvements

We made the following debugging improvements:

- Just My Code now enables you to step-over code from system or 3rd party C++ libraries in addition to collapsing those calls in the call-stack window. You can control this behavior for any C++ libraries when your code is compiled with /JMC and the non-user libraries paths are specified in a .natjmc file. If the system library calls into user-code, when you step in, the debugger skips all system code and stops on the first line of user-code callback *(Figure 10)*.



(Figure 10) Just My Code

- Data breakpoints can now be set inside the Watch, Quickwatch, Autos, and Locals windows, allowing you to break when a value stored in memory changes in just a few, short clicks.
- Source Link lets you embed information about an executable or library's original source code into the PDB during compilation.
- When debugging, the console window now stays open by default when the program terminates execution (similar to running the program without the debugger). This behavior can be toggled back to closing the console automatically in **Tools > Options > Debugging > General**.

JavaScript and TypeScript Improvements

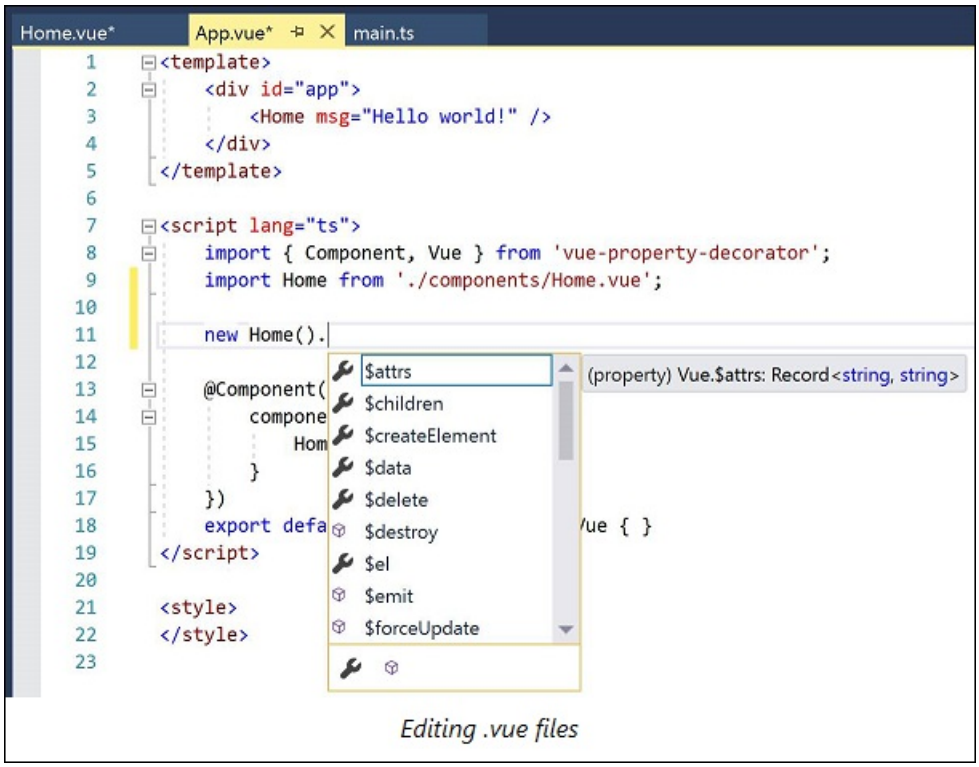
TypeScript 3.0

Visual Studio 2017 version 15.8 now includes TypeScript 3.0 by default. For more details on this release, see the TypeScript 3.0 release announcement.

Improved Vue.js support

Support for the Vue.js library has been improved, and in particular support for .vue files, also known as "single file components". This provides enhancements when editing script blocks inside .vue files, including support for script blocks authored in TypeScript through the `lang="ts"` attribute on the script element. (Note: A build process using WebPack or similar must be used to convert the .vue files to the HTML and JS files needed at runtime. See the Single File Components page for more details).

If the Node.js workload is installed, there will now be "Basic Vue.js Web Application" templates under the "JavaScript / Node.js" or "TypeScript / Node.js" paths in the New Project dialog. The below shows an example of editing TypeScript code inside a script block in a .vue file *(Figure 11)*.



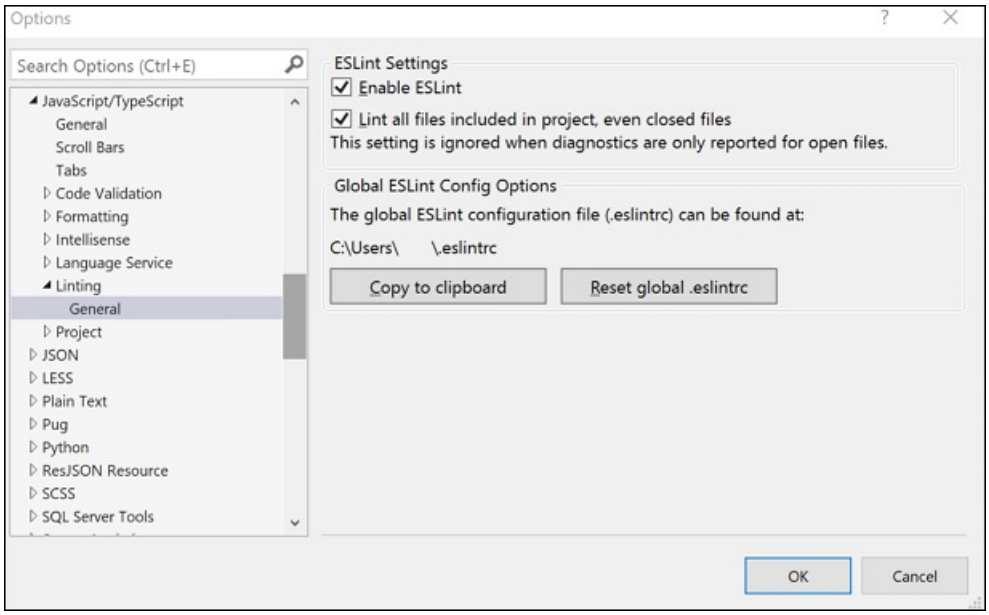
(Figure 11) Editing .vue Files

ESLint Improvements

We reimplemented ESLint support for this release. ESLint has the following improvements and enhancements:

- Rather than only linting saved files, Visual Studio will now lint JavaScript files as you edit, as well.
- Results can be reported for all JS files in your project, not just open files; if there are parts of your project you do not want to be linted, an .eslintignore file can now be used to specify directories and files that should be ignored.
- ESLint was updated to use ESLint 4 by default, but if your project has a local installation of ESLint, it will use that version instead.

ESLint may be disabled globally in Visual Studio by unchecking the *Enable ESLint* setting in the **Tools > Options > Text Editor > Javascript/Typescript > Linting ** *(Figure 12)*.



(Figure 12) ESLint Options

Open Folder Improvements for Node.js

There are numerous improvements working with JavaScript and TypeScript in the Open Folder scenario when the "Node.js workload" is installed. For example, managing NPM packages, building TypeScript, launching and debugging with Node.exe, executing NPM scripts, and running unit tests.

See Develop JavaScript and TypeScript code in Visual Studio without solutions or projects} for more information.

Editor Performance Improvements

In previous releases, all JavaScript and TypeScript language service operations were serviced by a single Node.js process. This could cause editor delays if commands that impact user typing (such as automatic formatting after a newline) were sent while a potentially lengthy operation was already in process (such as analyzing code for errors). To mitigate this, a separate process is now used for the operations that impact editing the most. This process is significantly lighter on system resources than the existing language service process. However, if you wish to disable the new process, check the *Disable dedicated syntax process* box in **Tools > Options > Text Editor > JavaScript/TypeScript > Language Service**.

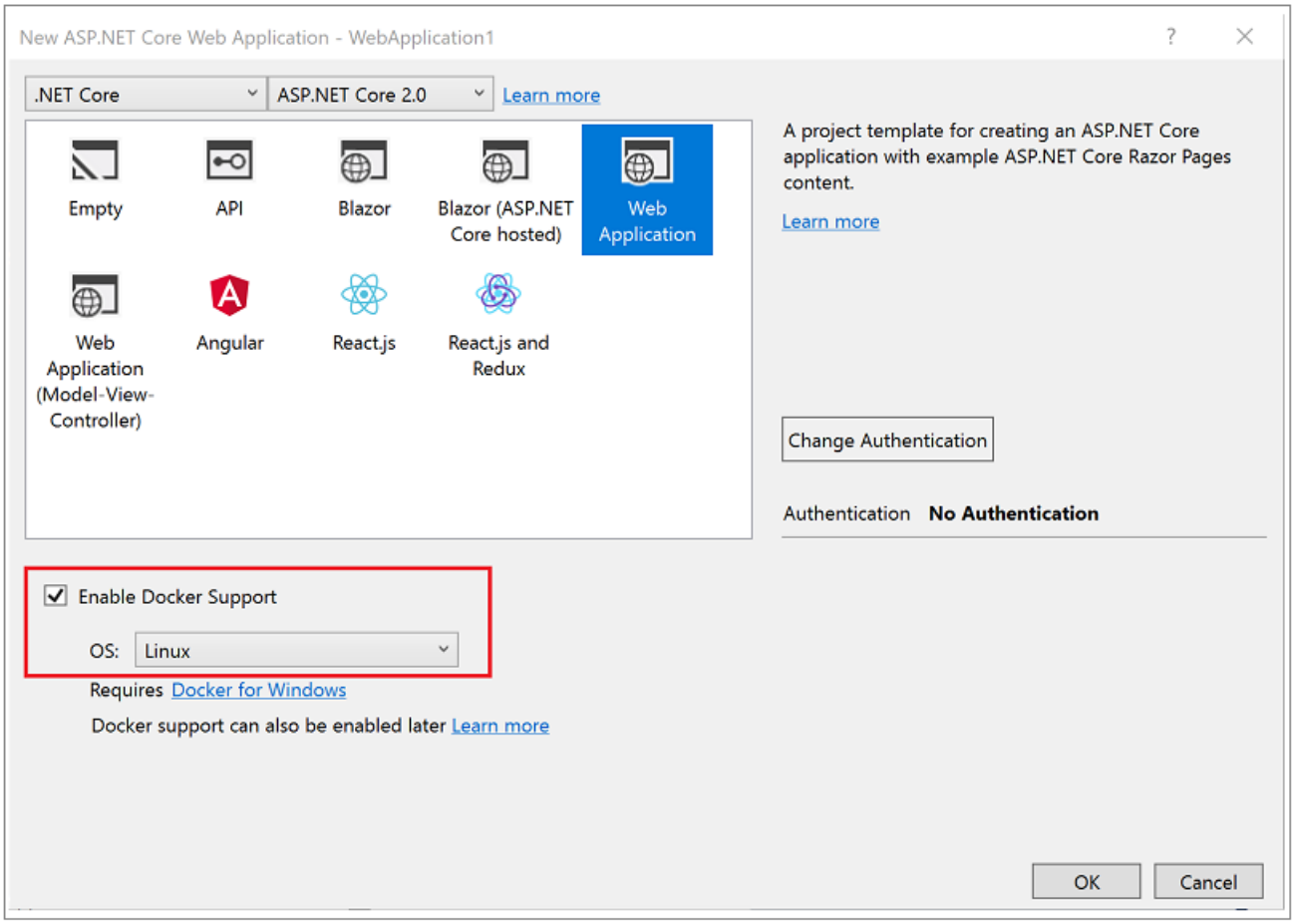
Visual Studio Web Tools

Library Manager is a new feature included in Visual Studio 2017. It helps you manage client-side libraries in your web projects.

Container Tools

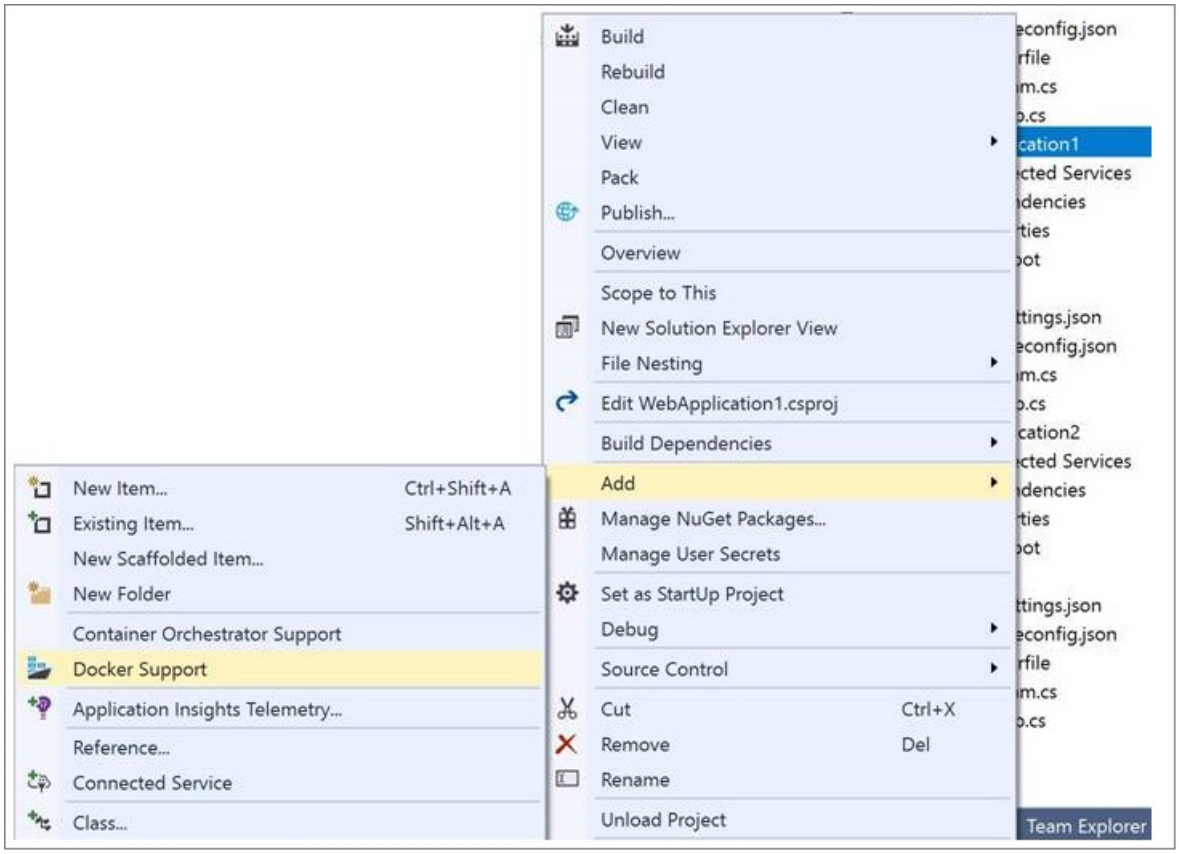
We added a new single project Docker container experience for ASP.NET Core web projects. This supplements the existing Docker Compose-based container tooling and provides a simpler, easier way to create, debug, and build Docker containers right from Visual Studio.

You can add Docker support when creating the project *(Figure 13):*



(Figure 13) Enable Docker Support

Or, you can enable Docker support to an existing project through the project's context menu in Solution Explorer *(Figure 14)*. Once you do this, Visual Studio creates a single Dockerfile in the project. You have the option to choose either Windows or Linux.



(Figure 14) Add Docker Support

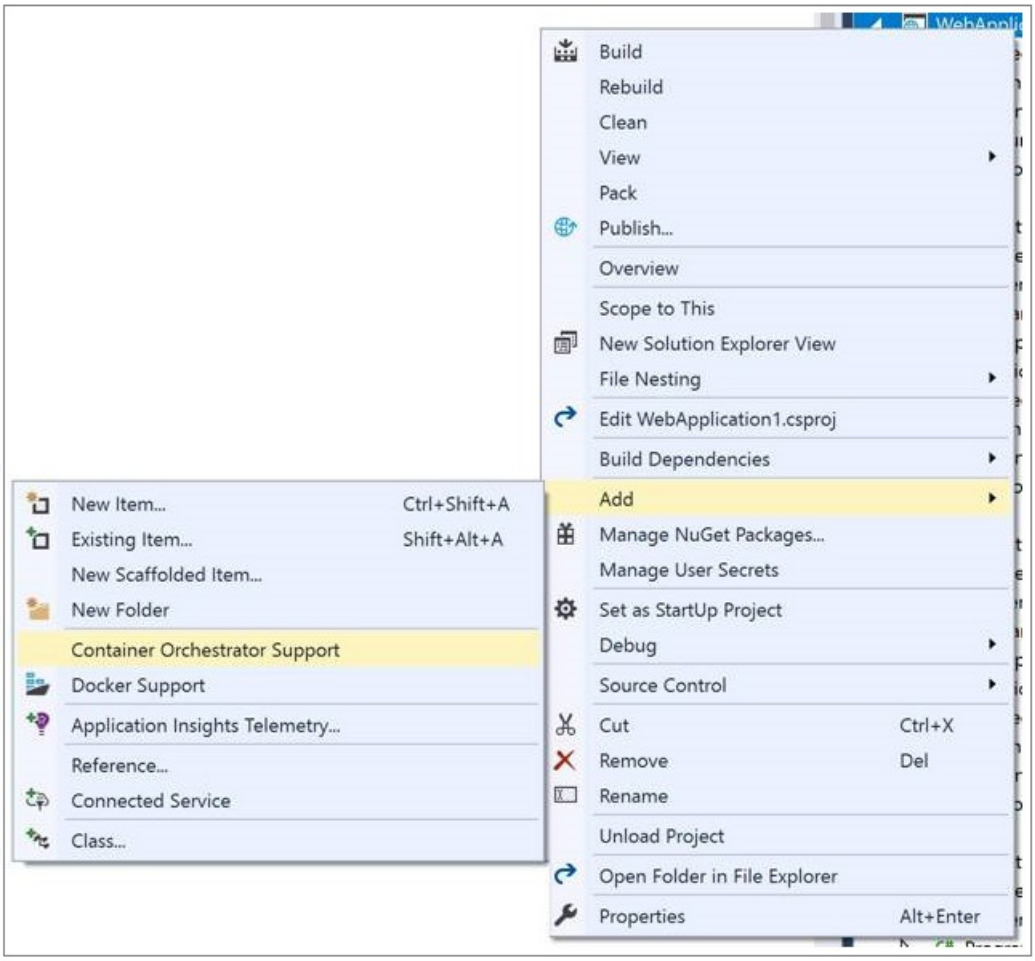
Visual Studio also adds a debugger launch profile *(Figure 15)* for Docker so that the project can be debugged while running within a container.

(Figure 15) Docker Launch Profile

If you have a solution with multiple Docker projects, by default, only one container runs when you choose to start the solution. If you wish to run multiple containers simultaneously, you can right-click on the solution in Solution Explorer and select **Set Startup Projects** and then **Multiple startup projects**, and then set the *Action* drop-down to **Start** or **Start without debugging** for all projects that you want to run.

Once you have your containerized project running the way you want, you can right-click on the project and select **Build Docker Image** to build an image locally, when ready to push to Azure Container Registry or DockerHub.

You can also add the existing Docker Compose-based functionality to an ASP.NET Core web project through the new *Container Orchestrator Support* option *(Figure 16)*. Right-click on the ASP.NET Core web project in Solution Explorer, select **Add > Container Orchestrator Support**, and then select **Docker Compose** from the drop-down menu.



(Figure 16) Add Orchestration Support

Publish Improvements

This release contains the following publish improvements:

- When publishing a Docker container to a container registry, you can now customize the tag for the image. You can either manually add a tag (default is "latest") or use an auto-generated tag to make sure each tag is unique.
- When creating a new Azure App Service, you can also configure Application Insights to collect telemetry

automatically. If you choose a region that also has Application Insights, it is enabled by default. If you choose a region that does not yet contain Application Insights, you can manually specify a different region for your Application Insights resource from the dropdown.

- When publishing Azure Functions projects you can choose to publish using the new Run-From-Zip feature.

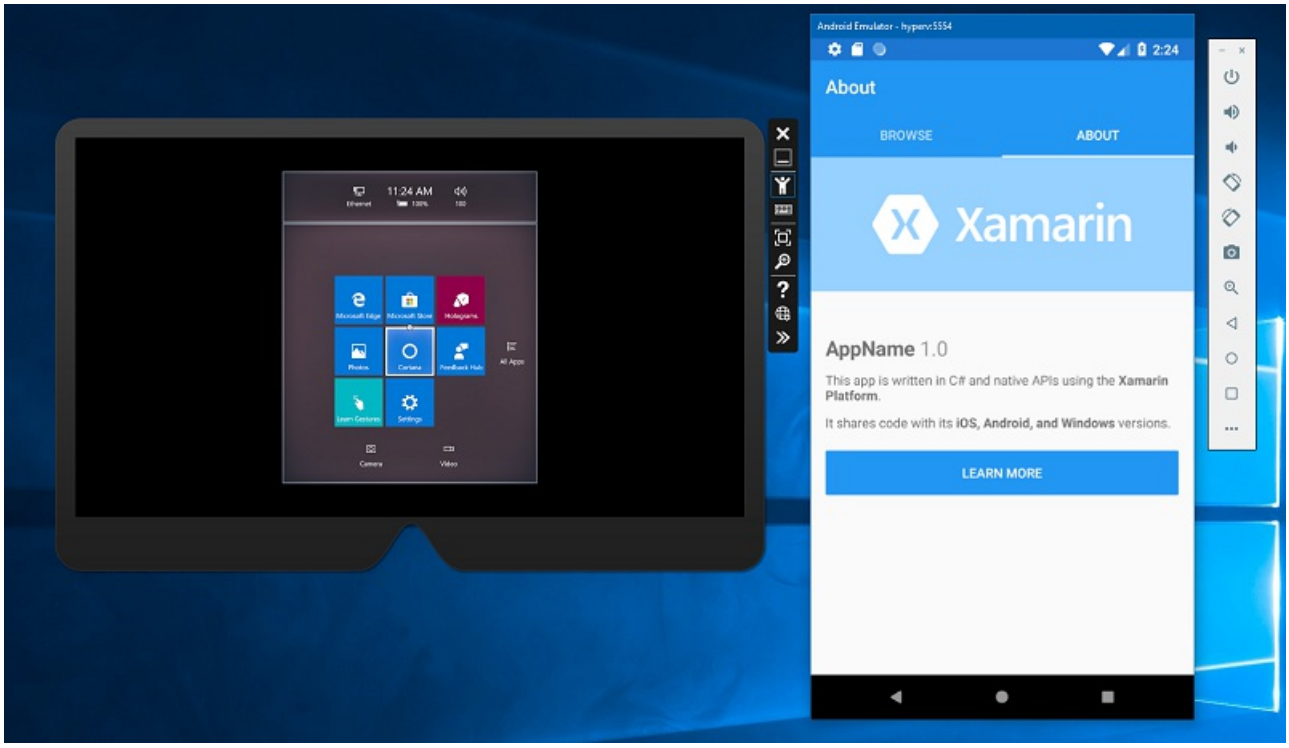
Visual Studio Tools for Xamarin

This release contains the following updates for Xamarin:

- We added support for Xcode 9.4.
- When you create a new Xamarin.Forms project, the default code sharing option is now .NET Standard. The shared project option is still available.
- We added Android incremental build improvements. Xamarin.Android uses files generated in the intermediate output directory to achieve incremental builds that are faster than full builds. Previously, if you changed your project's target framework it would invalidate the files and result in a full build on the next run. In this release we now preserve the files in per-framework folders so you can switch between different target frameworks and still benefit from incremental builds. Cleaning the project allows you to reclaim the disk space used by the preserved files.
- We have added minimum support for Xamarin.Mac binding projects in Visual Studio 2017. This enables Visual Studio to load and recognize Xamarin.Mac binding projects as supported. You can also build Xamarin.Mac binding projects. However, the build process is performed locally without using the native Mac tool chain, so the generated IL assemblies cannot be used for running or debugging in apps.

Hyper-V Android Emulator Support

This release adds support for the Google Android emulator that is compatible with Hyper-V when running on the Windows 10 April 2018 Update *(Figure 17)*. This enables you to use Google's Android emulator side-by-side with other Hyper-V based technologies, including Hyper-V virtual machines, Docker tooling, the HoloLens emulator, and more. Mobile app developers who use Hyper-V now have access to a fast Android emulator that always supports the latest Android APIs, works with Google Play Services out of the box, and supports all features of the Android emulator, including camera, geolocation, and Quick Boot.

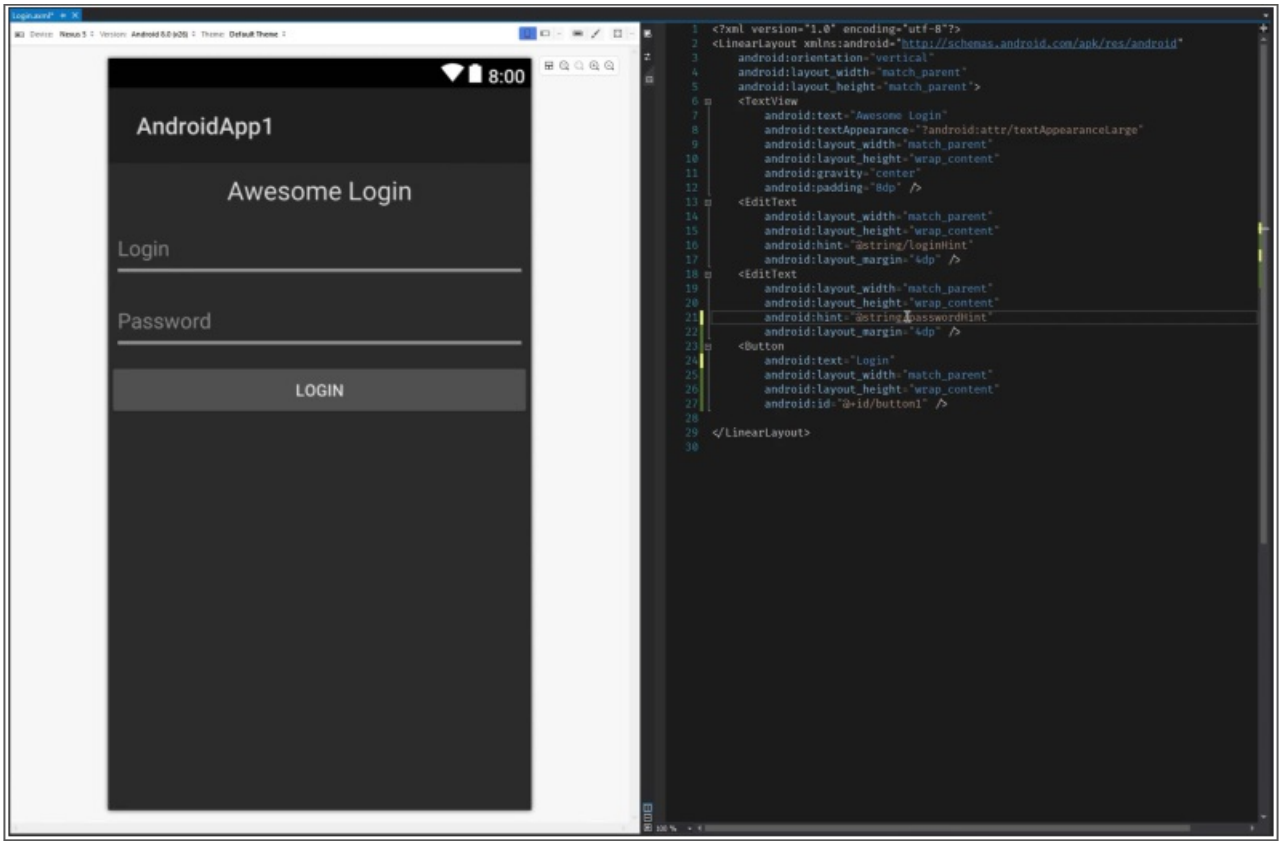


(Figure 17) Google Android Emulator and HoloLens Emulator

Xamarin.Android Designer

We made significant improvements to the designer experience for Xamarin.Android. Highlights include:

- A split-view editor was introduced which allows you to create, edit, and preview your layouts at the same time *(Figure 18)*.



(Figure 18) Xamarin.Android Split-view Editor

- Improved IntelliSense experience and reliability of custom controls.
- Sample data support for system provided values.

Xamarin.Forms Previewer

The Xamarin.Forms Previewer now has toolbox support when using Xamarin.Forms version 3.1.0.583944 or higher. Xamarin.Forms controls will appear in the toolbox so they are more discoverable for those new to the toolkit. You can also drag and drop a control onto the XAML code editor to add the control to the page. The Xamarin.Forms Previewer is now part of the XAML editor. You can open and close it with the expand icon on the edge of the editor pane.

Python

This release adds the following improvements for Python developers:

- Python IntelliSense now uses typeshed definitions to provide richer results for libraries where auto-completions cannot be inferred by static analysis.
- The experimental debugger, first announced in the 15.7 preview releases, is now the default debug engine used for Python, providing faster and more reliable debugging for Python code.
- We have added support for Python 3.7, including fixes to enable debug attach, profiling, and mixed-mode (cross-language) debugging features.
- For more information about the above features, be sure to check out our Python in Visual Studio 2017 version 15.8 blog post.

Migrate Local Azure Function Settings

The "Managed Application Settings" dialog available from the publish summary page now displays values from your local.settings.json file and enables you to migrate values to your remote Azure Function app hosted in Azure.

Connected Services

You can now configure continuous delivery for Azure functions directly from Visual Studio 2017 for solutions with Azure Function Projects.

Test Explorer Improvement

Test Explorer now displays a more informative test status summary pane (lower pane of test explorer) when one of the groupings in the hierarchy view is selected. The pane now displays how many tests failed, passed, or not run in that grouping.

New Extensibility Features

Language Server Protocol

Visual Studio now has native support for the Language Server Protocol. Extension authors can create extensions that communicate with existing language servers to add additional language support to Visual Studio. Extension users can install these extensions to start using their favorite language inside Visual Studio like Rust.

AsyncPackage Template

Extension authors can now use item templates to create AsyncPackages to optimize their extension's performance. Read more about AsyncPackages.

Extension Packs

Easily share your favorite set of extensions or set up a new install of Visual Studio with all your extensions by using an extension pack. Extension packs allow you to create a list of extensions, package them in an extension, and use it quickly to install those extensions in bulk.

Command Line Extension Publishing

Publish your extensions to the Visual Studio Marketplace using the command line.

.NET Core SDK 2.1.400

Visual Studio 2017 version 15.8 includes .NET Core SDK 2.1.400. New SDK features include:

- Added NUnit templates
- Added support for signed global tools
- Improved help text for better clarity

32 issues were closed on the .NET Core CLI.
20 issues were closed on the .NET Core SDK.

Source Control

For .NET Core projects, files added directly to the project through File Explorer will now show the correct Git and TFS tracking icons in the Solution Explorer without needing to reload the solution.

.NET Test Adapter Extension

The .NET Test Adapter has the following breaking change and deprecation:

- Breaking Change: All test projects must include their .NET test adapter NuGet reference in their csproj. If they do not, this test output will appear on the project if discovery by a test adapter extension is kicked off after a build or if the user tries to run the selected tests:
  - Test project {} does not reference any .NET NuGet adapter. Test discovery or execution might not work for this project. It is recommended to reference NuGet test adapters in each test project in the solution.
- .NET test frameworks have been releasing their adapters in NuGet packages and moving away from Visual Studio extensions. The support for .NET test adapters delivered through extensions is deprecated, but still supported. This means that two new options are available in **Tools > Options > Test**.
  - The first option allows Visual Studio to only use the test adapters it finds in the test assembly folder (populated by the test adapter NuGet reference) or as specified in the runsettings file.
  - The second option allows Visual Studio to "fallback" to the old behavior and search for test adapter extensions for projects that do not have a test adapter NuGet reference. Both options are checked by

default, so no default behavior will change in this release.

- Note: Non-.NET test adapters are not affected with this change.

ASP.NET .NET Framework Secrets Support

For ASP.NET, .NET Framework projects that target .NET Framework 4.7.1 or higher, you can now open and store secrets you do not want in your source code in usersecrets.xml by right-clicking on the project and selecting "Managed User Secrets".

.NET Framework 4.7.2

Visual Studio 2017 version 15.8 now offers the .NET Framework 4.7.2 development tools to all supported platforms with the 4.7.2 runtime included. The .NET Framework 4.7.2 offers several new features and improvements as well as numerous reliability, stability, security, and performance fixes.

You can find more details about the .NET Framework 4.7.2 in these articles:

- .NET Framework 4.7.2 RTM Blog
- .NET Framework 4.7.2 release notes
- Known issues in the .NET Framework 4.7.2
- Application Compatibility in the .NET Framework 4.7.2
- API Changes

Delay Package Load

Visual Studio now delays the loading of asynchronous packages that are configured to autoload until after the Visual Studio IDE has fully started and the solution has loaded. This change does not affect synchronously autoloaded packages. Users can look at the Task Status Center in the lower left corner of the status bar to monitor progress. Extension authors that create asyncpackages should test their extension. For more information, see Improving the responsiveness of critical scenarios by updating auto load behavior for extensions.

Visual Studio 2017 version 15.8 Security Advisory Notices

Visual Studio 2017 version 15.8.4 Service Release -- *released on September 11, 2018*

CVE-2018-8409 .NET Core Denial Of Service Vulnerability

A denial of service vulnerability exists in .NET Core 2.1 when System.IO.Pipelines improperly handles requests. An attacker who successfully exploited this vulnerability could cause a denial of service against an application that is leveraging System.IO.Pipelines. The vulnerability can be exploited remotely, without authentication. A remote unauthenticated attacker could exploit this vulnerability by providing specially crafted requests to the application.

CVE-2018-8409 ASP.NET Core Denial Of Service Vulnerability

A denial of service vulnerability exists in ASP.NET Core 2.1 that improperly handles web requests. An attacker who successfully exploited this vulnerability could cause a denial of service against an ASP.NET Core web application. The vulnerability can be exploited remotely, without authentication. A remote unauthenticated attacker could exploit this vulnerability by providing a specially crafted web requests to the ASP.NET Core application.

Visual Studio 2017 version 15.8 -- *released on August 14, 2018*

CVE-2018-0952 Diagnostic Hub Standard Collector Elevation of Privilege Vulnerability

An elevation of privilege vulnerability exists in a visual studio service, which can lead to system privileges by a non-admin user when writing files. An attacker who took advantage of this could write files as system while only having user level access. This security update addresses this issue by impersonating the current user to validate access to the file location.

Visual Studio 2017 version 15.8 Preview 4 -- *released on July 10, 2018*

CVE-2018-8172 Visual Studio Remote Code Execution Vulnerability

A remote code execution vulnerability that can lead to exploitation of a user's machine by opening a specially crafted project, or resource file. The security update addresses the vulnerability by correcting how Visual Studio

checks the source markup of a file.

CVE-2018-8260 .NET Framework Remote Code Execution Vulnerability

A remote code execution vulnerability exists in .NET software that can lead to exploitation of a user's machine by allowing attackers to run arbitrary code in the context of the current user. The security update addresses the vulnerability by correcting how .NET checks the source markup of a file.

CVE-2018-8232 .NET Microsoft Macro Assembler Tampering Vulnerability

Tampering vulnerability related to the Microsoft Macro Assembler improperly validating code. The security update addresses the vulnerability by ensuring that Microsoft Macro Assembler properly validates code logic.

CVE-2018-8171 ASP.NET Core Security Feature Bypass Vulnerability

An ASP.NET Core Security Feature Bypass Vulnerability exists when the number of incorrect login attempts is not validated that can lead to an attacker trying infinite authentication attempts. The update addresses the vulnerability by validating the number of incorrect login attempts.

Visual Studio 2017 version 15.8 Preview 3 -- *released on June 26, 2018*

CVE-2018-11235 Microsoft Security Advisory for Git Security Vulnerability

We fixed a security vulnerability in Git that was disclosed by the Git community. The vulnerability can lead to arbitrary code execution when a user clones a malicious repository.

Visual Studio 2017 version 15.8 Preview 1 -- *released on May 08, 2018*

CVE-2018-0765 Microsoft Security Advisory for .NET Core Denial of Service Vulnerability

- Microsoft is releasing this security advisory to provide information about a vulnerability in .NET Core and .NET native version 2.0. This advisory also provides guidance on what developers can do to update their applications to remove this vulnerability.
- Microsoft is aware of a denial of service vulnerability that exists when .NET Framework and .NET Core improperly process XML documents. An attacker who successfully exploited this vulnerability could cause a denial of service against a .NET Framework, .NET Core, or .NET native application.
- The update addresses the vulnerability by correcting how .NET Framework, .NET Core, and .NET native applications handle XML document processing.
- If your application is an ASP.NET Core application, developers are also advised to update to ASP.NET Core 2.0.8.

Visual Studio 2017 version 15.8.1

released on August 17, 2018

Top Issues Fixed in 15.8.1

These are the issues addressed in 15.8.1:

- Fixed an issue where Visual Studio would close unexpectedly when a browser window was closed while debugging a Web project.

Visual Studio 2017 version 15.8.2

released on August 28, 2018

Top Issues Fixed in 15.8.2

These are the customer-reported issues addressed in 15.8.2:

- Publish single file context menu item missing.
- Git network operations not working: Cannot spawn git-askpass.exe.
- "Visual C++ Resource Editor Package" load failed.

- Preview Transform gives error.
- Cannot type close brace.
- Object reference not set to an instance of an object when deploying Service Fabric project after update 15.8.0.
- "The given path's format is not supported" installation error.
- Task ExpandPriContent failed. Illegal characters in path.
- "Could not create instance of type System.Collections.Generic.Iset'1[Signer]" installer error.
- Service fabric deployment fails in 15.8.0.
- Cannot create app package for Windows store.
- Incorrect code generation for strongly-typed resources.
- Keyboard binding for Italian layout.
- IntelliSense broken #9999.
- When working on a Unity project and Unity rewrites multiple project files, the Visual Studio Tools for Unity will now automatically reload the solution.
- The customer will now get a dialog to close Office applications if they are blocking the Visual Studio install and causing a failure of 1303.

Visual Studio 2017 version 15.8.3

released on September 6, 2018

Top Issues Fixed in 15.8.3

These are the customer-reported issues addressed in 15.8.3:

- Visual Studio 2017 version 15.8.2 contained a pre-release build of .NET Core SDK 2.1.401 that is incompatible with Visual Studio. We have corrected this issue with Visual Studio 2017 version 15.8.3.
- VSCppUnit test case creation failure.
- Python IntelliSense not working on Python 2.7.
- XAML designer causing Visual Studio to close unexpectedly.

Visual Studio 2017 version 15.8.4

released on September 11, 2018

Top Issues Fixed in 15.8.4

These are the customer-reported issues addressed in 15.8.4:

- Constexpr function with explicit (void) parameter crashes MSVC 15.8.0 release.
- TypeScriptLspCodeActionProvider encountered an error and has been disabled.
- The 'Shared Web Components' package did not load correctly.
- Editing CSS with Browser Link enabled can crash Visual Studio.
- Service Fabric Deployment fails in 15.8.0.
- Incorrect error for constexpr pointer to members with VS2017 15.8.
- Constexpr doesn't support __declspec(dllimport) in VS2017 15.8.
- C++ VS 2017 Update 8 difference in member function pointer between C++14 and C++17 modes leads to crashes.
- TestExplorer locking up UI in latest build.
- [WebKit] "Already defined symbol" for static template member definition (regression).
- VS 15.8.0 Internal compiler error.
- Internal compiler error in constexpr function.

- Test Explorer Window remains blank with Express.
- Visual Studio 2017 fails to open on FIPS-enabled machine..
- VS2017 15.8 Internal Compiler Error 'msc1.cpp', line 1518 when evaluating simple constexpr function.
- Fatal error C1001 after update to VS 15.8.0.
- Migrating to VS2017 15.8: Internal compiler error when using constexpr functions as template parameters.
- Internal compiler error when compiling a template function with a constexpr in VC++ 15.8.0.
- Code generated by VC 15.8.1 compiler lead to crash.

Visual Studio 2017 version 15.8.5

released on September 20, 2018

Summary of What's New in 15.8.5

- Visual Studio Tools for Xamarin now supports Xcode 10.

Top Issues Fixed in 15.8.5

These are the customer-reported issues addressed in 15.8.5:

- Debugger tooltip does not expand in Xamarin.iOS debugging session.
- Debugger won't let me leave execution point (Android).
- After updating to 15.8.1, data tip does not show when debugging.
- Debugger data tip does not expand [Editor].
- Exception:MSB0001: Internal MSBuild Error: Already registered for build-time.
- Wrong path to resource in manifest for ClickOnce after upgrading to VS 15.8.1.
- ClickOnce Manifest error in 15.8.
- ClickOnce Publish is missing certain references.
- ClickOnce: Language file copied to wrong location.
- Reference missing in the ClickOnce publish folder.
- Compiling non-.NET SDK style F# projects results in an internal MSBuild error unless MSBuild parallelism is set to 1.
- F# is not NGEN'd in Visual Studio.
- F# deploys the FSharp.Core 4.5.0.0 binary in the 4.4.3.0 location.

Details of What's New in 15.8.5

Visual Studio Tools for Xamarin

Visual Studio Tools for Xamarin now supports Xcode 10 that allows you to build and debug apps for iOS 12, tvOS 12, and watchOS 5. See how to get ready for iOS 12 and our introduction to iOS 12 for more details on the new features available.

Visual Studio 2017 version 15.8.6 New

released on October 02, 2018

Summary of What's New in 15.8.6

- The latest Windows 10 SDK is available.

Top Issues Fixed in 15.8.6

These are the customer-reported issues addressed in 15.8.6:

- NuGet packages for Intellisense are broken.
- Visual Studio closes after opening sln.
- App.Config XML editor still freezing in 15.8.4.

Details of What's New in 15.8.6

Latest Windows 10 SDK for Universal Windows Platform developers

The latest Windows 10 SDK (build 17763) is now available as an optional component for the Universal Windows Platform development workload. You can add this SDK to the workload by selecting the **Windows 10 SDK (10.0.17763.0)** checkbox.

Known Issues

See all existing known issues and available workarounds in Visual Studio 2017 version 15.8.

Known Issues

Feedback

We would love to hear from you! For issues, let us know through the Report a Problem option in the upper right-hand corner of either the installer or the Visual Studio IDE itself. The  icon is located in the upper right-hand corner. You can track your issues in the Visual Studio Developer Community, where you can ask questions and find answers. You can also make a product suggestion through UserVoice or get free installation help through our Live Chat support.

Blogs

Take advantage of the insights and recommendations available in the Developer Tools Blogs site to keep you up-to-date on all new releases and include deep dive posts on a broad range of features.

Developer Tools Blogs

Visual Studio 2017 Release Notes History

For more information relating to past versions of Visual Studio 2017, see the Visual Studio 2017 Release Notes History page.

Top of Page

10/2/2018 • 8 minutes to read

Visual Studio 2017 Preview Release Notes

Developer Community | System Requirements | Compatibility | Distributable Code | License Terms | Blogs | Known Issues |

IMPORTANT

This release is not "go-live" and not intended for use on production computers or for creating production code.

Click a button to download the newest preview version of Visual Studio 2017. For instructions on installing and updating Visual Studio 2017, see the Update Visual Studio 2017 to the most recent release documentation.

Download Community Preview | Download Professional Preview | Download Enterprise Preview | Download Build Tools Preview

What's New in Visual Studio version 15.9

The Visual Studio Blog is the official source of product insight from the Visual Studio Engineering Team. You can find in-depth information about the Visual Studio 15.9 releases in the following posts:

- Visual Studio 15.9 Preview 2

Visual Studio 2017 version 15.9 Preview Releases

- October 2, 2018 -- Visual Studio 2017 version 15.9 Preview 3 New
- September 11, 2018 -- Visual Studio 2017 version 15.9 Preview 2
- August 20, 2018 -- Visual Studio 2017 version 15.9 Preview 1

Known Issues

See all existing known issues and available workarounds in Visual Studio 2017 version 15.9.

Known Issues

Visual Studio 2017 version 15.9 Preview 3 New

released October 2, 2018

Summary of What's New in 15.9 Preview 3

- Language service support for TypeScript 3.0 features for semantic file renaming and project references.

- Improved Node.js development by updating Vue.js templates and adding support for unit testing using the Jest framework.
- Added and improved features for Universal Windows Platform developers, including ARM64 support, the latest preview SDK, and XAML Designer improvements.
- Visual Studio Tools for Xamarin now supports Xcode 10.
- We made improvements to the Xamarin.Android build performance.

Issues Fixed in 15.9 Preview 3

These are the customer-reported issues addressed in this release. If you are looking for a specific customer-reported issue that has been fixed in 15.9 Preview 3, visit the Developer Community.

- VS2017 15.8 Build doesn't start if XAML files are not manually saved first.
- Scrolling up with the arrow key causes Visual Studio to page up.
- VS Diff Compare missing grayout (slashes) that indicate added/removed lines.
- Navigation bar in editor has trouble handling long method names.
- Solution File ExtensibilityGlobals Section randomly switch order when saved.
- compilation error with constexpr if and std::underlying_type under /permissive-.
- C++ 15.8.2 crash when editing comments above code.
- Unable to cancel an ongoing "Find All References" task.
- C++ constexpr lambda compiles, however, Intellisense shows errors.
- New all .NET core project exception.
- False positive for C26495.
- False positive for C26495.
- C++ Visual Studio 2017 version 15.4 Editor Intellisense gives E0028 with constexpr lambda.
- LNK4099 warnings when number of parallel project builds is greater than 1.
- C26440 can be declared 'noexcept' generated for virtual functions marked as noexcept(false).
- Constexpr if incorrectly instantiates template classes on false branches that should not be evaluated.
- SHIFT + END visually highlighed selection stops on long lines in VS15.8.
- VS 15.8 broken stl/clr _ITERATOR_DEBUG_LEVEL=2: error C2039: '_Debug_message': is not a member of 'std'.
- .eslintignore file does not seem to be working.
- SFINAE fails to compile.

Details of What's New in 15.9 Preview 3

Language Service Support

- We added refactoring to fix up references to a file after it has been renamed. We also added support for project references, letting you split your TypeScript project up into separate builds that reference each other.

Node.js Development

- We updated to the latest Vue CLI 3.0 and improved linting in Vue.js template files. You can also write and run unit tests using the Jest framework.

Tools for Universal Windows Platform Developers

- The latest Windows 10 Insider Preview SDK (build 17763) is included as an optional component in the Universal Windows Platform development Workload.
- You can now build ARM64 UWP applications. For .NET UWP applications, only .NET Native is supported for ARM64, and you must set the Minimum Version of your application to the Fall Creators Update (Build 16299) or higher.
- The UWP XAML designer reliability has been improved in this release. Developers now have the option to enable Platform Only Mode for all design time experiences. This prevents the designer from executing any custom control code to improve reliability of the design surface.

Visual Studio Tools for Xamarin

- Visual Studio Tools for Xamarin now supports Xcode 10, which allows you to build and debug apps for iOS 12, tvOS 12, and watchOS 5. See how to get ready for iOS 12 and our introduction to iOS 12 for more details on the new features available.

Initial Xamarin.Android Build Performance Improvements

- Xamarin.Android 9.1 includes initial build performance improvements. See our Xamarin.Android 15.8 vs. 15.9 build performance comparison for more details.

Visual Studio 2017 version 15.9 Preview 2

released September 11, 2018

Summary of What's New in 15.9 Preview 2

New Features in 15.9 Preview 2

We added the following new features for this release:

- You can now import and export an installation configuration file that specifies which workloads and components should be installed with an instance of Visual Studio.
- Step back in debugger is now available in C++.
- We fixed several bugs in the F# compiler when working with `byref` types.

Issues Fixed in 15.9 Preview 2

These are the customer-reported issues addressed in this release. If you are looking for a specific customer-reported issue that has been fixed in 15.9 Preview 2, visit the Developer Community.

- No way to change "Find All References" background color.
- System.InvalidProgramException: Common Language Runtime detected an invalid program.
- /experimental:external generates a lot of C4193 warnings.
- Incorrect C3520 within noexcept expression in alias template.
- Compilation error with alias template and enable_if.
- Failed to add connection point in ATL project.
- VS2017 Pro Editor complains polymorphism smart pointer casting, while compiler doesn't.
- Memory leak on solution reload.
- ICE with constexpr and failed initializer_list conversion.
- (C++ compiler regression) Cannot compile my code with VS2017, whereas VS2015 compiles it well.
- Type-deduction appears to fail by an array-syntaxed parameter of pointer to abstract class type.
- Error C2672 "no matching overloaded function found" on correct code.
- Variadic template function overloading resolution failure.
- external:I include paths not working.
- Experimental C++ module compiler fails with a "constexpr" usage.
- ActiveX control - unable to add custom event.
- Encapsulation problem in 'using' with templates.
- C++ module with using declaration error C2061.
- Various issues with modules, ixx, and cxx files.
- LNK1179 duplicate COMDAT in Visual Studio 15.8 Preview 4.
- Problem with explicit destructor call for std::ostringstream.
- ICE When exporting module with a class that has a member of type basic_string.
- More warnings reported when trying to suppress warnings with /external:I.

- Memory leak on solution reload.
- VC++ 15.8: typename in templatized function default value expression.
- 15.8 VC++ Resource Editor will not load.
- ATL project add property or other stuff not worked as documented.

Details of What's New in 15.9 Preview 2

Install

- We made it easier to keep your installation settings consistent across multiple installations of Visual Studio. You can now use the Visual Studio Installer to export a .vsconfig file for a given instance of Visual Studio. This file will contain information about what workloads and components you have installed. You can then import this file to add these workload and component selections to another installation of Visual Studio.

C++

- We added the step back feature in the debugger for C++. Step back enables you to go back in time to view the state of your application at a previous point in time.

F# compiler

- We fixed a bug where extension methods that take `byref` values could mutate an immutable value.
- We improved the compile error information for overloads on `byref` / `inref` / `outref`, rather than displaying a more obscure error previously.
- Optional Type Extensions on `byref`s are now disallowed entirely. They used to be able to be declared, but could not be used in any way, resulting in a confusing experience.

Tools for Universal Windows Platform Developers

- The latest Windows 10 Insider Preview SDK (build 17754) is included as an optional component in the Universal Windows Platform development Workload.
- We added support for creating .MSIX packages for both the Universal Windows Platform projects, as well as in the Windows Application Packaging Project template. To create an .MSIX package, the minimum version of your application must be the latest Windows 10 Insider SDK Preview (build 17754).
- We made improvements to the F5 (Build + Deploy) speed for Universal Windows Platform applications. This will be most noticeable for deployments to remote targets using Windows authentication, but will impact all other deployments as well.
- The UWP XAML designer reliability is improved in this release. The XAML designer now replaces controls that throw with catchable exceptions with fallback controls, rather than having the designer crash. Fallback controls now have a yellow border to cue in developers that the control has been replaced at design time.

Visual Studio 2017 version 15.9 Preview 1

released August 20, 2018

Summary of What's New in 15.9 Preview 1

- We fixed several bugs in the F# compiler when working with `byref` types.
- We made substantial improvements to the experience of using authenticated package feeds.

Issues Fixed in 15.9 Preview 1

These are the customer-reported issues addressed in this release. If you are looking for a specific customer-reported issue that has been fixed in 15.9 Preview 1, visit the Developer Community.

- Source code has been misoptimized when targeting arm64 since 15.5.
- Shader compiler dxc used for pixel shader < 6.
- `noexcept` does not appear in the debug watch window.

Details of What's New in 15.9 Preview 1

NuGet Credential Provider improvements

This release substantially improves the experience of using authenticated package feeds, especially for Mac and Linux users:

- Visual Studio, MSBuild, NuGet.exe, and dotnet now support a new Credential Provider plugin interface. Previously, only NuGet.exe and Visual Studio accepted Credential Providers.
- Visual Studio editions (including the Build Tools edition) now deliver a VSTS Credential Provider with certain workloads, so that you can easily use VSTS feeds in the course of your development. To use these improvements, install the *NuGet package manager* or *NuGet targets and build tasks* components, or the *.NET Core* workload.

Feedback

We would love to hear from you! For issues, let us know through the Report a Problem option in the upper right-hand corner of either the installer or the Visual Studio IDE itself. The  icon is located in the upper right-hand corner. You can track your issues in the Visual Studio Developer Community, where you can ask questions and find answers. You can also make a product suggestion through UserVoice or get free installation help through our Live Chat support.

Blogs

Take advantage of the insights and recommendations available in the Developer Tools Blogs site to keep you up-to-date on all new releases and include deep dive posts on a broad range of features.

Developer Tools Blogs

Visual Studio 2017 Release Notes History

For information relating to past versions of Visual Studio 2017, see the Visual Studio 2017 Release Notes History page.

Top of Page

Visual Studio 2017 Release Notes History

9/17/2018 • 2 minutes to read

NOTE

If you are accessing this page from a non-English language version, and want to see the most up-to-date content, please visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

Click a button to download the latest version of Visual Studio 2017. For instructions on installing and updating Visual Studio 2017, see Update Visual Studio 2017 to the most recent release. Also, see instructions on how to install offline.

Download Community 2017 | Download Professional 2017 | Download Enterprise 2017

Visit the visualstudio.com/downloads page to download other Visual Studio 2017 products.

Preview Release Notes

- Visual Studio 2017 version 15.9 Release Notes

Current Release Notes

- Visual Studio 2017 version 15.8 Release Notes

Archived Release Notes

- Visual Studio 2017 version 15.7 Release Notes
- Visual Studio 2017 version 15.6 Release Notes
- Visual Studio 2017 version 15.5 Release Notes
- Visual Studio 2017 version 15.4 Release Notes
- Visual Studio 2017 version 15.3 Release Notes
- Visual Studio 2017 version 15.2 Release Notes
- Visual Studio 2017 version 15.1 Release Notes
- Visual Studio 2017 version 15.0 Release Notes

Top of Page

10/1/2018 • 21 minutes to read

Visual Studio 2017 for Mac version 7.6 Release Notes

Developer Community | System Requirements | Compatibility | Distributable Code | Xamarin | Blogs | Servicing |

NOTE

If you are accessing this page from a non-English language version, and want to see the most up-to-date content, visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

Click the button to download the latest version of Visual Studio 2017 for Mac. For instructions on setup and install, see the Setup and Install Visual Studio for Mac documentation.

Download Visual Studio 2017 for Mac

To learn more about Visual Studio 2017 for Mac, see Mac System Requirements and Mac Platform Targeting and Compatibility.

To learn more about other related downloads, see the Downloads page.

What's New in 7.6

Visual Studio 2017 for Mac version 7.6 Releases

- October 1, 2018 – Visual Studio 2017 for Mac version 7.6.8 New
- September 25, 2018 – Visual Studio 2017 for Mac version 7.6.7
- September 19, 2018 – Visual Studio 2017 for Mac version 7.6.6
- September 13, 2018 – Visual Studio 2017 for Mac version 7.6.5
- September 10, 2018 – Visual Studio 2017 for Mac version 7.6.4
- August 30, 2018 – Visual Studio 2017 for Mac version 7.6.3
- August 29, 2018 – Visual Studio 2017 for Mac version 7.6.2
- August 23, 2018 – Visual Studio 2017 for Mac version 7.6.1
- August 20, 2018 – Visual Studio 2017 for Mac version 7.6

Release Highlights

- Azure Functions support has been expanded. You can now choose the trigger type when creating a new Azure Functions project.
- Visual Studio for Mac now also supports publishing Azure Functions to Azure.
- We continue to improve performance and stability in Visual Studio for Mac.

Top Issues Fixed in this Release

- We fixed an issue where the editor becomes so slow that it's unusable
- We fixed an issue where Visual Studio is incredibly slow and laggy

- We fixed an issue where search results are not displayed
- We fixed a number of issues with the updater:
  - Cannot update VS if one of the updates does not download
  - Extensions were failing to download
  - Visual Studio for Mac appears to offer no updates.
- Fixed and issue where InteliSense does not work for auto-generated fields via Xamarin.Forms XAML
- We fixed numerous issues with highlighting:
  - TODO doesn't highlight
  - Javascript syntax highlighting does not work in VS 2017 for Mac
- We extended TestSessionTimeout to the maximum possible value, meaning that you're no longer restricted by longer running testing in VS Test.

Known Issues

Refer to the Known Issues section.

Details of What's New in 7.6

Visual Studio 2017 for Mac version 7.6

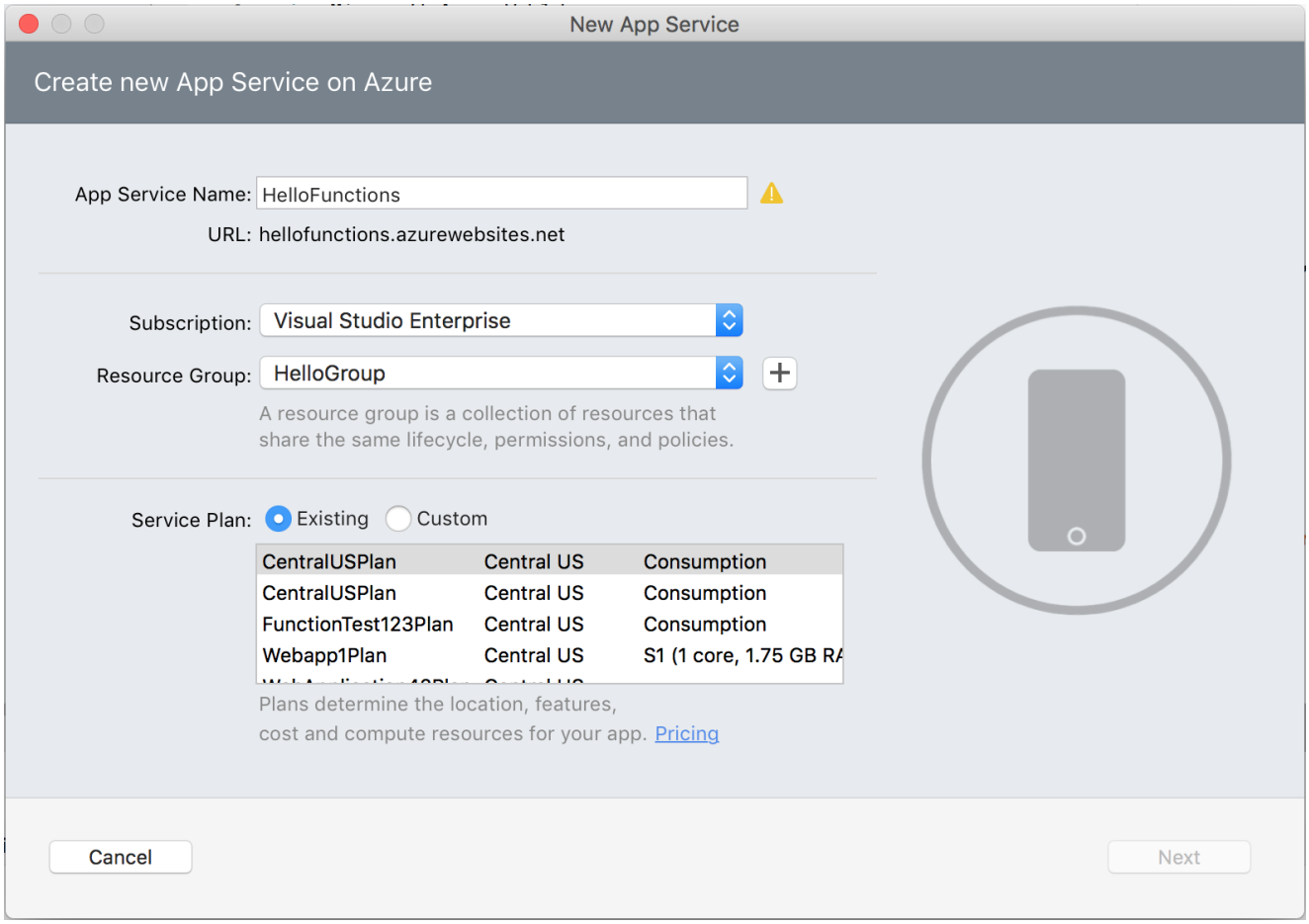
released August 20, 2018

New Features in 7.6

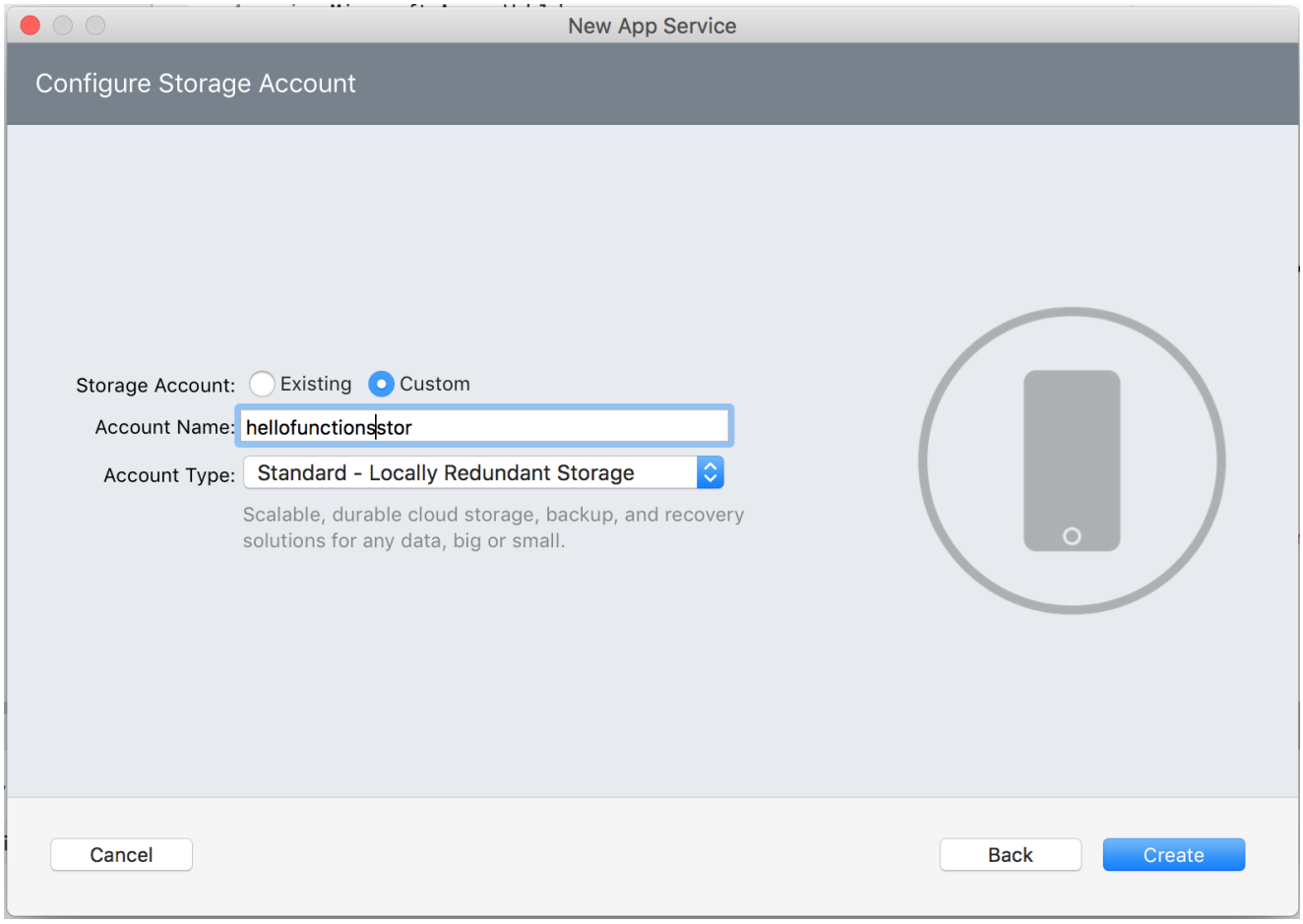
Azure Functions

Publishing Azure Functions

It's now possible to publish your Azure Function through Visual Studio for Mac. To publish, right-click the Function project and select **Publish**, then create or publish to an existing Azure App Service, as illustrated in the following screenshots.



(Figure 5) Azure Functions publishing; Create a new App Service

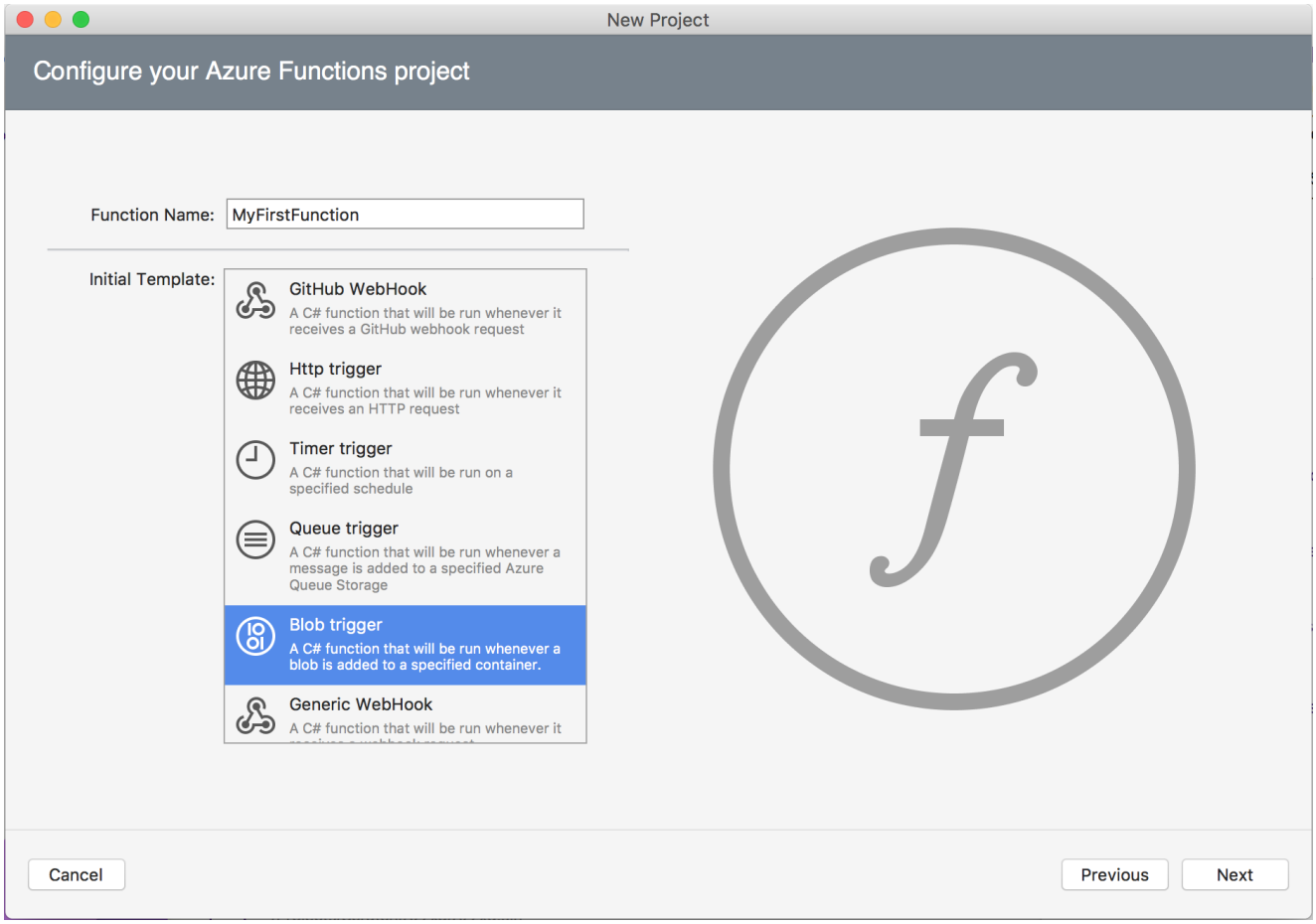


(Figure 6) Azure Functions publishing; Configure Storage Account

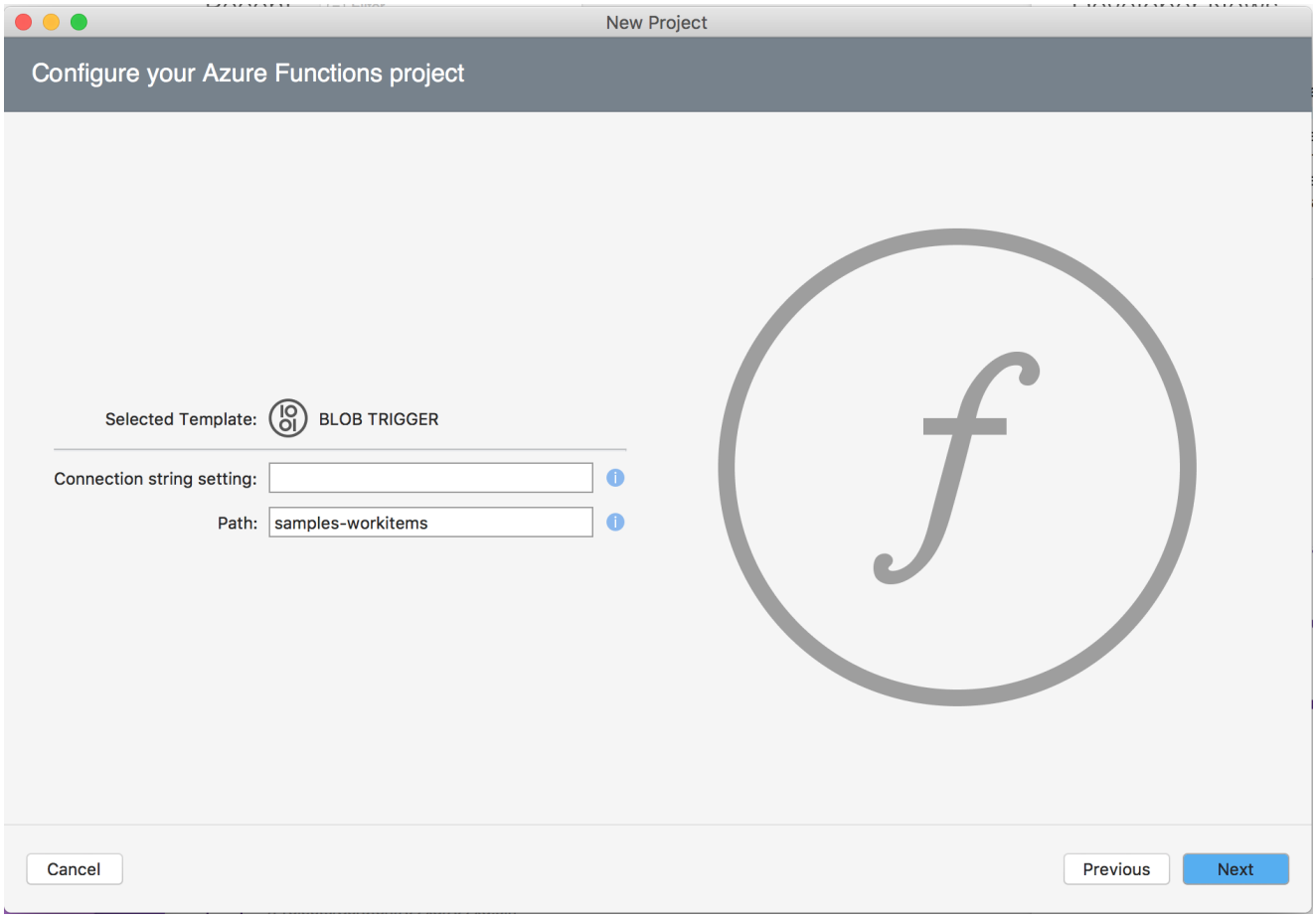
For more information on publishing your Azure function to Azure, see the Azure Functions documentation.

New Azure Functions templates

We updated and improved the experience when creating a new Functions project. Previously the new project would default with an HttpTrigger function. Visual Studio for Mac now offers available trigger types that you can select when creating a new project.



(Figure 3) New Azure Function; select trigger type



(Figure 4) New Azure Function; provide trigger details

For more information on using Azure functions in Visual Studio for Mac, see the Azure Functions documentation.

Issues fixed in 7.6

Performance

We made the following performance fixes:

- Fixed an issue where the WebReference addin slows down folder building in the solution pad.
- We have made additional gains on improving the amount of work performed by the IDE on startup.
- Fixed issues related to FileWatcherService resource starvation.
- Fixed an issue where code fix preview menu hangs several seconds on large fixes.
- Fixed an issue where LoadProject is called twice on solution load due to configuration change.
- Fixed an issue where F# highlight usages does too much work on the foreground thread.
- Visual Studio for Mac now disables full solution analysis when the OS triggers a warning about memory.
- Load syntax schemes on demand for each file type.
- Fixed an issue that could cause the UI to freeze after opening a solution.
- VSTest adapter delays solution load.
- Improved the performance of the Find References command.

Code Editor

We made the following fixes to Code Editor:

- Fixed an issue where typing block comments throws a NotImplementedException.
- Fixed an issue with Korean language input where the final character does not highlight when selecting from end of line to beginning of line, until code the completion dialog box disappears.
- Fixed an issue where toggle line comment menu no longer exists.
- Fixed an issue where the "Show stack trace" command in the exception bar does not show or display any

information.

- Fixed an issue where a NullReferenceException in MonoDevelop.SourceEditor.FileRegistry.HasMultipleIncorrectEolMarkers is thrown when opening a file.
- Fixed an issue where Line/col for TODO issues is off by 1.
- Fixed a rendering issue where the "Show scopes" command includes markup.
- Fixed an issue where debugger tooltips are not always displayed.
- Fixed some graphical glitches in the text editor.
- Fixed an issue where renaming a method should not trigger the removal of generic type arguments.
- Fixed an issue where EditorConfig settings are not applied to new files added to the project.
- Fixed an issue where an exception is thrown when clicking in specific locations of a source file.
- Fixed an issue where "Add Reference" quick fix does not add a reference.
- Fixed an issue where search results do not display in full screen with a second monitor.
- Fixed an issue where an error, "There was an error saving the changes", displays when enabling code folding.
- Fixed an issue that caused the editor to occasionally become unresponsive to user input.
- We have added an overtype marker in editor to show that the closing bracket can be typed over.

```
 1 using System;
 2 using System.Collections.Generic;
 3 using System.Linq;
 4 using System.Text;
 5 using System.Threading.Tasks;
 6 using Xamarin.Forms;
 7
 8 namespace NewFormsApp
 9 {
10     public partial class MainPage : ContentPage
11     {
12         public MainPage()
13         {
14             InitializeComponent()
15         }
16     }
17 }
18
```

(Figure 1) Editor overtype marker

- Fixed some issues with documentation comments: Exception, Param ref.
- Fixed incorrect XSLT validation error for xsl:variable.
- Fixed an issue with the cursor location when expanding a code snippet.
- Fixed an empty error dialog that is shown when the user has a corrupt highlighting theme.
- Fixed an issue where the gutter on the left corrupts when you hover over the '...' associated with a compile error.
- Fixed an issue where the editor doesn't work with .NET Standard 2.0 NuGet.
- Fixed an issue with Korean language input where the final character is not highlighted when selecting from end of line to beginning of line.
- Fixed an issue where Korean characters are transposed during selection/highlight (e.g. to copy + paste).
- Fixed an issue where analyzers and formatting do not update after modifying editorconfig.
- Fixed an issue where Single-line members were shown as collapsible.
- Fixed an issue where Creating a new event handler would always put the method at the start of the class.
- The closing curly brace is removed on deleting curly braces above it in a **.cs** file.
- Fixed an issue where Collapsed regions would auto expand.

- Fixed an issue where **Go to Definition** would not work reliably when navigating to the assembly browser.
- Fixed the assembly browser, which was broken in 7.5 build 1254.
- Fixed an issue causing Visual Studio for Mac to freeze when saving a file.
- Fixed an issue causing Right-click -> **FindReferences** returns too many useless results when used in an interface.
- Fixed an issue where Setting Place System directives first cannot be turned off.
- Fixed an issue where `TODO` doesn't highlight anymore.
- Fixed an issue where braces are not indented when the Mac language is Hebrew - (RTL language).
- Code Folding is now enabled by default.
- Fixed an issue where tooltips do not disappear.
- Fixed an issue where a suggested "Convert anonymous group to method group" replacement would be invalid.
- Tooltip timeout is now shorter, which lowers the quick info popup time.
- Fixed an issue where Alt+Up at beginning of the document results in an exception.
- C# classification does not update when compilation changes.

Accessibility

We made the following Accessibility fixes:

- Code folding shows when VoiceOver is being used, instead of auto hiding, which is the default behavior.
- Fixed an issue where VoiceOver would read invisible elements in the 'Project Options'.

Project

We made the following fixes to the project system:

- Fixed an issue that causes an intermittent crash when building Xamarin Forms Control Gallery.
- Fixed an issue where configuration mapping changes for a project does not notify modifications.
- Fixed a NullReferenceException when loading an sdk-style project with no TargetFramework.
- Fixed an issue where circular dependencies between MSBuild imports would result in a StackOverflowException.
- Improved error messaging when failing to evaluate an MSBuild project.
- Fixed an issue that caused Visual Studio for Mac to freeze when opening solution.
- Fixed a NullReferenceException when a file is added externally to the project.
- We fixed an issue where creating a new Functions project would result in IntelliSense errors until the project was closed and reopened.
- The `Clean` target now runs asynchronously.

.NET Core

We made the following fixes to our .NET Core support:

- Fixed an issue where a .NET Core project is not restored after changing project properties.
- Fixed the default build action for new files.
- Fixed a UI hang when building an ASP.NET Core Angular project.
- When the ASP.NET Core addin tries to install the HTTPS development certificate, the macOS message dialog that asks for the username and password will now display:

  > dotnet-dev-certs wants to make changes.

- We fixed an issue where it wasn't possible to add declarations to an empty C# file using suggestions from IntelliSense.

NuGet

We made the following fixes to our NuGet support:

- We fixed an issue where NuGet source transformations are not recognized by IntelliSense.
- You will no longer encounter the issue where updating PackageReference removed metadata.
- We now support no-op restores on opening a solution. If the package references have not changed, then a no-op restore will occur.
- The NuGet SDK Resolver that is included with Mono's MSBuild is now supported by Visual Studio for Mac. This allows SDKs to download from NuGet and consumed by SDK style projects.

F#

We fixed the following issues for F# developers:

- Fixed an issue where adding FSharp.Core 4.5.0 NuGet package shows 4.3.3 as the displayed dependency in the solution pad.
- Fixed some inaccurate F# autocompletion.
- We removed 'System.ValueTuple.4.3.0' from templates as it is not required and is not on NuGet.
- We updated the version of Xamarin.Forms that is referenced in new projects to 3.0.
- Fixed an issue where F#'s "Find References" wrongly returns "0 matches" too often.
- Fixed an issue where F# error in Android resource type provider when using Xamarin.Forms 3.0.
- Fixed an issue where F# interactive outputs initial output message without any user input.
- Fixed an issue where F# Xamarin.Forms project template description mentions PCL when it should be .NET Standard.
- Fixed an issue where **File > New Blank Forms App** in F# sets the Android SDK to 8.0 instead of Use Latest.
- Fixed an issue where F# interactive command history is off by one.
- Fixed an issue where implicit folders are incorrectly ordered in F# projects.
- Improved how F# gets its list of keywords by using FCS instead of a hardcoded list.

Web Tools

We made the following updates to our web tooling:

- Fixed an issue where code completion doesn't commit if the caret is immediately followed by `}`.
- Fixed an issue where JavaScript syntax highlighting breaks when passing parameter to a function.
- Fixed an issue where Razor directives are not colorized correctly.
- Updated JavaScript/TypeScript TextMate Language files.
- Fixed an issue where the New Project dialog does not close after creating a new Azure Functions project.
- Fixed an issue where code in indented `@{}` block jumps up a line when committed and cannot be repositioned.
- Fixed an exception when typing in TypeScript file: System.ArgumentException Message=The specified SnapshotPoint or SnapshotSpan is on a different ITextSnapshot than this SnapshotSpan..
- Fixed an issue where there was no IntelliSense for .css files.
- Fixed an issue where backspace was not available on smart indented lines.
- We fixed a fatal error when editing an **appsettings.json** file in a new ASP.NET Core Web app from template.
- We fixed an issue where web references to ASP.NET MVC project could not be added.

Debugger

- We fixed an issue where the Immediate window completion only contains locals.

Shell

This release introduces the following feature:

- It is now possible to create an Interactive Workbook from the Welcome Page if Workbooks is installed.

The following issues were addressed in the Visual Studio for Mac Shell:

- Fixed an issue that when the system language is changed to Chinese, it is not reflected in Visual Studio for Mac.
- Fixed an issue where the pin icon in the Welcome Page was too small.
- Fixed an issue where Visual Studio for Mac crashes when selecting a device with a name starting with "#".
- Fixed an issue where Drag-and-drop into the welcome page does not work anymore.
- Fixed an issue where you Cannot rename solution items.
- Fixed an issue where you Cannot add new file to solution items.
- Fixed an issue where the move to previous notebook menu item is not working.
- Fixed an issue where multiple Save dialogs are shown when you close using keyboard shortcut multiple times.

Version Control

- We fixed an issue with the Git stash comment not displayed in Stash Manager dialog.
- We fixed an issue with Git apply stash - Gtk warning - operation done on background thread.

Xamarin

Xamarin.Forms

- Fixed an issue where IntelliSense does not work for auto-generated fields through Xamarin.Forms XAML.
- We fixed some XAML editing issues for Xamarin.Forms:
  - Xamarin Forms - SomePage.xaml.cs -> InitializeComponent() says it's ambiguous.
  - Ambiguity between same members using a linked XAML file.
  - Fixed an issue where the user is unable to save files and add new resources to the Xamarin.Forms project.

For more information, see the Xamarin release notes.

Xamarin.Android

- If Java 9 or above is installed, Visual Studio for Mac will use `apksigner` to sign Android .apks when publishing through the publishing workflow.
- We fixed an issue where the app would not launch when starting the app in Release configuration.
- You are now notified if Android SDK Tools are missing when starting an emulator] (https://bugzilla.xamarin.com/show_bug.cgi?id=57082).
- We fixed an issue where apps would be reported as failed to install, even though they installed correctly.
- We fixed a crash when publishing to the Play Store.

For more information, see the Xamarin release notes.

Xamarin.Mac

- You can now open a storyboard file through Xcode.
- We fixed a number of visual glitches when publishing Mac apps.

For more information, see the Xamarin release notes.

Xamarin.iOS

Support for ClassKit entitlements has been added in this release. ClassKit enables features in your education apps for iPad that can be used by educational institutions with Apple School Manager and Managed Apple IDs.

Xamarin.iOS Provisioning

Automatic provisioning in Visual Studio for Mac now attempts to create and use wildcard App ID and provisioning profiles by default, instead of creating an App ID based on the Bundle Identifier specified in the Info.plist file. Using wildcard IDs by default reduces the number of profiles and IDs that need maintenance in the developer portal.

For more information on provisioning in Xamarin.iOS, see the Automatic Provisioning in Xamarin.iOS documentation.

We also fixed the following issues with Xamarin.iOS projects:

- Fixed an issue where WatchKit Complication Assets Set specified in Info.plist are not recognized on compile.
- Fixed an issue where the `aps-environment` is not getting set to `production` after publishing a release build.
- Fixed an issue where Asset catalogs in a shared project are listed twice in the solution explorer.
- Fixed an issue where the `Enable Classkit` checkbox in **Entitlements.plist** does not display as checked when the Entitlements.plist file is re-opened.
- Visual Studio for Mac will now alert you if a particular iOS or Mac constructor (from Xamarin.iOS or Xamarin.Mac, respectively) is unavailable due to deprecation or introduced into a later version of the OS than the version the project is targeting (e.g. the "Deployment Target" as set in the Info.plist).
- Fixed an issue where IWKNavigationDelegate has two interface methods, and only one is shown in IntelliSense.
- When connecting the debugger to your app, Visual Studio for Mac now uses the device's connection port instead of using global preferences. This means USB is used for debugging (when connected through USB) even if the preferences are for WiFi debugging.
- Fixed an error when trying to connect to Apple Developer Account.
- Fixed an issue where `$(AppIdentifierPrefix)` is not inserted before the automatically generated bundle ID entry when enabling keychain access groups.
- Fixed an issue with new watchOS projects where the namespace would not match the iOS project.
- Fixed an issue where the issue analyzer would not trigger after changing the deployment target.
- Fixed an issue that prevented profiling a tvOS application over WiFi.

For more information, see the Xamarin release notes.

Visual Designers

iOS Designer

- The selected view's parent is now highlighted when it is not the root of the view controller.
- UINavigationControllers with child view controllers are now supported in XIB files.
- Constraints are now preserved when using Embed/Un-embed in View.
- Items dragged from the toolbox are now displayed as icons until they are dragged over a suitable drop area.

Android Designer

- With full-fidelity layout editing, your XAML modifications are now preserved even in design mode.
- We updated renderer to the latest Android Oreo.
- Improve reliability of custom controls during inflation
- Sample data support for system provided values

Xamarin.Forms Previewer

- Added toolbox support for C# and F# Xamarin.Forms projects
  - When your C# or F# Xamarin.Forms project has a dependency on Xamarin.Forms version 3.1.0.583944 or greater, Xamarin.Forms controls will appear in the toolbox, aiding discoverability for those new to the toolkit.
  - This release includes the ability to drag and drop these controls onto your code editor, providing a XAML element snippet.

Other Fixes and enhancements

- Fixed an issue where Visual Studio for Mac appears to offer no updates.
- Fixed an issue where Visual Studio for Mac checkout functionality freezes entire program.
- Improved navigation and flow of feedback survey.
- Fixed an issue where the assembly browser does not show public members if visibility is "Only public members".
- Fixed an issue where Source Control selection in **Tools -> Options -> Source Control** is lost when opening context menu..
- Fixed an issue where it is not possible to re-run tests after stopping tests.

Visual Studio for Mac 7.6.1 (7.6.1.9)

released August 23, 2018

This release addresses the following issues:

- We fixed an issue where IntelliSense was not working and caused "System.Object is not defined or imported" errors.
- We fixed an issue that caused the IDE to crash when you change the PCL profile.
- We fixed an issue where publishing Azure functions fail when the plan is set to "Consumption".

Visual Studio for Mac 7.6.2 (7.6.2.33)

released August 29, 2018

This release addresses the following issues:

- We fixed an issue where the completion description tooltip sometimes stays open after completion is closed.
- We fixed an issue where there is High CPU usage viewing "Search in Solution" results, including UI delays and unresponsiveness.
- We fixed an issue where `FUNCTIONS_EXTENSION_VERSION` is not set as **beta** in the App Service for a published Azure Function.
- We fixed an issue causing TypeScript/JavaScript TextMate syntax highlighting errors.
- We fixed an issue that caused unexpected UI after pasting c# code from docs sample.
- We fixed an issue where VS for Mac crashes during debug/iphone compile against wifi connected device.
- We fixed an issue where Not able to use enterprise license on Visual Studio for Mac.

We also continued to make improvements to the core IDE to fix IntelliSense issues, hangs, and crashes.

Visual Studio for Mac 7.6.3 (7.6.3.1)

released August 30, 2018

This release addresses the following issues:

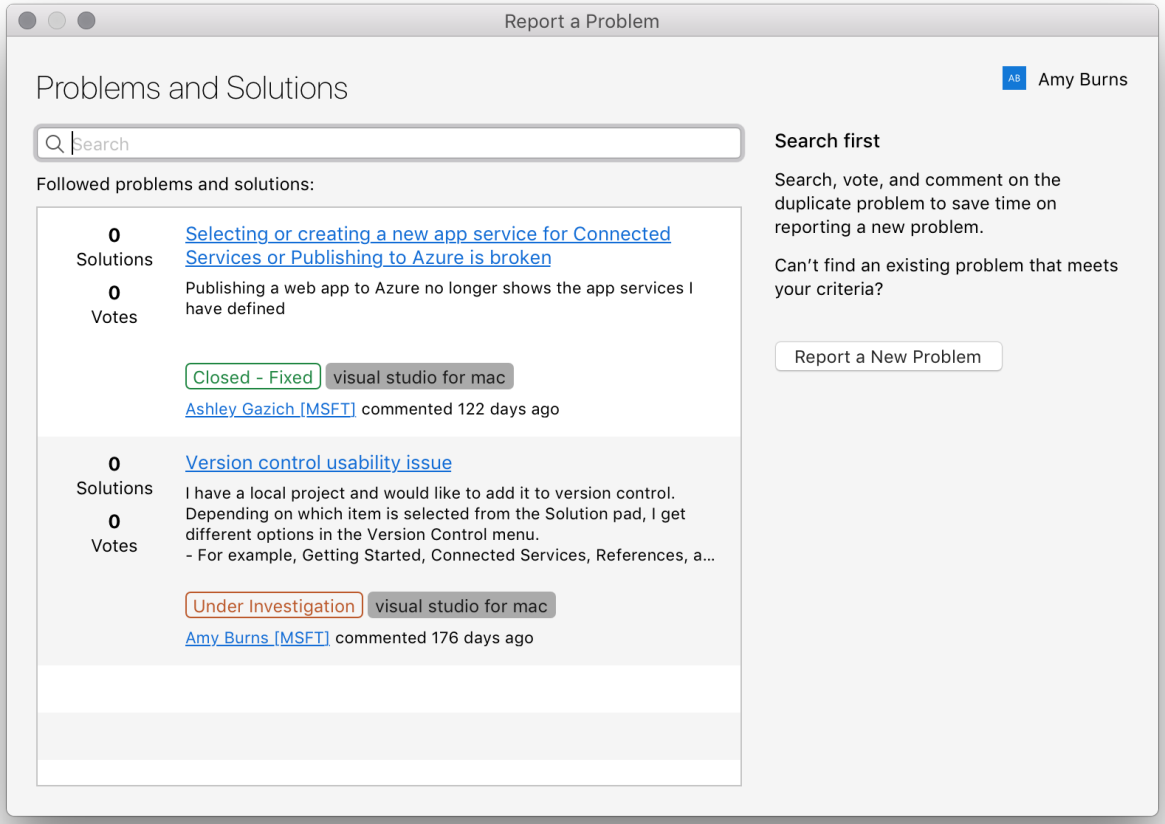
- We fixed an issue where Android APKs being signed with the ad-hoc archiving process resulted in a non-signed APK.

Visual Studio for Mac 7.6.4 (7.6.4.22)

released September 10, 2018

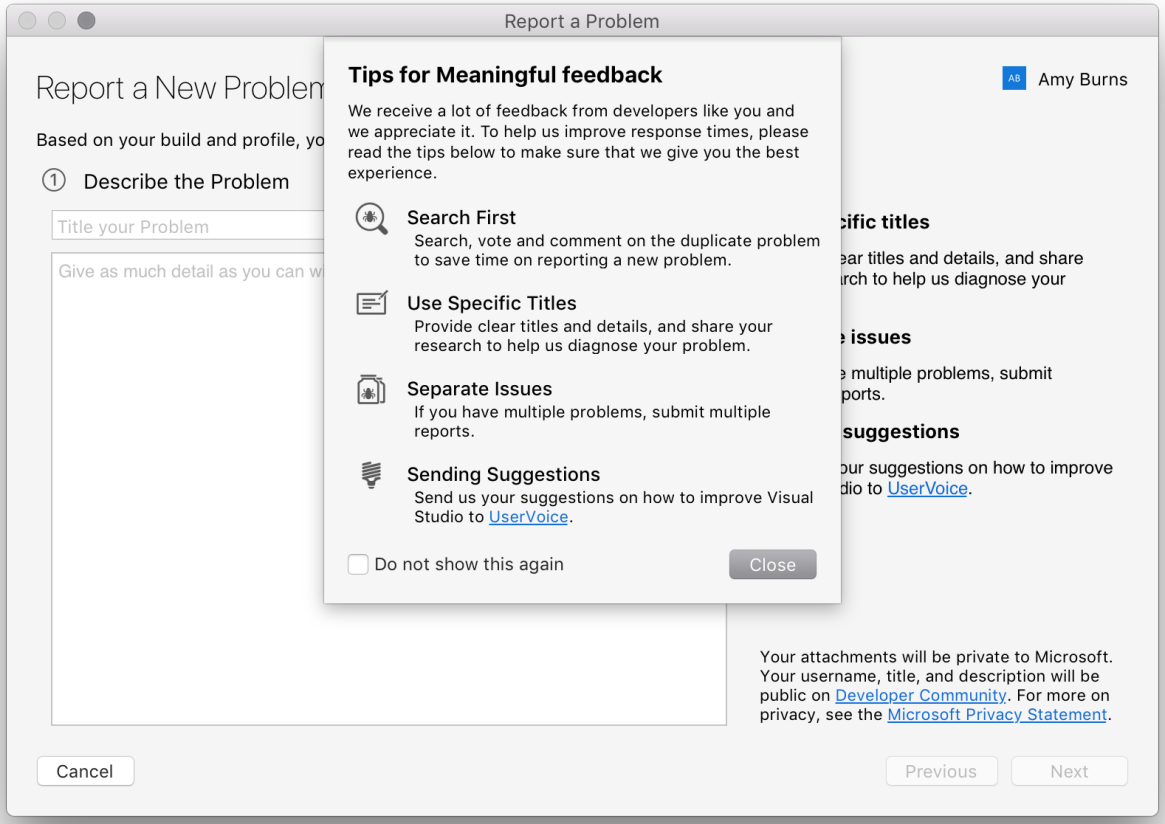
This release introduces a new and easier way to provide feedback to us, straight from the IDE. With this feature we've made it easier for you to create a ticket or find existing tickets for the same issue.

To access the new tool, select **Help > Report a Problem**:



(Figure 7) New Report a Problem dialog

From this dialog you can browse your followed issues, search for similar issues, or report a new problem:



(Figure 8) New Report a Problem dialog

This release also addresses the following issues:

- We fixed an issue where TFS Extension does not show up in gallery
- We fixed an issue where Android Property panel disappears when typing in "Text" property field due to focus issue
- We fixed an issue where the Title bar is not visible with macOS 10.14 beta 4
- We fixed an issue with XAML Syntax Highlighting
- We fixed an issue where "Find in files"/"Find References" results show wrong line numbers.
- We fixed an issue where search results displayed as empty when line that contains the search term is too long.
- We fixed an issue where search results returned do not match search terms.
- We fixed an issue where JSON IntelliSense inserts three quotes where two are needed under certain circumstances.
- We fixed an issue where Unity projects targeting netstandard 2.0 have broken IntelliSense.

Visual Studio for Mac 7.6.5 (7.6.5.20)

released September 13, 2018

This release addresses the following issues:

- We fixed an issue where you could not build a project using vstool after upgrading to version 7.6.
- We fixed an issue where Xamarin.Forms projects did not reference .NET Portable Subset assemblies, breaking IntelliSense.
- We fixed an issue where Visual Studio for Mac status-bar icons were not displaying on macOS Mojave beta.
- We fixed an issue where Visual Studio for Mac crashes randomly while editing XAML files.

- We fixed a JavaScript issue where an incorrect line break is added when inserting a new line.
- We fixed an issue where JSON IntelliSense inserts three quotes where two are needed under certain circumstances. We made a fix for this in the last service release, but it was still appearing. The issue should now be fully fixed.
- We fixed scrolling for Developer News.

Visual Studio for Mac 7.6.6 (7.6.6.17) New

released September 19, 2018

This release adds support for Xcode 10. Xcode 10 provides support for iOS 12, watchOS 5, macOS 10.14, and tvOS 12.

This release also fixes the following issue:

- Objects in List are being garbage collected during debug

Visual Studio for Mac 7.6.7 (7.6.7.49)

released September 25, 2018

This release addresses the following issues:

- We fixed an issue where the user would be continually prompted for git credentials.
- We fixed an issue where files added to a project with soft links on macOS High Sierra were not indexed for IntelliSense.
- We fixed an issue where the height of text in search results were different.
- We fixed an issue where Find References showed invalid search results.
- We fixed an issue where the wrong sections are visually commented out with preprocessor directives.
- We fixed an issue where the NuGet credential provider dialog had tabbing and initial focus problems.
- We fixed an issue where deploying a new Azure App Service for a Function would fail if the Storage Account was in a different Resource Group to the new App Service.
- We fixed a crash that was caused by the IoT extension.
- We fixed an issue that was causing delays when searching.
- We addressed some localization issues in the Feedback Client.

Visual Studio for Mac 7.6.8 (7.6.8.38) New

released October 1, 2018

This release addresses the following issues:

- We fixed an issue where modifying a .csproj file that is inside SolutionFolder doesn't reload the project.
- We fixed an issue where the pick members to override for a class window doesn't render options correctly.
- We fixed an issue where it is not possible to change the font colors for VB.Net in Visual Studio Community 2017 for Mac.
- We fixed an issue where a solution could not be loaded by double clicking on the .sln file.
- We improved our support for Mojave dark mode.

Known Issues

The following is a list of all existing known issues in Visual Studio 2017 for Mac version 7.6:

- Visual Studio for Mac debugger often crashes when debugging Xamarin.iOS.

Team Foundation Version Control

Team Foundation Version Control (TFVC) support is available as an extension for Visual Studio for Mac. For more information on installing and using the extension, see the Connecting to Team Foundation Version Control guide.

Feedback

We would love to hear from you! You can report a problem through the Report a Problem option the Visual Studio for Mac IDE and track your feedback in the Developer Community portal. For suggestions, let us know through UserVoice.

Blogs

Take advantage of the insights and recommendations available in the Developer Tools Blogs site to keep you up-to-date on all new releases and include deep dive posts on a broad range of features.

Developer Tools Blogs

Visual Studio 2017 for Mac Release Notes History

You can view prior versions of Visual Studio 2017 for Mac release notes:

- Visual Studio 2017 for Mac version 7.5 Release Notes
- Visual Studio 2017 for Mac version 7.4 Release Notes
- Visual Studio 2017 for Mac version 7.3 Release Notes
- Visual Studio 2017 for Mac version 7.2 Release Notes
- Visual Studio 2017 for Mac version 7.1 Release Notes
- Visual Studio 2017 for Mac version 7.0 Release Notes

Top of Page

10/2/2018 • 12 minutes to read

Visual Studio 2017 for Mac Preview Release Notes

Developer Community | System Requirements | Compatibility | Distributable Code | Documentation | Blogs | Servicing |

Download

Install the most recent Preview version of Visual Studio for Mac by changing the updater channel to **Beta**.

> **NOTE**
> This release is not "go-live" and not intended for use on production computers or for creating production code.

To learn more about Visual Studio 2017 for Mac, see Mac System Requirements and Mac Platform Targeting and Compatibility.

To learn more about other related downloads, see the Downloads page.

Feedback

We would love to hear from you! You can report a problem through the Report a Problem option in Visual Studio for Mac IDE. You can track your feedback in the Developer Community portal. For suggestions, let us know through UserVoice.

Release History

- October 2, 2018 – Visual Studio 2017 for Mac version 7.7 Preview 3 New
- September 12, 2018 – Visual Studio 2017 for Mac version 7.7 Preview 2
- August 23, 2018 – Visual Studio 2017 for Mac version 7.7 Preview 1

Visual Studio 2017 for Mac version 7.7 Preview 1 (7.7.0.977)

released August 23, 2018

Code Editor

For this release we fixed the following Code Editor issues:

- Fixed an issue where incorrect colors were displayed when dragging a document tab.
- Fixed an issue where Ctrl+Right and Ctrl+Left would skip too much text in the editor.
- Fixed an issue where Cmd-click navigation sometimes gets stuck.
- Fixed an issue where Cmd does not light up navigation for symbol under mouse
- Fixed an issue where Breakpoints and caret location were not displayed in the scrollbar.
- Fixed an issue where Extract Interface dialog layout would flicker.
- Fixed an issue where "Copy comments from base" refactoring would suggest including comments from the base class library.
- Fixed an issue where it was not possible to see the mouse over content when using the keyboard.

- Fixed an issue where `CopyCommentsFromBaseCodeRefactoringProvider` would throw an XML parse exception.
- Fixed an issue where an error in document creation causes a cascade of errors leading to a crash.
- Fixed an issue where the text Editor outdents to hard left instead of going back an indent level.
- Fixed an issue where the editor would not render whitespace marks for virtual space.
- Fixed an issue where the regex completion provider would crash.
- Fixed an issue where C# completion improperly escapes \ within @ strings.
- Fixed an issue where code completion would incorrectly insert code.
- Fixed an issue where the IntelliSense window would obscuring code when "Show function type signatures" is enabled.
- Fixed an issue where Minimap could be turned off.
- Fixed an issue where Razor completion would not commit the change if the caret is immediately followed by '}'.
- Fixed an issue where renaming a class would ask to reload the file.
- Fixed an issue where "[TestCaseSource ("xzy")]" was not seen as a test in the text editor.
- Fixed an issue where Alt-shift-left/right would not extend the selection.
- Fixed an issue where an error would be logged for the Text Mate text editor extension.
- Fixed an issue where it was not possible to configure tabs in XAML files with an EditorConfig file.
- Fixed an issue where the Tab size in the editor was wrong.

F#

- 'Find Reference Usages' is now supported.
- 'Find Implementing Symbols' is now supported.
- Fixed an issue where a fatal exception could occur when building a F# project.
- Fixed an issue where the IDE would hang when getting extension methods for F#.

Xamarin

This release includes the following bug fixes and improvements for Xamarin:

Xamarin.Android

- The Device Manager startup time has been optimized.
- The available system images now reflect the available options for the selected manifest. This enables Android P support for the device manager when the google manifest is selected in the SDK manager.
- The device manager no longer relies on the `JAVA_HOME` environment variable being set and it is exported it for processes that require it.
- Fixed an issue where AVD Rename would crash the app for the custom `ANDROID_AVD_HOME` case.
- Fixed an issue where the Device Manager would not indicate there had been system image installation failure.
- Fixed an issue where jarsigner would be used instead of apksigner.
- Fixed an issue where it would not be possible to run an Android app in the emulator.
- Building an Android project when the Java SDK cannot be found is correctly reported as a build error instead of reporting that the Android SDK cannot be found.

Xamarin.iOS

- Fixed an error updating an Xcode project.
- Fixed an issue where Automatic iOS provisioning profile dialog did not have clickable links.
- Fixed an issue where archiving would use a generic iOS device target even if a specific device is selected.

Xamarin.Mac

- Added support for changing the .NET Framework version to use for "Full Framework" projects. Projects are no longer limited to just .NET 4.5.

Other

- Fixed an issue where the version control Diff view would not show the correct changed text.

- Fixed an issue where the Class pad shows compiler generated fields and constructors.
- Fixed an issue where an error was logged when the news feed could not be retrieved.
- Fixed an issue where the updater would not display "Checking for updates..." until the channel was changed.
- Fixed an issue where the Sign in button for publishing to Azure was not rendered correctly.
- Fixed an issue where running unit tests always ran a full build.
- Fixed an issue with running NUnit 3 tests due to a misnamed test runner application.
- The Welcome page is no longer displayed when loading a solution from Finder.

Visual Studio 2017 for Mac version 7.7 Preview 2 (7.7.0.1282)

released September 12, 2018

Code Editor

- We aligned quote matching behavior to match Visual Studio.
- We fixed an issue where Cut and Paste would not convert `@""` and `""` formatting automatically.
- We fixed an issue where code snippets would fail to load when there is a malformed snippet.
- We fixed an issue where the indentation in multi-line field initializers is wrong when working with arrays.
- We fixed an issue where .editorConfig doesn't work when using .NET Core Rules.
- We fixed an issue where C# indentation inside method is not aligning with existing code.
- We fixed an issue where there is inconsistent indentation between VS Windows and VS Mac.
- We fixed an issue where it is not obvious which "fix all" is which.

Project

- We fixed an issue where intermittent errors would appear after creating a new .NET Core project.
- We fixed an issue where OutputBase no longer includes the build configuration.
- We fixed an issue where Visual Studio for Mac builds all projects instead of just the executable and dependencies only when running the project.
- We fixed an issue where the project GUID changes when building in VS2017 vs building in Visual Studio for Mac.

Accessibility

- We fixed the following issues:
  - there was a loss of focus when configuring the location of a new project.
  - is not possible to invoke the "Error" window and "Visual Studio Update" window using the keyboard.
  - VoiceOver announces the text typed into password secured fields.
  - VoiceOver focus is not moving to the search results along with the keyboard focus.

.NET Core

- We added support for .NET Core projects that use explicit Sdk imports.

  For example:

  ```
  <Import Sdk="Microsoft.NET.Sdk" Project="Sdk.props" />
  ```

  ```
  <Import Sdk="Microsoft.NET.Sdk" Project="Sdk.targets" />
  ```

  ```
  <Sdk Name="Microsoft.NET.Sdk" Version="1.0.0" />
  ```

Feedback Client

- We fixed an issue where the Feedback dialog sometimes disappears and is always visible in screenshot.

Web Tools

- We fixed an issue where pressing Tab doesn't complete the statement in .cshtml files.
- We fixed an issue where an error is thrown when inserting new line between `<div>` and `</div>` in HTML files.

Xamarin

Xamarin.iOS

- We added support for editing Background Modes for WatchKit Extensions.
- We added support for CarPlay assets.

Xamarin.Android

- We fixed an issue where Visual Studio for Mac would fail to show new API levels in Minimum and Target Android version drop downs.
- We improved the error messaging when no JDK is installed. Previously it was not obvious whether the JDK or the Android SDK was not installed.

Visual Studio 2017 for Mac version 7.7 Preview 3 (7.7.0.1470)

released October 2, 2018

Code Editor

- We fixed an issue were an exception is thrown when using syntax highlighting in Visual Studio for Mac 7.6.
- We fixed an issue where Visual Studio for Mac IntelliSense does not work correctly in version 7.6.1 (build 9).
- We fixed an issue where Visual Studio for Mac IDE themes look wrong.
- We fixed an issue where deleting emoji in source file creates gray bars.
- We fixed an issue where sorting usings does not work.
- We fixed some editor IntelliSense/Code Generation bugs.
- We fixed an issue where completion suggestion mode does not work as expected in collection initializers.
- We fixed an issue where format document does not use the correct formatting settings.
- We fixed an issue where the indentation after end of method is incorrect.
- We fixed an issue where cut and paste is broken in the editor.
- We fixed an issue with the Oblivion color style.
- We fixed an issue where some C# templates do not work when invoked in the editor.

Shell

- We fixed an issue where there are no tooltips on toolbar.
- We fixed an issue where new Application Output windows are created on each run.

Web Tools

- We fixed an issue where tooltips do not close in Visual Studio for Mac.
- We fixed an issue where the user could not type in .json files.

NuGet

- We fixed an issue where PackageReferences with no version were not supported.

Version Control

- Subversion support on Mac should depend on either Xcode or the CommandLineTools, not just Xcode.

Xamarin

Xamarin.iOS

This release adds support for new ARKit 2.0 asset catalog resources: `ARReferenceGroup`, `ARReferenceImage` and `ARObject`.

ARKit 2.0 supports 2D and 3D objects detection. `ARReferenceImage` is the new asset catalog type for 2D images

that can be used and recognized by ARKit. *Width* and *Height* of the `ARReferenceImage` can be set in the asset editor so ARKit can render virtual overlays of the correct dimensions. The `ARObject` type takes an '.arobject' file that can be created by some ARKit samples.

We also made the following improvements:

- We updated our CoreML support:
  - CoreML Model Viewer has been updated to display the new Custom Layers that were introduced with Xcode 10 and iOS 12.
  - CoreML code-behind generator has been updated to add more convenience methods.
- We fixed an issue that caused random crashes - Xamarin.MacDev.MobileDeviceException: Device is invalid.
- We fixed an issue where the iOS project properties window for image resource sets incorrect property in .csproj file.
- We fixed an issue with migration from WatchKit to Xamarin.WatchOS.

Xamarin.Android

- With Visual Studio 2017 15.9 and Visual Studio for Mac 7.7, we are moving from Oracle's JDK to a lightweight distribution of Open JDK meant for mobile development. Open JDK will be offered as a component to install along with other Visual Studio for Mac updates and you will be prompted to update your JDK settings to use Open JDK once installed.

Other

- We fixed an issue where the UnitTest text editor extension causes slowness in the IDE.
- We made the following accessibility improvements:
  - better contrast in the New Project Dialog
  - better contrast in the Exception dialog
  - VoiceOver improvements

Team Foundation version control extension – Release Notes

Team Foundation version control support is available as an extension for Visual Studio for Mac. For more information on installing and using the extension, see Connecting to Team Foundation version control.

0.3.2

Released 7/6/2018

Implemented enhancements:

- Added functionality to change the paths of a WorkingFolder in a Workspace.
- Added hightlights filtering projects or paths.
- Improved the amount and details of the feedback messages, including, but not limited to: validations and errors.
- Minor UI changes.

Fixed bugs:

- Fixed critical bug obtaining the list of projects from a collection on some TFS On premise servers.
- Fixed critical bug getting a bad behavior of the version control menu with a Git project and the TFS addin installed.
- Fixed bug loading remote workspaces.

0.3.1

Released 6/28/2018

Implemented enhancements:

- Improved Source Explorer experience. The Refresh button has been removed since everything is now refreshed automatically.
- Simplified workspaces creation process.
- Minor UI changes.

Fixed bugs:

- Fixed bug with folders validation. It could affect the creation of workspaces.

0.3

Released 6/27/2018

Implemented enhancements:

- Improved project filter. Added option to filter by server URL and username.
- Improved Source Explorer loading time.
- Improvements to the UI refresh process in Source Explorer. Some of the UI elements that have been improved are: The loading indicator and the enable/disable buttons).
- Added more options to Source Explorer like open folders in Finder, etc.
- Improved Work Items loading time.
- Changes in some literals to adapt TFVC branding.
- More validations: check if the workspace local path exists, etc.
- Minor UI changes.

Fixed bugs:

- Added workspace cache to avoid unnecessary request validating project (big performance improvement).
- Fixed bug opening Source Explorer.
- Fixed bug creating workspaces. The workspace creation takes some time, moved to another thread.
- Fixed bug launching OAuth dialog from Source Explorer.
- Fixed bug relating to adding new files from Source Explorer when there were not any working folder specified in the workspace.
- Fixed some incorrect literals.
- Fixed bug with focus on Open from Source Control dialog.
- Fixed bug that refreshed Source Explorer content after creating a new workspace.

0.2.1

Released 6/11/2018

Fixed bugs:

- Fixed bug detecting if a project is in some of the TFS Servers (performance).

0.2

Released 6/11/2018

Implemented enhancements:

- New authentication flow.
- New projects dialog where can easily manage all servers, collections and projects.
- Added auto discover servers and projects functionality.
- Added project filter.
- New dialog where select project paths to map.
- Added files filter.

- Added new dialog to create workspaces.
- Added new option to delete working folders from an existing workspace.
- Included more detailed forms validation.
- Improved error management and user feedback. More detailed messages.
- Improvements in SourceControlExplorer: file types are detected and specific icons are displayed, etc.
- Improved loading times in SourceControlExplorer.
- Added new logs option in the Settings.
- Improvements in addin cache (settings, tokens, etc).
- Minor UI changes and improvements (loading indicators, etc).

Fixed bugs:

- Fixed bug authenticating in some TFS Servers using NTLM.
- Fixed bug creating workspaces related to workspaces cache.
- Fixed bug refreshing workspace after removing a project.
- Fixed bug refreshing OAuth token under some conditions.
- Fixed bug changing Visual Studio theme and minor UI details.
- Fixed bug getting the workspace owner name.

0.1.1 (Beta)

Released 5/8/2018

Fixed bugs:

- Fixed bug loading OAuth Webview from some Visual Studio for macOS versions.

0.1 (Beta)

Released 5/8/2018

Implemented enhancements:

- Basic and OAuth Authentication.
- Choose server projects.
- Create, edit and delete Workspaces.
- SourceControlExplorer.
- Map and get.
- Added checkout functionality.
- Added checkin functionality.
- Added rename functionality.
- Added delete folders and files functionality.
- Added lock and unlock functionality.
- Added undo changes functionality.

Visual Studio 2017 for Mac Preview Release Notes History

You can view prior Preview versions of Visual Studio 2017 for Mac release notes:

- Visual Studio 2017 for Mac version 7.6 Preview Release Notes

---

Top of Page

9/7/2018 • 2 minutes to read

Visual Studio 2015 Release Notes History

| Developer Community | System Requirements | Compatibility | Distributable Code | License Terms | Blogs | Known Issues |

NOTE

This is not the latest version of Visual Studio. To download the latest release, please visit the current release notes for Visual Studio 2017.

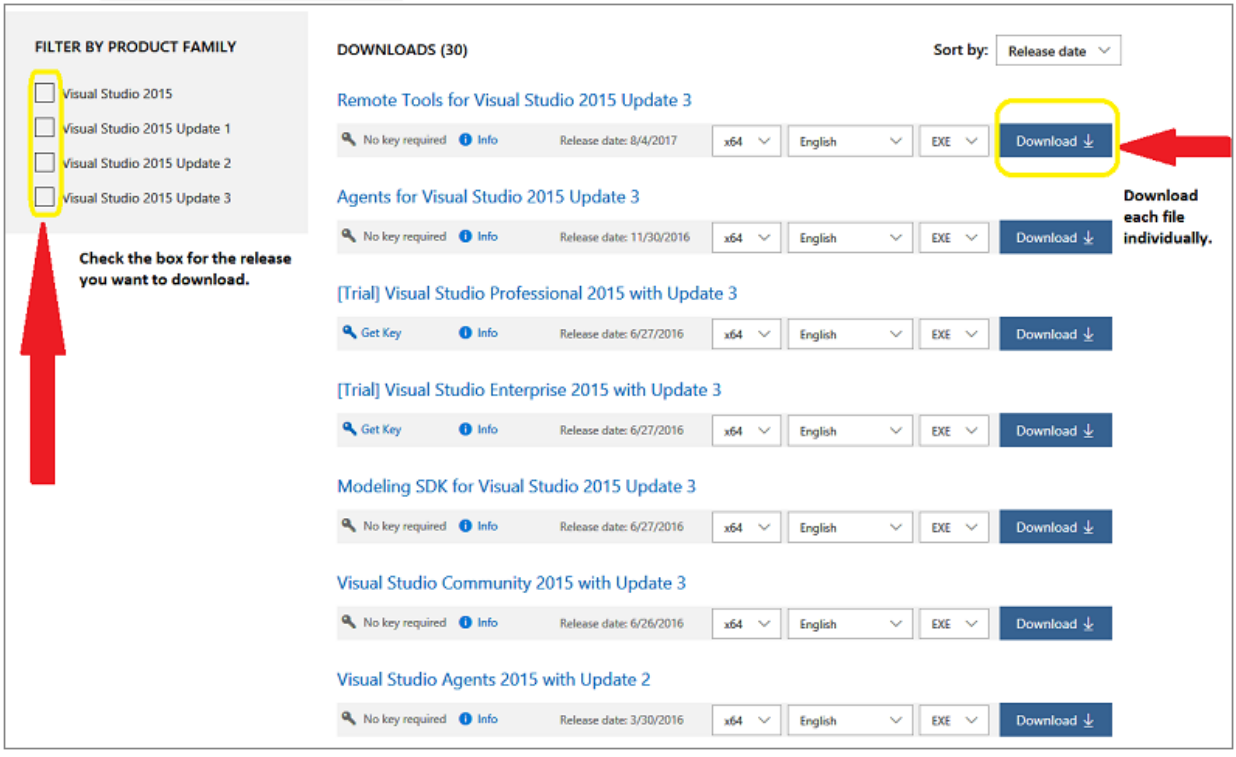
Download Visual Studio 2015

To download Visual Studio 2015, click on the download button. The files are downloaded from our free Dev Essentials subscription-based site. Please see the Dev Essentials page for information on our free comprehensive developer program and services.

Download Visual Studio 2015

Follow these steps to download:

1. Click the **Download Visual Studio 2015** button.
2. Check the desired product family in the *Filter by Product Family* box. In *Downloads*, click the download button for each product you wish to download in that family (Figure 1).



(Figure 1) Available downloads for Visual Studio 2015.

3. Repeat steps 1 and 2 for each product family you wish to download.

Visual Studio 2015 Releases

- June 27, 2016 -- Visual Studio 2015 Update 3
- March 30, 2016 -- Visual Studio 2015 Update 2
- November 30, 2015 -- Visual Studio 2015 Update 1
- July 20, 2015 -- Visual Studio 2015

Visual Studio 2015 Product Information

- SHA-1 Values

Visual Studio Blogs

You can find archived information for Visual Studio 2015 in the Developer Tools Blogs site. These blogs provide product insight on a broad range of features.

Developer Tools Blogs

Top of Page

9/7/2018 • 2 minutes to read

Visual Studio 2013 Release Notes History

| Developer Community | System Requirements | Compatibility | Distributable Code | License Terms | Blogs | Known Issues |

NOTE

This is not the latest version of Visual Studio. To download the latest release, please visit the current release notes for Visual Studio 2017.

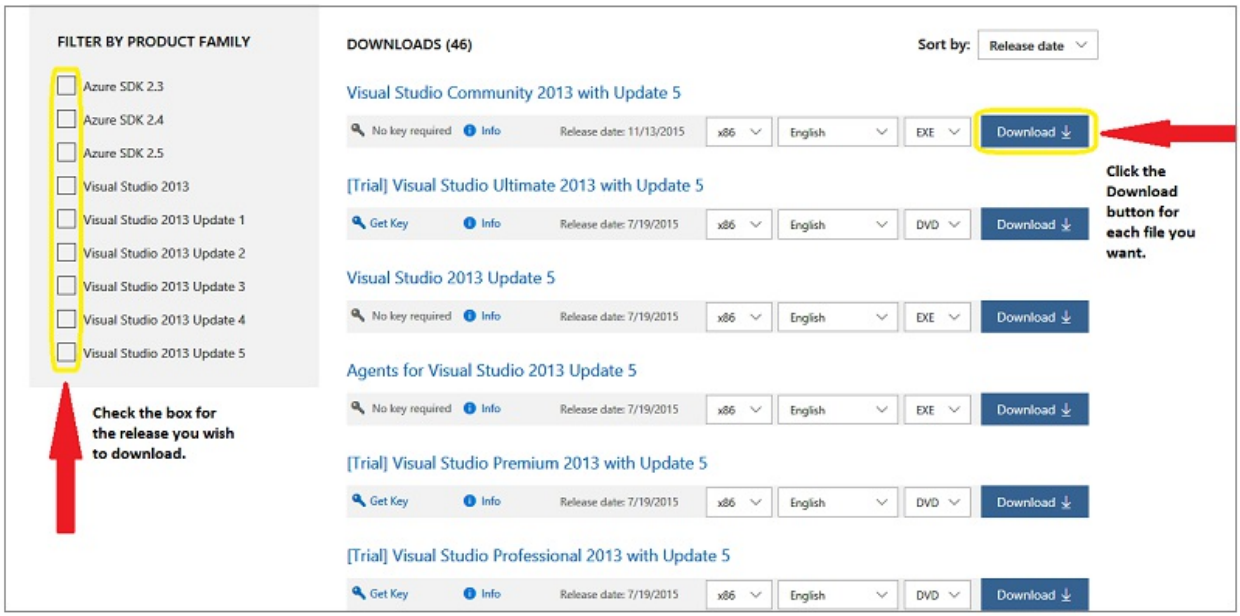
Download Visual Studio 2013

To download Visual Studio 2013, click on the download button. The files are downloaded from our free Dev Essentials subscription-based site. Please see the Dev Essentials page for information on our free comprehensive developer program and services.

Download Visual Studio 2013

Follow these steps to download:

1. Click the **Download Visual Studio 2013** button.
2. Check the desired product family in the *Filter by Product Family* box. In *Downloads*, click the download button for each product you wish to download in that family *(Figure 1)*.



(Figure 1) Available downloads for Visual Studio 2013.

3. Repeat steps 1 and 2 for each product family you wish to download.

Visual Studio 2013 Releases

- July 20, 2015 -- Visual Studio 2013 Update 5
- November 12, 2014 -- Visual Studio 2013 Community
- November 12, 2014 -- Visual Studio 2013 Update 4
- August 4, 2014 -- Visual Studio 2013 Update 3
- May 12, 2014 -- Visual Studio 2013 Update 2
- April 2, 2014 -- Visual Studio 2013 Update 2
- January 20, 2014 -- Visual Studio 2013 Update 1
- October 17, 2013 -- Visual Studio 2013

Visual Studio 2013 Product Information

- SHA-1 Values

Visual Studio Blogs

You can find archived information for Visual Studio 2013 in the Developer Tools Blogs site. These blogs provide product insight on a broad range of features.

Developer Tools Blogs

Top of Page

9/7/2018 • 2 minutes to read

Visual Studio 2012 Release Notes History

| Developer Community | System Requirements | Distributable Code | License Terms | Blogs | Known Issues |

NOTE

This is not the latest version of Visual Studio. To download the latest release, please visit the current release notes for Visual Studio 2017.

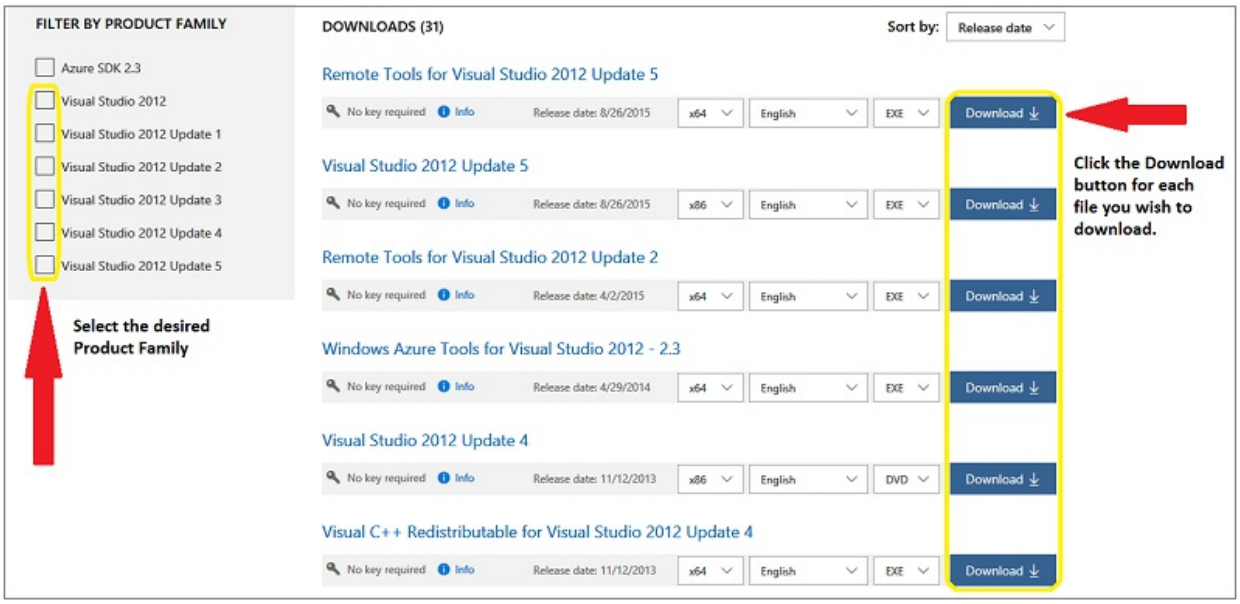
Download Visual Studio 2012

To download Visual Studio 2012, click on the download button. The files are downloaded from our free Dev Essentials subscription-based site. Please see the Dev Essentials page for information on our free comprehensive developer program and services.

Download Visual Studio 2012

Follow these steps to download:

1. Click the **Download Visual Studio 2012** button.
2. Check the desired product family in the *Filter by Product Family* box. In *Downloads*, click the download button for each product you wish to download in that family *(Figure 1)*.



(Figure 1) Available downloads for Visual Studio 2012.

3. Repeat steps 1 and 2 for each product family you wish to download.

Visual Studio 2012 Releases

- November 13, 2013 -- Visual Studio 2012 Update 4
- June 26, 2013 -- Visual Studio 2012 Update 3
- April 4, 2013 -- Visual Studio 2012 Update 2
- November 26, 2012 -- Visual Studio 2012 Update 1

Visual Studio Blogs

You can find archived information for Visual Studio 2012 in the Developer Tools Blogs site. These blogs provide product insight on a broad range of features.

Developer Tools Blogs

Top of Page

9/7/2018 • 2 minutes to read

Visual Studio 2010 Release Notes History

| Developer Community | License Terms | Blogs | Known Issues |

> **NOTE**
>
> **This is not the latest version of Visual Studio.** To download the latest release, please visit the current release notes for Visual Studio 2017.

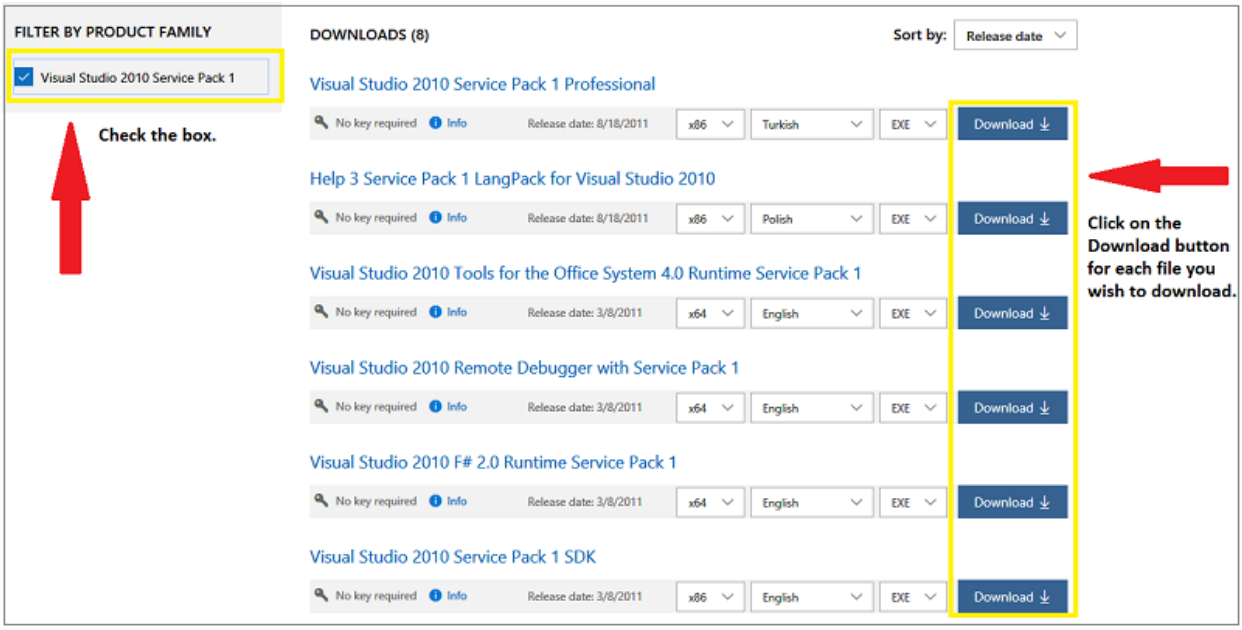
Download Visual Studio 2010

To download Visual Studio 2010, click on the download button. The files are downloaded from our free Dev Essentials subscription-based site. Please see the Dev Essentials page for information on our free comprehensive developer program and services.

Download Visual Studio 2010

Follow these steps to download:

1. Click the **Download Visual Studio 2010** button.
2. Check the box in *Filter by Product Family*. In *Downloads*, click the download button for each product you wish to download *(Figure 1)*.



(Figure 1) Available downloads for Visual Studio 2010.

3. Repeat steps 1 and 2 for each product family you wish to download.

Visual Studio 2010 Releases

- August 18, 2011 -- Visual Studio 2010 Service Pack 1

Visual Studio Blogs

You can find archived information for Visual Studio 2010 in the Developer Tools Blogs site. These blogs provide product insight on a broad range of features.

Developer Tools Blogs

Top of Page

Team Foundation Server 2018 Update 3 Release Notes

9/12/2018 • 5 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

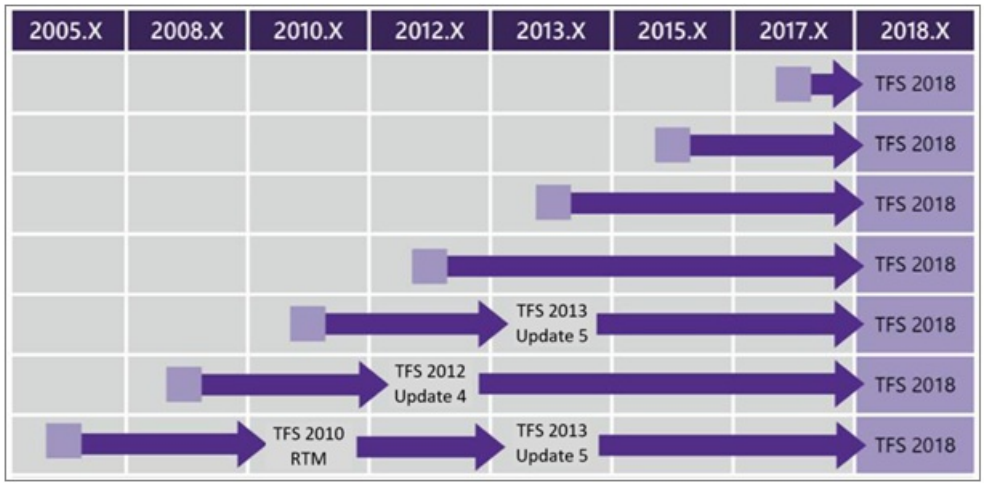
If you are accessing this page from a non-English language version, and want to see the most up-to-date content, please visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding the newest release for Team Foundation Server 2018. Click the button to download.

Team Foundation Server 2018

To learn more about Team Foundation Server 2018, see the Team Foundation Server Requirements and Compatibility page. Visit the visualstudio.com/downloads page to download other TFS 2018 products.

Direct upgrade to Team Foundation Server 2018 Update 3 is supported from TFS 2012 and newer. If your TFS deployment is on TFS 2010 or earlier, you need to perform some interim steps before upgrading to TFS 2018 Update 3. Please see the chart below and the TFS Install page for more information.



TFS Upgrade Matrix

IMPORTANT

You do not need to upgrade to TFS 2018 RTM before upgrading to TFS 2018 Update 3.

Release Date: September 12, 2018

Summary of What's New in TFS 2018 Update 3

Team Foundation Server 2018 Update 3 includes bug fixes for Team Foundation Server 2018. It includes fixes in the following areas:

- Code
- Work
- Build and Release
- Test
- Reporting
- Administration

Details of the bugs fixed in TFS 2018 Update 3

Code

- "There is a problem on the server" error appears in Visual Studio when doing a code review.
- Large TFVC repos take a long time for search indexing. Users can now exclude folders from indexing to speed it up.
- Code search may be slow on collections with a high number of files.
- When code search jobs fail, job yield data is deleted, which causes the next jobs to restart indexing.
- Code search considers the underscore as a special character when it shouldn't.
- A security patch for Git clients was released since TFS 2018 Update 2. To protect unpatched Git clients, we made a change in TFS 2018 Update 3 to reject pushes that exploit the vulnerability. For more information, see Remediating the May 2018 Git Security Vulnerability.

Work

- The + icon is missing on the backlog page.
- The **Name** and **DisplayName** properties are not set in all legacy work items APIs.
- The attachments REST API does not support a **FileID** parameter to set the attachment URL.
- Work item resources and attachment resources sometimes returned project-scoped URLs, which were breaking backwards compatibility.

Build and Release

- Builds are not getting deleted based on the build retention policy.
- Deleting a build does not delete the drop location or symbols.
- A build will not queue if the build number format string results in an invalid build number.
- Build task versions get automatically updated when upgrading TFS.
- Performance issues in XAML builds with many build definitions.
- Build definitions migrated from TFS 2017 get a "definition.Repository.Mappings.Mapping.LocalPath" error.
- The link to Jira items from the Release Summary or Deploy Environment pop up does not work.
- A pending approval notification for a deployment is not delivered when TFS is installed in German locale.
- Task groups variable detection has started recognizing Build.BinariesDirectory as a system variable.
- "Cannot insert duplicate key row in object 'Release.tbl_TagString' with unique index 'PK_tbl_TagString'" error when adding a tag to a release.
- Deployments get cancelled if gates evaluation exceeds six hours.
- "TF400898 An internal error occurred. ActivityId" error occurs when adding or editing artifacts in release definitions.
- Release variables like Release.Reason can be used in custom phase conditions.
- "Lock Hierarchy violation" error occurs when deleting deployment pools.

- A release job fails when a path variable has square brackets.
- Azure Virtual Machine Scale Sets are not updated when the deployment script is updated.
- A release definition does not save when a user with **edit release definition** permissions, but no **release approver** permissions tries to edit the definition.
- The Azure App Service Deploy task version 3 is now available.

Test

- The error, "Expecting end of string. The error is caused by <<->>" occurs when loading test suites.
- "Argument out of range" error occurs when clicking on the **New Test Case** button.
- The Release Path of a bug is incorrectly changed after linking it to a Test Result.
- The Test Run is In Progress even though the test is marked as pass or fail.
- When invoking the **Update Test Result API** and passing the same test result multiple times, a primary key violation exception is raised from SQL.
- Exporting a test case with shared steps to email may fail due to email size limits.
- The Title column pastes incorrectly from Excel when using **Add New Tests** with the grid.
- In the Test Plan grid view, the shared step names are not escaped correctly, such as with the '<' character.

Reporting

- TFSConfig addProjectReports does not add reports if the folder already exists.

Administration

- When TFS databases are hosted on non-enterprise edition of SQL Server 2016 SP1 or above, page compression is not enabled on several tables during upgrade from TFS 2012 or 2013, which has a negative impact on upgrade and runtime performance.
- "Update PR Merge service hook subscriptions" error occurs when upgrading to TFS 2018.
- The **Configure-GvfsCacheServer.psm1** file is being copied during TFS Proxy installs, even though the file cannot be used to configure the proxy.
- "TF400856: The following service is not registered in the database" error occurs when running TFSConfig OfflineDetach.
- Indexes are automatically enabled in the node configuration of Elastic Search.
- Re-indexing is not triggered when Elastic Search is newly configured and has stale data.
- In case of high-volume job failures, the search indexer pipeline does not throttle itself and has potential high resource usage.
- The Elastic Search service installation fails if the ES_JAVA_OPTS environment variable is set.
- When a collection is deleted, collection-level search records are not deleted.
- The process template editor displays errors such as "Requested value 'MANAGE_TEST_SUITES' was not found".
- The process template editor incorrectly shows the collection name in some identities while editing a workflow.
- When setting up a service hook on a code branch, the branch is set back to **[Any]** after saving.
- There are mail delivery errors due to a small timeout value.
- Basic authorization is now enabled on the communication between the TFS and Search services to make it more secure. Any user installing or upgrading to Update 3 will need to provide a user name / password while configuring Search (and also during Search Service setup in case of remote Search Service).
- "The user does not have a license for the extension" error occurs when purchasing or assigning licenses to extensions.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get

advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2018 Update 2 Release Notes

9/7/2018 • 45 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

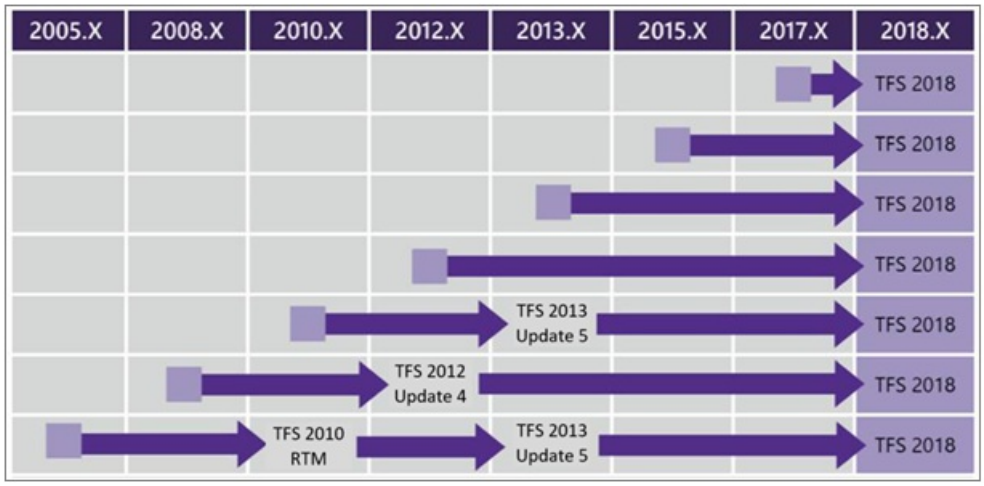
If you are accessing this page from a non-English language version, and want to see the most up-to-date content, please visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding the newest release for Team Foundation Server 2018. Click the button to download.

Team Foundation Server 2018

To learn more about Team Foundation Server 2018, see the Team Foundation Server Requirements and Compatibility page. Visit the visualstudio.com/downloads page to download other TFS 2018 products.

Direct upgrade to Team Foundation Server 2018 Update 2 is supported from TFS 2012 and newer. If your TFS deployment is on TFS 2010 or earlier, you need to perform some interim steps before upgrading to TFS 2018 Update 2. Please see the chart below and the TFS Install page for more information.



TFS Upgrade Matrix

IMPORTANT

You do not need to upgrade to TFS 2018 RTM before upgrading to TFS 2018 Update 2.

Release Date: May 7, 2018

You can now upgrade to TFS 2018 Update 2 and continue to connect your XAML controllers and run XAML builds. When we removed support for XAML build in TFS 2018 RTW and Update 1, some of you could not upgrade due to having legacy XAML builds, and we want to unblock you. Although TFS 2018 Update 2 supports XAML builds for your legacy builds, XAML build is deprecated and there will be no further investment, so we highly recommend converting to a newer build definition format. See the Evolving TFS/Team Services build automation capabilities blog for more information about XAML build deprecation.

Summary of What's New in TFS 2018 Update 2

We have added a lot of new value to Team Foundation Server 2018 Update 2. Some of the highlights include:

- View pull request merge commit
- Help reviewers using pull request labels
- Mention a pull request
- Pull request comment notifications include the thread context
- Pull request status extensibility
- Custom fields and tags filter in work item tracking notifications
- Modernized column options
- Added support for Not In query operator
- Query for @MyRecentActivity and @RecentMentions
- Filtering on plans
- Release gates
- Build with continuous integration from GitHub Enterprise
- Enhancements to multi-phase builds
- Skip scheduled builds if nothing has changed in the repo
- Seamlessly use public packages using upstream sources
- Retention policies in TFS feeds
- Filtering in package management
- Wiki search
- Reference work items in Wiki
- Preview content as you edit Wiki pages
- Paste rich Wiki content as HTML
- Profile cards
- Circle avatars

Details of What's New in TFS 2018 Update 2

You can find details about features in each area:

- Code
- Work
- Build and Release
- Package
- Test
- Wiki
- General

Code

Get a permanent link to code

When viewing a file, you usually see the version at the tip of the selected branch. The version of a file at the tip may change with new commits. If you copy a link from this view, your links can become stale because the URL only includes the branch name, not the commit SHA. You can now easily switch the **Files** view to update the URL to refer to the commit rather than the branch. If you press the "y" key, your view switches to the tip commit of the current branch. You can then copy permanent links.

Recover a recently-deleted repository through API

Sometimes mistakes can be made when cleaning up old repositories in source control. If a Git repository is deleted within the last 30 days, it can be recovered through the REST API. See the documentation for the list and recover operations for more information.

SSH: Support additional ciphers/keys and deprecate outdated ciphers

To improve security and compatibility, we have updated the list of ciphers supported for SSH. We have added two new ciphers and deprecated three, matching OpenSSH's direction. The deprecated ciphers continue to work in this release. They will be removed in the future as usage falls off.

Added:

- AES128 CTR
- AES256 CTR

Deprecated:

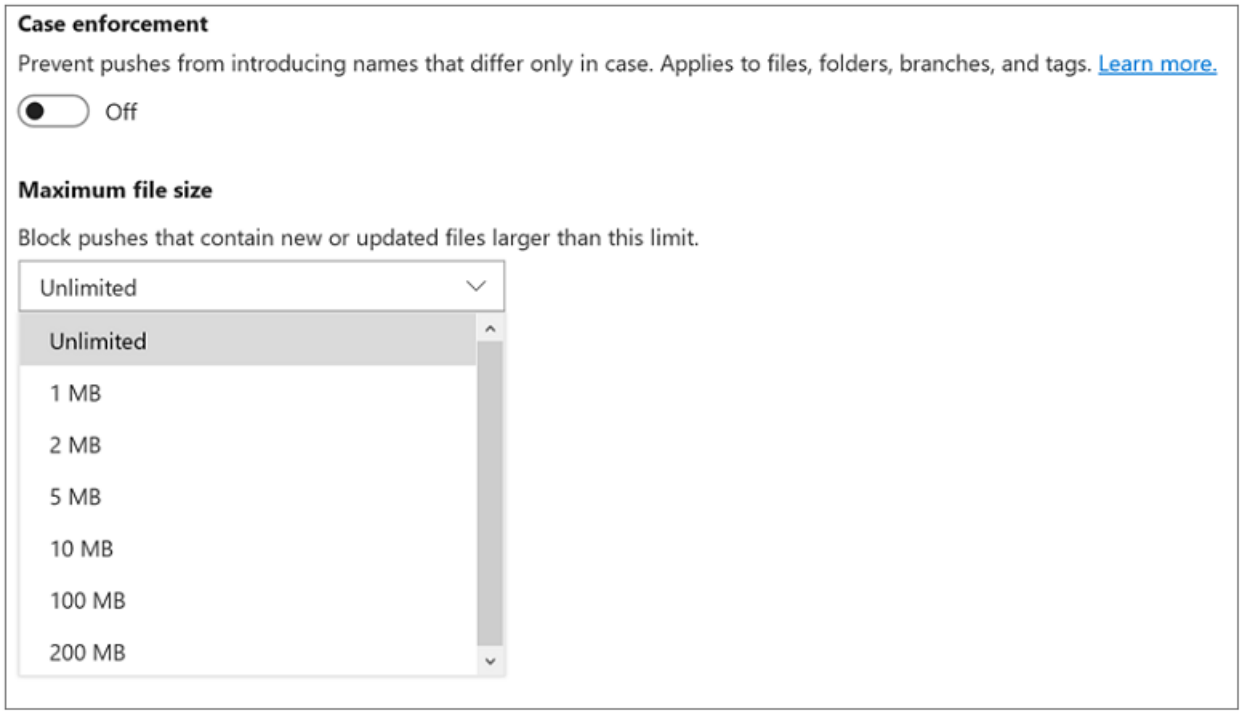
- AES128
- AES192
- AES256

Avoid overwrites and protect performance using repository settings

In this Update, you will find two new repository settings to help keep Git running smoothly.

Case enforcement switches the server from its default case-sensitive mode, where "File.txt" and "file.txt" refer to the same file, to a Windows and macOS-friendly mode where "File.txt" and "file.txt" are the same file. This setting affects files, folders, branches, and tags. It also prevents contributors from accidentally introducing case-only differences. Enabling case enforcement is recommended when most of your contributors are running Windows or macOS.

Limit file sizes allows you to prevent new or updated files from exceeding a size limit you set. The greater number of large files that exist in a Git repository's history, the worse clone and fetch operation performance becomes. This setting prevents accidental introduction of these files.

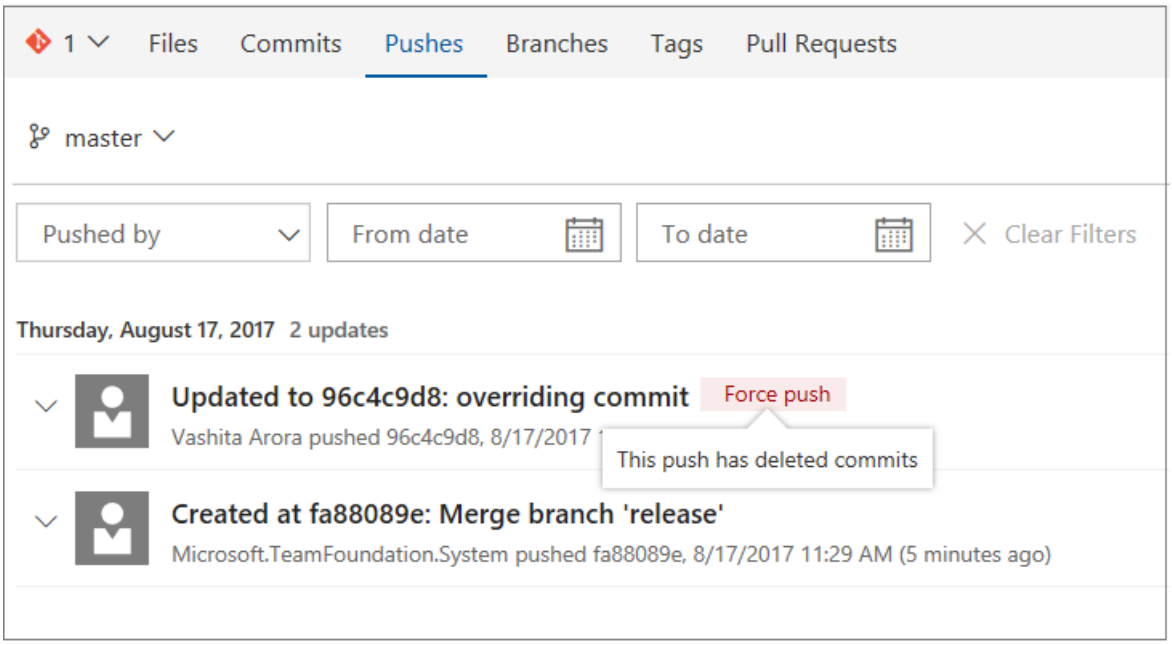


Enhanced filter capability for commits with more than 1000 files changed

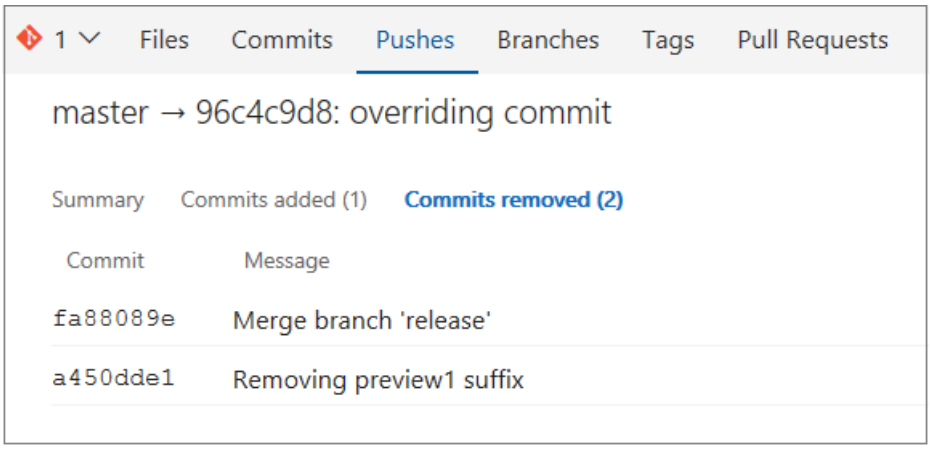
Searching for a file in commits or pull requests that have modified more than 1000 files was inefficient; you would need to click on **Load more** link several times to find the file that you were interested in. Now, when you filter content in the tree view, the search for that file is done across all files in the commit instead of just looking at the top 1000 files loaded. The performance of the commit details page is also improved when there are more than 1000 files modified.

Find lost commits due to a Force Push

You can perform a Git force push and update a remote ref even if it is not an ancestor of the local ref. This may cause others to lose commits and it can be very hard to identify the root cause. In the new pushes view, we have made force pushes noticeable in order to help troubleshoot issues related to missing commits.

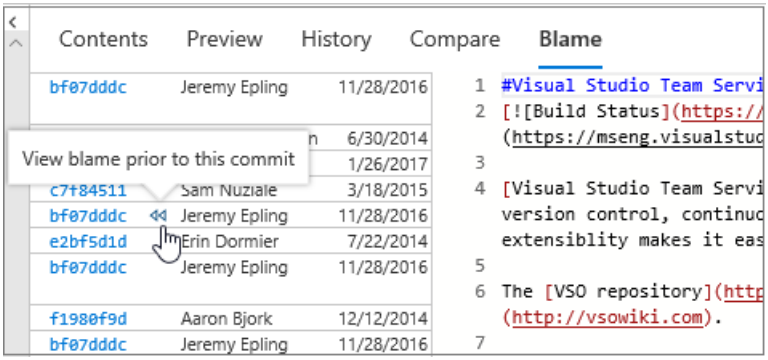


Clicking on the force push tag takes you to the removed commit.



Blame now has history

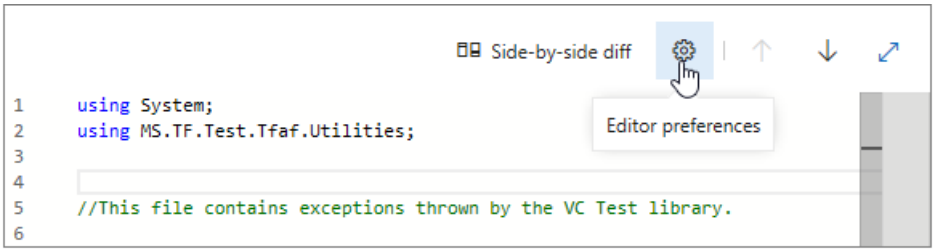
The **Blame** view is great for identifying the last person to change a line of code. However, sometimes you need to know who made the *previous* change to a line of code. The newest improvement in blame can help - **View blame prior to this commit**. As the name suggests, this feature allows you to jump back in time to the version of the file prior to the version that changed a particular line, and view the blame info for that version. You can continue to drill back in time looking at each version of the file that changed the selected line of code.



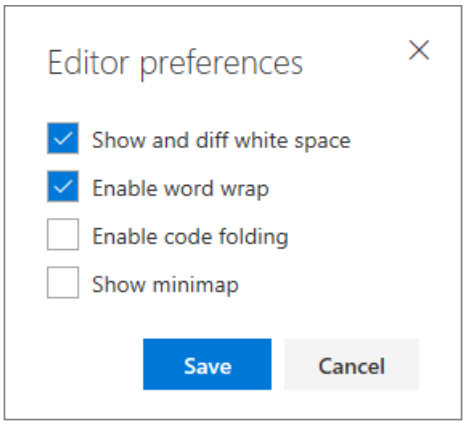
Toggle word wrap and white space in diff views

Two new features are available in the file diff viewer: **Toggle Word Wrap** and **Toggle White Space**. The first allows the word wrap setting to be applied while in a diff view. This is particularly useful for reviewing PRs that contain files without frequent line breaks - markdown files are a good example. The option to toggle white space is helpful when only whitespace has changed in a line or file. Toggling this setting displays and highlights the whitespace characters (dots for spaces, arrows for tabs, etc.) in the diff.

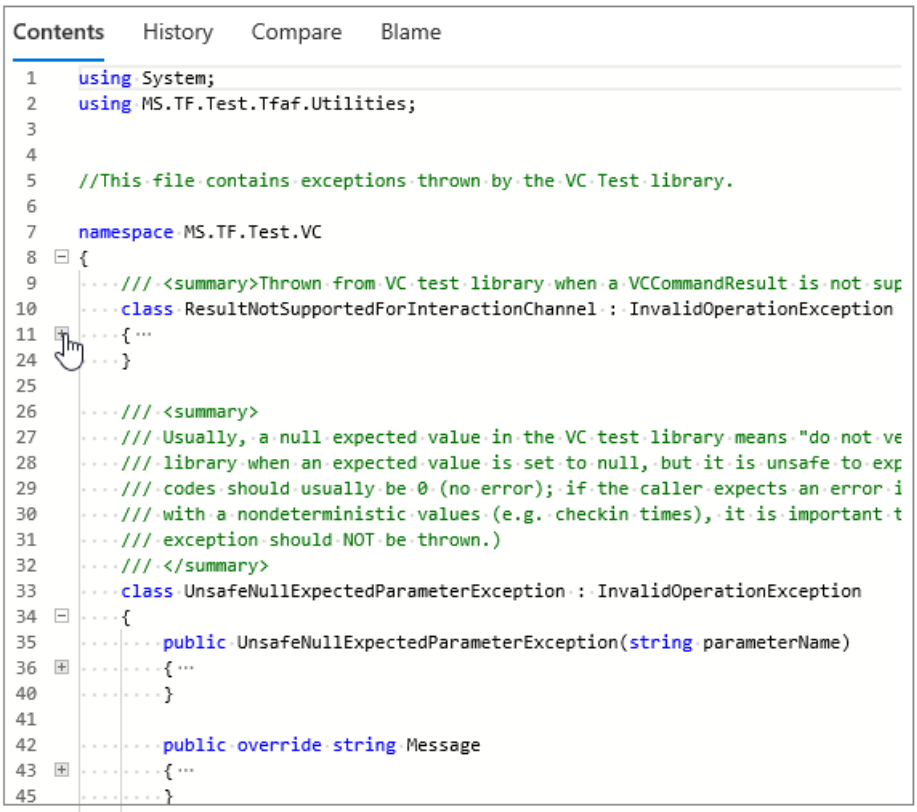
To manage these settings, click on the editor preferences gear in the pull request editor or diff view. In the **Files** view, select the **User Preferences** option on the right-click menu.



Select the various editor features including **Show and diff white space**, **Enable word wrap**, **Enable code folding**, and **Show minimap**.

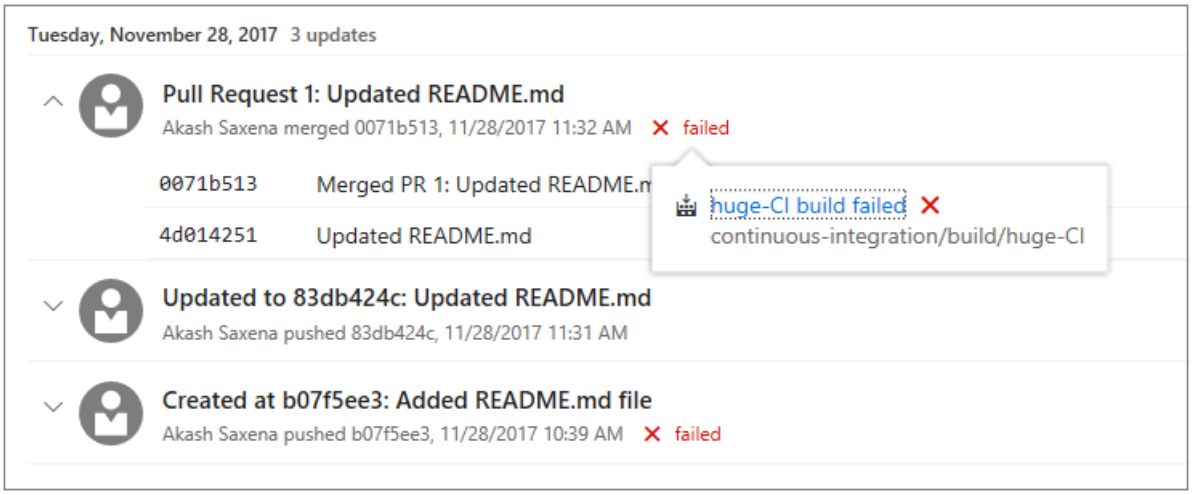


Code folding (called "outlining" in some editors) is also being enabled for the web view. When code folding is enabled, click on the minus signs to collapse sections of code -- click on plus signs to expand collapsed sections. The F1 command palette also exposes options for folding various indentation levels across an entire file, making it easier to read and review large files.



Track code pushes to a Git repo to builds and releases

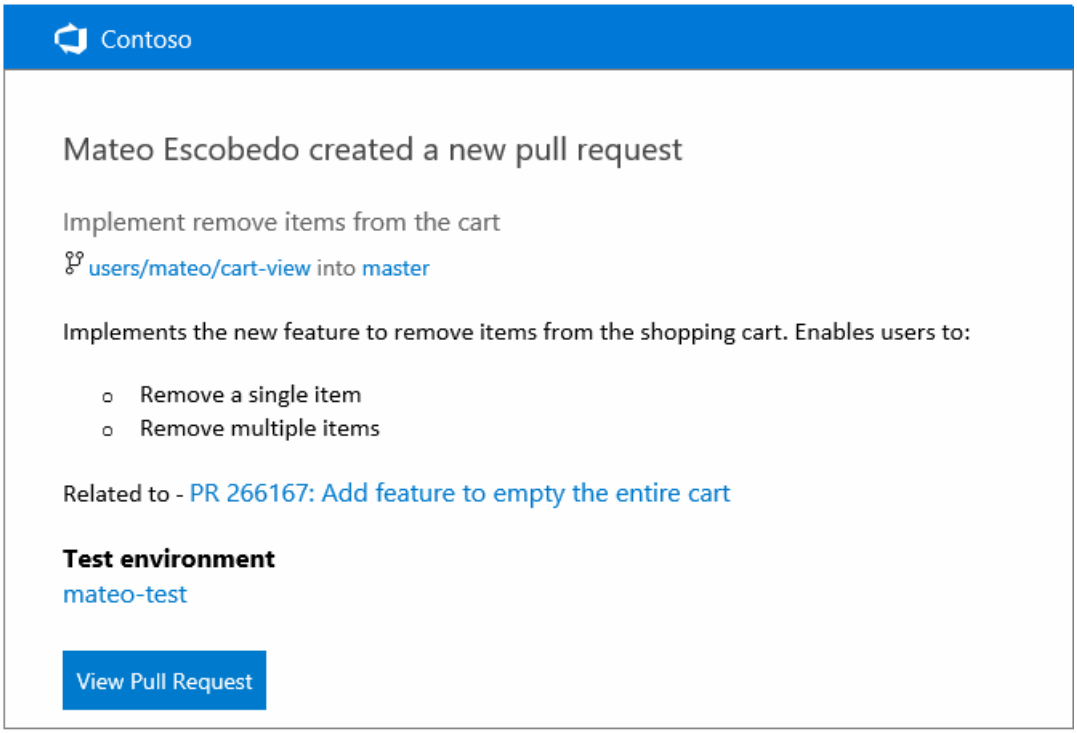
Now you can view the build and release status of merge commits in the **Pushes** page. By clicking the status next to the push, you will find the specific build or release that the push is included in so that you can verify success or investigate failure.



Rendered markdown in email notifications

Markdown is great for adding rich formatting, links, and images in pull request (PR) descriptions and comments. Email notifications for PRs now display the rendered markdown instead of the raw contents, which improves readability.

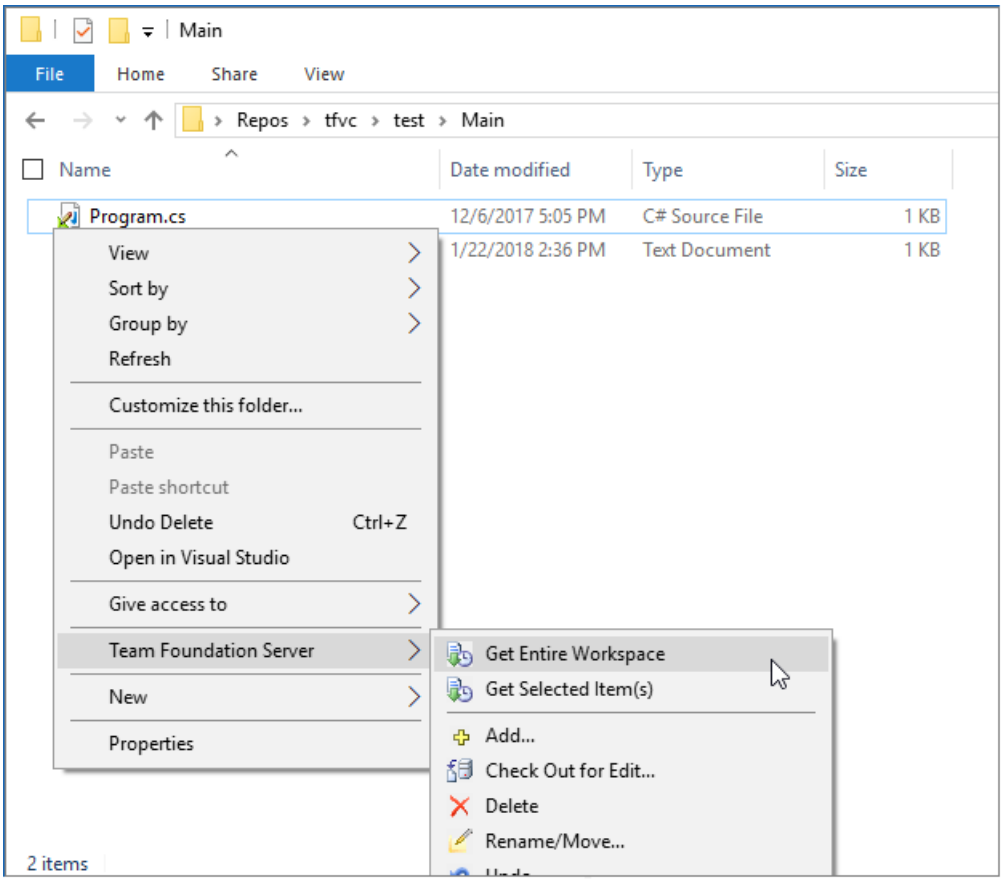
Inline images are not yet rendered inline (they are just shown as links), but we have that on our backlog to add in the future.



Perform TFVC commands right from Windows Explorer

The TFVC Windows Shell Extension, that gives a lightweight version control experience integrated into Windows File Explorer, now supports TFS 2018. This tool gives convenient access to many TFVC commands right in the Windows Explorer context menu.

Formerly part of the TFS Power tools, the tool has been released as a standalone tool on the Visual Studio Marketplace.



Control who can contribute to pull requests

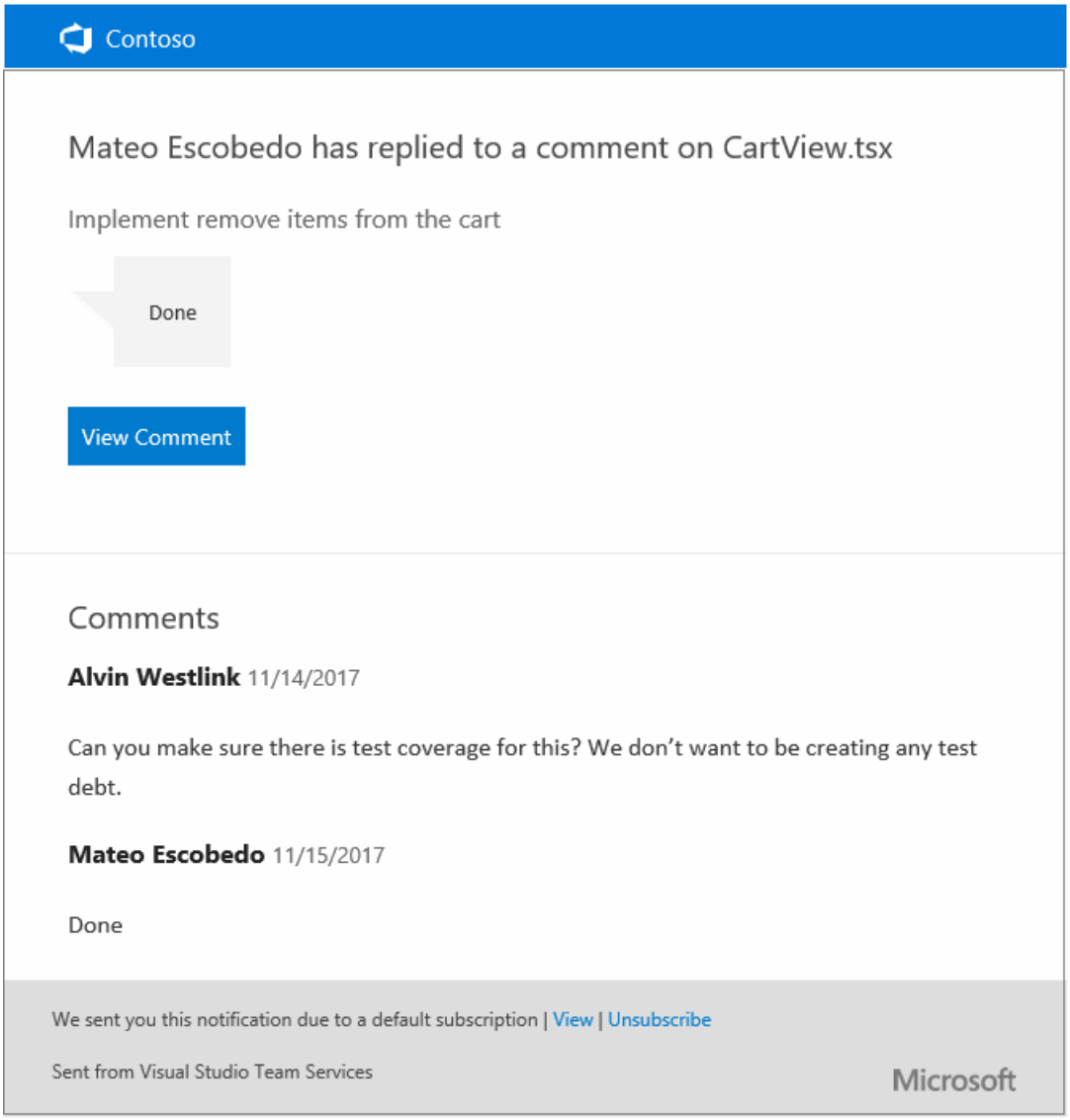
Previously, anyone who could view a Git repository could work with its pull requests. We have added a new permission called **Contribute to pull requests** that controls access to creating and commenting on pull requests. All users and groups that previously held the **Read** permission are also granted this new permission by default. The introduction of this new permission gives administrators additional flexibility and control. If you require your **Readers** group to be truly read-only, you can deny the **Contribute to pull requests** permission.

See the quickstart documentation for setting repository permissions for more information.

Pull request comment notifications include the thread context

Many times, replies to pull request (PR) comments are pretty brief, acknowledging that a change will be or has been made. This is not a problem when viewing these comments in the web view, but if you are reading a comment in an email notification, the context of the original comment is lost. A simple "I'll fix it" has no meaning.

Now, whenever a reply is made to a PR comment, the comment emails include the prior replies in the body of the email message. This allows the thread participants to see the full context of the comment right from their inbox - no need to open the web view.



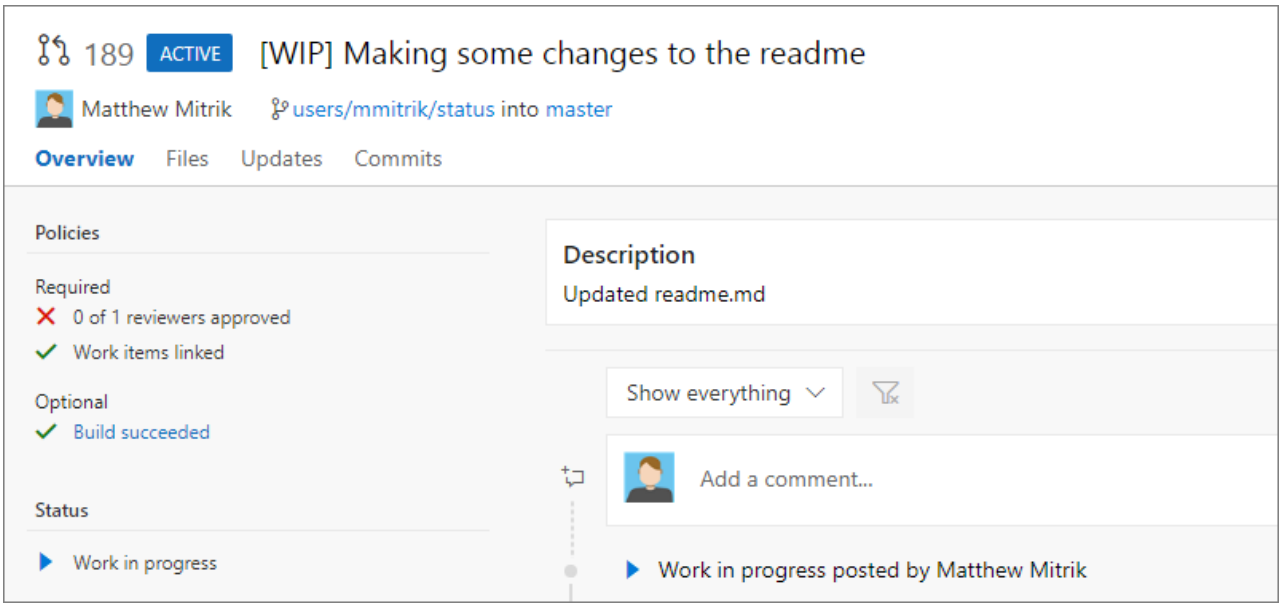
Complete work items settings

The feature to complete work items when completing pull requests now has a new repository setting to control the default behavior. The new setting to **Remember user preferences for completing work items with pull requests** is enabled by default, and honors the user's last state when completing future pull requests in the repo. If the new setting is disabled, then the **Complete linked work items after merging** option defaults to disabled for all pull requests in the repository. Users can still choose to transition linked work items when completing PRs, but they will need to opt-in each time.

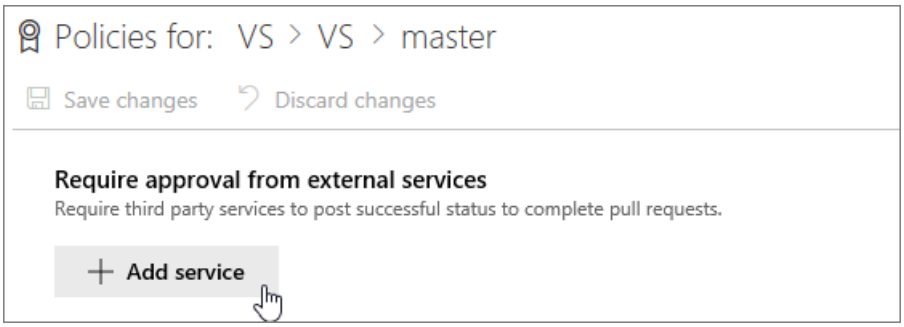
Pull request status extensibility

Using branch policies can be a great way to increase the quality of your code. However, those policies have been limited to only the integrations provided natively by TFS. Using the new pull request Status API and the corresponding branch policy, third party services can participate in the pull request workflow just like native TFS features.

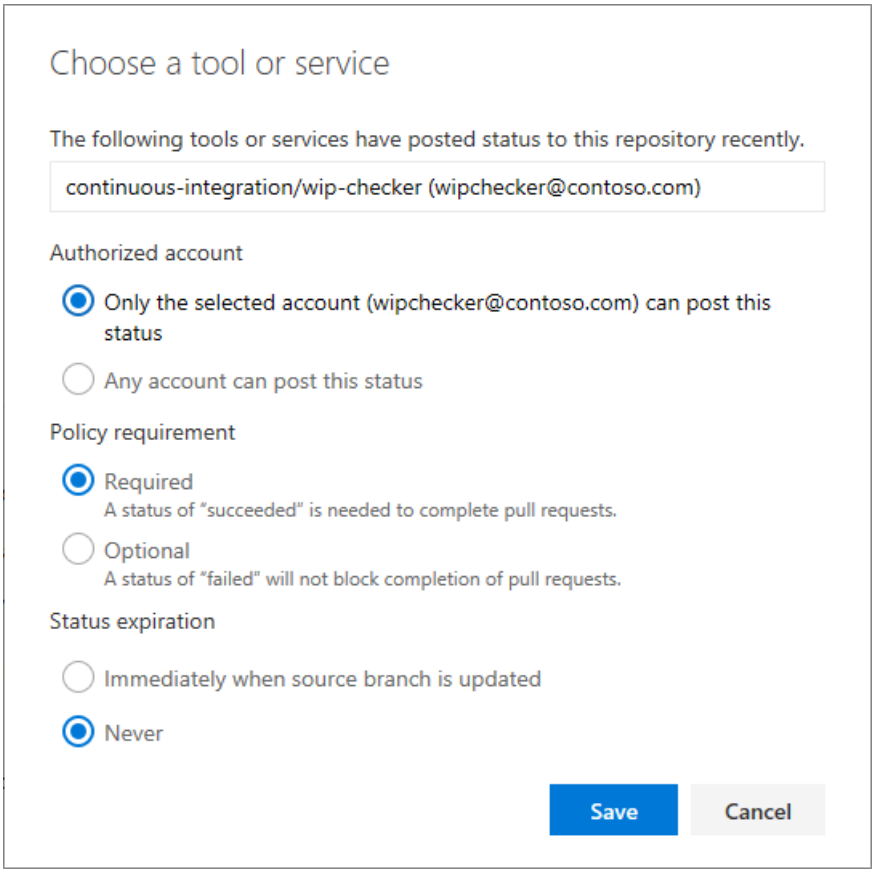
When a service posts to the Status API for a pull request, it immediately appears in the **PR details** view in a new **Status** section. The status section shows the description and creates a link to the URL provided by the service. Status entries also support an action menu (...) that is extensible for new actions added by web extensions.



Status alone does not block completion of a PR - that is where the policy comes in. Once PR status has been posted, a policy can then be configured. From the branch policies experience, a new policy is available to **Require approval from external services**. Select **+ Add service** to begin the process.



In the dialog, select the service that is posting the status from the list and select the desired policy options.

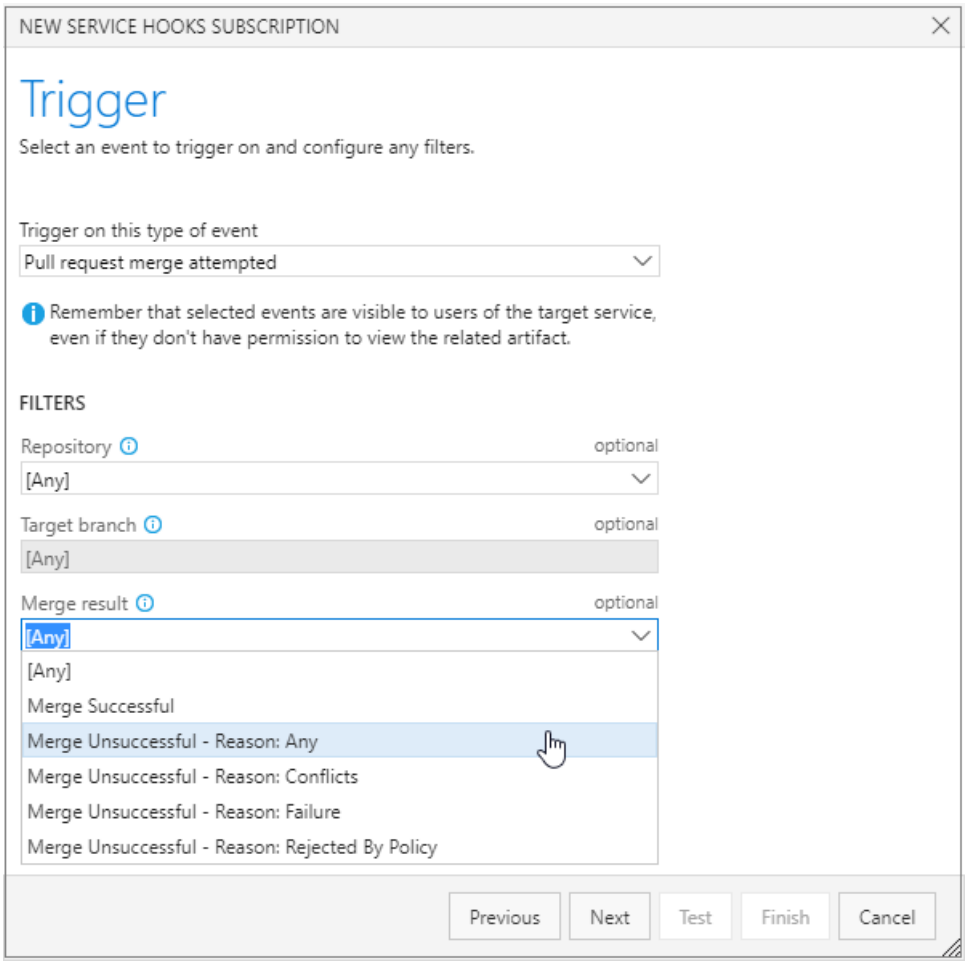


Once the policy is active, the status is shown in the **Policies** section, under **Required** or **Optional** as appropriate, and the PR completion is enforced as appropriate.

To learn more about the status API, and to try it out for yourself, check out the documentation and samples.

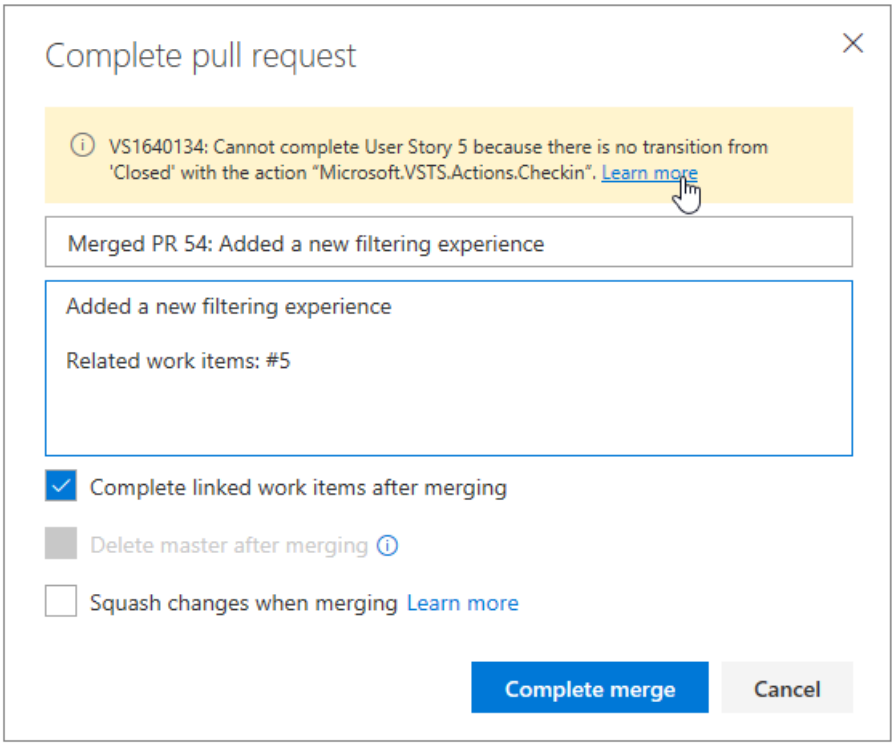
Pull request service hooks merge events

Extensions using pull request service hooks now have more details and filtering options for merge events. Any time a merge is attempted, the event is fired regardless of the success or failure of the merge. When a merge attempt results in a failure, details about the reason for the failure is included.



Improved error messages for work items completing with a pull request

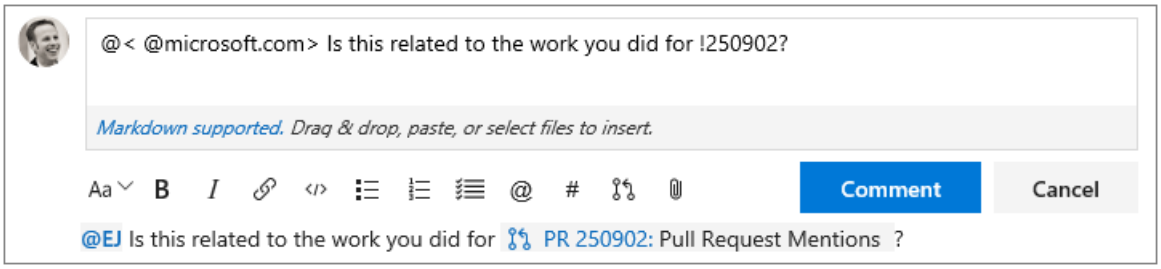
When attempting to complete work items with a pull request, it is possible that the associated work item cannot be transitioned to the completed state. For example, a specific field might be required and needs user input before the state can transition. We have improved the experience to inform you when something is blocking the work item transition, enabling you to take action to make the necessary changes.



Mention a pull request

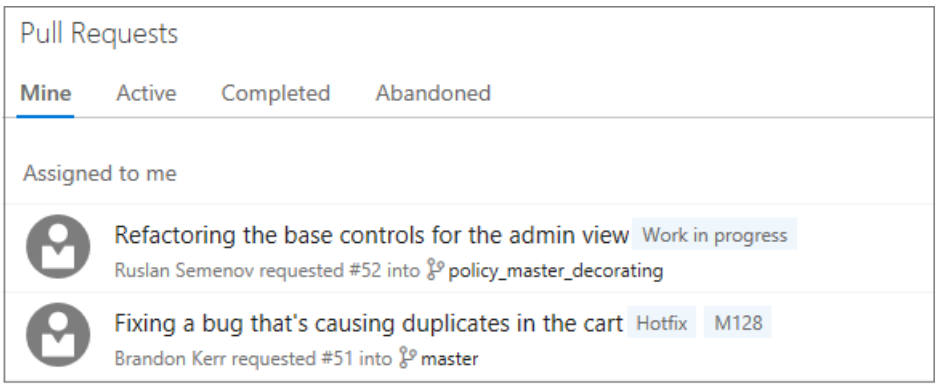
You can now mention pull requests in PR comments and work item discussions. The experience for mentioning a PR is similar to that of a work item, but uses an exclamation point `!` instead of a hash mark `#`.

Whenever you want to mention a PR, enter a `!`, and you will see an interactive experience for picking a PR from your list of recent PRs. Enter keywords to filter the list of suggestions, or enter the ID of the PR you want to mention. Once a PR is mentioned, it is rendered inline with the ID and the full title, and it will link to the PR details page.



Help reviewers using pull request labels

Sometimes it is important to communicate extra information about a pull request to the reviewers. Maybe the pull request is still a work in progress, or it is a hotfix for an upcoming release - so you append some extra text in the title, perhaps a "[WIP]" prefix or "DO NOT MERGE". Labels now provide a way to tag pull requests with extra information that can be used to communicate important details and help organize pull requests.

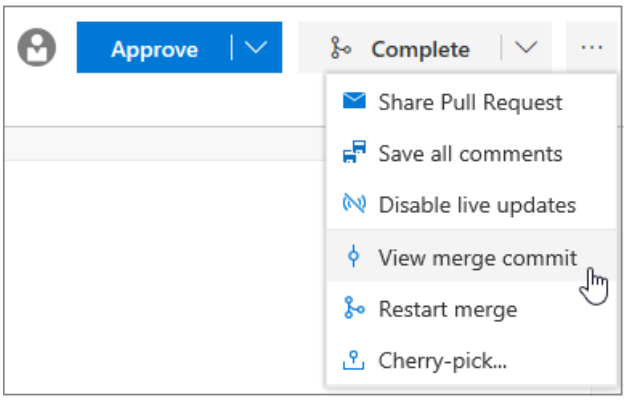


Pull request comments follow renamed files

Sometimes files are renamed or moved while a pull request is active. Previously, if there were comments on those renamed files, the latest view of the code would not display the comments. We have now improved comment tracking to follow the renames, displaying comments on the latest version of renamed or moved files.

View pull request merge commit

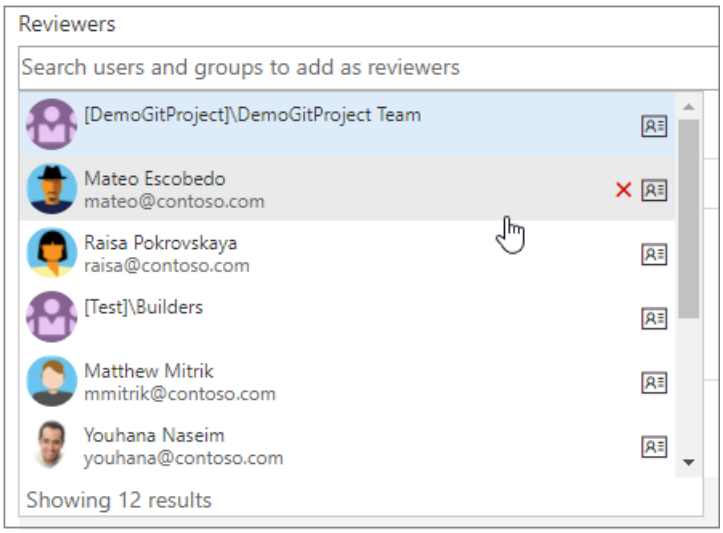
Pull request diff views are great at highlighting the changes introduced in the source branch. However, changes to the target branch may cause the diff view to look different than expected. A new command is now available to view the diff of the "preview" merge commit for the pull request - **View merge commit**. This merge commit is created to check for merge conflicts and to use with a pull request build, and it reflects what the merge commit will look like when the pull request is eventually completed. When the target branch has changes not reflected in the diff, the merge commit diff can be useful for seeing the latest changes in both the source and target branches.



Another command that is useful in conjunction with the **View merge commit** command is **Restart merge** (available on the same command menu). If the target branch has changed since the pull request was initially created, running this command creates a new preview merge commit, updating the merge commit diff view.

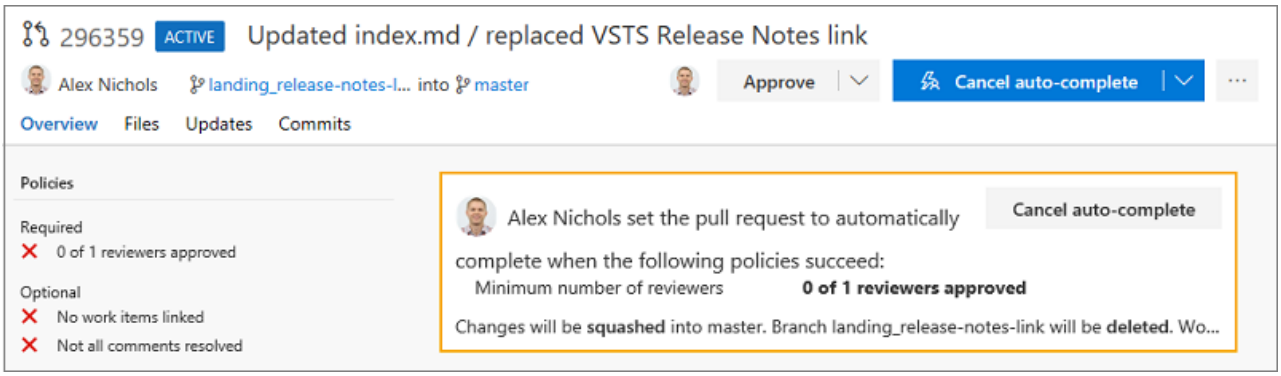
Recently used reviewers

If you frequently have your code reviewed by the same individuals, you will find it easier than ever to add reviewers. When adding reviewers to your pull requests, a list of your recently added reviewers is automatically displayed when you put focus into the reviewers input box -- no need to search by name. Select them as you would any reviewer.



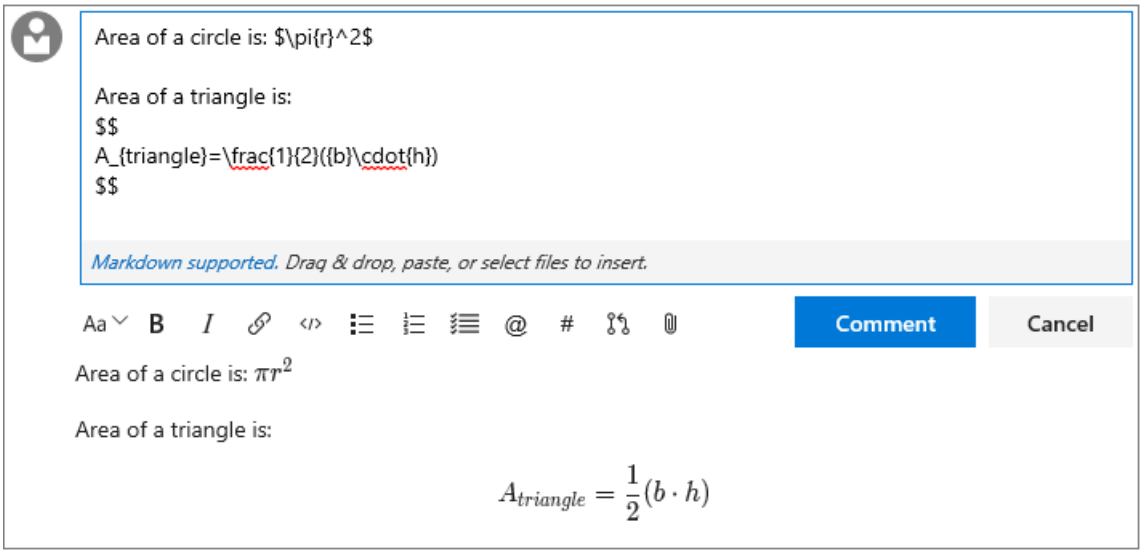
View remaining policy criteria for pull request auto-complete

Auto-complete is a useful feature for teams using branch policies, but when using optional policies, it can be unclear exactly what is blocking a pull request from being completed. Now, when setting auto-complete for a pull request, the exact list of policy criteria that are holding up completion are clearly listed in the callout box. As each requirement is met, items are removed from the list until there are no remaining requirements and the pull request is merged.



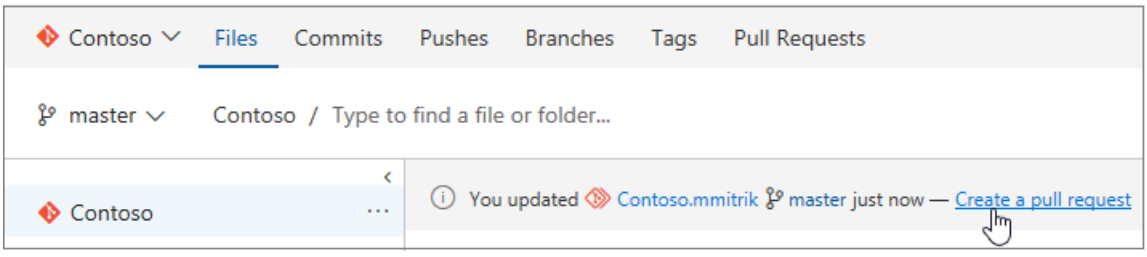
Discuss math in pull requests

Need to include an equation or mathematical expression in your pull request comments? You can now include KaTeX functions in your comments, using both inline and block commenting. See the list of supported functions for more information.



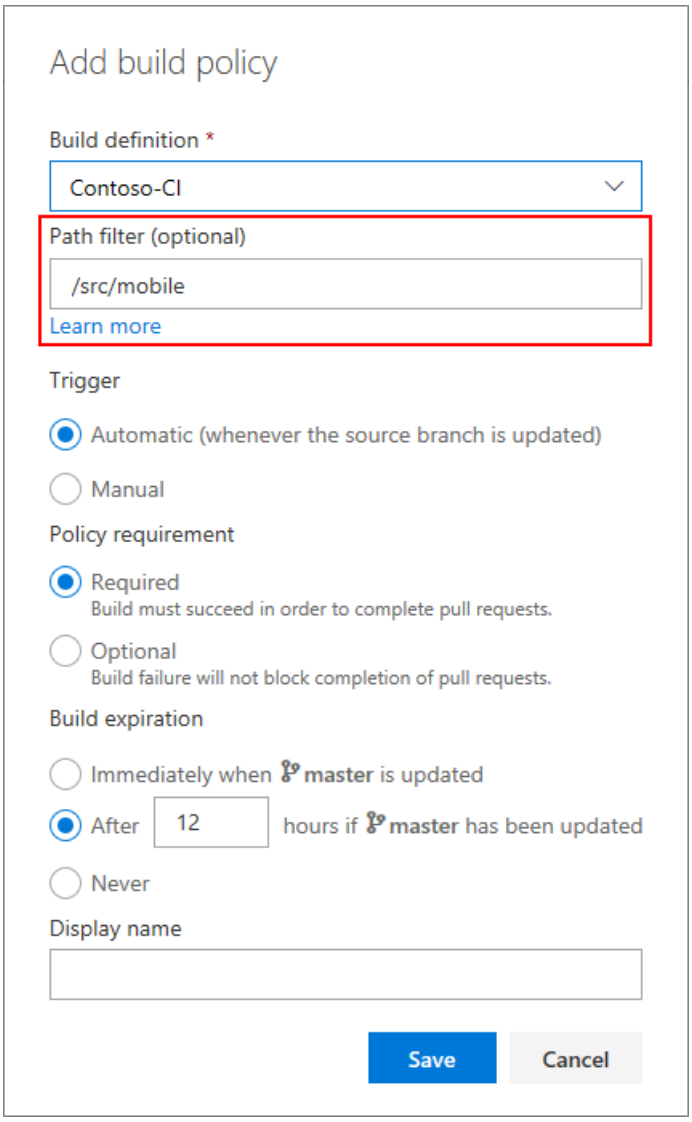
Pull request suggestions for forks

Whenever a topic branch is updated in a repository, a "suggestion" to create a new pull request (PR) for the topic branch is shown. This is very useful for creating new PRs, and we have enabled it for those working in a forked repo, too. If you update a branch in a fork, the next time you visit the **Code** hub for either the fork or the upstream repo, you will see the suggestion to create a pull request. If you select the "Create a pull request" link, you will be directed to the create PR experience, with the source and target branches and repos pre-selected.



Path filters for pull request policies

Many times, a single repository contains code that is built by multiple continuous integration (CI) pipelines to validate the build and run tests. The integrated build policy now supports a path filtering option that makes it easy to configure multiple PR builds that can be required and automatically triggered for each PR. Just specify a path for each build to require, and set, the trigger and requirement options as desired.



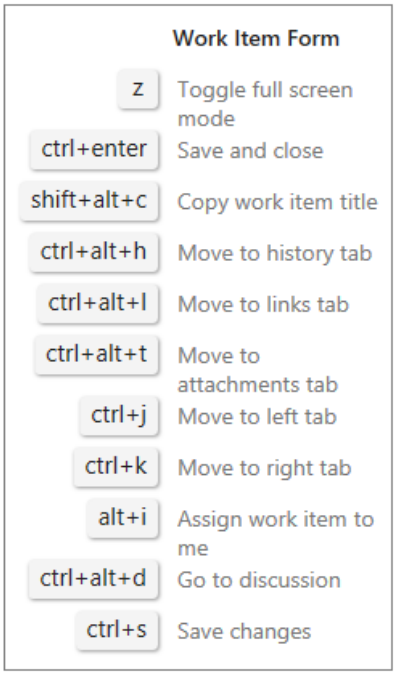
In addition to build, status policies also have the path filtering option available. This allows any custom or third party policies to configure policy enforcement for specific paths.

This feature was prioritized based on a suggestion.

Work

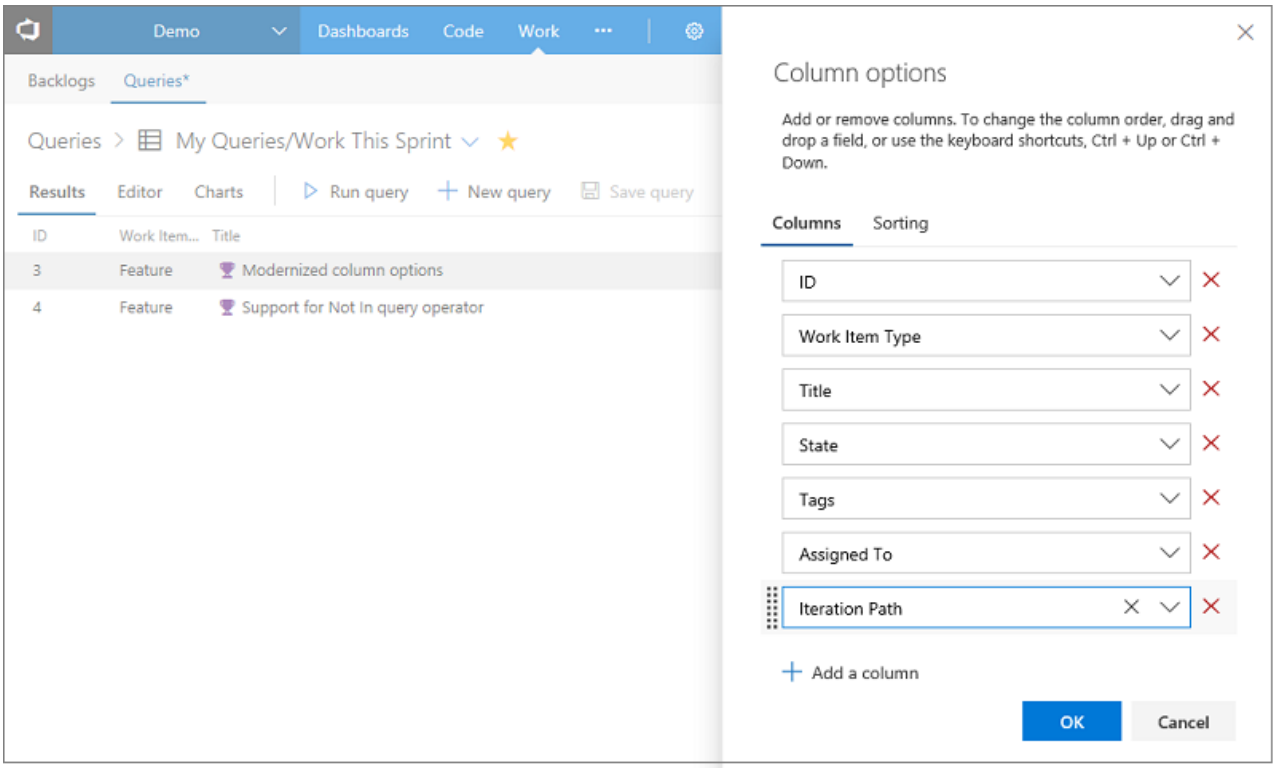
Keyboard shortcuts in the work item form

Assign a work item to yourself (Alt + i), jump to discussion (Ctrl + Alt + d), and copy a quick link to the work item (Shift + Alt + c) using keyboard shortcuts. For the full list of new shortcuts, type "?" with a work item form open or see the table below.



Modernized column options

The **Column options** dialog used to configure the columns of the work item grid in the **Backlog**, **Queries**, and **Test** hubs has been updated to use a new panel design. Search to find a field, drag and drop to reorder columns, or remove existing columns you no longer want.

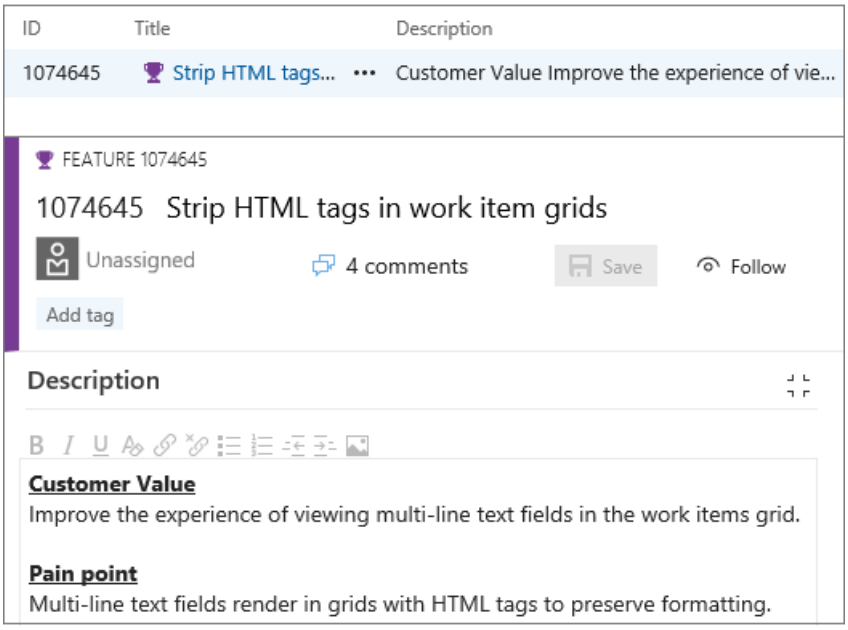


Query last run by information

As your project's **Shared Queries** tree grows, it can be difficult to determine if a query is no longer used and can be deleted. To help you manage your **Shared Queries**, we have added two new pieces of metadata to our query REST APIs, last executed by and last executed date, so that you can write clean-up scripts to delete stale queries.

HTML tags stripped in work item grids

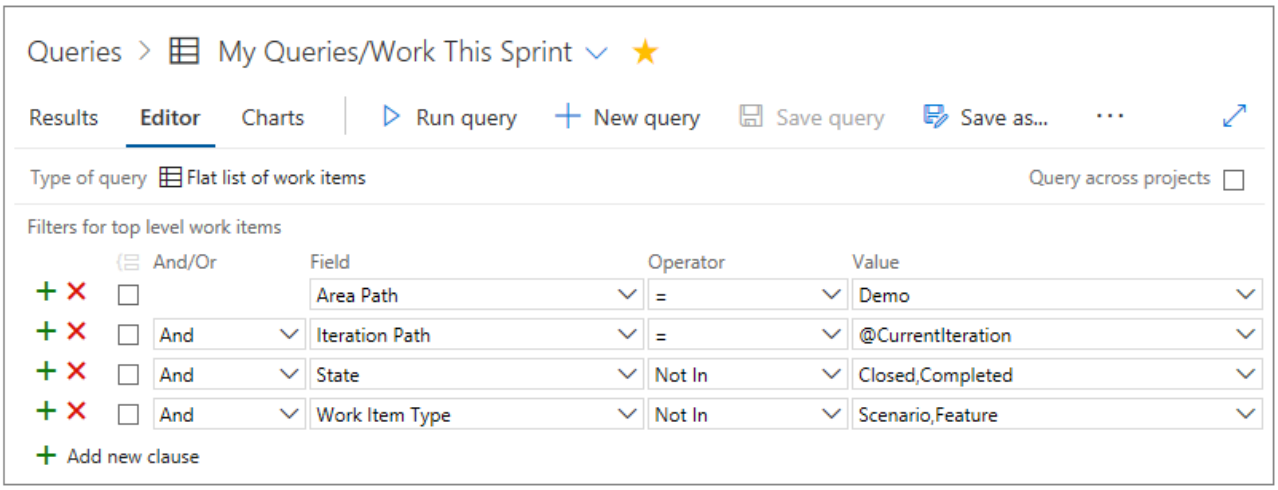
Based on customer feedback, we have updated the behavior of multi-line text fields in work item query results views in the web, Excel, and Visual Studio IDE to remove HTML formatting. When added as a column to the query, multi-line text fields now display as plain text. Here is an example of a feature with HTML in the description.



In the past, the query results would have rendered something like `<div><b><u>Customer Value</u>...`

Added support for Not In query operator

Fields that support the "In" query operator now support "Not In". Write queries for work items "Not In" a list of IDs, "Not In" a list of states, and much more, all without having to create many nested "Or" clauses. This feature was prioritized based on a customer suggestion. Keep submitting those ideas and voting up those most important to you.

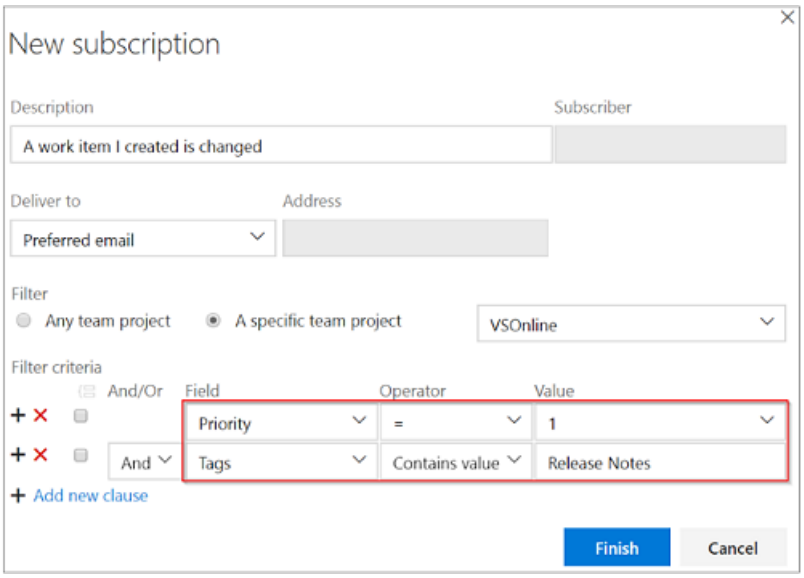


Query for @MyRecentActivity and @RecentMentions

We have introduced two new query macros for the **ID** field to help you find work items that may be important to you. See what items you were mentioned in over the last 30 days using **@RecentMentions** or take a look at work items you have recently viewed or edited using **@MyRecentActivity**.

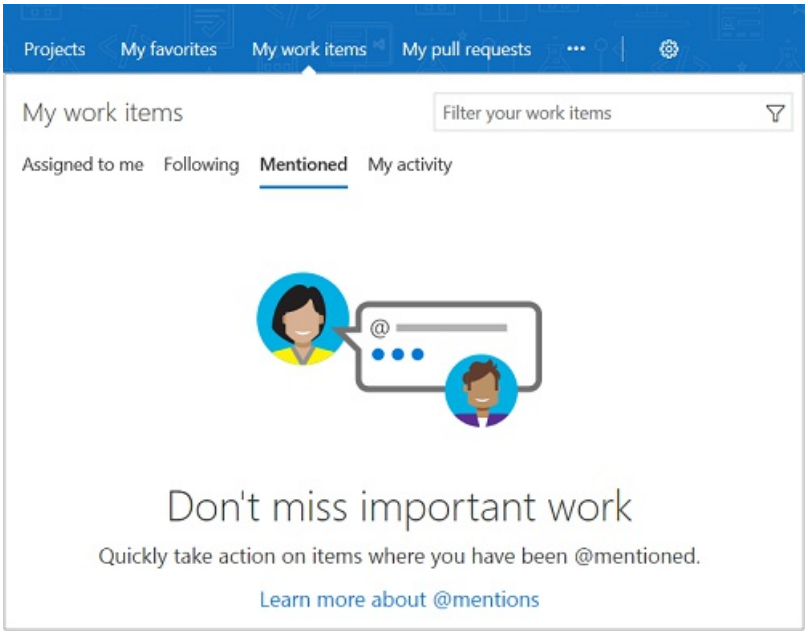
Custom fields and tags filter in work item tracking notifications

Notifications can now be defined using conditions on custom fields and tags; not only when they change but when certain values are met. This has been a top customer suggestion in UserVoice (see 6059328 and 2436843), and allows for a more robust set of notifications that can be set for work items.

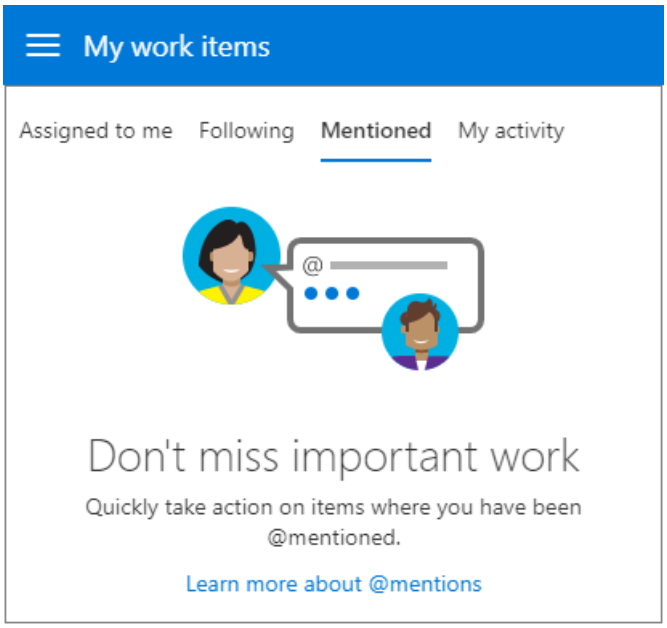


Mentioned support for the My work items page

We have added a new **Mentioned** pivot under the **My work items** page. Inside this pivot, you can review the work items where you have been mentioned in the last 30 days. With this new view, you can quickly take action on items that require your input and stay up to date on conversations that are relevant to you.

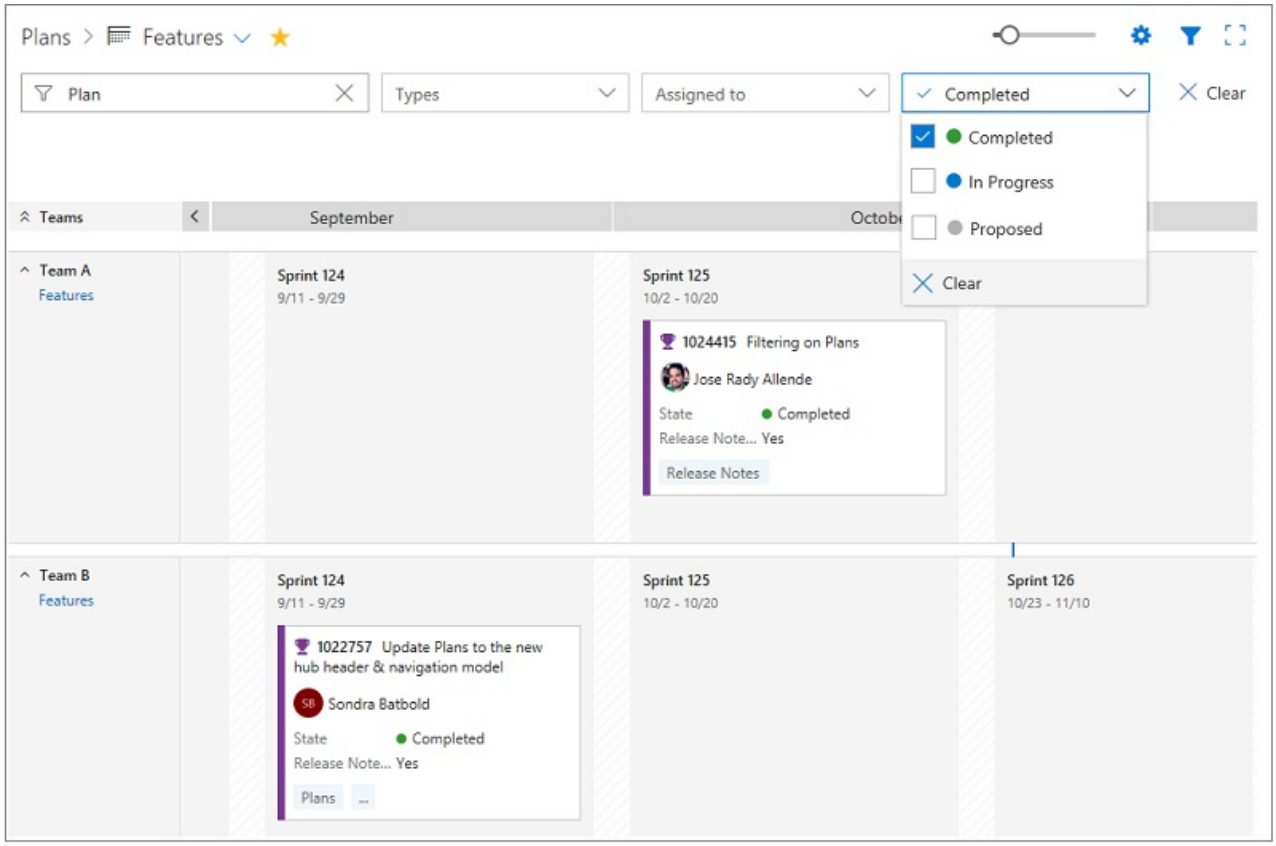


This same pivot is also available through our mobile experience, bringing consistency between both mobile and desktop.



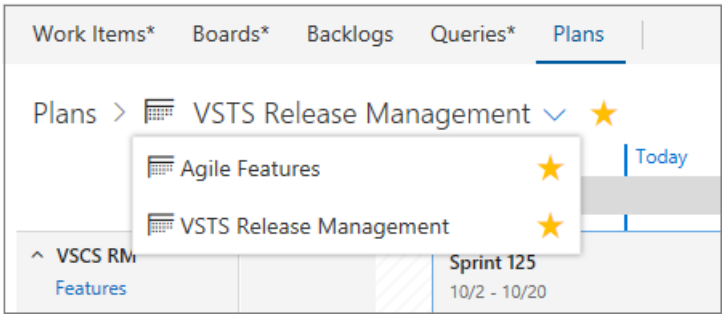
Filtering on plans

The **Delivery Plans** extension now makes use of our common filtering component, and is consistent with our grid filtering experiences for work items and **Boards**. This filtering control brings improved usability and a consistent interface to all members of your team.



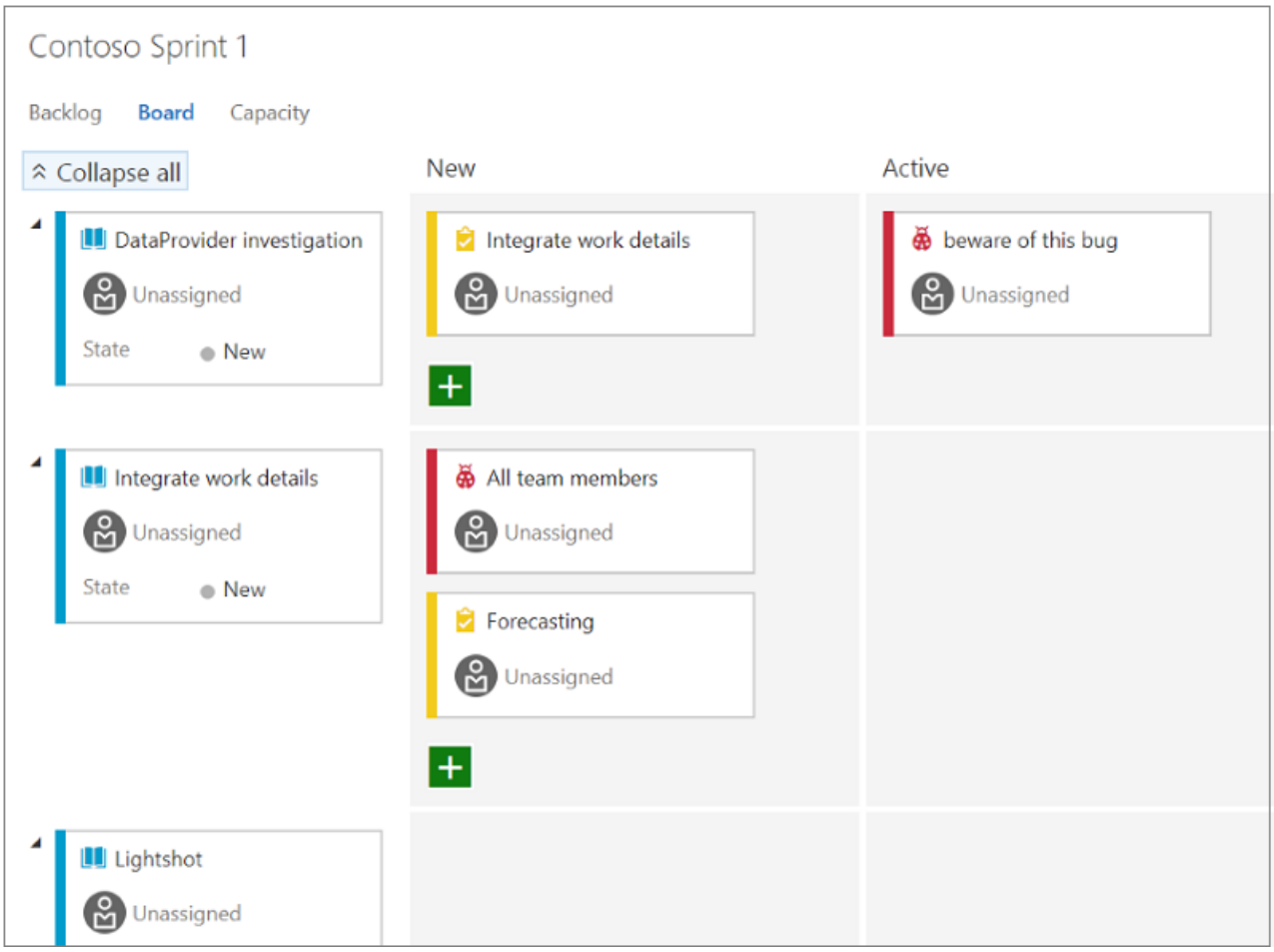
Updated plans navigation

Many of you care about a specific plan or set of plans and use favorites for quick access to the content. First, we have updated the **Plans** hub to navigate to your most recently visited plan instead of the directory page. Second, once there, you can use the favorites picker to quickly switch to another plan or use the breadcrumb to navigate back to the directory page.



Expand/collapse requirements/people on the task board

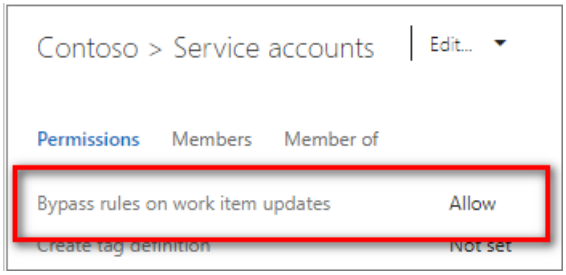
You can now expand or collapse all the items on the sprint **Task board** with just a single click.



Grant the bypassrule permission to specific users

Often, when migrating work items from another source, organizations want to retain all the original properties of the work item. For example, you may want to create a bug that retains the original created date and created by values from the system where it originated.

The API to update a work item has a bypassrule flag to enable that scenario. Previously, the identity who made that API request had to be a member of the Project Collection Administrators group. We have added a permission at the project level to execute the API with the bypassrule flag.



Build and Release

XAML builds

In TFS 2015, we introduced a web-based, cross-platform build system. XAML builds are not supported in TFS 2018 RTW or Update 1, but we have re-enabled XAML builds in TFS 2018 Update 2. We encourage you to migrate your XAML builds.

When you upgrade to TFS 2018 Update 2:

- If you have any XAML build data in your team project collection, you will get a warning about the deprecation of XAML build features.
- You will need to use VS or Team Explorer 2017 to edit XAML build definitions or to queue new XAML builds.
- If you need to create new XAML build agents, you will need to install them using the TFS 2015 build agent installer.

For an explanation of our XAML build deprecation plan, see the Evolving TFS/Team Services build automation capabilities blog post.

Enhancements to multi-phase builds

You have been able to use phases to organize your build steps and to target different agents using different demands for each phase. We have added several capabilities to build phases so that you can now:

- Specify a different agent queue for each phase. This means you can, for example:
  - Run one phase of a build on a macOS agent and another phase on a Windows agent. To see a cool example of how useful this can be, see this Connect(); 2017 video: CI/CD DevOps Pipeline for mobile apps and services.
  - Run build steps on a build agent pool and test steps on a test agent pool.
- Run tests faster by running them in parallel. Any phase that has parallelism configured as "Multi-agent" and contains a "VSTest" task now automatically parallelize test execution across the configured agent count.
- Permit or deny scripts to access the OAuth token each phase. This means, for example, you can now allow scripts running in your build phase to communicate with VSTS over REST APIs, and in the same build definition block the scripts running in your test phase.
- Run a phase only under specific conditions. For example, you can configure a phase to run only when previous phases succeed, or only when you are building code in the master branch.

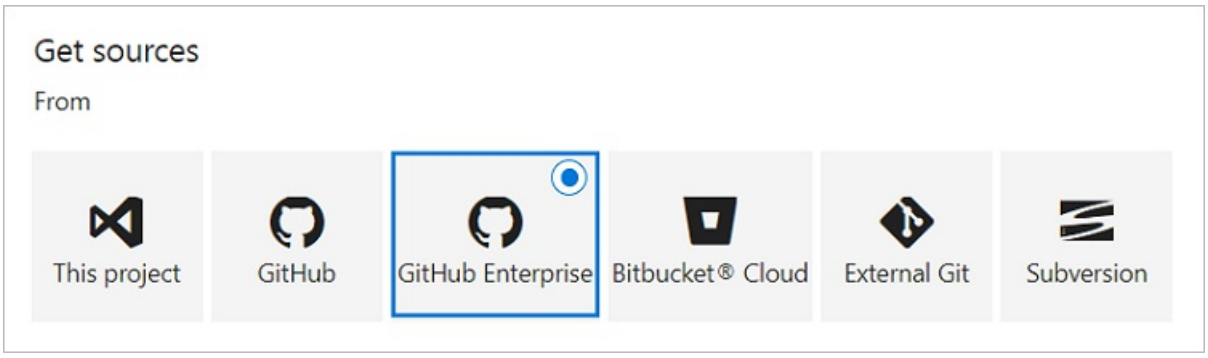
To learn more, see Phases in Build and Release Management.

Skip scheduled builds if nothing has changed in the repo

By popular request on UserVoice, you can now specify that a scheduled build not run when nothing has changed in your code. You can control this behavior using an option on the schedule. By default, we will not schedule a new build if your last scheduled build (from the same schedule) has passed and no further changes have been checked in to your repo.

Build with continuous integration from GitHub Enterprise

You now have better integration for performing continuous integration (CI) builds if you use **GitHub Enterprise** for version control. Previously, you were limited to polling for code changes using the **External Git** connector, which may have increased the load on your servers and caused delays before builds were triggered. Now, with official **GitHub Enterprise** support, team CI builds are immediately triggered. In addition, the connection can be configured using various authentication methods, such as LDAP or built-in accounts.

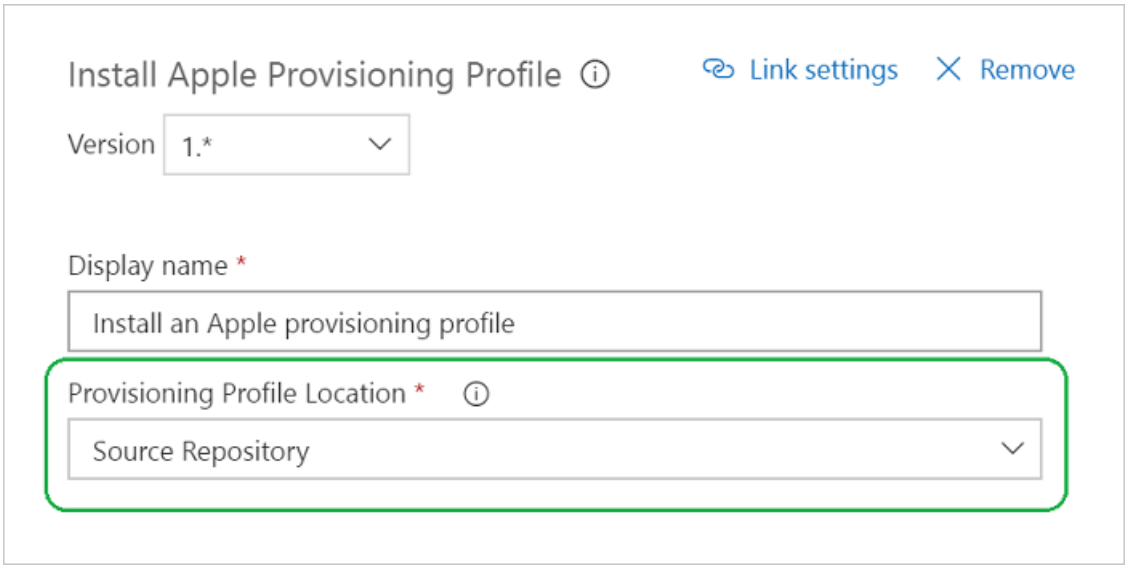


Secure files can be downloaded to agents during build or release

The new **Download Secure File** task supports downloading (to agent machines) encrypted files from the **VSTS Secure Files** library. As the file is downloaded, it is decrypted and stored on the agent's disk. When the build or release completes, the file is deleted from the agent. This allows your build or release to use sensitive files, such as certificates or private keys that are otherwise securely encrypted and stored in VSTS. For more information, see Secure files documentation.

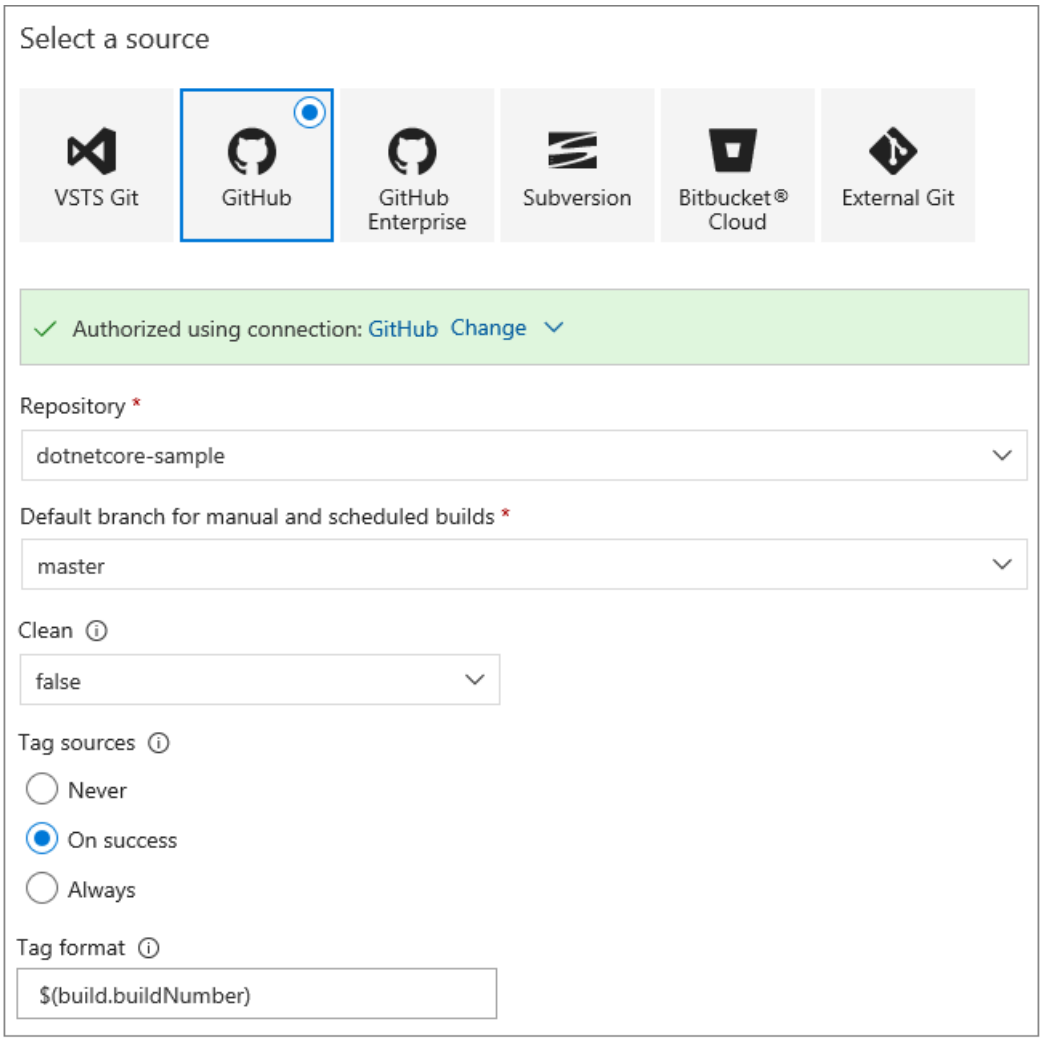
Apple provisioning profiles can be installed from source repositories

The **Install Apple Provisioning Profile** task already supports installing (on agent machines) provisioning profiles that are stored in the **VSTS Secure Files** library. Provisioning profiles are used by Xcode to sign and package Apple apps, such as for iOS, macOS, tvOS, and watchOS. Now, provisioning profiles can be installed from source code repositories. Though use of the Secure Files library is recommended for greater security of these files, this improvement addresses provisioning profiles already stored in source control.

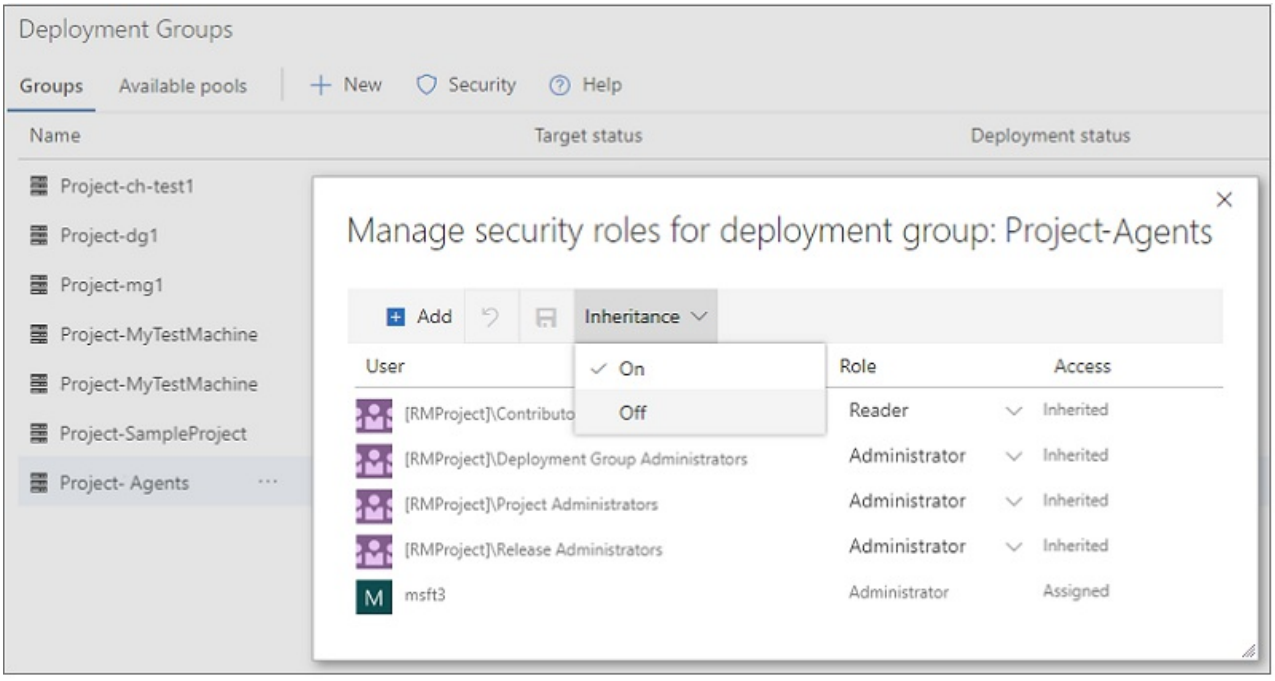


Trace GitHub sources to builds using build tags

Builds from GitHub or GitHub Enterprise already link to the relevant commit. It is equally important to be able to trace a commit to the builds that built it. That is now possible by enabling source tagging in TFS. While choosing your GitHub repository in a build definition, select the types of builds you want to tag, along with the tag format.



Then watch build tags appear on your GitHub or GitHub Enterprise repository.



Specific Java Development Kits (JDKs) can be installed during builds and releases

For building certain Java projects, specific JDKs may be required but unavailable on agent machines. For example, projects may require older or different versions of IBM, Oracle, or open-source JDKs. The **Java Tool Installer** task downloads and installs the JDK needed by your project during a build or release. The JAVA_HOME environment variable is set accordingly for the duration of the build or release. Specific JDKs are available to the **Java Tool Installer** using a file share, a source code repository, or Azure Blob Storage.

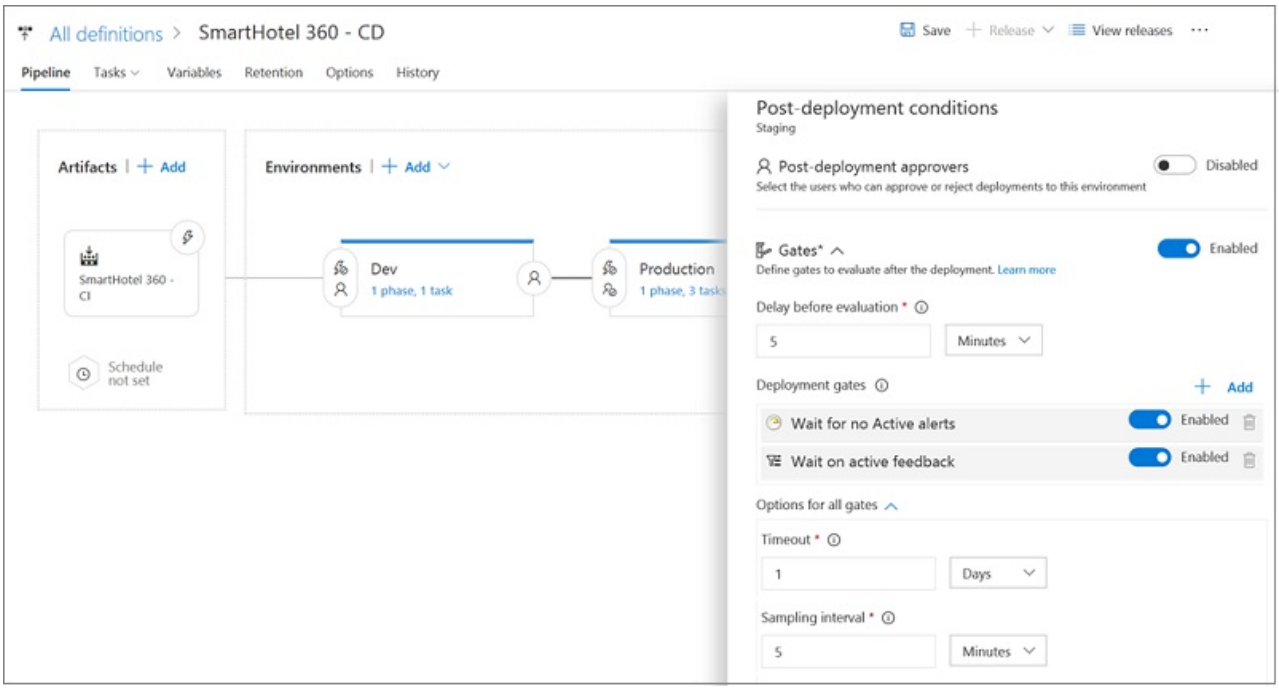
Improved Xcode build configuration

The Xcode task has been updated with a new major version (4.*) that improves configuration of Xcode building, testing, and packaging. If your Xcode project has a single, shared scheme, it is automatically used. Additional inline help was added. Deprecated features, such as xcrun packaging, were removed from the Xcode task's properties. Existing build and release definitions must be modified to use this latest 4.* version of the Xcode task. For new definitions, if you need a previous Xcode task version's deprecated capabilities, you can select that version in your definition.

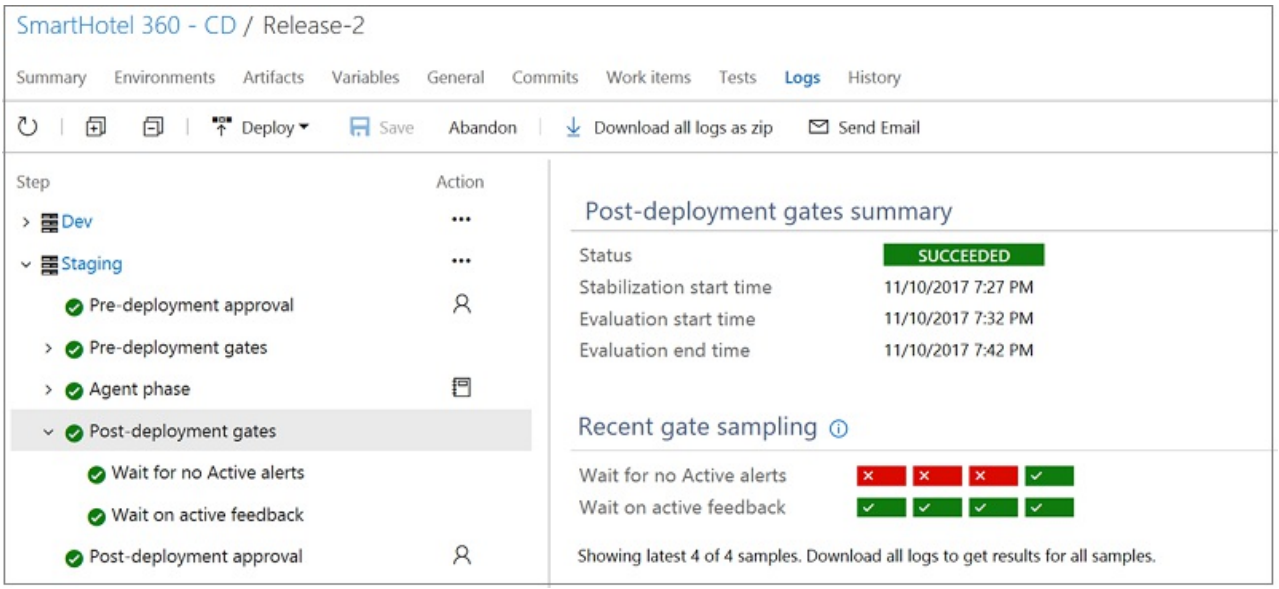
Release gates

Continuous monitoring is an integral part of DevOps pipelines. Ensuring the app in a release is healthy after deployment is as critical as the success of the deployment process. Enterprises have adopted various tools for automatic detection of app health in production and for keeping track of customer reported incidents. Until now, approvers had to manually monitor the health of the apps from all the systems before promoting the release. However, Release Management now supports integrating continuous monitoring into release pipelines. Use this to ensure the system repeatedly queries all the health signals for the app until all of them are successful at the same time, before continuing the release.

You start by defining pre-deployment or post-deployment gates in the release definition. Each gate can monitor one or more health signals corresponding to a monitoring system of the app. Built-in gates are available for "Azure monitor (application insight) alerts" and "Work items". You can integrate with other systems using the flexibility offered through Azure functions.



At the time of execution, the **Release** starts to sample all the gates and collect health signals from each of them. It repeats the sampling at each interval until signals collected from all the gates in the same interval are successful.



Initial samples from the monitoring systems may not be accurate, as not enough information may be available for the new deployment. The "Delay before evaluation" option ensures the **Release** does not progress during this period, even if all samples are successful.

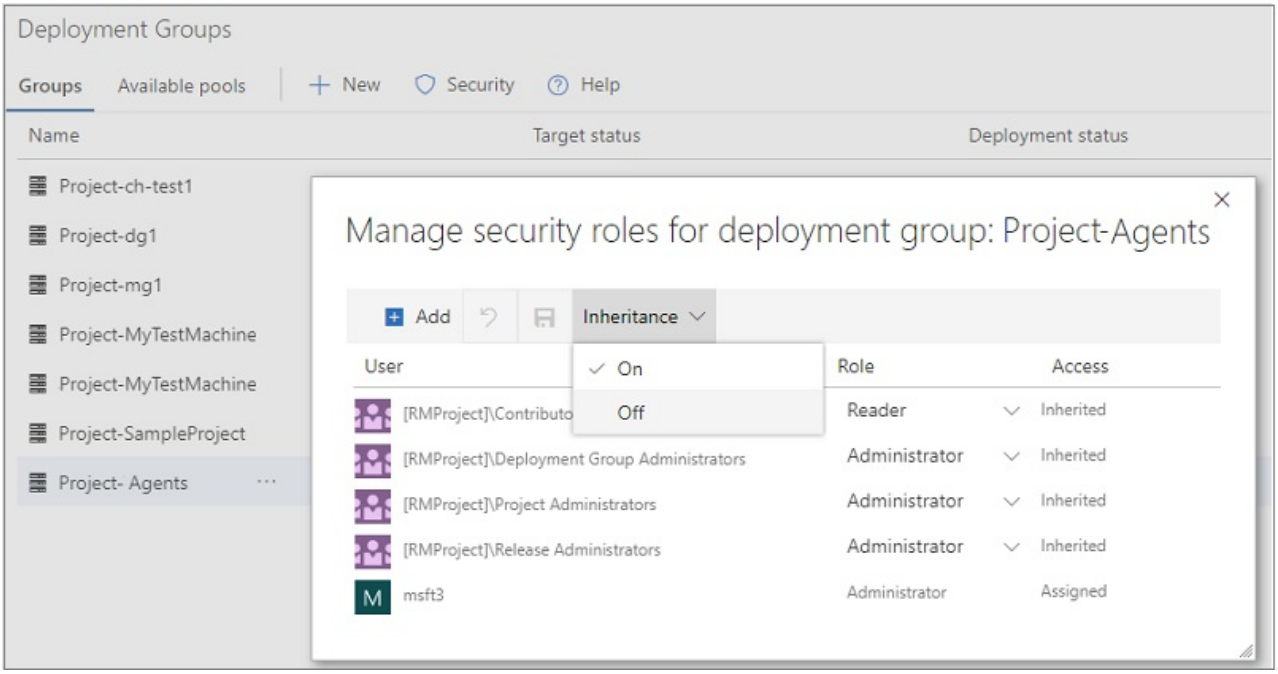
No agents or pipelines are consumed during sampling of gates. See the documentation for release gates for more information.

Deploy selectively based on the artifact triggering a release

Multiple artifact sources can be added to a release definition and configured to trigger a release. A new release is created when a new build is available for either of the sources. The same deployment process is executed regardless of what source triggered the release. You can now customize the deployment process based on the triggering source. For auto-triggered releases, the release variable **Release.TriggeringArtifact.Alias** is now populated to identify the artifact source that triggered the release. This can be used in task conditions, phase conditions, and task parameters to dynamically adjust the process. For example, if you only need to deploy the artifacts that changed through environments.

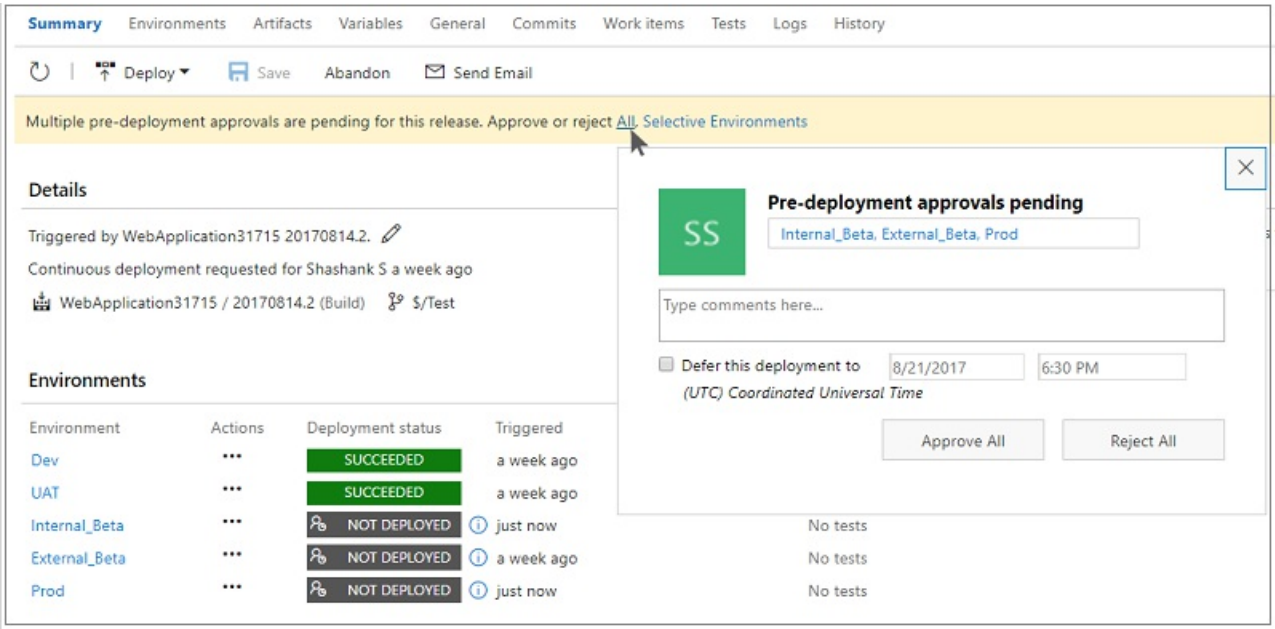
Manage entity-specific security

Previously in role based security, when the security access roles were set, they were set for a user or group at hub level for Deployment groups, Variable groups, Agent queues, and Service endpoints. Now you can turn on and off inheritance for a particular entity so you can configure security just the way you want to.



Approve multiple environments

Managing approvals with releases is now simpler. For pipelines having the same approver for multiple environments that deploy in parallel, the approver currently needs to act on each of the approvals separately. With this feature, you can now complete multiple pending approvals at the same time.



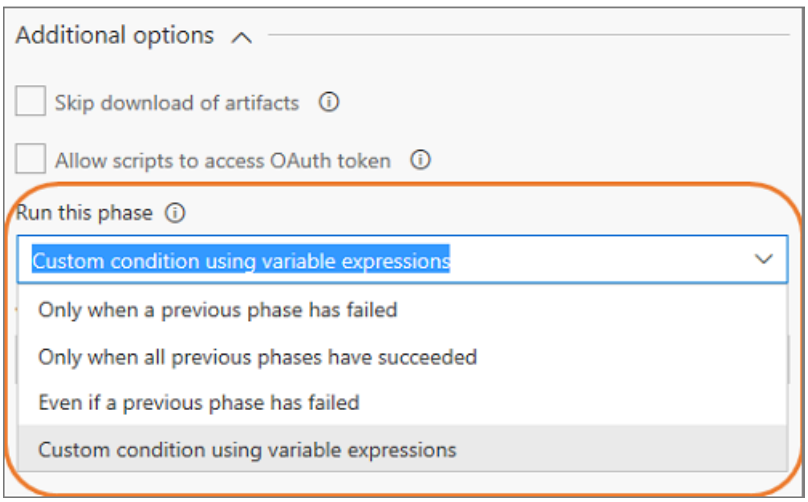
Release template extensibility

Release templates let you create a baseline for you to get started when defining a release process. Previously, you could upload new ones to your account, but now authors can include release templates in their extensions. You can find an example on the GitHub repo.

Conditional release tasks and phases

Similar to conditional build tasks, you can now run a task or phase only if specific conditions are met. This will help you in modeling rollback scenarios.

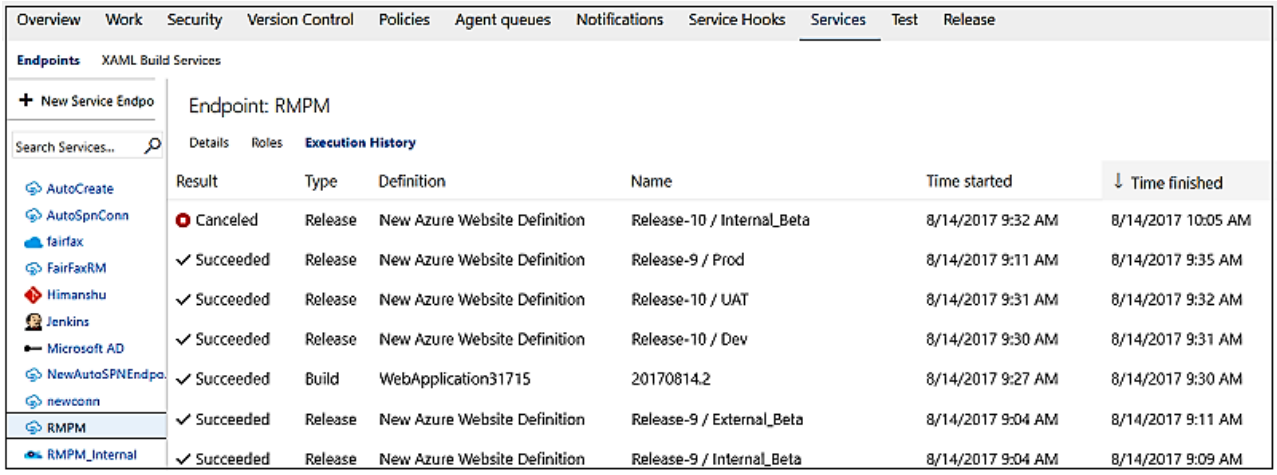
If the built-in conditions do not meet your needs, or if you need more fine-grained control over when the task or phase runs, you can specify custom conditions. Express the condition as a nested set of functions. The agent evaluates the innermost function and works its way outward. The final result is a Boolean value that determines if the task is to be run.



Requests history for service endpoints

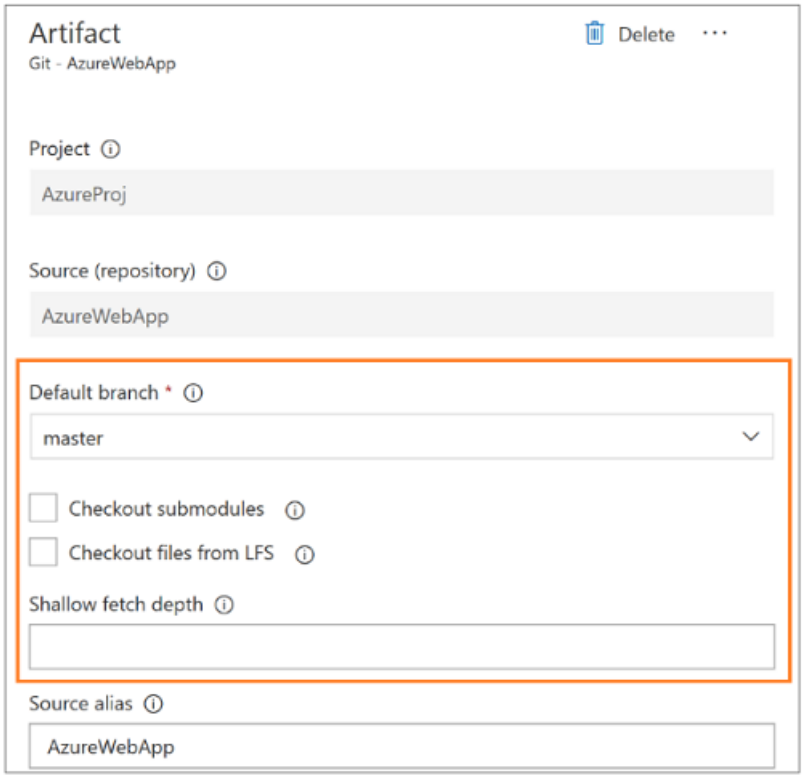
Service endpoints enable connection to external and remote services to execute tasks for a build or deployment. The endpoints are configured in project scope and shared between multiple build and release definitions. Service endpoint owners can now get a consolidated view of builds and deployments using an endpoint, that can help to

improve auditing and governance.



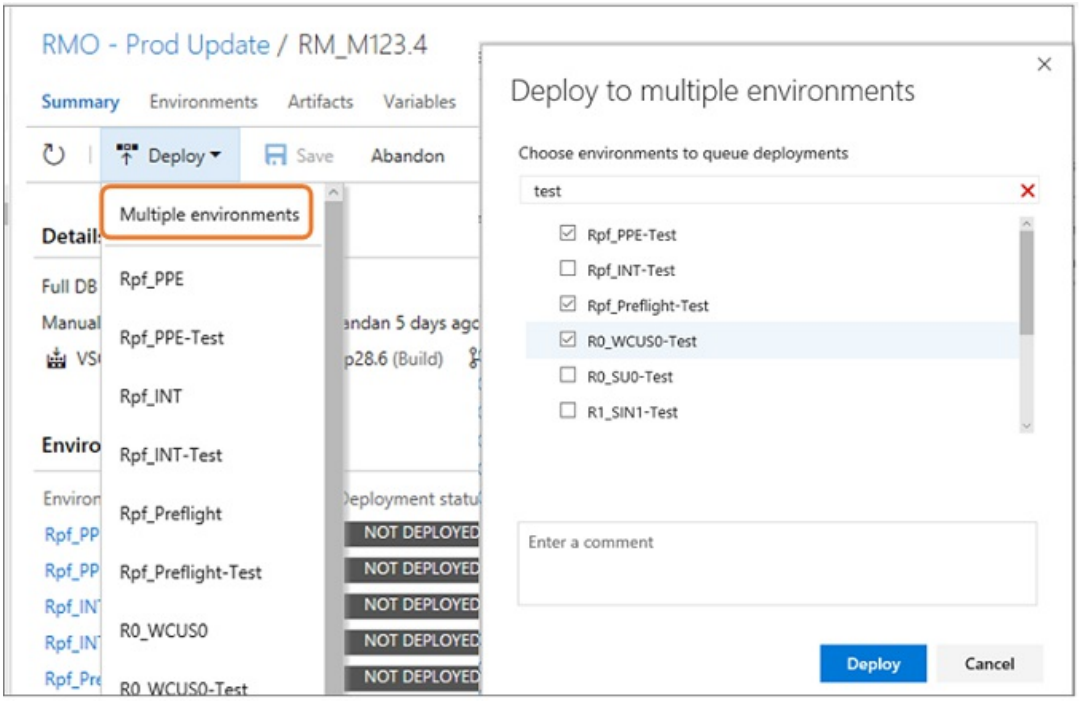
Default properties for Git and GitHub artifact types are now editable

You can now edit the default properties of Git and GitHub artifact types even after the artifact has been linked. This is particularly useful in scenarios where the branch for the stable version of the artifact has changed, and future continuous delivery releases should use this branch to obtain newer versions of the artifact.



Bulk deploy environments manually from release view

You can now manually trigger a **Deploy** action to multiple environments of a release at the same time. This allows you to select multiple environments in a release with failed configurations or deployments, and re-deploy to all of the environments in one operation.

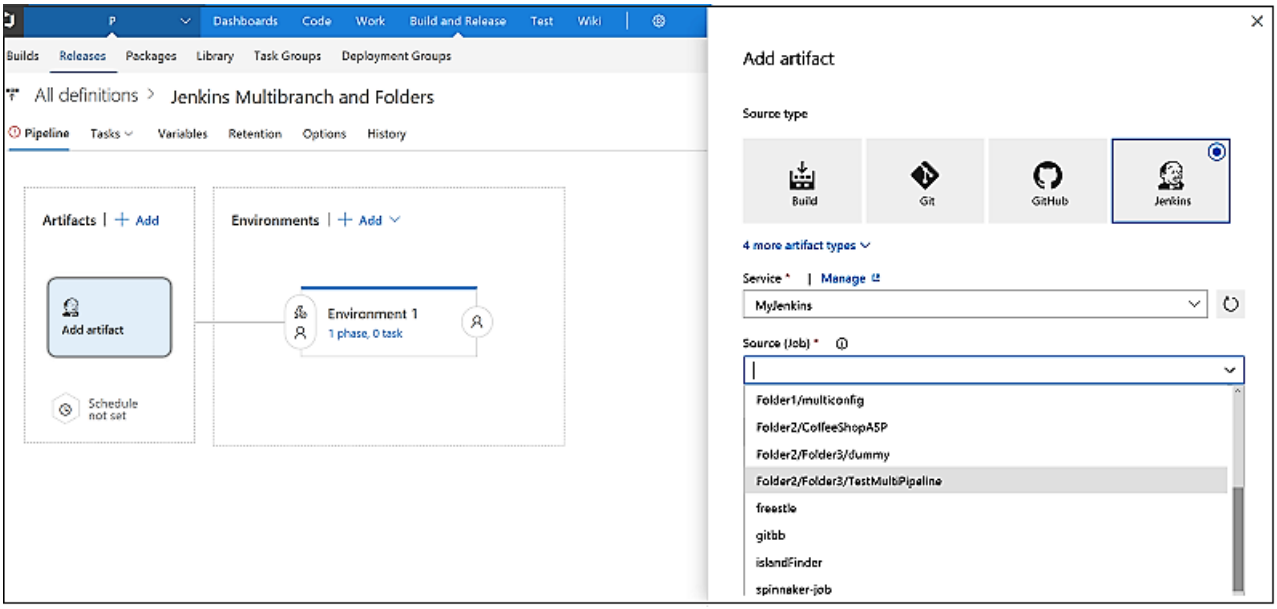


Jenkins multi-branch pipeline support and link jobs organized in folders

Consuming projects from Jenkins just got even better.

First, you can now consume Jenkins multi-branch pipeline projects as artifact sources in a release definition.

Second, while previously you could link Jenkins projects as artifacts only from the root folder of a Jenkins server, now Jenkins projects can be consumed when organized at folder level. You see the list of Jenkins projects, along with folder paths, in the list of sources from which you select the project to be consumed as artifact source.

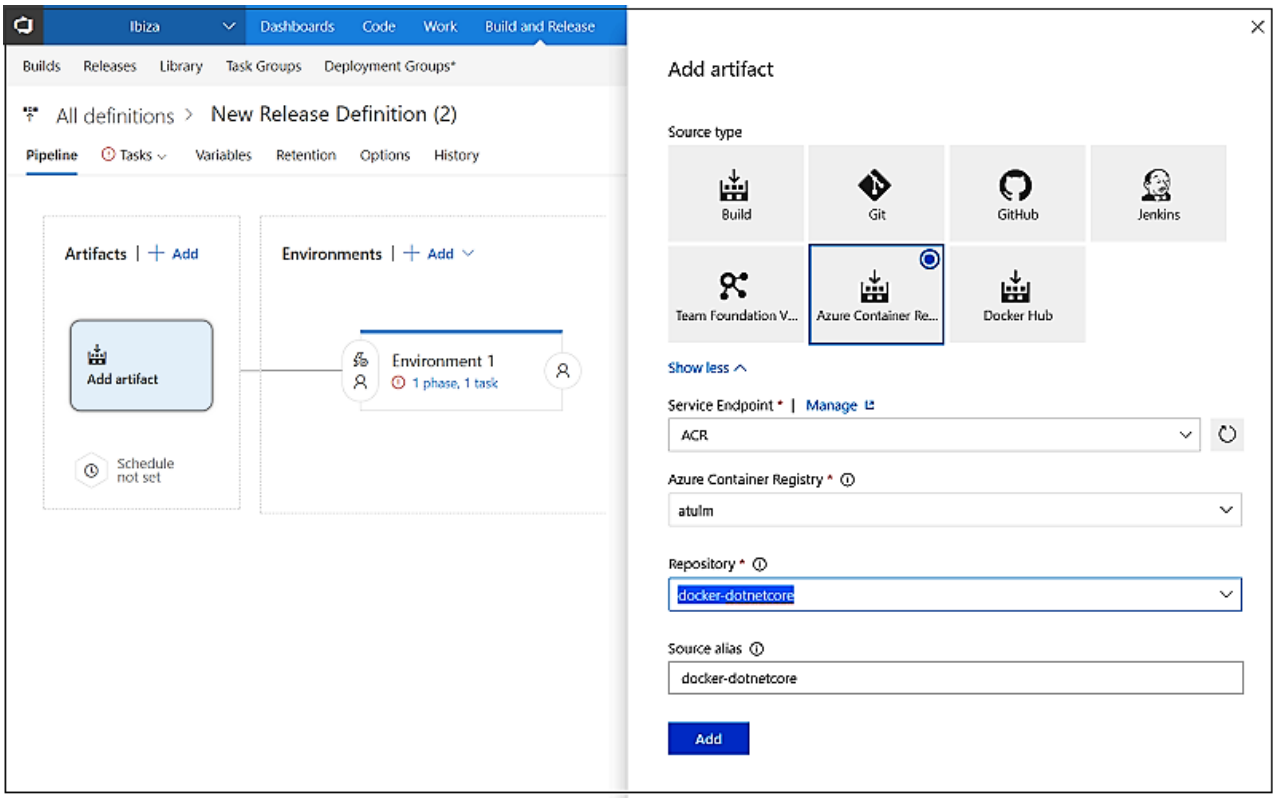


Docker Hub or Azure Container Registry as an artifact source

This feature enables releases to use images stored in a Docker Hub registry or an Azure Container Registry (ACR). This is a first step towards supporting scenarios such as rolling out new changes region-by-region by using the geo-replication feature of ACR or deploying to an environment (such as production) from a container registry that has images for only the production environment.

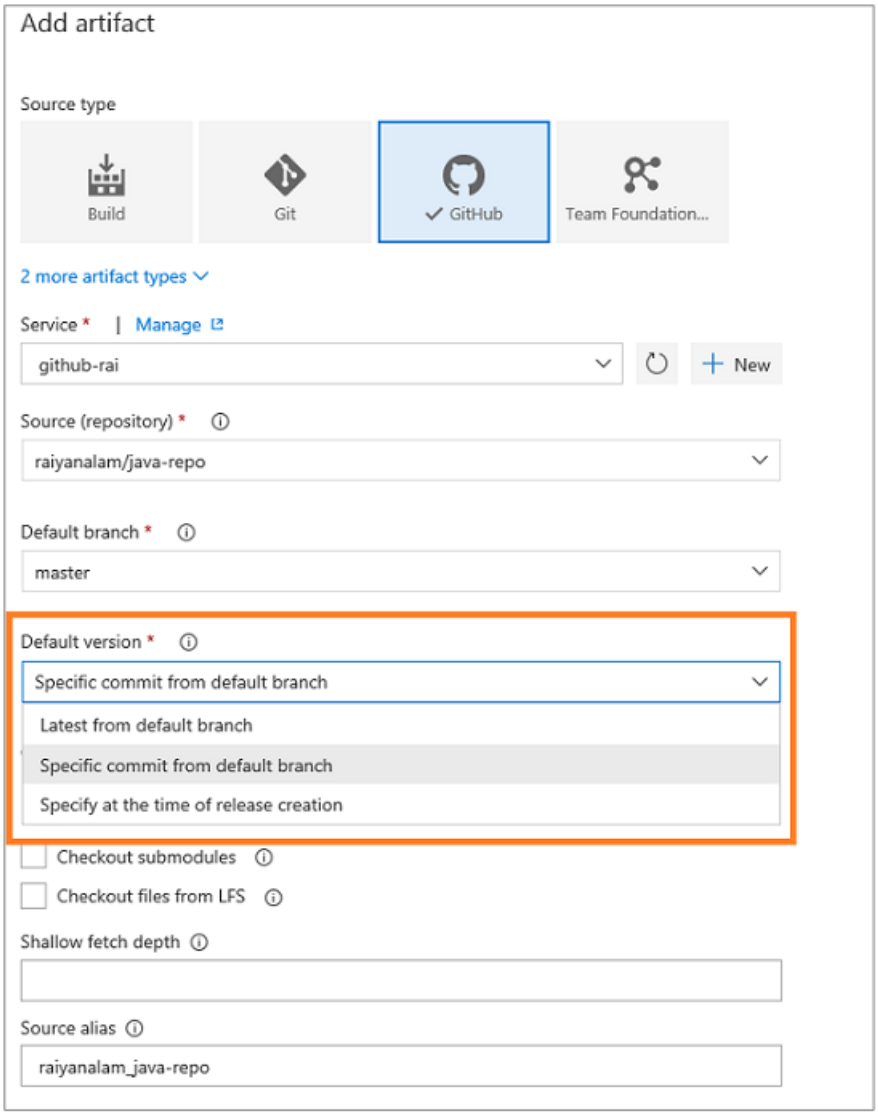
You can now configure Docker Hub or ACR as a first-class artifact in the **+ Add** artifact experience of a release definition. For now the release has to be triggered manually or by another artifact but we look forward to adding a

trigger based on the push of a new image to the registry soon.



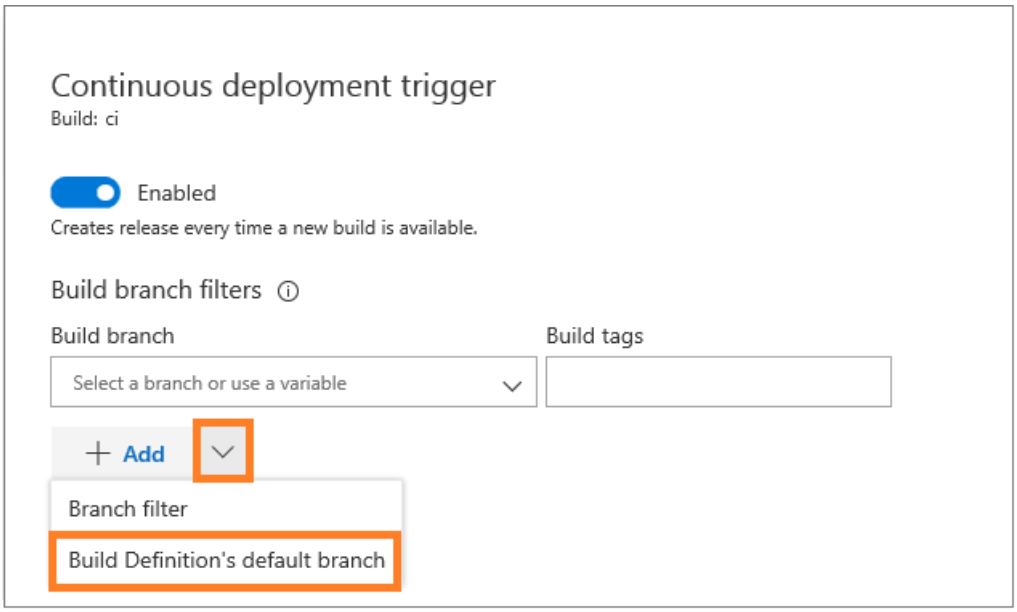
Default artifact versions

There are now several default version options when linking version control artifacts to a release definition. You can configure a specific commit/changeset or simply configure the latest version to be picked from the default branch. Normally you configure it to pick up the latest version, but this is especially useful in some environments where a golden artifact version needs to be specified for all future continuous deployments.



Release triggers branch enhancements

You can now configure a release trigger filter based on the default branch specified in the build definition. This is particularly helpful if your default build branch changes every sprint and the release trigger filters needs to be updated across all the release definitions. Now you just need to change the default branch in the build definition and all the release definitions automatically use this branch. For example, if your team is creating release branches for each sprint release payload, you update it in the build definition to point to a new sprint release branch and the release picks this up automatically.

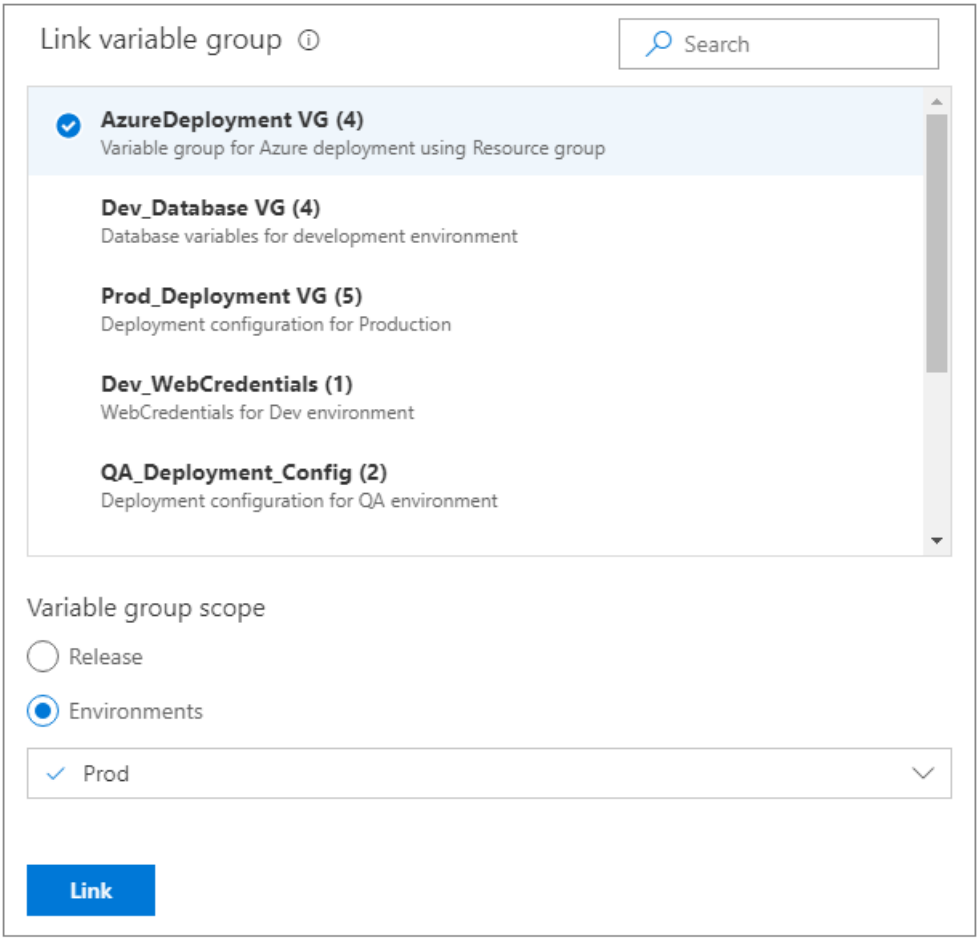


Release trigger for a package management artifact

Now you can set a trigger on a **Package Management** artifact in a Release definition so that a new release is automatically created when a new version of the package has been published. See the documentation for triggers in Release Management for more information.

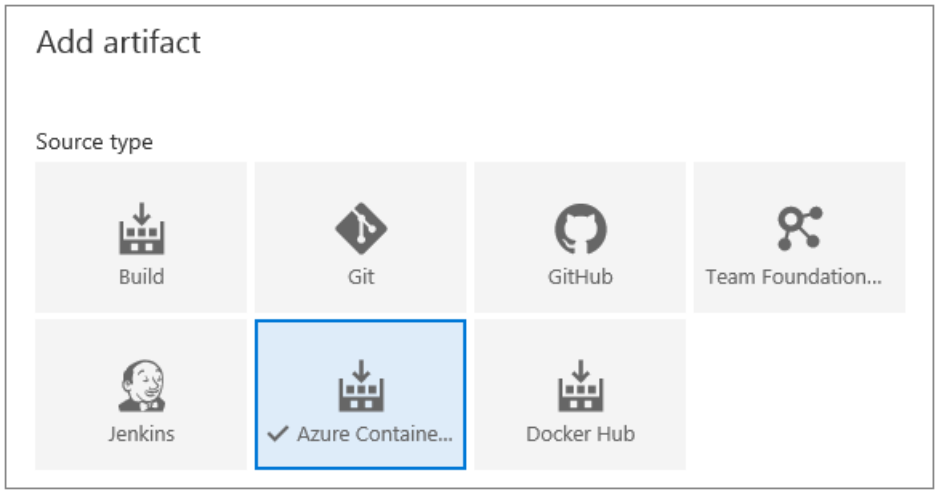
Scope a variable group to specific environments

Previously, when a variable group was added to a release definition, the variables it contained were available to all the environments in the release. Now, you have the flexibility to scope the variable groups to specific environment(s) instead. This makes them available to one environment but not other environments of the same release. This is great when you have an external service, such as an SMTP email service, which is different between environments.



Release automatically from Azure Container Registry and Docker Hub

When deploying containerized apps, the container image is first pushed to a container registry. After the push is complete, the container image can be deployed to a Web App for Containers or a Kubernetes cluster. You can now enable automatic creation of releases on updates to the images stored in **Docker Hub** or **Azure Container Registry** by adding them as an artifact source.

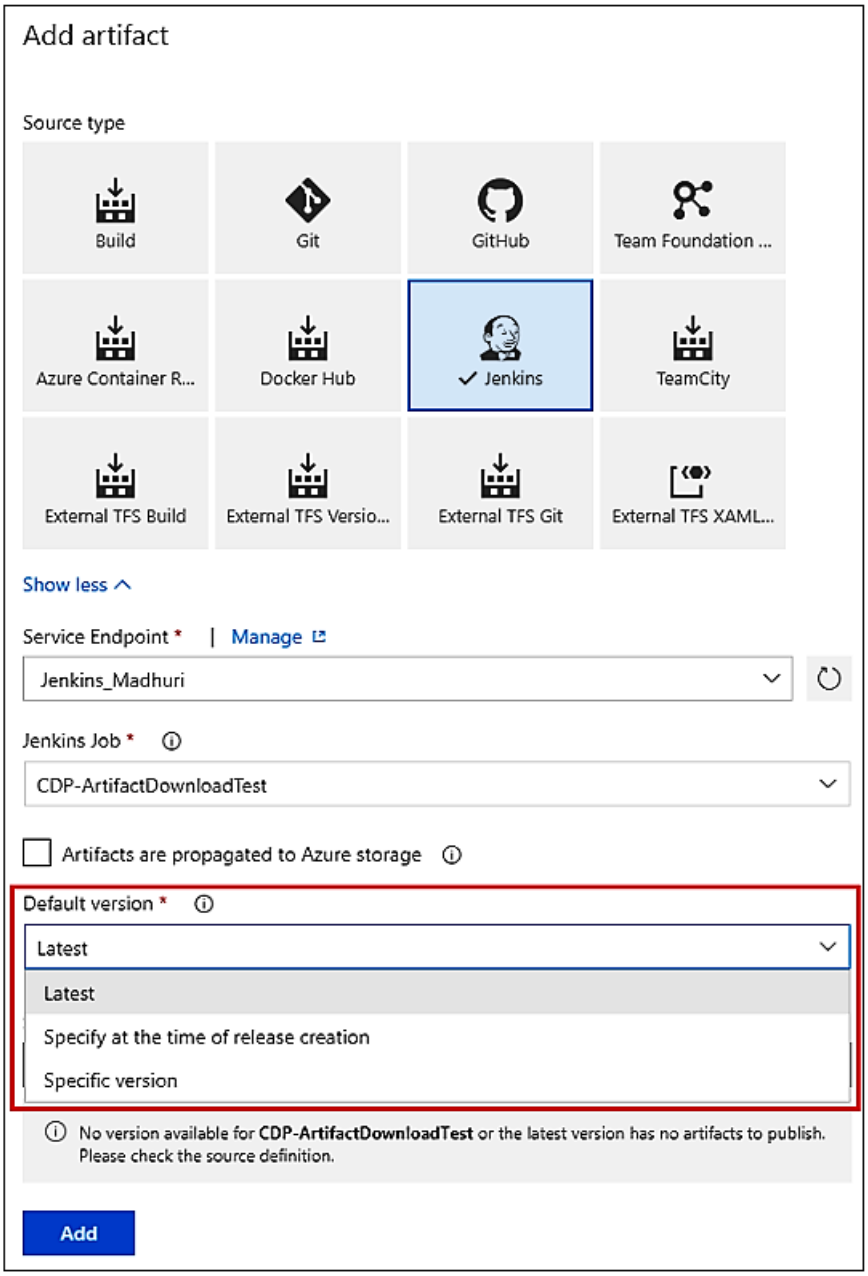


Specify a default version for Jenkins artifacts

When a release with multiple artifacts is auto-triggered, default versions saved in the release definition are picked up for all artifacts. Previously, Jenkins artifacts did not have a default version setting, and so you couldn't set a continuous deployment trigger on a release using Jenkins as the secondary artifact.

Now, you can specify a default version for Jenkins artifacts, with the options you are familiar with:

- Latest
- Specify at the time of release creation
- Specific version



Contribute release gates from extensions

Release gates enable addition of information driven approvals to the release pipelines. A set of health signals are collected repeatedly prior to or post deployment, to determine whether the release should be promoted to the next stage or not. A set of built-in gates are provided, and "Invoke Azure function" has so far been recommended as a means to integrate with other services. We now simplify the route to integrate with other services and add gates through marketplace extensions. You can now contribute custom gate tasks and provide release definition authors an enhanced experience to configure the gate.

Learn more about authoring gate tasks.

Scale deployments to Virtual Machines using Deployment Groups

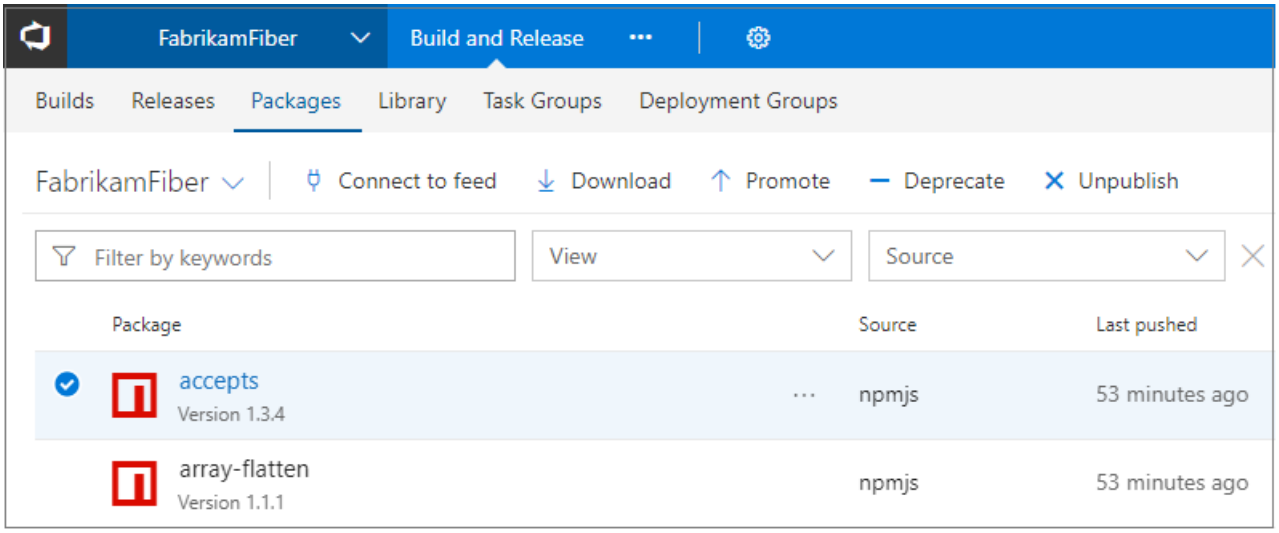
Deployment Groups, that gives robust, out-of-the-box multi-machine deployment, is now generally available. With Deployment Groups, you can orchestrate deployments across multiple servers and perform rolling updates, while ensuring high availability of your application throughout. You can also deploy to servers on-premises or virtual machines on Azure or any cloud and have end-to-end traceability of deployed artifact versions down to the server level.

The agent-based deployment capability relies on the same build and deployment agents that are already available. You can use the full task catalog on your target machines in the Deployment Group phase. From an extensibility perspective, you can also use the REST APIs for deployment groups and targets for programmatic access.

Package

Seamlessly use public packages using upstream sources

Upstream sources for nuget.org and npmjs.com are now available. Benefits include the ability to manage (unlist, deprecate, unpublish, delete, etc.) packages saved from upstream sources as well as guaranteed saving of every upstream package you use.

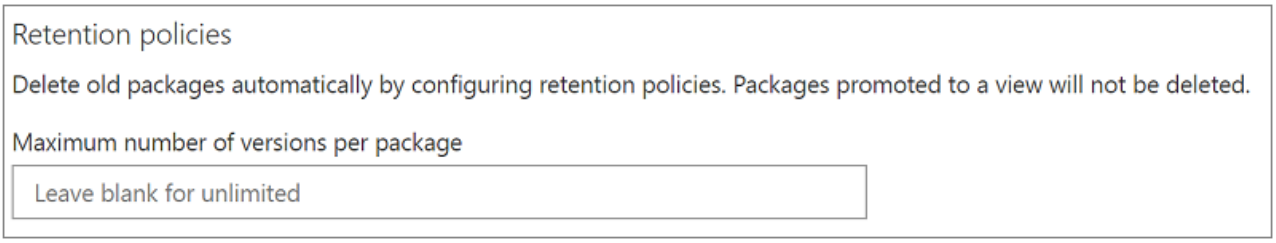


Retention policies in TFS feeds

Until now, TFS package feeds have not provided any way to automatically clean up older, unused package versions. For frequent package publishers, this could result in slower feed queries in the NuGet Package Manager and other clients until some versions were manually deleted.

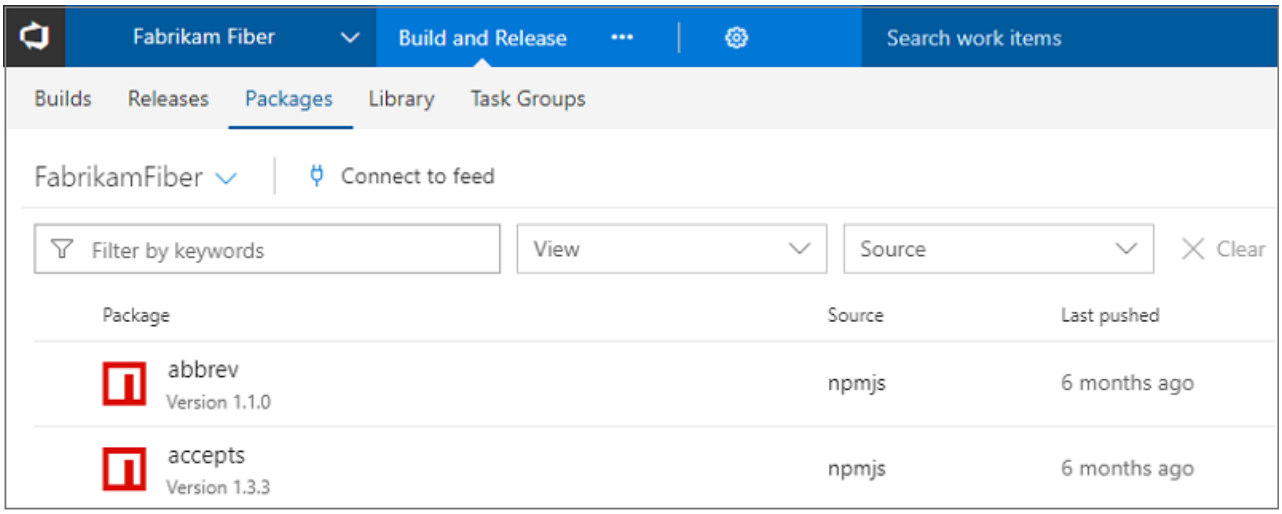
We have now enabled retention policies on TFS feeds. Retention policies automatically delete the oldest version of a package once the retention threshold is met. Packages promoted to views are retained indefinitely, giving you the ability to protect versions that are used in production or used widely across your organization.

To enable retention policies, edit your feed and enter a value in the **Maximum number of versions per package** in the **Retention policies** section.



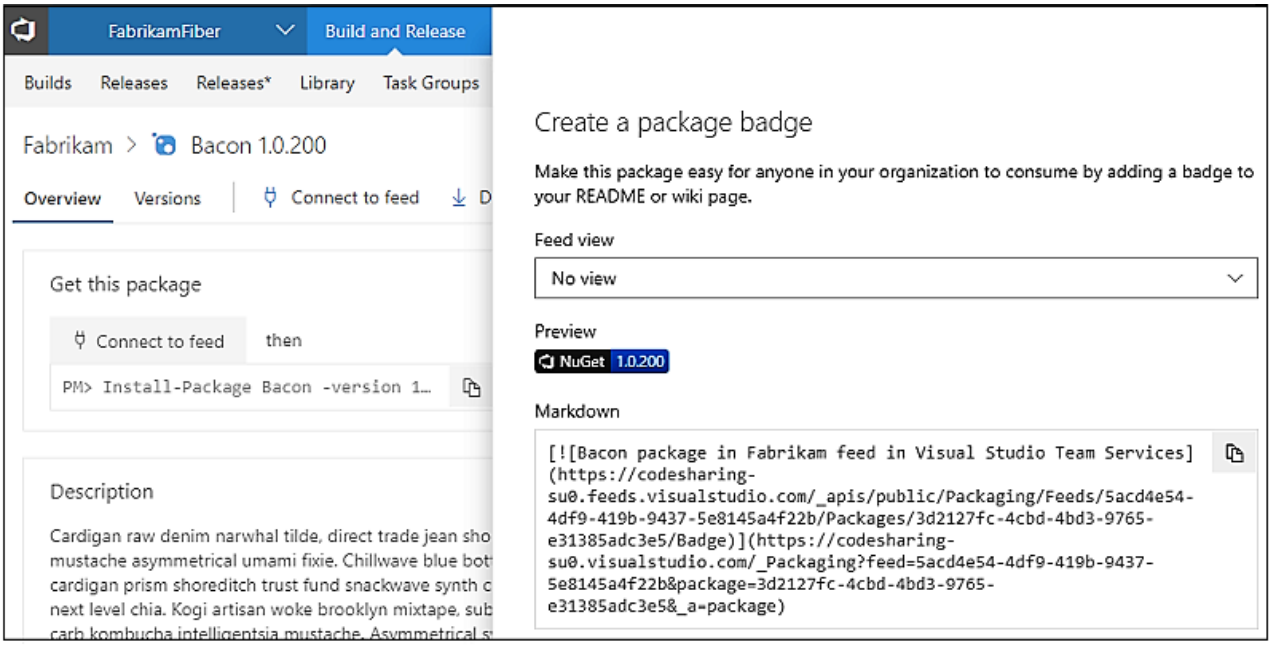
Filtering in package management

The **Packages** page has been updated to use our standard page layout, command bar control, and the new standard filter bar.



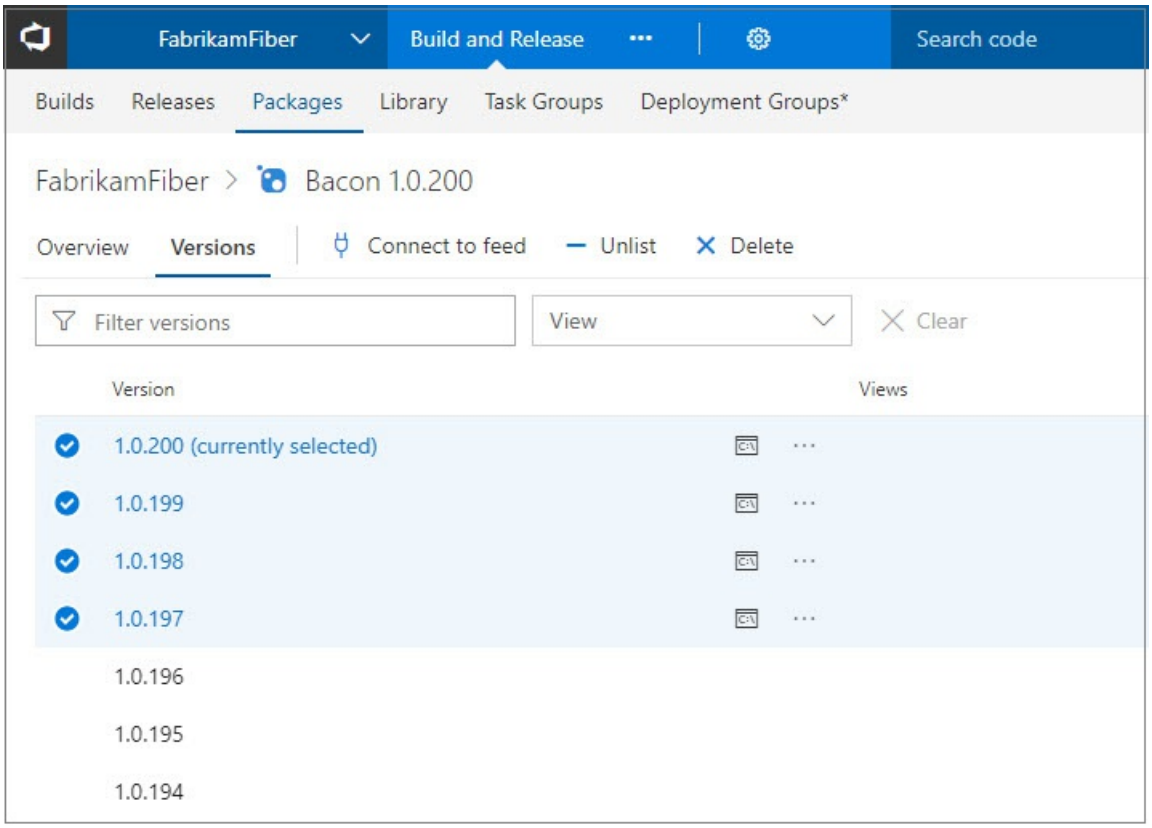
Share your packages using a badge

In the open source community, it is common to use a badge that links to the latest version of your package in your repository's README. You can now create badges for packages in your feeds. Just check the **Enable package badges** option in feed settings, select a package and then click **Create badge**. You can copy the badge URL directly or copy pre-generated Markdown that links the badge back to your package's details page.



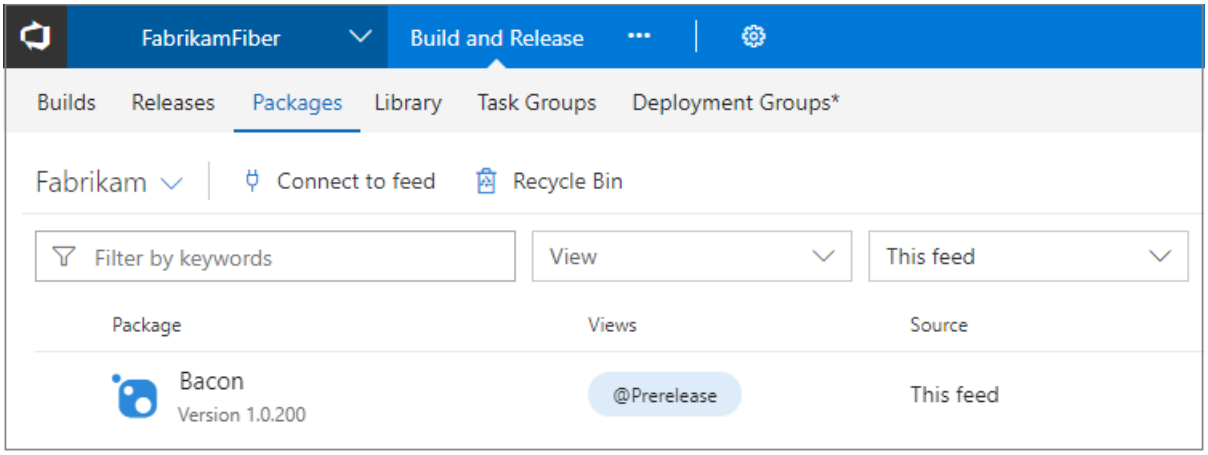
Previous package versions are now a full-page list

We received a lot of feedback on the updated Package Management experience, where we moved the list of previous package versions into a breadcrumb picker on the package details page. We have added a new **Versions** pivot that brings more information about prior versions and makes it easier to copy the version number or get a link to an old version.



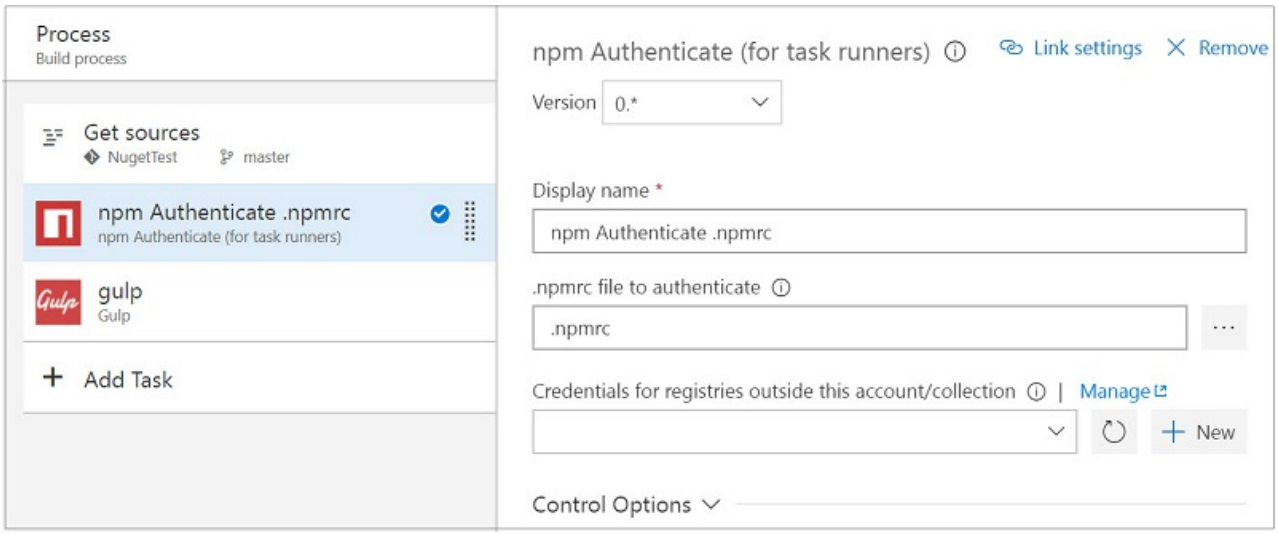
View quality of a package version in the package list

On the package list, you can now see the view(s) of each package version to quickly determine their quality. See the release views documentation for more information.) documentation for more information.



Gulp, Yarn, and more authenticated feed support

The npm task today works seamlessly with authenticated npm feeds (in **Package Management** or external registries like **npm Enterprise** and **Artifactory**), but until now it has been challenging to use a task runner like **Gulp** or an alternate npm client like **Yarn** unless that task also supported authenticated feeds. We have added a new **npm Authenticate** build task that adds credentials to your .npmrc so that subsequent tasks can use authenticated feeds successfully.

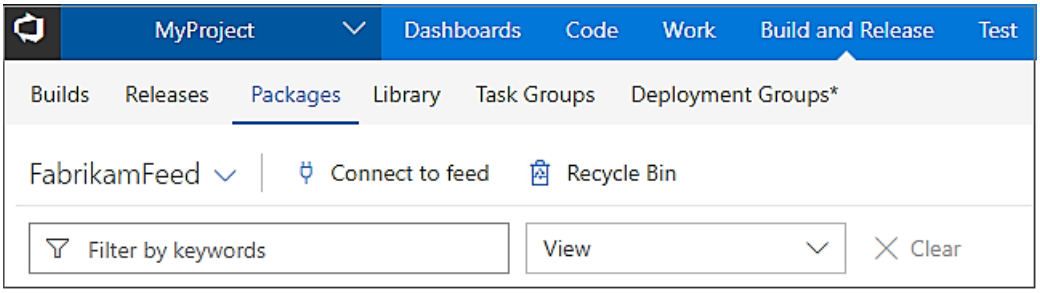


Package feed default permissions now include Project Administrators

In the past, creating a feed sets the creating user as the only feed owner, which can cause administration challenges in large organizations if that user switches teams or leaves the organization. To remove this single point of failure, creating a feed now uses the user's current project context to get the **Project Administrators** group and make it an owner of the feed as well. As with any permission, you can remove this group and further customize feed permissions using the feed settings dialog.

Recycle and restore packages

Deleting unused packages can help keep the package list clean but sometimes it can be done by mistake. Now you can restore deleted packages from the **Recycle Bin**. Deleted packages are retained in the Recycle Bin for 30 days, giving you ample time to restore if you need to.



Link to packages from anywhere

Although you could share the URL to a package found in the **Packages** hub in the past, it was often difficult to use because you needed to include a project in the URL, that may or may not apply to those using the link. With this Update, you can now share packages using a URL that automatically select a project the recipient has access to.

The URL format is: `https://<TFSserverURL>/_packaging?feed=<feed>&package=<package>&version=<version>&protocolType=<NuGet|Npm|Maven>&_a=package`

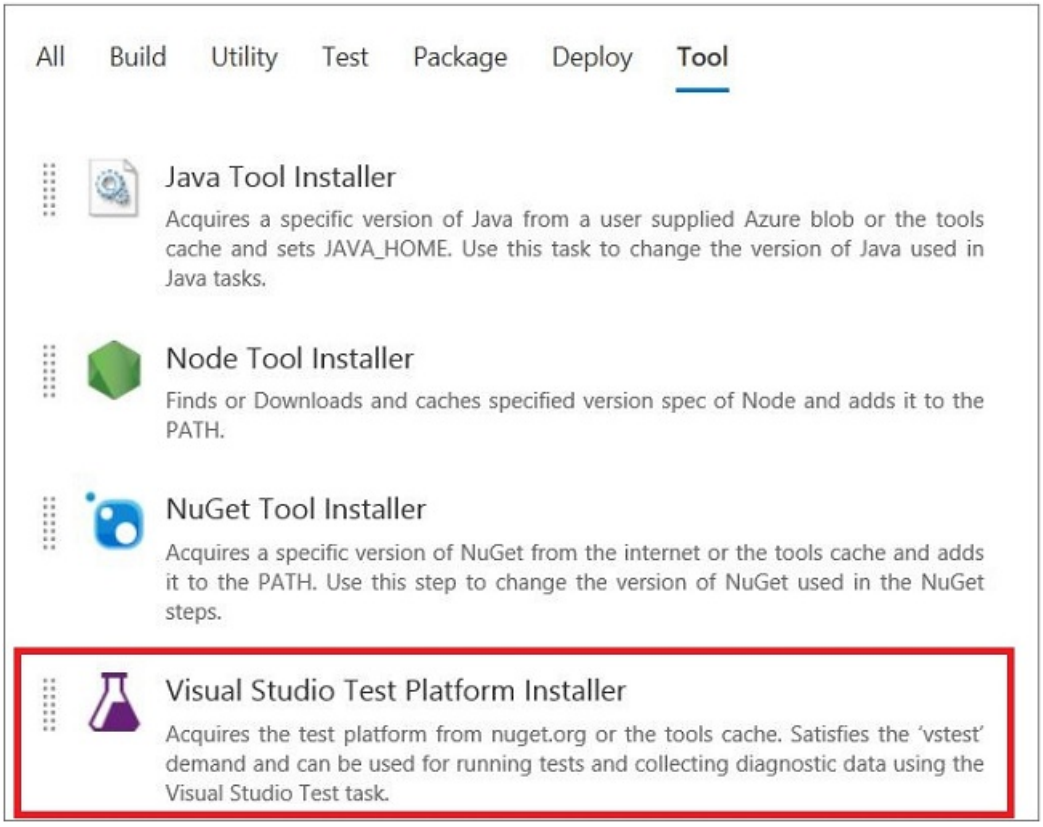
All parameters except `<TFSserverURL>` are optional, but if you provide a package, you must provide the protocol type.

Test

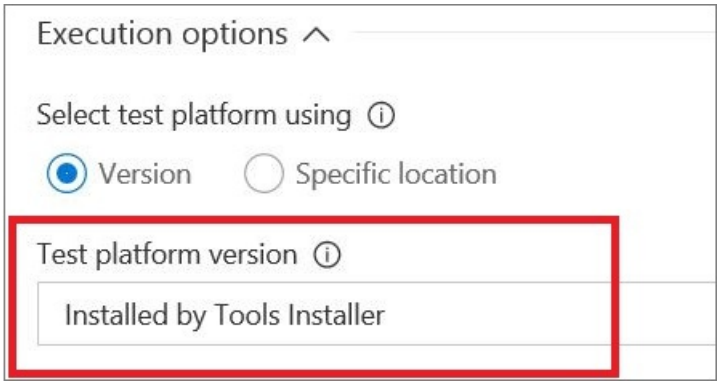
Visual Studio Test task does not need full Visual Studio

The **Visual Studio Test** task in build/release requires Visual Studio on the agent to run tests. Rather than installing Visual Studio to run tests in production environments or for merely distributing tests over multiple agents, use the new **Visual Studio Test Platform Installer** task. This task acquires the test platform from nuget.org and adds it to the tools cache. The installer task satisfies the **vstest** demand and a subsequent **Visual Studio Test** task in the definition can run without needing a full Visual Studio install on the agent.

From the task catalog, add the installer task in your definition.



Configure the subsequent **Visual Studio Test** task to use the bits acquired through the installer.



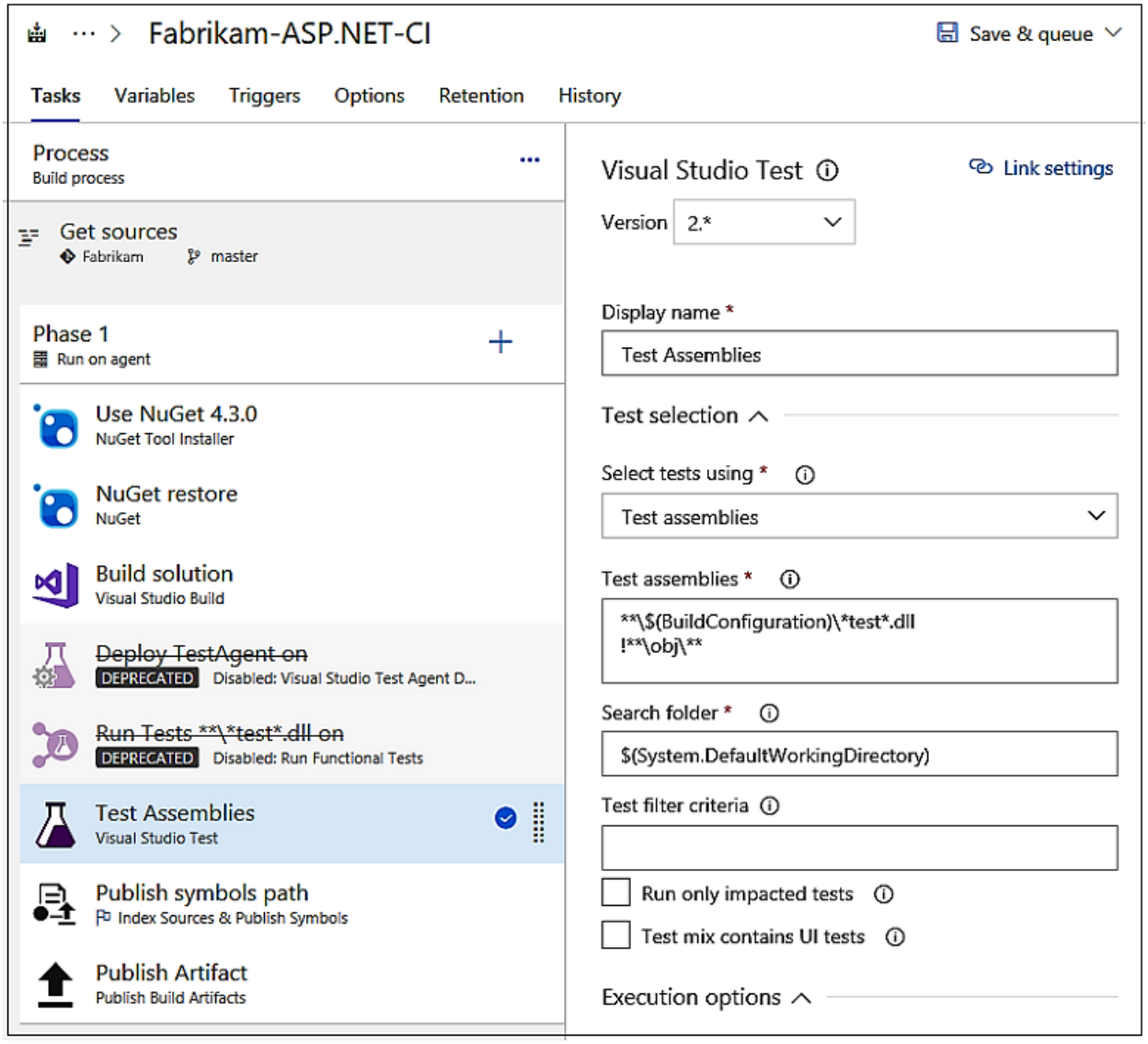
> **NOTE**
> Limitations: The Test Platform package on NuGet currently does not support running Coded UI test. Enabling support for Coded UI test is on the backlog. The Test Platform package on NuGet is cross-platform, but VSTest task currently does not support running .NET core tests. To run .NET core tests, use the 'dot net' task.

Run Functional Tests and Deploy Test Agent tasks are now deprecated

Last year, we started on the journey to unify agents across build, release, and test. This was intended to address various pain points associated with using WinRM based **Deploy Test Agent** and **Run Functional Tests** tasks. It also enables you to use the **Visual Studio Test** (VSTest) task for all your testing needs, including:

- Unit tests
- Functional (UI/non-UI) tests
- MSTest based tests
- Third party framework-based tests
- Assembly-based test specification or running tests with Test Plan/Test Suite
- Single agent test execution as well as distributing tests over multiple agents

The unified agents approach also allows administrators to manage all machines that are used for CI/CD in a uniform manner.



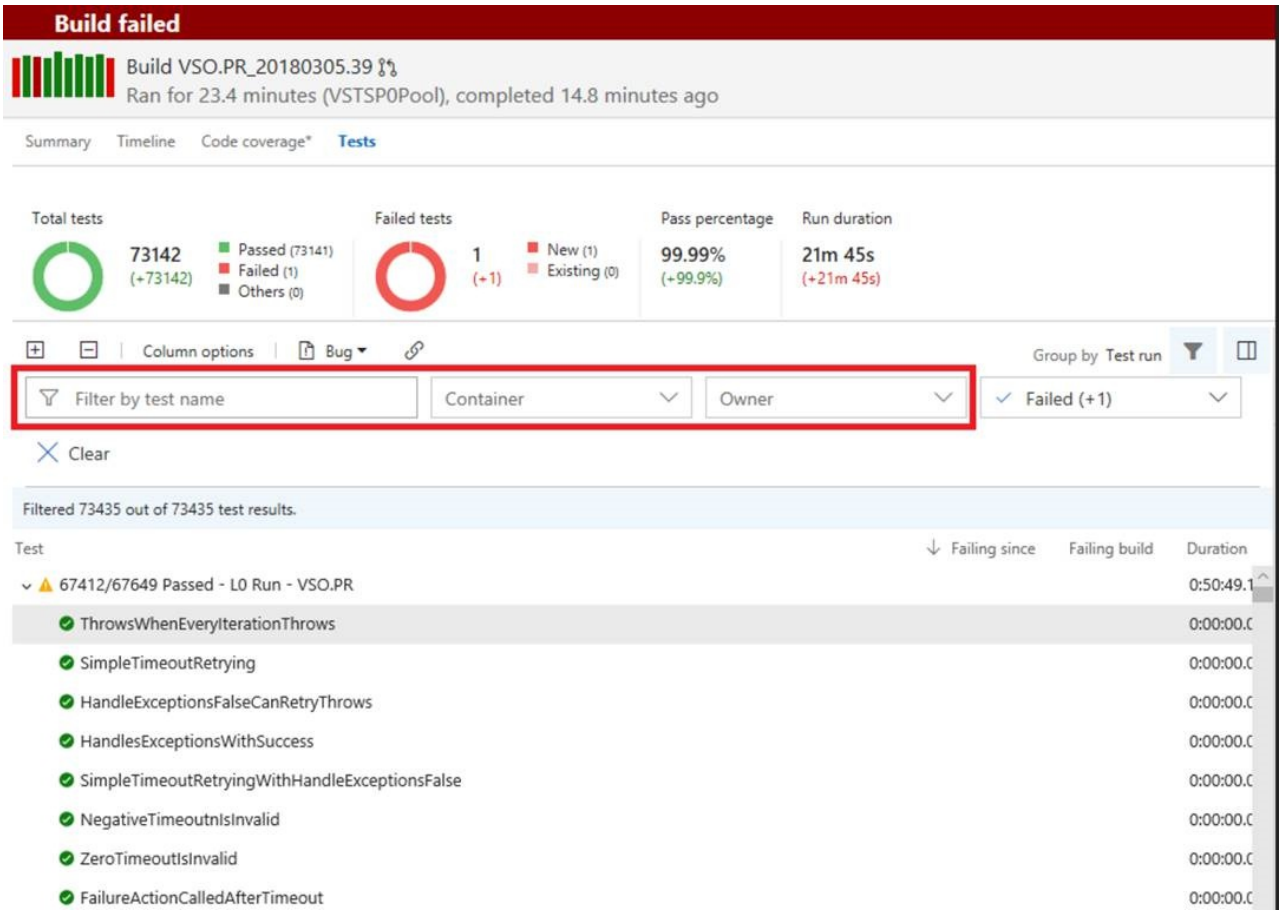
We have delivered several crucial pieces to enable this capability, including:

- Agents can be configured for UI testing
- Visual Studio Test Platform Installer allows VSTest task to run without needing Visual Studio pre-installed
- Both Build and Release definitions can be created with multiple phases and have the ability to use different agent queues for each phase
- Automated test cases can be run from the Test hub using the VSTest task

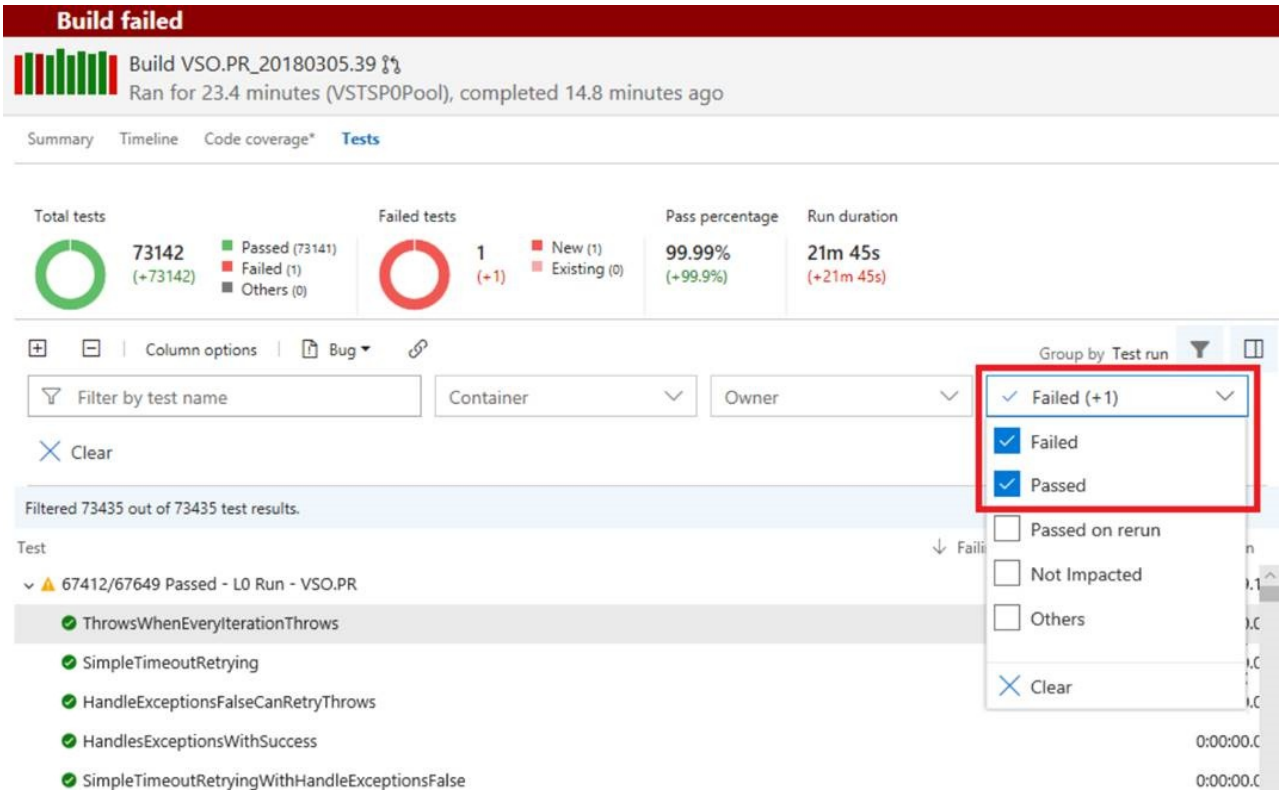
With all the above now in place, we are ready to deprecate these two tasks. While existing definitions that use the deprecated tasks will continue to work, we encourage you to move to using VSTest to take advantage of continued enhancement over time.

Filter large test results

Over time test assets accrue, and large applications can easily grow to thousands of tests. Teams are looking for better ways to navigate through large sets of test results to be productive while identifying test failures, associated root cause, or ownership of issues. To enable this, we have added three new filters under **Tests Tab** in Build and Release as **Test Name**, **Container** (DLLs) and **Owner** (Container Owner).



Additionally, the existing **Outcome** filter now provides the ability to filter for multiple outcomes. The various filter criterion are cumulative in nature. As a user, when I want to see the outcome of my tests for a change I just committed, I can filter on the **Container** (DLL name), **Owner** (DLL owner), **Test Name**, or all of them, to get to the results relevant to me.

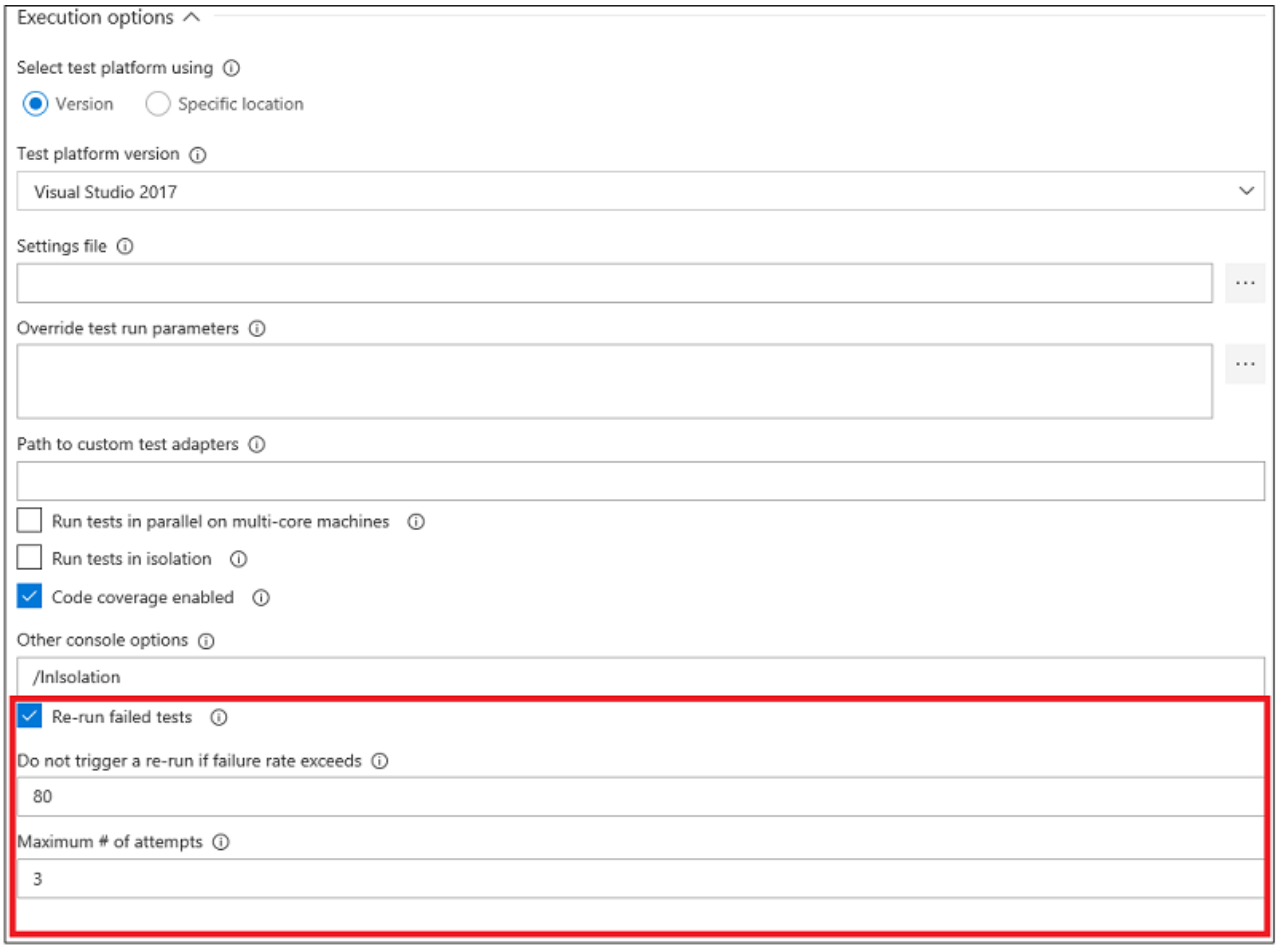


Identify flaky tests

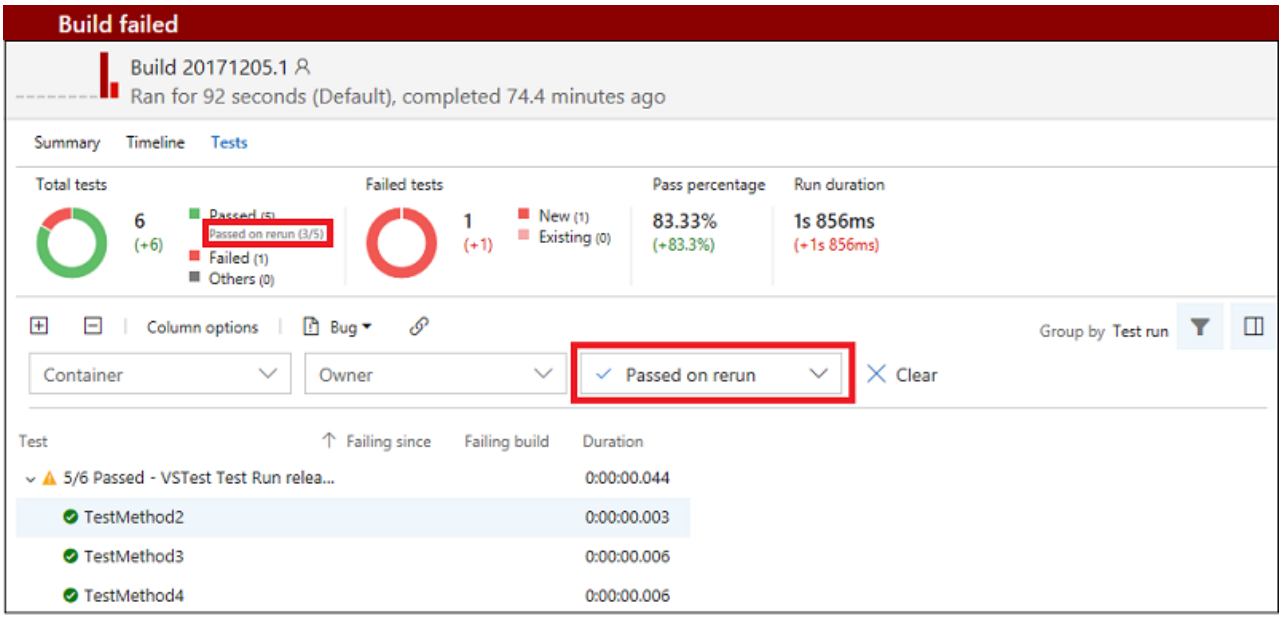
Sometimes tests are flaky - they fail on one run and pass on another without any changes. Flaky tests can be frustrating and undermines confidence in test effectiveness - causing failures to be ignored and bugs to slip

through. With this Update, we have deployed the first piece of a solution to help tackle the problem of flaky tests. You can now configure the **Visual Studio Test** task to re-run failed tests. The test results then indicate which tests initially failed and then passed on re-run. Support for re-run of data driven and ordered tests are coming later.

The **Visual Studio Test** task can be configured to control the maximum number of attempts to re-run failed tests and a threshold percentage for failures (e.g. only re-run tests if less than 20% of all tests failed) to avoid re-running tests in event of wide spread failures.



In the **Tests** tab under **Build and Release**, you can filter the test results with Outcome as "Passed on rerun" to identify the tests that had an unreliable behavior during the run. This currently shows the last attempt for each test that passed on re-run. The Summary view is also modified to show "Passed on rerun (n/m)" under Total tests, where n is the count of tests passed on re-run and m is total passed tests. A hierarchical view of all attempts is coming in next few sprints.



Preview improvements and support for different log types generated by Visual Studio Test task

We enhanced the VSTest task to publish logs generated by different kind of logging statements corresponding to standard output and standard error for failed tests. We have also improved the preview experience to support viewing text and log file formats, with capability to search in the log files.

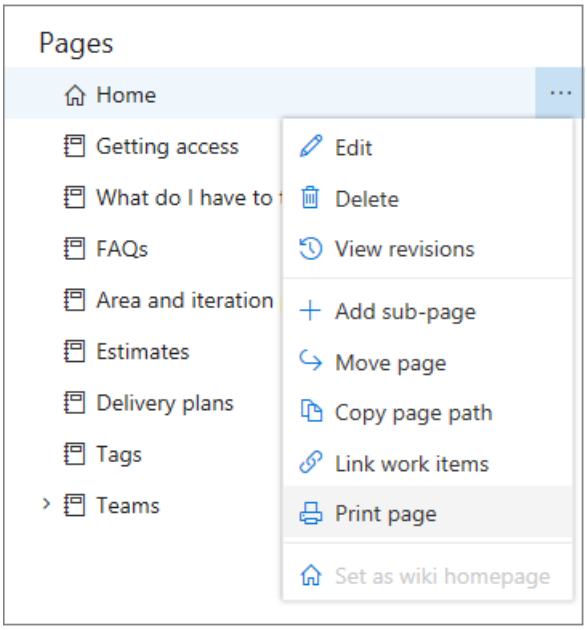
Wiki

Wiki search

You can search for your favorite Wiki pages by title or content right alongside code and work items. You can read more about Wiki search in the Microsoft DevOps Blog.

Print Wiki pages

Wiki can be used for a variety of content. Sometimes it can be useful to print content from **Wiki** to read in your spare time, add comments using pen and paper, or even share an offline PDF copy with those outside of your VSTS project. Now, simply click on the context menu of a page and select **Print page**. This feature was prioritized based on a suggestion.

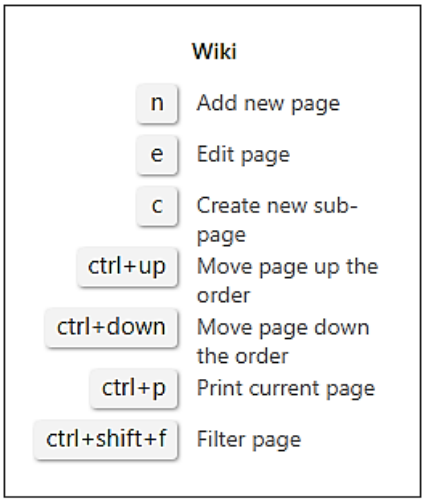


> **NOTE**
> Currently this feature is not supported on Firefox.

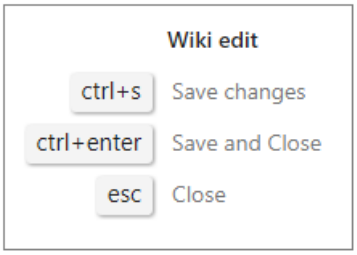
Contribute to Wiki pages with ease using keyboard shortcuts

You can now use shortcuts to perform common edit and view actions in **Wiki** even faster using only your keyboard.

While viewing a page, you can add, edit, or create a subpage, for example:



While editing a page, you can quickly save, save and close, or just close.



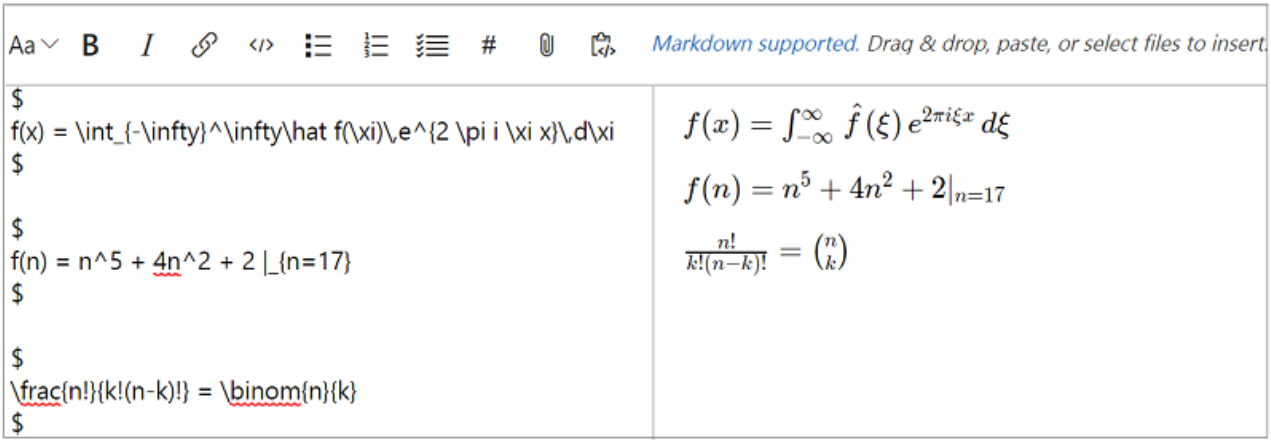
These are in addition to standard editing shortcuts such as Ctrl+B for **bold**, Ctrl+I for *italics*, Ctrl+K for linking etc. See the full list of keyboard shortcuts for more information.

Rich markdown rendering in code repo markdown

You can now create rich README.MD files in the code repositories. The markdown rendering of the MD files in code repositories now supports HTML tags, Block quotes, Emojis, image resizing, and mathematical formulas. There is parity in markdown rendering in Wiki and MD files in code.

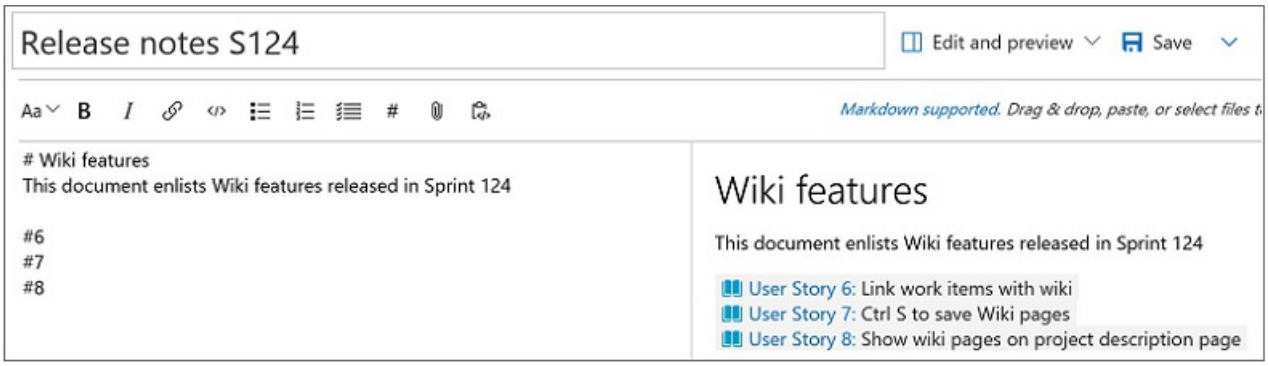
Wiki supports mathematical formulas

If your application deals with mathematical formulas and equations, you can now put them in Wiki using the LaTeX format.



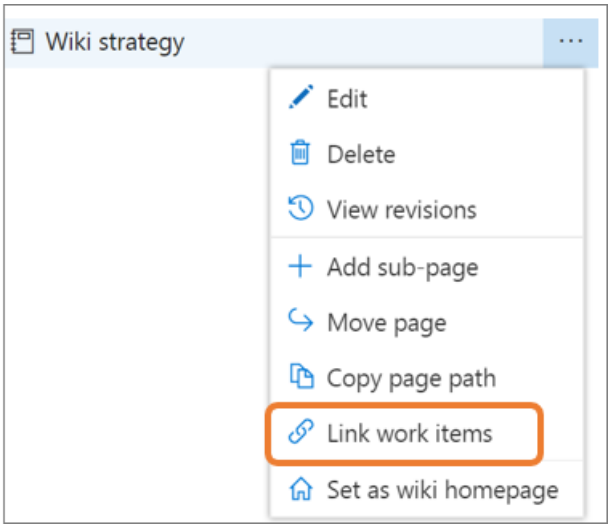
Reference work items in Wiki

Now you can reference work items in Wiki pages by pressing the '#' key to get a list of the most recently accessed work items and selecting the work item of interest. This is particularly useful while writing release notes, epics, specs, or other pages that require referring to a work item.

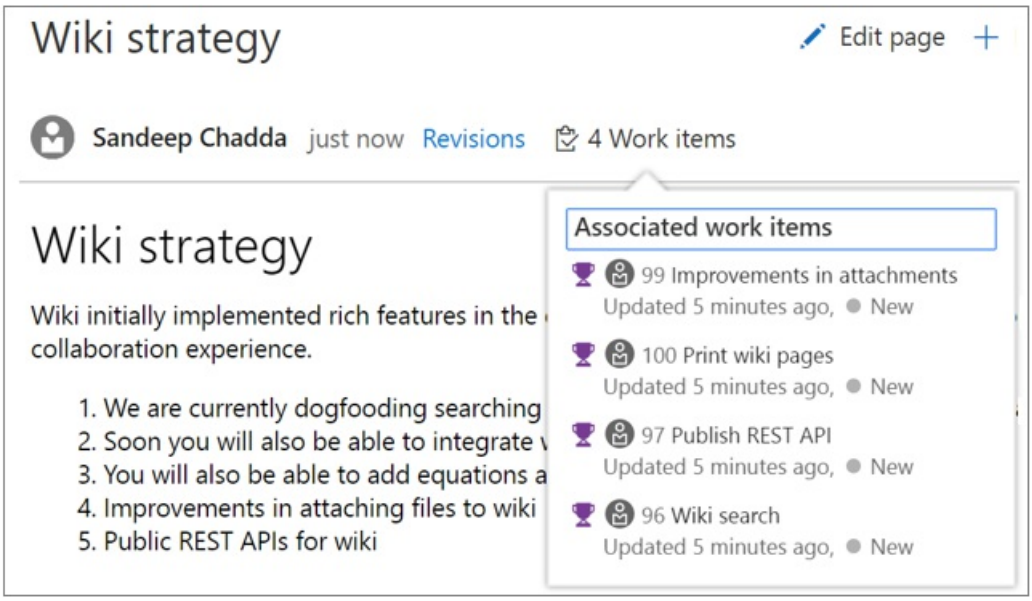


Link work items and Wiki pages

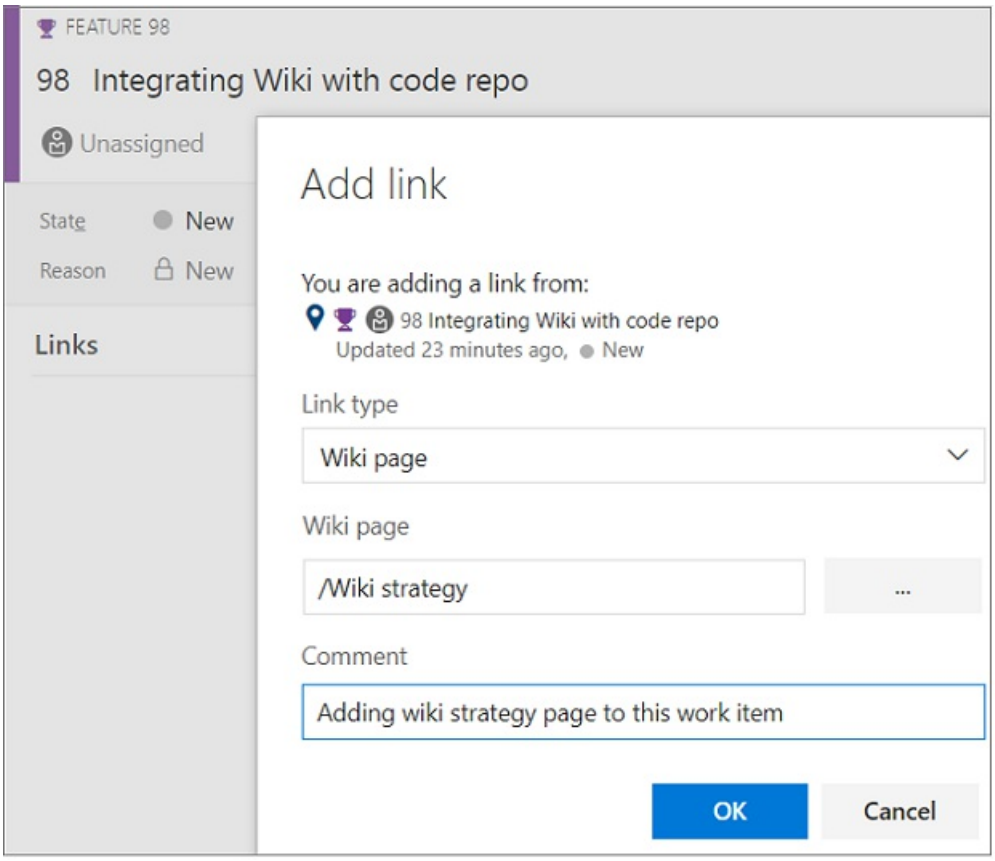
Now you can link a work item to a Wiki and vice versa. You can link work items to Wiki to create epic pages, release notes, and planning content that helps you track the work items associated with a Wiki page and validate what percentage of your epic page is complete.



Linked work items then show up on the Wiki page.

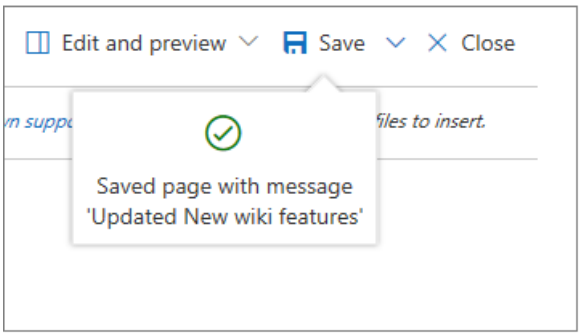


Add a link to a Wiki page from a work item through the new "Wiki page" link type.



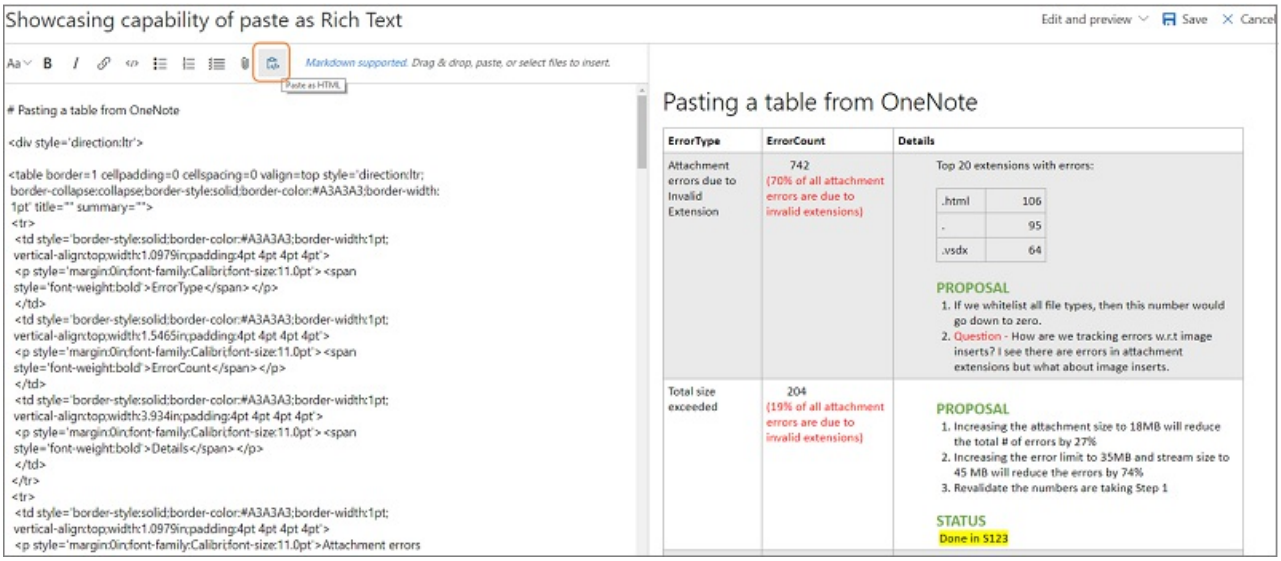
Ctrl+S to save Wiki page

We heard you wanted a quicker and easier way to save a Wiki page. Now you can simply use Ctrl+S keyboard shortcut to save a page with a default revision message and continue editing. If you would like to add a custom revision message just click on the chevron next to the save button.



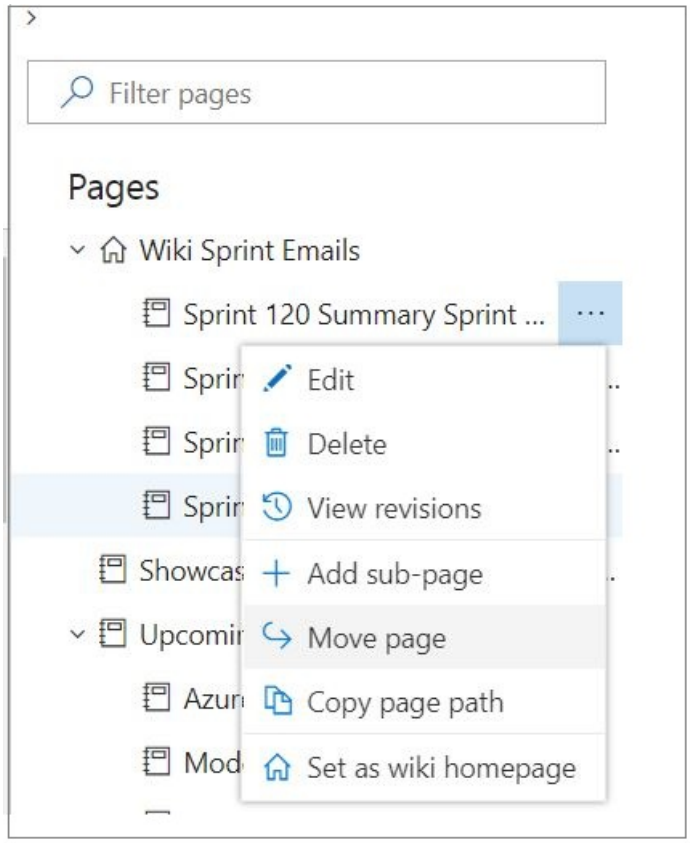
Paste rich Wiki content as HTML

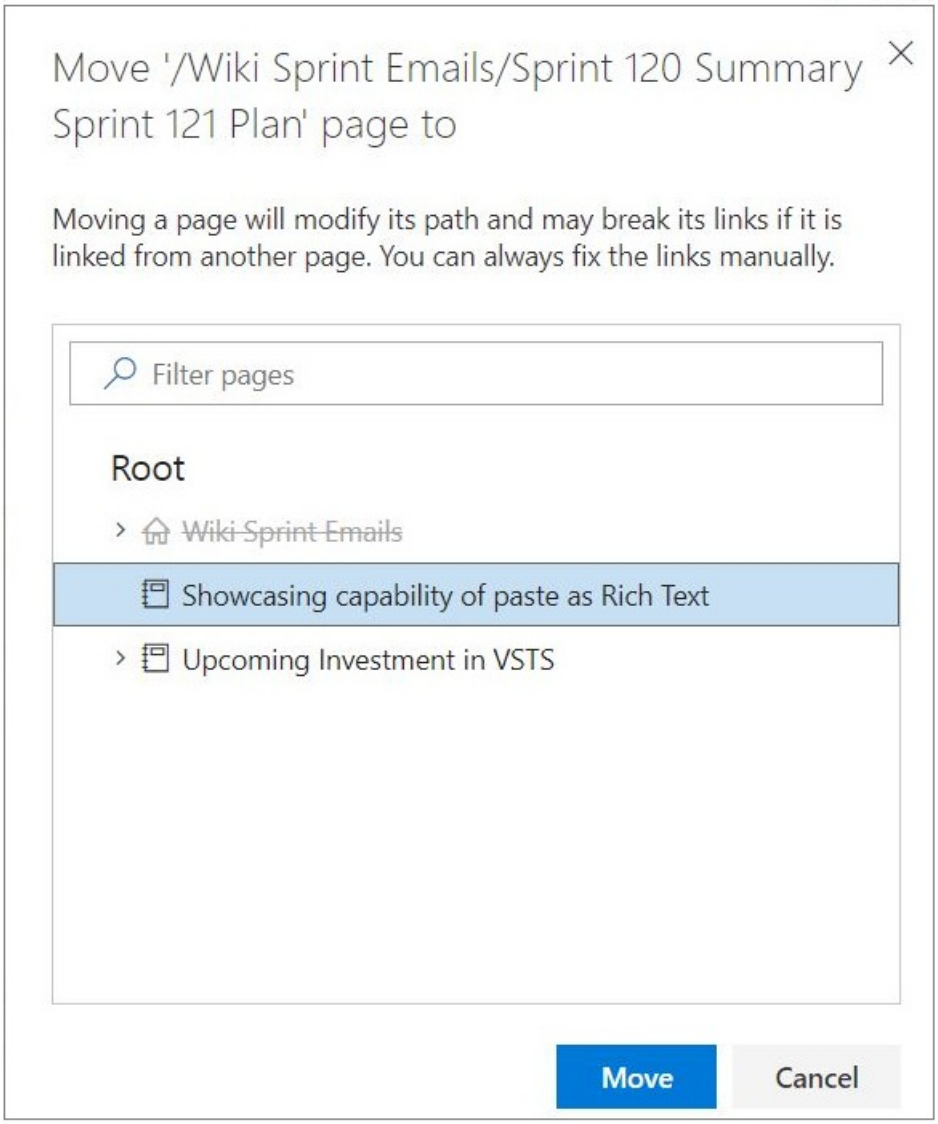
You can now paste rich text in the markdown editor of **Wiki** from any browser-based applications such as Confluence, OneNote, SharePoint, and MediaWiki. This is particularly useful for those who have created rich content such as complex tables and want to show it in **Wiki**. Simply copy content and paste it as HTML.



Move page in Wiki using keyboard

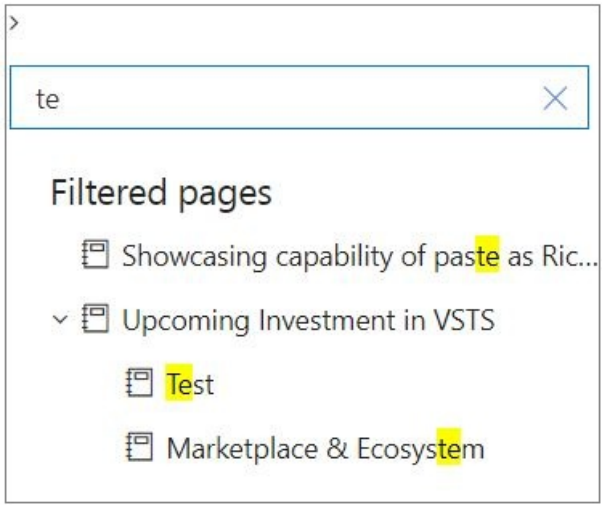
Earlier in **Wiki**, users could not reorder or re-parent pages using keyboard and this would impact users who prefer with keyboard operations. Now you can reorder pages by using Ctrl + Up or Ctrl + Down commands. You can also re-parent pages by clicking **Move page** in the context menu of a page and select the new parent page to move.



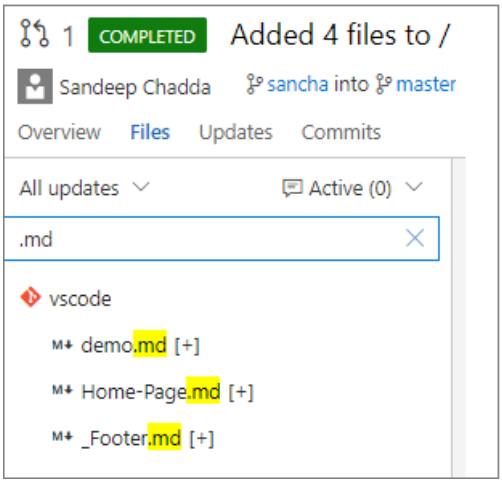


Filter text highlighting

Filtering the navigation pane in **Wiki** shows the entire page hierarchy. For example, if you filter a page titled "foobar" the filtered navigation pane would show all parent pages as well. This can cause confusion as to why pages not titled "foobar" are showing up in filtered sets of results. Now, filtering content in **Wiki** highlights the text being searched to give a clear picture of the titles that are filtered and those that are not.

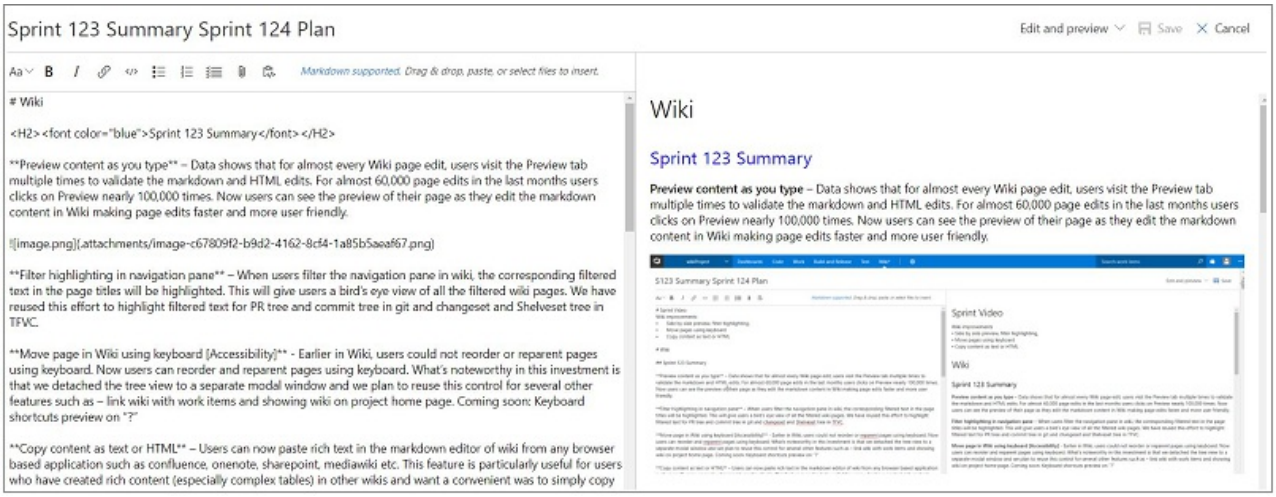


You will observe similar behavior in all code navigation panes as well. For example, the file navigation pane in pull requests, commits, changesets, and shelvesets.



Preview content as you edit Wiki pages

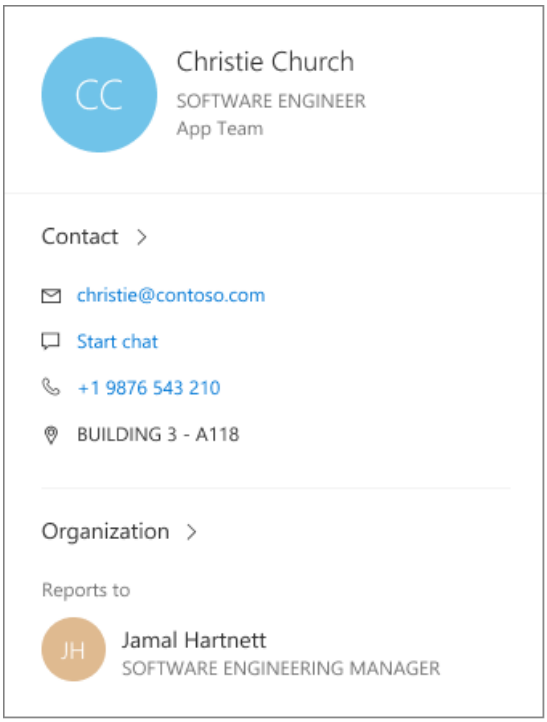
Data shows that users almost always **Preview** a Wiki page multiple times while editing content. For each page edit, users click on **Preview** 1-2 times on average. This results in a slow and sub-optimal edit experience and can be particularly time consuming for those new to markdown. Now you can see the preview of your page while editing.



General

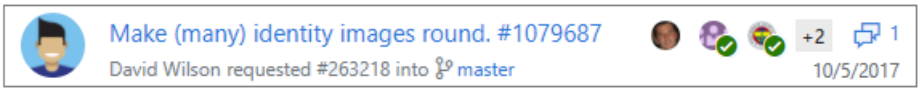
Profile Cards

There are multiple areas in TFS where information associated to a particular individual is shown, such as, but not limited to: pull requests created by an individual, and work items assigned to an individual. However, there is limited information about the individual itself for you to gain complete context. The new Profile Card replaces the existing profile card in TFS. The updated profile card allows you to interact with and learn more about users within your TFS account. Through integrations with your default email and IM client, Active Directory (AD) users can send emails and start chats directly from the profile card. AD users can also see the organizational hierarchy within the profile card. Profile cards can be activated within project home page - team members section, version control, work items and Wiki sections by clicking on the contact card icon, profile picture, or users name within comments.



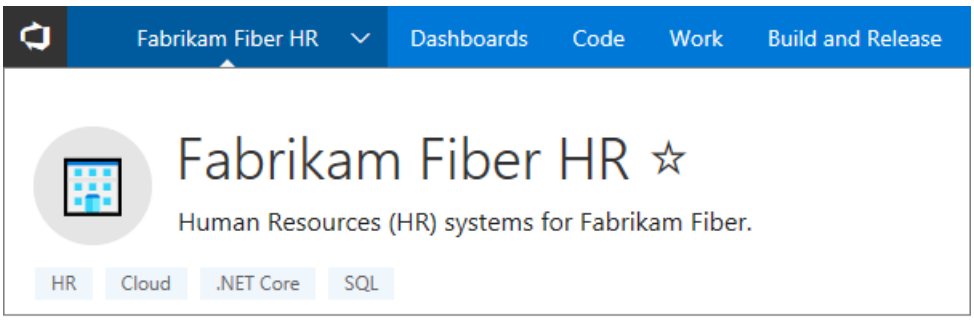
Circle avatars

Circle avatars are here! All profile pictures in the service now displays in a circle shape, rather than a square. As an example, here is the actual pull request for this change (note the circular, non-square avatars).



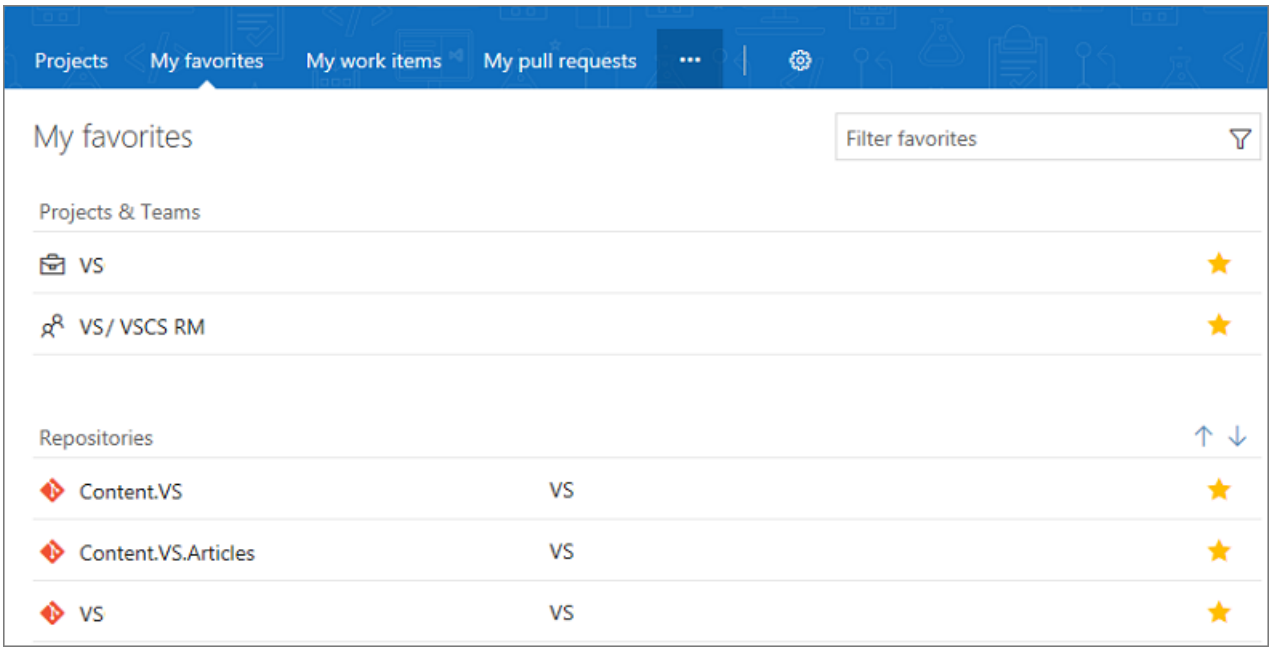
Project tags

You can now adorn projects with important keywords (tags). Tags are easily added and deleted directly from the project home page (by administrators) allowing users to quickly understand more about the purpose and scope of the project. We have more planned for how project tags can be leveraged, so stay tuned for more news here.



Re-order favorite groups

You can now re-order the groups on the account **My favorites** page using the up and down arrows in each group header.



Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2018 Update 1 Release Notes

9/12/2018 • 7 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

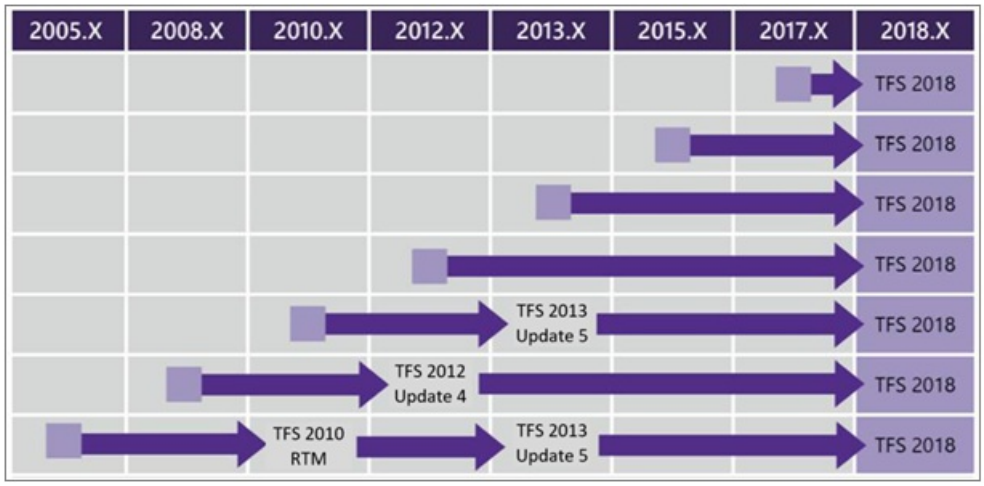
If you are accessing this page from a non-English language version, and want to see the most up-to-date content, please visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2018 Update 1. Click the button to download.

Team Foundation Server 2018

To learn more about Team Foundation Server 2018, see the Team Foundation Server Requirements and Compatibility page. Visit the visualstudio.com/downloads page to download other TFS 2018 products.

Direct upgrade to Team Foundation Server 2018 Update 1 is supported from TFS 2012 and newer. If your TFS deployment is on TFS 2010 or earlier, you need to perform some interim steps before upgrading to TFS 2018 Update 1. Please see the chart below and the TFS Install page for more information.



TFS Upgrade Matrix

NOTE

You do not need to upgrade to TFS 2018 RTM before upgrading to TFS 2018 Update 1.

Release Date: September 12, 2018

Team Foundation Server 2018 Update 1.1

Basic authorization is now enabled on the communication between the TFS and Search services to make it more secure. Any user installing or upgrading to TFS 2018 Update 1.1 will need to provide a user name / password while configuring Search (and also during Search Service setup in case of remote Search Service).

Release Date: February 20, 2018

Summary of What's New in Team Foundation Server 2018 Update 1

This is an update for Team Foundation Server 2018. It contains updates and bug fixes, including accessibility fixes and fixes for cross-site scripting (XSS) and other security vulnerabilities, since Team Foundation Server 2018 RTM. See the blog post for more information on the fixed security vulnerabilities.

Updates:

- GVFS caching support in the TFS proxy.
- Access information pertinent to test cases in your automated tests when running in the CI/CD pipeline.
- Automated tests that use TestCase as a data source can now be run using the VSTest task.

Bug Fixes:

- Code
- Work
- Build and Release
- Test
- Wiki
- Reporting
- Upgrade and Installation
- Administration

Details of What's New in TFS 2018 Update 1

GVFS caching support in the TFS proxy

The TFS proxy can serve as a Git Virtual File System (GVFS) cache. GVFS virtualizes the file system beneath your Git repository so that Git tools see what appears to be a normal repository when, in fact, the files are not actually present on disk. GVFS only downloads files as they are needed. The TFS proxy can now power these downloads, allowing you to put caches close to your distributed development teams.

Access information pertinent to test cases in your automated tests when running in the CI/CD pipeline

When running automated tests in the CI/CD pipeline using Test Plan/Test Suite or when running automated tests from the **Test** hub, crucial TCM properties can now be accessed using the TestContext object. To use this, you need Visual Studio 2017.5 or higher. This includes:

- __Tfs_TestPlanId__
- __Tfs_TestCaseId__
- __Tfs_TestPointId__
- __Tfs_TestConfigurationId__
- __Tfs_TestConfigurationName__
- __Tfs_TfsServerCollectionUrl__
- __Tfs_TeamProject__

Automated tests that use TestCase as a data source can now be run using the VSTest task

The following pre-requisites are required to use VSTest task:

1. Visual Studio 2017.6 or higher. If you are using the Test Platform Installer task to run tests using the VSTest task, make sure you pick the appropriate version of the package.

2. Create a PAT that is authorized for the scope "Work Items (full)".
3. Add a secure Build or Release variable called Test.TestCaseAccessToken with the value set to the PAT created in step 2.

> **NOTE**
> Tests that use TestCase as a data source cannot be used with the **Run Functional Tests** task.

Bug Fixes in this Release

Code

- If a repository is deleted while Search is indexing it, the job fails with GitRepositoryNotFoundException.
- During the first indexing of a TFVC repository, there may be folder drops and indexing taking a long time.
- The Search indexing patch job may fail if there are many files that fail to download.
- When a code file fails indexing and continues to fail during retries, you will indefinitely get "Lots of files rejected by Elasticsearch, failing this job" error messages.
- Customers see a permanent message of "One or more projects in your account are still being indexed, and so you are not seeing any results". With this update, the error appears a maximum of ten times for a given set of files.

Work

- No links are shown on a work item if there is a TFVC link from the work item and the TFVC artifact has been deleted.
- Identity information may not correctly synchronize to work items.
- **Allowed Values** custom fields may not work when **ALLOWEDVALUES** is defined inside a **WHEN** rule.

Build and Release

- The build summary does not show associated work items if the associated changes are empty.
- Pausing a build definition does not pause gated builds.
- The **Local path under $(build.sourcesDirectory)** unexpectedly populates when configuring the **Get Sources** step of a build definition.
- Save of a release definition gives a "service endpoint not available or not accessible" error, even when there is no change to the endpoints used in the tasks.
- In the release pipeline, rearranging the order in which parallel environments are visualized is not possible. To fix this, an option has been added in the environment panel of the editor to move an environment up/down.
- A SQL exception may occur when queuing an agent pool maintenance job to an agent with a previous version.

Test

- The **Run Timeout** test setting is not being honored.
- The **Test Category** filter does not work when running tests in distributed mode.
- The **Test Plan** filter does not show if the filter query is invalid.
- Cannot extract more than ten steps from a Test Case into a Shared Step.
- When a build completes with no test results, an incorrect message is shown, directing users to **Visual Studio Test** tasks.
- The **Go** button on the **Test Runs** page is clipped in Russian and French.
- "An error occurred while trying to load this content" error is displayed in the **My Favories** page after making a Test Plan a favorite.
- The context menu for assigning testers is missing.
- The total number of hidden tests are not accurate after deleting existing test cases on the task board.
- The count in the Test Chart is incorrect when rolling up from child test plans.
- Users cannot delete **Test Plans** and **Test Suites** when having **manage test plans / test suites** permissions or

delete test runs permissions.

- Hierarchical views of ordered test cases are not shown when publishing test results.

Wiki

- There is no error when an inserted file fails to upload due to a large size.

Reporting

- The color picker is not visible when configuring the **Chart for Work Items**.
- The warehouse jobs are running after opting out of reporting during an upgrade to TFS 2018.

Upgrade and Installation

- ArgumentNullException is thrown running the **Migrating CFD data** step when upgrading to TFS 2018.
- "Invalid length for a Base-64 char array or string" error occurs in the **Migrate to latest CoD Migrations (On prem)** step when upgrading from TFS 2017 to TFS 2018 with a large number of NuGet package versions.
- The upgrade to TFS 2018 can take a long time when configuring scheduled backups.
- The reporting server and analytics server values are not populated in the upgrade wizard.
- The upgrade to TFS 2018 causes duplicate Microsoft.TeamFoundation.Warehouse.OptimizeDatabases jobs.
- "TF50620: The Team Foundation identity scope does not exist" error when upgrading TFS.
- The Configuration Wizard fails when the machine names includes Unicode Extension B characters.
- "1 error(s) occurred while executing upd_AnalyticsToDev15M108PreSchema.sql script" error when upgrading to TFS 2018.
- "Execution Timeout Expired. The timeout period elapsed prior to completion of the operation or the server is not responding." error when upgrading to TFS 2018.
- Configuring Search fails when a team project collection is offline.
- Upgrades to TFS 2018 take a long time when there are many NuGet packages.
- The **prc_CopyContainerItemFiles** job may fail due to exceeding the timeout when upgrading to TFS 2018.

Administration

- The **Team Foundation Server Activity Logging Administration** job fails on a team project collection after timing out.
- The certificate is invalid when using Git clients with OpenSSL if using a self-signed SSL certificate generated by TFS.
- "TF246017: Team Foundation Server could not connect to the database." error occurs when trying to unconfigure reporting in the TFS Admin Console when the warehouse database is not reachable.
- "Could not find validator for token" error occurs when configuring an agent after changes to a TFS hostname, such as changing the public URL or moving the TFS databases.
- tfsconfig offlinedetach does not warn if the collection appears offline.
- Unable to save a new notification when using one of the notification templates, such as **work item is created** if the user locale is not in English.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2018 Release Notes

10/3/2018 • 50 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

If you are accessing this page from a non-English language version, and want to see the most up-to-date content, please visit this Release Notes page in English. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2018. Click the button to download.

Team Foundation Server 2018

To learn more about Team Foundation Server 2018, see the Team Foundation Server Requirements and Compatibility page. Visit the visualstudio.com/downloads page to download other TFS 2018 products.

Please see the TFS Install page for more information.

Release Date: November 15, 2017

Summary of What's New in TFS 2018

We have added a lot of new value to Team Foundation Server 2018. Some of the highlights include:

- We have improved the Project Creation Wizard and Process Template Manager on the web.
- You can now customize the work item form header.
- We optimized the mobile work item form.
- We added support for Git forks.
- You can manage massive Git repositories with GVFS.
- You can view, filter, delete, and set the security of Git tags.
- We added file minimap, bracket matching, and toggle white space to web code editing.
- We made many improvements to pull requests.
- You have a new improved Wiki experience.
- We have added support for Maven packages.
- You can import and export, and pause build definitions.
- The new Release Definition Editor has opt-in by default.
- You can deploy with virtual machine deployments.
- We improved exploratory testing traceability.
- We added test batching.
- You can now view a chart widget for test plans and test suites.

What's New in TFS 2018 video

https://channel9.msdn.com/Shows/Visual-Studio-Toolbox/Whats-New-in-TFS-2018/player?autoplay=false

XAML Build

We had originally listed XAML build as removed from TFS 2018 RTW and Update 1. However, that resulted in too many customers being unable to upgrade or having to contact support to re-enable it after the upgrade completed. In TFS 2018 Update 2, XAML build is enabled but has been deprecated. This means there is no further investment in XAML Build, and Microsoft Test Manager (MTM) no longer supports using XAML builds. We highly recommend converting to one of the newer build definition formats. You may continue to connect your XAML controllers and run XAML builds with TFS 2018 Update 2. More info

Features Removed in TFS 2018 RTW

- Support for Lab Center and automated testing flows in Microsoft Test Manager has been removed.
- We discontinued TFS Extension for SharePoint.
- We removed the Team Room feature.

Details of What's New in TFS 2018

Work Item Tracking

Project Creation Wizard on the web

We have improved the experience for creating a Team Project from web access. It now includes most of the features available when you create a Team Project in the Visual Studio client. The benefit of using the web interface is that you do not need a matching Visual Studio version. The difference of using Visual Studio or the web version is that the web version does not provision your reports in SSRS. If you used the web version of the Team Project creation, you can run the `tfsconfig` command on the Application Tier to provision or update the SSRS reports. See details in add project reports.

Process Template Manager on the web

With TFS 2018, you can use web access to upload your process templates. The web interface is a much easier experience because you do not have to install the correct version of Visual Studio to interact with your process templates. Visual Studio 2017 Update 4 and earlier will still show the **Process Template Manager** dialog, although we recommend using the web interface. Visual Studio 2017 Update 5 and later will redirect you to the web automatically.

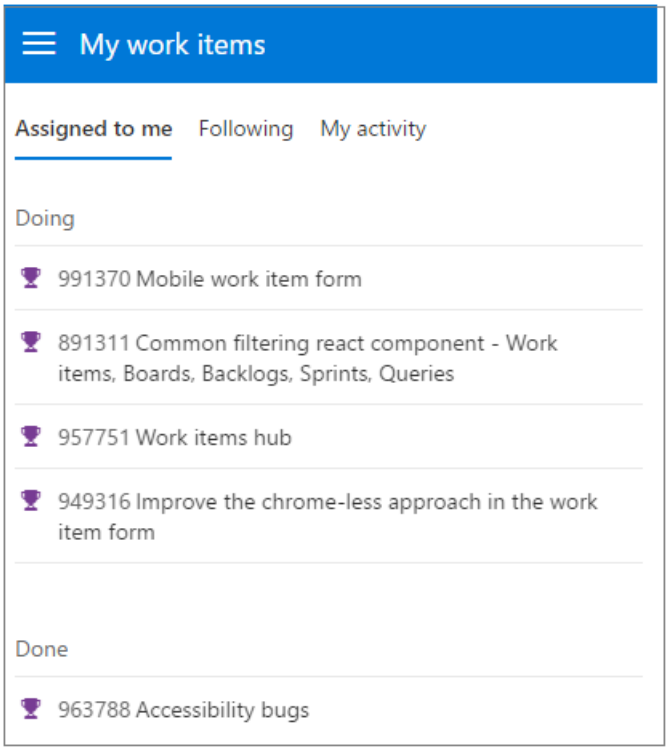
Customize the work item form header

You can now customize the work item form header area by replacing the existing controls, or hiding controls, that are not relevant to your process. This will enable replacing Area path with a custom Team field, hiding Iteration if your teams are more Kanban focused, and replacing Reason with a custom field. The State field cannot be hidden or replaced.

> **TIP**
>
> See the documentation for WebLayout and Control elements for more information.

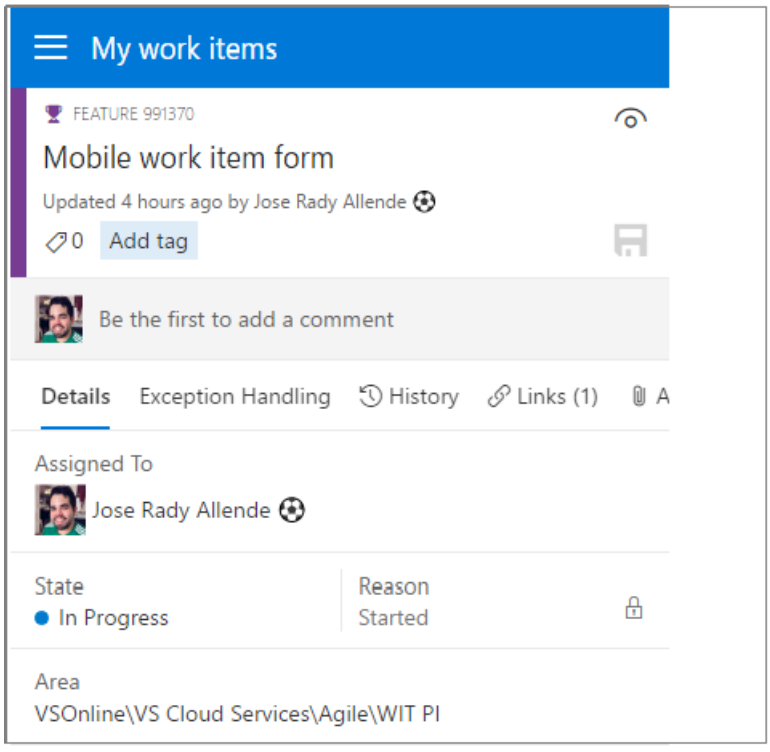
Mobile work item form

We have a full end-to-end experience that includes an optimized look and feel for work items *(Figure 1)*. It provides an easy way to interact with items that are assigned to you, are following, have visited, or edited recently from your phone.



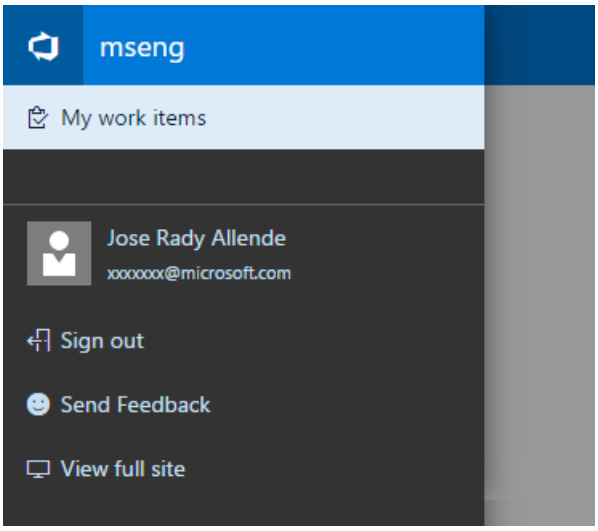
(Figure 1) Mobile work item query

Along with the good looks, this experience supports optimized controls for all field types *(Figure 2)*.



(Figure 2) Mobile work item form

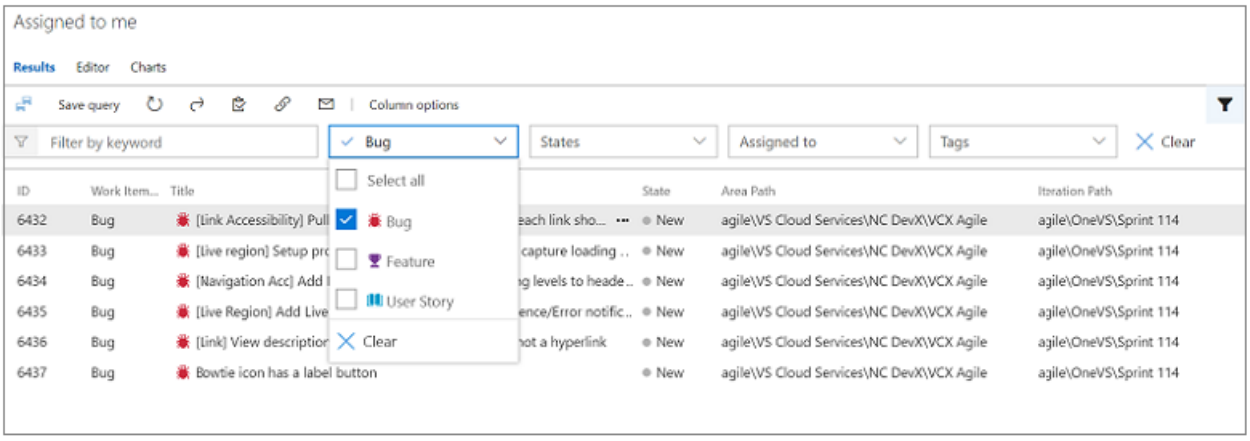
With the new mobile navigation *(Figure 3)*, users can reach any other mobile-ready parts of TFS and get back to the full desktop site in case they need to interact with other hubs.



(Figure 3) Mobile navigation

Filtering on backlogs, Kanban boards, sprints, and queries

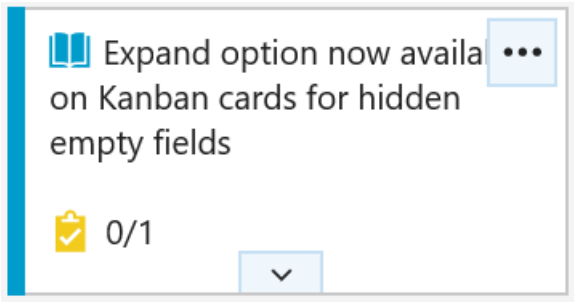
All of our work item tracking grid experiences (queries, backlogs, Kanban boards, sprints backlogs, and test case management) now make use of our common, consistent filtering component *(Figure 4)*. Beyond applying a keyword filter across displayed columns and selecting tags, you can also filter on work item types, states, and assigned to, in order to quickly get to the work items you are looking for.



(Figure 4) Filtering on queries

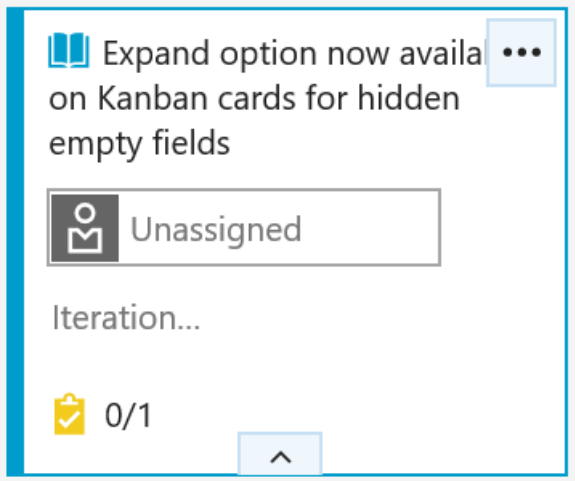
Expand to show empty fields on a Kanban card

Today, you have the option to add additional fields to a card and then **hide empty fields** *(Figure 5)* in board settings to remove unnecessary clutter from the board. The drawback to this feature was that once an empty field was hidden, the only way to update the field was to open the work item form. With the newly available expand option on Kanban cards, you can now benefit from hiding empty fields across the board, but still have single click access to update a particular field on a card. Simply hover over the card and look for the down chevron at the bottom of the card to update the hidden field.



(Figure 5) Hidden field on Kanban card

Click the down chevron at the bottom of the card to update the field *(Figure 6)*.



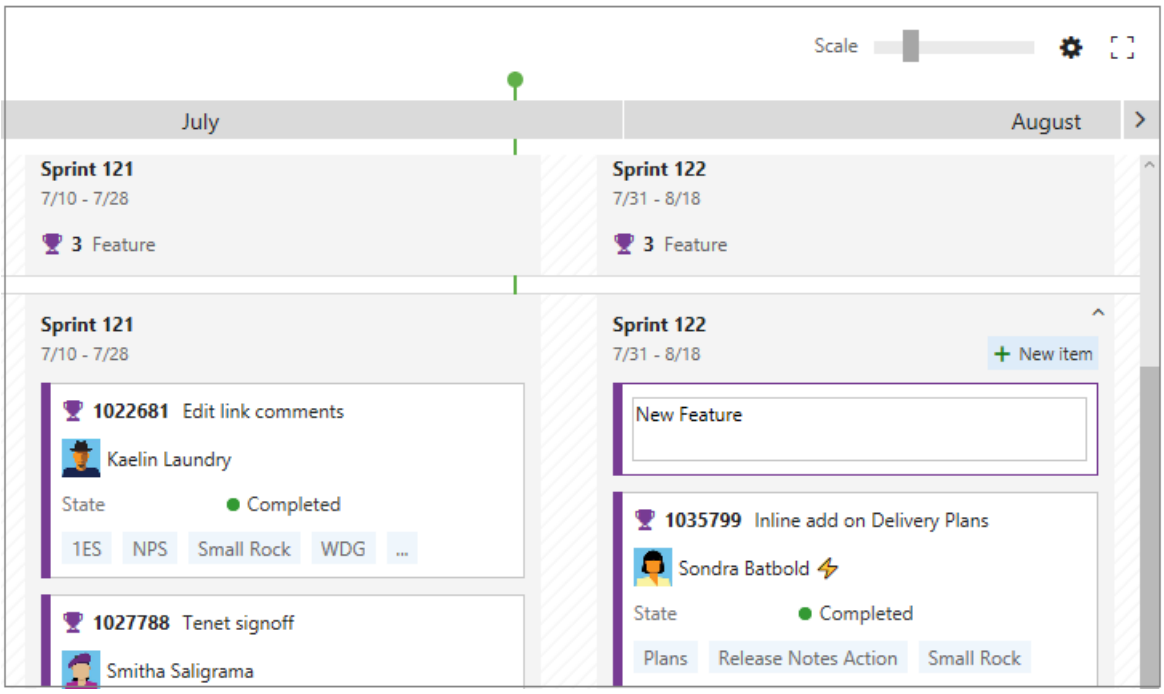
(Figure 6) Update hidden field on Kanban card

Extensions block work item save

Work item form custom controls, groups, and pages can now block work item save to validate data and ensure the user fills out any required information before saving the work item form.

Inline add on Delivery Plans

New feature ideas can arrive at any moment, so we have made it easier to add new features directly to your **Delivery Plans** *(Figure 7)*. Simply click the **New item** button available on hover, enter a name, and hit enter. A new feature is created with the area path and iteration path you would expect.

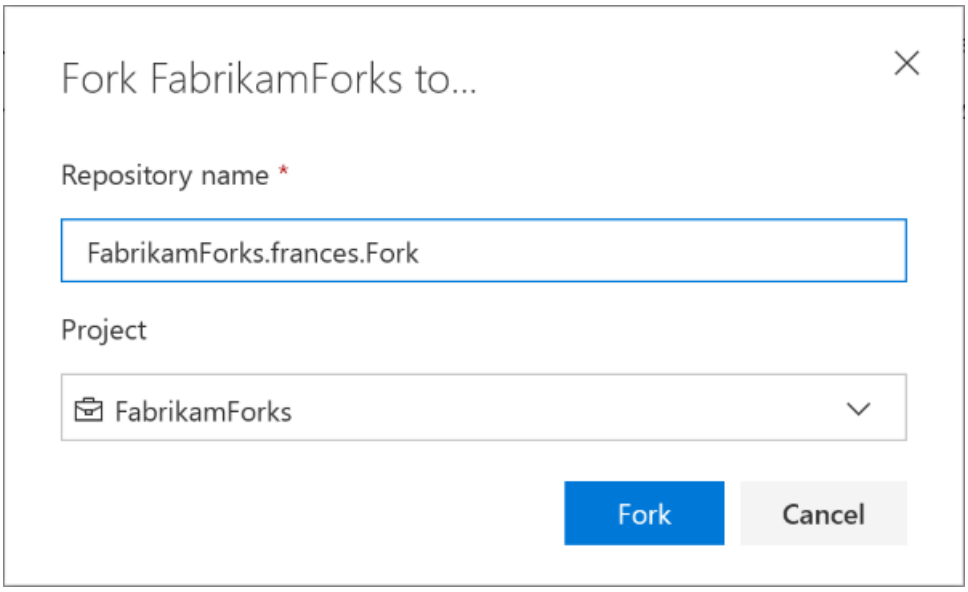


(Figure 7) Inline add on delivery plans

Version Control

Forks

TFS 2018 adds support for Git forks *(Figure 8)*. A fork is a server-side copy of a repository. Using forks, you can allow a broad range of people to contribute to your repository without giving them direct commit access. Instead, they commit their work to their own fork of the repository. This gives you the opportunity to review their changes in a pull request before accepting those changes into the central repository.



(Figure 8) Git forks

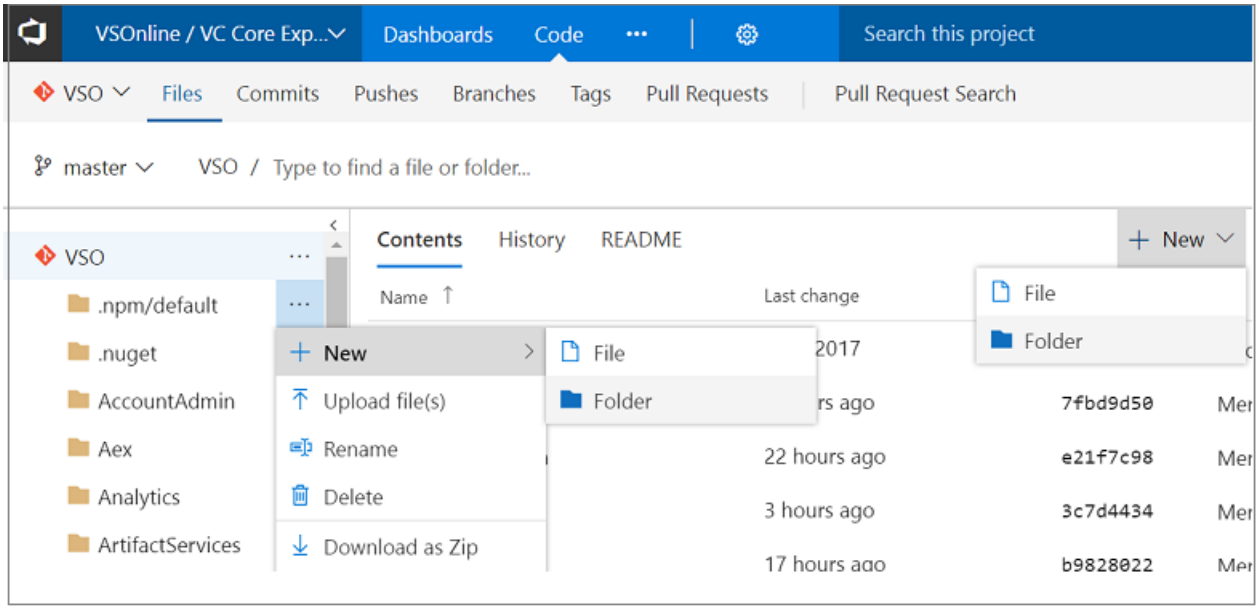
GVFS

Git Virtual File System (GVFS) is now supported. GVFS allows Git repositories to scale to millions of files by virtualizing and optimizing how Git operates on the filesystem.

Create a folder in a repository using web

You can now create folders through the web in your Git and TFVC repositories *(Figure 9)*. This replaces the Folder Management extension, which will now undergo the process of deprecation.

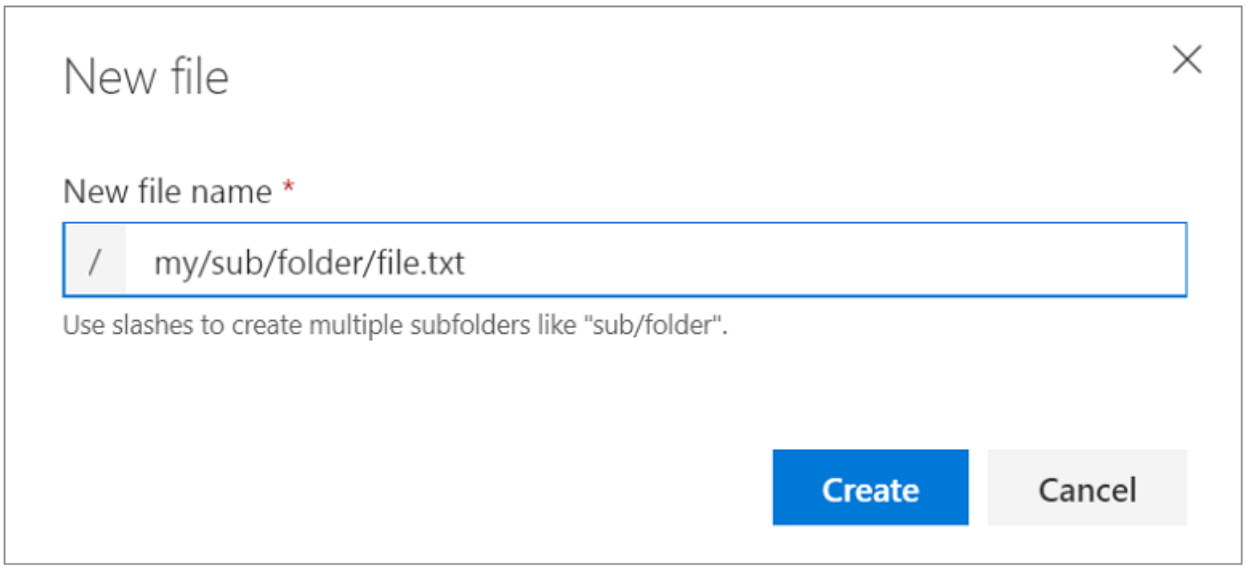
To create a folder, click **New > Folder** in either the command bar or context menu:



(Figure 9) New folder option

For TFVC, you will specify a folder name and then check it in. For Git, because empty folders are not permitted, you will also have to specify a file name, optionally edit the file, then commit it.

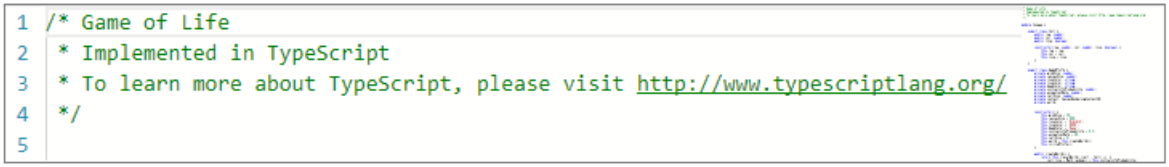
Additionally, for Git, The **New file** dialog *(Figure 10)* has been enhanced to accept slashes to create subfolders.



(Figure 10) New file dialog

File minimap

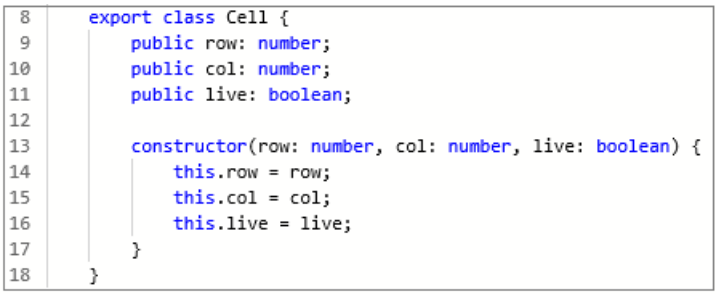
You can now view a minimap of a file as you view or edit to give you a quick overview of the code *(Figure 11)*. To turn on the minimap, open the **Command Palette** (F1 or right-click) and select **Toggle Minimap**.



(Figure 11) File minimap

Bracket matching

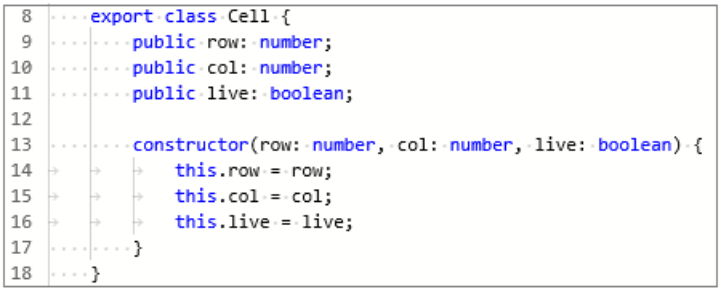
When editing or viewing a file, there are now guidelines on the left side to make it easy to match your brackets *(Figure 12)*.



(Figure 12) Bracket matching

Toggle white space

You can now toggle white space on and off when viewing or editing a file. We are still developing a feature that will allow you to toggle white space when diff'ing. To view white space *(Figure 13)*, open the **Command Palette** (F1 or right-click) and select **Toggle White Space**, which allows you to differentiate between spaces and tabs.



(Figure 13) Toggle white space

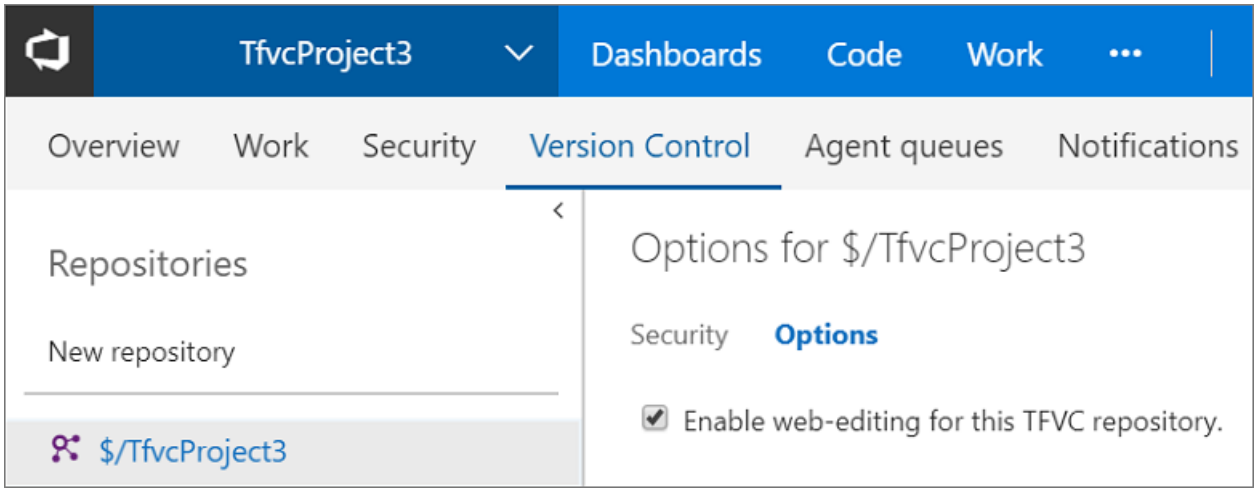
Setting to turn off web editing for TFVC repos

Teams that use TFVC often use check-in policies in Visual Studio to ensure code quality. However, because check-in policies are enforced on the client, code that is edited on the web is not subjected to the same policies.

Several people have asked for a way to disable web-editing to protect against changes that bypass check-in policies. We have enabled a way for you to turn off web-editing (adding, deleting, renaming, and editing) for TFVC on a project/repository basis.

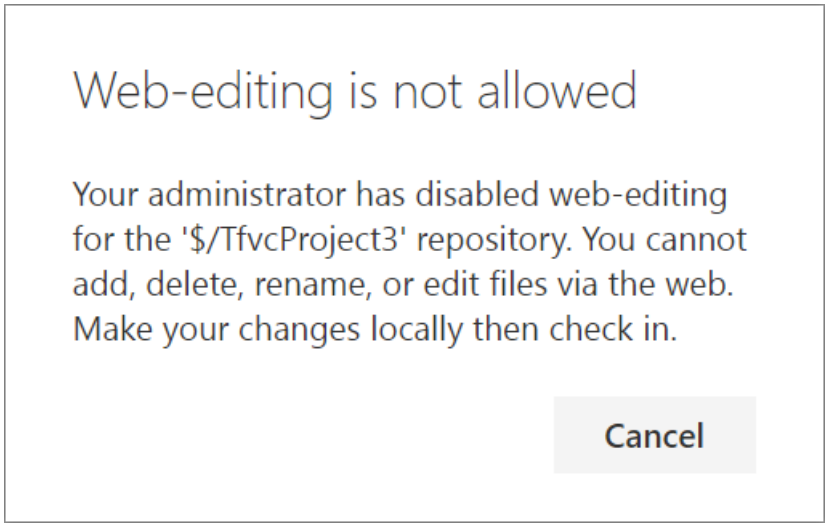
To disallow web-editing from the **Files** page, go to **Settings** then **Version Control** *(Figure 14)*. Click on the TFVC repo in the tree, navigate to the Options pivot, and uncheck **Enable web-editing for this TFVC repository**. By default, web-editing is enabled.

> **NOTE**
> Editing the README from the **Project Overview page** is unaffected.



(Figure 14) Turn off web editing

If you attempt a web-edit in a project with web-editing disabled, you are notified that web-editing is not allowed *(Figure 15)*.



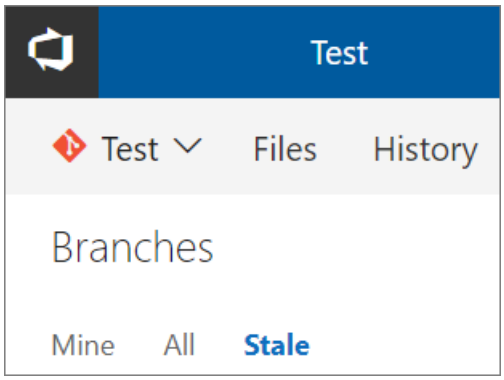
(Figure 15) Web editing not allowed dialog

TIP

This has been developed based on a related suggestion.

Identify stale branches

Keeping your repository clean by deleting branches you no longer need enables teams to find branches they care about and set favorites at the right granularity. However, if you have lots of branches in your repo, it can be hard to figure out which are inactive and can be deleted. We have now made it easier to identify "stale" branches (branches that point to commits older than 3 months). To see your stale branches, go to the **Stale** pivot on the **Branches** page *(Figure 16)*.

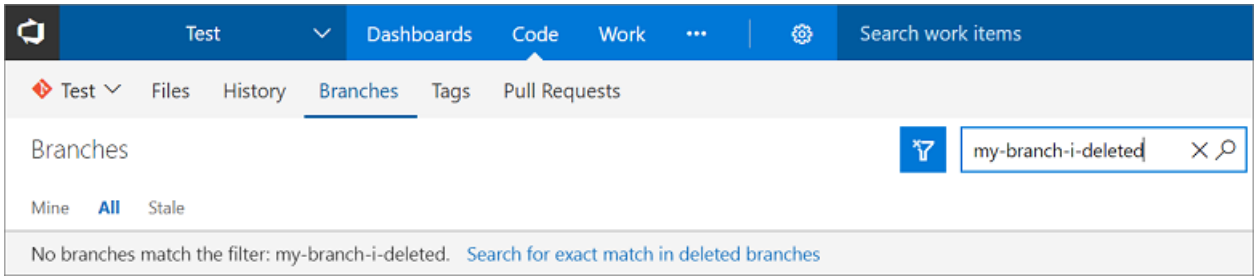


(Figure 16) Stale branches

Search for a deleted branch and re-create it

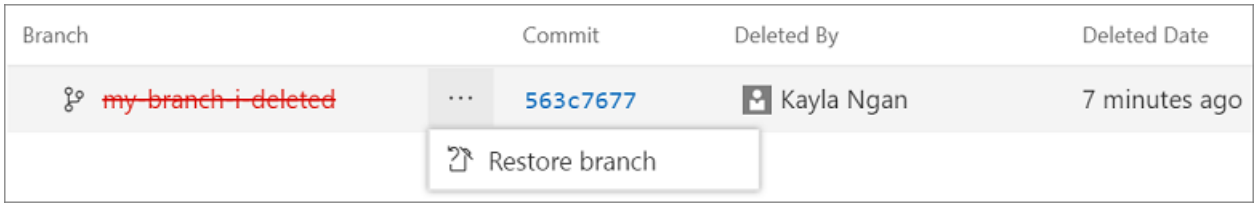
When you accidentally delete a branch from the server, it can be difficult to figure out what happened to it. Now you can search for a deleted branch, see who deleted it and when, and re-create it if you wish.

To search for a deleted branch, enter the full branch name into the branch search box. It will return any existing branches that match that text. You will also see an option to search for an exact match in the list of deleted branches. Click the link to search deleted branches *(Figure 17)*.



(Figure 17) Search for deleted branches

If a match is found, you will see who deleted it and when. You can also restore the branch *(Figure 18)*.



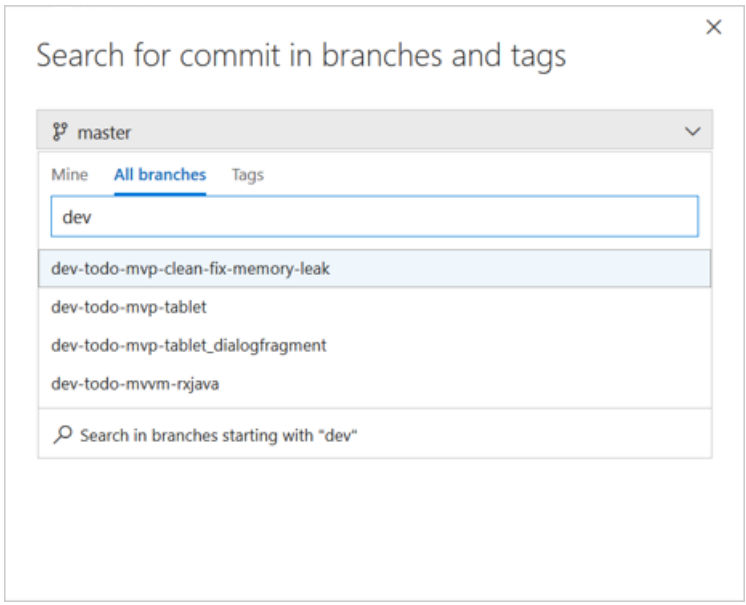
(Figure 18) Restore deleted branches

Restoring the branch will re-create it at the commit to which is last pointed. However, it will not restore policies and permissions.

Search for a commit in branches starting with a prefix

If you have branch structure in a hierarchical format where all branches are prefixed with a text, then this feature

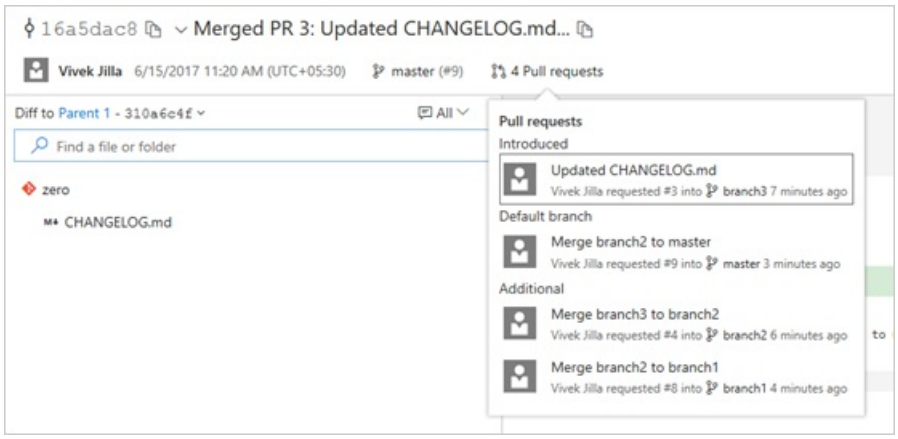
will help you to find a commit in all the branches starting with that prefix text. For example, if you want to see whether a commit made its way to all branches that are prefixed with "dev" then simply type "dev" in the search box and select **Search in branches starting with "dev"** *(Figure 19)*.



(Figure 19) Search for a commit

Richer pull request callout on commit details page

The pull request callout on the commit details page shows more relevant information to better diagnose *(Figure 20)*. Now we also show the **first pull request** that introduced the commit to any branch and the **pull request associated with the default branch** in the callout.

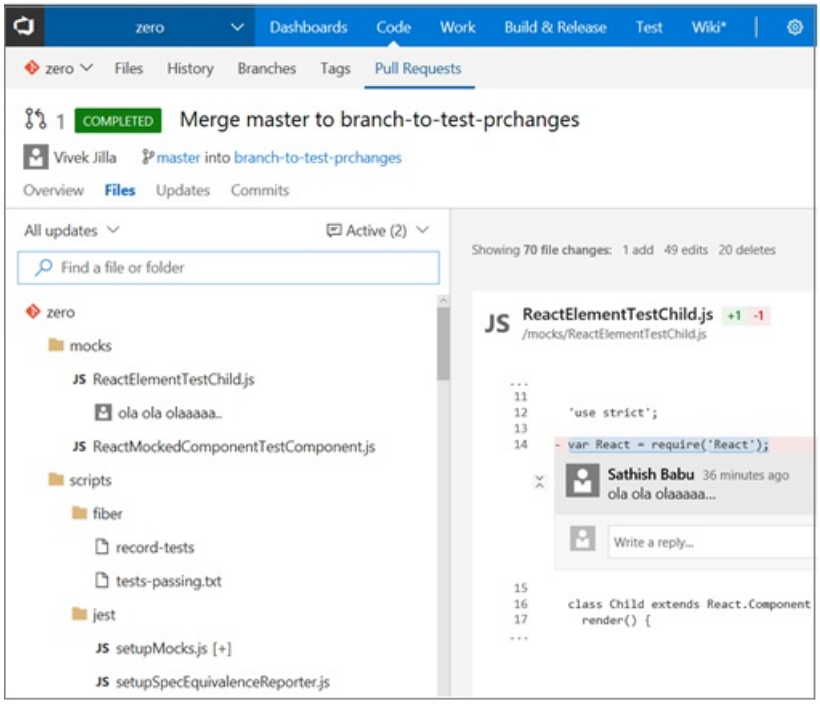


(Figure 20) Pull request callout

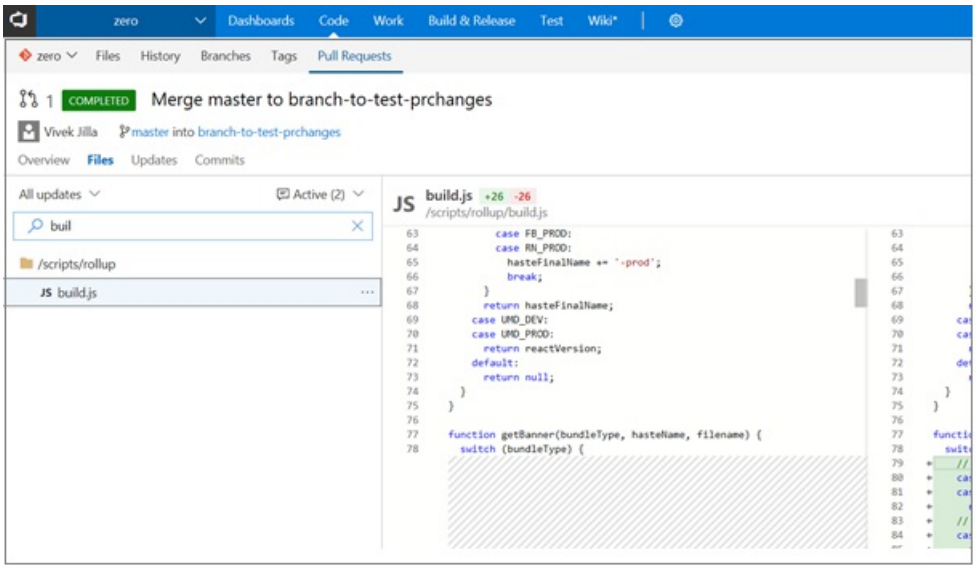
Filter tree view in Code

Now you do not need to scroll through all the files that a commit may have modified to just get to your files. The tree view on commit details, pull requests, shelveset details, and changeset details page now supports file and folder filtering. This is a smart filter that shows child files of a folder when you filter by folder name and shows a collapsed tree view of a file to display the file hierarchy when you filter by file name.

Find a file or folder filter on commit tree *(Figure 21)* and *(Figure 22)*:



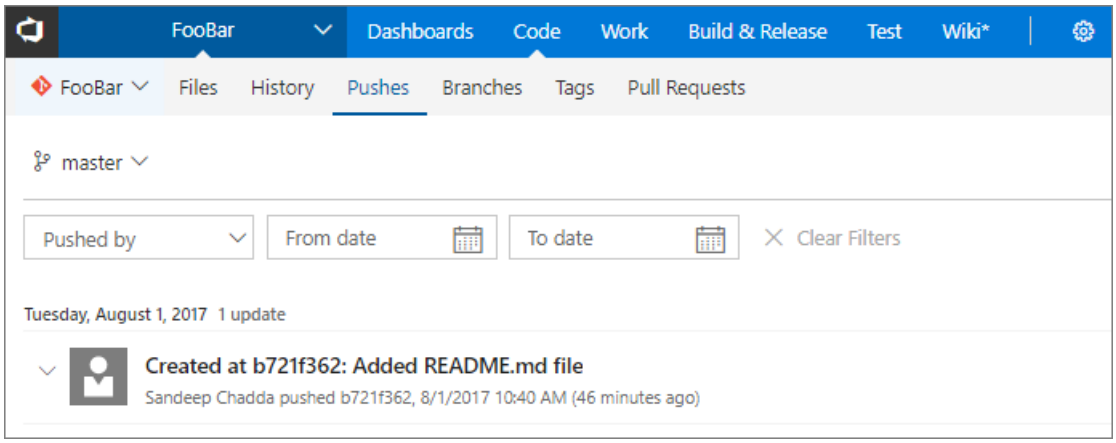
(Figure 21) Find a file or folder



(Figure 22) Filtered view on commit tree

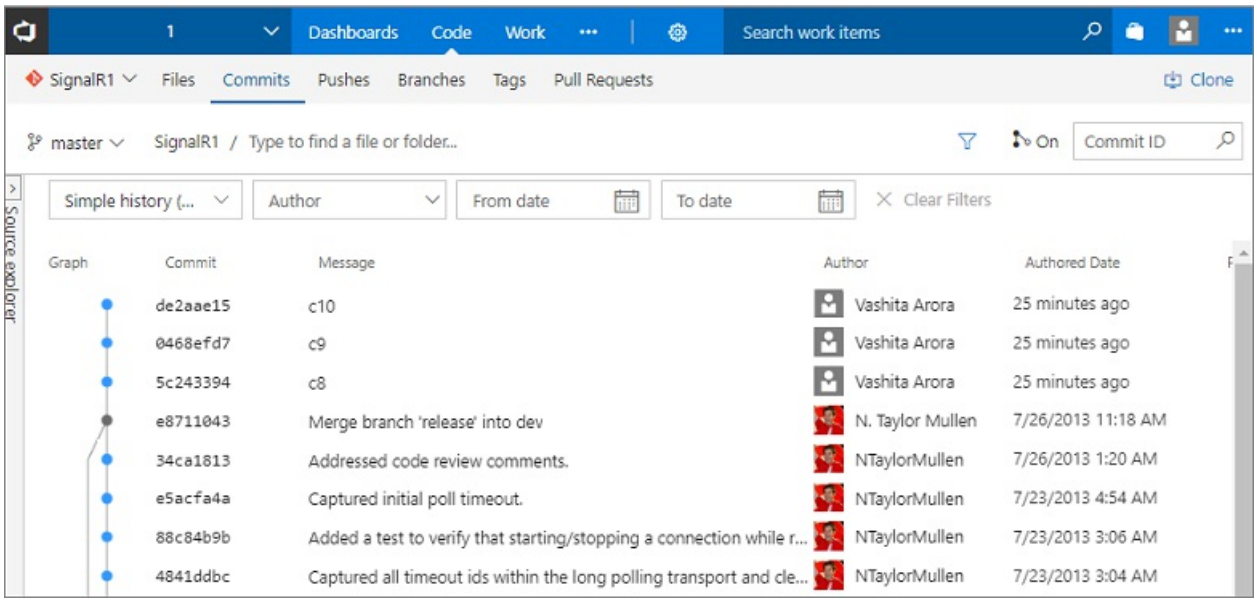
Branch updates page is now Pushes

The **Branch Updates** page has tremendous value. However, it was hidden as a pivot under the **History** hub. Now the branch updates page are visible as a hub called **Pushes** *(Figure 23)* under **Code**, alongside **Commits**, **Branches**, **Tags**, and **Pull Requests**. The new URL for the pushes page is: `\<tfsserverurl\>/\<projectname\>/_git/\<reponame\>/pushes`. The old URLs will continue to function.



(Figure 23) Pushes page

At the same time, the **History** hub is now renamed to **Commits** *(Figure 24)* since the hub only shows commits. We received feedback that people were finding it difficult to troubleshoot commit related issues because the commit list view only showed detailed time on-hover. Now the commit list view across your instance will show date and time in dd/mm/yy hh:mm format. The new URL for commits page is: `\<tfsserverurl\>/\<projectname\>/_git/\<reponame\>/commits`. The old URLs will continue to function.



(Figure 24) Commits page

Retain filename when moving from Files to Commits

We heard feedback from people that when they filter the directory to a particular file in the **Files** pivot of the **Code** hub and later flip to the **History** pivot, the filename does not persist if the commit changed more than 1,000 files. This resulted in people needing to load more files and filter content to find the file, which impacted productivity. Developers normally work in the same directory and want to persist to the directories they work in as they trace changes. Now, we persist the filename as you move between **Code** hub pivots regardless of the number of files changed in a commit. This means that you do not have to click on **Load More** to find the file you want.

View Git tags

You can view all the tags in your repository on the **Tags** page *(Figure 25)*. If you manage all your tags as releases, then a user can visit the tags page to get a bird's-eye view of all the product releases.

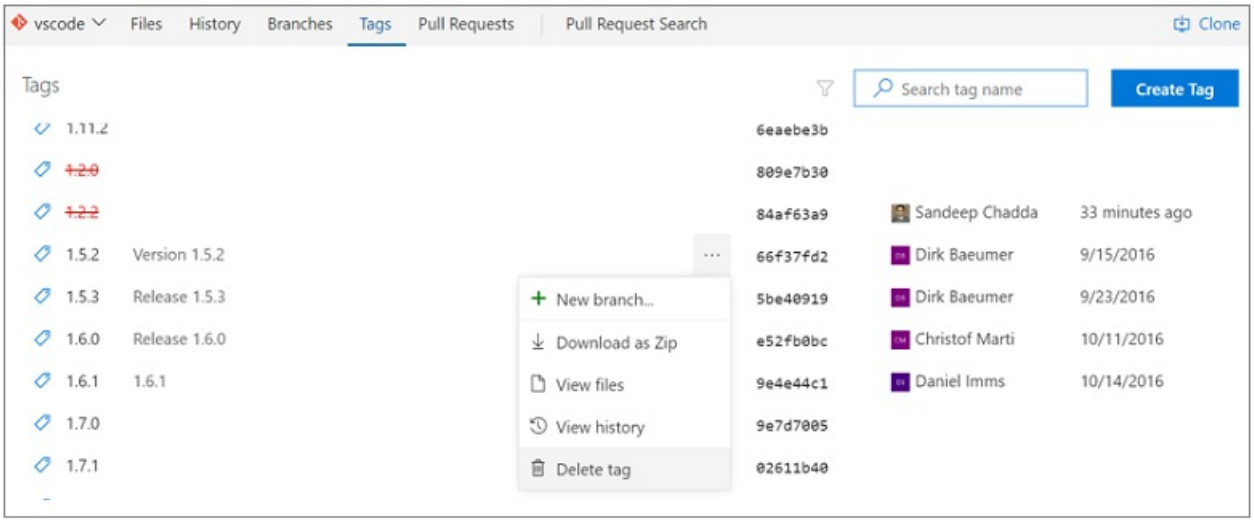
(Figure 25) View Git tags

You can easily differentiate between a lightweight and an annotated tag. Annotated tags show the tagger and the creation date alongside the associated commit, while lightweight tags only show the commit information.

Delete Git tags

There can be times when you want to delete a tag from your remote repo. It could be due to a typo in the tag name, or you might have tagged the wrong commit. You can easily delete tags from the web UI by clicking the context menu of a tag on the **Tags** page and selecting **Delete tag** *(Figure 26)*.

> **WARNING**
> Deleting tags on remote repos should be exercised with caution.

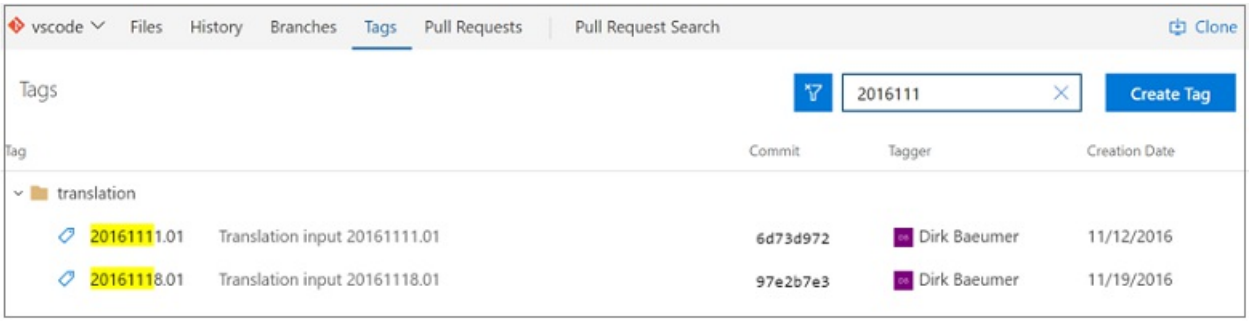


(Figure 26) Delete Git tags

Filtering Git tags

For old repositories, the number of tags can grow significantly with time; there can also be repositories that have tags created in hierarchies, which can make finding tags difficult.

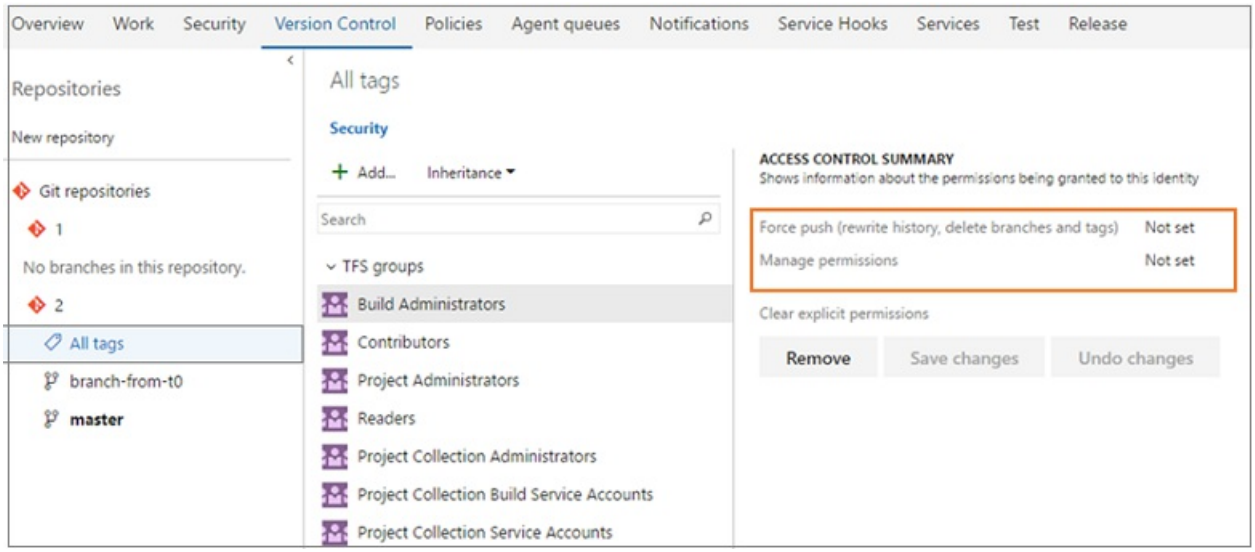
If you are unable to find the tag that you were looking for on the **Tags** page, then you can simply search for the tag name using the filter on top of the **Tags** page *(Figure 27)*.



(Figure 27) Filter Git tags

Git tags security

Now you can grant granular permissions to users of the repo to manage tags. You can give users the permission to delete tags or manage tags from this interface *(Figure 28)*.



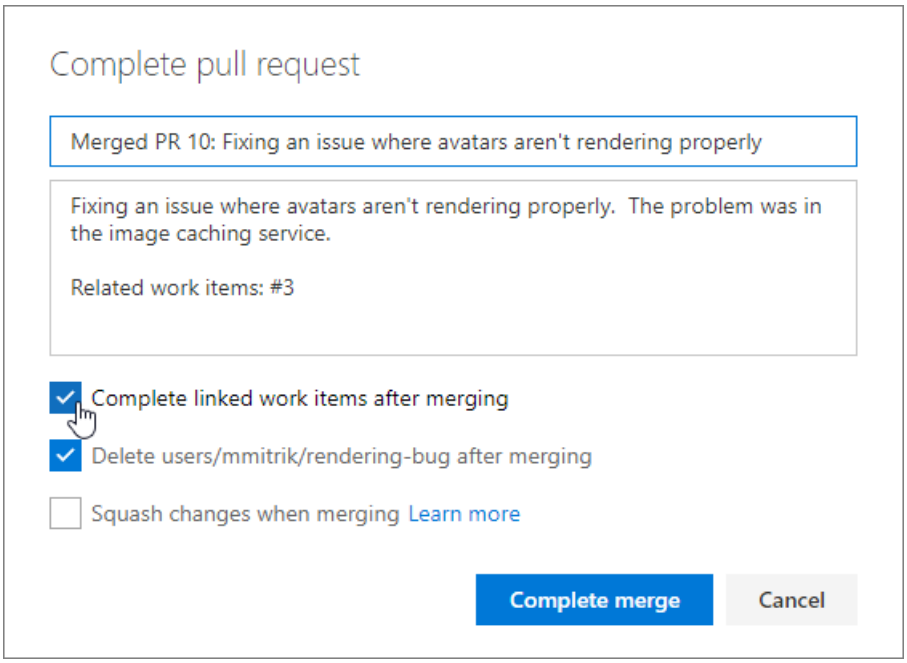
(Figure 28) Git tag security

TIP

Read more about Git tags at the Microsoft DevOps blog.

Automatically complete work items when completing pull requests

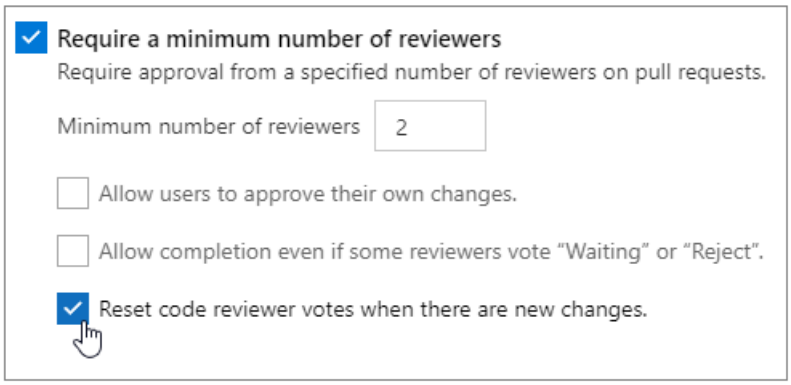
If you are linking work items to your pull requests, keeping everything up to date just got simpler. Now, when you complete a PR, you will have the option to automatically complete the linked work items after the PR has been merged successfully *(Figure 29)*. If you are using policies and set pull requests to auto-complete, you will see the same option. No more remembering to revisit work items to update the state once the pull request has completed. This is done for you automatically.



(Figure 29) Complete linked work items

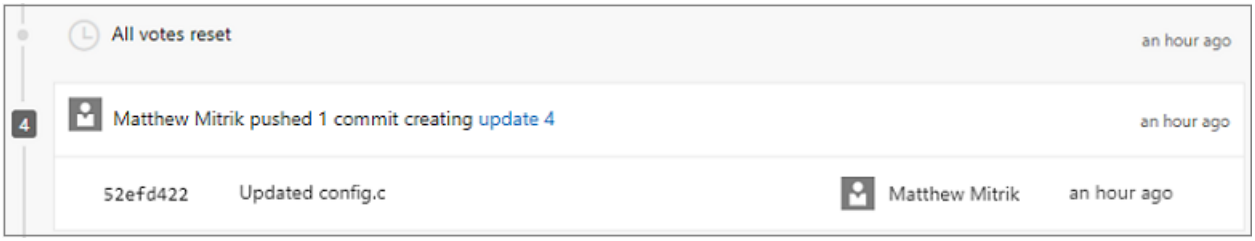
Reset votes on push/new iteration

Teams opting for a more strict approval workflow in PRs can now opt-in to reset votes when new changes are pushed *(Figure 30)*. The new setting is an option under the policy to **Require a minimum number of reviewers**.



(Figure 30) Reset votes setting

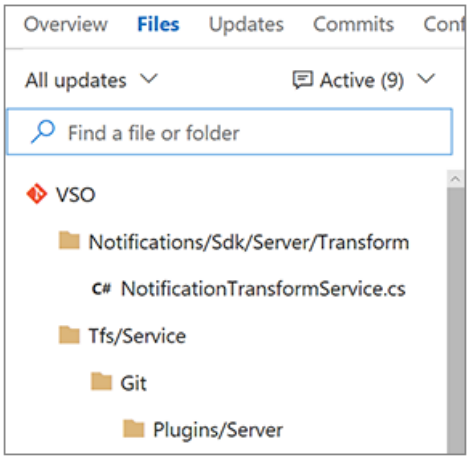
When set, this option will cause all votes from all reviewers to be reset any time the source branch of the PR is updated. The PR timeline will record an entry any time the votes are reset as a result of this option *(Figure 31)*.



(Figure 31) Reset votes in timeline

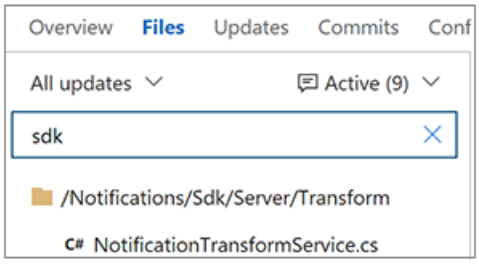
Filter pull request tree by file name

Finding a specific file in a pull request is easier than ever. The new filter box in the **Files** view lets users filter down the list of files in the tree view *(Figure 32)*.



(Figure 32) Find file or folder in pull request

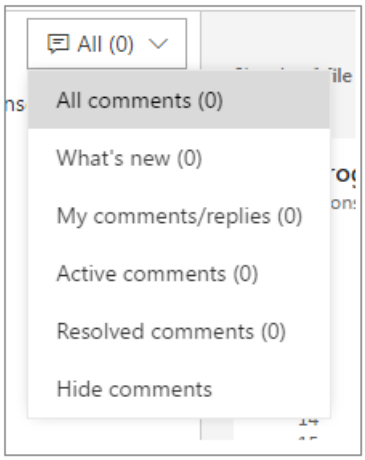
The filter matches any part of the path of the files in the pull request, so you can search by folder names, partial paths, file names, or extensions *(Figure 33)*.



(Figure 33) Find results

More pull request comments filtering options

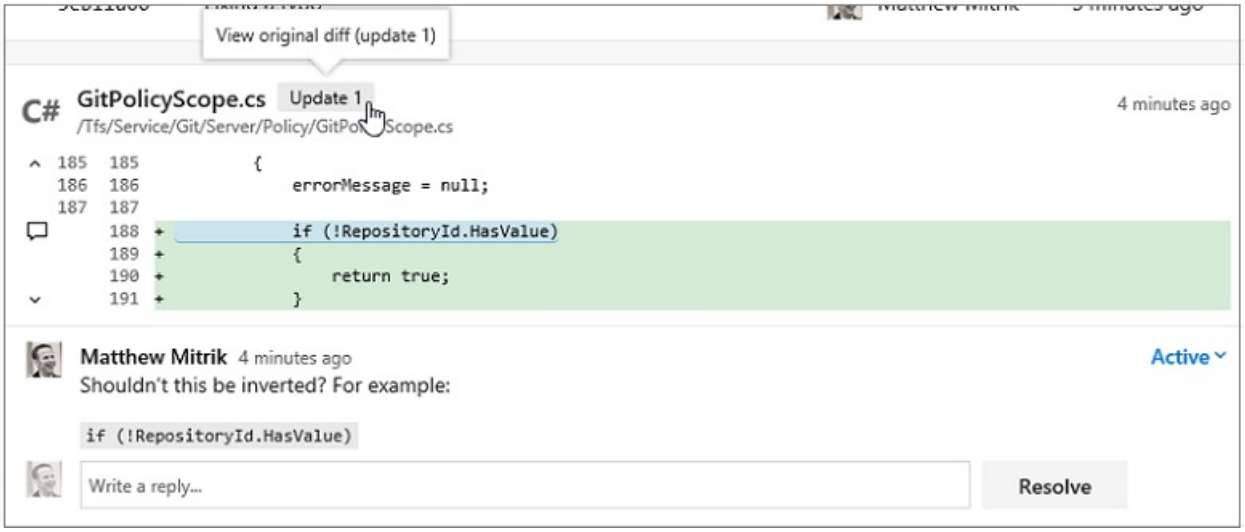
Comments in both the pull request overview and the files view now have the same options *(Figure 34)*. You can also filter to see only the discussions you participated in.



(Figure 34) PR comment filtering

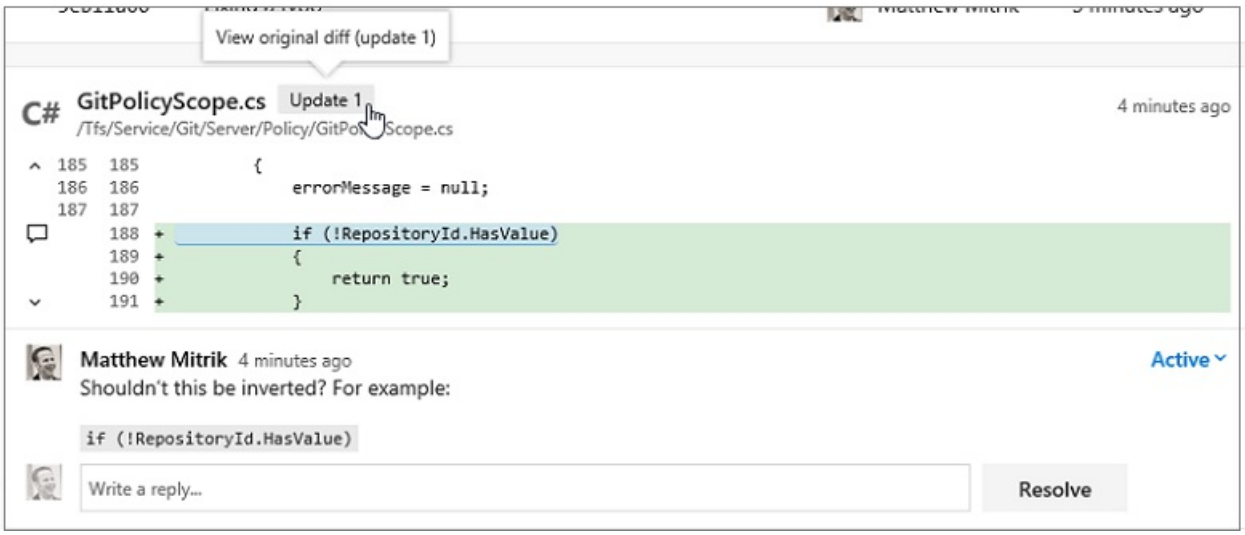
View original diff for code comments in pull request details

Sometimes, it is hard to make sense out of a PR comment after the code it is referencing has changed (many times, when a requested change has been made) *(Figure 35)*.



(Figure 35) View original diff

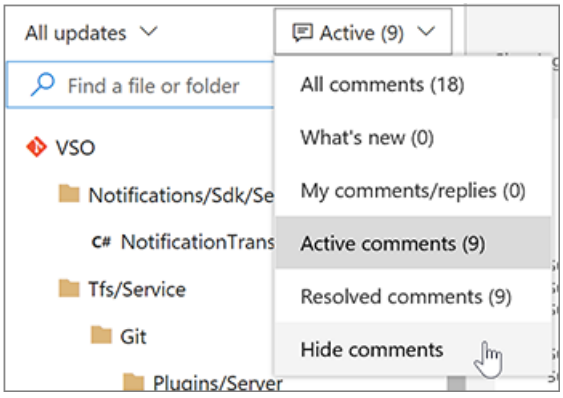
When this happens, you will now see a badge with an update number that you can click to see what the code looked like at the time the comment was originally created *(Figure 36)*.



(Figure 36) Update badge

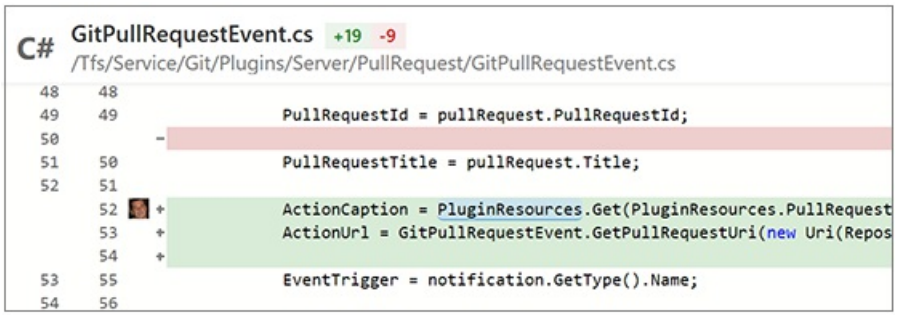
Collapsible pull request comments

Reviewing code is a critical part of the pull request experience, so we have added new features to make it easier for reviewers to focus on the code. Code reviewers can easily hide comments to get them out of the way when reviewing new code for the first time *(Figure 37)*.



(Figure 37) Hide comments

Hiding comments *(Figure 38)* hides them from the tree view and collapses the comment threads in the file view:



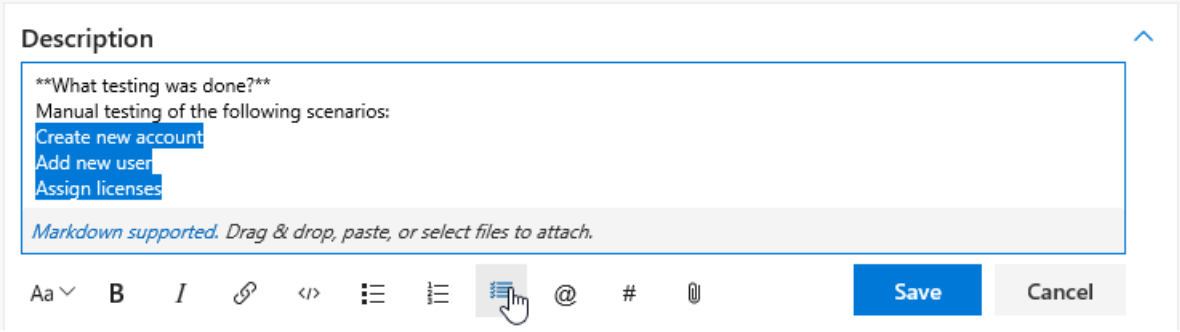
(Figure 38) Collapsed comments

When comments are collapsed, they can be expanded easily by clicking the icon in the margin, and then collapsed again with another click. **Tooltips** *(Figure 39)* make it easy to peek at a comment without seeing the entire thread.

(Figure 39) Collapsed comment tooltip

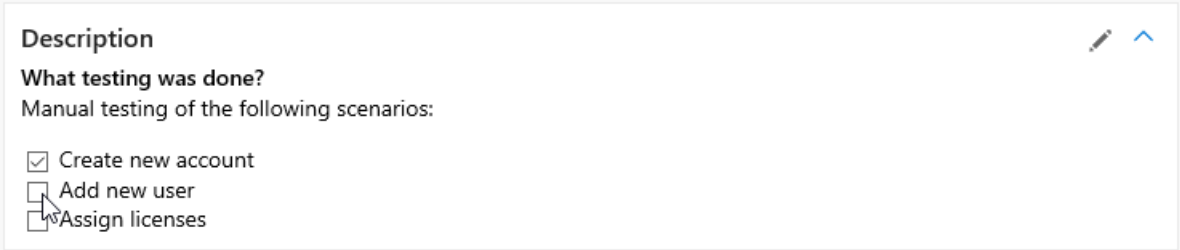
Task lists in pull request descriptions and comments

When preparing a PR or commenting you sometimes have a short list of things that you want to track but then end up editing the text or adding multiple comments. Lightweight task lists are a great way to track progress on a list of to-dos as either a PR creator or reviewer in the description or a single, consolidated comment. Click on the Markdown toolbar to get started or apply the format to selected text *(Figure 40)*.



(Figure 40) Task list toolbar

Once you have added a task list *(Figure 41)*, you can simply check the boxes to mark items as completed. These are expressed and stored within the comment as `[ ]` and `[x]` in Markdown. See Markdown guidance for more information.



(Figure 41) Task list

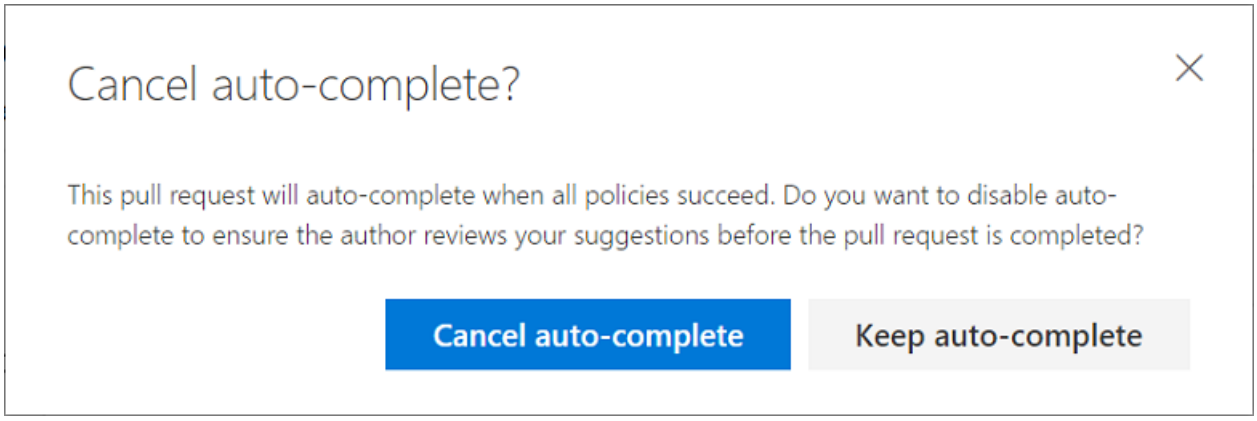
Ability to "Like" comments in pull requests

Show your support for a PR comment with a single click on the **like** button *(Figure 42)*. You can see the list of all people that liked the comment by hovering over the button.

(Figure 42) Like pull request comments

Improved workflow when approving with suggestions

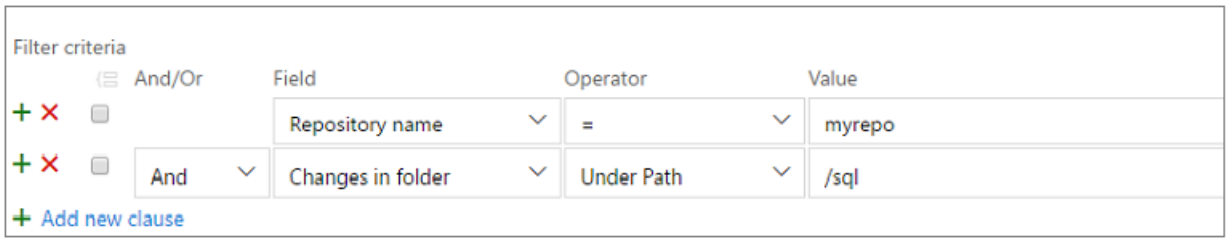
Using the **auto-complete** option *(Figure 43)* with pull requests is a great way to improve your productivity, but it should not cut short any active discussions with code reviewers. To better facilitate those discussions, the **Approve with suggestions** vote will now prompt when a pull request is set to complete automatically. The user will have the option to cancel the auto-complete so that their feedback can be read, or keep the auto-complete set and allow the pull request to be completed automatically when all policies are fulfilled.



(Figure 43) Cancel auto-complete dialog

Path filtering support for Git notifications

Instead of getting notifications for all folders in a repo, you can now choose to get notified when team members create pull requests or push code in only the folders that you care about. When creating custom Git push or Git pull requests email notification subscriptions, you will see a new option to filter these notifications by folder path *(Figure 44)*.



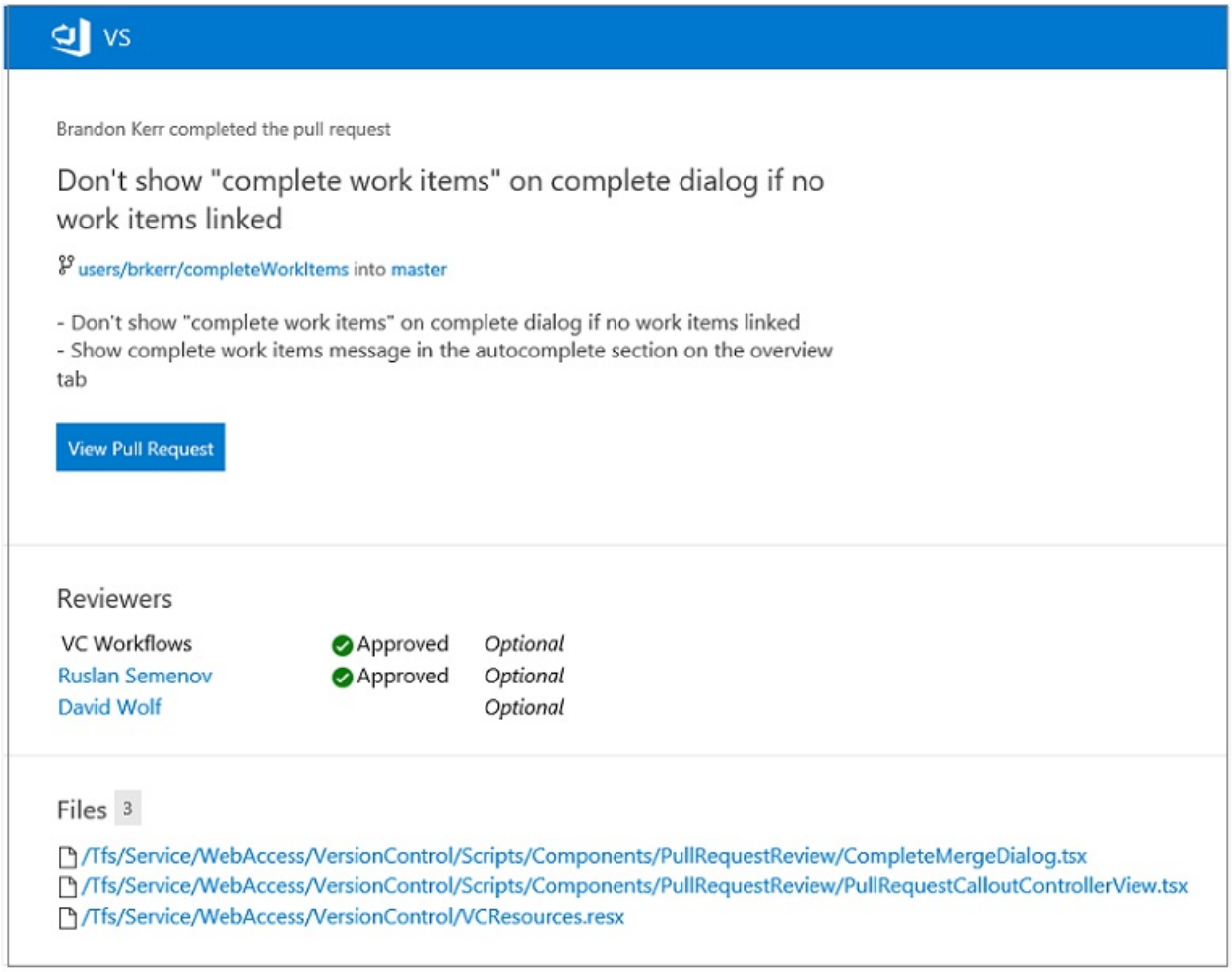
(Figure 44) Path filtering for notifications

Updated email templates for pull request workflows

Pull request email alerts have been refreshed to make them clear, concise, and actionable *(Figure 45)*. The subject

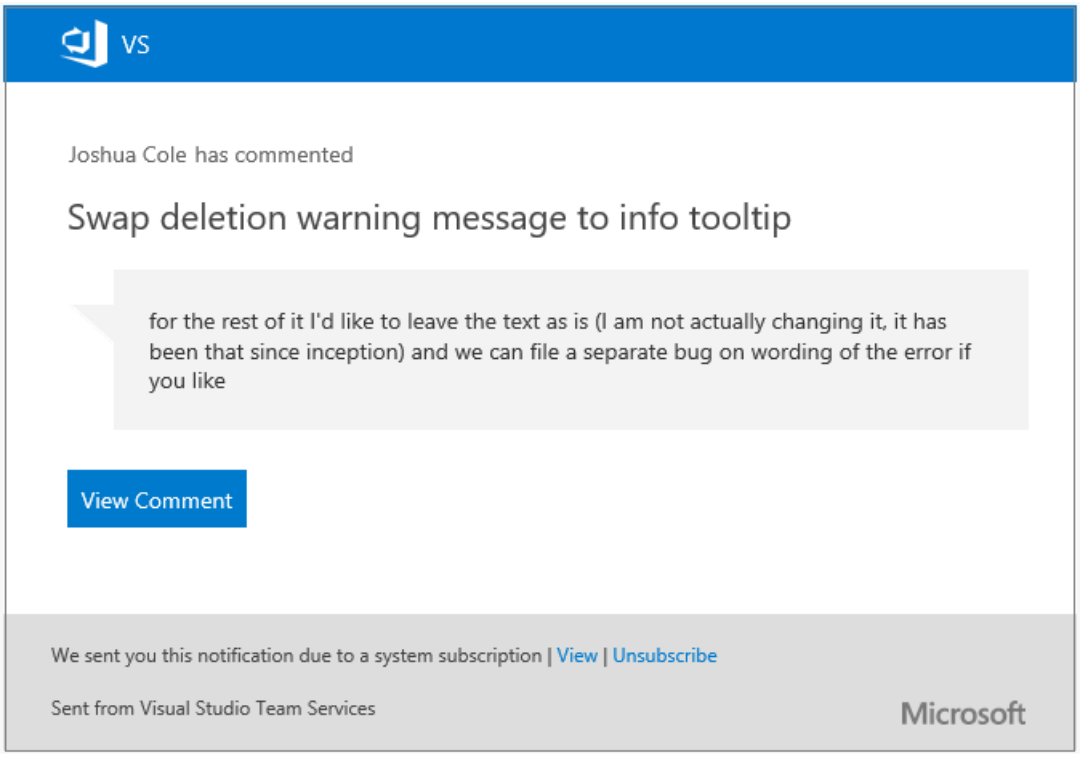
line now begins with the PR title and secondary information, like the repo name, and ID is deferred to the end. The name of the author has been added to the subject to make it simpler to apply rules and filters based on the person that created the PR.

The body of the alert emails has a refreshed template that first summarizes why the alert was sent, followed by the critical metadata (title, branch names, and description), and a main call-to-action button. Additional details like the reviewers, files, and commits are included further down the email.



(Figure 45) Improved email template

For most alerts, the call-to-action *(Figure 46)* is to view the pull request in the web. However, when you are notified about a specific comment, the call-to-action will **link directly to that comment** so you can easily find the code and prior conversation for context.

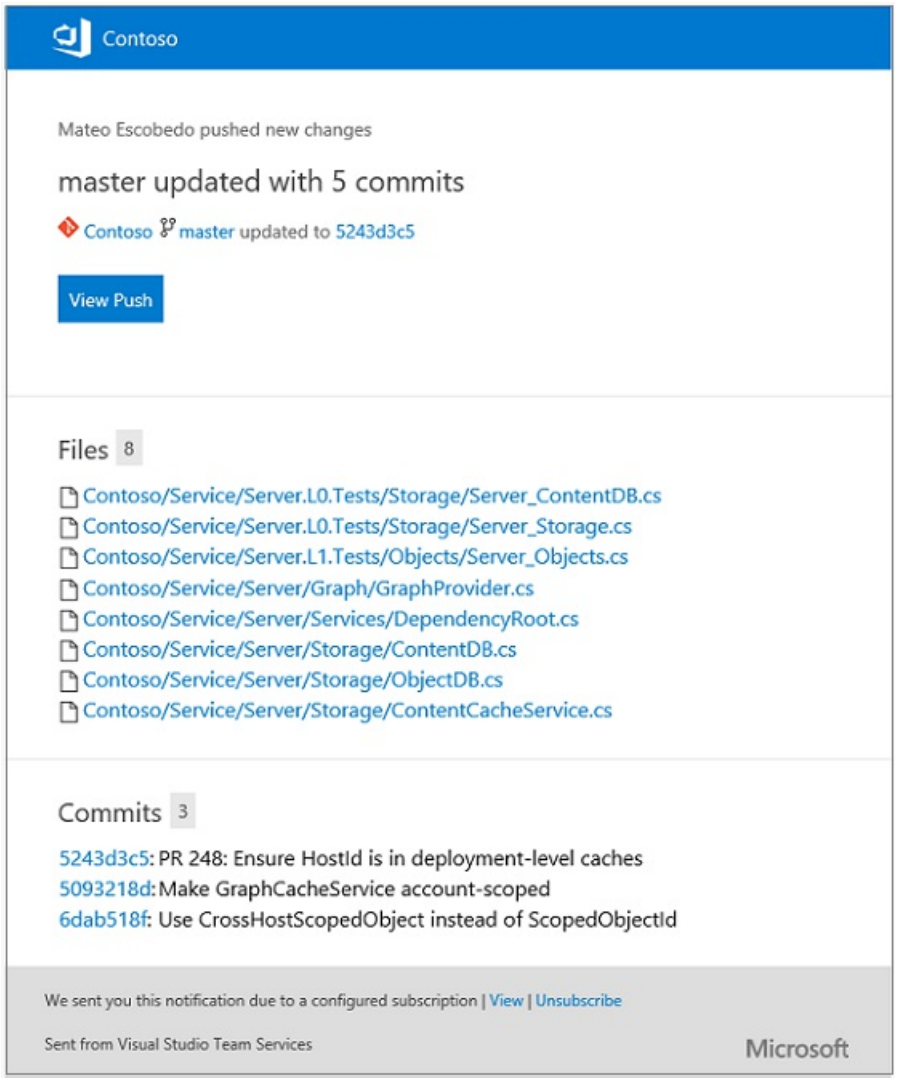


(Figure 46) Email call-to-action

Updated email templates for push notifications

Push notifications have been updated to match the new email templates that are optimized to be clear, concise, and actionable *(Figure 47)*. The subject line helps you clearly distinguish push emails, identify the branch, repo, and author, and summarize the number of commits included in the push. These changes also make it easier to create rules and filters to help manage these email notifications.

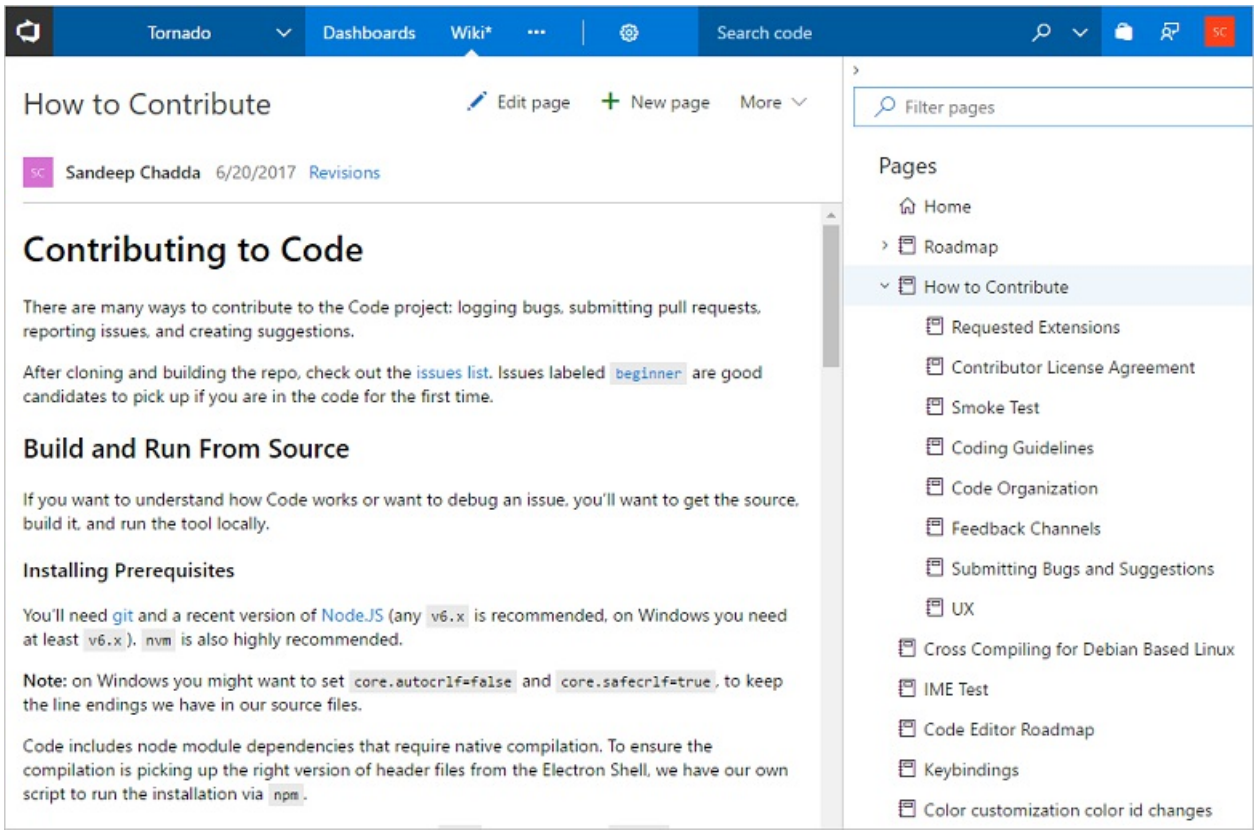
The email body is consistent with other emails, emphasizing why the email was sent, who initiated the action, and exactly what happened. Specific to push alerts, the details about the repo, branch, files, and commits are all included to help inform the recipients about the scope of the changes. The main call-to-action for push alerts is **View Push**, which will open the pushes view for the specific push that generated the alert.



(Figure 47) Push template

Wiki

Each project now supports its own Wiki *(Figure 48)*. Now you can conveniently write pages that help your team members understand, use, and contribute to your project.



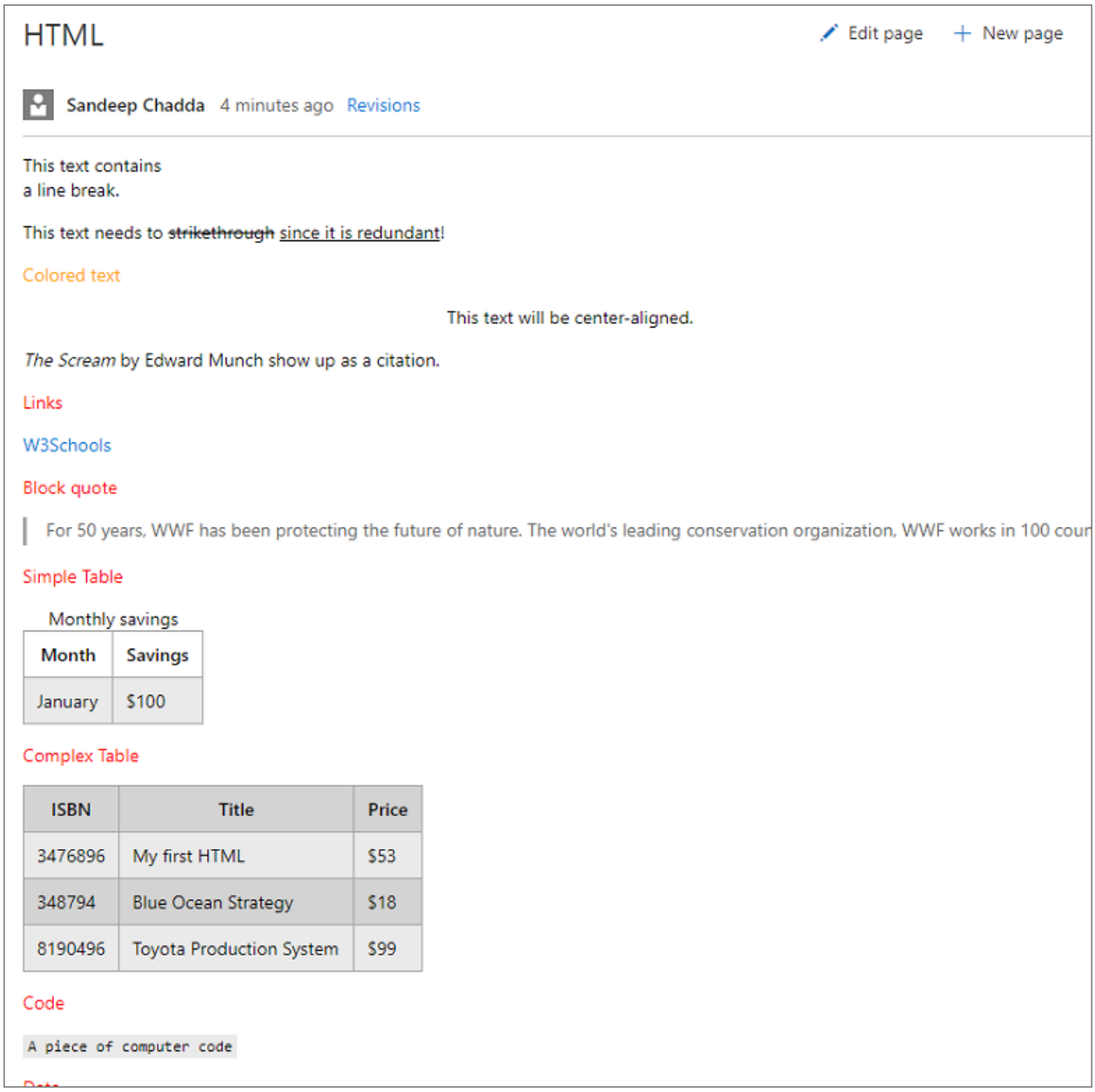
(Figure 48) PR Wiki page

Some of the key features of the new Wiki include:

- Simplified editing experience using markdown syntax.
- The new page allows you to specify a title and add content. *(Figure 49)*

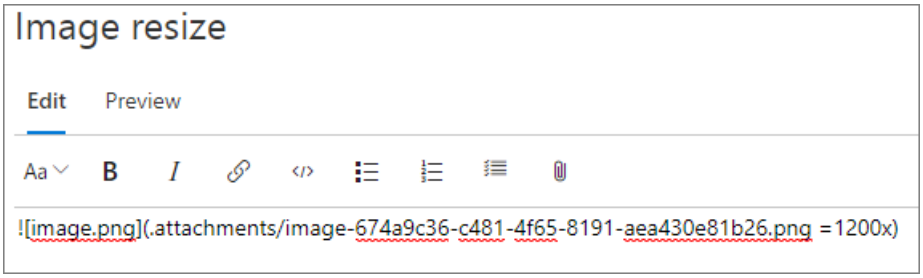
(Figure 49) PR Title Wiki

- Support for HTML tags in markdown *(Figure 50)*.



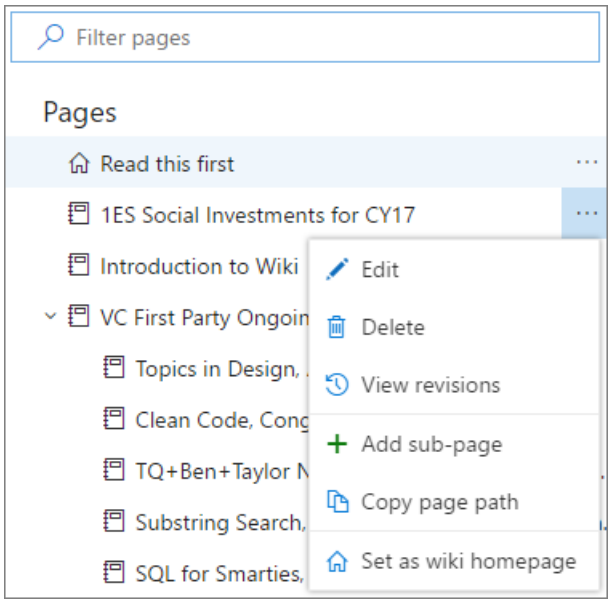
(Figure 50) PR Wiki HTML tags

- Conveniently resize images in the markdown folder *(Figure 51)*.



(Figure 51) PR Image resize

- Powerful page management pane that allows you to reorder, re-parent, and manage pages.
- Ability to filter pages by title for large Wikis *(Figure 52)*.

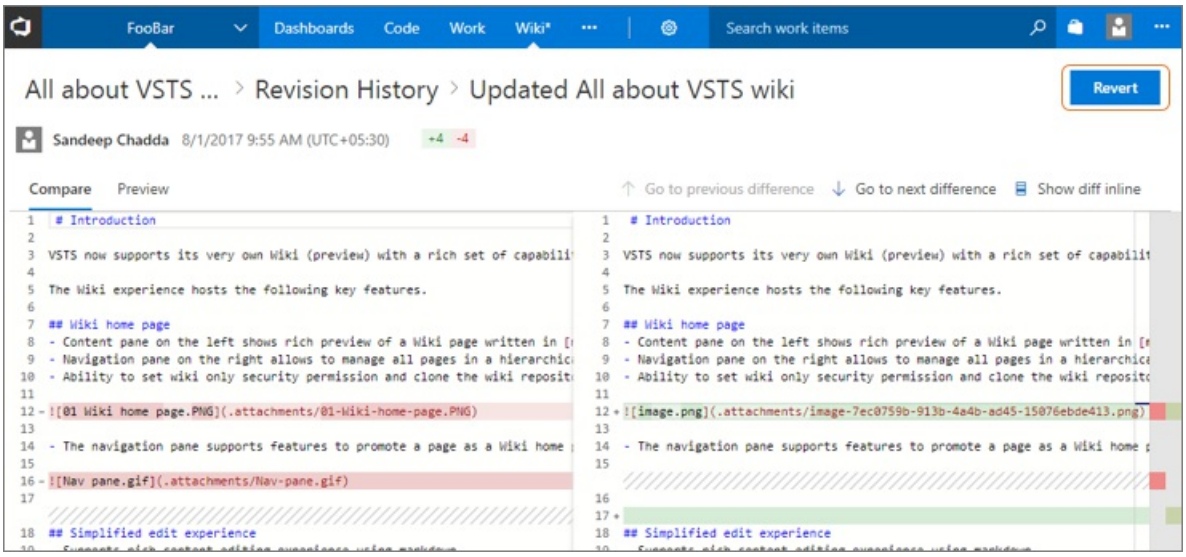


(Figure 52) PR Wiki menu

- Offline updates of Wiki for power users.

> **TIP**
> Learn more about getting started with Wiki.

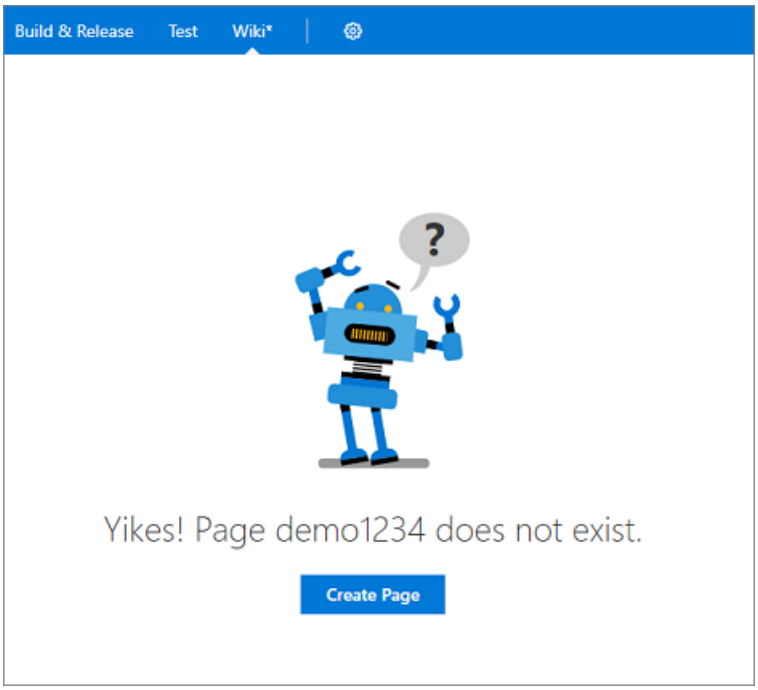
As you use Wiki more, there is a chance you will save unintended changes. Now you can revert a revision to a Wiki page by going to the revision details and clicking on the **Revert** button *(Figure 53)*.



(Figure 53) PR Wiki revert button

Create a Wiki page from a broken link

We observed a pattern during Wiki creation where a table of contents on a Wiki page included non-existent links *(Figure 54)*. Users would click on these links in an attempt to create an actual page. We previously handled this scenario by giving a warning suggesting that the link was broken, or that the page did not exist. Now, we are handling this as a mainstream scenario for Wiki, by allowing you to create pages instead.



(Figure 54) PR Create Wiki page

Wiki page deep linking

Wiki now supports deep linking sections within a page and across pages, which is extremely useful for creating a table of contents. You can reference a heading in the same page or another page by using the following syntax:

- Same page: `[text to display](#section-name)`
- Another page: `[text to display](/page-name#section-name)`

The Wiki extension on the Marketplace is now deprecated. If you are an existing Wiki extension user, then you can migrate your Wiki pages to the new Wiki using this migration tool. Learn more about how to migrate your existing Wiki pages to the new Wiki.

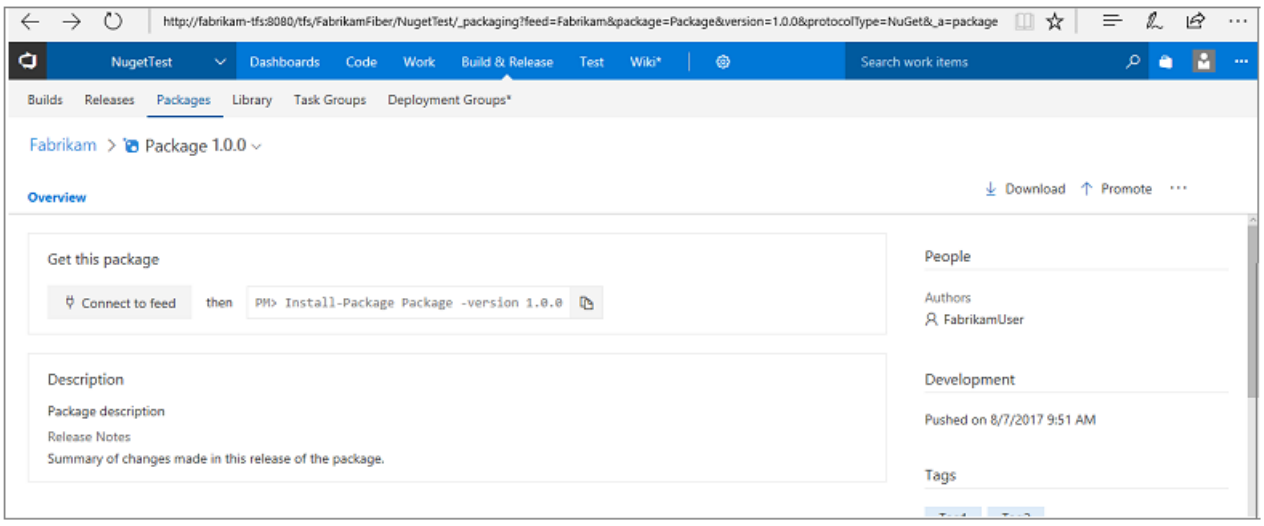
Package Management

Package Management experience updates

Package URLs now work with the package name and version, rather than using GUIDs. This makes it easier to hand-craft package URLs *(Figure 55)*. The format is:

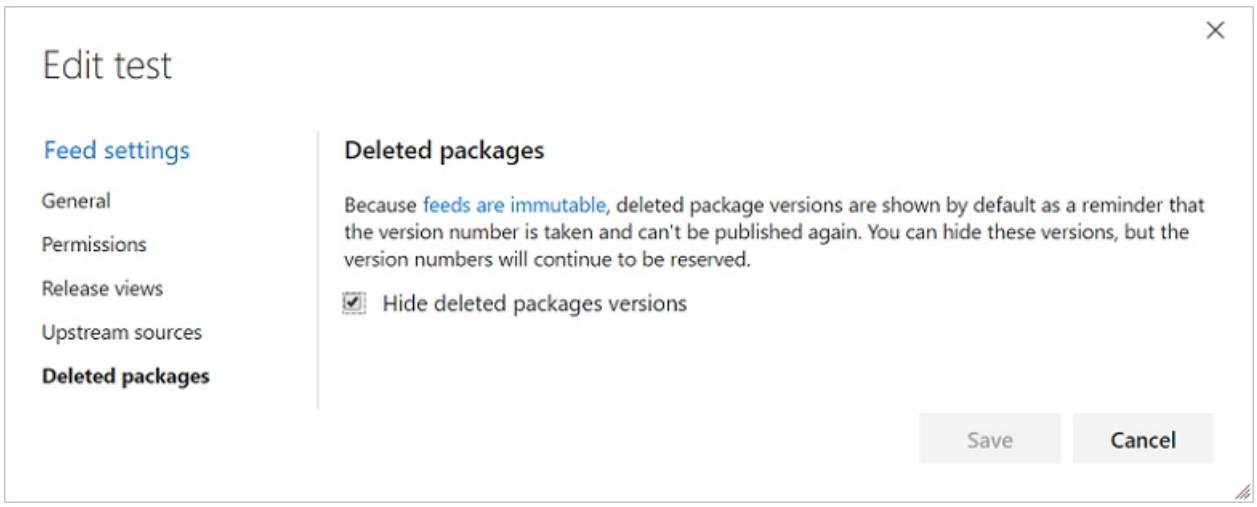
```
\<tfsserverurl\>/\<project|team\>/_packaging?feed=\<feed\>&package=\<package\>&version=\
<version\>&protocolType=\<NuGet|Npm|Maven\>&_a=package
```

.



(Figure 55) PR Package URL

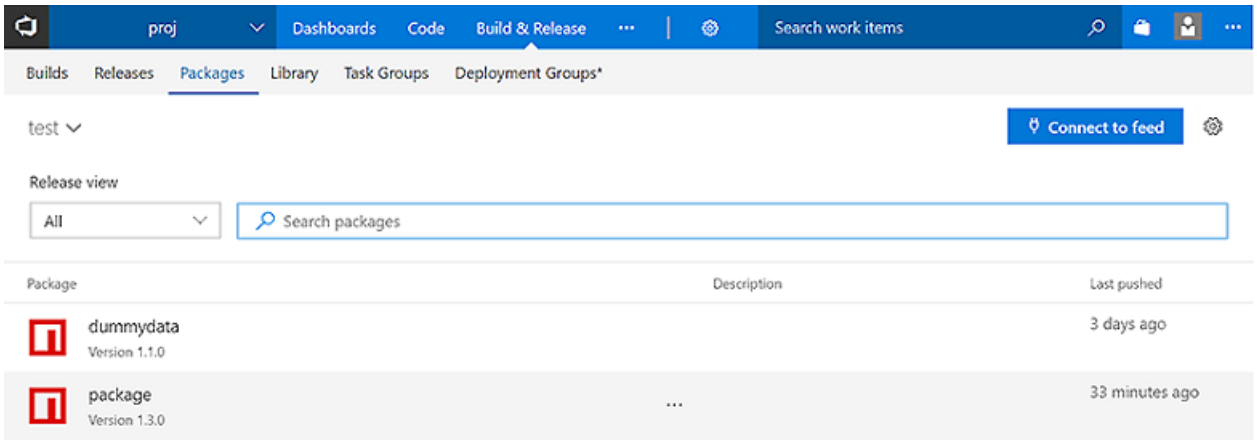
You can now hide deleted package versions *(Figure 56)* from all feed users (no more strikethrough packages!), in response to this UserVoice suggestion.



(Figure 56) Hide deleted packages

Any action that you could perform on the package details page can now be performed from the context menu in the list of packages.

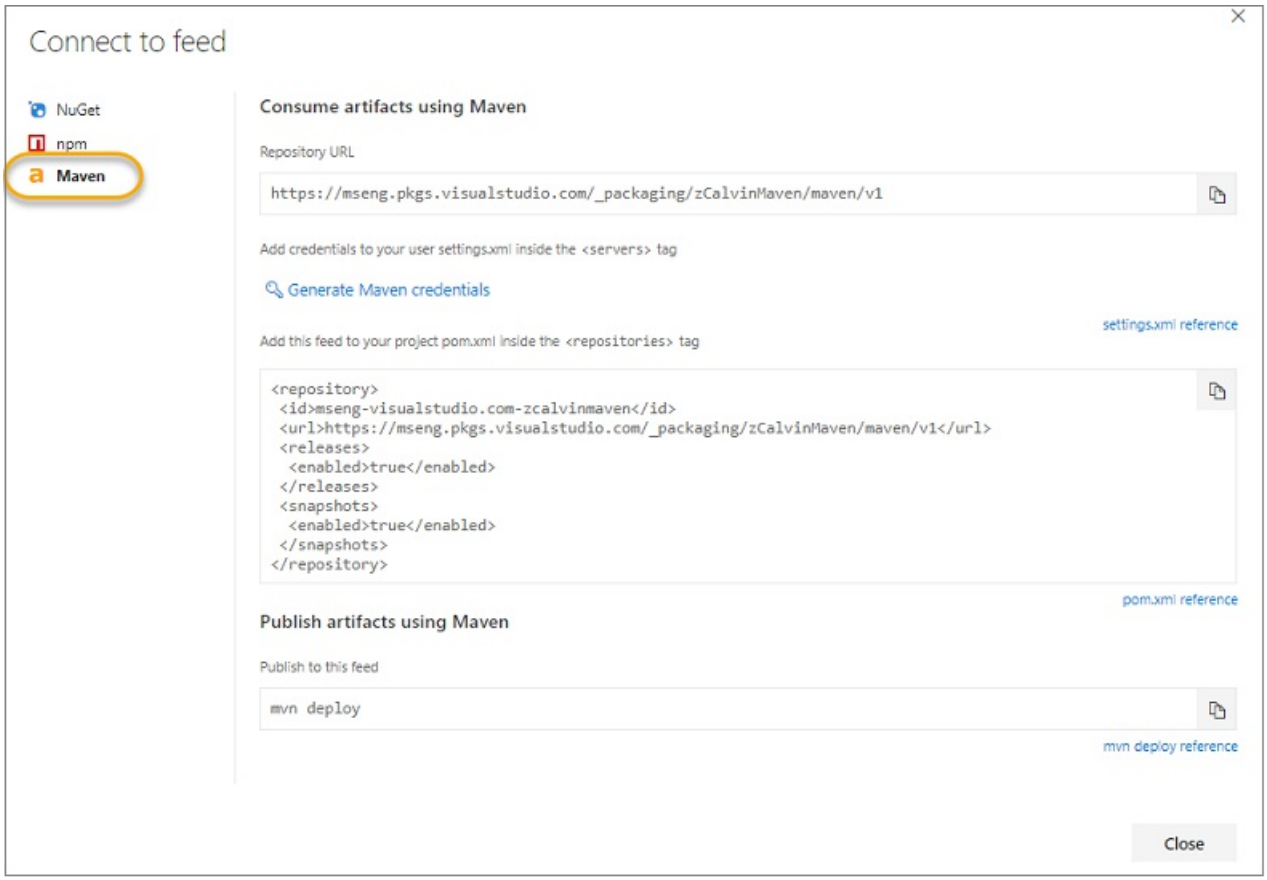
The package list contains a new **Last pushed** column *(Figure 57)* with humanized dates so you can easily find recently-updated packages.



(Figure 57) Last pushed column

Maven packages

We have launched support for hosting Maven artifacts in TFS 2018 *(Figure 58)*. Maven artifacts enable Java developers to share code and components seamlessly. Check out our getting started guide for how to share Maven artifacts using Package Management.

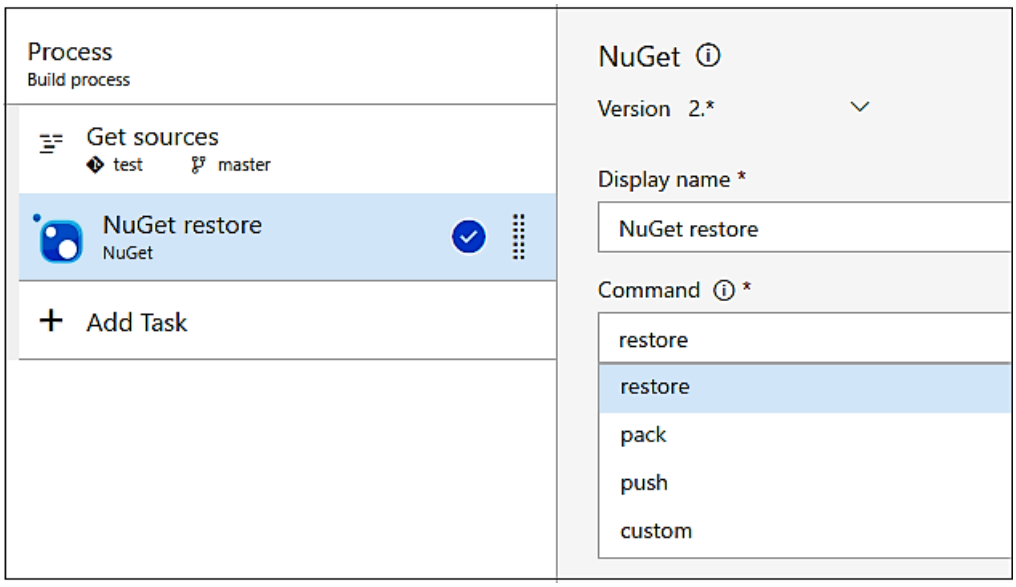


(Figure 58) Maven packages

New unified NuGet task

We have combined the **NuGet Restore**, **NuGet Packager**, and **NuGet Publisher** task into a unified **NuGet** build task to align better with the rest of the build task library; the new task uses NuGet 4.0.0 by default. Accordingly, we have deprecated the old tasks, and we recommend moving to the new NuGet task as you have time. This change coincides with a wave of improvements outlined below that you will only be able to access by using the combined task.

As part of this work, we have also released a new **NuGet Tool Installer** task that controls the version of NuGet available on the PATH and used by the new NuGet task. So, to use a newer version of NuGet, just add a **NuGet Tool Installer** task at the beginning of your build *(Figure 59)*.



(Figure 59) NuGet task

TIP

Read more about using the latest NuGet in your build on Microsoft DevOps Blog.

NuGet "Allow duplicates to be skipped" option

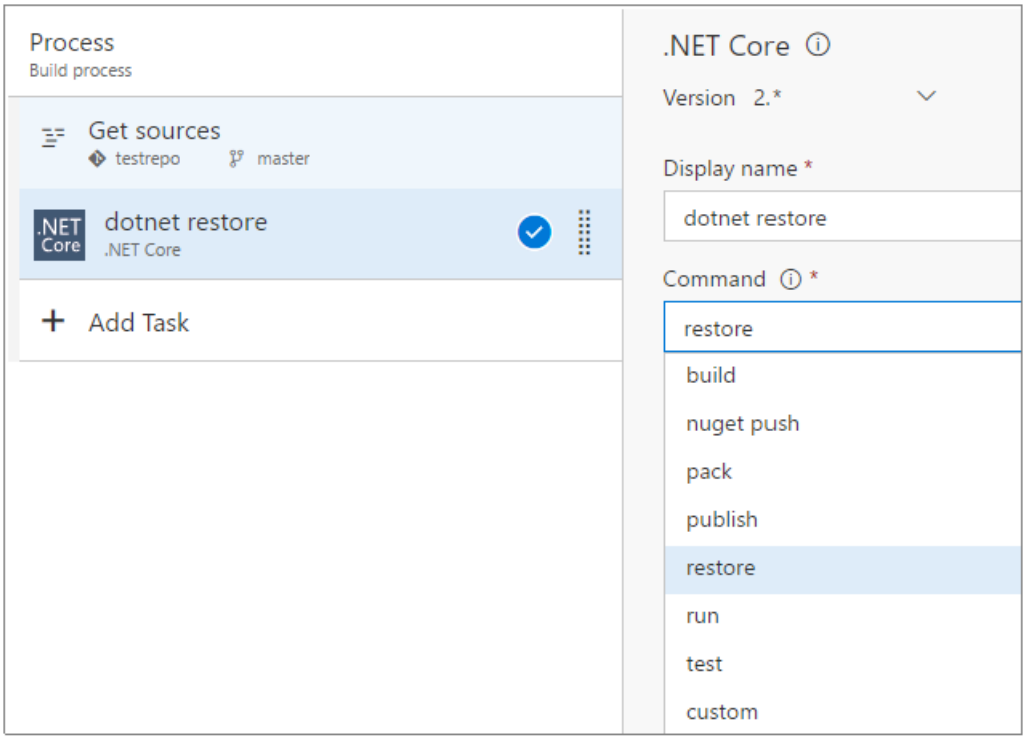
We heard from many NuGet customers that they generate a set of packages, only some of which may have updates (and therefore updated version numbers). The **NuGet** build task has a new **Allow duplicates to be skipped** option that enables the task to continue if it tries to push packages to a VSTS/TFS feed where the version is already in use.

npm build task updates

Whether you are building your **npm** project on Windows, Linux, or Mac, the new **NPM** build task works seamlessly. We have also reorganized the task to make both **npm install** and **npm publish** easier. For **install** and **publish**, we have simplified credential acquisition so that credentials for registries listed in your project's `.npmrc` file are safely stored in a service endpoint. Alternatively, if you are using a VSTS/TFS feed, we have a picker that allows you select a feed, and then generates a `.npmrc` with requisite credentials that are used by the build agent.

Maven now supports authenticated feeds

Unlike **NuGet** and **npm**, the **Maven** build task did not previously work with authenticated feeds. We have updated the **Maven** task so you can work easily with VSTS/TFS feeds *(Figure 60)*.



(Figure 60) dotnet task

dotnet task supports authenticated feeds, web projects

The next major version of the **dotnet** task (2.x) addresses many of your feedback requests and fixes a set of bugs we have tracked for a while. This includes the following:

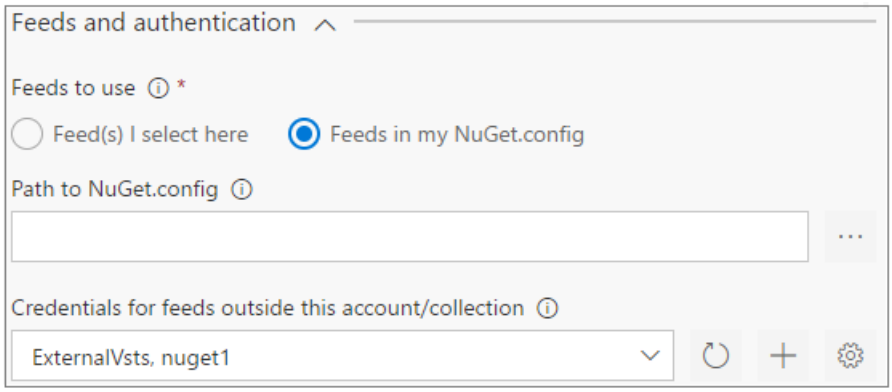
1. First, **dotnet** now supports authenticated package sources like Package Management, so you do not need to use the **NuGet** task anymore to restore packages from private package sources.
2. The behavior of **Path to project(s)** field has changed in the 2.0 version of the task. In previous versions of the task, if the project file(s) matching the specified pattern were not found, the task used to log a warning and then succeed. In such scenarios it can sometimes be challenging to understand why the build succeeded but dependencies were not restored. Now the task fails if the project file(s) matching the specified pattern are not found. This is in line with the behavior of other tasks and is easy to understand and use.

3. In previous versions of the task's publish command, the task modified the output path by putting all the files in a folder that was named after the project file name, even when you passed an explicit output path. This makes it hard to chain commands together. Now you have control over the output path file.

We have also released a new **dotnet Tool Installer** task that controls the version of dotnet available on the PATH and used by the new dotnet task. So, to use a newer version of **dotnet**, just add a **dotnet Tool Installer** task at the beginning of your build.

Working outside your account/collection

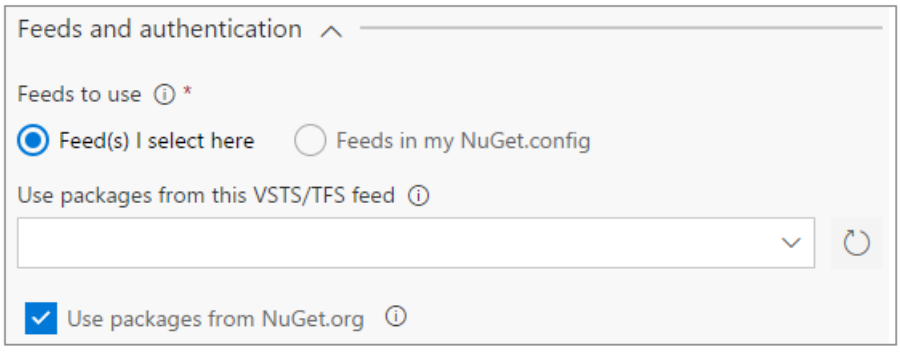
It is now easier to work with feeds *(Figure 61)* outside your TFS server or VSTS account, whether they are **Package Management** feeds in another VSTS account or TFS server or non-Package Management feeds like NuGet.org/npmjs.com, Artifactory, or MyGet *(Figure 60)*. Dedicated **Service Endpoint** types for NuGet and npm make it easy to enter the correct credentials and enable the build tasks to work seamlessly across package download and package push operations.



(Figure 61) Feed to use

Feed picker for VSTS/TFS feeds

We always recommend checking in a configuration file (e.g. NuGet.Config, .npmrc, etc.) so that your source repository has a record of where your packages came from. However, we have heard a set of scenarios where this isn't ideal, so we have added a new **Use packages from this VSTS/TFS feed** option that allows you to select a feed and automatically generate a configuration file that is used for that build step *(Figure 62)*.



(Figure 62) Feed picker

Build and Release

XAML Builds

In TFS 2015, we introduced a web-based, cross-platform build system. XAML builds are not supported in TFS 2018 RTW or Update 1, but we have re-enabled XAML builds in TFS 2018 Update 2. We encourage you to migrate your XAML builds. If you are not ready to migrate and need to continue using XAML builds, please upgrade to TFS 2018 Update 2.

When you upgrade to TFS 2018 RTW or Update 1:

- If you have any XAML build data in your team project collection, you will get a warning about the removal

of XAML build features.

- You can view completed XAML builds, but you will not be able to queue new ones.
- There's no new version of the XAML build controller or agent in TFS 2018.

When you upgrade to TFS 2018 Update 2:

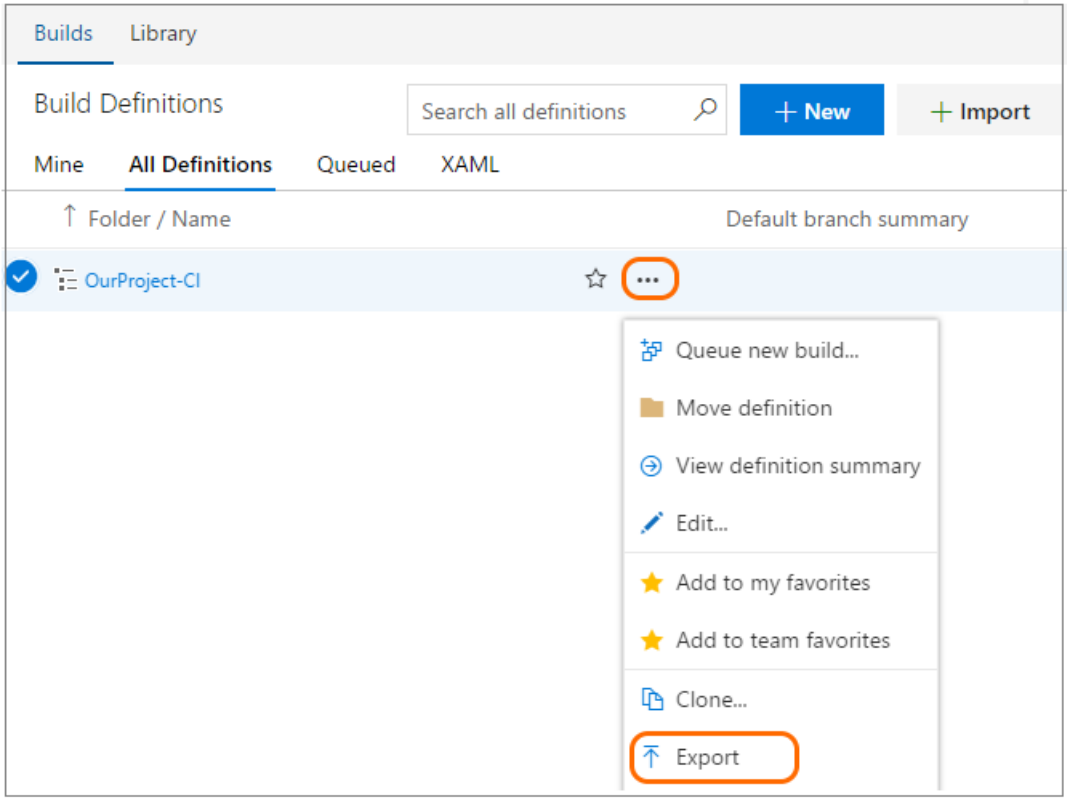
- If you have any XAML build data in your team project collection, you will receive a warning about the deprecation of XAML build features.
- You need to use Visual Studio or Team Explorer 2017 to edit XAML build definitions or to queue new XAML builds.
- If you need to create new XAML build agents, you need to install them using the TFS 2015 build agent installer.

> **TIP**
> For an explanation of our XAML build deprecation plan, see the Evolving TFS/Team Services build automation capabilities blog post.

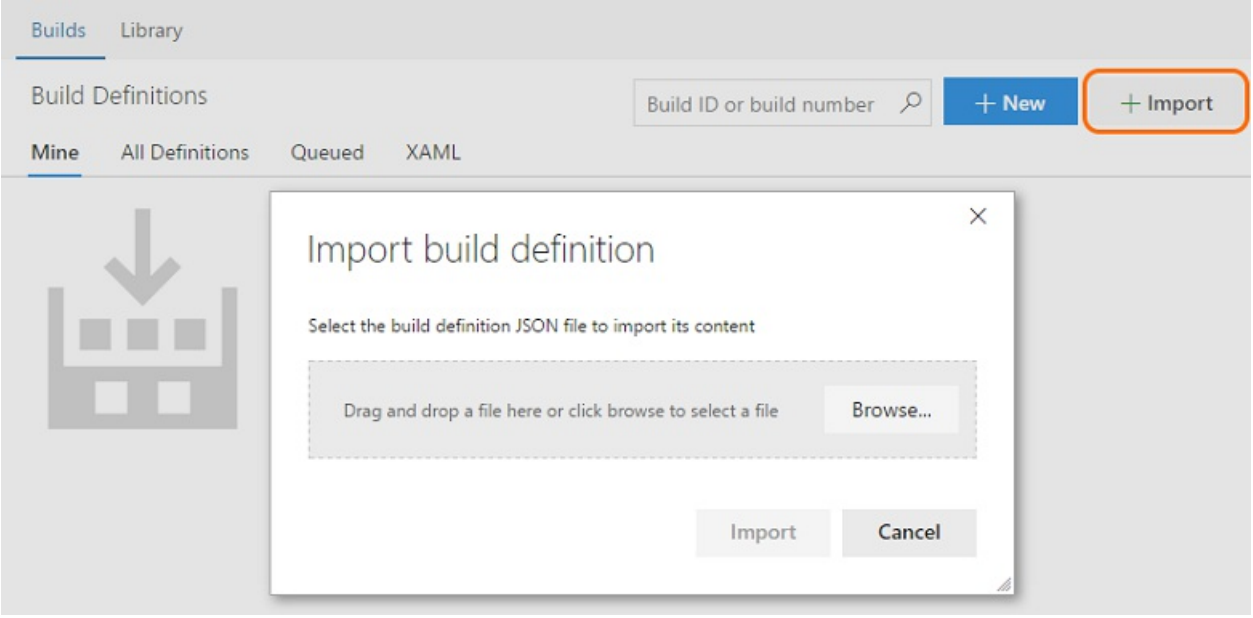
Export and import build definitions

Build definitions are implemented internally as .json files, so you can see details on changes in the file's history. You can already clone and make templates from your build definitions, but many users have wanted to take a copy of their CI build logic and reuse it in another team project. This has been a top-ten request on UserVoice.

We are pleased to announce that this is now possible *(Figure 63)* and *(Figure 64)*!



(Figure 63) Export build definition



(Figure 64) Import build definition

Extensions with build templates

Build templates let you create a baseline for users to get started with defining their build process. We ship a number of them in the box today and while you could upload new ones to your account, it was never possible for extension authors to include new templates as part of an extension. You can now include build templates in your extensions. For example:

```
{  "id": "Template1",
   "type": "ms.vss-build.template",
   "targets": [ "ms.vss-build.templates" ],
   "properties": { "name": "Template1" } }
```

For the full example, see https://github.com/Microsoft/vsts-extension-samples/tree/master/fabrikam-build-extension.

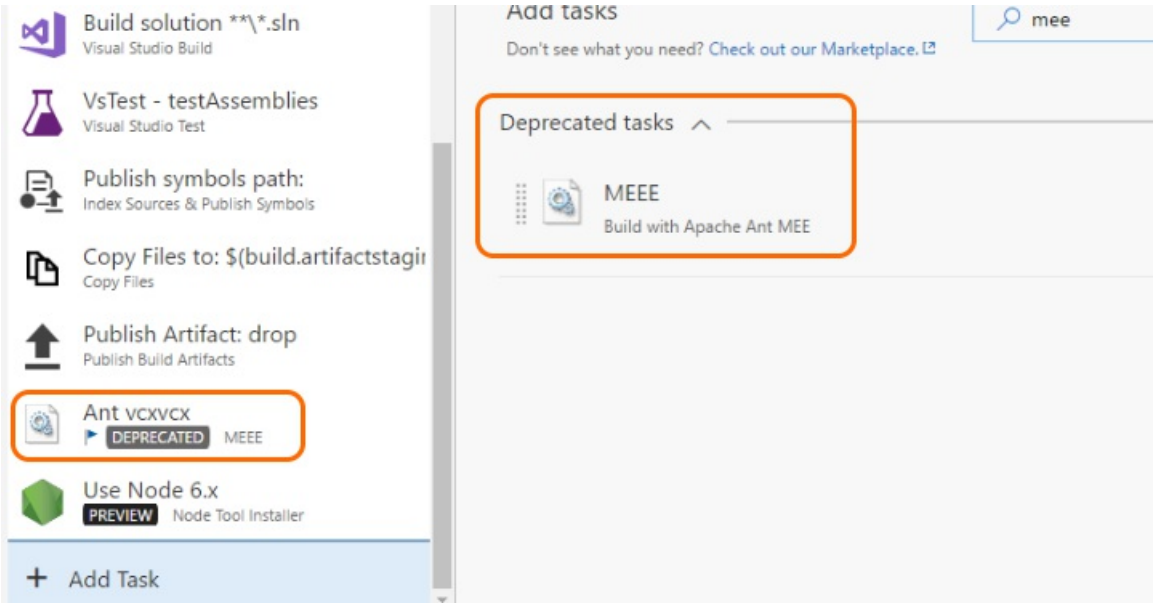
> **TIP**
> You can use this capability to offer and share the same custom template across all your team projects.

Deprecate a task in an extension

You can now deprecate a task in your extension. To make it work, you must add the following variable to the latest version of your task:

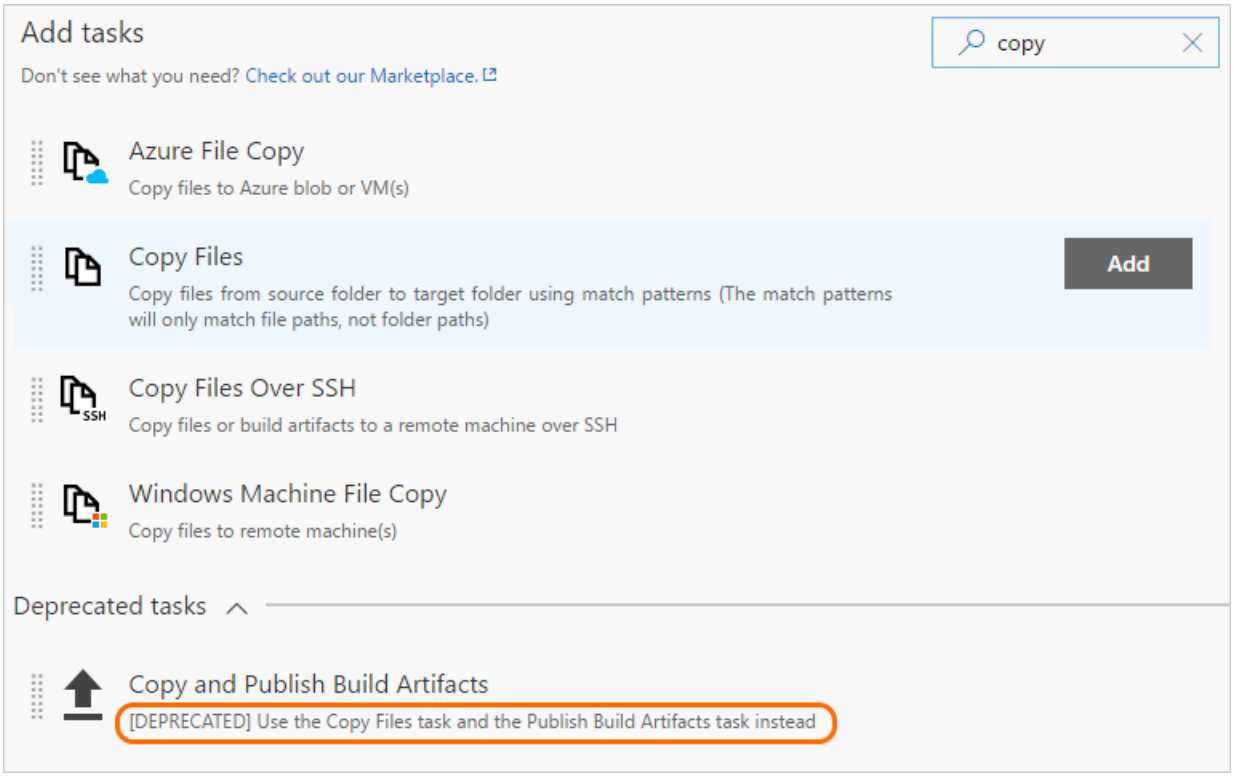
```
"deprecated": true
```

When the user searches for deprecated tasks *(Figure 65)*, we push these tasks to the end and group them under a collapsible section that is collapsed by default. If a definition is already using a deprecated task, we show a deprecated task badge to encourage users to switch to the replacement.



(Figure 65) Deprecated task badge

You can help your users learn about the replacement task by mentioning it in the task description *(Figure 66)*. The description then points users using the task in the right direction from both the task catalog and the existing build/release definitions.



(Figure 66) Deprecated task description

Let contributed build sections control section visibility

Previously, if you were using an extension that had build tasks and build summary sections, you would see the build summary section even if you were not using the build task in that build. Now, you can choose to hide or show that section in the build summary page by adding the following line in your extension code and setting the value to true or false:

```
VSS.getConfiguration().setSectionVisibility("$(publisherId).$(extensionId).$(sectionId)", false);
```

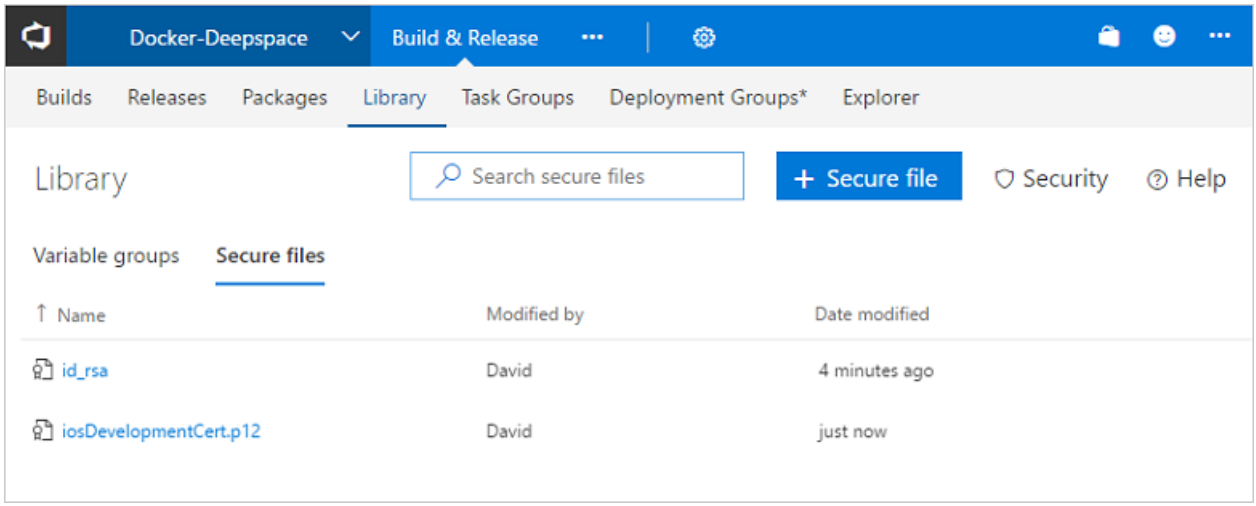
View the sample included in the Microsoft vsts-extension-samples repository.

Variable group support

Variable groups have been available to use in release definitions, and now they are ready to be used in build definitions, too. Learn more about creating a variable group. This has been developed and prioritized based on related suggestions for project-level build/release variables and variable groups in build definitions.

Work with secure files such as Apple certificates

We have added a general-purpose secure files library *(Figure 67)*.



(Figure 67) Secure files library

Use the secure files library to store files such as signing certificates, Apple Provisioning Profiles, Android Keystore files, and SSH keys on the server without needing to commit them to your source repository.

The contents of secure files are encrypted and can only be used during build or release processes by referencing them from a task. Secure files are available across multiple build and release definitions in the team project based on security settings. Secure files follow the Library security model.

We have also added some Apple tasks that leverage this new feature:

- Utility: Install Apple Certificate
- Utility: Install Apple Provisioning Profile

Pause build definitions

You can now pause or disable build definitions. If you plan to make changes to your build definition and you want to avoid queuing any new builds until you are done, simply disable the build definition. Similarly, if you plan to upgrade agent machines, you can choose to pause a build definition, which enables VSTS to still accept new build requests but hold them in queue without running until you resume the definition.

Task input validations support

Typing the parameters in build definition tasks can sometimes be error prone. With task input validation, task authors can ensure appropriate values are specified. Validation expressions follow the familiar expression syntax used for task conditions and can use any of the supported functions besides the general functions supported by task conditions, including URL, IPV4, email, number range, sha1, length, or match.

> **TIP**
>
> Read more about the goals and usage on the vsts-tasks repo page.

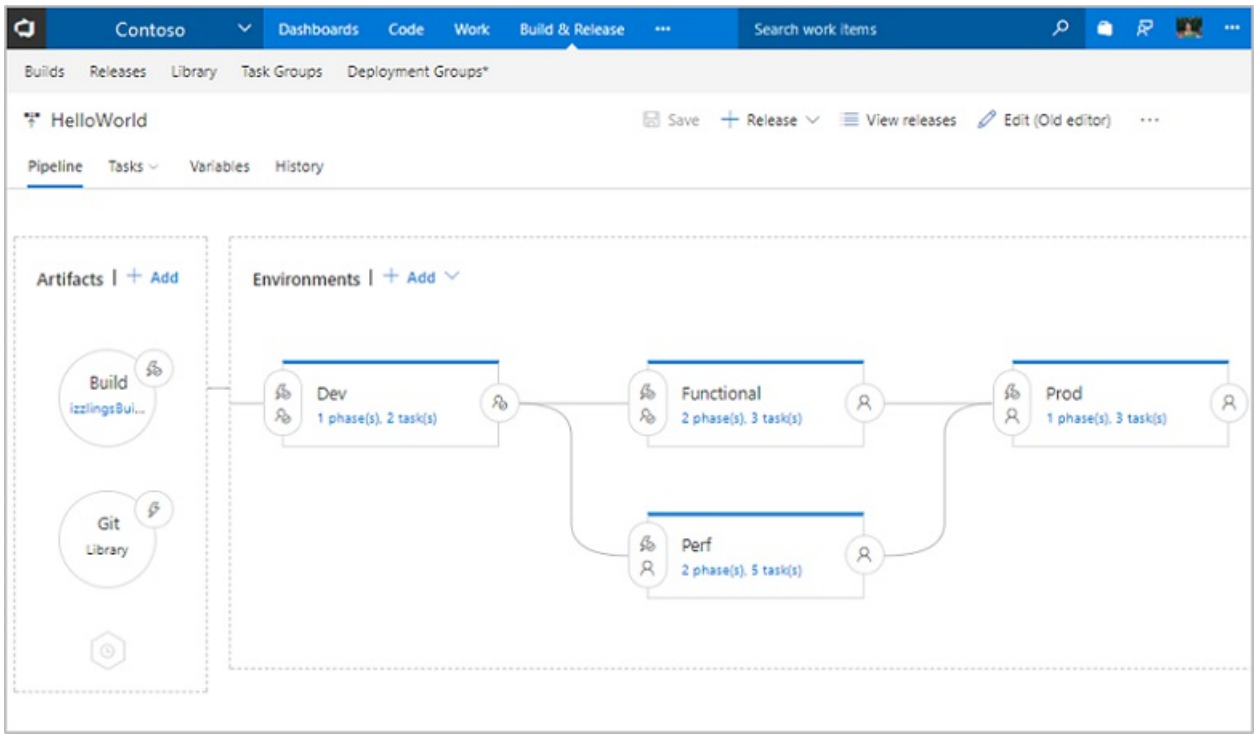
New Release Definition Editor

Continuing on our journey of refreshing the Build and Release experiences, we have re-imagined the release definition editor to provide a more intuitive experience, fix some pain points, and add new capabilities. One of the most powerful features of the new editor is its ability to help you visualize how deployments to your environments would progress. In addition to this, approvals, environment properties, and deployment settings are now in-context

and easily configurable.

Visualization of the pipeline

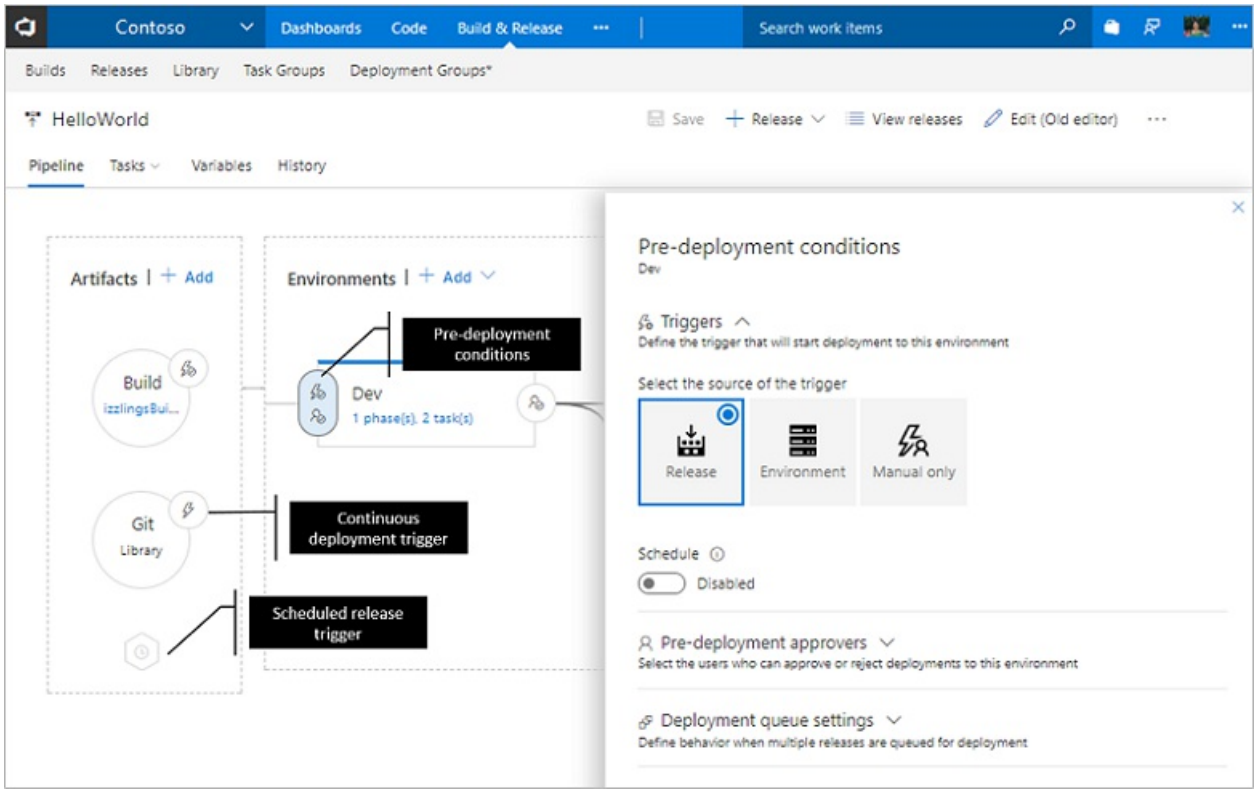
The pipeline *(Figure 68)* in the editor provides a graphical view of how deployments progress in a release. The artifacts are consumed by the release and deployed to the environments. The layout and linking of the environments reflects the trigger settings defined for each environment.



(Figure 68) Release pipeline

In context configuration UI

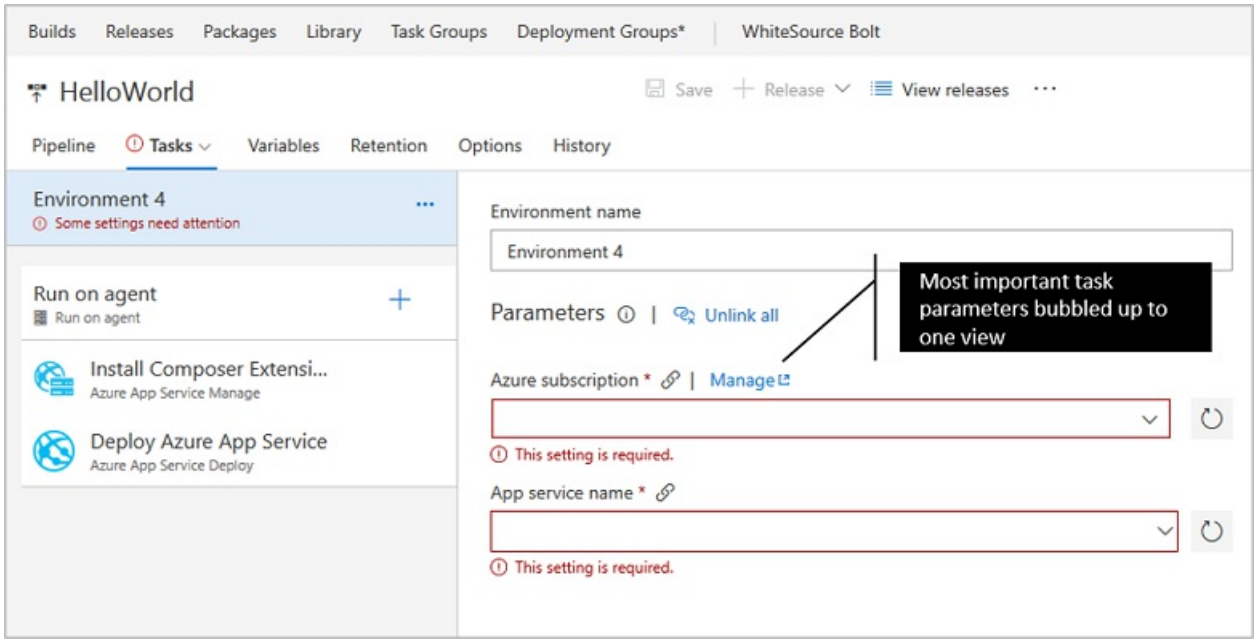
Artifacts, release triggers, pre-deployment and post-deployment approvals, environment properties, and deployment settings are now in-context and easily configurable *(Figure 69)*.



(Figure 69) Release configuration

Getting started with deployment templates

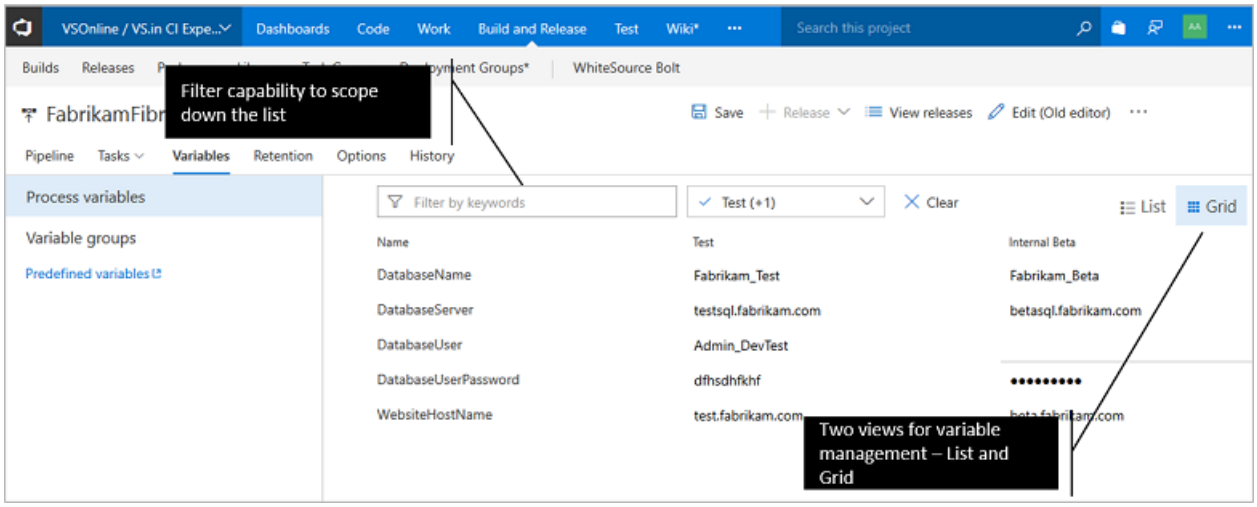
All in-built deployment templates are equipped with process parameters that simplifies the process for users to get started by specifying the most important parameters without needing to go deep into the tasks *(Figure 70)*.



(Figure 70) Deployment templates

Simplified management of release and environment variables

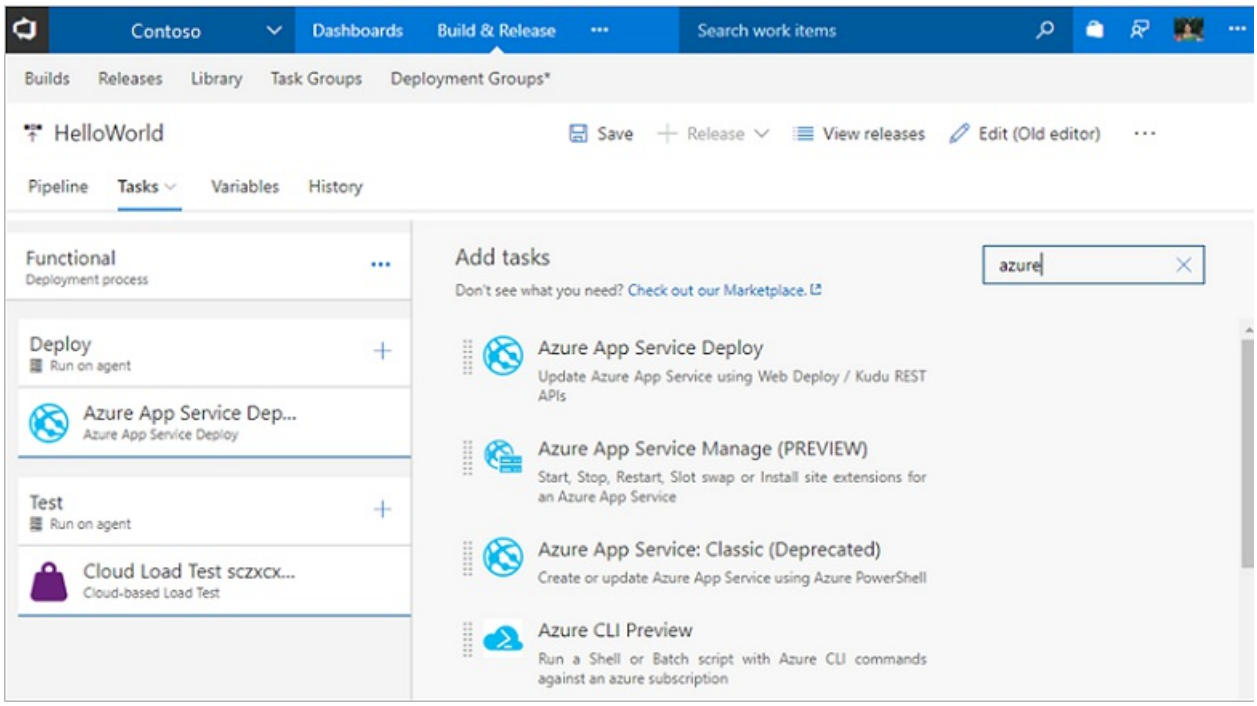
Use the **List** view to quickly add release or environment variables and the **Grid** view to compare and edit variables across scopes side-by-side *(Figure 71)*. Additionally, you can use the filter and keyword search to manage the set of variables to work with in both the views.



(Figure 71) Simplified management of variables

Improved task and phase editor

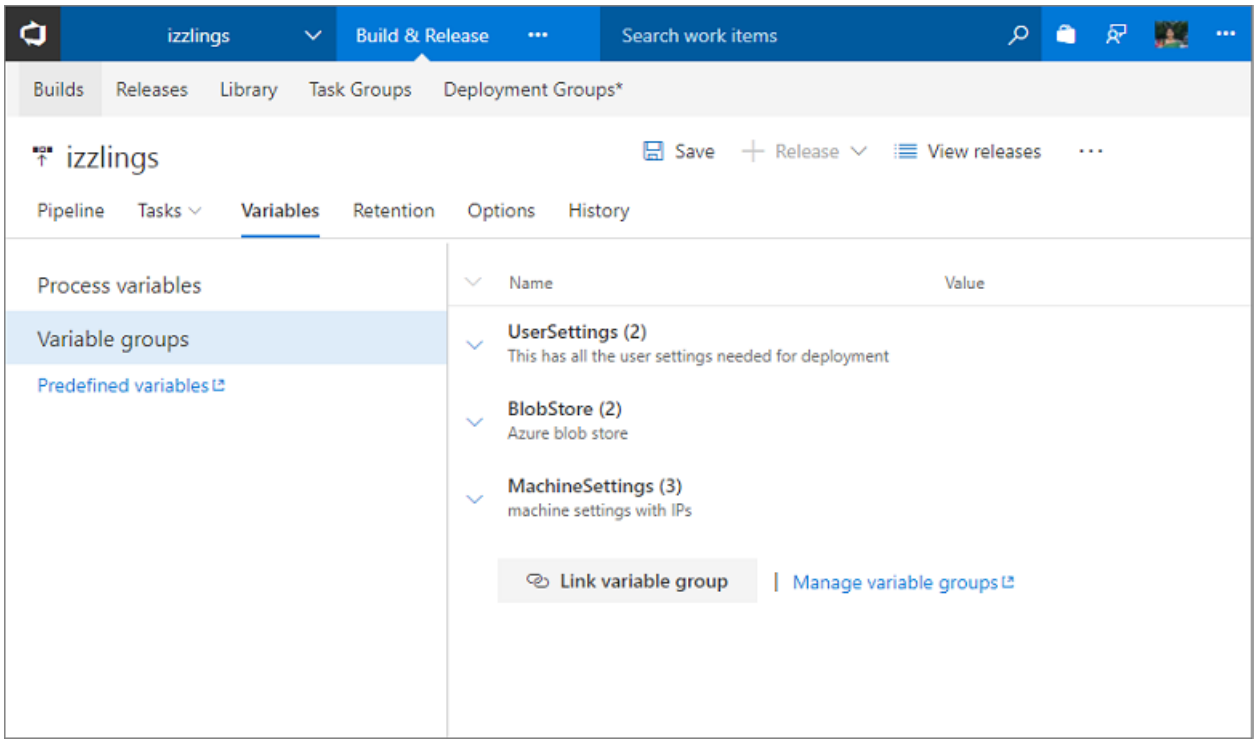
All the enhancements in the new build definition editor are now available in the release definition editor, as well *(Figure 72)*. You can search for tasks and add them by either using the **Add** button or by using drag/drop. You can reorder or clone tasks using drag/drop.



(Figure 72) Task editor

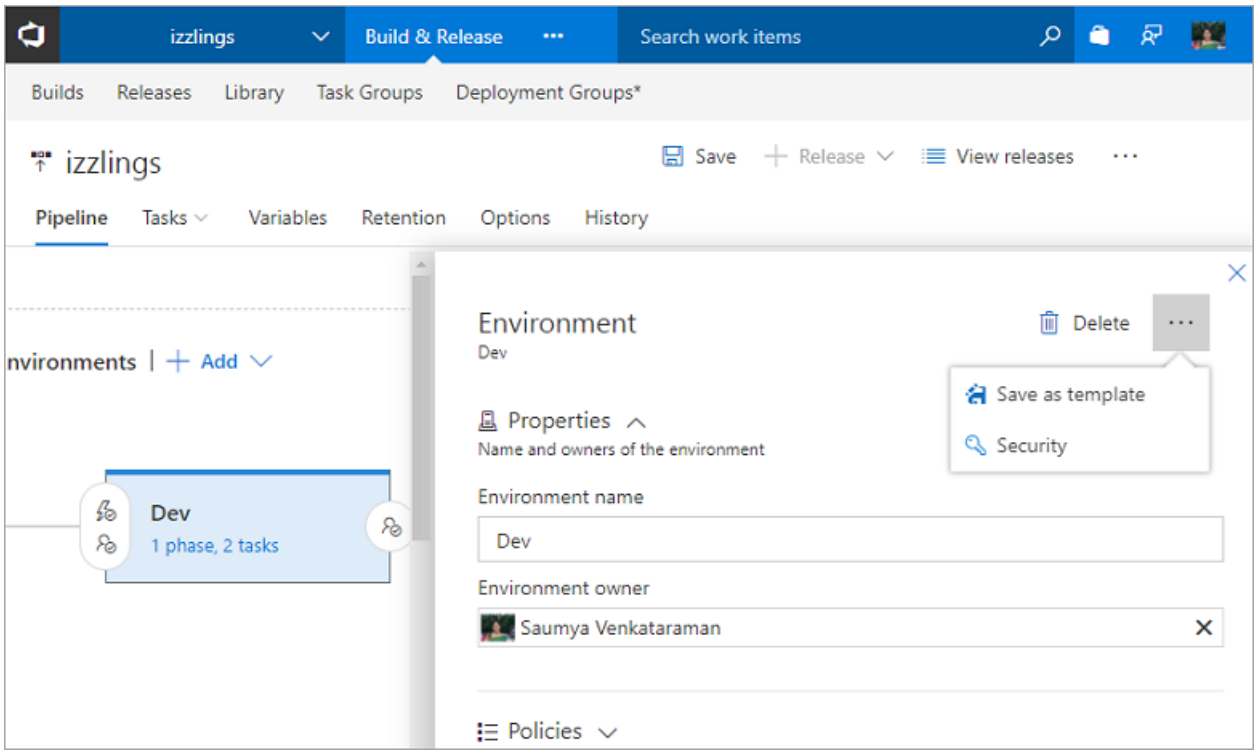
Variable groups, Retention, and Options tabs

You can now link/unlink to variable groups *(Figure 73)*, set retention policy for individual environments, and modify release definition-level settings such as release number format from the **Options** tab. You can also save an environment as a deployment template, set environment level permissions, and re-order phases within the **Tasks** tab.



(Figure 73) Variable groups

Use environment level operations to save as template and set security *(Figure 74)*.



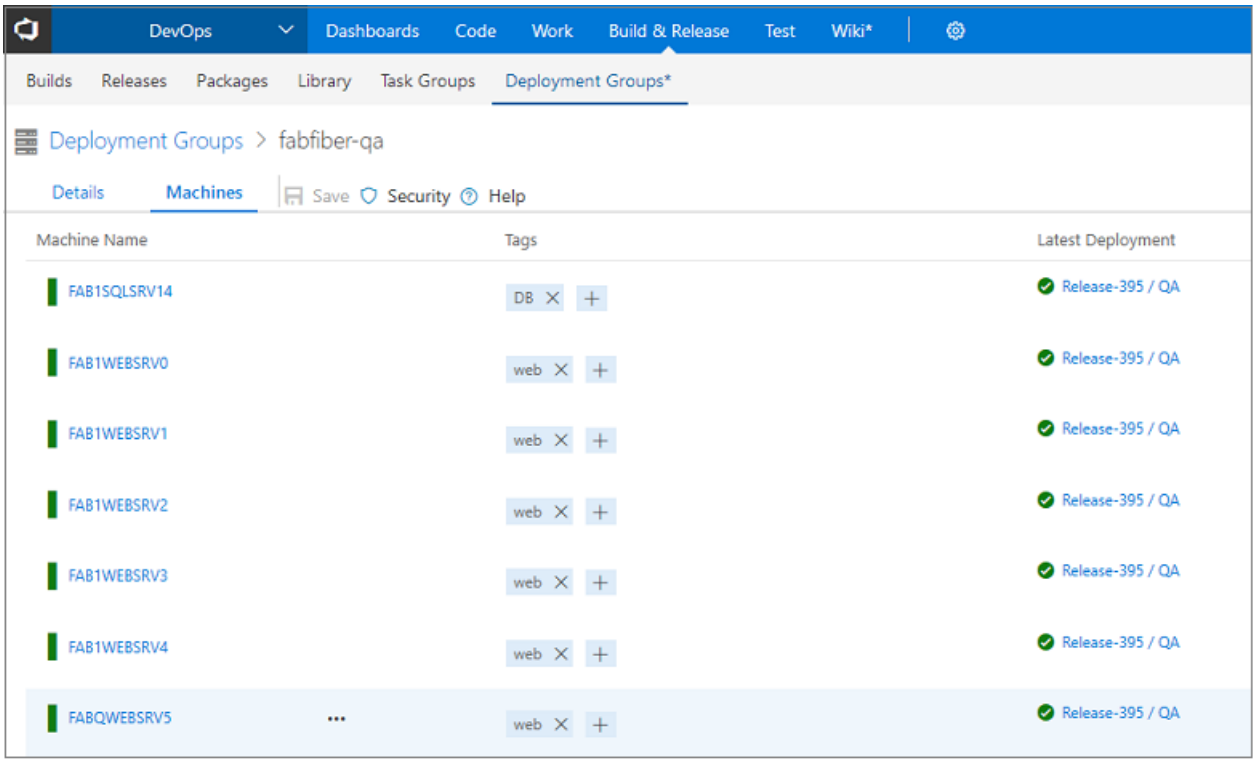
(Figure 74) Environment menu

Virtual Machine Deployment using Deployment Groups

Release Management now supports robust out-of-the-box multi-machine deployment. You can now orchestrate deployments across multiple machines and perform rolling updates while ensuring high availability of the application throughout.

Agent-based deployment capability relies on the same build and deployment agents. However, unlike the current approach where you install the build and deployment agents on a set of proxy servers in an agent pool and drive deployments to remote target servers, you install the agent on each of your target servers directly and drive rolling deployments to those servers. You can use the full task catalog on your target machines.

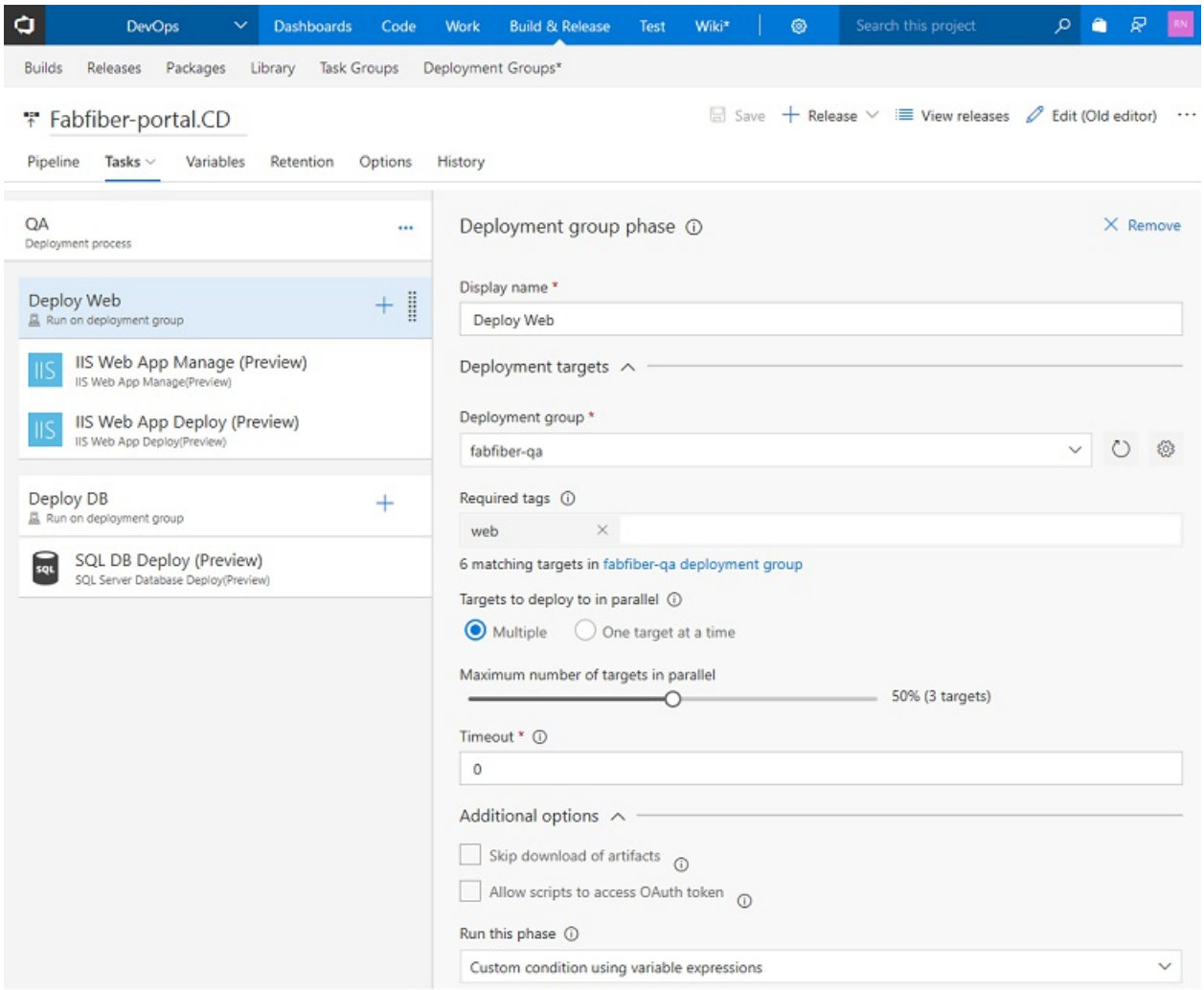
A deployment group *(Figure 75)* is a logical group of targets (machines) with agents installed on each of them. Deployment groups represent your physical environments, such as single-box Dev, multi-machine QA, and a farm of machines for UAT/Prod. They also specify the security context for your physical environments.



(Figure 75) Deployment groups

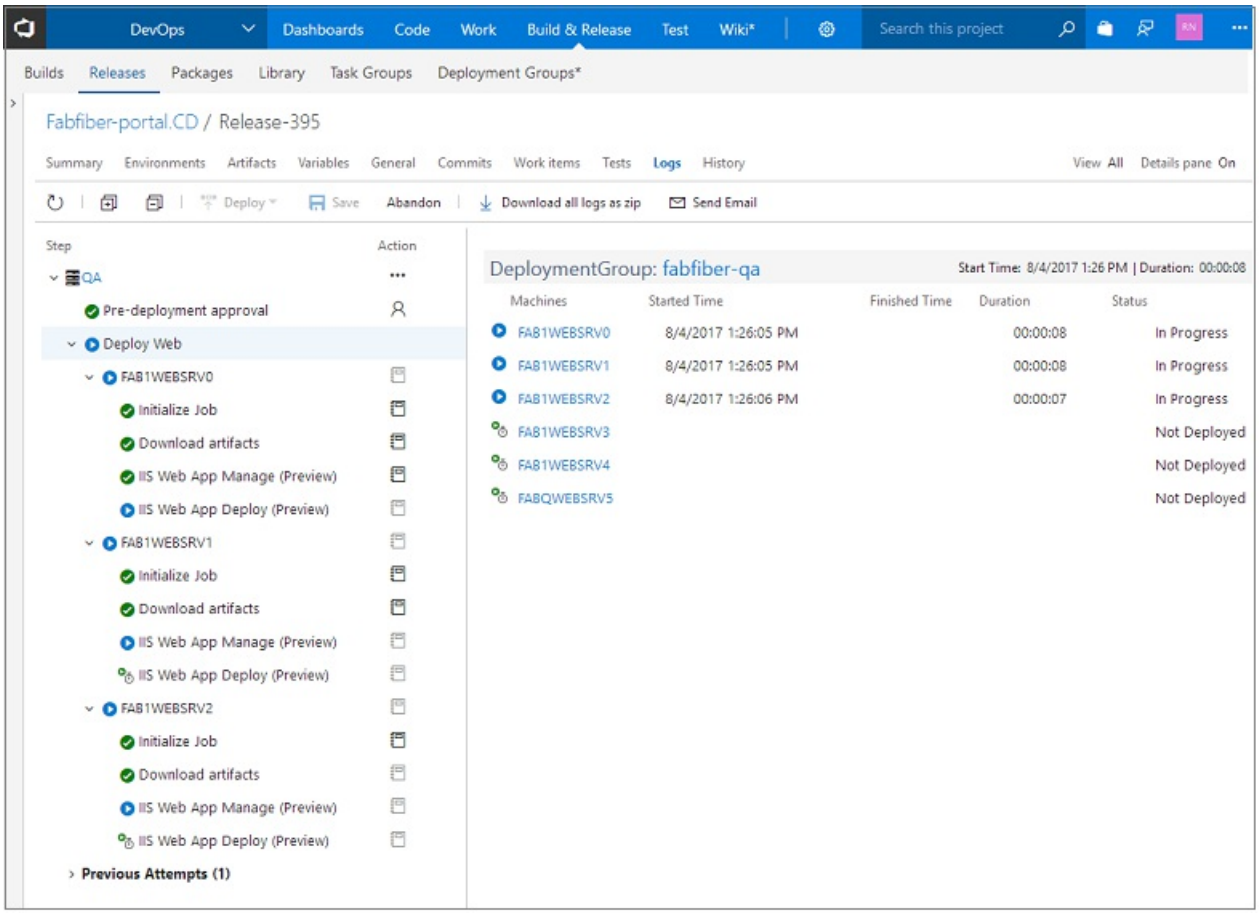
You can use this against any virtual machine that you register our agent with. We have also made it very easy to register with Azure with support for an Azure virtual machine extension that auto-installs the agent when the virtual machine spins up. Tags are automatically inherit on the Azure virtual machine when it is registered.

Once you have a deployment group, you simply configure what you want us to execute on that deployment group *(Figure 76)*. You can control what gets run on which machines using tags and control how fast or slow the rollout happens.



(Figure 76) Configure deployment groups

When the deployment is run, the logs show the progression across the entire group of machines you are targeting *(Figure 77)*.

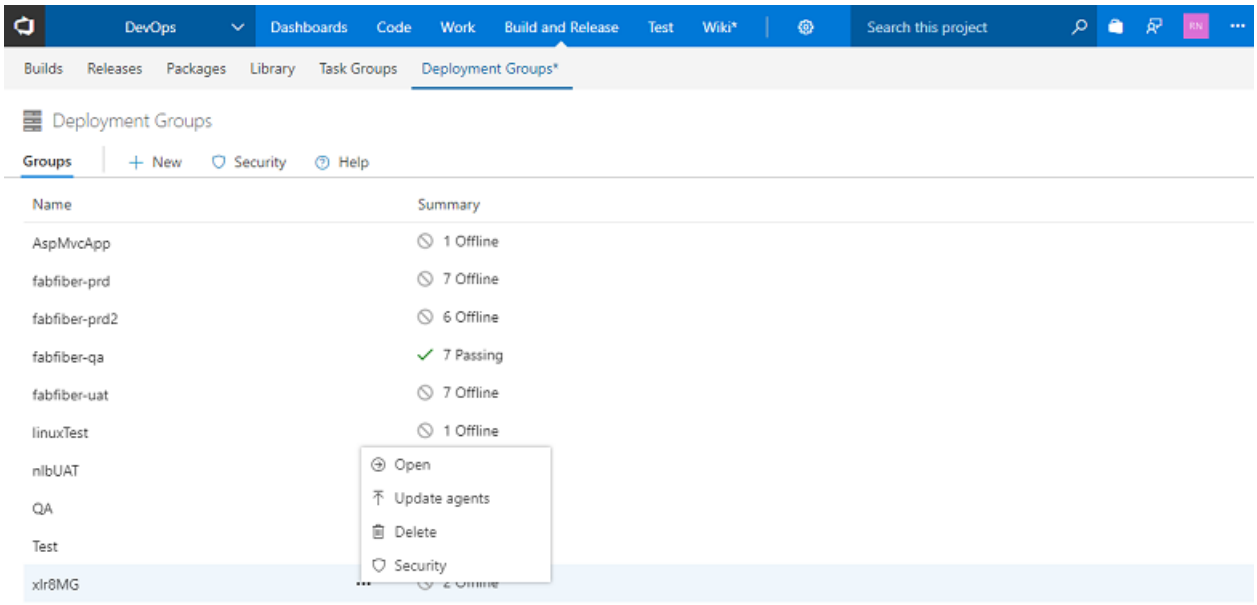


(Figure 77) Deployment group progress

This feature is now an integrated part of Release Management. No additional licenses are required.

Improved Deployment Groups UI

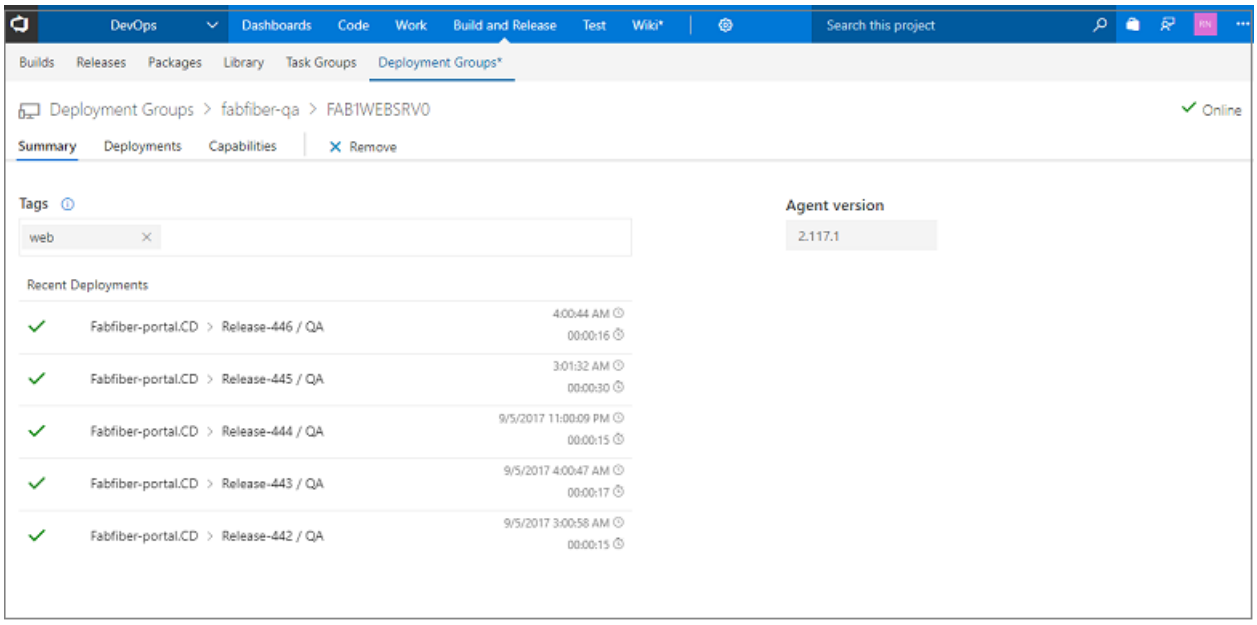
Continuing our journey of refreshing the **Build** and **Release** experiences, we have now re-imagined our deployment groups pages to make it a more clean and intuitive experience *(Figure 78)*. From the landing page, you can view the health of the targets in the deployment group. You can also manage security for an individual deployment group, or set default permissions across deployment groups.



(Figure 78) Deployment groups UI

For a target within a deployment group, you can view a summary, recent deployments, and the target's capabilities *(Figure 79)*. You can set tags on the target, and control what is run on each target. We are adding filter support for

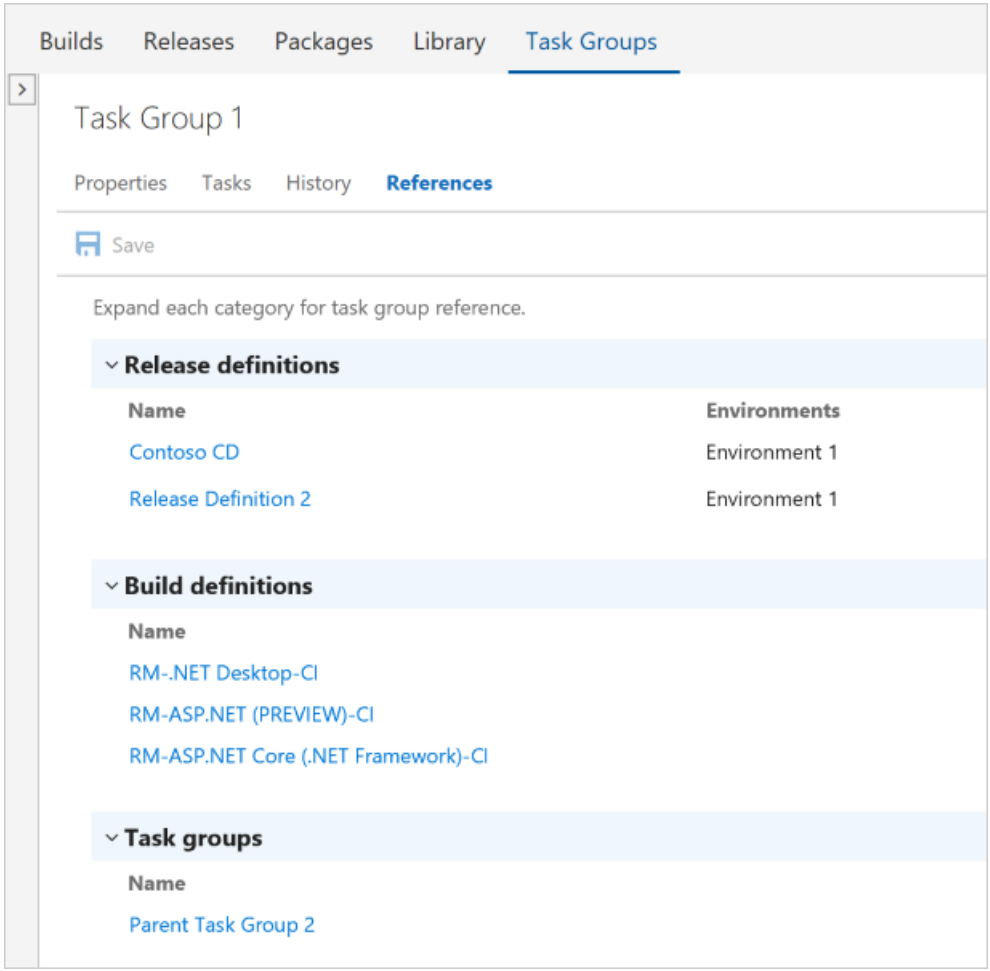
deployment groups in upcoming releases.



(Figure 79) Deployment groups UI tags

Task group references

Task groups let you define a set of tasks that you can add to your build or release definitions *(Figure 80)*. This is handy if you need to use the same grouping of tasks in multiple builds or releases. To help you track the consumers of a task group, you now have a view into the build definitions, release definitions, and task groups that reference your task group *(Figure 79)*.



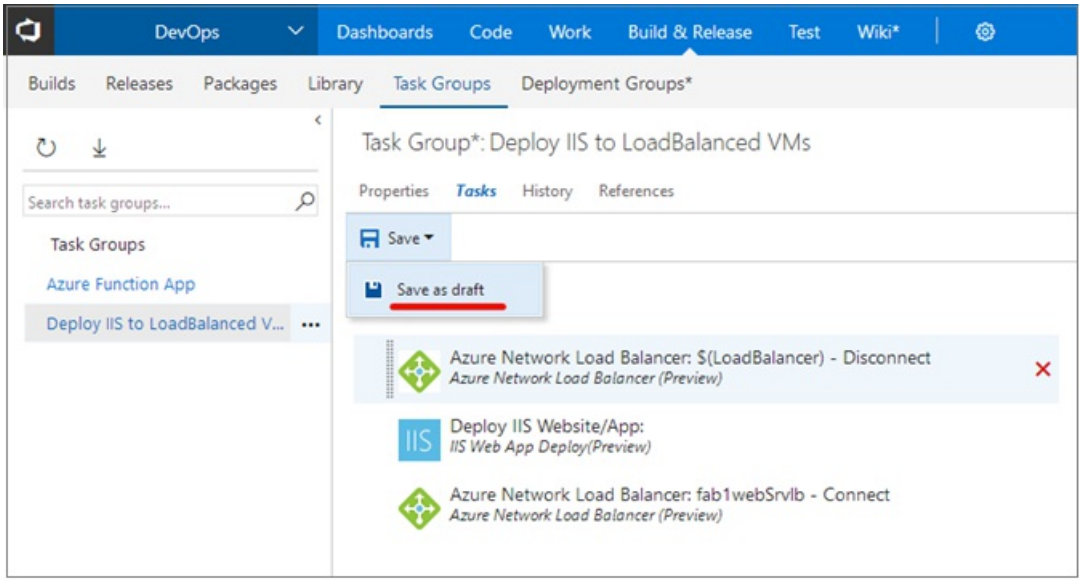
(Figure 80) Task group references

When you try to delete a task group that is still referenced, we warn you and give you a link to this page.

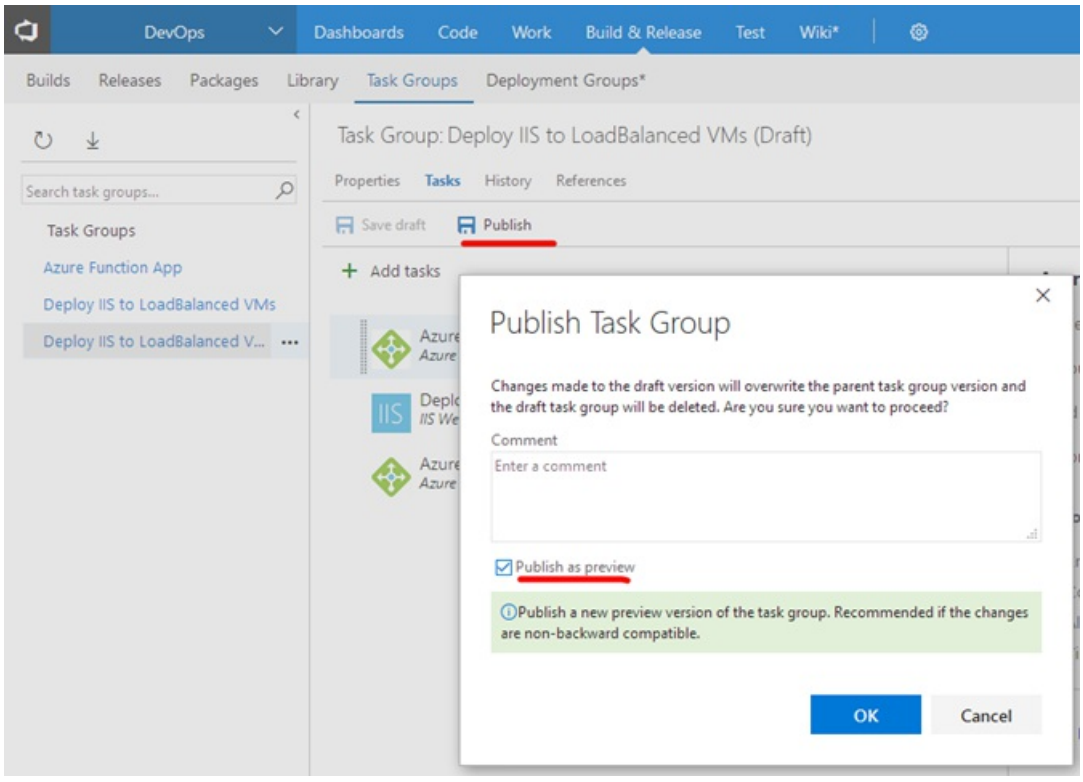
Task group versioning

When making changes to task groups, it can feel risky because the change is effective to all definitions that use the task group. With task group versioning, now you can draft and preview task group versions while still offering stable versions to your most important definitions until you are ready to switch. After some drafting and iteration, you can publish a stable version and, while publishing, if the changes are breaking in nature, you can choose to publish the task group as preview (a new major version). Alternatively, you can publish it directly as an updated stable version *(Figure 81)*.

When a new major (or preview) version of the task group is available, the definition editor will advise you that there is a new version. If that major version is preview, you even see a "try it out" message. When the task group comes out of preview, definitions using it are auto-updated, sliding along that major channel *(Figure 82)*.



(Figure 81) Save task group as draft

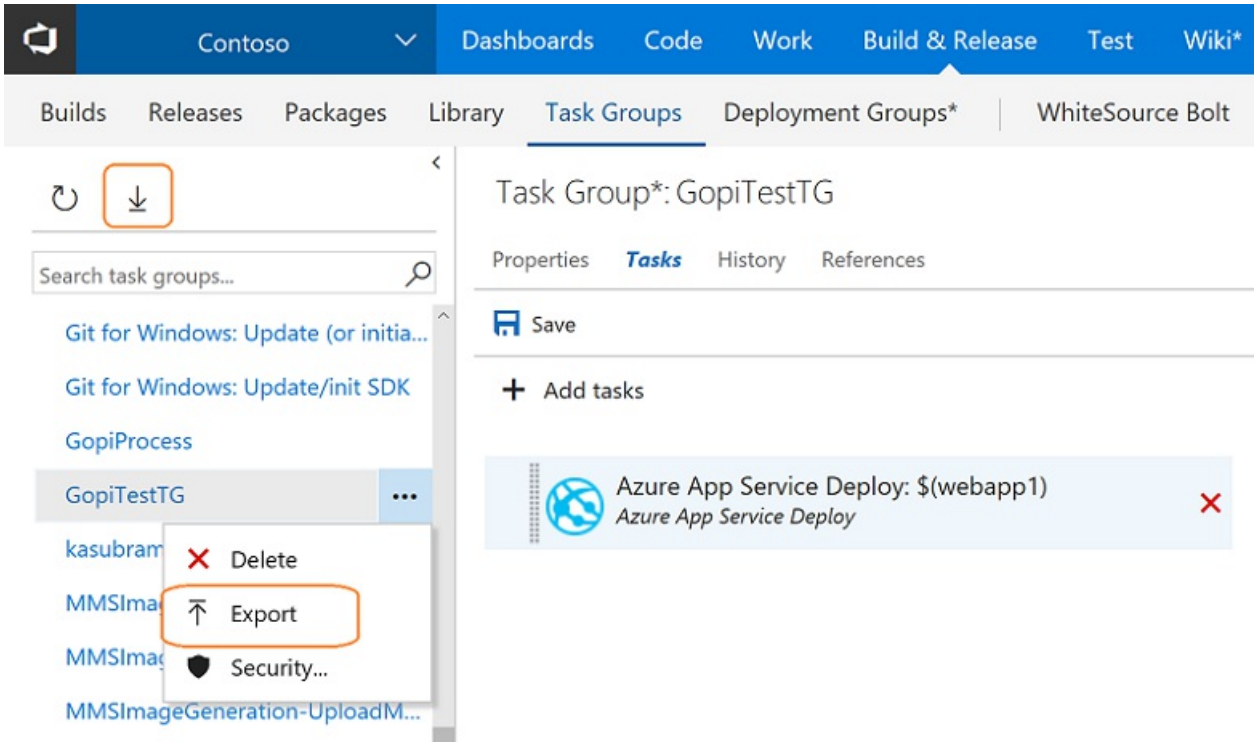


(Figure 82) Publish task group as preview

Task group import and export

Although task groups have enabled reuse within a project, we know that recreating a task group across projects and accounts can be painful. With task group import/export *(Figure 83)*, as we have done for release definitions,

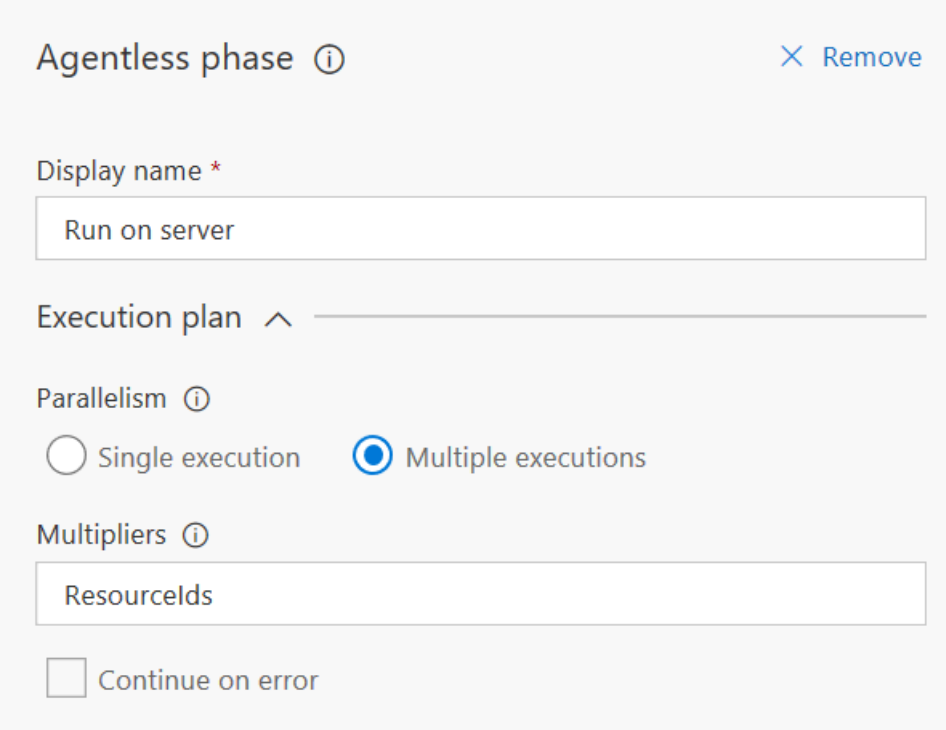
now you can export as a JSON file and import where you want it. We have also enabled nested task groups, which first expand when they are exported.



(Figure 83) Export task group

Multi Configuration support in Server Side (Agentless) tasks

By specifying variable multipliers for server side (agentless) tasks *(Figure 84)*, you can now run the same set of tasks in a phase on multiple configurations, which run in parallel.

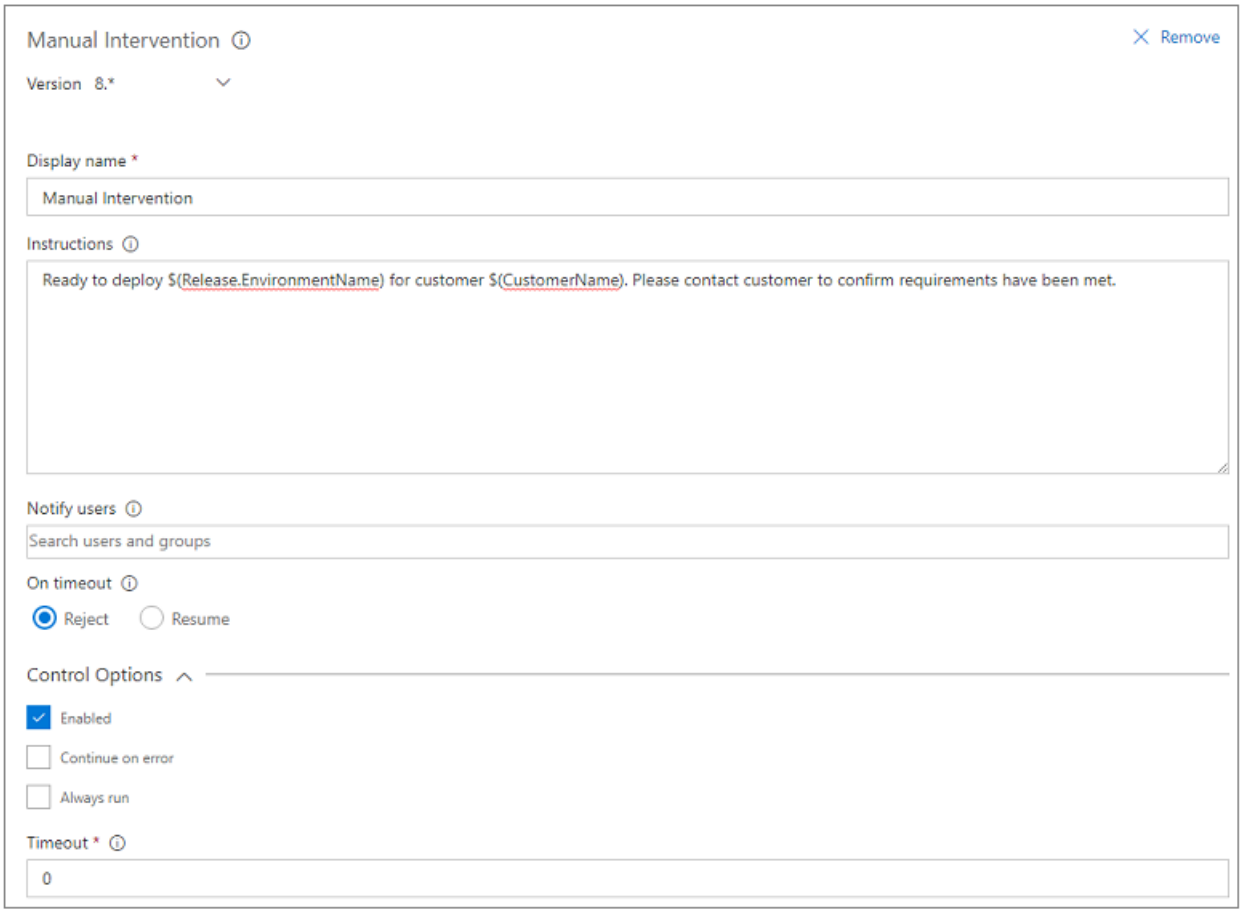


(Figure 84) Multi configuration of agentless tasks

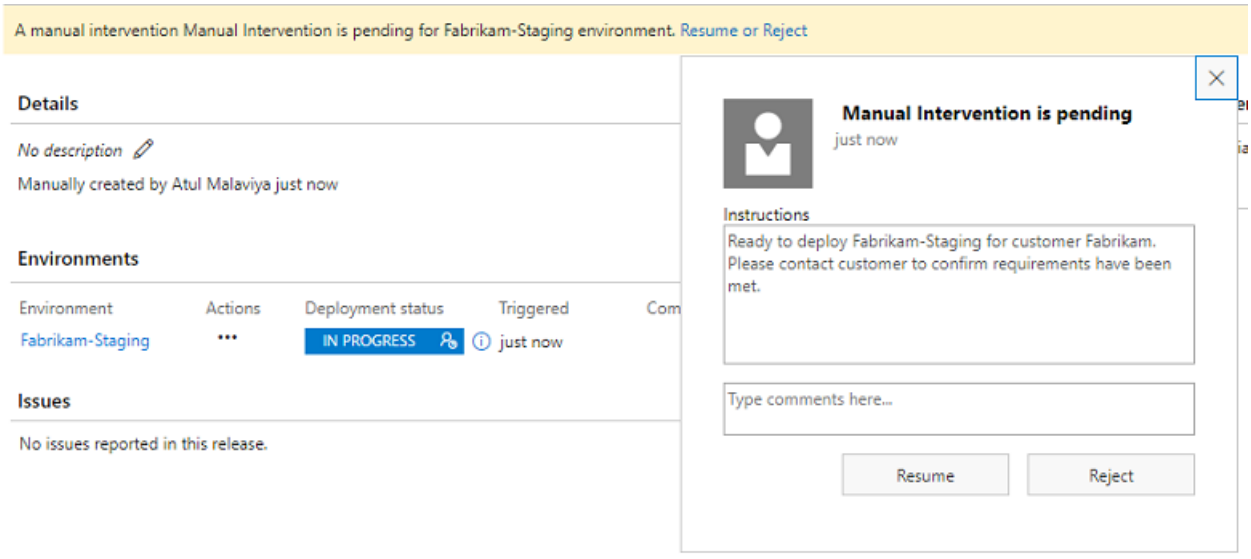
Variables Support in Manual Intervention task

The **Manual Intervention** task *(Figure 85)* now supports the use of variables within the instruction text shown to users when the task runs, at the point where the user can resume execution of the release process or reject it. Any variables defined and available in the release can be included, and the values are used in the notifications as well as

in the email sent to users *(Figure 86)*.



(Figure 85) Manual intervention task



(Figure 86) Manual intervention pending dialog

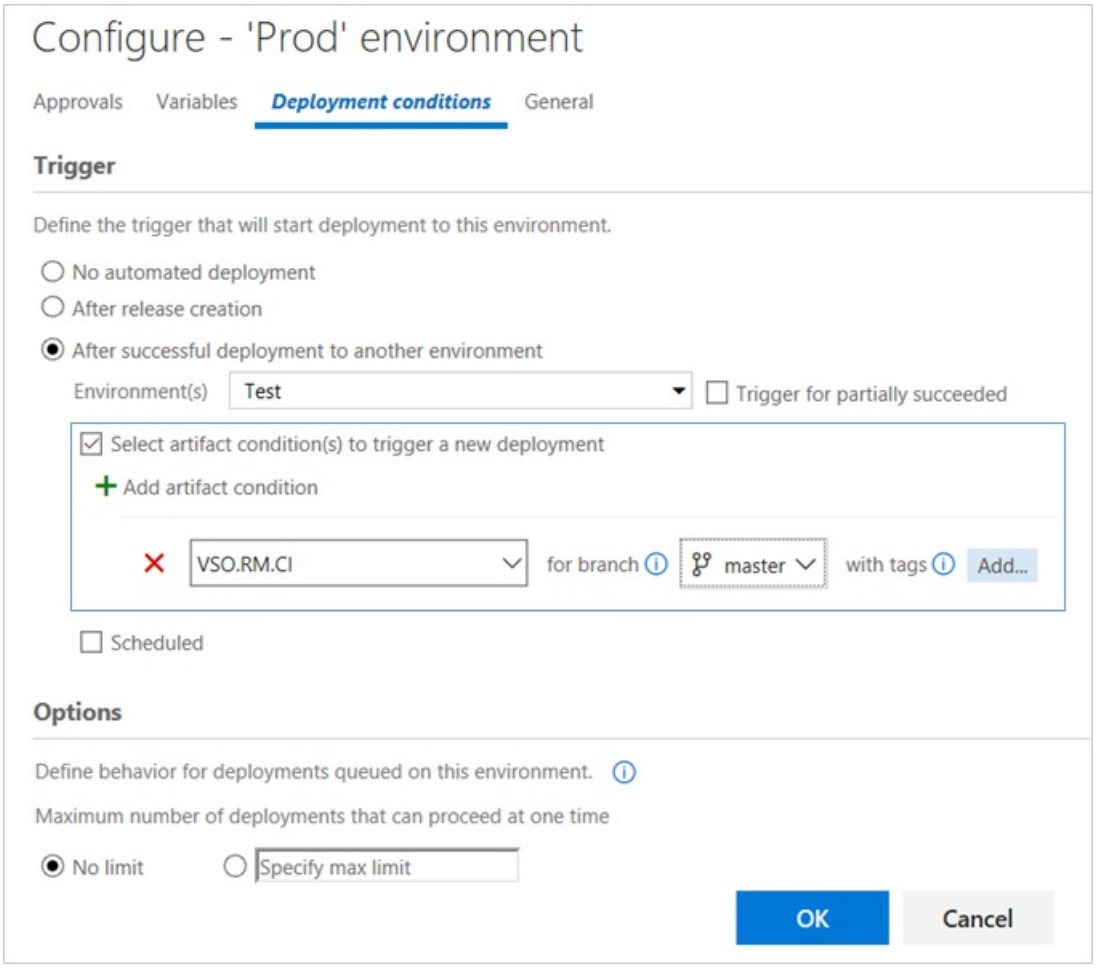
Control releases to an environment based on the source branch

A release definition can be configured to trigger a deployment automatically when a new release is created, typically after a build of the source succeeds. However, you may want to deploy only builds from specific branches of the source, rather than when any build succeeds.

For example, you may want all builds to be deployed to Dev and Test environments, but only specific builds deployed to Production. Previously you were required to maintain two release pipelines for this purpose, one for the Dev and Test environments and another for the Production environment.

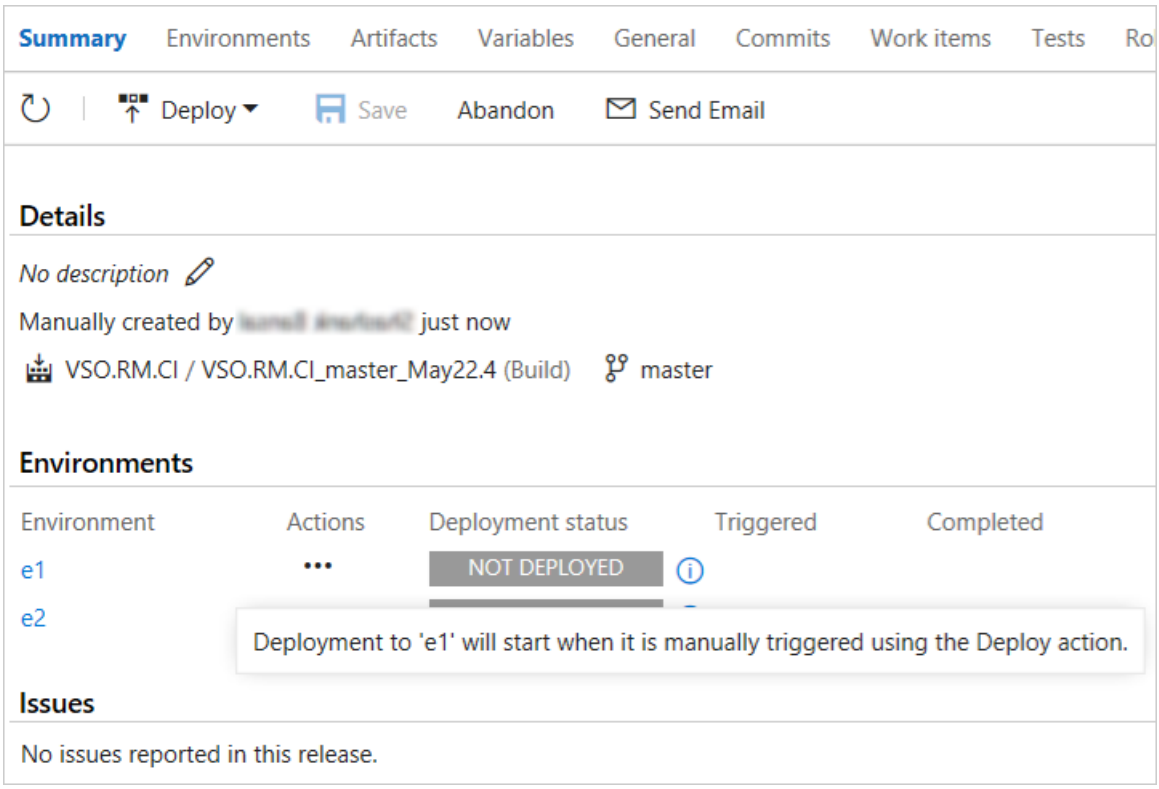
Release Management now supports the use of artifact filters for each environment. This means you can specify the

releases that are deployed to each environment when the deployment trigger conditions (such as a build succeeding and creating a new release) are met. In the **Trigger** section of the environment **Deployment conditions** dialog *(Figure 87)*, select the artifact conditions such as the source branch and tags for builds that will trigger a new deployment to that environment.



(Figure 87) Deployment conditions dialog

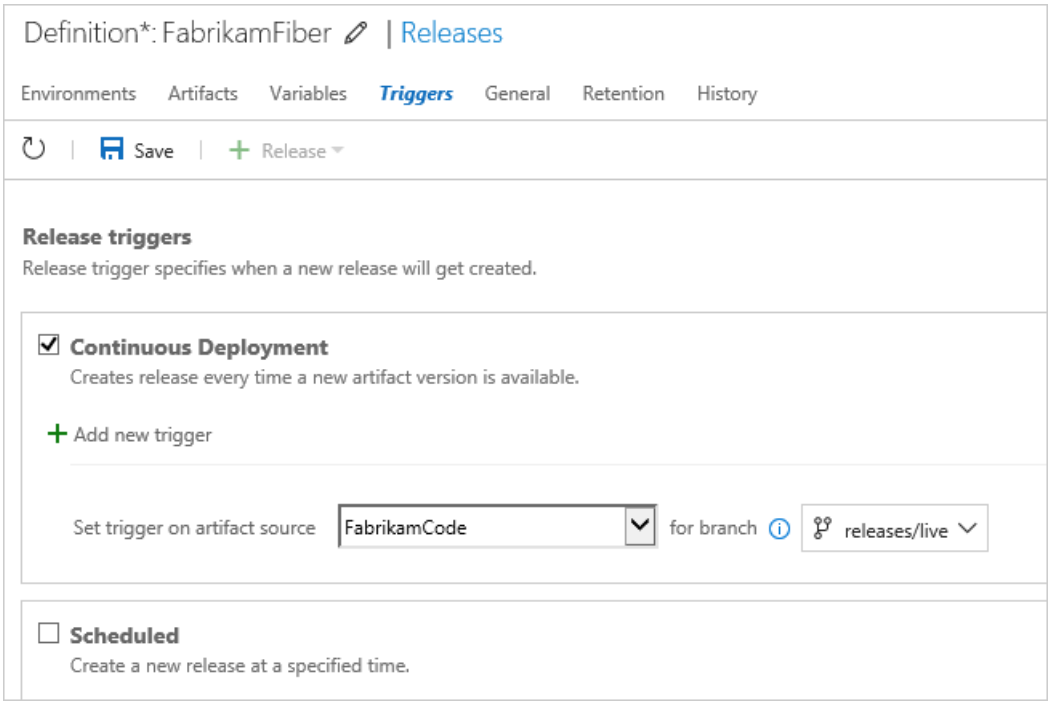
In addition, the **Release Summary** page *(Figure 88)* now contains a pop-up tip that indicates the reason for all "not started" deployments to be in that state, and suggests how or when the deployment will start.



(Figure 88) Release summary tip

Release Triggers for Git repositories as an artifact source

Release Management now supports configuring a continuous deployment trigger *(Figure 89)* for Git repositories linked to a release definition in any of the team projects in the same account. This lets you trigger a release automatically when you make a new commit to the repository. You can also specify a branch in the Git repository for which commits will trigger a release.



(Figure 89) Release triggers

Release Triggers: Continuous deployment for changes pushed to a Git repository

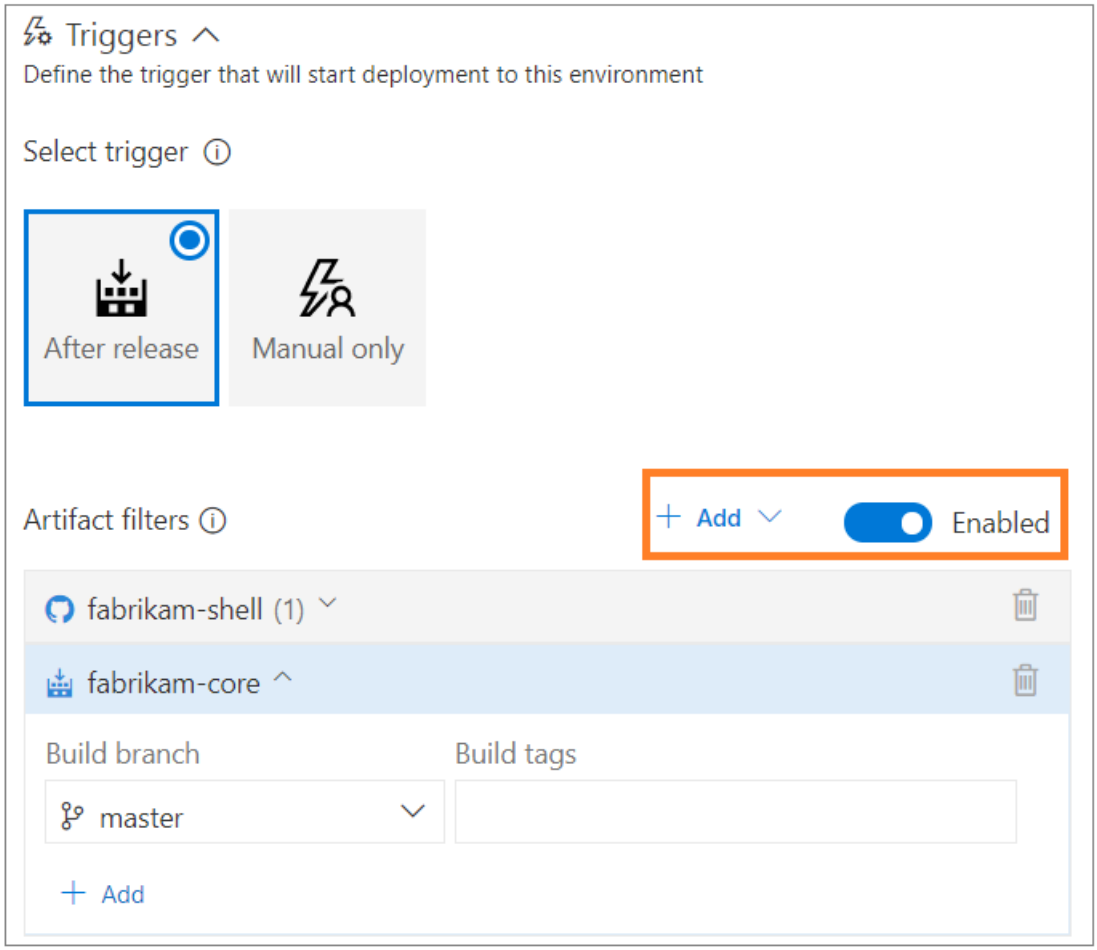
Release Management has always provided the capability to configure continuous deployment when a build completes. However, now you can also configure continuous deployment on Git Push. This means that you can link GitHub and Team Foundation Git repositories as artifact sources to a release definition, and then trigger releases automatically for applications such as Node.JS and PHP that are not generated from a build and so do not need a

build action for continuous deployment.

Branch filters in environment triggers

In the new release definition editor you can now specify artifact conditions for a particular environment. Using these artifact conditions, you will have more granular control on which artifacts should be deployed to a specific environment. For example, for a production environment you may want to make sure that builds generated only from the master branch are deployed. This filter needs to be set for all artifacts that you think should meet this criterion.

You can also add multiple filters for each artifact that is linked to the release definition *(Figure 90)*. Deployment is triggered to this environment only if all the artifact conditions are successfully met.

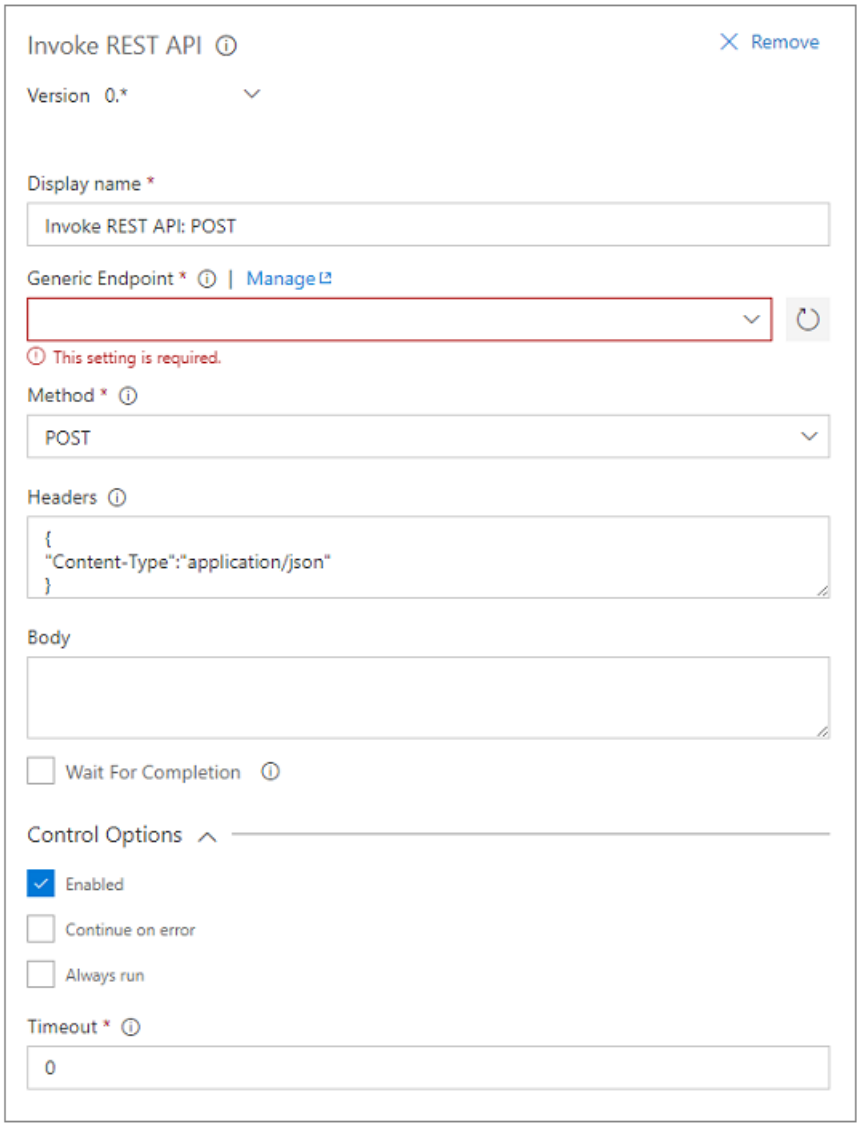


(Figure 90) Branch filters

Enhancements to server-side tasks

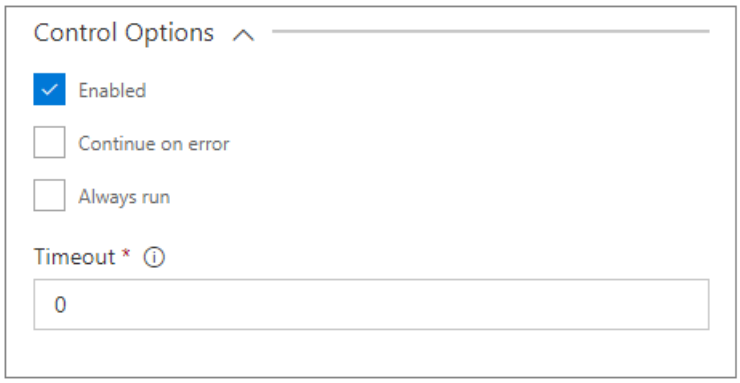
We have made two enhancements to server-side tasks (tasks that run within a server phase).

We have added a new task that invokes any generic HTTP REST API *(Figure 91)* as part of the automated pipeline. For example, it can be used to invoke specific processing with an Azure function, and wait for it to be completed.



(Figure 91) REST API task

We have also added a **Control options** *(Figure 92)* section to all server-side tasks. Task behavior now includes setting the **Enabled**, **Continue on error**, **Always run**, and **Timeout** options.

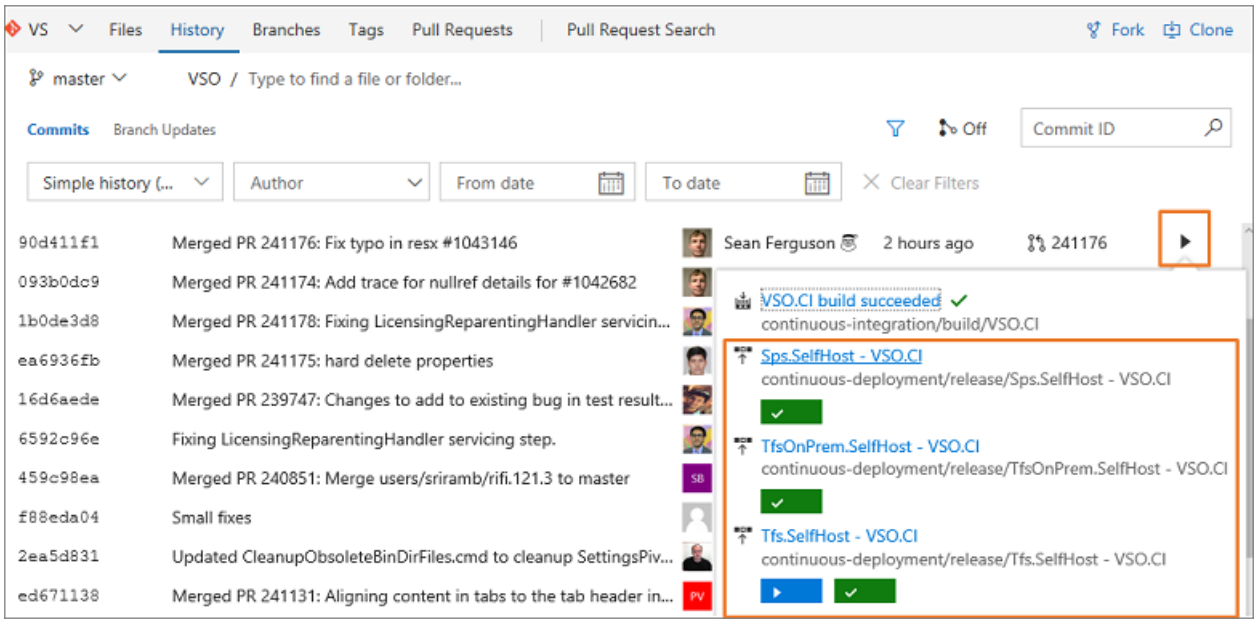


(Figure 92) Task control options

Release status badge in Code hub

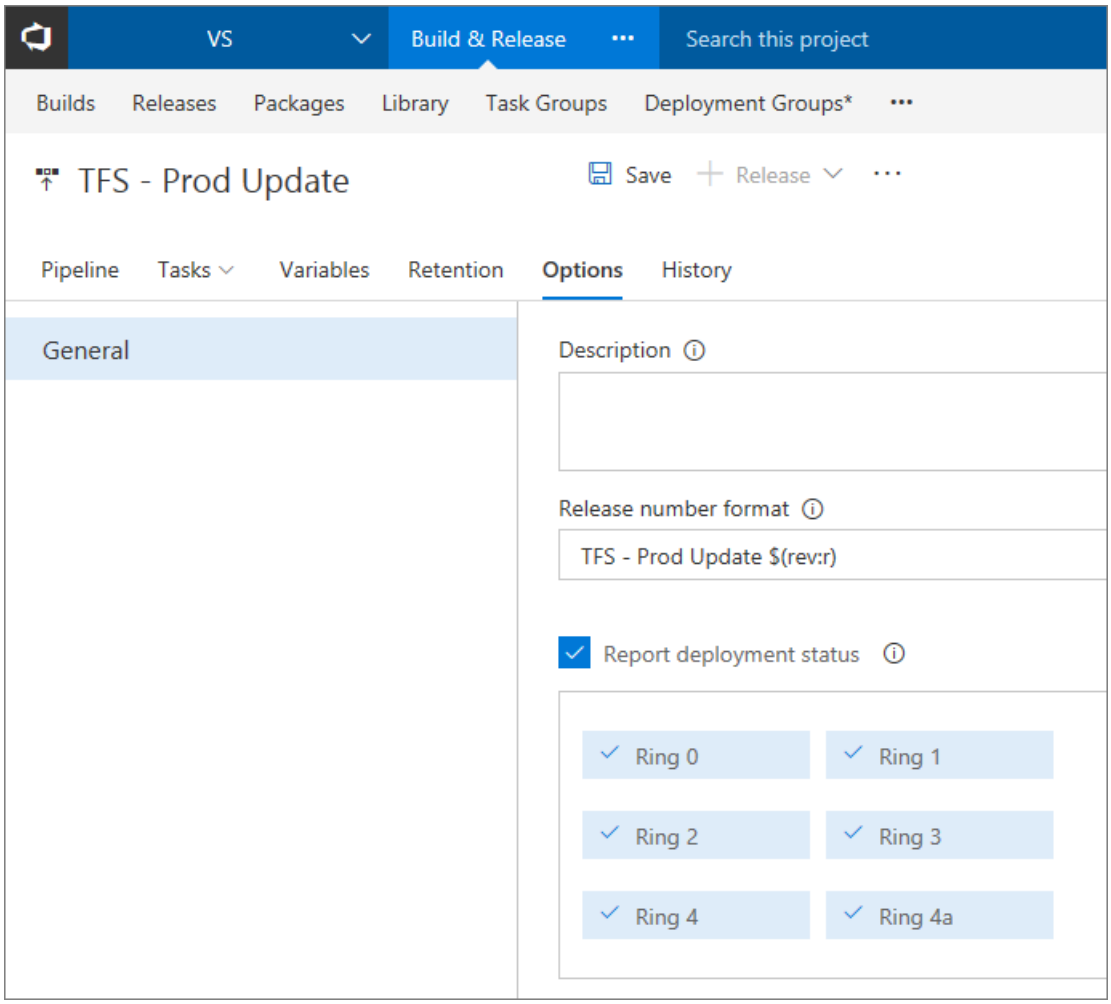
Today, if you want to know whether a commit is deployed to your customer production environment, you first identify which build consumes the commit and then check all the release environments where this build is deployed. Now this experience is much easier with the integration of the deployment status in the **Code** hub status badge to show the list of environments that your code is deployed to. For every deployment, status is posted to the latest commit that was part of the deployment. If a commit is deployed to multiple release definitions (with multiple environments) then each has an entry in the badge, with status shown for each environment *(Figure 93)*.

This improves the traceability of code commit to deployments.



(Figure 93) Release status badge

By default, when you create a release definition, deployment status is posted for all environments. However, you can selectively choose the environments whose deployment status should be displayed in the status badge (e.g. only show production environments) *(Figure 94)*.

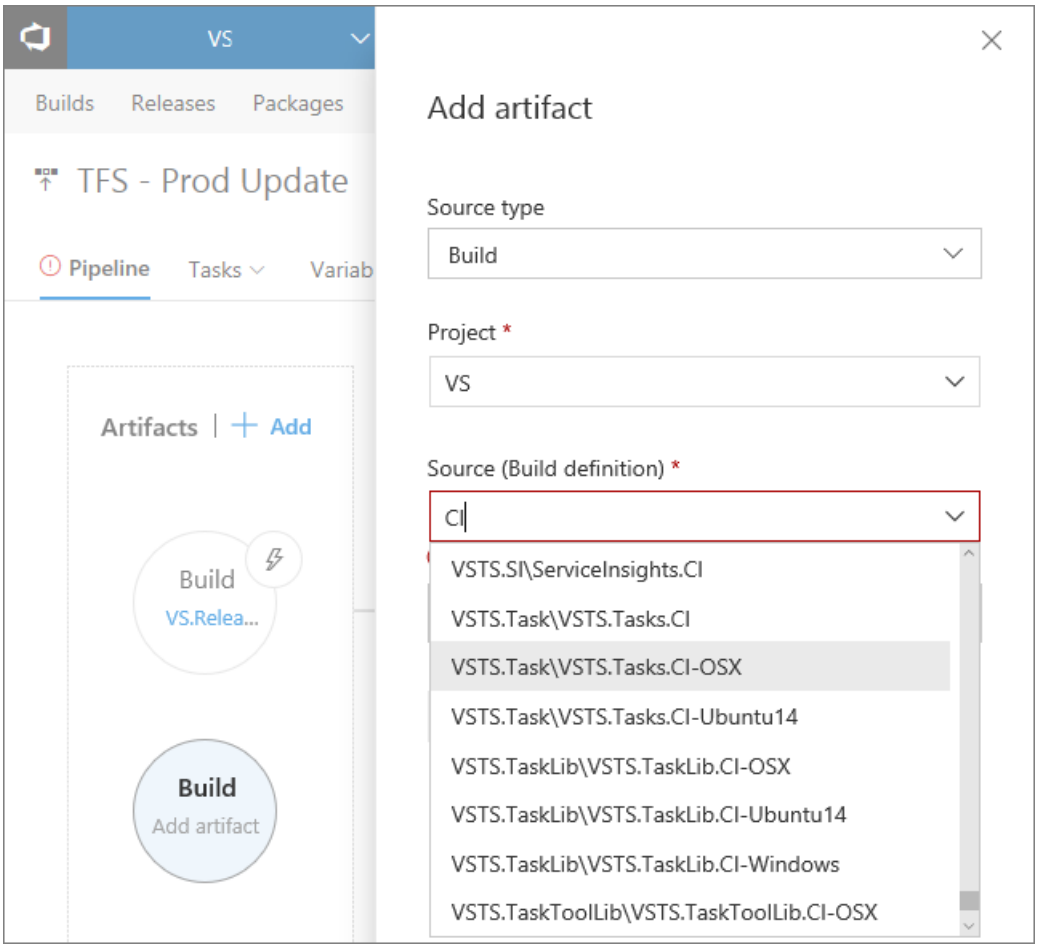


(Figure 94) Deployment options dialog

Enhancements to Build definition menu when adding artifacts

When adding build artifacts to a release definition, you can now view the definitions with their folder organization

information and simplify choosing the desired definition *(Figure 95)*. This makes it easy to differentiate the same build definition name but in different folders.

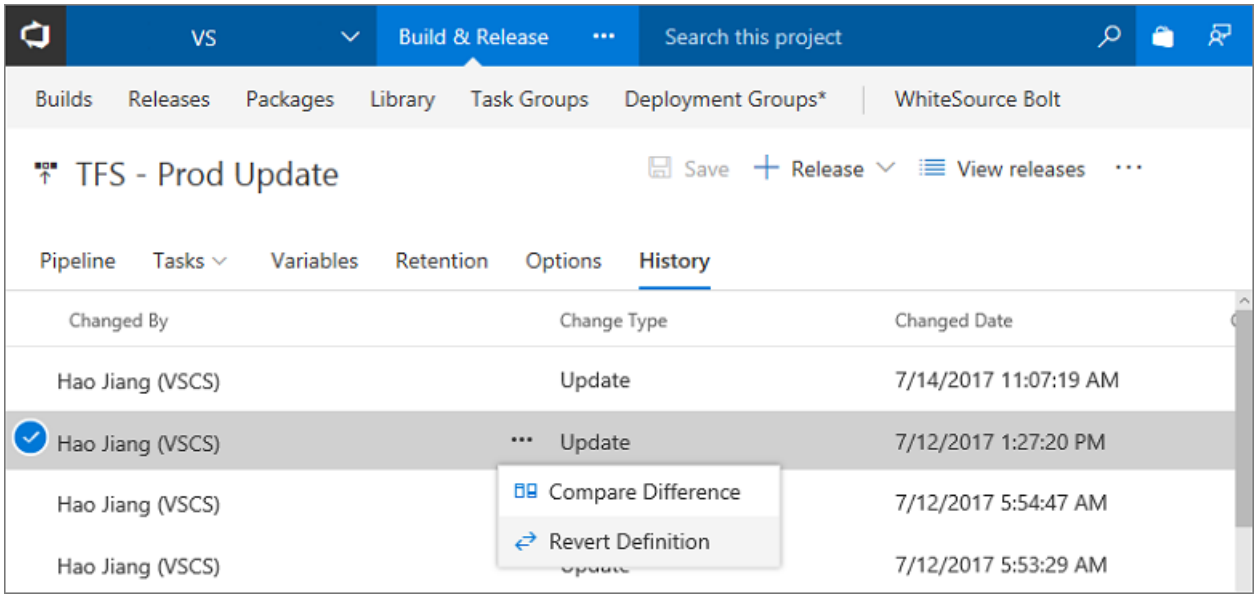


(Figure 95) Add artifact

The list of definitions is filtered based on those that contain the filter term.

Revert your release definition to older version

Today, if a release definition is updated, you cannot directly revert to a previous version. The only way is to look into the release definition history to find the diff of the changes, and then manually edit the release definition. Now, using the **Revert Definition** feature *(Figure 96)*, you can choose, and revert back to any older version of a release definition from the release definition **History** tab.



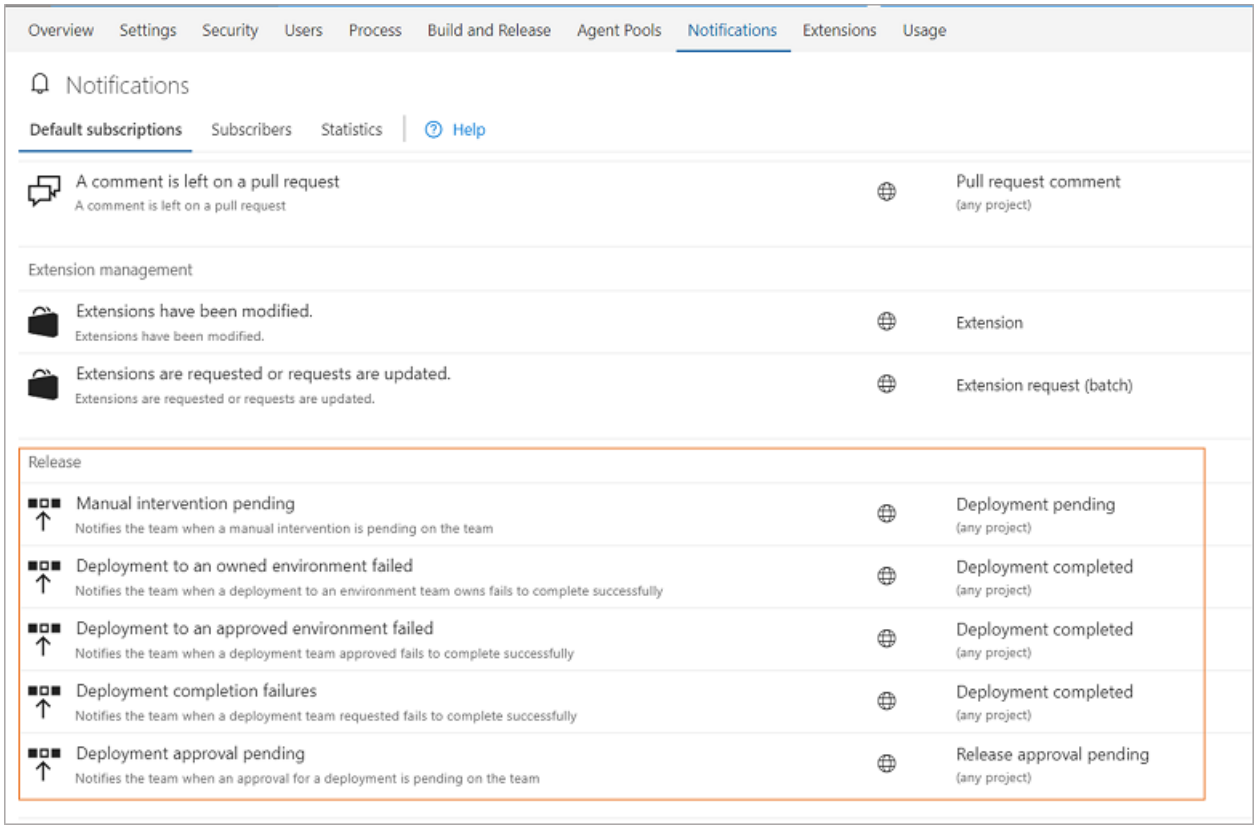
(Figure 96) Revert release definition

Personalized notifications for releases

Release notifications are now integrated with the VSTS notification settings experience. Those managing releases are now automatically notified of pending actions (approvals or manual interventions) and important deployment failures. You can turn off these notifications by navigating to the **Notification** settings under the profile menu and switching off **Release Subscriptions**. You can also subscribe to additional notifications by creating custom subscriptions. Admins can control subscriptions for teams and groups from the **Notification** settings under **Team** and **Account** settings.

Release definition authors no longer need to manually send emails for approvals and deployment completions.

This is especially useful for large accounts that have multiple stakeholders for releases, and for those other than the approver, release creator, and environment owner that might want to be notified *(Figure 97)*.



(Figure 97) Release notifications

> **TIP**
>
> See the post for managing release notifications for more information.

Testing

Removing support for Lab Center and automated testing flows in Microsoft Test Manager

With the evolution of Build and Release Management, XAML builds are no longer supported in TFS 2018 and consequently we are updating support for using Microsoft Test Manager (MTM) with TFS. Using Test Center/Lab Center in MTM for automated testing is no longer supported by TFS, starting with TFS 2018. If you are not ready to migrate away from XAML builds and Lab Center, you should not upgrade to TFS 2018.

Please see the impact of upgrading to TFS 2018 below:

Lab Center:

- No longer supported:
  - Test Controllers cannot be registered with TFS for creating and managing Lab environments.
  - Any existing Test Controllers registered with TFS will go offline and existing Lab environments will appear as 'Not Ready'.

- Recommended alternative:
  - You can connect to your SCVMM server using SCVMM TFS Extension, create and manage virtual machines and run your workflows on it. For more details see How to perform Lab management operations in Build and Release.

Automated Testing:

- No longer supported:
  - Automated testing workflows that rely on Test controller and Lab environments such as XAML Build-Deploy-Test workflow or running automated tests from a test plan using MTM are no longer supported.
- Recommended alternatives:
  - You can run automated tests in Build and Release.
  - You can run automated tests from a test plan using the Test hub.

Manual Testing:

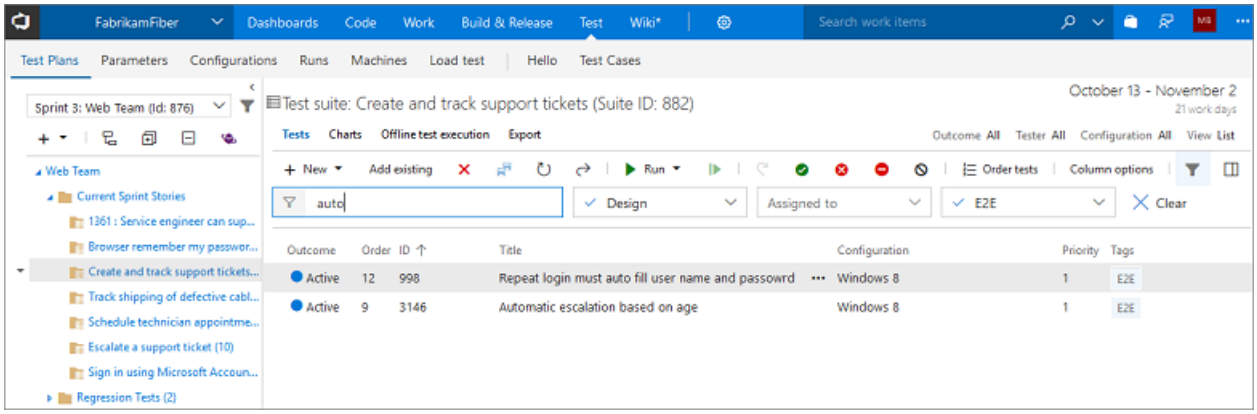
- All manual testing scenarios continue to be fully supported. While manual tests can be run using MTM with TFS 2018, Lab Environments cannot be used to run manual tests.
- For all manual testing scenarios, we strongly recommend using the Test hub in TFS web access.

Exploratory testing traceability improvements for work item links, iterations, and area paths

Based on the feedback we received from teams doing exploratory testing, we are improving traceability links while filing bugs, tasks, or test cases from the Test & Feedback extension. Bugs and tasks created while exploring requirements are now created with the same area path and iteration as that of the requirement instead of team defaults. Test cases created while exploring requirements will now be linked with a **Tests <-> Tested By** link instead of the **Parent <-> Child** link so that the test cases you create are automatically added to requirement based test suites. Finally, work items created while not specifically exploring any requirement are filed in the team's default iteration instead of the current iteration so that no new work items come into the current iteration after the sprint planning is complete.

Filters for Test Case work items in Test Plans and Suites in Test Hub

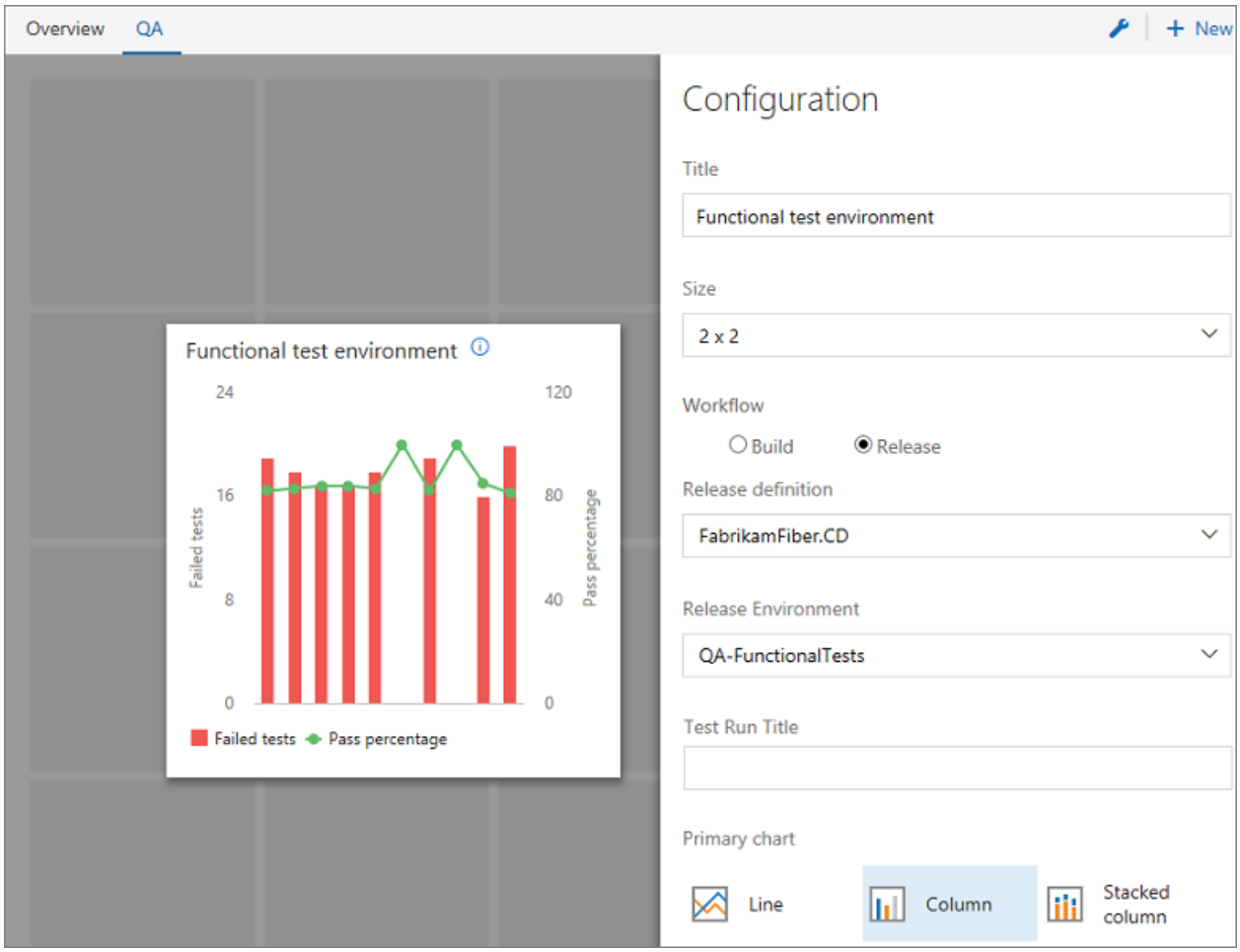
In addition to the filters on **Test** fields like **Outcome**, **Configuration**, and **Tester**, you can now filter on Test Case work item fields like **Title**, **State**, and **Assigned To** *(Figure 98)*.



(Figure 98) Test case filters

Test trend charts for Release Environments and Test Runs

We are adding support for **Release Environments** in the **Test Result Trend** widget *(Figure 99)* so that you can track status of test environments on VSTS dashboards. Like the way you to do for test results in **Build**, you can now create trend charts showing test pass rate, count of total, passed or failed tests, and test duration for **Release Environments**. You can also filter the charts to a specific test run within an environment with the **Test Run** title filter.



(Figure 99) Test trend chart

Markdown formatting support for Test Run and Test Result comments

We are adding support for formatting **Test Run** and **Test Result** comments with markdown syntax. You can use this feature to create formatted text or quick links to URLs in your comments. You can update **Test Result** comments in the **Result Summary** page with **Update analysis** and **Test Run** comments in the **Run Summary** page with **Update comments** in **Test** hub.

Add link to existing bug for a failing test

While analyzing the test result in the **Build** or **Release** summary page or in the **Test** hub, you can now associate an existing bug to a failed test. This is helpful when a test is failing for a known reason that has a bug already filed.

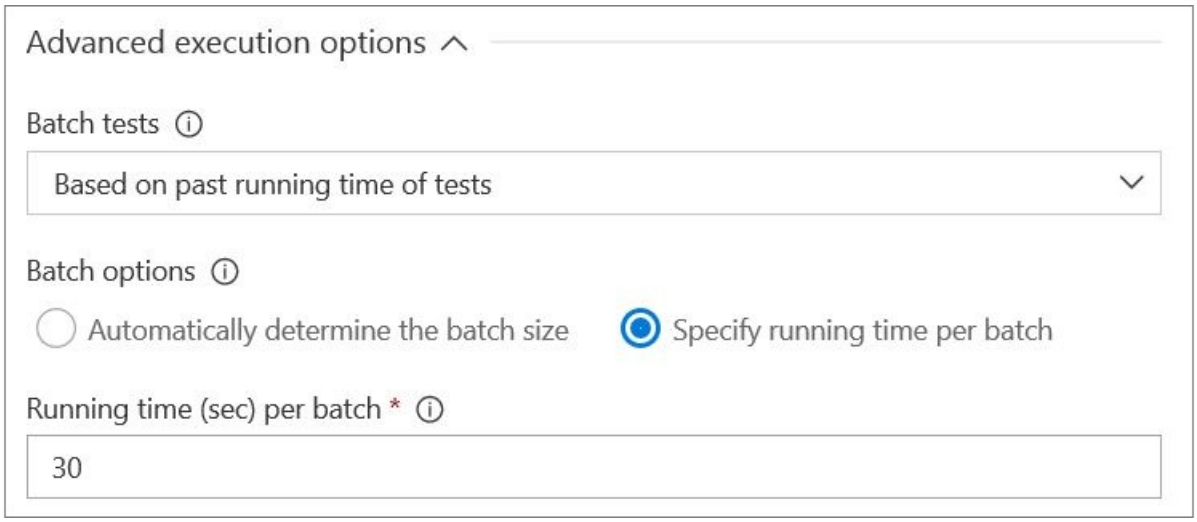
Upload attachments to test runs and test results

You can now attach files such as screenshots and log files to test runs or test results as additional information. Up to this point, this capability was only available through the Microsoft Test Manager (MTM) client, forcing you to switch context between the **Test** hub in VSTS/TFS and the MTM client.

Test batching

In the Visual Studio test task in Build/Release management, options are available to control how tests should be grouped (batched) for efficient execution. Tests can be grouped in two ways:

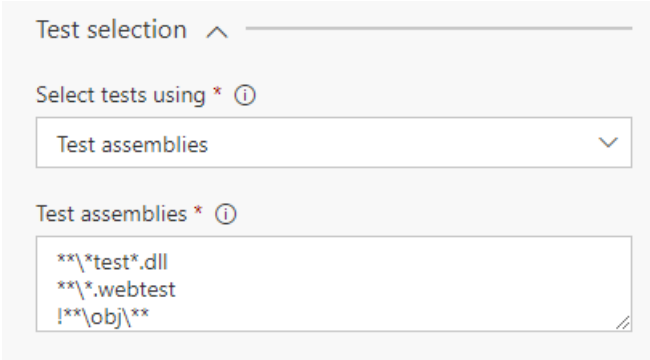
1. Based on the number of tests and agents participating in the run, which simply groups tests into a number of batches of a specified size.
2. Based on past running time of tests, which considers past running time to create batches of tests such that each batch has approximately equal running time *(Figure 100)*. Quick running tests are batched together while longer running tests may belong to a separate batch. This option can be combined with the multi-agent phase setting to reduce the total test time to a minimum.



(Figure 100) Test batching

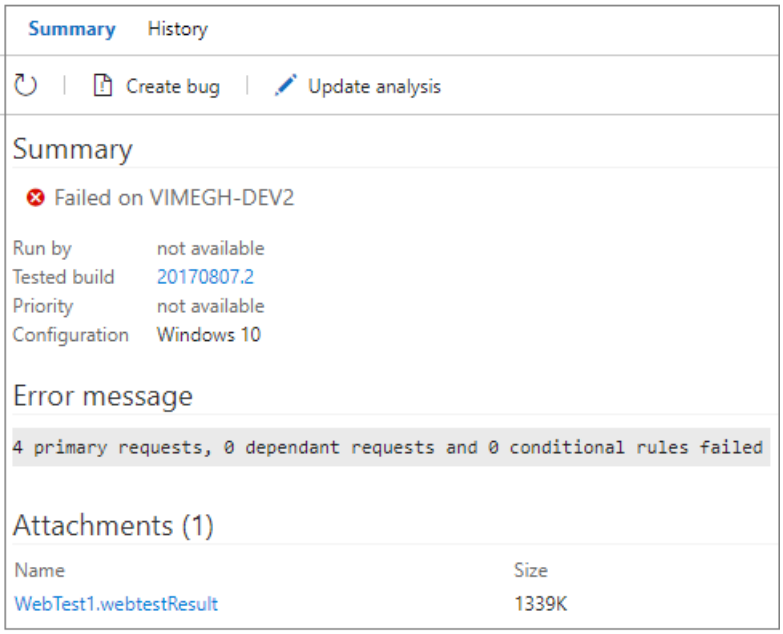
Run webtests using the VSTest task

Using the Visual Studio test task, webtests, also known as web performance tests, can now be run in the CI/CD pipeline. Webtests can be run by specifying the tests to run in the task assembly input. Any test case work item that has an "associated automation" linked to a webtest, can also be run by selecting the test plan/test suite in the task *(Figure 101)*.



(Figure 101) Test selection

Webtest results are available as an attachment to the test result *(Figure 102)*. This can be downloaded for offline analysis in Visual Studio.



(Figure 102) Test summary

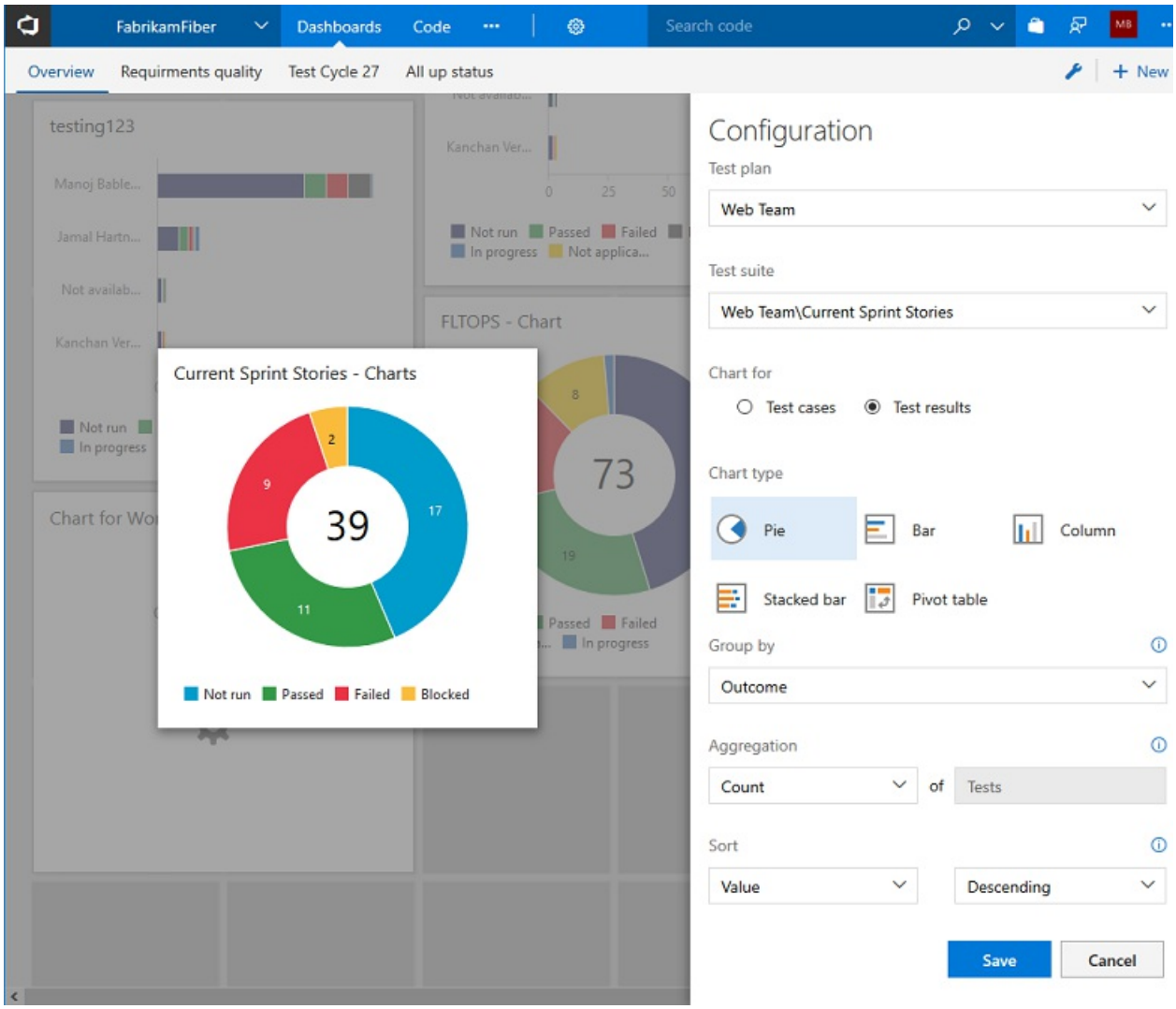
This capability is dependent on changes in the Visual Studio test platform and requires that Visual Studio 2017 Update 4 is installed on the build/release agent. Webtests cannot be run using prior versions of Visual Studio.

Similarly, webtests can be run using the **Run Functional Test** task. This capability is dependent on changes in the Test Agent, that will be available with the Visual Studio 2017 Update 5.

> **TIP**
> See the Load test your app in the cloud using Visual Studio and VSTS quickstart as an example of how you can use this together with load testing.

Chart widget for test plans and test suites

Previously, you could create charts for test plans and suites in **Test** hub and pin them to dashboard. We have now added a widget that enables creating charts for test plans and suites from the widget catalog on the dashboard. You can create charts for test authoring status or test execution status. Moreover, adding charts from the widget allows you to create larger charts when you have more data to be shown on a chart *(Figure 103)*.



(Figure 103) Chart widget

Screenshot and annotation support for desktop apps with Chrome browser for manual tests

We are adding support for one of the top suggestions from manual testing - capturing screenshots of desktop applications from the **Web Test Runner** in **Test** hub. Until now, you had to use **Test Runner** in **Microsoft Test Manager** to capture screenshots of desktop apps. You need to install the Test & Feedback extension to use this functionality. We are rolling out support for the Chrome browser, and Firefox will follow shortly thereafter.

Discontinuing TFS Extension for SharePoint

TFS 2018 and later versions will no longer support the TFS Extension for SharePoint. Additionally, the screens used to configure integration between a TFS Server and a SharePoint server have been removed from the Team Foundation Administration Console.

> **NOTE**
> If you are upgrading to TFS 2018 from a previous version configured to integrate with SharePoint, you will need to disable the SharePoint integration after upgrade, or your TFS SharePoint sites will fail to load.

We have created a solution that allows you to disable integration on the SharePoint server. For more information, please see the post on the future of our TFS/SharePoint Integration.

Discontinuing Team Rooms

Modern development teams heavily depend on collaboration. People want (and need) a place to monitor activity (notifications) and talk about it (chat). A few years back, we recognized this trend and set out to build the Team Room to support these scenarios. Since that time, we have seen more solutions to collaborate emerge in the market. Most notably, the rise of Slack. And more recently, the announcement of Microsoft Teams.

With so many good solutions available that integrate well with TFS and Visual Studio Team Services, we announced in January the plans to remove our Team Room feature from both TFS 2018 and Visual Studio Team Services.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2018 Release Notes History

9/7/2018 • 2 minutes to read

Release Notes History

- Team Foundation Server 2018 Update 3 Release Notes
- Team Foundation Server 2018 Update 2 Release Notes
- Team Foundation Server 2018 Update 1 Release Notes
- Team Foundation Server 2018 Release Notes

Team Foundation Server 2017 Update 3 Release Notes

9/7/2018 • 4 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

This is not the latest version of Team Foundation Server. To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2017 Update 3. Click the button to download.

Team Foundation Server 2017

To learn more about Team Foundation Server 2017, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: February 28, 2018

Details of What's New in Team Foundation Server 2017 Update 3.1

This update fixes potential cross-site scripting (XSS) and other security vulnerabilities. See the blog post for more information. It is a full upgrade, so you can upgrade directly to TFS 2017 Update 3.1.

Release Date: November 6, 2017

Details of What's New in Team Foundation Server 2017 Update 3

This is an update for Team Foundation Server 2017 that includes bug fixes since Team Foundation Server 2017 Update 2.

- Work
- Code
- Build
- Release
- Test
- Reporting
- Administration

Work

- Exporting a template with ASCII character code >127 does not have WebLayout and includes incorrect file names.
- Board and Card Settings does not handle Work Item Type rename.
- Kanban board card reordering in Turkish should be by stack rank.
- REST API WorkItemSearchConditionalFaultIn should throw NotSupportedException for Search.
- StructureChangeNotification event not firing for WIT.
- Add link to more info when exceeding max links limit.
- Optimize work item field data migration execution plan to improve upgrade performance.
- Assigned To context menu should use combo string instead of display name to disambiguate.

Code

- Maven: Code coverage will now be generated.
- HTML files will no longer default to Preview mode in the new Explorer.
- We have added a scroll bar when viewing changesets.
- We fixed the vertical scrolling in Web (both Code->Changesets as Files) will now work in IE 11/Chrome.
- Scrolling now works in Source Explorer (IE/Edge).
- You can now filter changesets for users who have left the project.
- You are now able to select a file, then select back in the root directory, of the left tree, and remain in full screen mode.
- The search URL will no longer throw an exception when it exceeds the defaulted supported length if there are a large number of repositories.
- File folders are no longer configured if there is no default branch in the Git Repository.
- Extension installs will not conflict with the jobs of previous extension uninstall operations.
- We fixed Search and it now works despite job failures.
- ReindexingStatus will no longer remain in an Inprogress state if the Accountfaultin job is run more than once.
- TFVC crawl will no longer fail due to version control (VC) permission issues.
- The Search function will no longer fail post upgrade to TFS 2017 Update 2 in the Japanese build.
- The Search function will no longer fail as it did when upgrading from TFS Update 1 to TFS Update 2.
- The Job Result message will now give more insights into the indexing.
- We have reduced the Patch operation failure count.
- TimeBoxed Crawler will now crawl at least one batch irrespective of job execution time limit.
- We fixed patch operation KPIs for failed files to be consistent with logs for Search.
- We improved the support for SSH protocol version in Git for Windows.

Build

- Error while attempting to register build agent: Authentication - "Insufficient stack to continue executing the program safely."
- Update vsts-agent to carry new version of git.
- Enable NuGet Tool Installer task in order to allow for NuGet.exe 4.3.0 to be included in builds for compatibility with .NET.
- RequestedFor field of scheduled XAML build is not set.

Release

- Upgrade from TFS 2017 failing migration of Azure-based connected service-to-service endpoint.

Test

- Deploy Test Agent task has multiple issues on Win7-SP1 machine.
- If the Test Agent path is incorrect, it is not logged as error, but appears only in debug.

- Test run/task should not fail if an attachment upload failed.
- GetTestPlatform API will look for a testplatform.config file in the probing directory based on configuration.
- Drop indexes on dbo.tbl_TestResult to significantly improve upgrade performance.
- Improve performance of test results migration servicing step to significantly improve upgrade performance.

Reporting

- The RDL Burndown Reports now shows the correct amount of time for deleted tasks.

Administration

- Users page Public URL should match the calling URL such as when a load balancer is involved.

Known Issues

Work item forms do not render correctly in the web

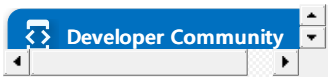
- **Issue:**

  If you have a custom control, such as the multi-value control, installed for the Visual Studio client but not the web client, work item forms in the web fail to render.

- **Workaround:**

  You will need to update to the latest version of your control. It is necessary to add a web layout that does not contain the missing control element. You can find the latest multi-value control for TFS 2017 Update on the Custom Controls for TFS Work Item Tracking page. For more information on the layout, see All FORM XML elements reference (TFS 2015) page.

See customer-reported issues reported for Team Foundation Server 2017.



Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2017 Update 2 Release Notes

9/7/2018 • 35 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

This is not the latest version of Team Foundation Server. To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2017 Update 2. Click the button to download.

Team Foundation Server 2017

To learn more about Team Foundation Server 2017, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: July 24, 2017

Summary of What's New in Team Foundation Server 2017 Update 2

We have added a lot of new value to Team Foundation Server 2017 Update 2. Some of the highlights include:

- Work items now have icons associated with each work item type.
- We have introduced Delivery Plans.
- You can search for work items using Work Item Search.
- There is a new branch policies configuration experience.
- There have been many pull request improvements.
- There are now Git graphs to visualize your Git history.
- You can now add and view Git tags.
- We have a new Package Management experience.
- There is a new build definition editor experience.
- Various updates when deploying to Azure Web Apps.
- Many improvements when deploying containers.
- We have introduced conditional build tasks.
- There are now out-of-the-box notifications.

You can see the details of all the new features by viewing the improvements by feature area:

- Work item tracking

- Version control
- Pull requests
- Package Management
- Build and release
- Testing
- Warehouse
- Administration
- Microsoft Teams integration

Details of What's New in TFS 2017 Update 2

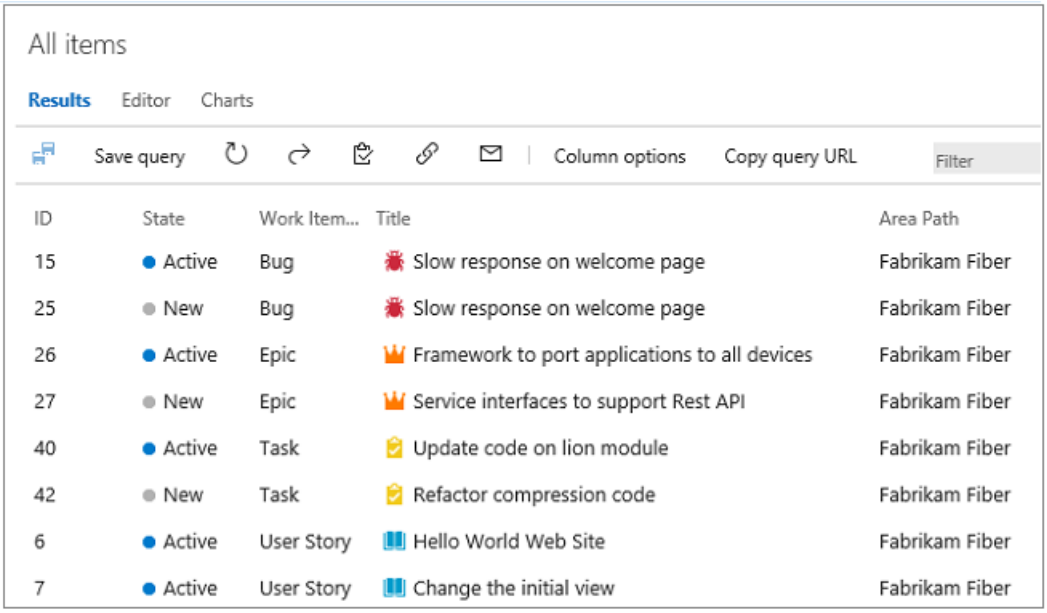
Work Item Tracking Improvements

Work item type icons

We have made a global commitment to make our products fully accessible to our customers. As part of that commitment, we have been working to find and address many accessibility issues – anywhere from keyboard patterns to visual design and layout.

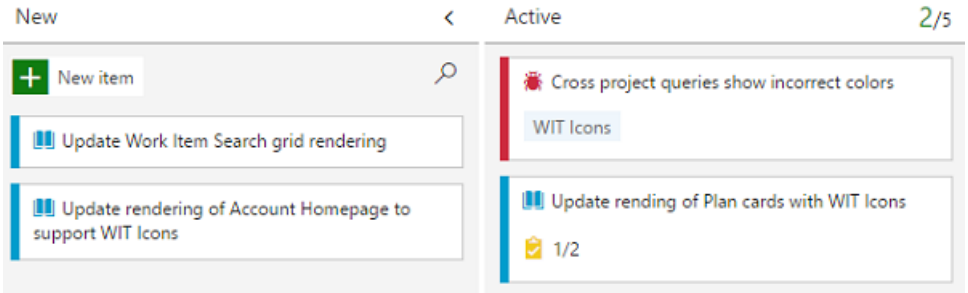
Work item tracking has relied solely on color in many experiences to convey work item type. However, this is problematic for our color blind or low vision users who may not be able to distinguish between items due to similarities in color. To increase the scanability of work item types for all our customers, we have introduced icons to the visual language of work item type. You can customize your work item types by choosing from a selection of our icon library.

Color bars conveying type on the backlog and queries grids have been replaced with colored icons *(Figure 1)*.



(Figure 1) Colored icons in query

Cards on the board now include a type icon *(Figure 2)*.



(Figure 2) Board with icon type

Delivery Plans

Delivery Plans is an organizational tool that helps you drive cross-team visibility and alignment by tracking work status on an iteration-based calendar. You can tailor your plan to include any team or backlog level from across projects in the account. Furthermore, **Field Criteria** on Plans enables you to further customize your view, while **Markers** highlight important dates.

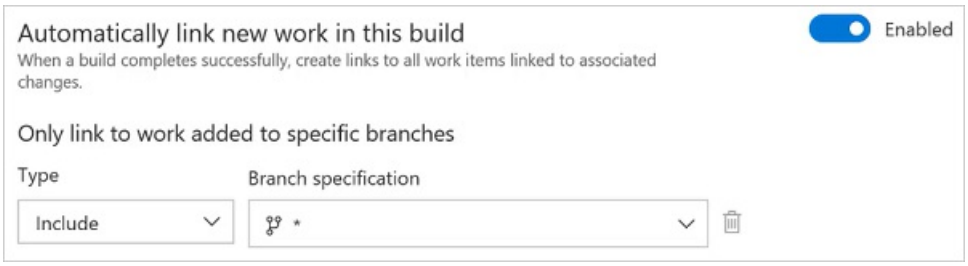
Check out the marketplace page for Delivery Plans to learn more and install the extension.

For users with a TFS instance that is disconnected from the Internet, Delivery Plans are available directly from the **Manage extensions** option in web access, without navigating to the VSTS Marketplace. From **Manage extensions**, click on **Browse local extensions**, then select **Delivery Plans** and click **Install**. See the documentation on pre installed extensions for more information.

Automatic linking from work items to builds

With this new setting in the build definition, you can track the builds that have incorporated your work without having to search through a large set of builds manually. Each successful build associated with the work item automatically appears in the development section of the work item form.

To enable this feature, toggle the setting under **Options** in your build definition *(Figure 3)*.



(Figure 3) WIT build linking

Deprecation of old work item form

Overall feedback for the new work item form has been positive and we now have 100% adoption on our hosted accounts. We want on-premises customers to tap into the same value that has delighted our VSTS users and so we have made the decision to deprecate the old work item form and old extensibility model. Read more about our plans on the Microsoft Application Lifecycle Management page.

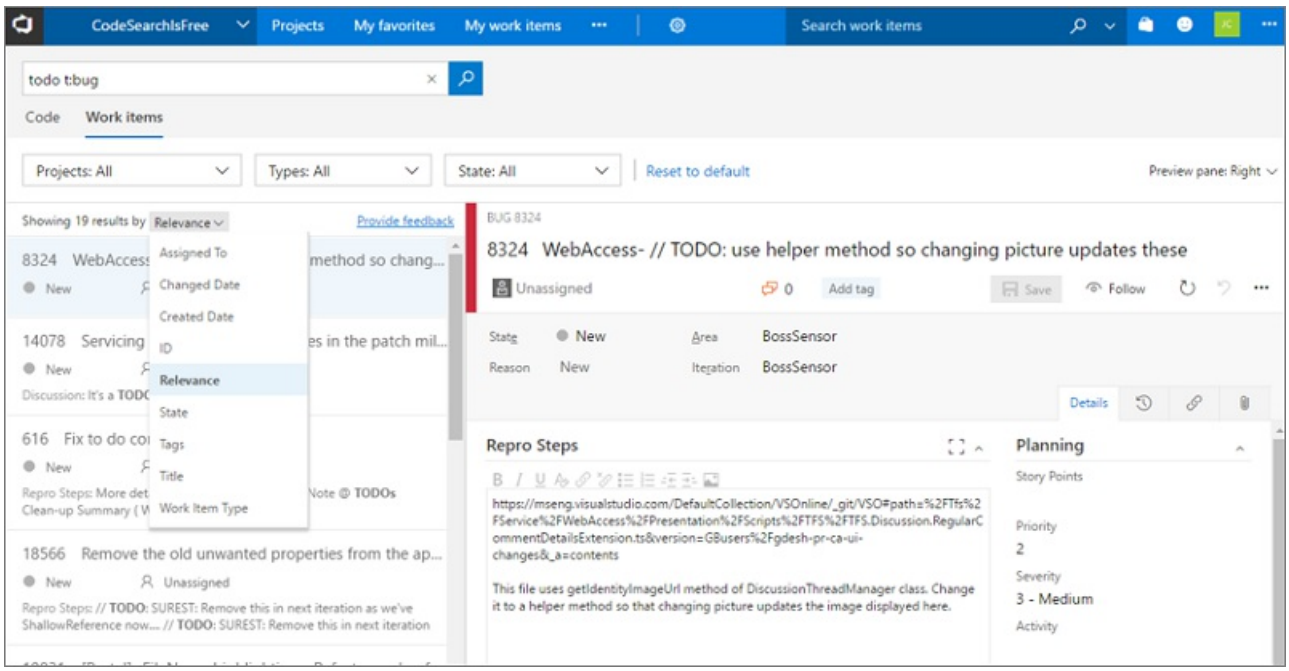
Work Item Search

Work Item Search provides fast and flexible search across all your work items over all projects in a collection *(Figure 4)*. You can use the Work Item Search full text search engine to easily search for terms across all work item fields and efficiently locate relevant work items. Use in-line search filters, on any work item field, to quickly narrow down to a list of work items.

Once the Search service is configured in TFS, you can get searching without the need to install anything else. By using Work Item Search you can:

- **Search over all your projects:** Search in your own and your partner teams' backlog. Use cross-project searches over all the work items to search across your organization's entire work items. Narrow your search by using project and area path filters.
- **Search across all work item fields:** Quickly and easily find relevant work items by searching across all work item fields (including ere fields). Use a full text search across all fields to efficiently locate relevant work items. The snippet view indicates where matches were found.
- **Search in specific fields:** Use the quick in-line search filters, on any work item field, to narrow down to a list of work items in seconds. The dropdown list of suggestions helps complete your search faster. For example, a search such as **AssignedTo:Chris WorkItemType:Bug State:Active** finds all active bugs assigned to a user named Chris.

- **Take advantage of integration with work item tracking:** The Work Item Search interface integrates with familiar controls in the Work hub, letting you view, edit, comment, share, and much more.

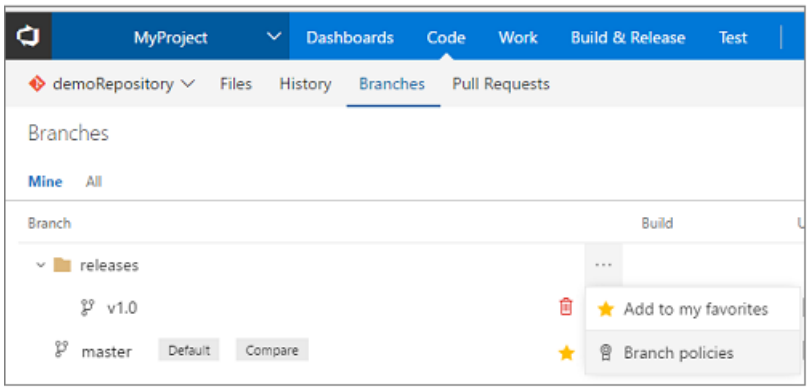


(Figure 4) Workitem search

Version Control Improvements

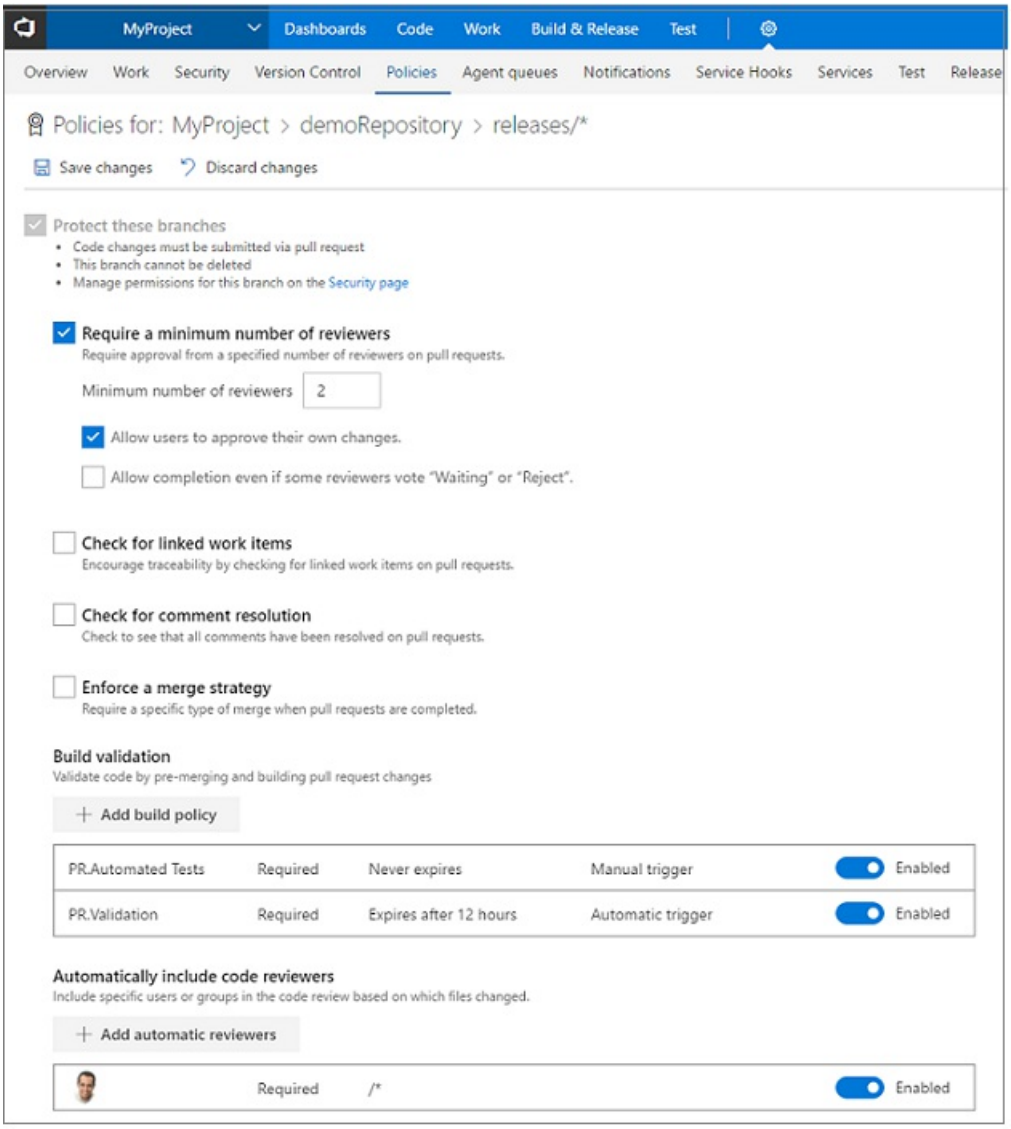
New branch policies configuration experience

We have redesigned the branch policies configuration experience and added some great new capabilities *(Figure 5)*. One of the most powerful features is the ability to configure policies for branch folders. You can do this from the **Branches** view by selecting a branch folder and choosing **Branch policies** from the context menu.



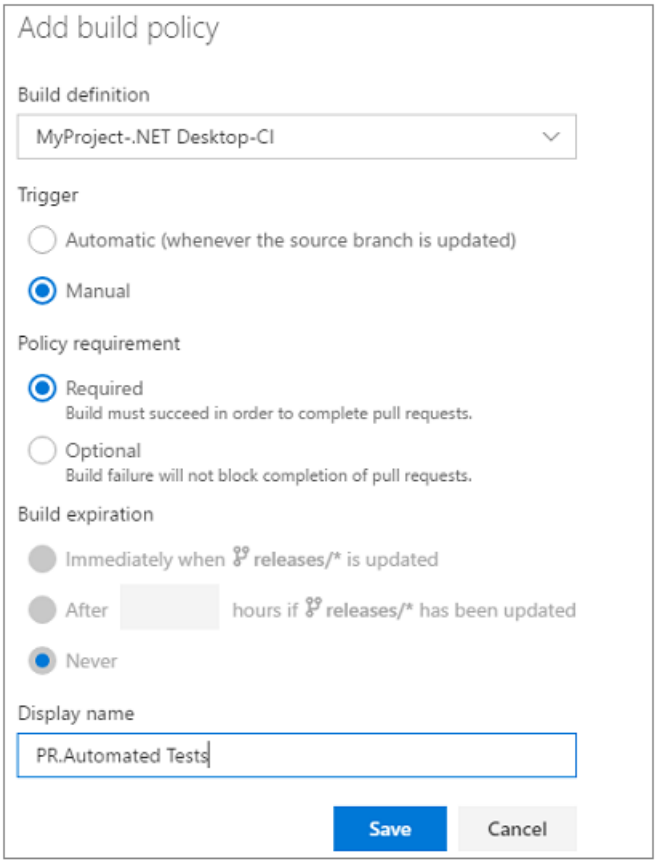
(Figure 5) Configure branch policies

This will open the new policies configuration UX, where you can configure policies that apply to all of the branches in the branch folder *(Figure 6)*.



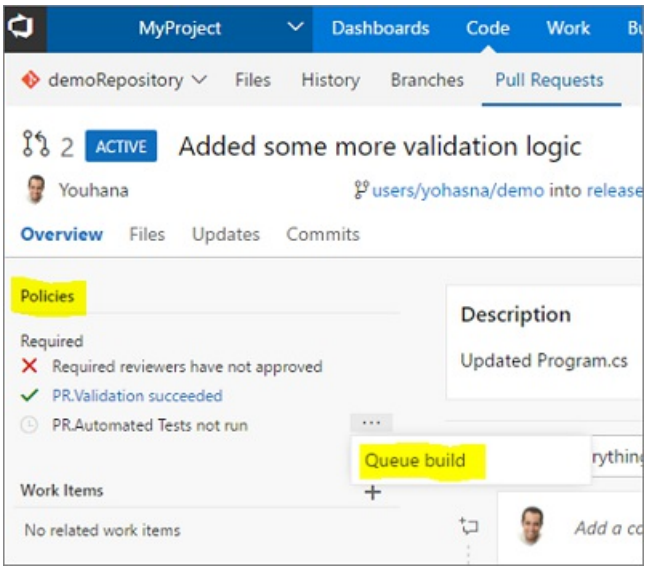
(Figure 6) Policies page

If you are using the build policy, you can now configure multiple builds for a single branch. There are also new options to specify an automatic or manual trigger *(Figure 7)*. Manual triggers are useful for things like automated test runs that might take a long time to run, and you only really need to run once before completing the pull request. The build policy also has a display name that is useful if you are configuring multiple builds.



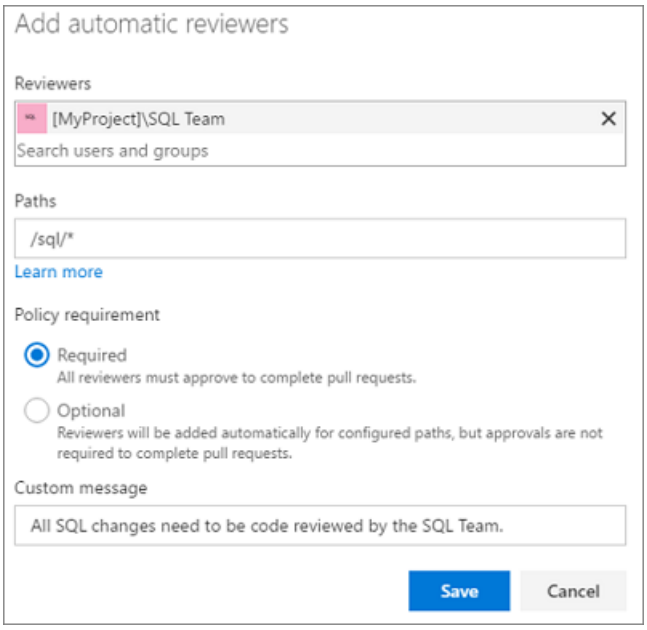
(Figure 7) Manual build

Once you have configured a manually triggered policy, you can run it by selecting the **Queue build** option in the **Policies** section for the pull request *(Figure 8)*.

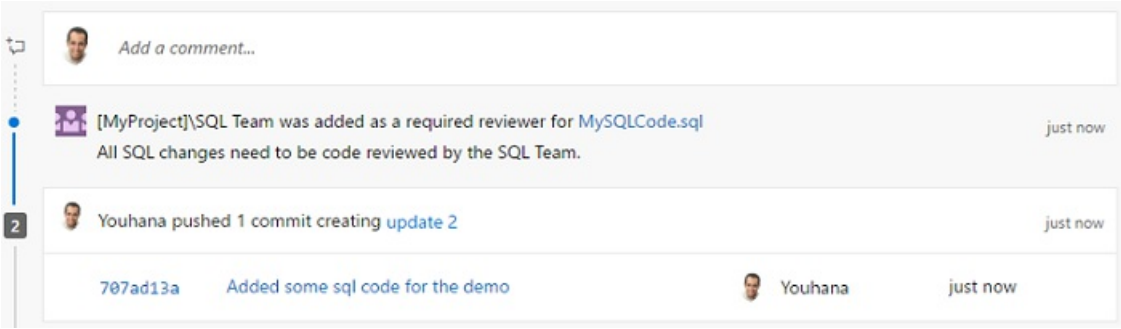


(Figure 8) Manual build queue

For required reviewer policies *(Figure 9)*, we added the ability for administrators to specify a note that can be appended to the pull request timeline when the policy applies *(Figure 10)*.



(Figure 9) Required reviewer dialog



(Figure 10) Required reviewer note

New policy for no active comments

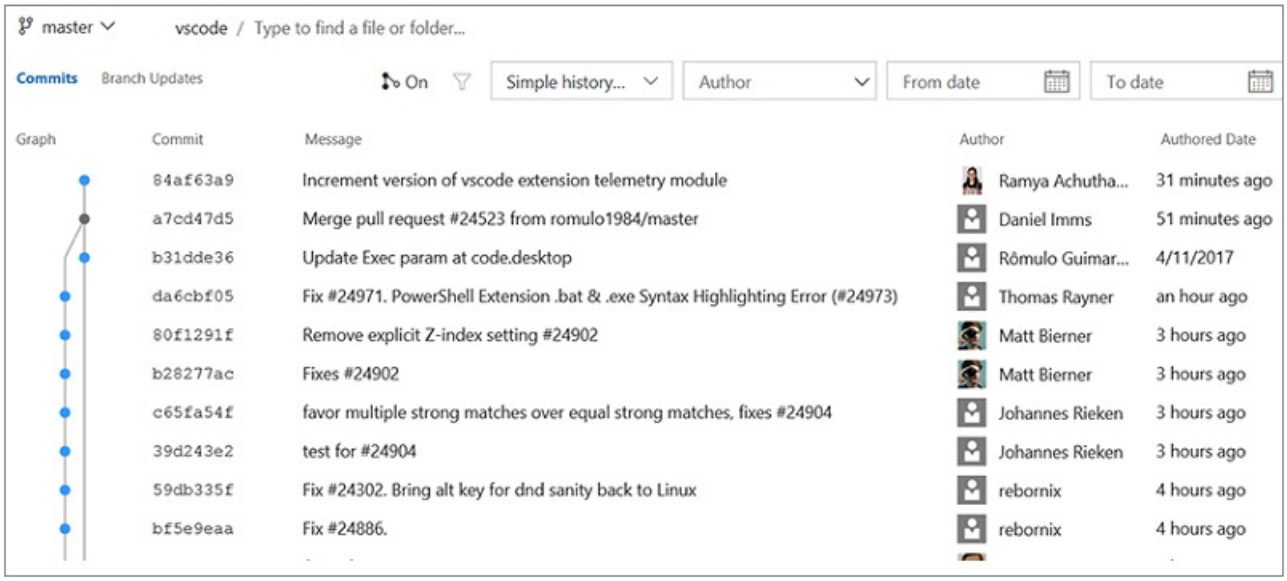
Ensure that all comments in your pull requests are addressed with the new **Comments** policy. With this policy enabled, active comments will block completion of the PR, forcing all comments to be resolved. Reviewers that leave comments for the PR author but optimistically approve the pull request can be sure that an author that is eager to merge won't miss any comments.

Files hub improvements

We have made several updates to the **Files** hub to improve the viewing and editing experiences.

For viewing, we have added pivots that let you view the README in the current folder *(Figure 11)*, preview Markdown files, compare a file to a previous version *(Figure 12)*, and view blame.

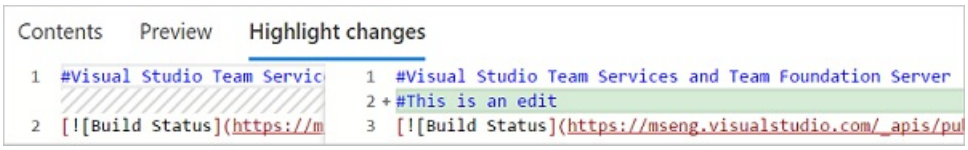
(Figure 11) Files viewing



(Figure 12) Git graph

>

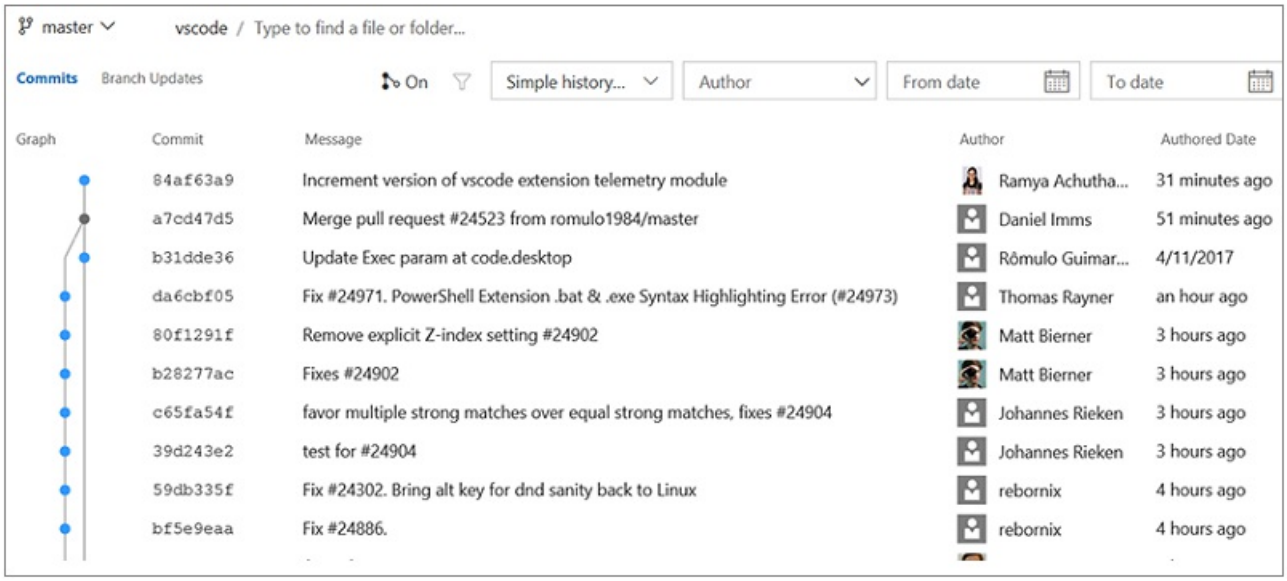
For editing, you can now preview your changes, easily add a comment, commit to a new branch, and link work items *(Figure 13)*.



(Figure 13) Files editing

Visualize your Git repository

You can now see a graph while showing commit history for repositories or files. This allows you to easily create a mental model of all your branches and commits for your git repositories using git graph *(Figure 14)*. The graph shows all your commits in topological order.

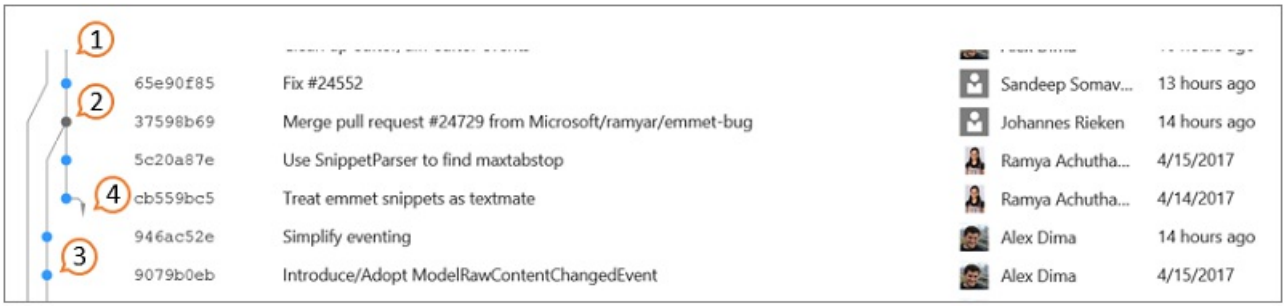


(Figure 14) Git graph

The key elements of Git graph include *(Figure 15)*:

1. Git graph is right-aligned, so commits associated with the default branch or the selected branch appear on the right while the rest of the graph grows on the left.

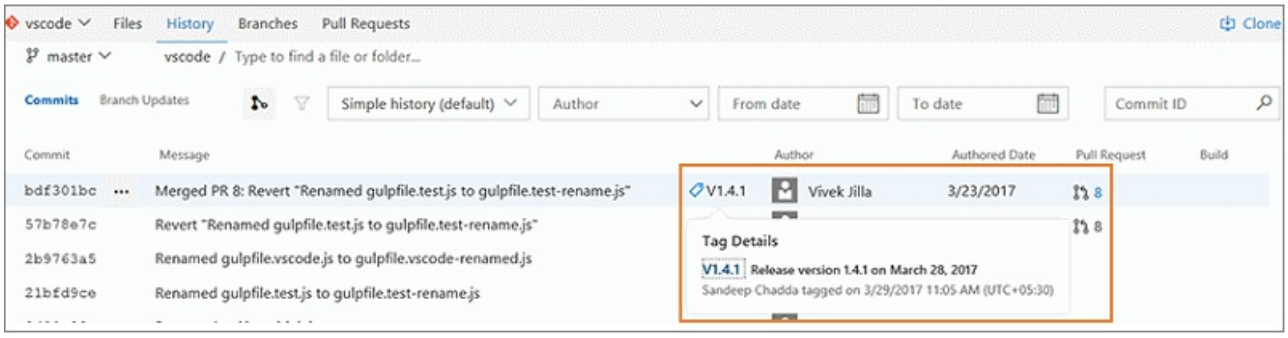
2. Merge commits are represented by grey dots connected to their first parent and second parent.
3. Normal commits are represented by blue dots.
4. If the parent commit of a commit is not visible in the view port on the next 50 commits, then we excise the commit connection. Once you click the arrow, the commit is connected to its parent commit.



(Figure 15) Git graph elements

View git tags on commits

If your team has been using Git tags to mark a specific point in the history of your repository, then your commits will now show the tags that you have created. You are able view tags *(Figure 16)* for a specific commit in the **commit list** view and the **details** page.



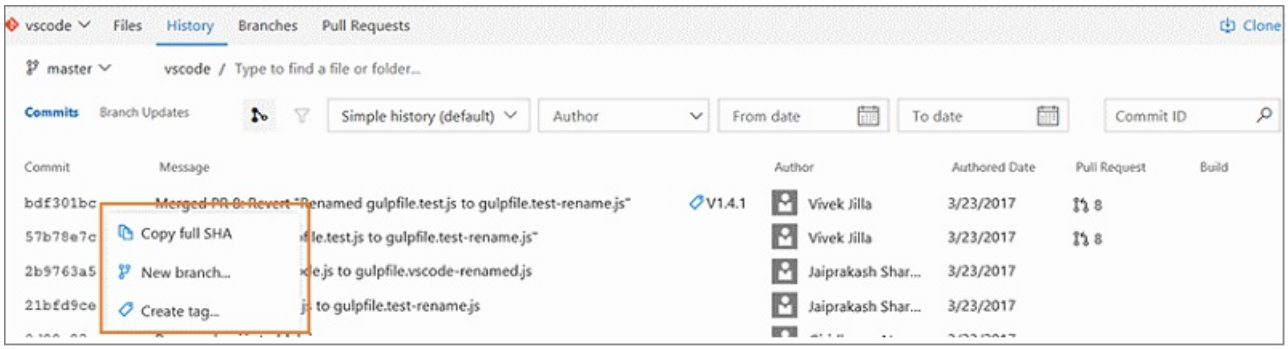
(Figure 16) Show tags

Add tags to commits

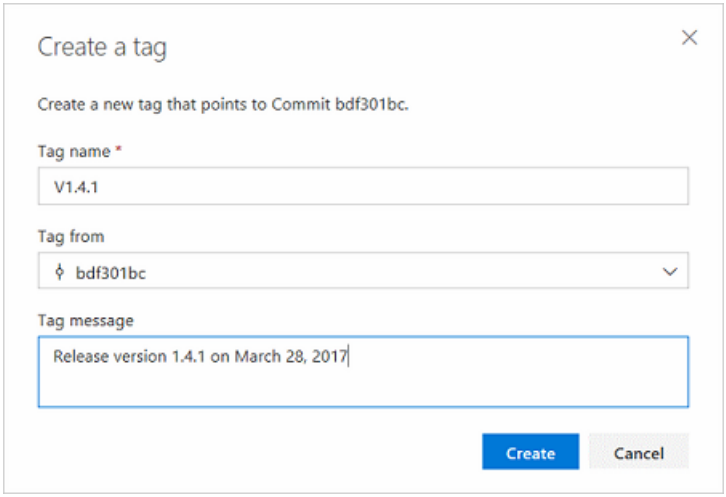
Instead of creating tags from the command line and pushing the tags to the repository, you can now simply go to a commit and add a tag *(Figure 17)*. The tag creation dialog will also let you tag any other ref on the repo.

(Figure 17) Create tag details

The commit list view also supports a context menu *(Figure 18)*. No need to go to the **commit details** page to create tags and create new branches *(Figure 19)*.



(Figure 18) Create tag history



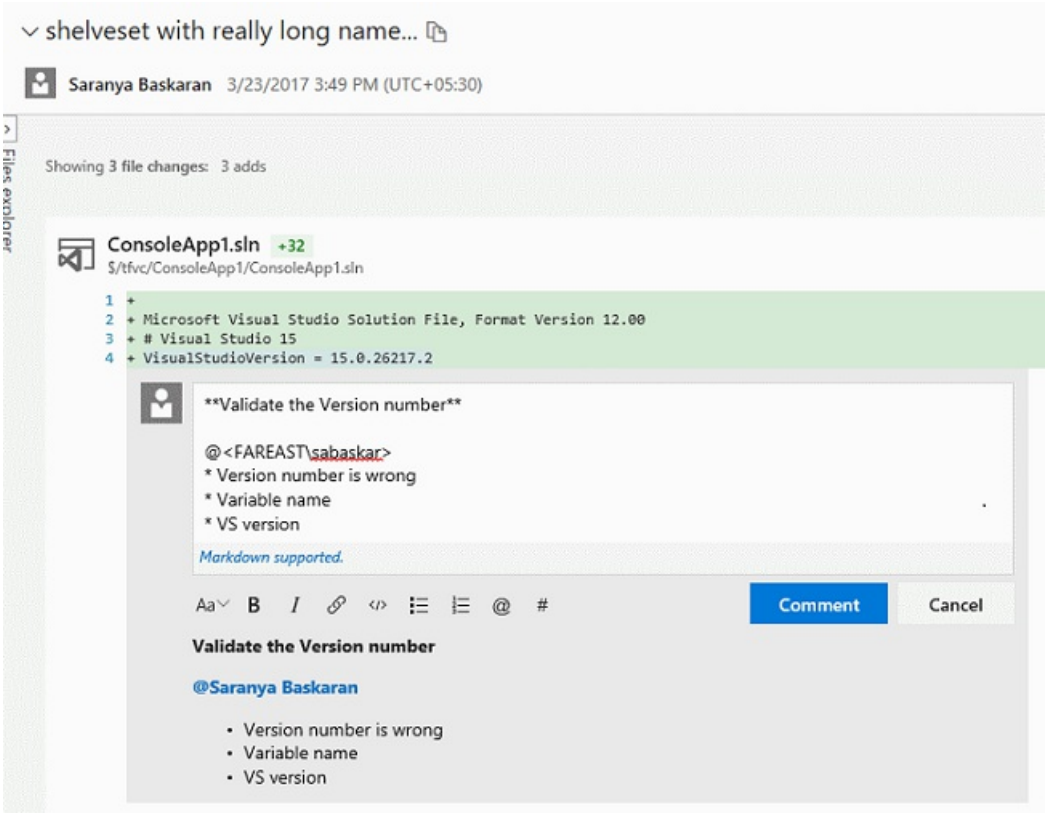
(Figure 19) Tag branch

Updated changeset and shelveset pages

We have modernized the changeset and shelveset pages in TFVC. Both pages are more accessible for those of you who use assistive technologies. The new pages also have a new header that contains the changeset title and associated information about the changeset, such as author details *(Figure 20)*.

(Figure 20) Changeset page

Both changeset and shelveset pages also host the a new markdown discussion control *(Figure 21)* that will allow to type comments in markdown, @mention users, associate work items using #, and easily attach files and images.

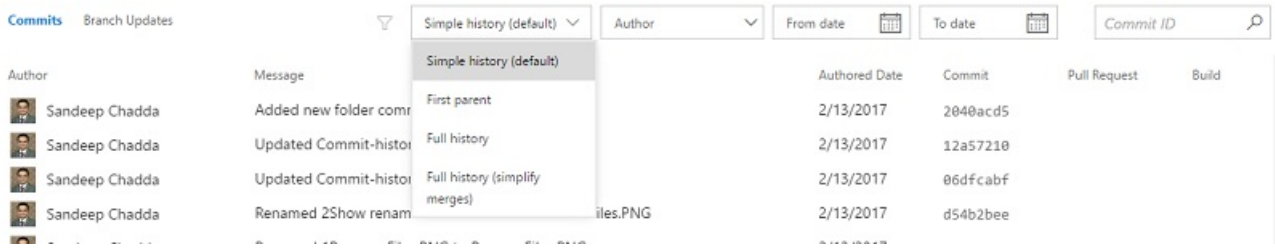


(Figure 21) Changeset discussion

Improved commit filtering

You can now filter the commit history results *(Figure 22)* by advanced filtering options. You can filter commits by:

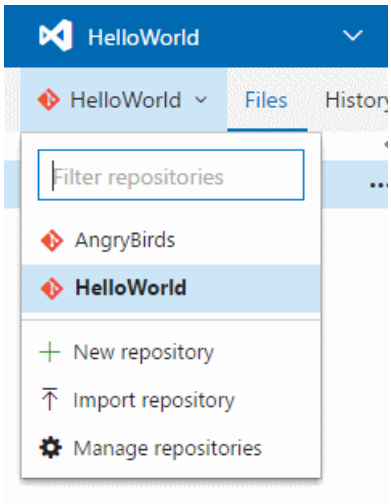
- full history.
- full history with simplified merges.
- first parent.
- simple history (this is the default filter setting).



(Figure 22) Improved commit filtering

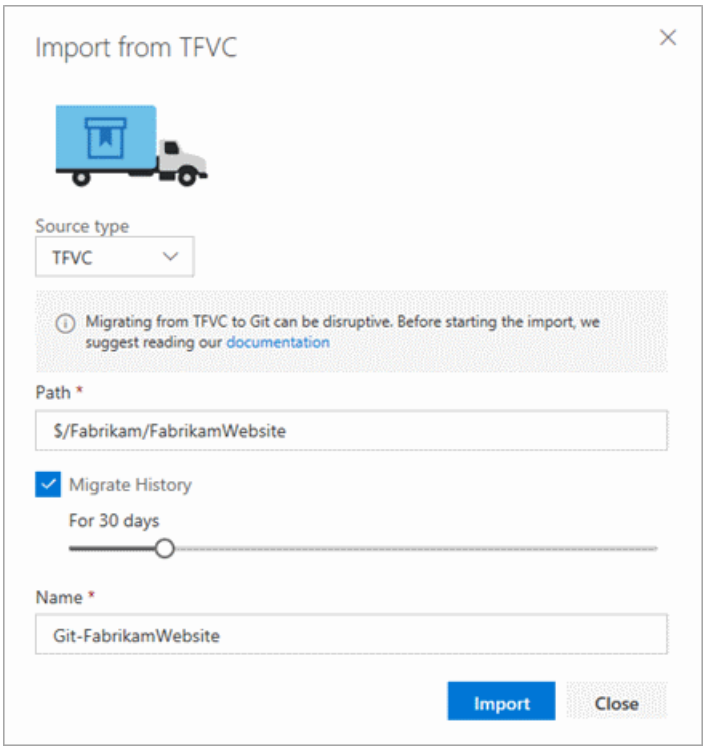
Import repositories from TFVC to Git

You can migrate code from your TFVC repositories to Git repositories in the same account. To start migration, select **import repository** from the repository selector drop-down *(Figure 23)*.



(Figure 23) Repository selector drop-down

Individual folders or branches can be imported to the Git repository, or the entire TFVC repository can be imported (minus the branches) *(Figure 24)*. You can also import up to 180 days of history.



(Figure 24) Import repo complete

Git LFS file locking

We have added the Git LFS file locking feature. This allows teams working with large, undiffable files to avoid losing work when two or more people attempt to edit the same file at once. Before anyone can begin editing the file, they take a lock, and that lock notifies the server. When anyone else attempts to take a lock, the server rejects the request, letting the second person know that someone else is already working on that file. Please upgrade to Git LFS 2.1 or higher to use this feature.

Git commit comments use the new discussion control

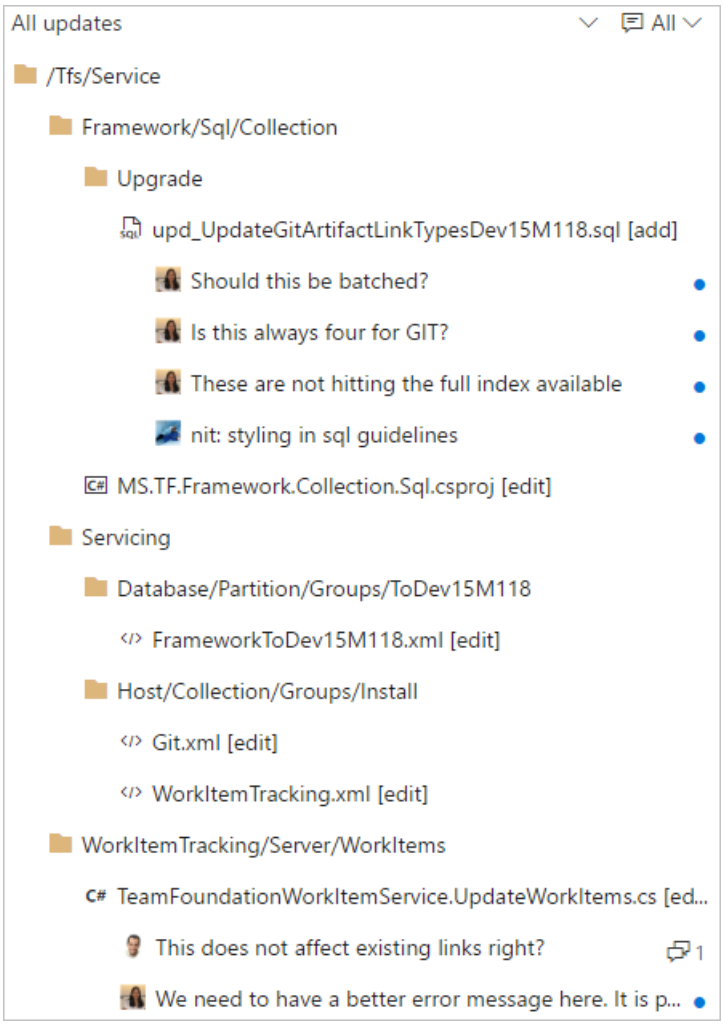
We have updated lightweight comments left on Git commits to use the new discussion control. This brings support for Markdown in those comments, and rounds out all of the code-commenting features in the web for both Git and TFVC to use the latest experience.

New tree view control

The Pull Request Files view, Git commit details, Git push details, TFVC Shelveset details, TFVC Changeset details, TFVC Changesets hub and Git history hub have been updated with a new tree view control *(Figure 25)*. The tree

view has a few usability improvements. First, we have changed the view to show a condensed tree view that automatically collapses empty folder nodes, maximizing the number of files that are in view.

The tree also shows comments in a more compact way. Files with comments show a child item for each comment thread, with the avatar indicating the user that created the thread. New comment threads and those with replies are indicated by the blue dot, and the count of replies is summarized with a count.



(Figure 25) New tree view

Pull Request Improvements

Improved CTAs for PR author and reviewers

For teams using branch policies, it can sometimes be hard to know exactly what action is required when you view a pull request. If the main call to action is the **Complete** button, does that mean it is ready to complete? Using information about the person viewing the page and the state of configured branch policies, the PR view will now present the call to action that makes the most sense for that user.

When policies are configured, but have not passed, the **Complete** button *(Figure 26)* now encourages the use of the **Auto-complete** feature. It is not likely that you can complete the PR successfully if policies are blocking, so we offer an option that completes the PR when those policies eventually pass.

(Figure 26) Auto-complete feature

For reviewers, it is more likely that you will want to approve a PR than complete it, so reviewers will see the **Approve** button *(Figure 27)* highlighted as the main CTA if you have not approved yet.

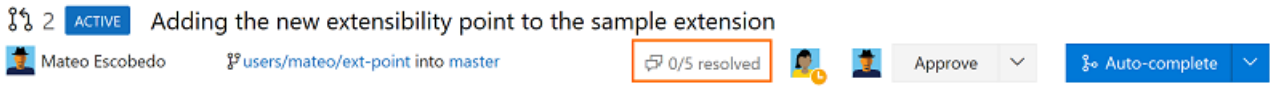
(Figure 27) CTA approve

Once approved, reviewers will see the **Complete** (or **Auto-complete**) button highlighted as the CTA for those cases where a reviewer is also the person completing the PR.

Actionable comments

In a PR with more than a few comments, it can be hard to keep track of all of the conversations. To help in comment management, we have simplified the process of resolving items that have been addressed with a number of enhancements:

- In the header for every PR, you will now see a count of the comments that have been resolved *(Figure 28)*.

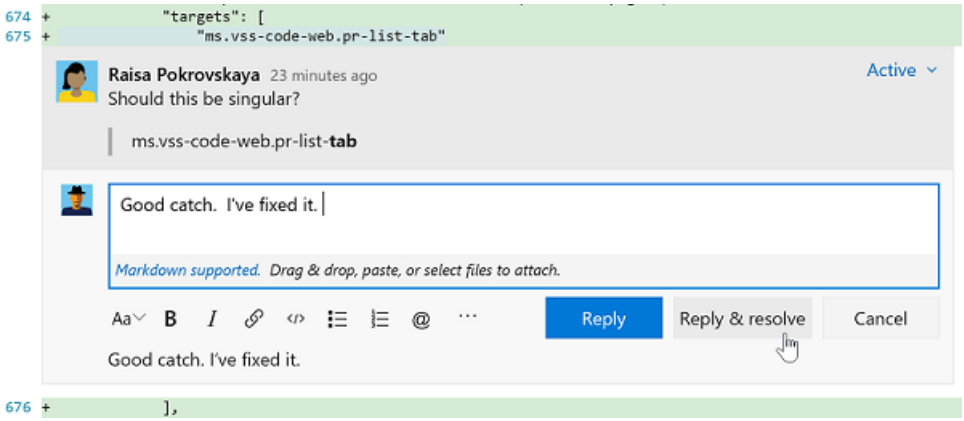


(Figure 28) PR header

- When a comment has been addressed, you can resolve it with a single click *(Figure 29)*.

(Figure 29) Resolve button

- If you have comments to add while you are resolving, you can reply and resolve in a single gesture *(Figure 30)*.

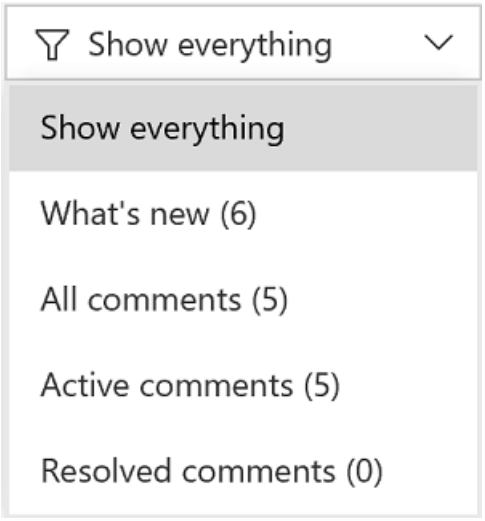


(Figure 30) Reply and resolve

- As comments are resolved, you will see the count go up until everything has been addressed *(Figure 31)*.

(Figure 31) Comment count address rate

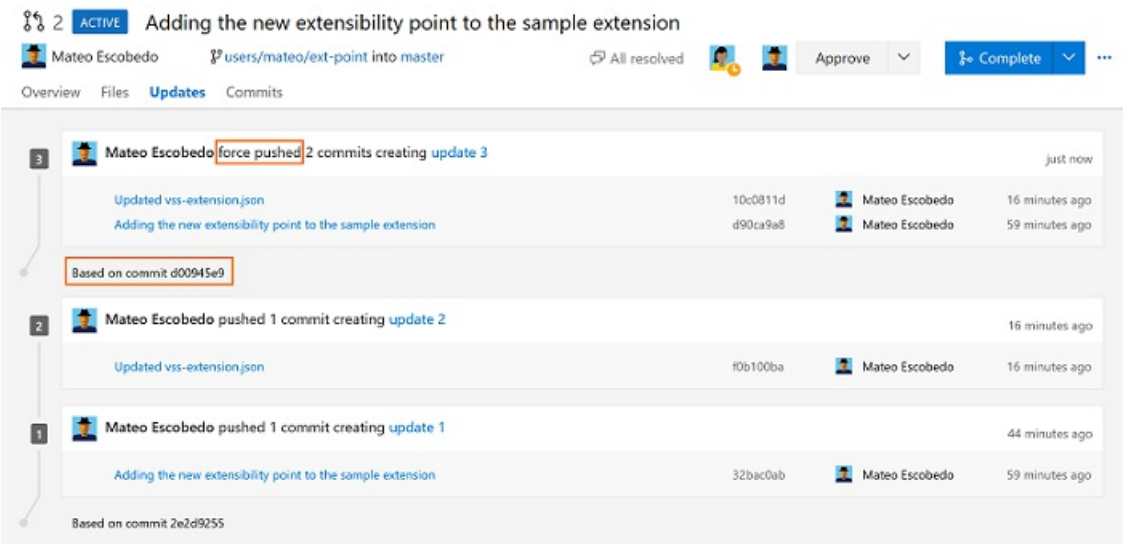
- We have improved the filter in the Overview to enable filtering by various comment states and to show the count of comments for each filter option *(Figure 32)*.



(Figure 32) Filter improvements

Updates view shows rebase and force push

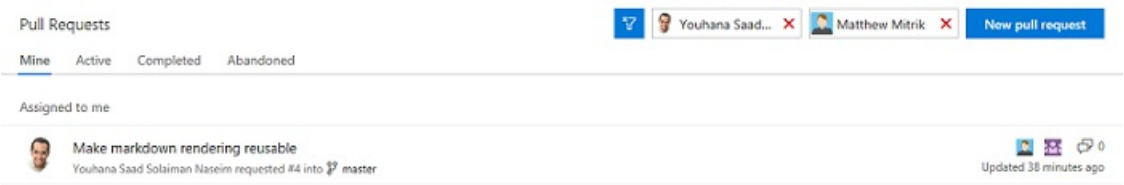
In the **Pull Request details** view, the **Updates** tab has been improved to show when a force push has occurred and if the base commit has changed *(Figure 33)*. These two features are extremely useful if you rebase changes in your topic branches before completing your PRs. Reviewers will now have enough info to know exactly what is happened.



(Figure 33) Updates views

Pull request filtering by people

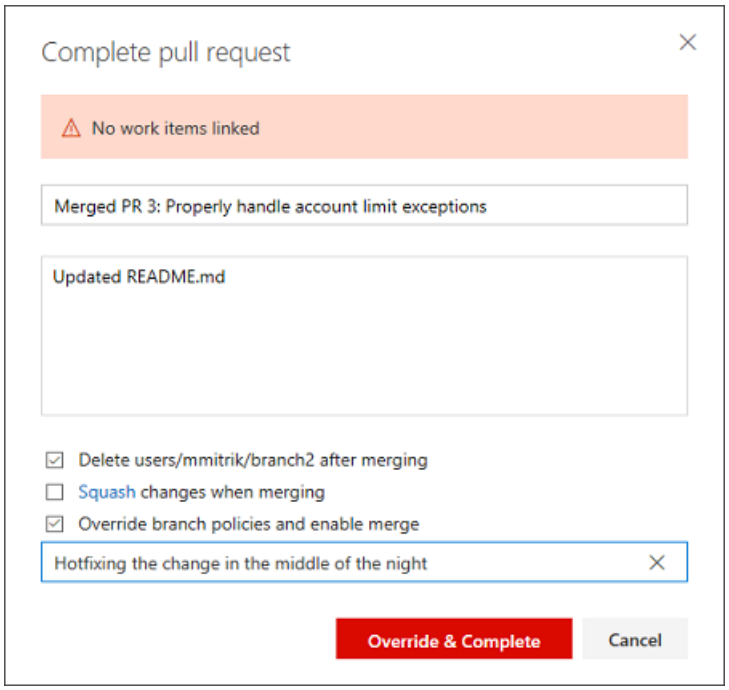
It is now easier to find pull requests! We have added new filtering options to allow you to find PRs created by a specific author or assigned to a specific reviewer *(Figure 34)*. Simply select a user from the author or reviewer filter, and the list will update to show only the PRs that match the filter.



(Figure 34) Filtering by people

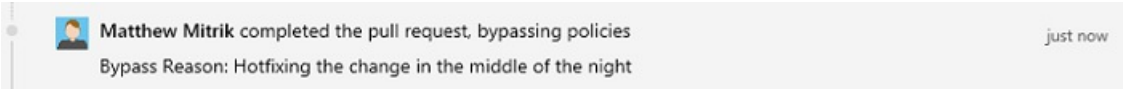
Reason required when bypassing pull request policies

When you are bypassing a pull request policies, you are required to specify a reason. In the **Complete pull request** dialog, you will see a new **Reason** field, if they choose to bypass *(Figure 35)*.



(Figure 35) Bypass dialog

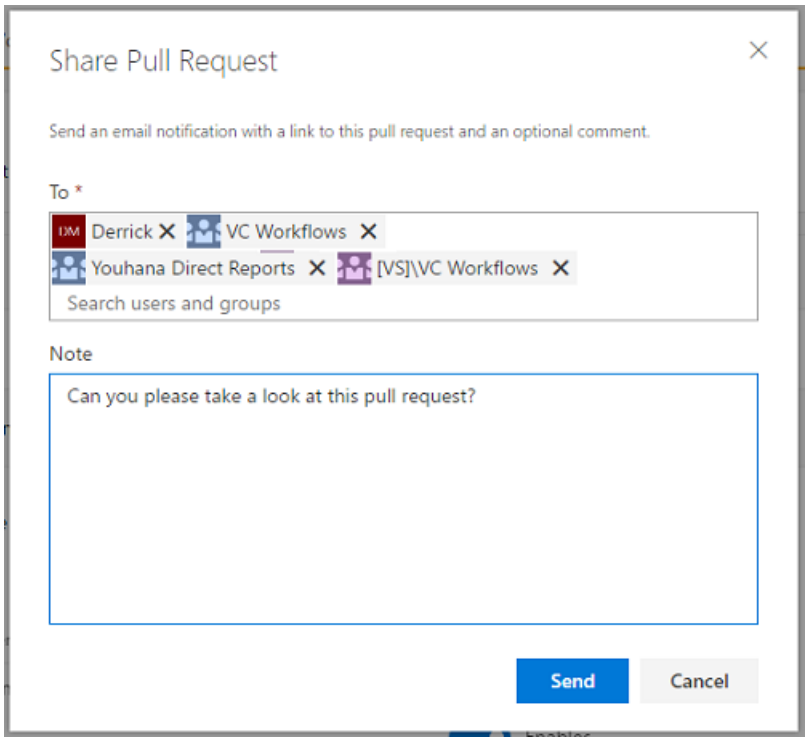
After entering the reason and completing the pull request, the message displays in the **Overview** *(Figure 36)*.



(Figure 36) Bypass message

Share pull requests with teams

The **Share Pull Request** action is a handy way to notify reviewers *(Figure 37)*. In this release, we have added support for teams and groups, so you can notify everyone involved the pull request in a single step.

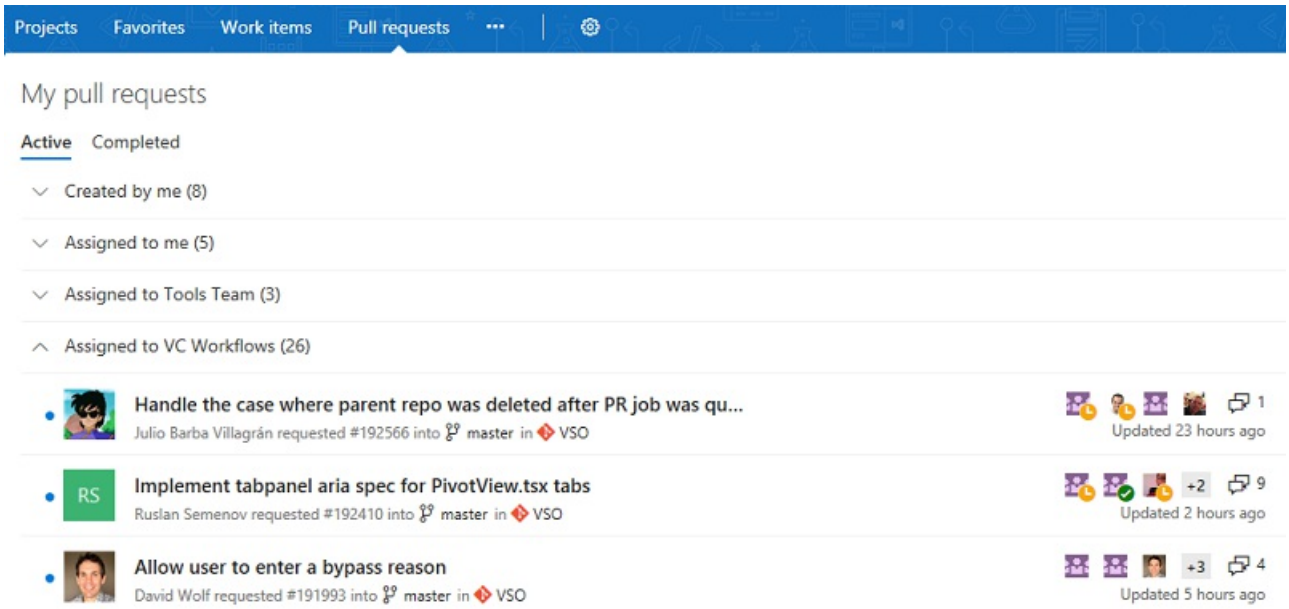


(Figure 37) Share PR with teams

Pull request improvements for teams

If you are a member of multiple teams, you will now see all of the PRs assigned to those teams listed in the **My**

Pull Requests view *(Figure 38)*. This makes the **My Pull Requests** view the one stop you need to visit to see all the PRs on your plate.

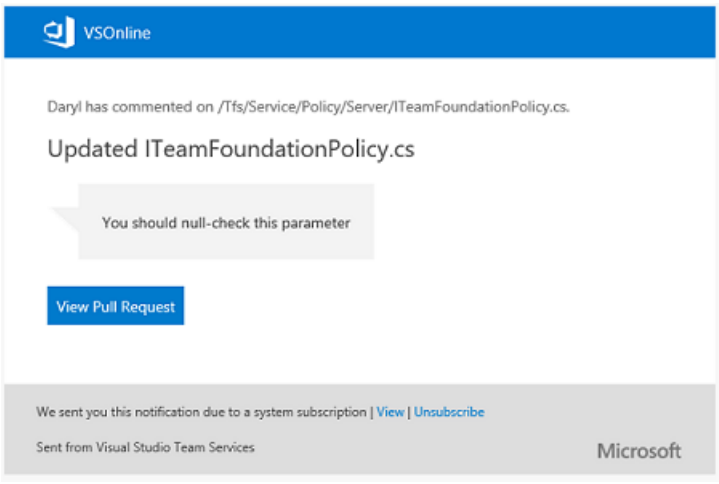


(Figure 38) PR improvements for teams

In a future release, we will add teams to the **Pull Requests** hub under **Code** to make it easier to see all of your PRs for a single project.

Default notifications for pull request comments

Stay up to date with the conversations happening in your PRs with the new comment notifications *(Figure 39)*. For PRs that you've created, you will automatically be notified any time a user adds a new comment thread or replies to an existing thread. When you comment on another user's PR, you'll be notified about any future replies to comment threads that you create or reply to.



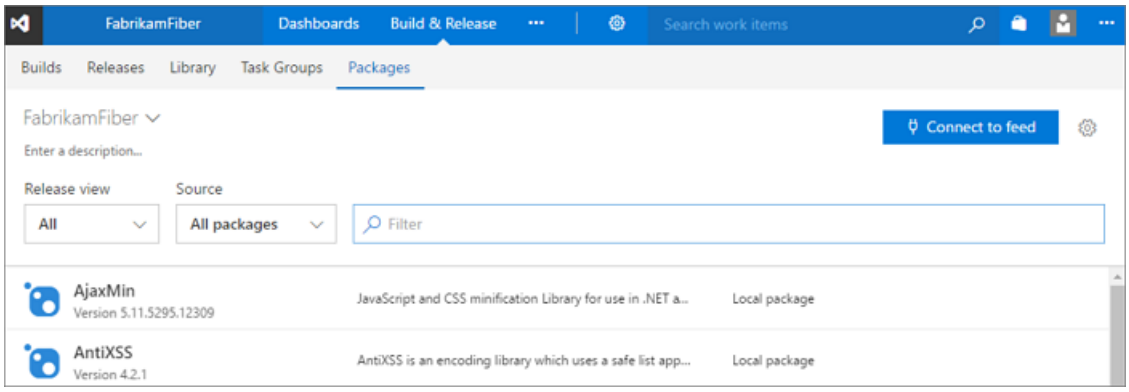
(Figure 39) Default PR notifications

These notifications are available as part of the out of the box subscriptions, and are configurable on the **Notifications** settings page.

Package Management Improvements

Updated Package Management experience

We have updated the Package Management user experience to make it faster, address common user-reported issues, and make room for upcoming package lifecycle features *(Figure 40)*. Learn more about the update on the Updated experience page.

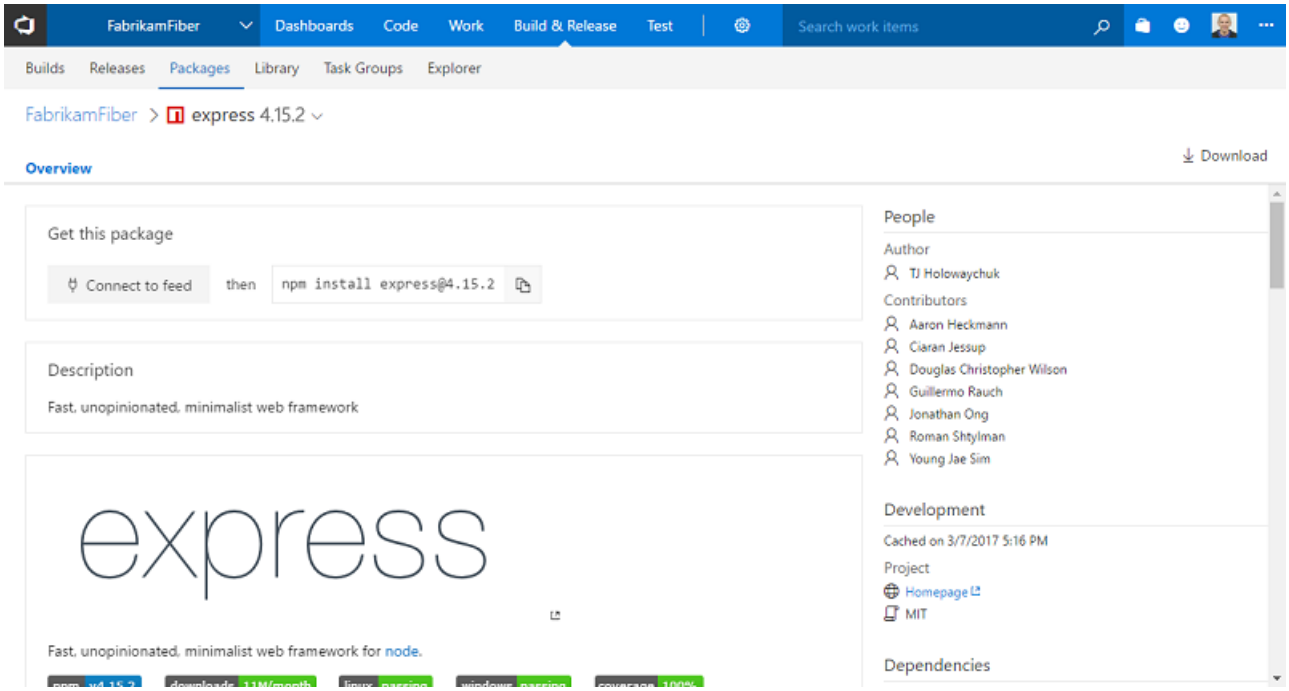


(Figure 40) Package Management

Package Management adds npm READMEs and download button

You can now see the README of any npm package that includes a README.md in the package *(Figure 41)*. READMEs can help your team document and share knowledge about your packages.

You can also download any npm package using the **Download** button in the command bar.



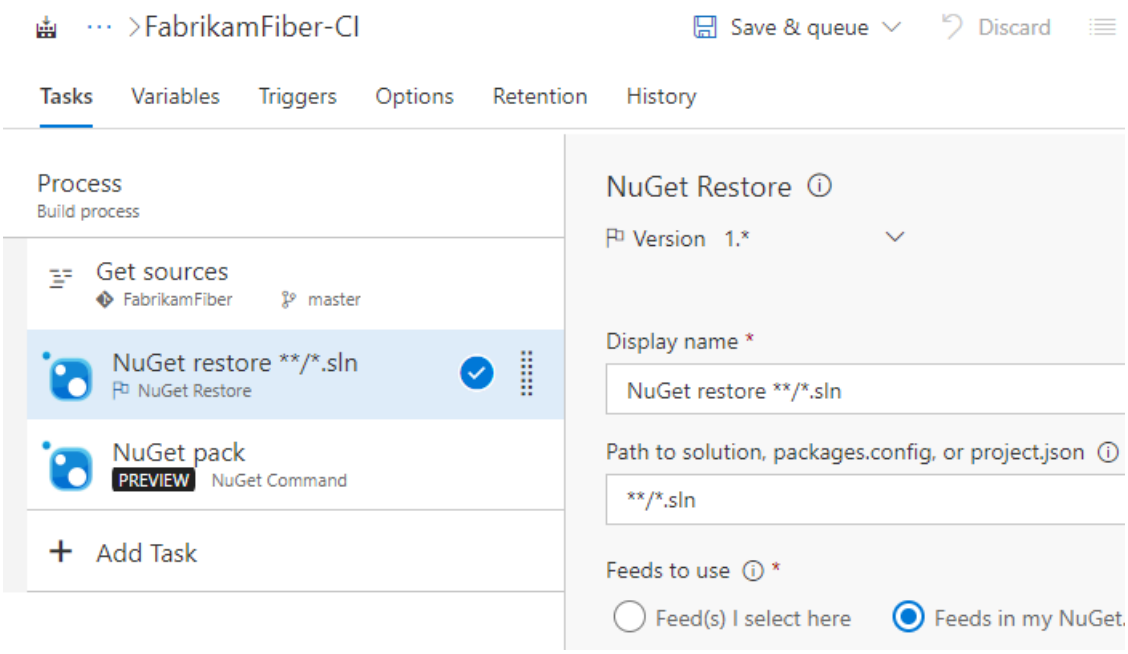
(Figure 41) Package Management npm README

NuGet Restore and NuGet Command build tasks

We have made major updates to the **NuGet Installer** (now called **NuGet Restore**) task, and added a new NuGet task: **NuGet Command**. Most notably, the **NuGet Command** and **NuGet Restore** tasks now use nuget.exe 4.0.0 by default.

NuGet Restore is now optimized for the most common scenario of restoring packages before a Visual Studio Build step. It also has better support for small projects that share a single NuGet feed: you can now pick a Team Services feed and have it added to an auto-generated NuGet.Config.

For more complex NuGet operations, the **NuGet Command** task provides the flexibility to specify any command and set of arguments *(Figure 42)*.



(Figure 42) NuGet command

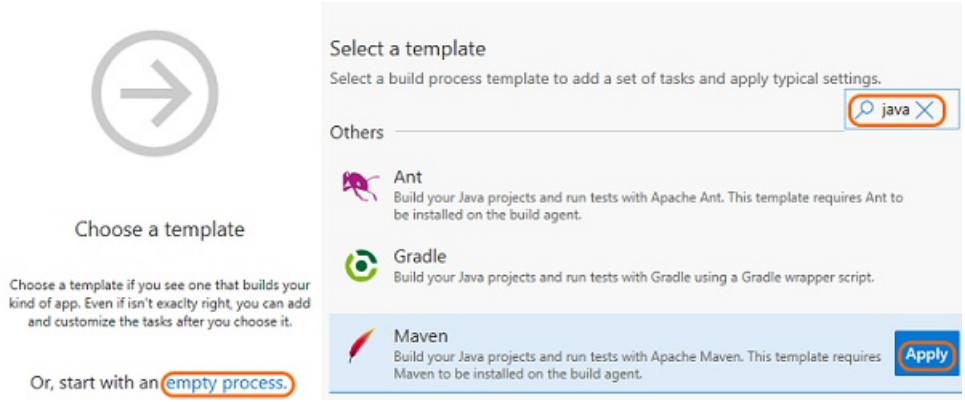
Build and Release Improvements

New build definition editor

We have redesigned our build definition editor to provide a more intuitive experience, fix some pain points, and add new capabilities. We hope that you will find it easier to use templates, add tasks, and change settings. Additionally, you can now use process parameters to make it easier to specify the most important bits of data without having to go deep into your tasks.

Search for a templates

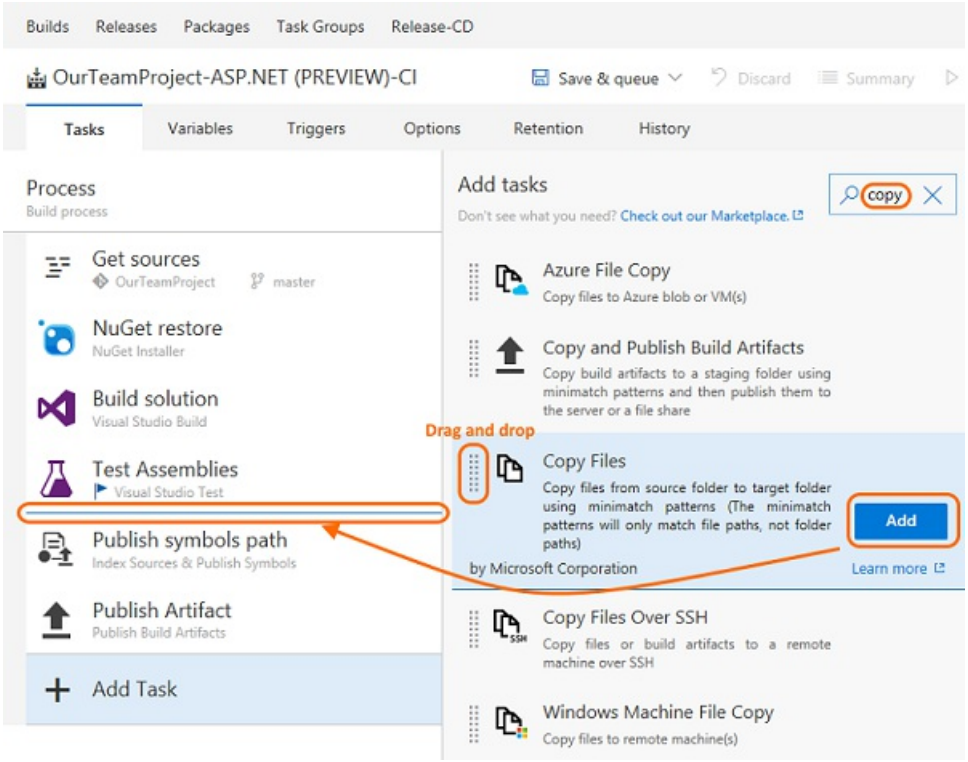
Search for the template you want and then apply it, or start with an empty process *(Figure 43)*.



(Figure 43) Build template search

Quickly find and add a task right where you want it

Search for the task you want to use, and then after you have found it, you can add it after the currently selected task on the left side, or drag and drop it where you want it to go *(Figure 44)*.

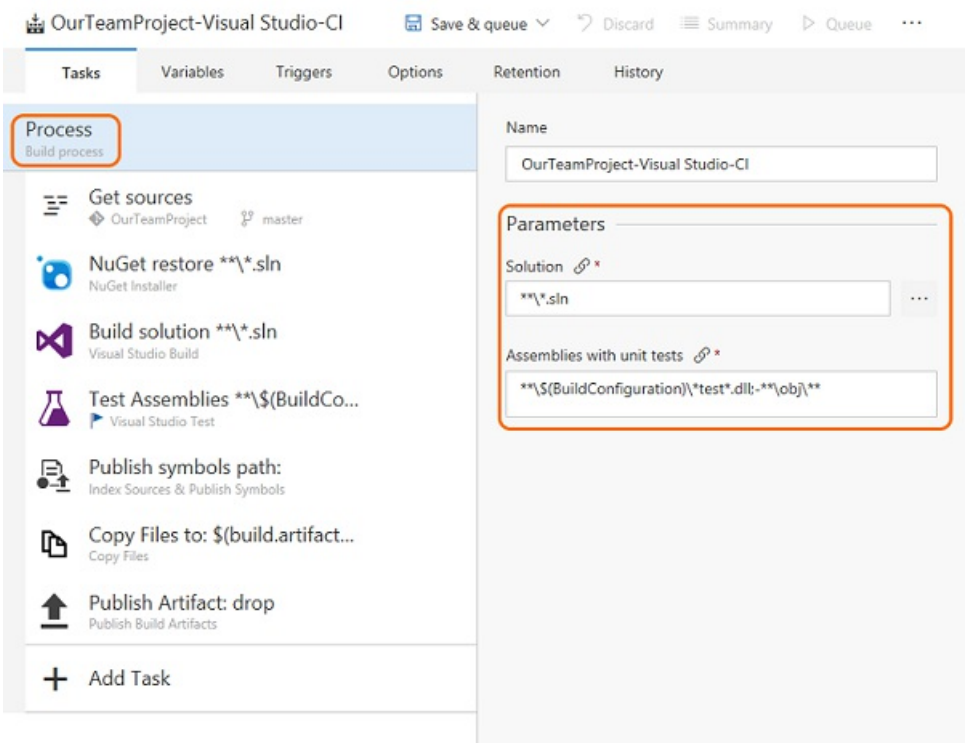


(Figure 44) Build task search

You can also drag and drop a task to move it, or drag and drop while holding the Ctrl key to copy the task.

Use process parameters to pass key arguments to your tasks

You can now use process parameters *(Figure 45* to make it easier for those who use your build definition or template to specify the most important bits of data without having to go deep into your tasks.



(Figure 45) Process parameters

If you create a new build from some of the built-in templates (for example **Visual Studio** and **Maven**) you can see examples of how these work. The new editor includes a few other enhancements, such as giving you quicker access to your sources settings.

For a walkthrough of creating your first build definition using the new editor, see CI/CD for newbies.

Learn more on the 2017 user experience page.

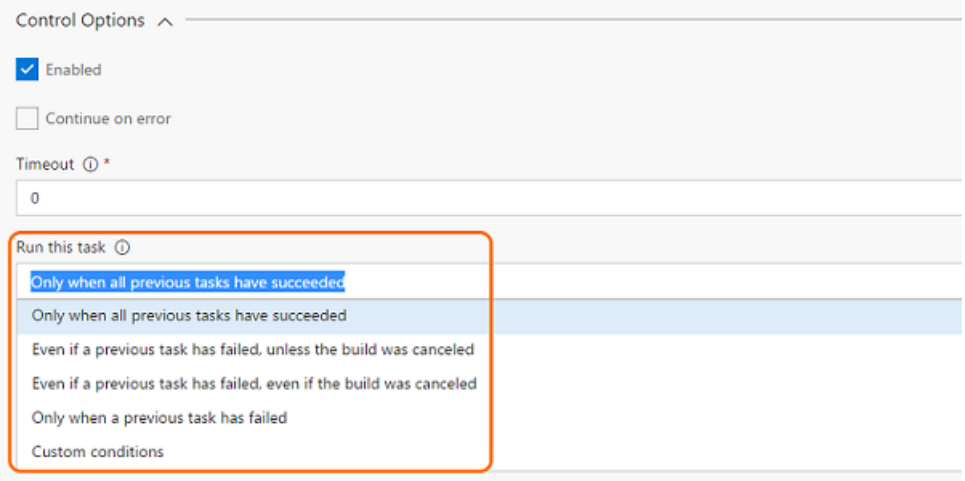
Multiple versions of Extension tasks

Extension authors can now create extensions with multiple versions of a given task, enabling them to ship patches for each major version they have in production.

See Reference for creating custom build tasks within extensions.

Conditional build tasks

If you are looking for more control over your build tasks, such as a task to clean things up or send a message when something goes wrong, we are now offering four built-in choices for you to control a running task *(Figure 46)*.



(Figure 46) Conditional build tasks

If you are looking for more flexibility, such as a task to run on specific branches, with certain triggers, under certain conditions, you can express your own **custom conditions**:

```
and(failed(), eq(variables['Build.Reason'], 'PullRequest'))
```

See Specify conditions for running a task page.

Built-in tasks for building and deploying container based applications

With this release we have pulled most of the tasks in our Docker extension into the product by default, improved them, and introduced a set of new tasks and templates for making a set of container scenarios easier.

- **Docker:** Build, push, or run Docker images, or run a Docker command. This task can be used with Docker or Azure Container registry. You can now use our built-in service principal authentication with ACR to make it even easier to use.
- **Docker-Compose:** Build, push, or run multi-container Docker applications. This task can be used with Docker or Azure Container registry.
- **Kubernetes:** Deploy, configure, or update your Kubernetes cluster in Azure Container Service by running kubectl commands.
- **Service Fabric:** Deploy containers to a Service Fabric Cluster. Service Fabric is the best choice today for running Windows Containers in the cloud.

Azure Web App deployment updates

We have made many enhancements for Azure Web Applications:

- Azure App Service deployment task supports deployment of Java WAR files, Node.js, Python, and PHP applications.
- Azure App Service deployment task supports deploying to Azure Web App for Linux using containers.
- Azure portal Continuous Delivery is expanded to support Node applications.

- Azure App Service manage task is added to Start, Stop, Restart or Slot swap for an Azure App Service. It also supports installing site extensions to enable installation of the required PHP or Python version or installing IIS Manager or Application Insights.

We have also introduced CI/CD support into the latest version of the Azure CLI for configuring CI/CD. Here is an example:

```
az appservice web source-control config --name mywebapp --resource-group mywebapp_rg --repo-url
https://myaccount.visualstudio.com/myproject/_git/myrepo --cd-provider vsts --cd-app-type AspNetCore
```

.NET Core tasks support project files

With the current update, we are enhancing .NET core tasks to support *.csproj files in addition to project.json. You can now use Visual Studio 2017 on your build agents to build .NET core applications using csproj files.

SSH deployment improvements

The **Copy Files Over SSH** build/release task now supports tildes(~) in the destination path to simplify copying files to a remote user's home directory. Also, a new option causes the build/release to fail when no files are found to copy.

The SSH build/release task now supports running scripts with Windows line endings on remote Linux or macOS machines.

Install an SSH key during a build or release

A new preview task, **Install SSH Key (Preview)**, installs an SSH key prior to a build or release and removes it from the agent when the build or release completes. The installed key can be used for fetching code from a Git repository or submodules, running deployment scripts, or other activities that require SSH authentication. This feature will improved in the future to support passphrases and other capabilities.

Tasks fail if Visual Studio 2017 is specified but not present on agent

The Visual Studio Build and MSBuild tasks enable you to select a specific version of Visual Studio. Until now, if the **Visual Studio 2017** version was not available, these tasks would automatically pick the next available version.

We are changing this behavior. Now the build will fail if you select **Visual Studio 2017** but it is not present on the agent.

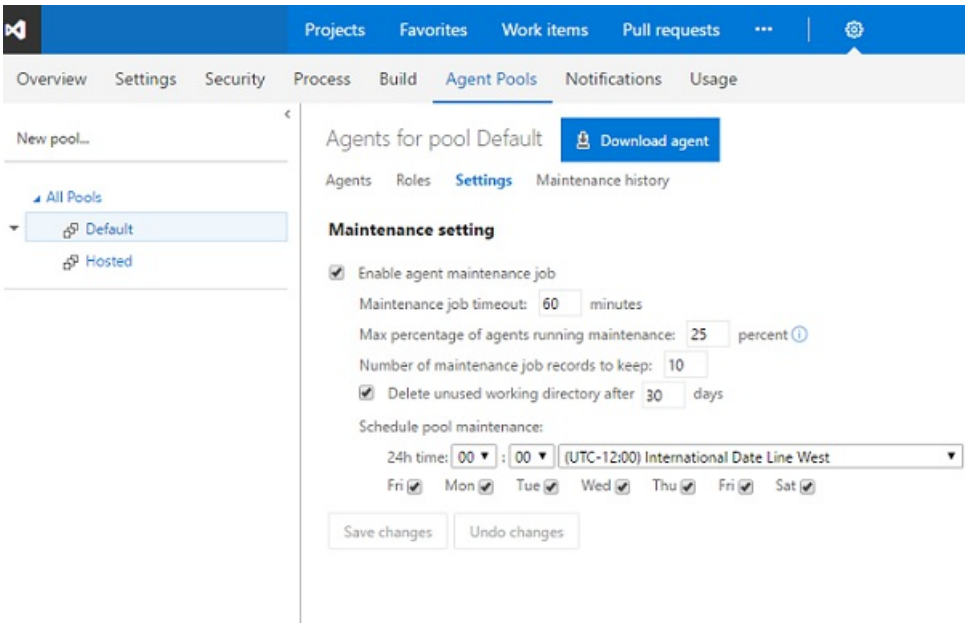
We made this change for the following reasons:

- Newer app types such as .NET Core do not compile with older build tools. They explicitly require Visual Studio 2017 or newer.
- You get more consistent and predictable results when you use the same exact version of Visual Studio.
- Whenever build tasks fall back, you may get compilation errors that are difficult to understand.

> **TIP**
> Make sure to use a queue connected with a pool that has agents with Visual Studio 2017, and no agents that have only earlier versions of Visual Studio.

Private agent automatic workspace cleanup

You can now configure an agent pool to periodically clean up stale working directories and repositories *(Figure 47)*. For example, the pool will delete workspaces left behind by deleted build and release definitions.



(Figure 47) Agent maintenance

Using this option should reduce the potential for your private build and release agents to run out of disk space. The maintenance is done per agent (not per machine), so if you have multiple agents on a single machine you could still run into disk space issues.

Build agent upgrade status

When an agent is upgraded, it now indicates the status of the upgrade in the queue and pool management portal.

Selection of private agents on machines not in use

The system now uses machine name as a factor when allocating a build or a release to a private agent. As a result, the system will prefer an agent on an idle machine over an agent on a busy machine when it allocates the job.

Pipelines queue

We have now moved from the agent-based pricing model to pipeline-based pricing model. In this new model, users can run as many builds or releases concurrently as the number of pipelines configured in their account. Additional builds and releases beyond this limit are queued and wait for earlier builds and releases to complete. The **Pipelines queue** feature provides users with more visibility into where their builds or releases are.

On launching the **Pipelines queue**, you can see the following information:

1. Builds and releases waiting for a pipeline to execute and their position in the waiting queue. 2. Builds and releases currently running using available pipelines.

While your build/release is waiting for a pipeline, you can also directly launch this view from inside the build/release logs page and find its current position in the pipeline queue and other details.

Release action in Build summary

We are now supporting a **Release** action, available in the **Build** summary action bar, so it is easy for you to create a release for a build.

Security for variable groups

Security for variable groups is now governed through a set of roles such as **Creator** and **Administrator**.

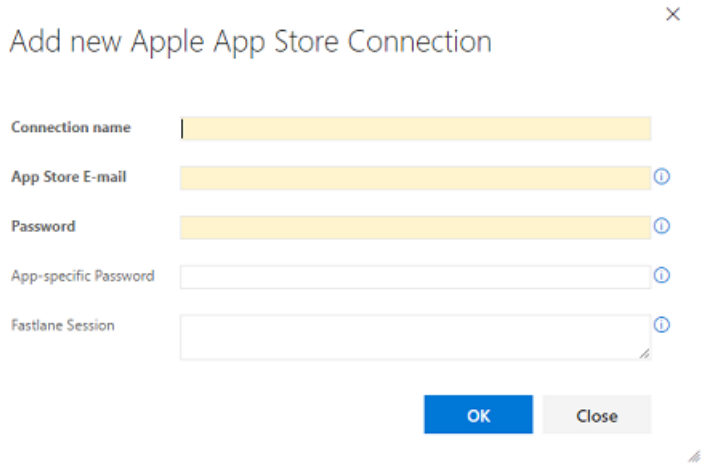
By default, the below roles are assigned.

- Creator role to Contributors
- Administrator role to Project Collection Administrators, Project Administrators, Build Administrators, and Release Administrators
- Reader role to Valid Project Users

The defaults can be overridden for all variable groups or for a specific one.

iOS DevOps enhancements

The Apple App Store extension now supports two-step verification (two-factor authentication) and releasing builds to external testers *(Figure 48)*.



(Figure 48) Apple App Store connection

Install Apple Certificate (Preview) is a new build task that installs a P12 signing certificate on the agent for use by a subsequent Xcode or Xamarin.iOS build.

Install Apple Profile (Preview) is a new build task for installing provisioning profiles on the agent for use by a subsequent Xcode or Xamarin.iOS build.

MSBuild, Xamarin.Android, and Xamarin.iOS build tasks now support building with the Visual Studio for Mac tool set.

Java code coverage enhancements

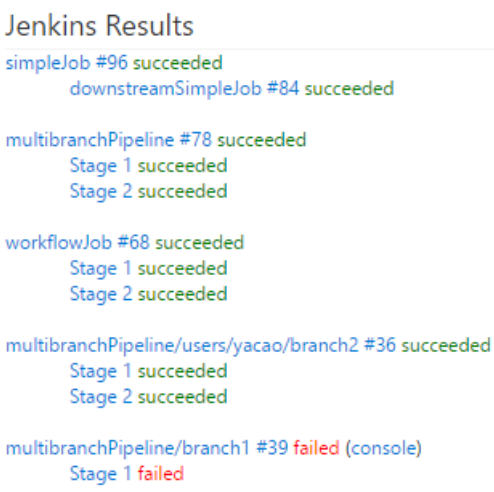
The **Publish Code Coverage Results** build task reports Cobertura or JaCoCo code coverage as part of a build. It now supports specifying wildcards and minimatch patterns in **Summary File** and **Report Directory** fields, allowing the files and directories to be resolved on a per-build basis for paths that change between builds.

Maven and SonarQube improvements

The Maven build task now allows specifying a SonarQube project for analysis results in cases where it differs from what is specified in the Maven pom.xml file.

Improved Jenkins integration

The **Jenkins Queue Job** build/release task now supports running Jenkins multibranch pipeline jobs while displaying the Jenkins console output in Team Services *(Figure 49)*. Pipeline results are published to the Team Services build summary.



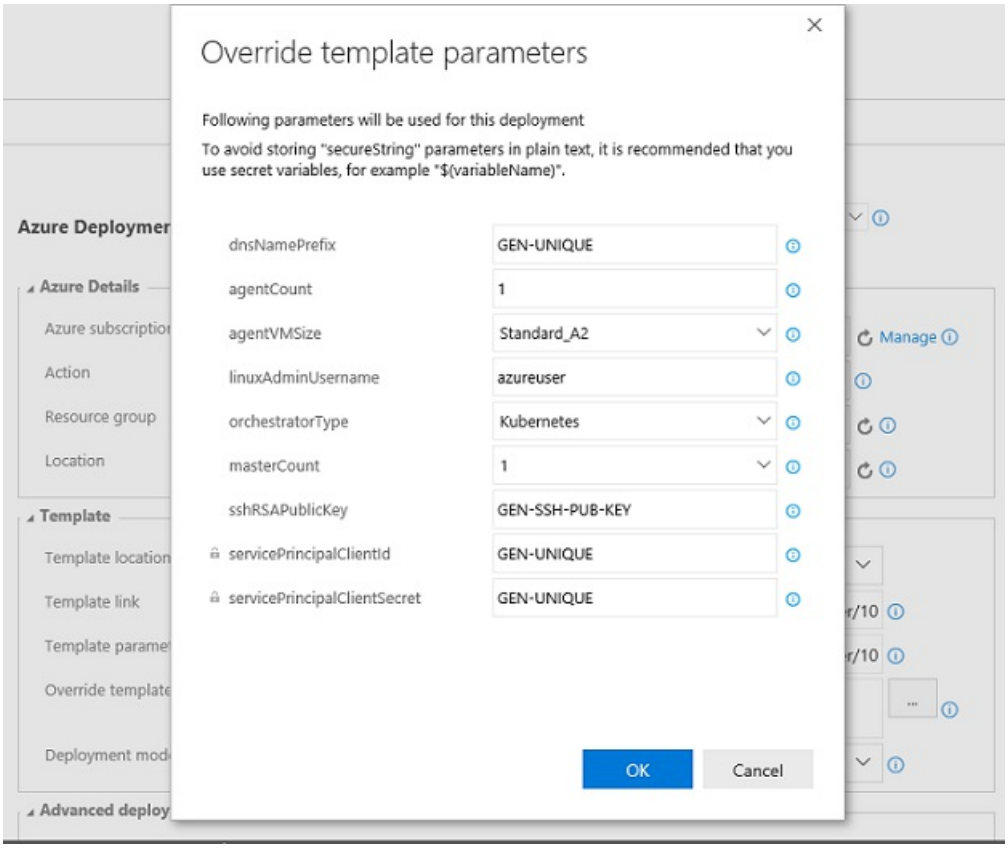
(Figure 49) Improved Jenkins integration

Azure virtual machine scale set deployment

A common pattern used for deployment is to create a full machine image for each version of the application and then deploy that. To make that easier we have a new **Build immutable machine image** task. This task uses Packer to generate a machine image after deploying applications and all the required prerequisites. The task takes either the deployment script or the packer configuration template to create the machine image and stores it in an Azure Storage account. This image can then be used for Azure virtual machine scale set deployments that work well with this type of immutable image deployment.

Override template parameters in Azure resource group deployments

Currently in Azure resource group deployment tasks, users select the template.json and the parameters.json and provide the override parameter values in a text box, following a specific syntax. This experience is now enhanced so the template parameters are rendered in a grid that allows them to be edited and overridden *(Figure 50)*. You can access this feature by clicking the **...** next to the override parameters field, which opens a dialog with the template parameters along with their default values and allowed values (if defined in the template and parameter .json files). This feature requires that you enable CORS rules at the source. If template and parameter .json files are in Azure storage blob, refer to the Azure Storage Services documentation to enable CORS.



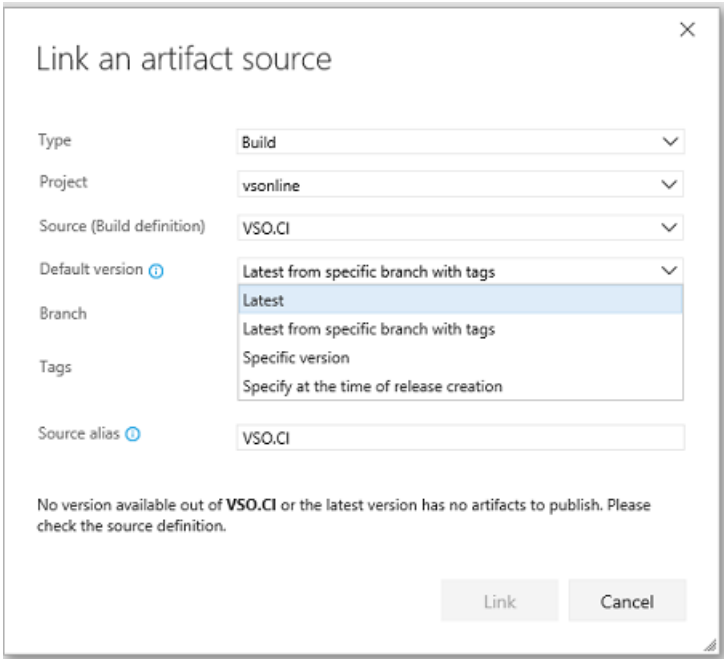
(Figure 50) Azure RG parameters

Multiple release triggers with branch and tag filters

Release management now supports setting up CD triggers on multiple artifact sources of type "Build". When added, a new release is created automatically when a new artifact version is available for any of the specified artifact sources. You can also specify the source branch that the new build should be from to trigger a release. Additionally, Tag filters can be set to further filter the builds that should trigger a release.

Set defaults for artifact sources in a release

Users can define the default artifact version to deploy in a release when linking an artifact source in a definition *(Figure 51)*. When a release is created automatically, the default version for all the artifact sources would be deployed.



(Figure 51) Default artifact version

Separation of duties for deployment requester and approvers

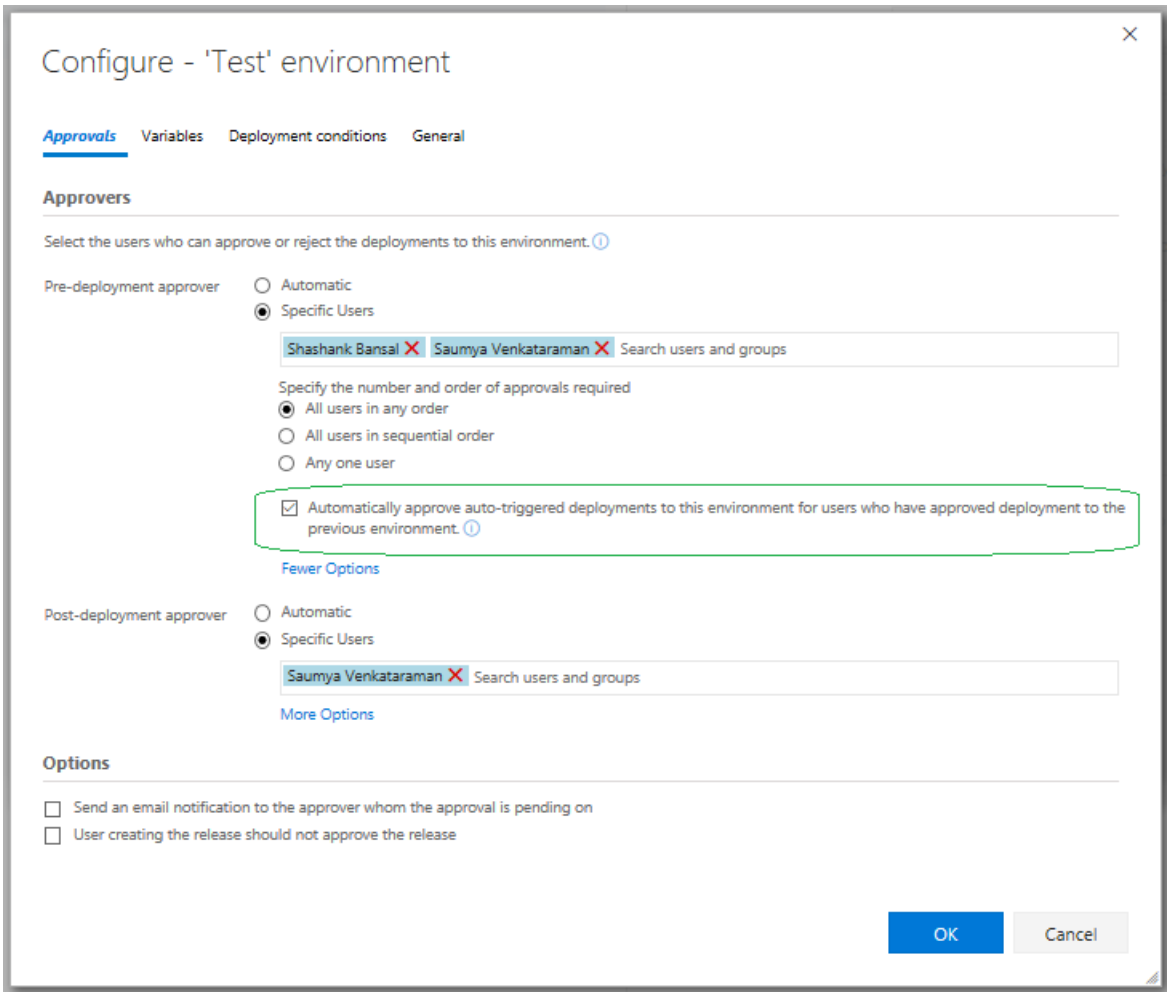
Previously, environment owners could restrict release creators from approving deployments of the release to an environment. You could, however, manually start deployment of a release created by another user, and approve it yourself.

We have now filled this gap by considering the deployment creator as a separate user role for deployments. Either the release creator or deployment creator can be restricted from approving the deployments.

Release level approvals

You can now choose to automatically approve deployments that were automatically triggered after successful deployment to another environment *(Figure 52)*. Approving a chain of deployments (that have the same approvers) can be done at one go if you choose to not approve every deployment.

In the case where you have two environments Dev and Test, with the predeployment approvers set to "userA" and "userB", with both of them required to approve the deployment. If the policy on Test is set as shown below, during deployment time it is sufficient for userA and userB to approve only Dev. Deployment to Test will get auto-approved. If the deployment to Test is triggered manually, approvals are required before deployment to ensure correct approvals.

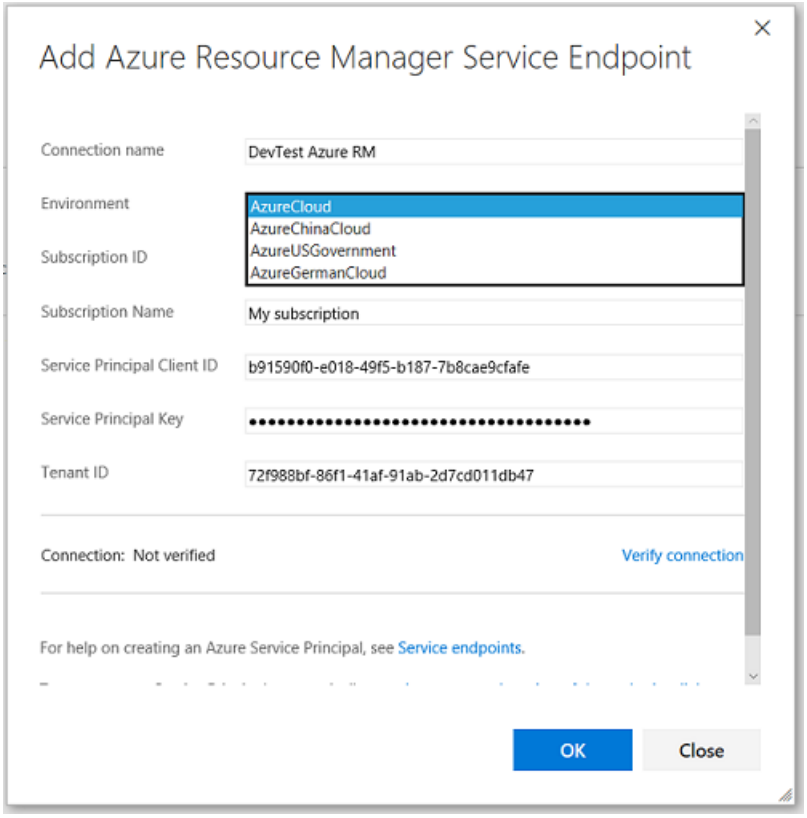


(Figure 52) Release level approvals

Deploy to Azure Government Cloud

Customers with Azure subscriptions in Government Clouds can now configure Azure Resource Manager service endpoint to target national clouds.

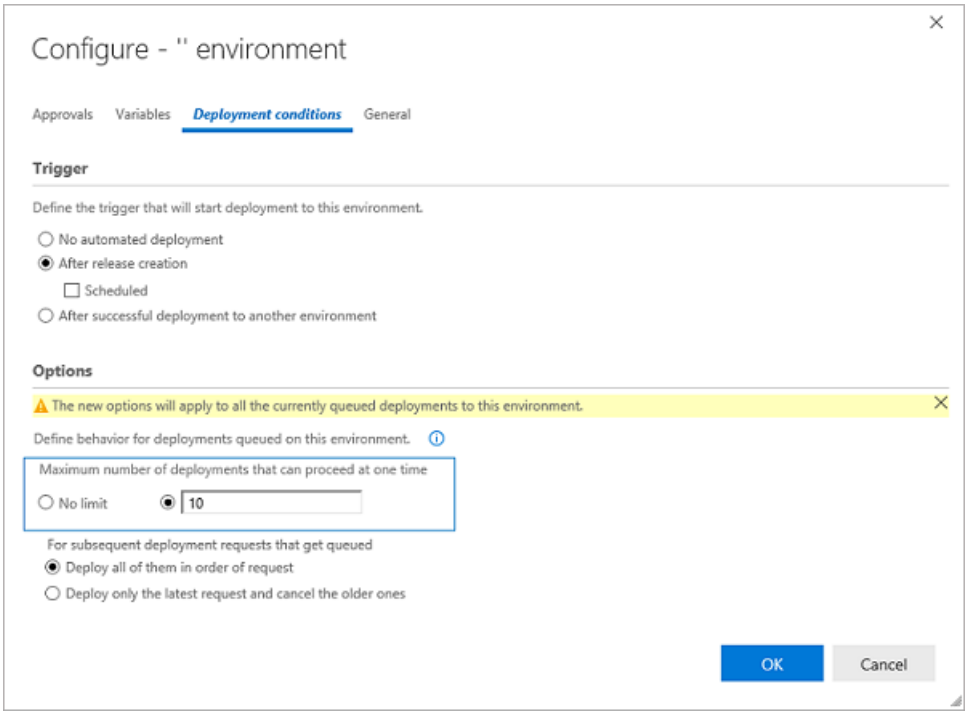
With this, you can now use Release Management to deploy any application to Azure resources hosted in government clouds, using the same deployment tasks *(Figure 53)*.



(Figure 53) Government cloud

Set maximum number of parallel deployments

This feature gives you control on how multiple pending releases are deployed into a given environment *(Figure 54)*. For example, if your release pipeline performs validation of builds in a QA environment and the rate of generation of builds is faster than the rate of completion of the deployments, you may configure multiple agents and as many builds to get validated in parallel. That means each of the builds generated is validated, and the wait time is dependent in the number of available agents. With this feature, we let you optimize validations by enabling you to perform validation on the n most recent builds in parallel and cancel the older deployment requests.



(Figure 54) Parallel deployments

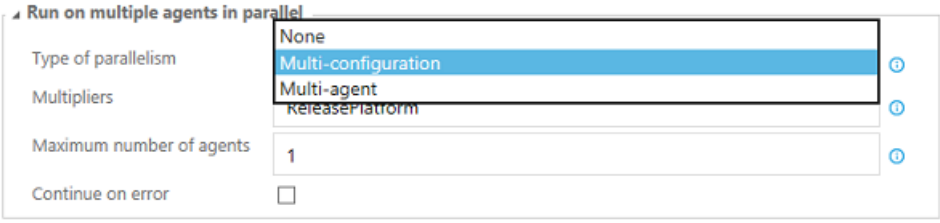
Timeout enhancements for the Manual Intervention task

The **Manual Intervention** task can now be automatically rejected or resumed after pending for the specified

timeout or 60 days, whichever is earlier. You can specify the timeout value in the control options section of the task.

Release Management parallel execution

Release Management now supports a parallel execution option for a phase *(Figure 55)*. Select this option to fan out a phase by using either Multi-configuration or Multi-agent as a phase multiplier option.



(Figure 55) Parallel execution support

Multi-configuration: Select this option to run the phase for each multi-configuration value. For example, if you wanted to deploy to two different geos at the same time, using a variable ReleasePlatform defined on the Variables tab with values "east-US, west-US" would run the phase in parallel, one with a value of "east-US" and the other "west-US". Multi-agent: Select this option to run the phase with one or more tasks on multiple agents in parallel.

Web app deployment history in Azure portal

Release management now updates the deployment logs of Azure App Service when a deployment is completed by using the App Service deployment task. You can view deployment history in the Azure portal by selecting the **Continuous delivery** option in the **App Service** blade.

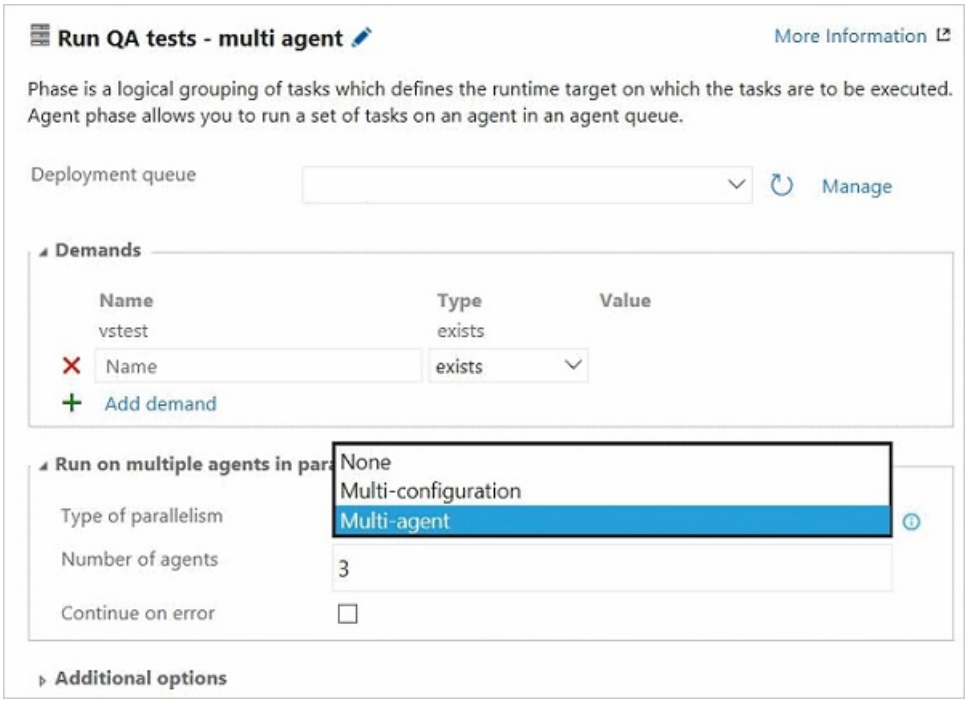
Testing Improvements

Run tests using agent phases

Using the **Visual Studio Test task**, you can now run automated tests using agent phases *(Figure 56)*.

We now have a unified automation agent across build, release and test. This brings in the following benefits:

1. You can leverage an agent pool for your testing needs.
2. Run tests in different modes using the same **Visual Studio Test task**, based on your needs—single agent–based run, multi-agent–based distributed test run or a multi-configuration run to run tests on, say, different browsers.

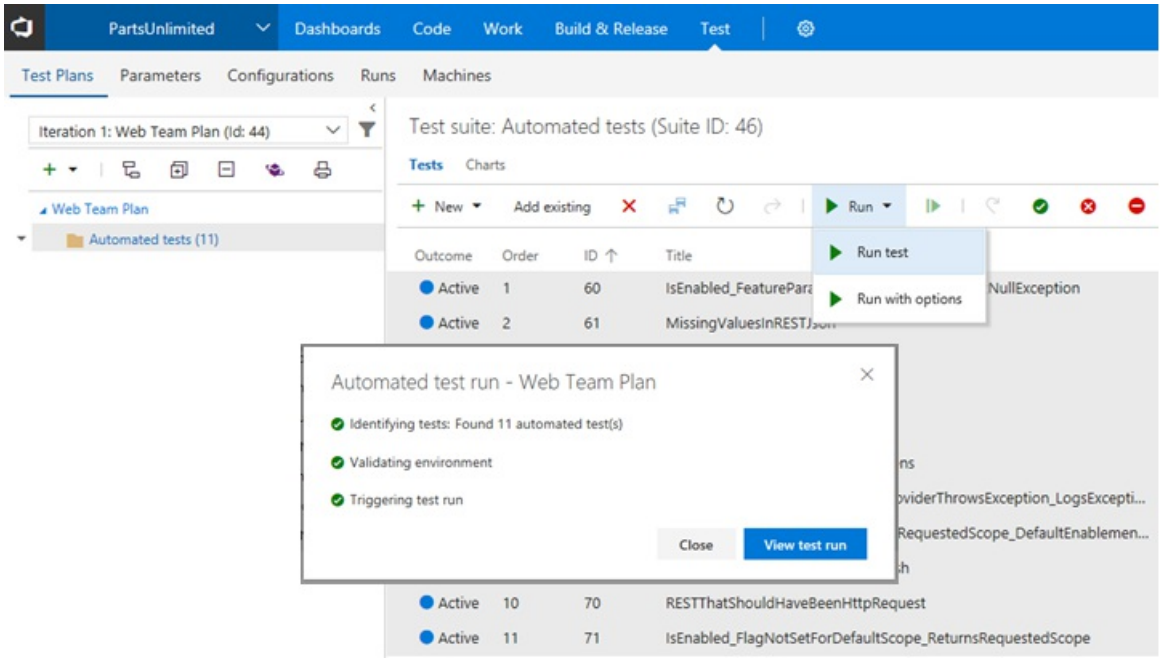


(Figure 56) Run tests using Agent Phases

For more information, refer to this Microsoft Application Lifecycle Management post.

On-demand triggering of automated tests

The **Test** hub now supports triggering automated test cases from test plans and test suites *(Figure 57)*. Running automated tests from the **Test** hub will need a setup similar to the way you run tests in a scheduled fashion in release environments. You will need to setup an environment in the release definition using the **Run automated tests from test plans** template and associate the test plan to run the automated tests. See the documentation for the step-by-step guidance on how to setup environments and run automated tests from the **Test** hub.



(Figure 57) On-demand automated tests trigger

Warehouse Improvements

Performance improvements in Analysis Services cube processing

We have made performance improvements to the **vDimWorkItemTreeOverlay** view, that is used to create **Work Item Tree Hierarchy** dimension based on the links. Although it depends on System.LinkTypes.Hierarchy links, we observed that the processing duration was affected by other links as well (e.g. System.LinkTypes.Related). We optimized the view to skip addition link types that limits the amount of data read. This change significantly decreases processing time for certain warehouses.

Case-insensitive schema reconciliation

The schema of the warehouse database is created by merging fields from all the attached collection databases in the schema reconciliation process. Previously, all comparisons were case-sensitive and administrators had to make sure there is an exact match on field reference names. This led to problems where there were subtle differences in casing. With this release we make the process more tolerant to such discrepancies.

Administration Improvements

Combined email recipients for notifications

Recipients for the same email notification are now included together on the to: line and sent a single email. Previously, individual emails were sent to each recipient. This made it difficult to know who else received the notification and to have a conversation about the event over email. This feature applies to out-of-the-box as well as team subscriptions that are capable of targeting multiple recipients. For example, all reviewers of a pull request are now sent a single email when a change is made to the pull request.

Learn more about combining email recipients.

Out-of-the-box notifications

Users and teams are now automatically notified through email when there is activity in the account directly relevant to them, such as when:

- a work item is assigned to a user.
- a user or team is added as a reviewer to a pull request.
- a user or team is a reviewer on a pull request that is updated.
- another user responds to a pull request comment.
- a build requested by a user completes.
- an extension is installed or requested (administrators only).

Users can unsubscribe from any of these subscriptions by going to **Notification** settings under the user profile menu and then switching off the appropriate toggle(s).

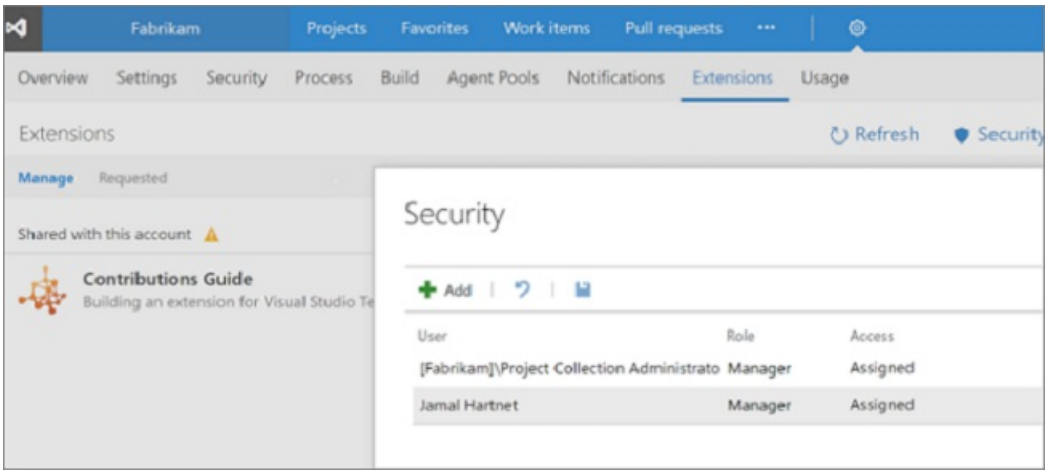
An account administrator can disable one or more of these automatic subscriptions by navigating to the collection-level **Notifications** hub under the settings gear. Any of these subscriptions can be disabled by clicking **Disable** under the "..." action. Once a subscription is disabled, it will no longer appear for users in their personal notification settings page.

Learn more about out-of-the-box notifications.

Extension management permissions

An administrator can now grant other users and groups permission to manage extensions for the collection *(Figure 58)*. Previously only collection administrators (i.e. members of the Project Collection Administrators group) could review extension requests, install, disable, or uninstall extensions.

To grant this permission, an administrator can navigate to the Extensions admin hub by opening the Marketplace menu, selecting Manage extensions, and then click the Security button:



(Figure 58) Extension management permissions

Getting notified when extensions are installed, require attention, and more

Administrators, or those with the ability to manage extensions, are now automatically notified when an extension is installed, uninstalled, enabled, disabled, or requires attention. This is especially useful in larger deployments where multiple people have the responsibility of managing extensions. Admins can turn off these notifications by navigating to **Notification** settings under the profile menu and switching off the extensions toggle.

Administrators can also define custom subscriptions for extension-related events. For example, an administrator can get notified whenever any extension is updated.

Users can also now turn off automatic notifications about their extension requests.

Allowing TFS admins to add subscribers to the advanced access level

The **Advanced** access level will be removed from future versions of Team Foundation Server. However, until that happens, TFS admins have the ability to add MSDN Platform and Visual Studio Test Professional subscribers to the **Advanced** access level with Update 2.

Visual Studio Enterprise subscribers should be added to the Visual Studio Enterprise access level instead of

Advanced. If you have purchased the Test Manager extension, please continue to manage this in the Users hub within the Team Project that you made the purchase.

Microsoft Teams Integration

Organizations using Microsoft Teams to collaborate can now see activity from their TFS projects within their team's channels. This allows teams to stay informed about relevant work item changes, pull requests, builds, and releases and more as they are working in Microsoft Teams. For more information, see our documentation.

Known Issues

Work item forms do not render correctly in the web

- **Issue:**

  If you have a custom control, such as the multi-value control, installed for the Visual Studio client but not the web client, work item forms in the web fail to render.

- **Workaround:**

  You will need to update to the latest version of your control. It is necessary to add a web layout that does not contain the missing control element. You can find the latest multi-value control for TFS 2017 Update on the Custom Controls for TFS Work Item Tracking page. For more information on the layout, see All FORM XML elements reference (TFS 2015) page.

TFS version is RC2 instead of the final release

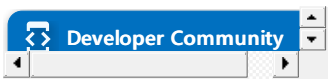
- **Issue:**

  After downloading TFS 2017 Update 2 before August 1, 2017, and installing, you have an RC2 version.

- **Workaround:**

  This was due to an intermittent issue in the installation links fixed on August 1, 2017. Please redownload TFS 2017 Update 2 and install this final release.

See customer-reported issues reported for Team Foundation Server 2017.



Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2017 Update 1 Release Notes

9/7/2018 • 26 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

This is not the latest version of Team Foundation Server. To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2017 Update 1. Click the button to download.

Team Foundation Server 2017

To learn more about Team Foundation Server 2017, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: March 7, 2017

Summary of What's New in Team Foundation Server 2017 Update 1

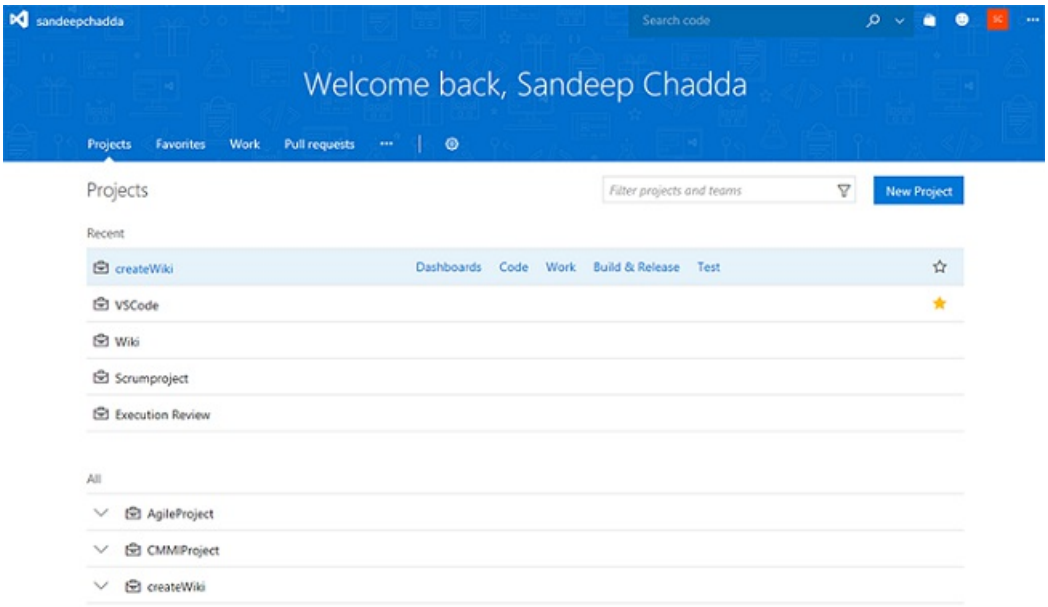
- More Personal Experiences
- Version Control Improvements
- Work Item Tracking Improvements
- Build Improvements
- Payment Required for Package Management
- Package Improvements
- Cross Platform Improvements
- Testing Improvements
- Release Management Improvements
- Code Search Update
- Code Insights Improvements
- Administration Improvements
- Team Room Deprecation
- Markdown no Longer Supports File Links
- Announcing the Process Template Editor

Details of What's New in Team Foundation Server 2017 Update 1

More personal experiences

Personalized collection home page

With this release, it is easy for you to access artifacts that are most important to you. The redesigned collection page *(Figure 1)* has a personalized experience that shows the Projects, Favorites, Work, and Pull Requests you care about. You can go to one place and quickly find everything you need and care about. See Account hub pages for more information.



(Figure 1) Redesigned collection page

Your project gets an identity

There is now one place to get an overview of your project. The new project page makes it easy to view and edit the project description, view or add members, and check on the latest activity. It is even easier to get started with a new project, and leverage all the built-in DevOps functionality of TFS.

Version Control Improvements

Repo admin permission changes

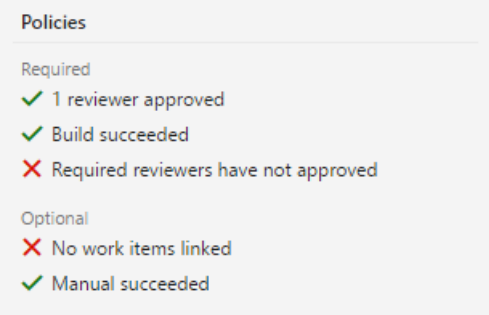
For Git repos, we have divided the **Administer** permission into several more granular permissions. This gives you more flexibility to decide who can perform what actions. For instance, you may allow anyone in your account to create new repositories, but disallow them from deleting repos or adding new users to a repo. The new permissions are:

- Manage permissions: Add/remove users and permissions.
- Create: Create a new repo.
- Delete: Delete a repo.
- Rename: Rename a repo.
- Edit policies: Configure branch policies.
- Remove others' locks: Remove branch locks set by another user.

These permissions can be applied to all repositories in a project, or to individual repositories.

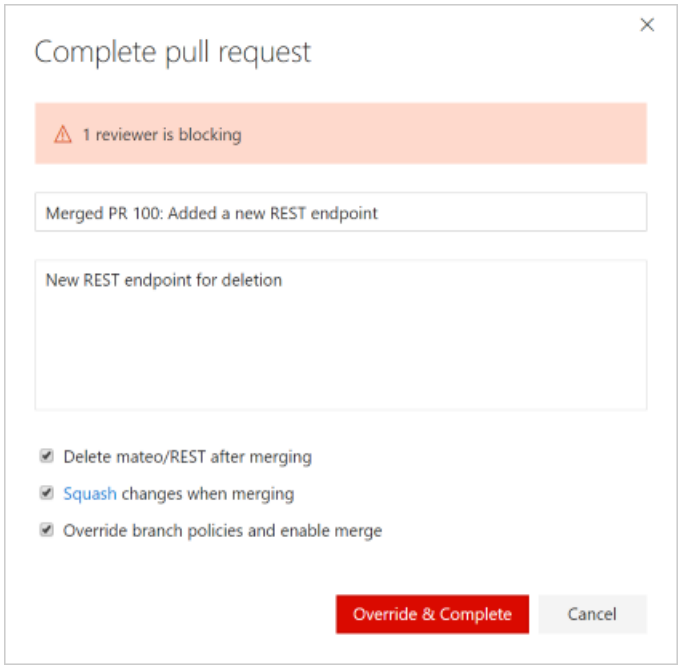
Branch policy improvements

In the **Policies** section *(Figure 2)*, the required and optional policies are now grouped into sections. This clarifies exactly which policies are required in order to complete a PR. Required reviewers are also summarized in the required section, and will only be marked as passing when all required reviewers have approved.



(Figure 2) Policies section

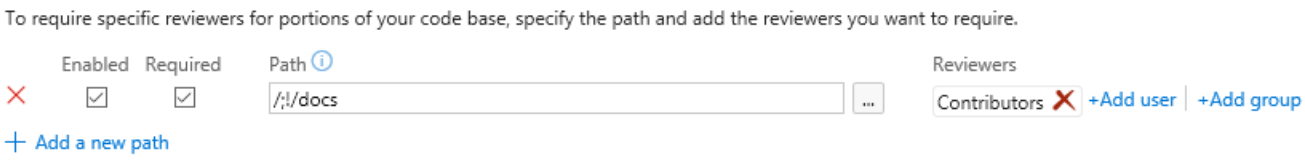
If you need to bypass policies (and have the required permissions), a new experience is displayed in the **Complete** dialog *(Figure 3)*. Any policies that are not met will be displayed in a warning message, and a new explicit option to opt-in to override policies will be presented. Checking the override option enables the **Override & Complete** action, which completes the PR, overriding any failing policies.



(Figure 3) Complete dialog

Support file exclusions in the required reviewer policy

When specifying required reviewers for specific file paths, you can now exclude paths by using a "!" prefix to the path you want to exclude. For example, you can use this to exclude a docs folder from your normally required signoff *(Figure 4)*.



(Figure 4) File exclusion support

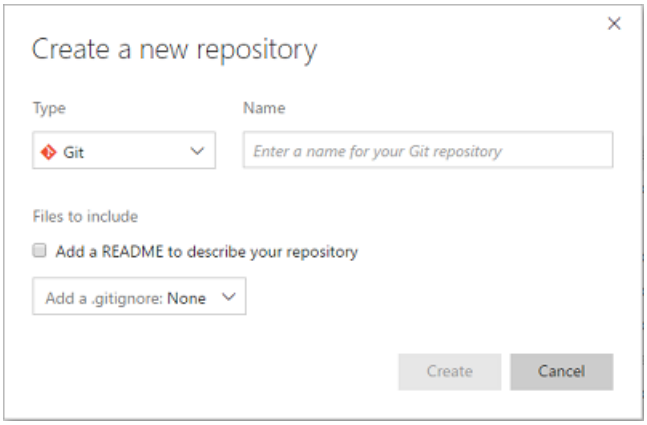
Import repository

You can now import a Git repository from GitHub, BitBucket, GitLab, or other locations. Import into either a new, or existing empty repository. For more information, see Import a Git repo.

Add .gitignore during repo creation

While creating a new Git repository, you can now add and associate a .gitignore file with your repository. A .gitignore file specifies files that Git should ignore while performing a commit.

The dialog allows you to select one of the many available .gitignore templates *(Figure 5)*.



(Figure 5) Add .gitignore during repo creation

Cherry-pick and revert

We have added two new features that make it easier to port or back out changes from the web portal: Cherry-pick and Revert.

Use the cherry-pick command to port changes in a pull request to multiple branches. A typical use case is when a bug needs to be hotfixed, but should also be fixed in the mainline. Once you have created your pull request that contains the fix to the hotfix branch, you can easily cherry-pick the same fix into the master branch. See Copy changes with cherry-pick for more information.

You can revert changes on completed PRs. Find the PR that introduced the bad change, click **Revert**, and follow the steps to create a PR that backs out the unwanted changes. For more information, see Undo Changes with Git.

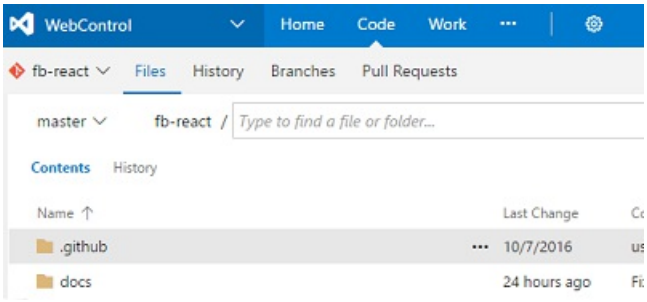
Configurable compare branch

You can now set your compare branch to something other than the default branch. This setting persists on a per-user basis. Pull requests and new branches created from the **Branches** page is based off the branch you set as the compare branch. See Manage your branches for more information.

Find a file or folder

You can quickly search for a file or folder in a repository using the **Code** hub in your Team Services project. The result lists items from your current folder followed by files and folders across the repository.

For any Git repository, go to the path control box *(Figure 6)*, and start typing to initiate a navigation search experience for the file, or folder you are looking for.



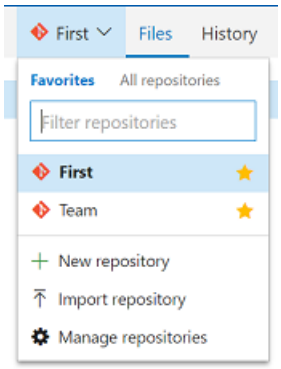
(Figure 6) Find a file or folder

Confirmation for deleting repos

To prevent accidental repository deletions, you now have to type the name of the repository that you wish to delete to confirm the action.

Repo favorites

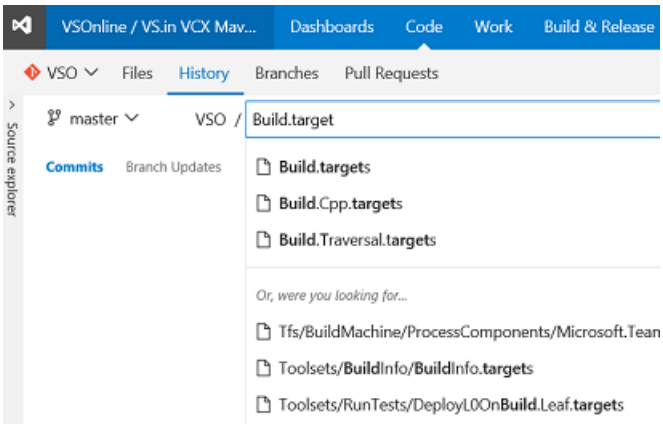
You can now favorite the repos you work with most frequently. In the repo picker *(Figure 7)*, you will see tabs for **All repositories** and your **Favorites**. Click the star to add a repository to your list of Favorites.



(Figure 7) Repo favorites

Search for a file or folder in commit history

Similar to the files tab, you can now search for a file or folder in a repository and see the history of commits for that file or folder. For any Git repository, go to the path control box on the History tab *(Figure 8)*, and start typing to initiate a history search experience for the file, or folder you are looking for.

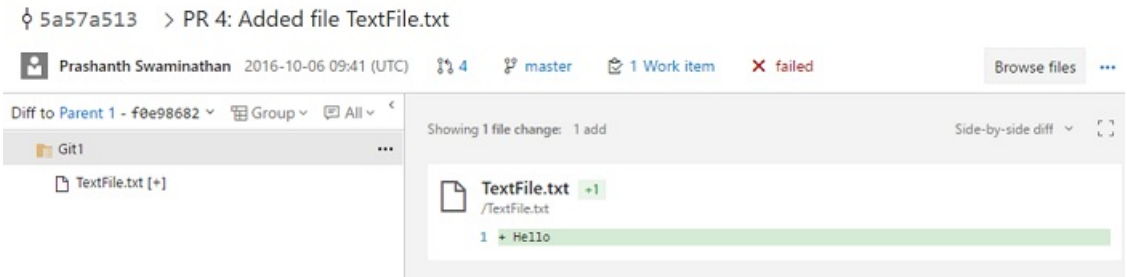


(Figure 8) Commit history

Commit page improvements

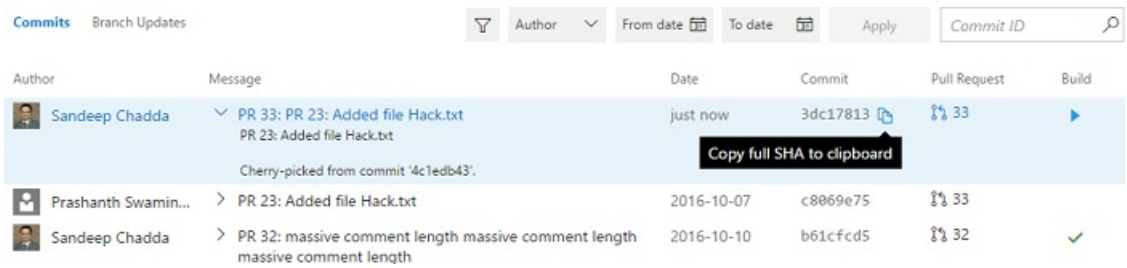
We made your experience of the commit details page and commit history page up-to-date and highly performant. You can now find, and act on, important information related to the commit at a bird's-eye view.

Here is an example of the commit details page *(Figure 9)*:



(Figure 9) Commit details

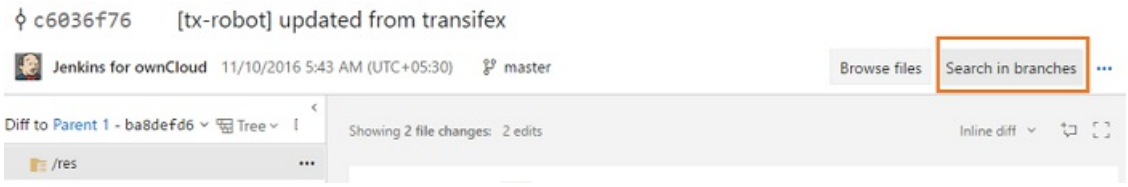
Here is the commit history page *(Figure 10)*:



(Figure 10) Commit history

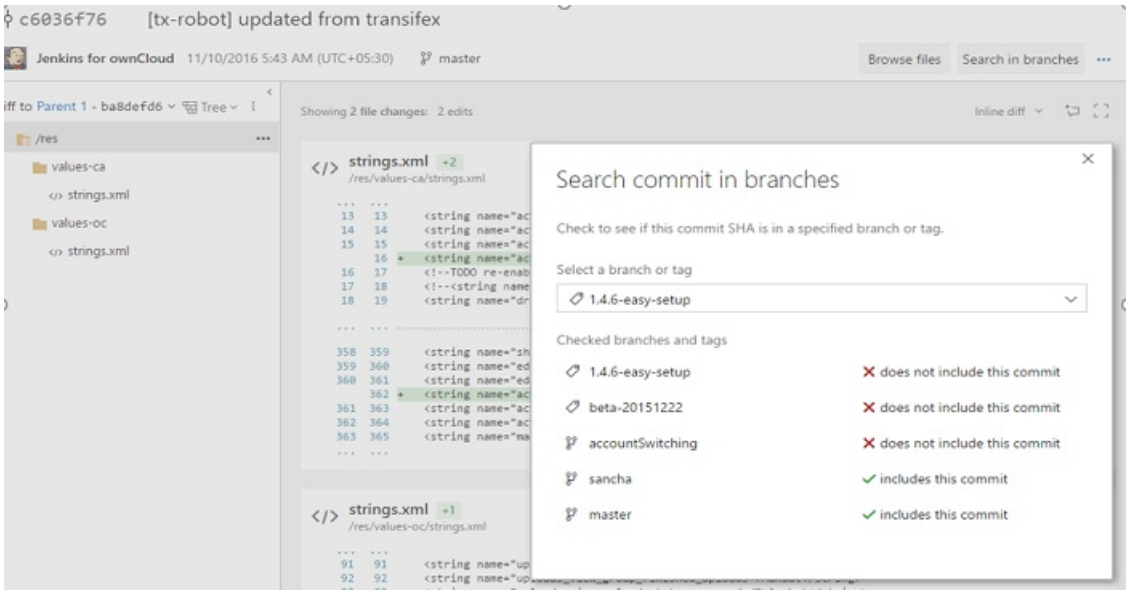
Search for commits in branches

You can now search for a commit in a specified branch or a tag by clicking on the **Search in branches** button on the commit details page *(Figure 11)*.



(Figure 11) Commit search

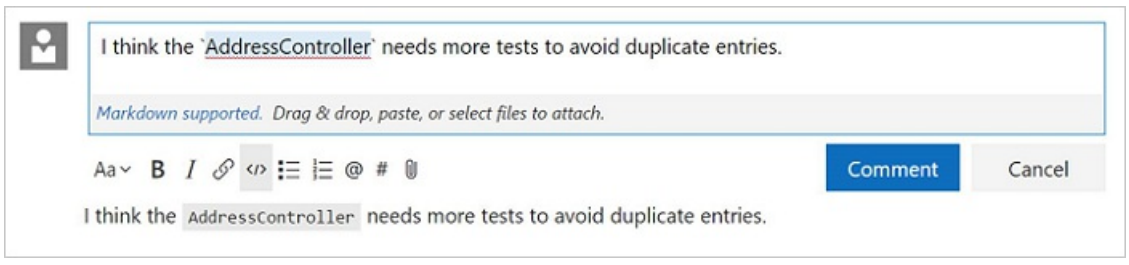
You can select tags and branches in the window to view, even if these branches and tags do not contain the particular commit *(Figure 12)*.



(Figure 12) Commit search dialog

Discussion control toolbar

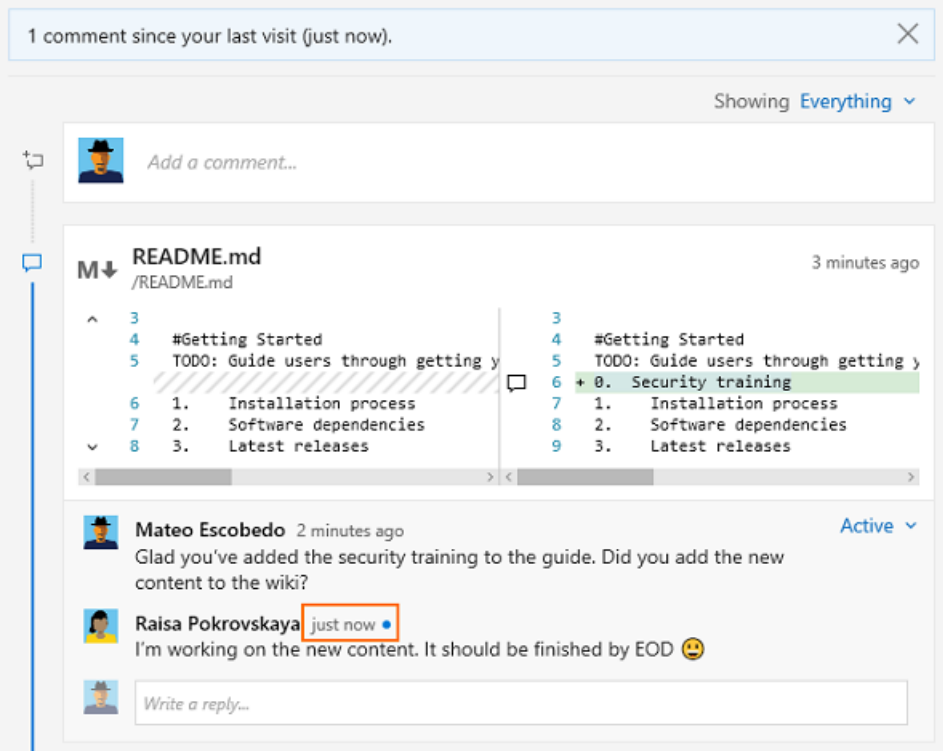
Markdown is a powerful tool when adding comments to pull requests, but it can be hard to remember the syntax. To make this easier, we have added a toolbar to the discussion control *(Figure 13)*. This inserts the appropriate Markdown syntax to add common formatting. Headings, boldface, italics, links, code, and lists can all be added using the new toolbar controls, and features like @ and # mentions can be entered using the toolbar as well. Keyboard shortcuts are available for boldface (CTRL + B), italics (CTRL + I), and creating links (CTRL + K).



(Figure 13) Discussion toolbar

PR comment improvements

To help you identify the new comments in a PR, we have added some additional decoration to the new replies in existing discussion threads. Comments in the files view also highlight threads that have new comments *(Figure 14)*.

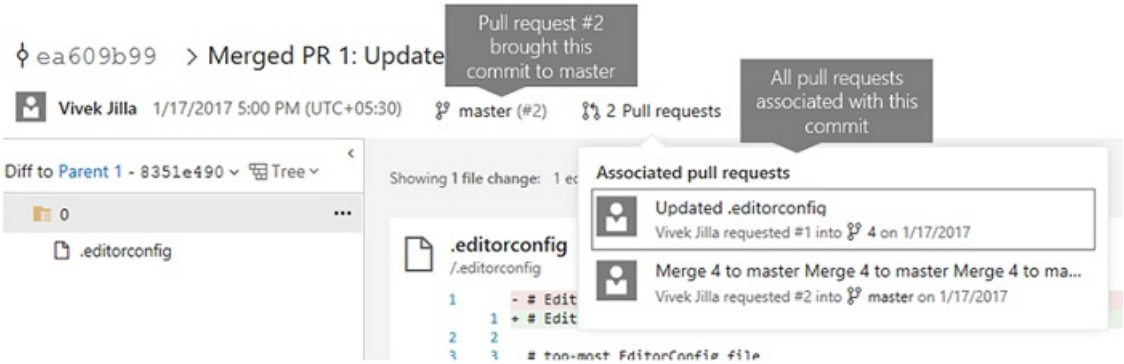


(Figure 14) PR comment improvements

View PRs for a commit

You can now view all associated pull requests for a commit on the **Commit details** page. From the image below *(Figure 15)*, you can see that:

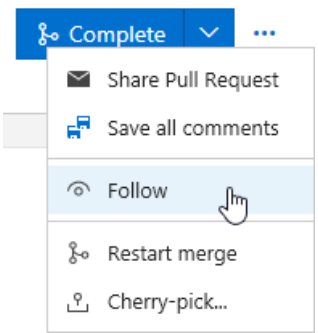
- In the associated pull request drop-down, there are two pull requests associated with this commit.
- Pull request #2 brought this commit to master.
- The same commit was brought into branch 4 through pull request #1.



(Figure 15) PR in commits

Follow a pull request

You can now follow a pull request to stay notified of any changes through email alerts. The option to **Follow** is available in the context menu *(Figure 16)*.



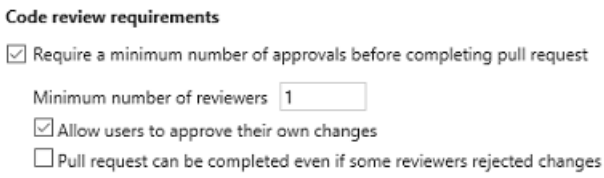
(Figure 16) Follow a pull request

Restart pull request merge

Another option has been added to re-attempt the merge for a pull request where the target branch has been updated. This **Restart merge** option is useful when you want to verify that recent changes to the target branch have not created conflicts or broken your PR build.

Completion blocked on rejected pull requests

Branches that have the code review policy set will show that the PR is unable to be completed if rejected by one or more reviewers. Many of you expected this behavior, so we have changed the default behavior. For teams that want the original behavior, there is a new option in the branch policy setting page *(Figure 17)*.



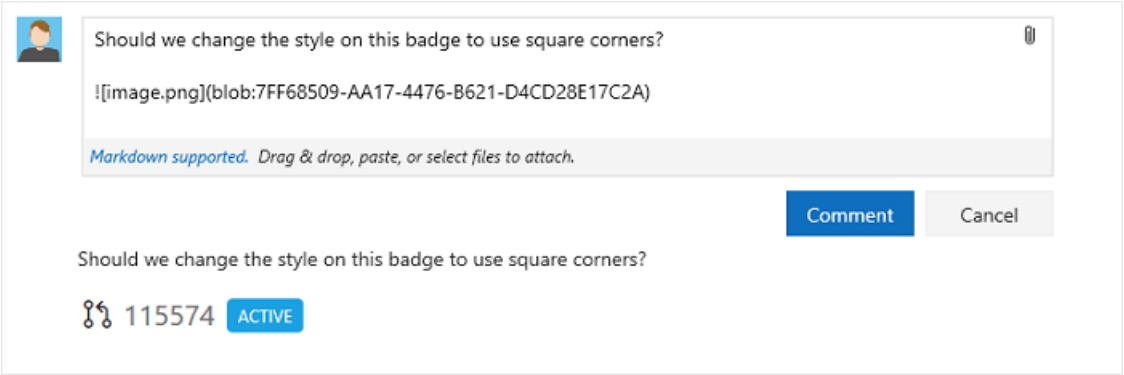
(Figure 17) Code review policy

Markdown in pull request description

Spice up your pull request description with Markdown. The same Markdown support you know and love in pull request comments is now available in the pull request description.

Attachments in PR discussions

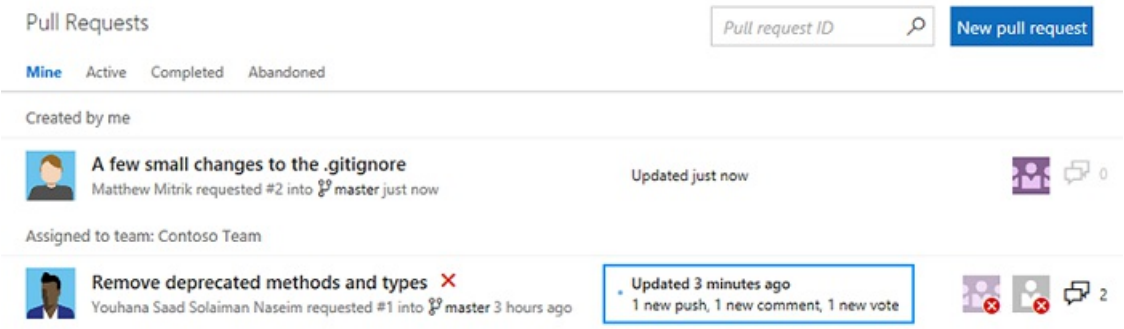
You can now add attachments to your pull request comments *(Figure 18)*. Attachments can be added by drag-and-drop or by browsing. For images, attachments are added by simply pasting from the clipboard. Adding an attachment automatically updates the comment to include a Markdown reference to the new attachment.



(Figure 18) Attachments in PR discussions

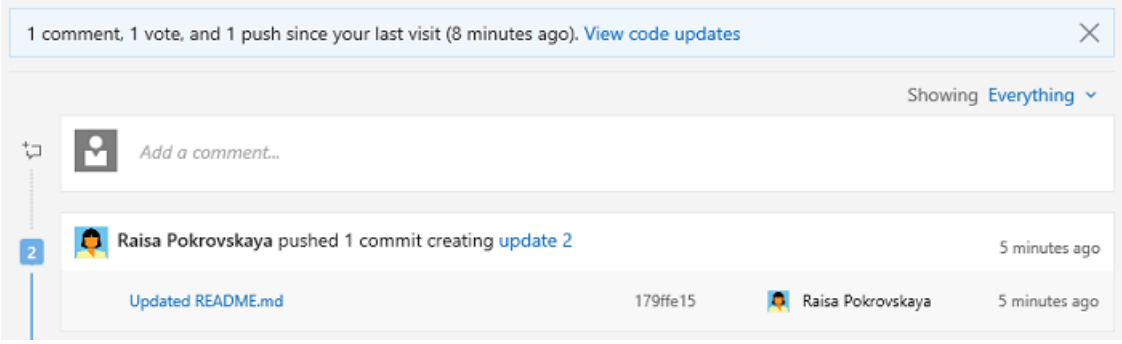
Highlight the PRs that have updates

It is now easier than ever to see the updates to your pull requests. In the PR list view *(Figure 19)*, PRs with changes since you have last seen them, are shown with a new updates column that shows a roll-up of the changes.



(Figure 19) PR updated files

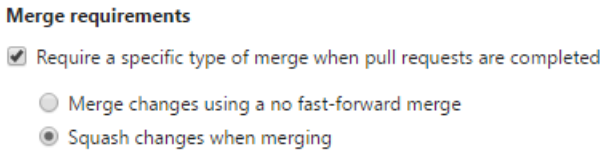
When you view a PR that has changes, you will see a similar summary message in the overview. New pushes and comment threads are highlighted in blue *(Figure 20)*. Clicking the **View code updates** link will navigate to the **Files** view, where a diff of the new changes since you last viewed the pull request is shown. This feature makes it easy to follow up on a PR where the author made changes in response to feedback.



(Figure 20) PR summary

Branch policy for PR merge strategy

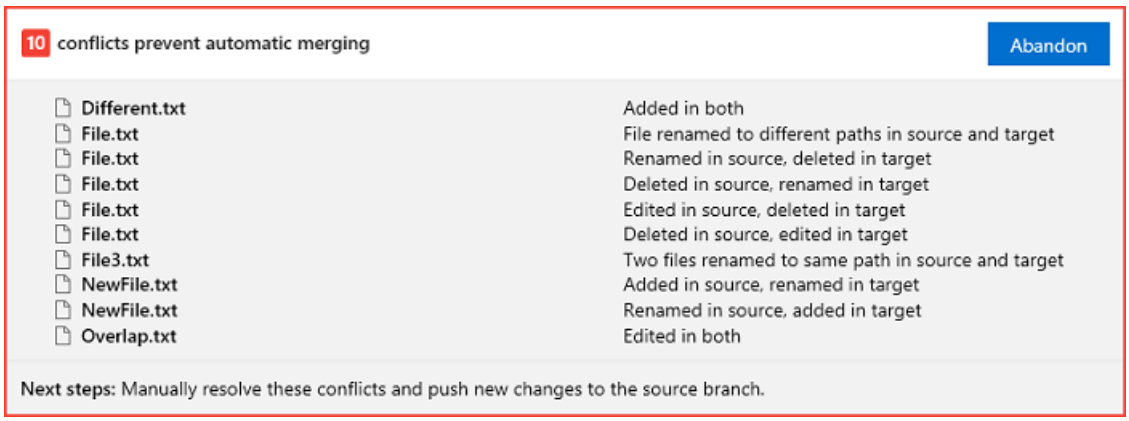
We have added a new branch policy *(Figure 21)* that lets you define a strategy for merging pull requests for each branch. Previously, you chose the decision to either merge or squash at the time a PR was completed. If enabled, this policy will override your preferences, enforcing the requirement set by the policy.



(Figure 21) Branch policy

Expose merge conflict information

If there are any files with conflicts in a pull request, the details about those conflicts is now visible in the overview *(Figure 22)*. Each conflicting file is listed, along with a short summary of the type of conflict between the source and target branches.



(Figure 22) Merge conflicts

Markdown preview button

When viewing a diff of a markdown file in a commit, push, or pull request, you can now easily toggle to see the resulting rendered view.

Work Item Tracking Improvements

Improved search experience for scoped identity fields

With this release, we updated the identity picker behavior for scoped identity fields, i.e. identity fields that are configured to only allow assignment to a specific group of users. In the updated experience, the picker's MRU list

and search results will only return members of the configured group, rather than show results for all valid users for the collection.

Build Improvements

Rollback build definitions

You can roll a build definition back to a previous version. You can do this when editing a build definition by going to the **History** tab.

Disable the sync and checkout of sources in a build

You can optionally disable the automatic source sync and checkout for Git. This will enable you to handle the source operations in a task or script, instead of relying on the agent's built-in behavior. All standard source-related variables like Source.Version, Source.Branch, and Build.SourcesDirectory are set.

Git shallow clone and git-lfs

The build agent now supports Git shallow clone and git-lfs. For more details, see the Build definition repository page.

Task versioning for Build and Release definitions

We have given you control over the major version of a task that you run in your build or release. This change results in fewer unpredictable errors that are caused by automatic updates to the agent and task version. You now specify the major version of the task on the **Build** tab of your definition, or on the **Environments** tab of your release definition.

When a minor version is released (for example, 1.2 to 1.3), you get that change automatically in your build. But if a new major version is released (for example 2.0), then your build stays locked to version 1.3 until you edit the definition and manually change to the new major version. A flag in the build definition alerts you to new major versions.

Payment required for Package Management

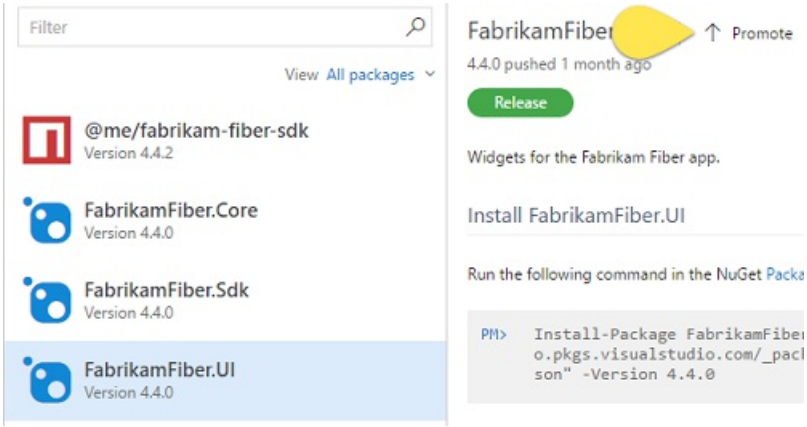
To continue using Package Management, you will either need a Visual Studio Enterprise subscription or a Package Management license purchased in the Marketplace. You can read more about licensing Package Management.

Package Improvements

Release views in Package Management

We have added a new feature to Package Management called **release views** *(Figure 23)*. Release views represent a subset of package-versions in your feed that you have promoted into that release view. Creating a release view and sharing it with your package's consumers enables you to control which versions they take a dependency on. This is particularly useful in continuous integration scenarios where you are frequently publishing updated package versions, but may not want to announce or support each published version.

Look for the quick start in Web Access or learn about release views for package CI/CD to get started.

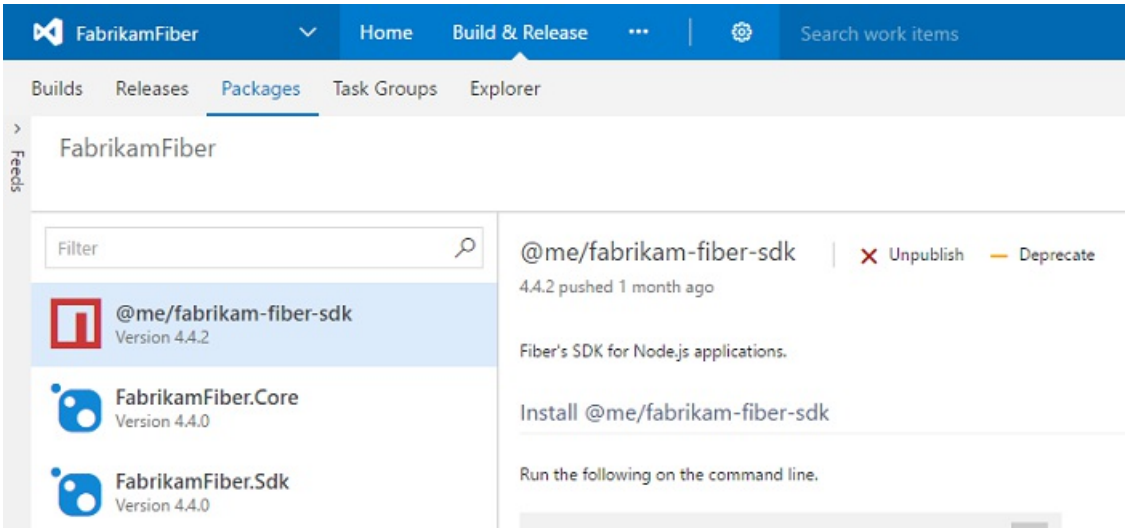


(Figure 23) Release views

npm in Package Management

Package Management feeds now support npm packages for Node.js and JavaScript development. In addition, npm feeds support npmjs.com as an "upstream source with caching." By enabling this option, your feed will transparently proxy and cache packages from npmjs.com (see Use packages from npmjs.com, which means that you will only need to get a particular package@version from npmjs.com once; future requests for that package are served directly from your TFS server. If a package is removed from npmjs.com, you will still be able to get the cached version from TFS.

To get started, look for the new npm option in the **Connect to feed** dialog *(Figure 24)*.

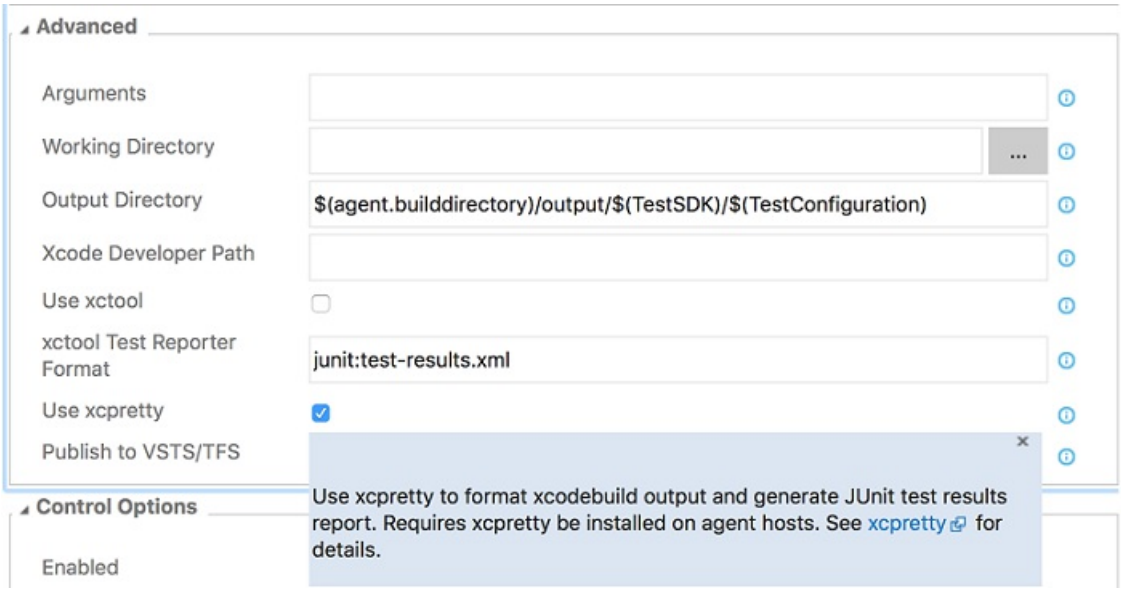


(Figure 24) npm in Package Management

Cross Platform Improvements

Xcode Build task xcpretty formatting

You can now format your xcode build output with xcpretty *(Figure 25)*. You can also publish JUnit test results to Team Services with xcodebuild. Previously, xctool had to be used as the build tool to publish test results. Now, to enable xcpretty, check **Use xcpretty** and uncheck **Use xctool** in the **Advanced** section of the Xcode task.



(Figure 25) Xcpretty formatting

Publish Jenkins test and code coverage results

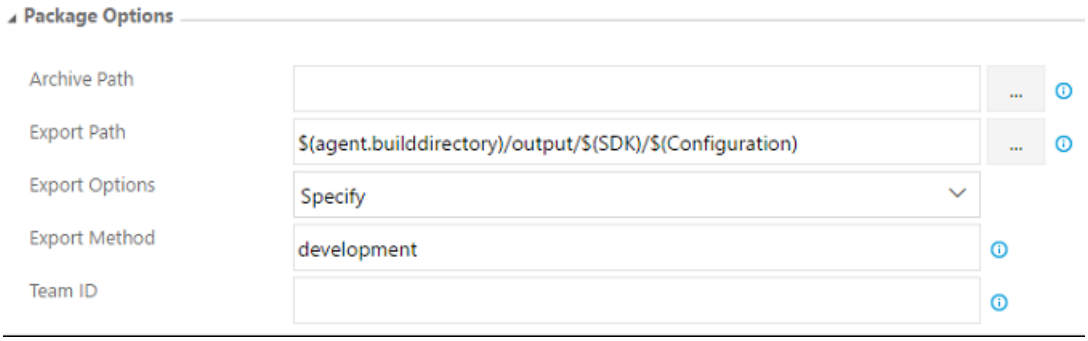
The **Jenkins Queue Job** build and release task can now retrieve test and code coverage results from a Jenkins job or pipeline. This requires installation of the TFS Plugin for Jenkins 5.2.0 or later on your Jenkins server and configuring the post-build action **Collect Results for TFS/Team Services**. After retrieving results from Jenkins, they are published with the **Publish Test Results** or **Publish Code Coverage** build tasks.

Xcode 8 signing and exporting packages in the Xcode Build Task

The Xcode task now supports building your projects using Xcode 8 automatic signing *(Figure 26)*. You can install the certs and provisioning profiles on the build server manually, or have the task install them by specifying the **File Contents** options.

(Figure 26) Xcode automatic signing

Xcode 8 requires specifying an export options plist *(Figure 27)* when exporting an app package (IPA) from an archive (.xcarchive). The Xcode task now automatically identifies the export method if you are using Xcode 8 or Xcode 7. You can specify the export method or specify a custom plist file from the Xcode task. If you are using an Xcode version older than Xcode 7, the task falls back to using the old tool (xcrun) for creating the app package.

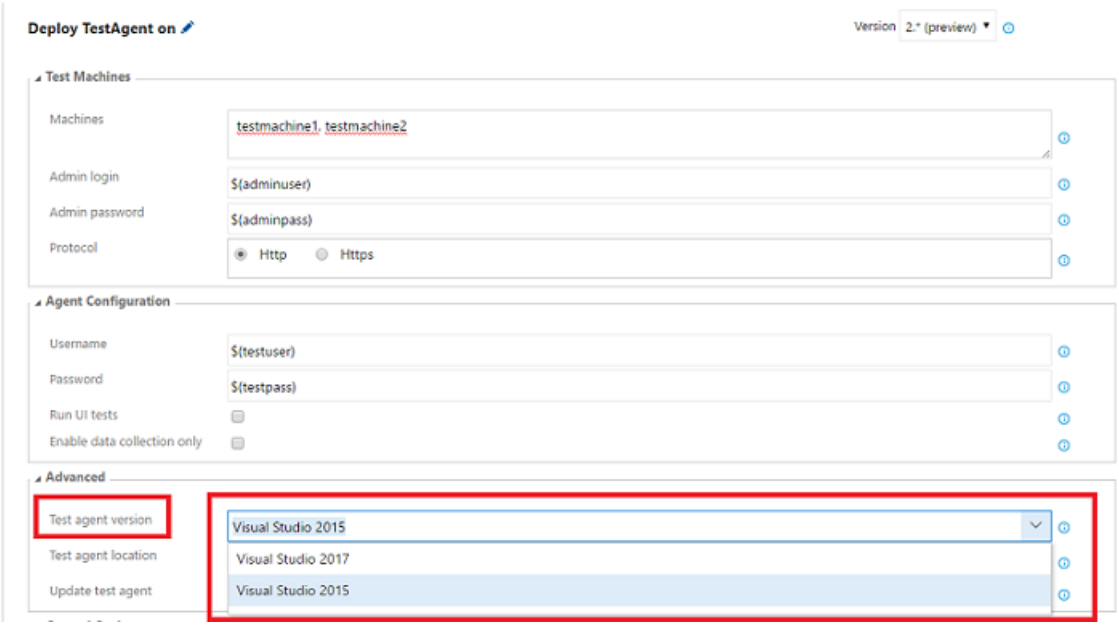


(Figure 27) Xcode export options

Test Improvements

Run tests built using Visual Studio 2017

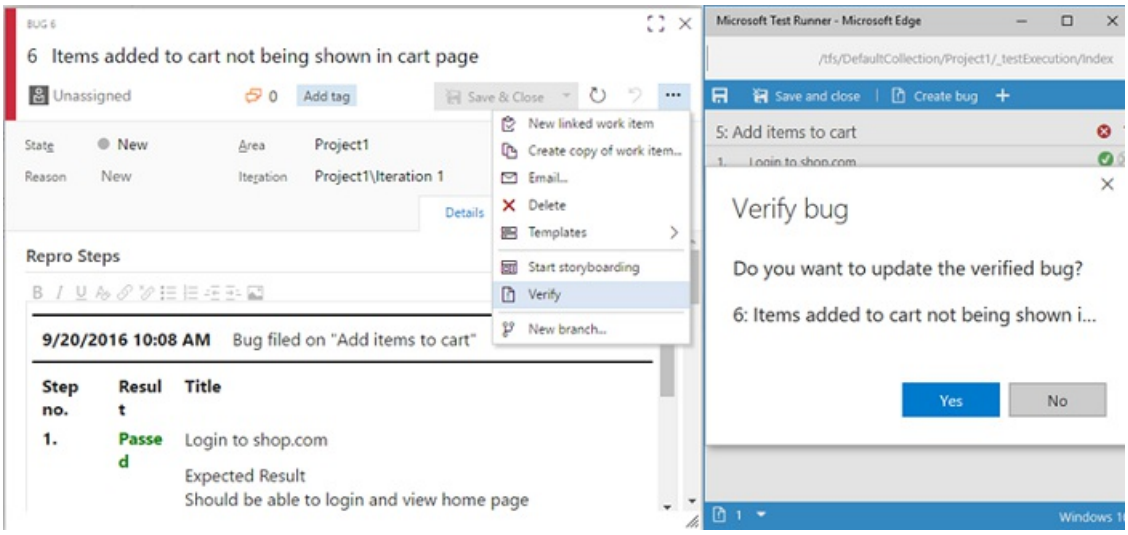
Using the **Deploy Test Agent** and **Run Functional Tests** tasks in CI/CD pipeline *(Figure 28)*, you can now install Test Agents for Visual Studio 2017 and run tests that were built using Visual Studio 2017.



(Figure 28) Run tests

Verify bugs from work item

You can now verify a bug by re-running the tests that identified the bug *(Figure 29)*. You can invoke the **Verify** option from the bug work item form context menu to launch the relevant test case in the web runner. Perform your validation using the web runner, and update the bug work item directly within the web runner.



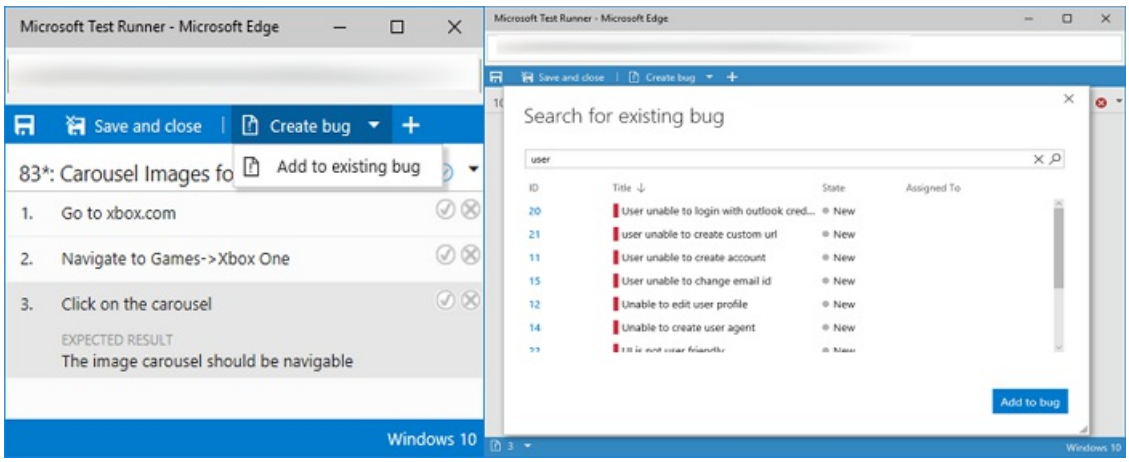
(Figure 29) Verify bugs from work item

REST client helpers for Test Step operations

You will now be able to create, modify, and delete test steps and test step attachments in Test Case work items using the helper classes we have added to the REST client (see the RestApi-Sample).

Update existing bugs from Web Runner

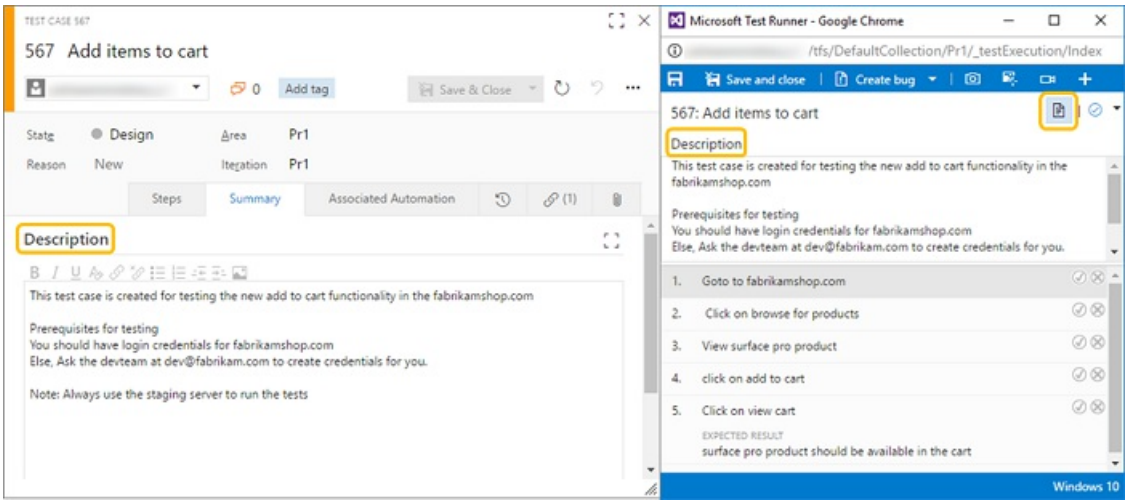
In addition to creating new bugs from the Web runner, now you can also update an existing bug *(Figure 30)*. All the diagnostic data collected, repro steps, and links for traceability from the current session are automatically added to the existing bug.



(Figure 30) Update existing bug

Test case description in Web Runner

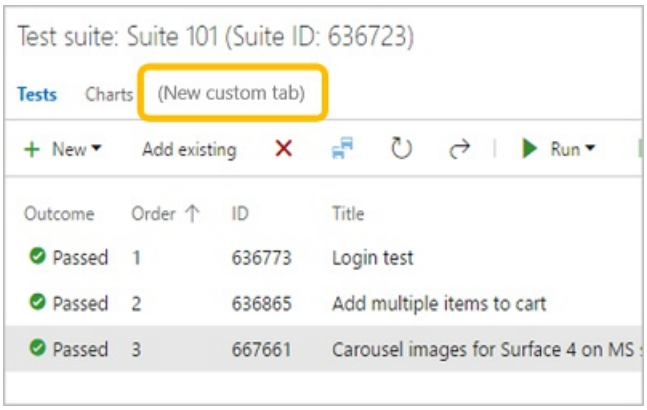
The test case description field was often used for capturing the prerequisites required before the test case execution can start. With this update, you are now be able to view the test case description information in the Web runner by using the **Show description** option *(Figure 31)*.



(Figure 31) Test case description

Test hub contribution point

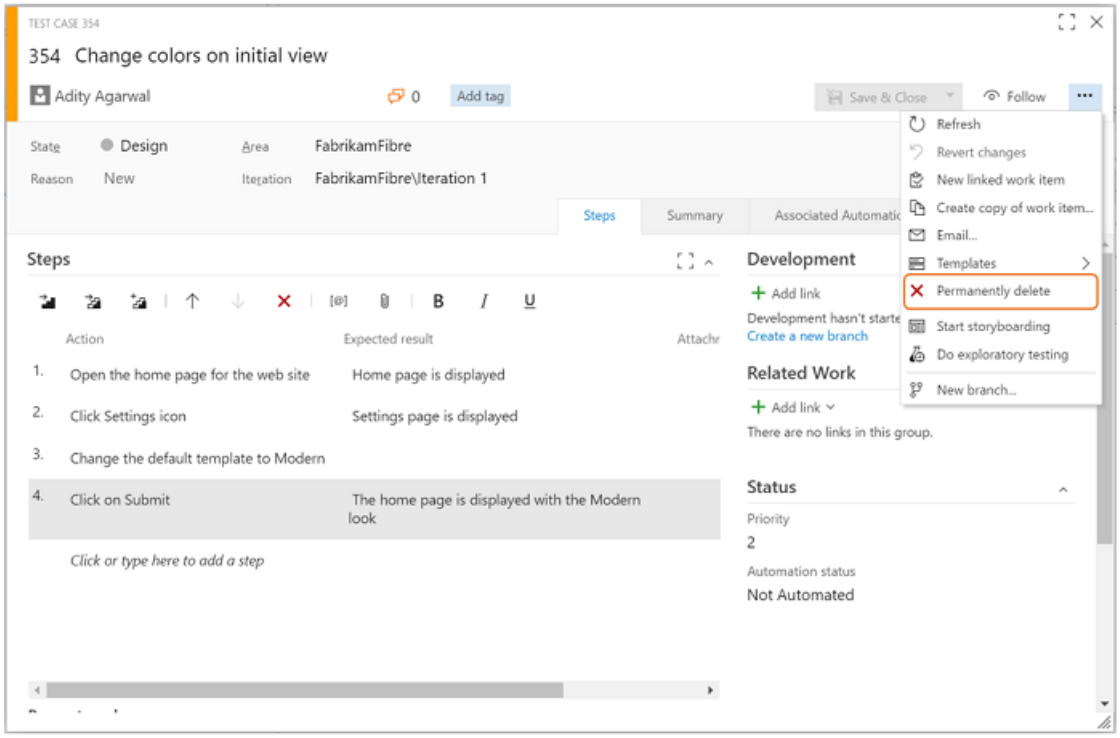
We have added a new contribution point ("ms.vss-test-web.test-plan-pivot-tabs") *(Figure 32)* within the **Test plan** hub to allow developers to write extensions as a pivot tab that appears next to the **Tests** and **Charts** tab.



(Figure 32) Contribution point

Delete test artifacts

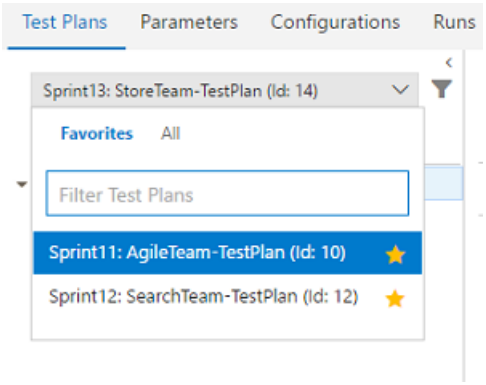
Prior to this release, your delete option was limited to work items. With this update, you now have the ability to permanently delete test artifacts—test plans, test suites, test cases, shared parameters, and shared steps—both from the **Test** hub and the **Work** hub, by using the **Permanently delete** *(Figure 33)* option in the work item form context menu.



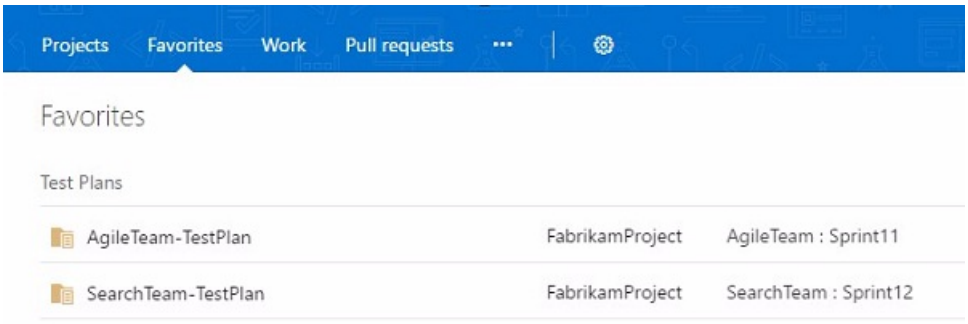
(Figure 33) Delete test artifacts

Favorites for Test Plans

You can now favorite the Test Plans you work with most frequently. In the **Test Plans** picker, you will see tabs for **All** your Test Plans and **Favorites** *(Figure 34)*. Click the star icon to add a Test Plan to your list of favorites. The favorited Test Plans are accessible in the Test Plans picker and from the **Favorites** tab in the new account home page. You can also filter Test Plans by searching on the title field *(Figure 35)*.



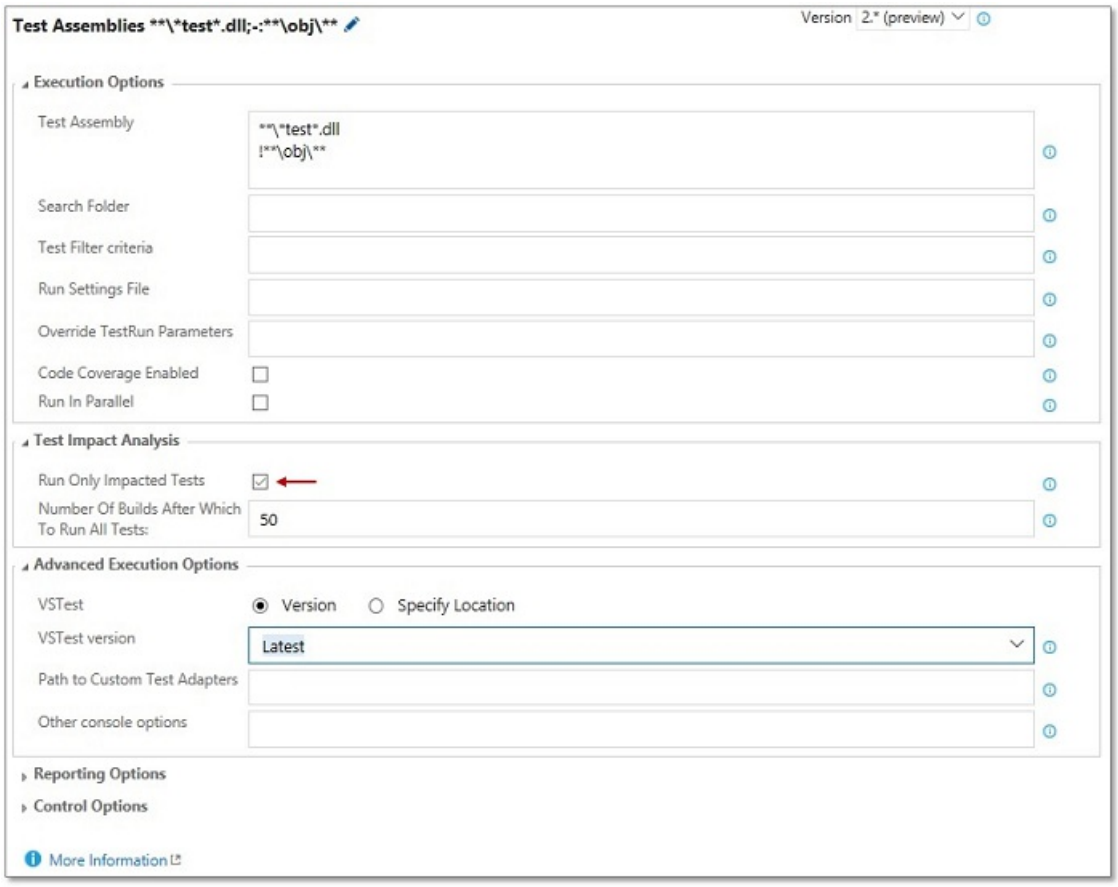
(Figure 34) Test plans



(Figure 35) Test favorites

Test Impact Analysis for managed automated tests

Test Impact Analysis for managed automated tests is now available through a checkbox in the **Version 2.* (preview)** version of the VSTest task *(Figure 36)*.



(Figure 36) Test impact analysis

If enabled, only the relevant set of managed automated tests needed to validate a given code change, will run. Test Impact Analysis requires the latest version of Visual Studio, and is presently supported in CI for managed automated tests.

Firefox support for Test & Feedback extension

We are happy to announce the General Availability of the Test & Feedback extension for Firefox. You can download the Firefox add-on from our marketplace site.

Note: Support for the Edge browser is also in the works; stay tuned for more updates.

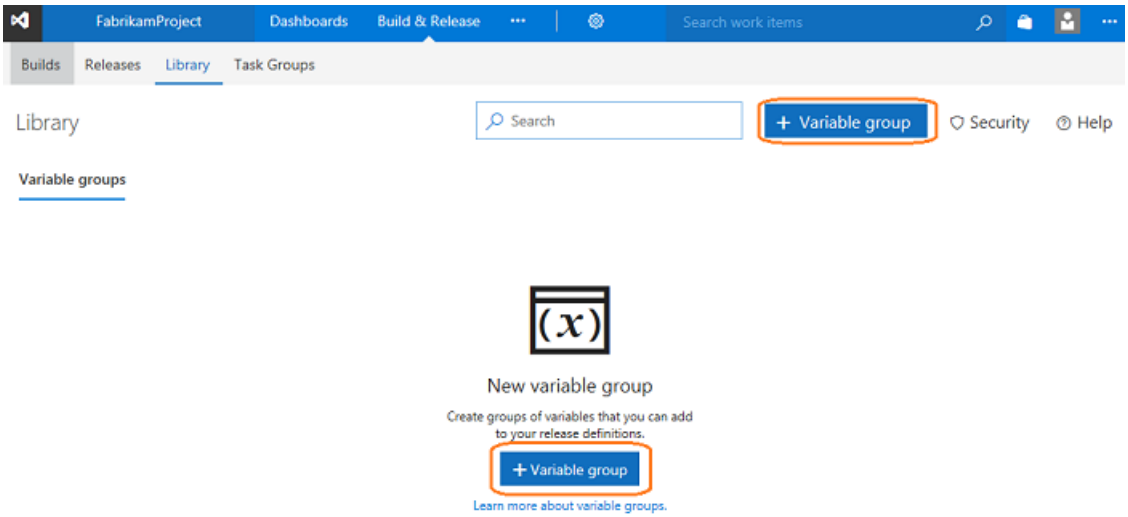
Release Management Improvements

Variable groups support in Release

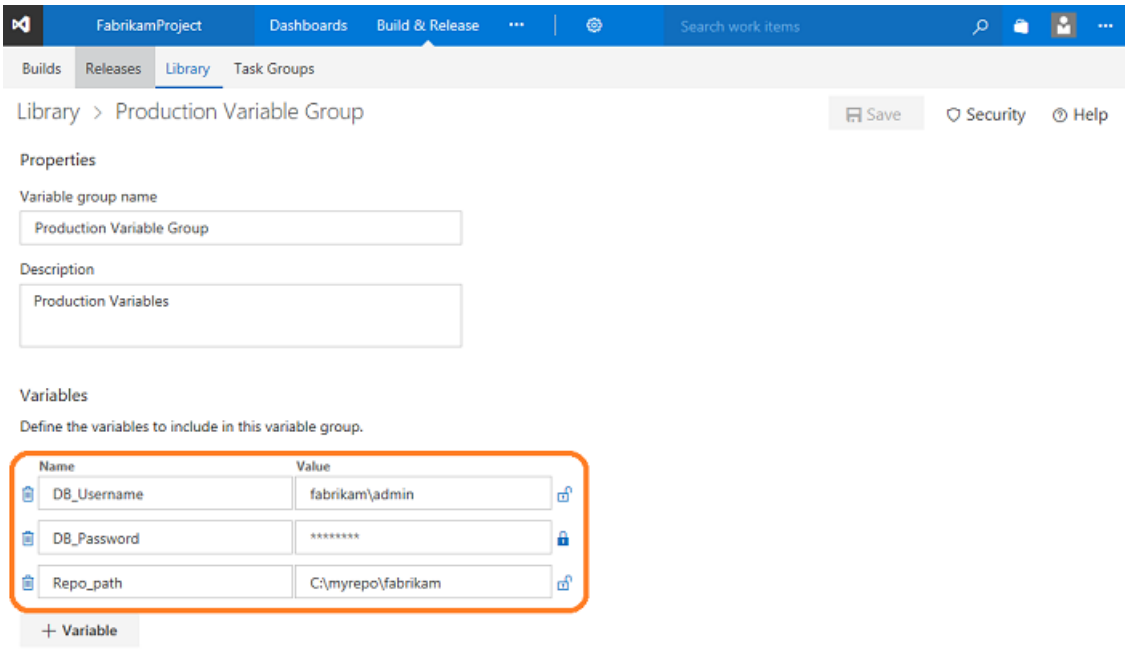
Variable groups are used to group your variables and their values to make them available across multiple release definitions. You can also manage security for variable groups and chose who can view, edit, and consume the variables from the variable groups in your release definitions.

Open the **Library** tab in the **Build & Release** hub and choose **+ Variable group** in the toolbar *(Figure 37)*. Currently, variable groups are consumed only in release definitions. Find more information about variable groups, Release definitions in Microsoft Release Management.

Create *(Figure 37)*, then edit *(Figure 38)* a variable group, as shown below:



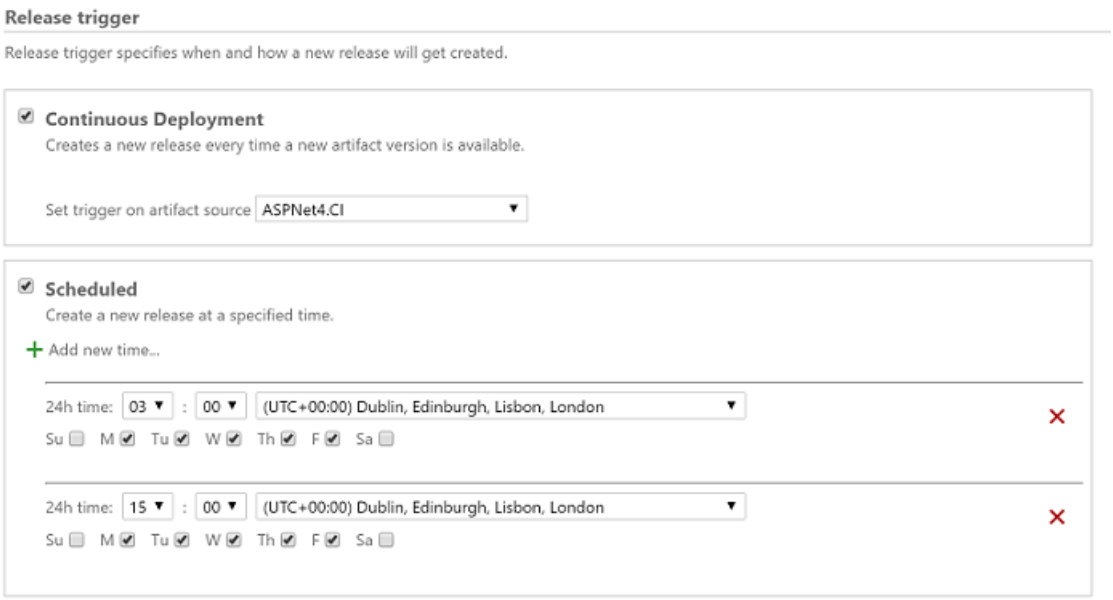
(Figure 37) Create variable group



(Figure 38) Edit variable group

Multiple schedules in releases

Want to schedule your releases to be created more than once a day? You can now configure multiple scheduled triggers in a release definition *(Figure 39)*.



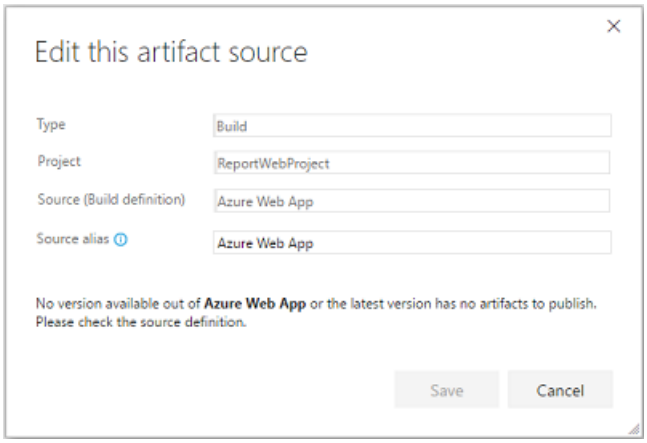
(Figure 39) Release schedule

Inline service connections in Build and Release

With this feature, you can create service connections right in the build/release definition without navigating to the **Services** tab. This is auto-enabled for all extensions that are defined declaratively, such as Docker, Jenkins, VMWare, and SCVMM.

Link build artifacts from another team project

Until now, release definitions could only link artifact sources from the current project. Now, you can link build artifacts *(Figure 40)* from another project as well. While linking an artifact, the project drop down will list all the projects in the account.

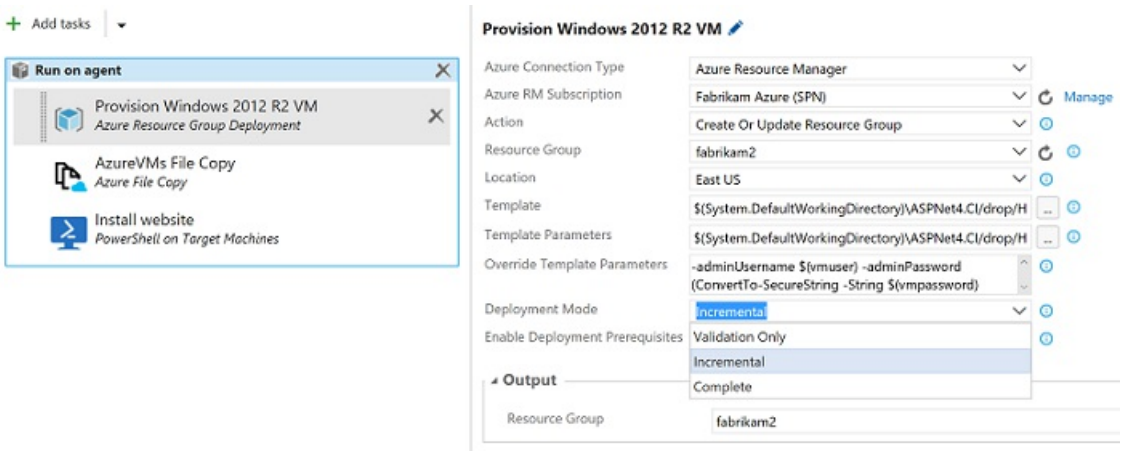


(Figure 40) Link build artifacts

Azure resource group improvements

Prior to this release, the Azure resource group task could not validate the ARM template syntax, or it would be accepted without actually deploying the resources. This enhancement allows a new deployment mode called **Validation Only**, where you can find problems with the template authoring before creating actual Azure resources.

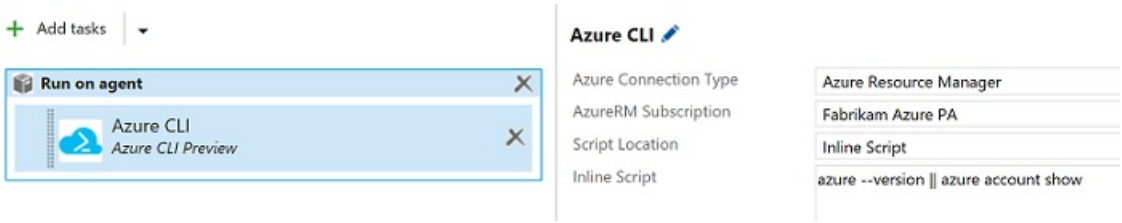
Another enhancement to the Azure resource group task is to allow either incremental, or complete deployments *(Figure 41)*. Previously, the task deployed the ARM templates using the Incremental mode. However, it did not modify resources that existed in the resource group not specified in the template. Complete mode deletes resources that are not in your template. The default is incremental mode.



(Figure 41) Azure resource groups

Azure CLI task

The new Azure CLI task *(Figure 42)* supports running Azure CLI commands on cross platform agents like Windows, Linux, and Mac. The task supports both Classic and ARM subscriptions. It supports two modes of providing the script, one as a linked artifact and another as an inline script.



(Figure 42) Azure CLI task

Code Search Update

In TFS 2017 Update 1, the Code Search service includes Elasticsearch version 2.4.1. If the Code Search service is configured on a server running TFS 2017, the Code Search service will update as part of the TFS upgrade. If the Code Search service is configured on a remote server, then copy the content of the **Search Service Package** provided with the installer to the remote machine and follow the instructions in the readme file to upgrade the search service manually.

Code Insights Improvements

SonarQube MSBuild tasks

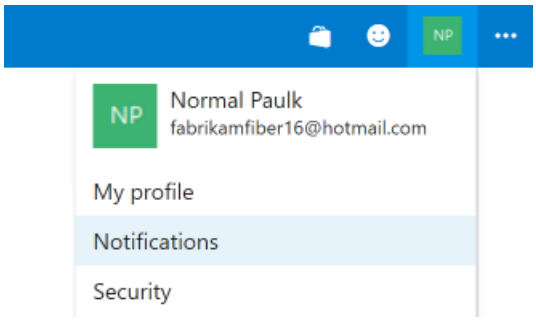
SonarQube MSBuild tasks are now available from an extension provided by SonarSource. For more details, please read SonarSource have announced their own SonarQube Team Services / TFS integration.

Administration Improvements

New notification settings experience

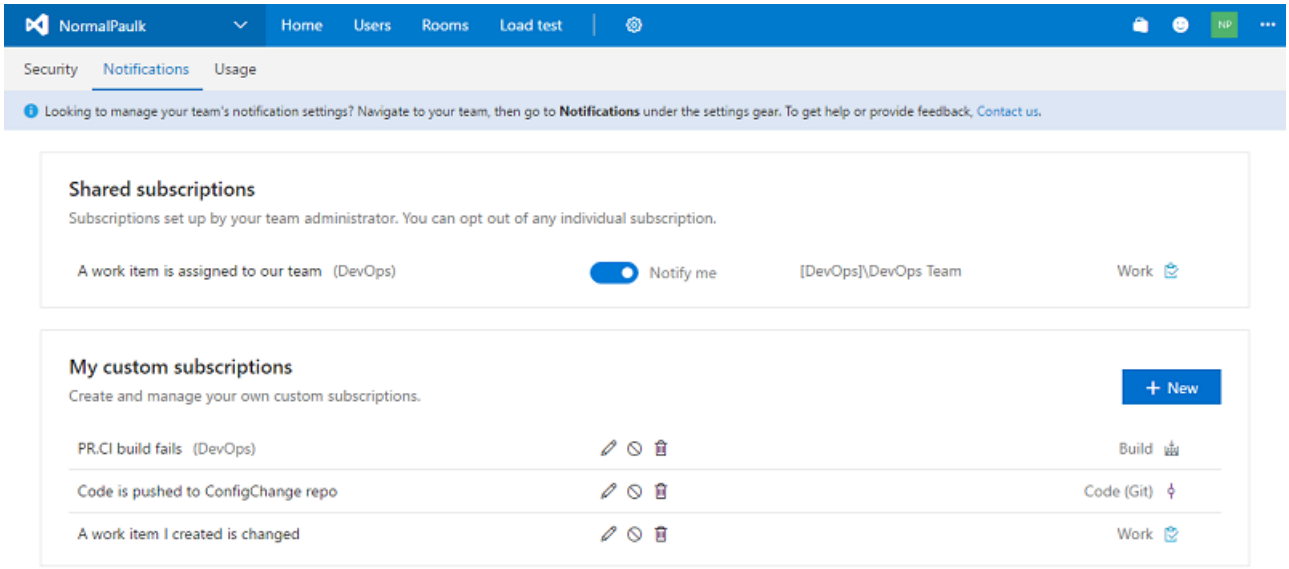
Notifications help you and your teams stay informed about activity in your Team Services projects. With this update, it is now easier to manage what notifications you and your teams receive.

You now have your own account-level experience in the profile menu for managing your notifications setting *(Figure 43)*.



(Figure 43) Notification settings

This view lets you manage personal subscriptions you create *(Figure 44)*. It also shows subscriptions created by team administrators for all projects in the account.



(Figure 44) Manage personal subscriptions

Learn more about managing personal notification settings.

addProjectReports is now in TfsConfig

You can now use the command addProjectReports to add reports to your team projects. This was a previous Power Tool command and is now part of the TfsConfig.exe command. For more information, see Upload reports to a team project.

Team Room Deprecation

With so many good solutions available that integrate well with TFS and Team Services, such as Slack and Microsoft Teams, we made a decision to deprecate our Team Room feature from both TFS and Team Services. If you are working in Team Services, you will see a new yellow banner appear that communicates our plan. Later this year, we plan to turn off the Team Room feature completely.

There are several alternatives you can use. The Team room is used both for a notification hub as well as for chat. TFS and Team Services already integrate with many other collaboration products including Microsoft Teams, Slack, HipChat, Campfire, and Flowdock. You can also use Zapier to create your own integrations, or get very granular control over the notifications that show up.

See more about the deprecation of Team Rooms in Team Services.

Markdown no longer supports file links

With Update 1, welcome pages, the markdown widget on team dashboards, and the Definition of Done on the Kanban boards will no longer support file links in their Markdown. As a workaround, you can include your file link as text in the Markdown. For more information, see Markdown guidance.

Announcing the Process Template Editor

We have released the Process Template Editor extension for Visual Studio 2017. This extension provides a convenient method for viewing and updating process templates, as well as tools for updating global lists and work item types, and viewing the attributes of work item fields. This works against TFS 2017 and TFS 2017 Update 1 servers.

Known Issues

Build doesn't work when upgrading to TFS 2017 Update 1 build 15.112.26301.0 from TFS 2013 or earlier

- **Issue:**

  Note that this issue only occurs if you upgraded to TFS 2017 Update 1 build 15.112.26301.0, released on 7 March, 2017. If you upgraded to build 15.112.26307.0, released on 9 March, you will not encounter this.

  After upgrading from TFS 2013 (RTM or any update) or earlier, Build shows an error of "counter with name TaskReferenceId does not exist".

- **Workaround:**

  Run the following script on your upgraded collection databases:

```
INSERT  tbl_Counter (PartitionId, DataspaceId, CounterName, CounterValue)
SELECT  DISTINCT
          dpm.PartitionId,
          ds.DataspaceId,
          N'TaskReferenceId',
          1
  FROM    tbl_DatabasePartitionMap dpm
  INNER LOOP JOIN Task.tbl_Hub h
  ON      h.PartitionId = dpm.PartitionId
  INNER LOOP JOIN tbl_Dataspace ds
  ON      ds.PartitionId = dpm.PartitionId
          AND ds.DataspaceCategory = h.DataspaceCategory
          AND ds.DataspaceIdentifier <> '00000000-0000-0000-0000-000000000000'
  WHERE   dpm.PartitionId > 0
          AND dpm.HostType = 4
          AND NOT EXISTS (
              SELECT  *
              FROM    tbl_Counter c
              WHERE   c.PartitionId = dpm.PartitionId
                      AND c.DataspaceId = ds.DataspaceId
                      AND c.CounterName = N'TaskReferenceId'
          )
```

Customers should update to Git LFS version 1.3.1 or higher

- **Issue:**

  Git LFS versions before 1.3.1 are no longer supported.

- **Workaround:**

  If you are using Git LFS, you must update to Git LFS version 1.3.1 or higher. Older versions of the LFS client are not compatible with authentication changes in this version of TFS.

Work item forms do not render correctly

- **Issue:**

  If you use a legacy custom control in your work item forms, such as the legacy multi-value control, your work item forms may fail to render.

- **Workaround:**

  You will need to update to the latest version of your control. You can find the latest multi-value control for TFS 2017 Update 1 here.

Work item forms do not render correctly in the web

- **Issue:**

  If you have a custom control, such as the multi-value control, installed for the Visual Studio client but not the web client, work item forms in the web fail to render.

- **Workaround:**

  You can add a web layout that does not contain the missing control element. You can have different layout targeting visual studio and web. For more info on the layout, see All FORM XML elements reference (TFS 2015).

Work item forms do not hide read only fields

- **Issue:**

  If you use old workitem form with **HideReadonlyEmptyFields** property set to true in the layout, your form will fail to hide read only and empty fields.

- **Workaround:**

  There is no workaround at this time. This will be fixed in TFS 2017 Update 2.

Work item forms become dirty on viewing

- **Issue:**

  This issue is specific to IE 11 on TFS 2017 Update 1 when opting into the new work item form. If you have your profile set to French, Korean, Russian, Turkish, Japanese, or Chinese, and the work item is assigned to any identity, you will see the work item form as dirty when viewing the work item. If you save the work item, the Assigned To field is set to unassigned.

- **Workaround:**

  Use another browser besides IE11. If you are using IE11, click undo/refresh from the work item toolbar to restore the correct Assigned To value.

Caching of upstream NPM packages fail

- **Issue:**

  If your TFS server is behind a proxy, the caching of upstream NPM packages will fail.

- **Workaround:**

  If your TFS server is behind a corporate proxy, make the following changes to your TFS server web.config (i.e. %ProgramFiles%\Microsoft Team Foundation Server 15.0\Application Tier\Web Services\web.config).

  Replace this configuration block:

```
<!-- ASP.NET Proxy Usage for HttpWebRequests
      "usesystemdefault"
         false - stops the server using the default proxy configuration or proxy
               auto-detection.
      "bypassonlocal"
         true - this tells all requests to a local address to ignore configured proxies.
  -->
<defaultProxy>
  <proxy usesystemdefault="False" bypassonlocal="True" />
</defaultProxy>
```

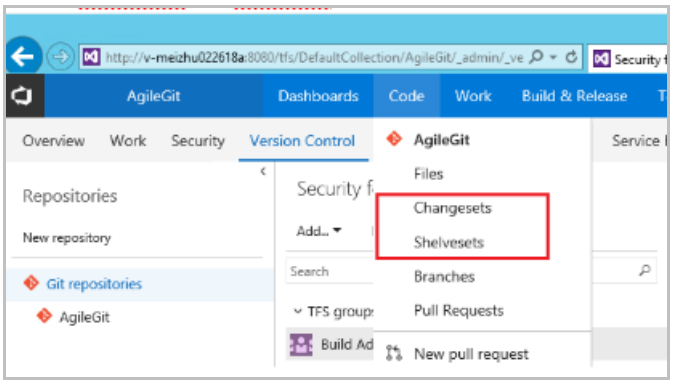
  With this:

```
<defaultProxy useDefaultCredentials="true" />
```

Code dropdown menu shows incorrect Version Control pages

- **Issue:**

  If you navigate to the admin page on a Git repository, as shown in the image below *(Figure 45)*, and click on the **Code** hub, they will see the **Changesets** and **Shelvesets** links, instead of the **History** link.



(Figure 45) Code dropdown menu

- **Workaround:**

  Navigate out of the Git repository admin page and you will see the correct links.

Extensions are not being auto-updated

- **Issue:**

  If you upgrade a prior version of TFS to reach TFS 2017 and are running TFS 2017 in connected mode then your extensions will not be auto-updated as they should be.

- **Workaround:**

  There is no workaround at this time. We have fixed the issue and the auto update behavior will reach you through TFS 2017 Update 2. If for any reason you cannot wait for Update 2 then reach us through the Support channel and we shall share the fix earlier.

Extensions cannot not be acquired or will not function correctly if Public URL is not set correctly

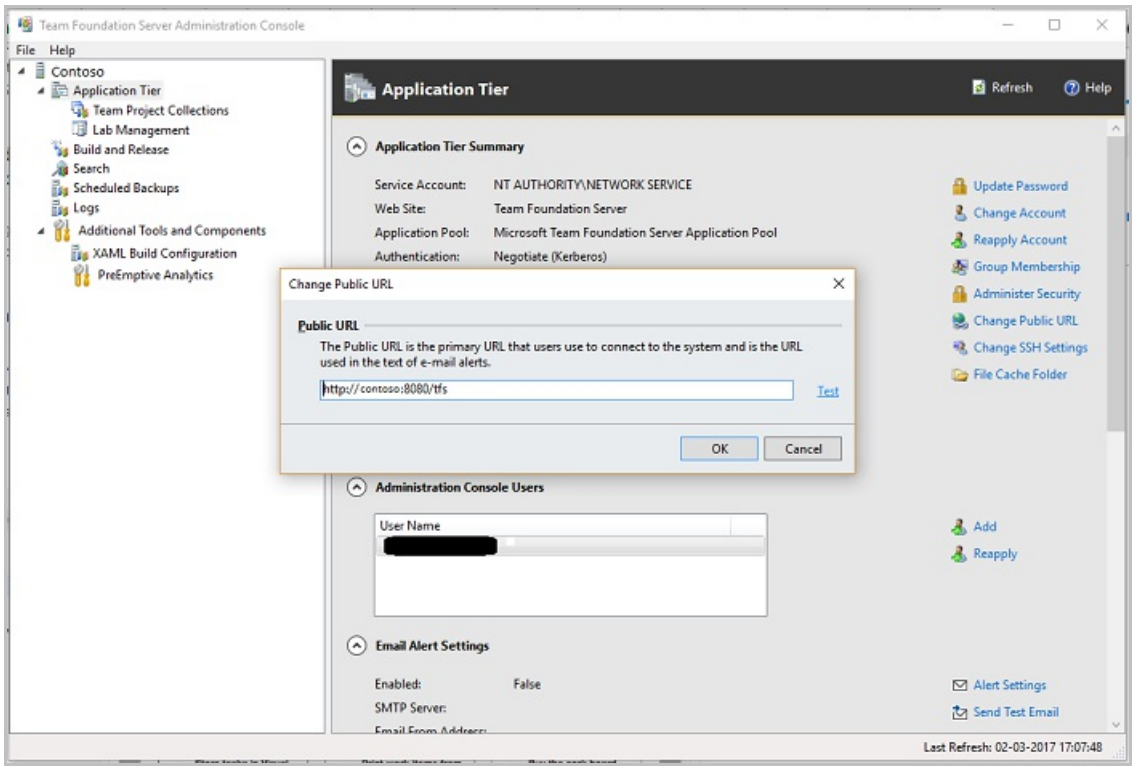
- **Issue:**

  Extension acquisition from Visual Studio Marketplace will fail.

  Already acquired extensions are likely to not function as expected.

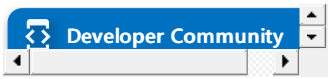
- **Workaround:**

  This is fixed in TFS 2017 Update 2 and we recommend upgrading. If you need this to work on Update 1, set the 'Public URL' in the TFS Server Administrator Console such that the URL is reachable from another system within your corporate environment *(Figure 46)*.



(Figure 46) Code menu

See customer-reported issues reported for Team Foundation Server 2017.



Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2017 Release Notes

9/7/2018 • 60 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

This is not the latest version of Team Foundation Server. To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2017. Click the button to download.

Team Foundation Server 2017

To learn more about Team Foundation Server 2017, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: February 28, 2018

This update fixes potential cross site scripting (XSS) and other security vulnerabilities. See the blog post for more information. It is a full upgrade, so you can upgrade directly to TFS 2017.0.1.

Release Date: November 16, 2016

Summary of What's New in Team Foundation Server 2017

- Code Search
- Package Management
- Agile Improvements
- Dashboards and Widget Improvements
- Git Improvements
- Build Improvements
- Release Management Improvements
- Test Improvements
- Marketplace Improvements
- Administration Improvements
- Personal Access Tokens

Known Issues

- Known Issues

Details of What's New in Team Foundation Server 2017

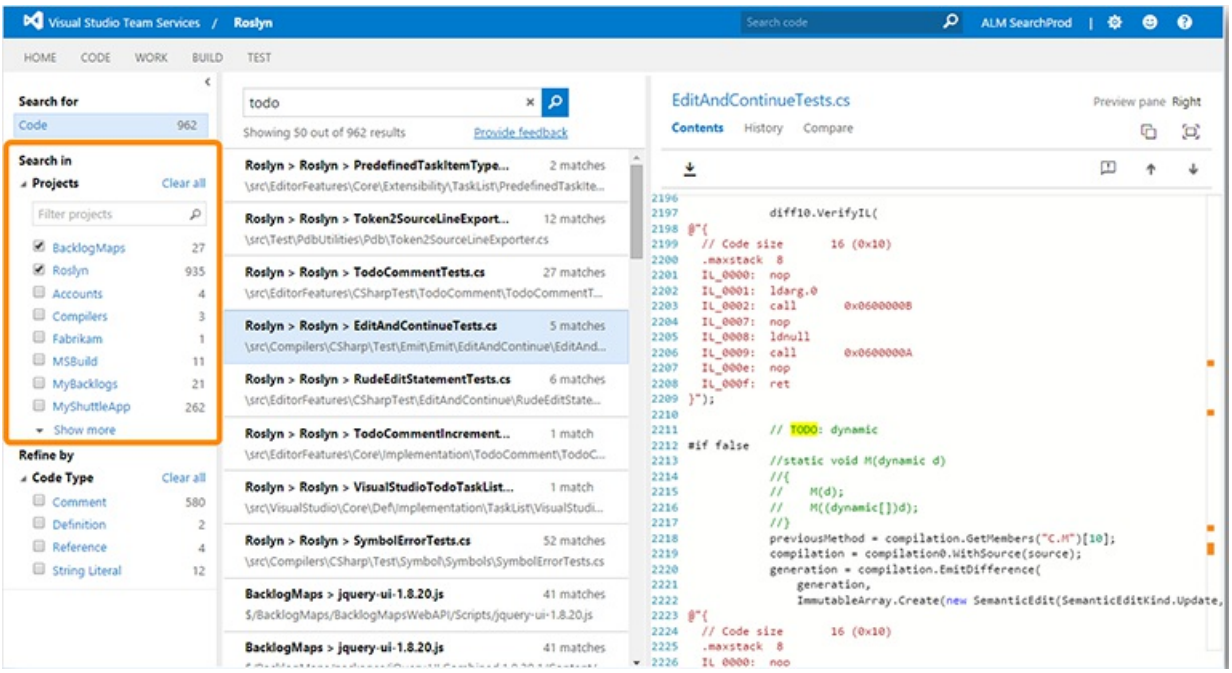
Code Search

Code Search provides fast, flexible, and accurate search across all your code. As your codebase expands and is divided across multiple projects and repositories, finding what you need becomes increasingly difficult. To maximize cross-team collaboration and code sharing, Code Search quickly and efficiently locates relevant information across all your projects.

From discovering examples of an API's implementation, browsing its definition, to searching for error text, Code Search delivers a one-stop solution for all your code exploration and troubleshooting needs *(Figure 1)*.

Code Search offers:

- Search across one or more projects
- Semantic Ranking
- Rich filtering
- Code collaboration



(Figure 1) Code Search

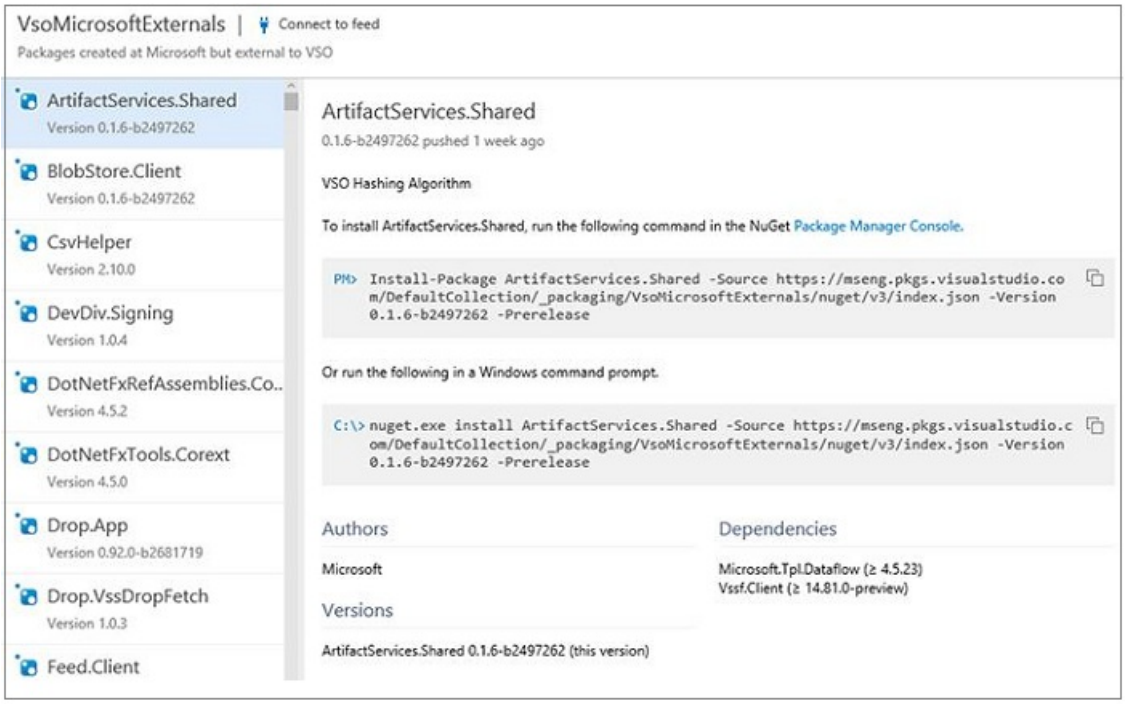
For details, see Search across all your code.

Package Management

Packages enable you to share code across your organization: you can compose a large product, develop multiple products based on a common shared framework, or create and share reusable components and libraries. Package Management *(Figure 2)* facilitates code sharing by hosting your packages, sharing them with the people you select, and making them easily accessible to Team Build and Release Management.

Package Management eliminates the need to host a separate NuGet server or file share by hosting NuGet packages directly in your Team Foundation Server. It has best-in-class support for NuGet 3.x as well as support for NuGet 2.x legacy clients. It works seamlessly with your existing TFS infrastructure, teams, and permissions, so there is no need to deal with synchronizing identities, managing groups in multiple places, etc. It also integrates easily with Team Build so you can create and use packages in continuous integration workflows.

For more details, see the Package Management overview.



(Figure 2) Package Management

Agile Improvements

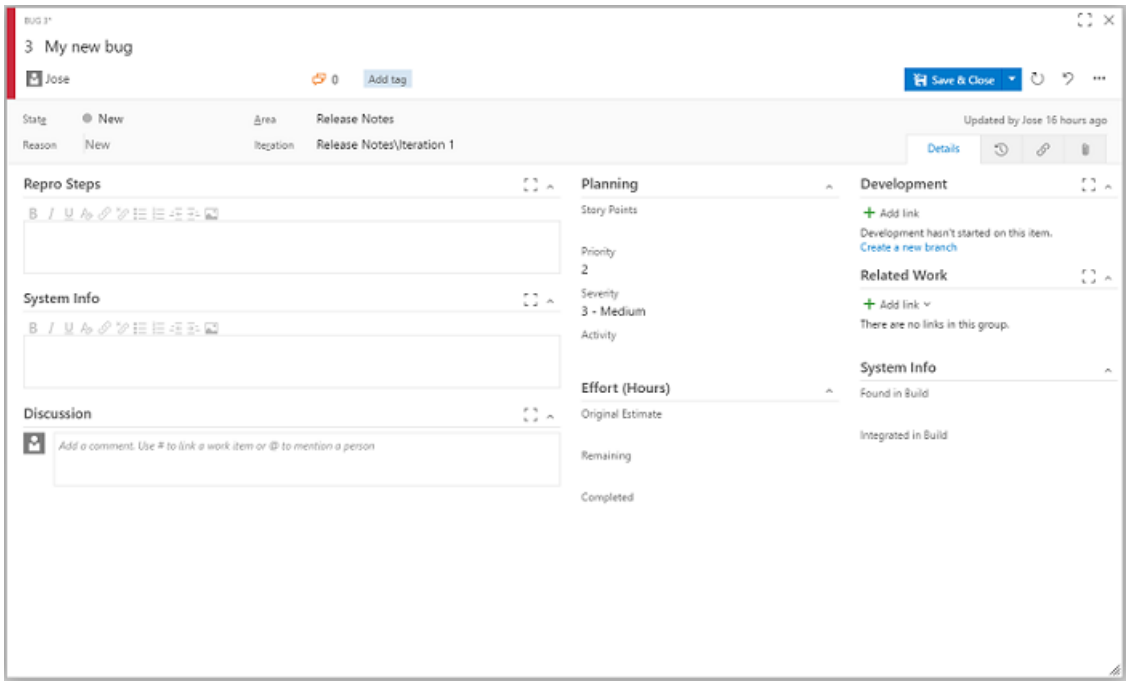
In Team Foundation Server 2017, we have added new features and functionality to work items and Kanban boards.

New work item form

The new work item *(Figure 3)* form has a new look and feel. It also adds some great new features:

- A rich work item discussion experience.
- Drag and drop support for attachments.
- Improved history experience (History & auditing).
- Improved code and build integration.
- State coloring.
- Responsive design.

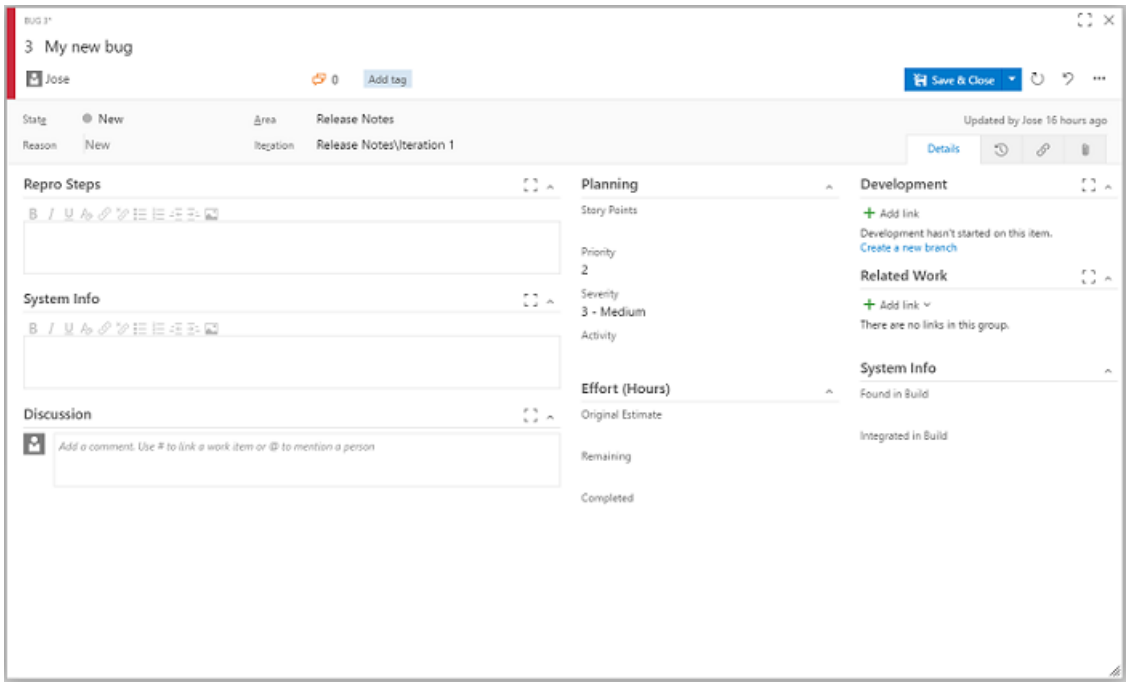
> **NOTE**
> The new work item form is the default for new collections only. If you're migrating an existing collection you will have to enable the new work item form from the admin settings. For more information, see Manage roll out of the new web form.



(Figure 3) New WIT Form

Follow a work item

You can now setup an alert for tracking changes to a single work item just by clicking on the new "Follow" button *(Figure 4)* in the form. When you follow a work item, you will be notified any time the work item changes – including field updates, links, attachments, and comments.



(Figure 4) New WIT Form

For details, see Follow a work item.

Kanban board live updates

Your Kanban board is now live!

Have you been hitting F5 to figure out what is going on throughout the day with your Kanban board? Try the icon in the screenshot below *(Figure 5)*.

(Figure 5) Kanban live updates

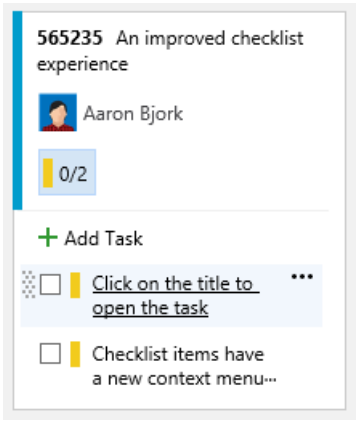
When anyone in your team creates, updates, or deletes a work item on the board, you will receive live updates on your board immediately. Also, if the administrator makes board or team level updates such as adding a new column or enabling bugs on backlog, you are notified to refresh the board to update your board layout. All you need to do is enable the tower icon on your Kanban board and start collaborating with your team.

For more information, see Kanban basics.

Checklist improvements

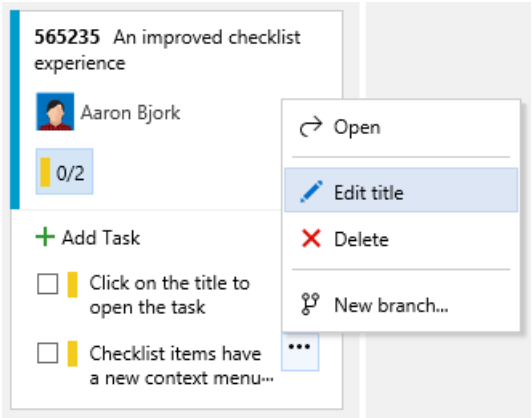
We have made several improvements to how Checklists work.

Checklists titles now appear as hyperlinks *(Figure 6)*. You can click on the title to open the work item form.



(Figure 6) Checklist hyperlinks

Checklists now also support context menus that allow you to open, edit, or delete checklist items *(Figure 7)*.

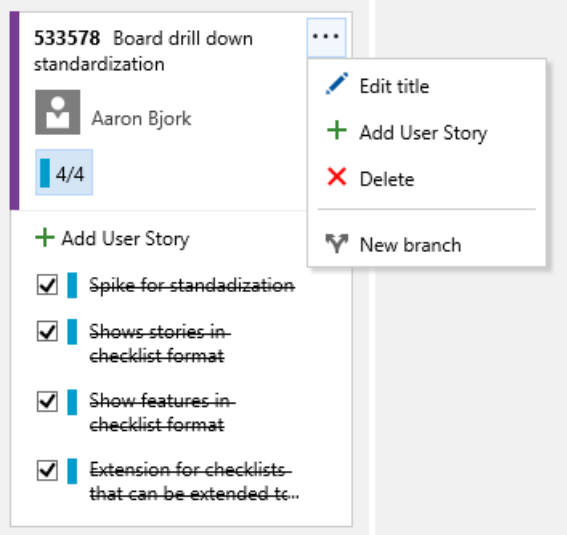


(Figure 7) Checklist context menu

For details, see Add task checklists.

Epic and Feature Board Drill-down

You now have the ability to drill down on your Epic and Feature boards *(Figure 8)*. The checklist format lets you easily mark work as completed, and provides a handy bird's eye view of the completed versus outstanding work.

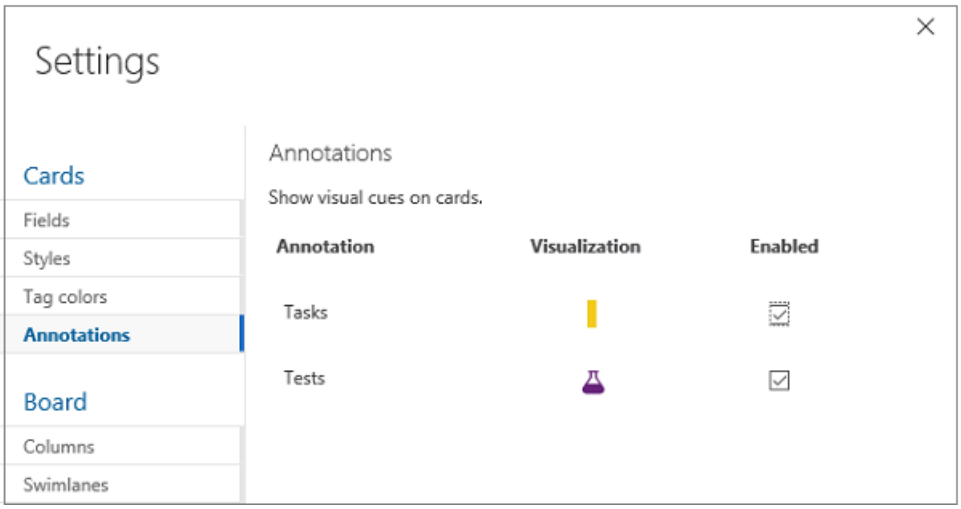


(Figure 8) Epic Feature drilldown

For more information, see Kanban features and epics.

Turning board annotations on/off

We are giving you more control of the additional information that shows on the cards on your boards. You can now select annotations that you want to view on your Kanban cards *(Figure 9)*. Simply unselect an annotation and it disappears from the cards on your Kanban board. The first two annotations to show up here are child work items (tasks in this example) and the Test annotation.



(Figure 9) Turn on/off board annotations

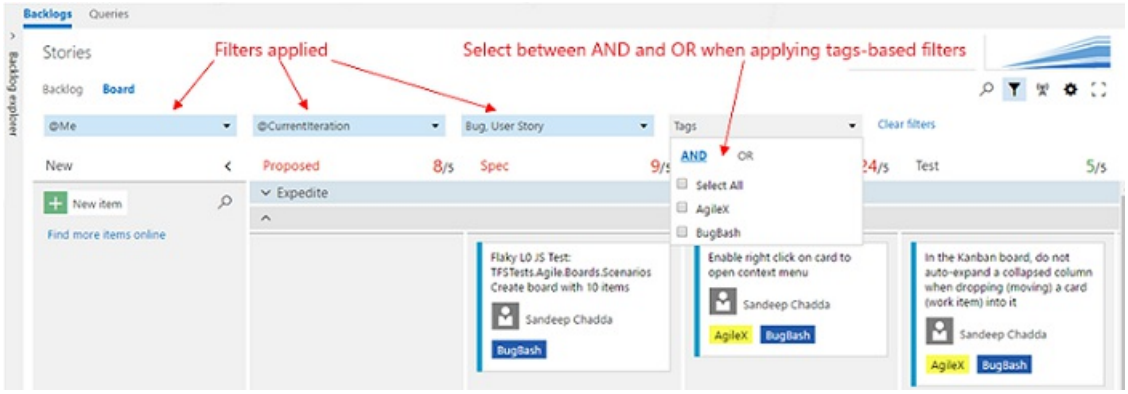
For more information, see Customize Cards.

Clear formatting command

We have added a new command to all rich text controls on work items that lets you clear all formatting from selected text. If you're like most users, you've probably been burned in the past by copying and pasting formatted text into this field that you cannot undo (or clear). Now you can simply highlight any text, select the Clear Formatting toolbar button (or press CTRL+Spacebar), and you will see the text return to its default format.

Filtering in Kanban board

Personalize your Kanban boards by setting filters on users, iterations, work item types, and tags *(Figure 10)*. These filters persist so that you can view your personalized board, even when you connect from multiple devices.



(Figure 10) Filtering in Kanban

Team members can also filter their boards to view progress accruing to a specific parent work item. For example, a user can view user stories that are linked to a feature, or view work across two or more features that roll up to an epic. This feature, much like Checklists, is one more step in our effort to bring visibility through to the different backlog levels.

For details, see Filter Kanban board.

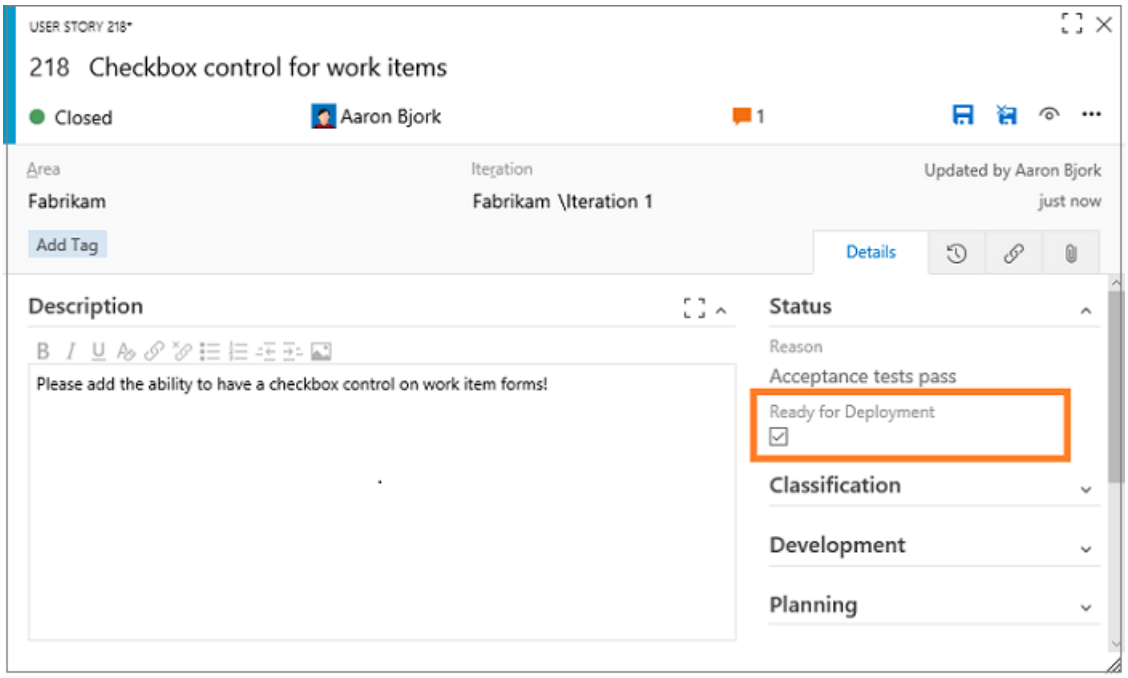
Default iteration path for new work items

When you create a new work item from the Queries tab or from the New Work Item dashboard widget, the iteration path of that work item is always set to the current iteration. This is not what all teams want, because it means that bugs could show up on the task board immediately. With this improvement, teams can choose the default iteration path (a specific one or the current iteration) that should be used for new work items. Navigate to the administration area for your team to choose a default iteration.

For more information, see the Customize area and iteration paths page.

Checkbox control

You can now add a checkbox control to your work items *(Figure 11)*. This new field type (Boolean) has all the properties of normal fields and can be added to any type in your process. When displayed on cards or in a query result, the value is shown as True/False.

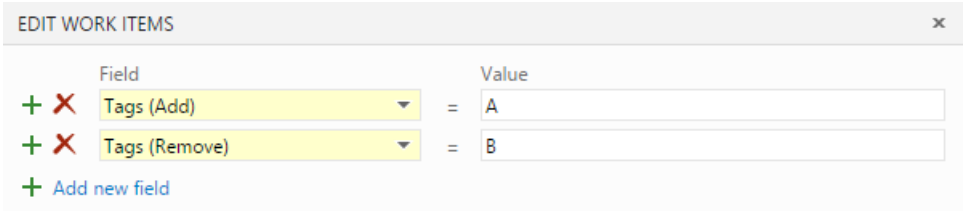


(Figure 11) Checkbox control

For details, see Customize a field.

Tags bulk editing

You can now add and remove tags from multiple work items using the bulk edit dialog *(Figure 12)*.



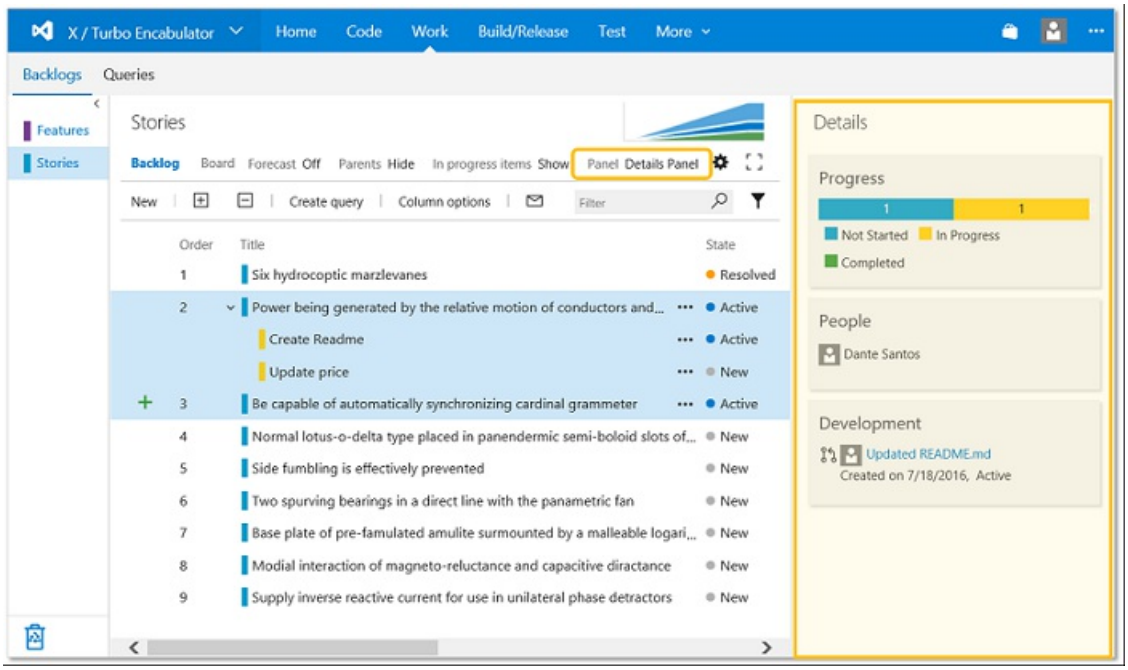
(Figure 12) Bulk edit dialog

For details, see Add tags to work items.

New extension points

We have added a new contribution point on the board and backlog pages to allow you to write extensions as a pivot tab next to Board/Backlog/Capacity tabs.

We have exposed a new extension point on the backlog. Extensions can target the pane on the right side, where mapping and work details are today *(Figure 13)*.

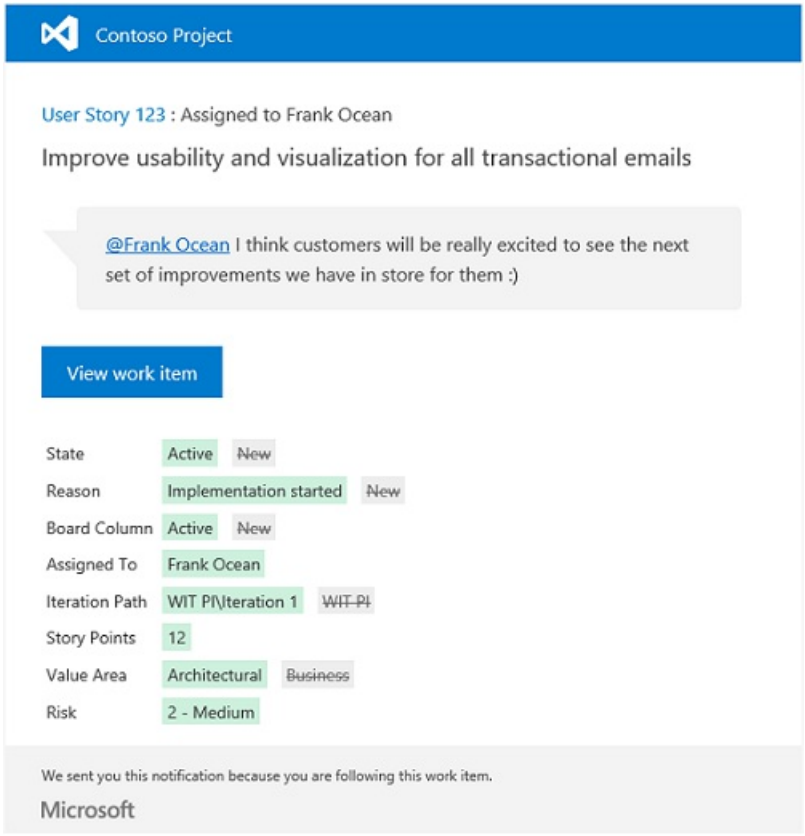


(Figure 13) Backlog extension points

For more information on extensions, see Extension Points.

Email improvements

We have significantly improved the formatting and usability of work item alerts, follows, and @mention emails sent by TFS *(Figure 14)*. Emails now include a consistent header, a clear call to action, and improved formatting to make sure the information in the mail is easier to consume and understand. Additionally, all these emails are being designed to ensure they render well on mobile devices.

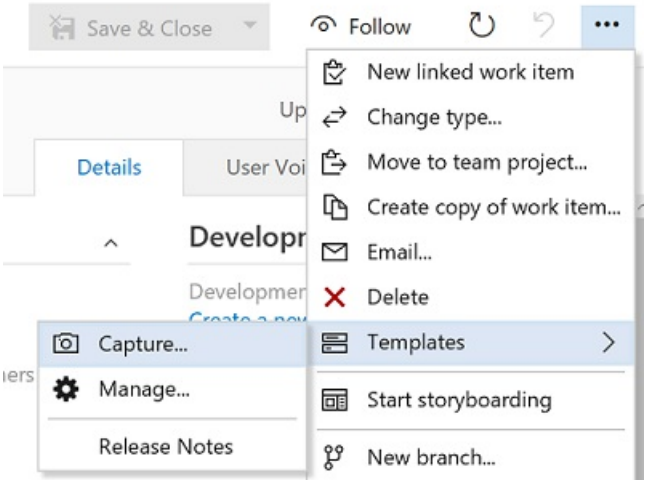


(Figure 14) Email improvements

For more information, see Work item alerts.

Work item templates

We added the ability to create rich work item templates directly into the native web experience *(Figure 15)*. This capability was previously very limited in the web, and only available in this new form through a Visual Studio power tool. Teams can now create and manage a set of templates for quickly modifying common fields.



(Figure 15) Work item templates

For details, see Work item templates.

Project server integration no longer supported

Team Foundation Server 2017 and later versions no longer support Project Server integration. As of RC2, if you upgrade a TFS database that has Project Server integration configured, you will receive the following warning:

We have detected that you have Project Server integration configured for this database. Team Foundation Server 2017 and later versions no longer support Project Server integration.

After upgrade, the Project Server integration no longer operates.

Going forward, we will be relying on Partners to provide integration solutions.

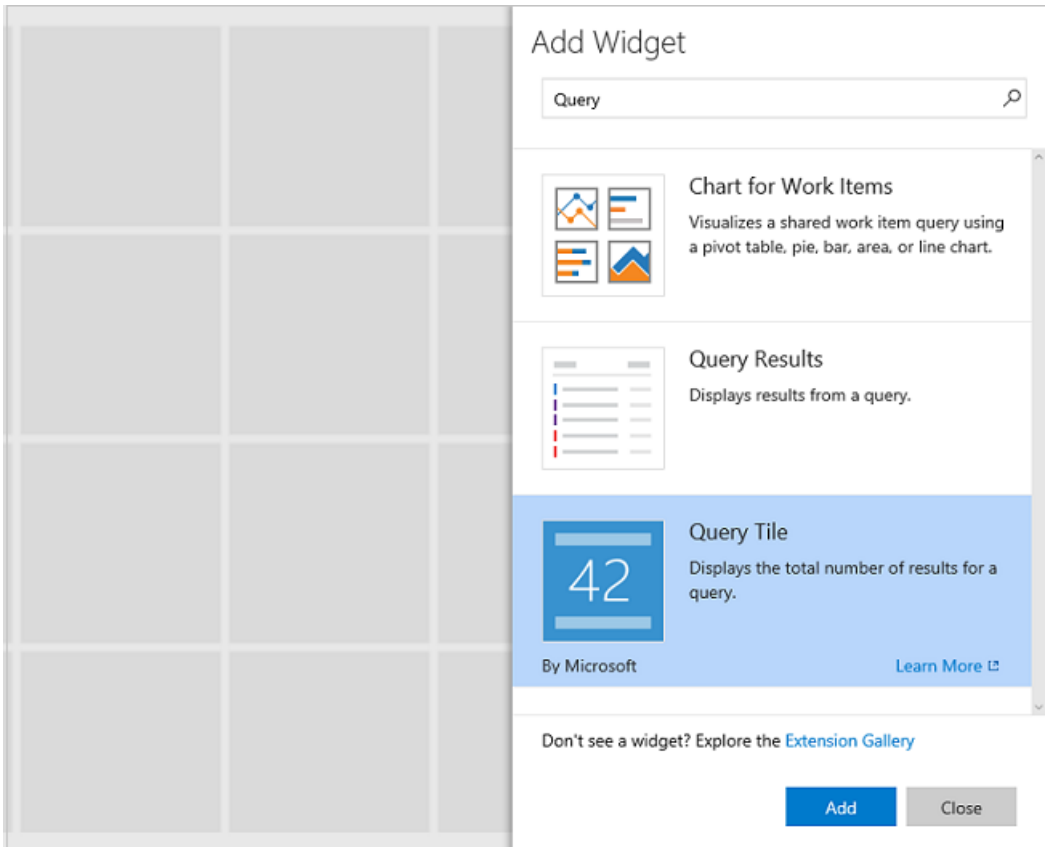
For more information on this change, please read the following topic: Synchronize TFS with Project Server.

Dashboards and Widgets Improvements

Team Foundation Server 2017 has made improvements on multiple widgets, such as the Query Tile and Pull Request widgets.

Redesigned widget catalog

We have redesigned our widget catalog to accommodate the growing set of widgets and deliver a better overall experience *(Figure 16)*. The new design includes an improved search experience and has been restyled to match the design of our widget configuration panels.

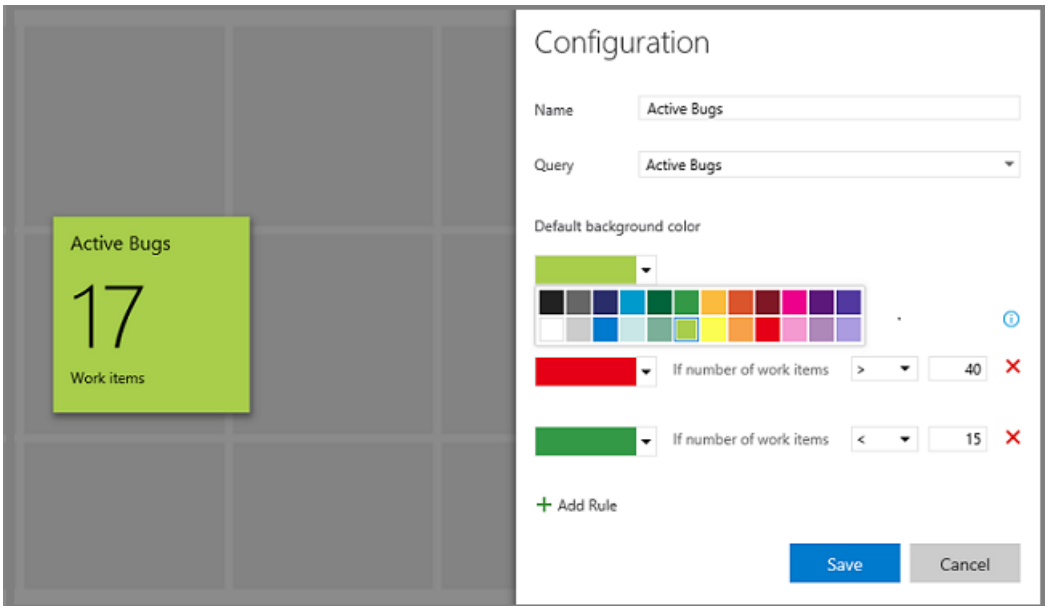


(Figure 16) Widget catalog

For more details, see Widget Catalog.

Widget updates

The Query Tile widget now supports up to 10 conditional rules and has selectable colors *(Figure 17)*. This is extremely handy when you want to use these tiles as key performance indicators (KPI) to identify health and/or action that may be needed.



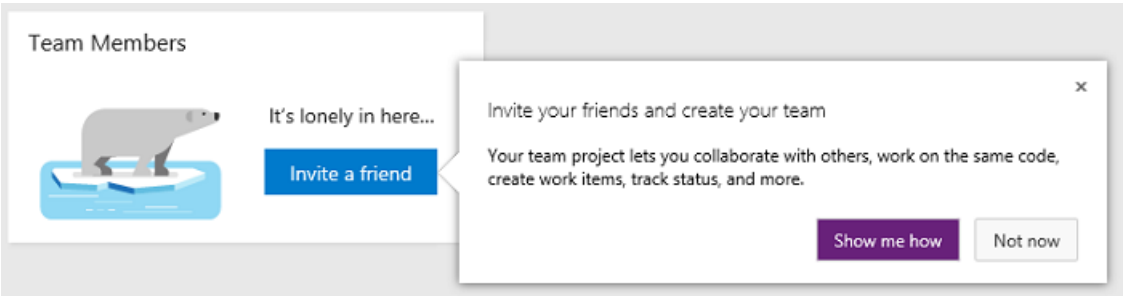
(Figure 17) Dashboard updates

The Pull Request widget now supports multiple sizes, allowing users to control the height of the widget. We're working on making most of the widgets we ship resizable, so look for more here.

The New Work Item widget now allows you to select the default work item type, instead of forcing you to select the most common type you're creating over and over from the drop-down list.

We have made the WIT chart widgets resizable. This allows users to see an expanded view of any WIT chart on the dashboard regardless of its original size.

We have updated the Team Members widget to make it easier to add somebody to your team *(Figure 18)*.

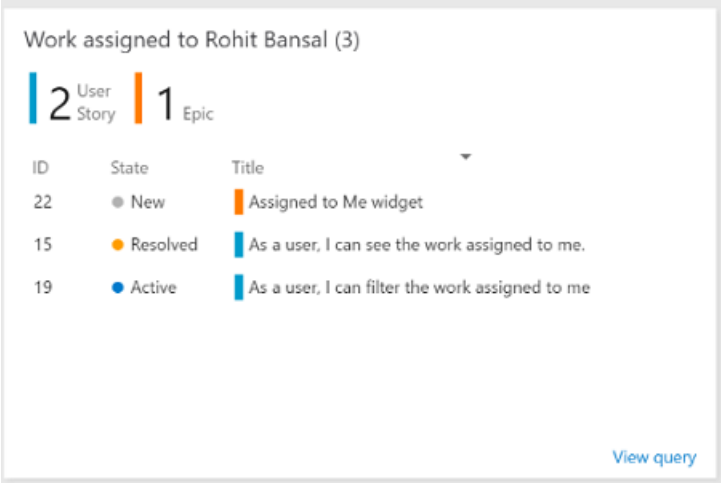


(Figure 18) Widget Update

Teams can now configure the size of the dashboard's Query Results widget, allowing it to display more results.

The Sprint Overview widget has been redesigned making it easier for teams to see if they are on track.

The Assigned to Me widget helps users manage the work assigned to them without leaving the dashboard context *(Figure 19)*. By providing a widget dedicated to this purpose, team admins can add this functionality to their dashboards with 16 fewer clicks, no context switches and no typing required. Users can now view, sort, filter, and manage the work assigned to them within the widget context.



(Figure 19) Assigned to me

Dashboards REST APIs

You can now use REST APIs to programmatically add, delete, and get information on a dashboard. The APIs also let you add, remove, update, replace, and get information on a widget or a list of widgets on a dashboard. The documentation is available on Visual Studio online docs.

Permissible dashboards

Non-admin users can now create and manage team dashboards. Team admins can restrict non-admin permissions through the dashboard manager.

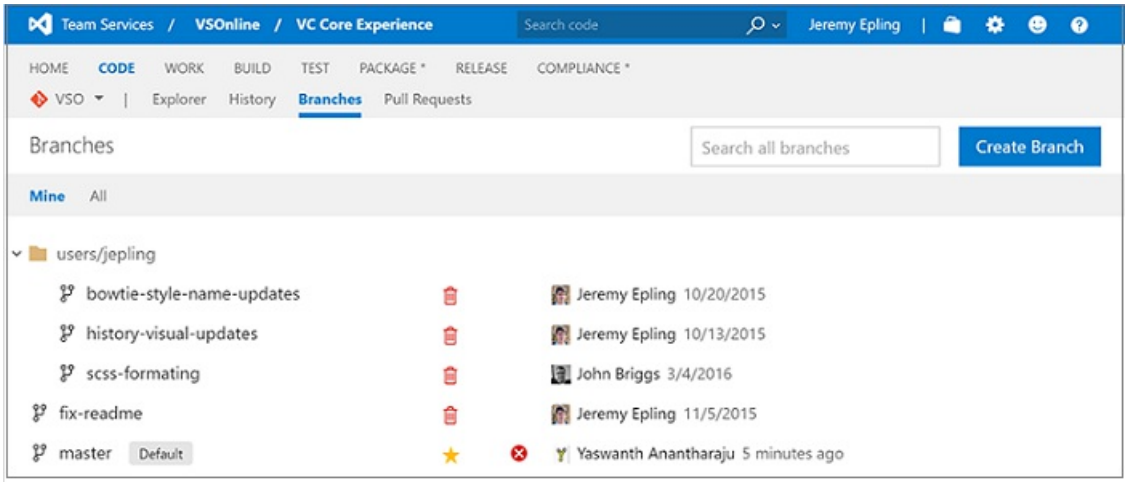
For more information, see Dashboards.

Git Improvements

Some major changes have been made in Git for Team Foundation Server 2017. Included are a redesign of the Branches page and a new option to "squash merge".

Redesigned Branches page

The Branches page has been completely redesigned. It has a "mine" pivot that shows the branches you created, pushed to, or favorited *(Figure 20)*. Each branch shows its build and pull requests status, as well as other commands like Delete. If there is a slash in a branch name, like "features/jeremy/fix-bug", it's shown as a tree, so it's easy to browse through a large list of branches. If you know the name of your branch, you can search to find the one you want quickly.



(Figure 20) Redesigned branches page

For more details on branches, see Manage branches.

New pull request experience

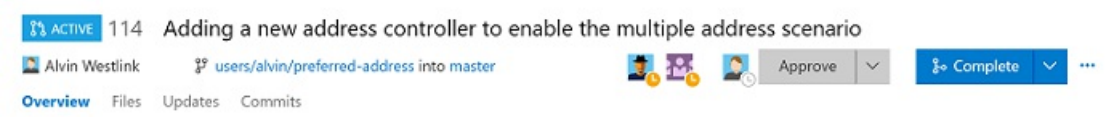
The pull request experience has some major updates this release, bringing some really powerful diff capabilities, a

new commenting experience, and an entirely refreshed UI.

For more details, see Review code with Pull Requests.

Redesigned UI

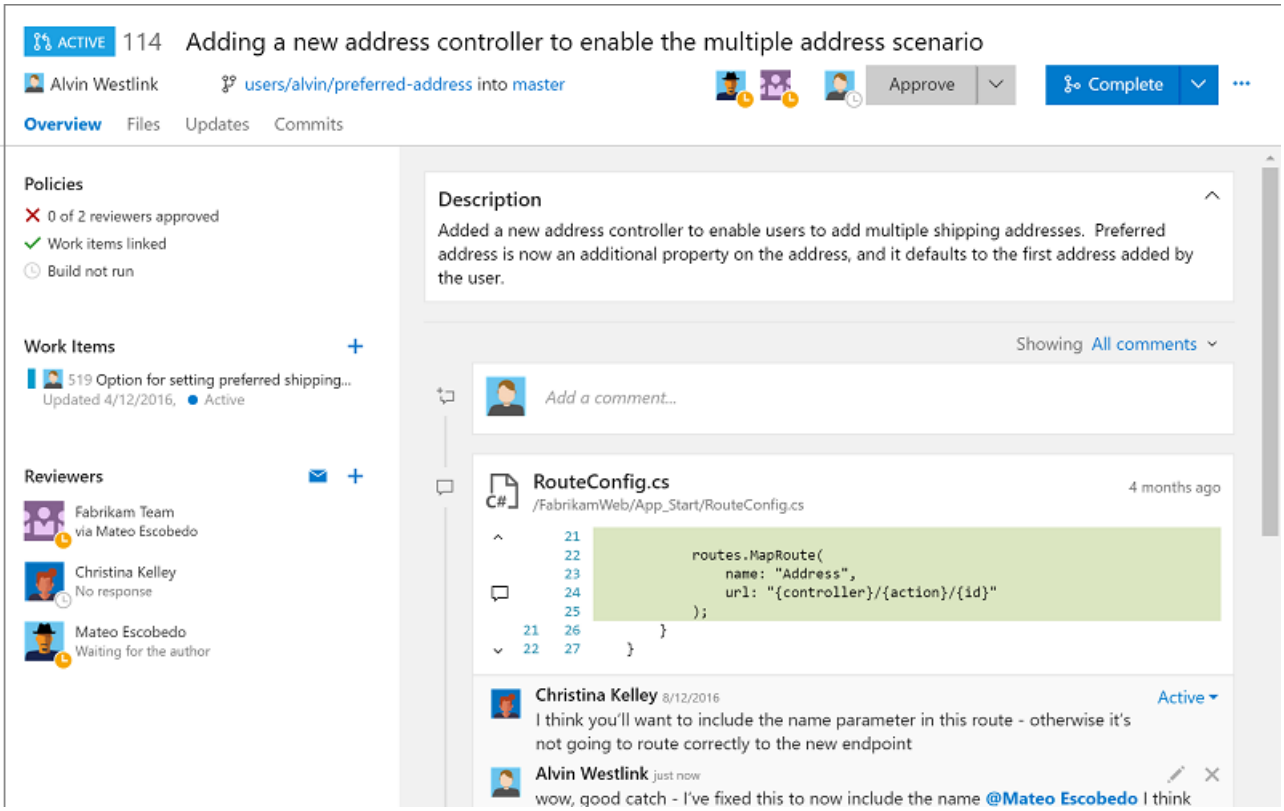
When opening a pull request, the new look and feel is evident immediately *(Figure 21)*. We have reorganized the header to summarize all the critical state and actions, making them accessible from every view in the experience.



(Figure 21) Pull request header

Overview

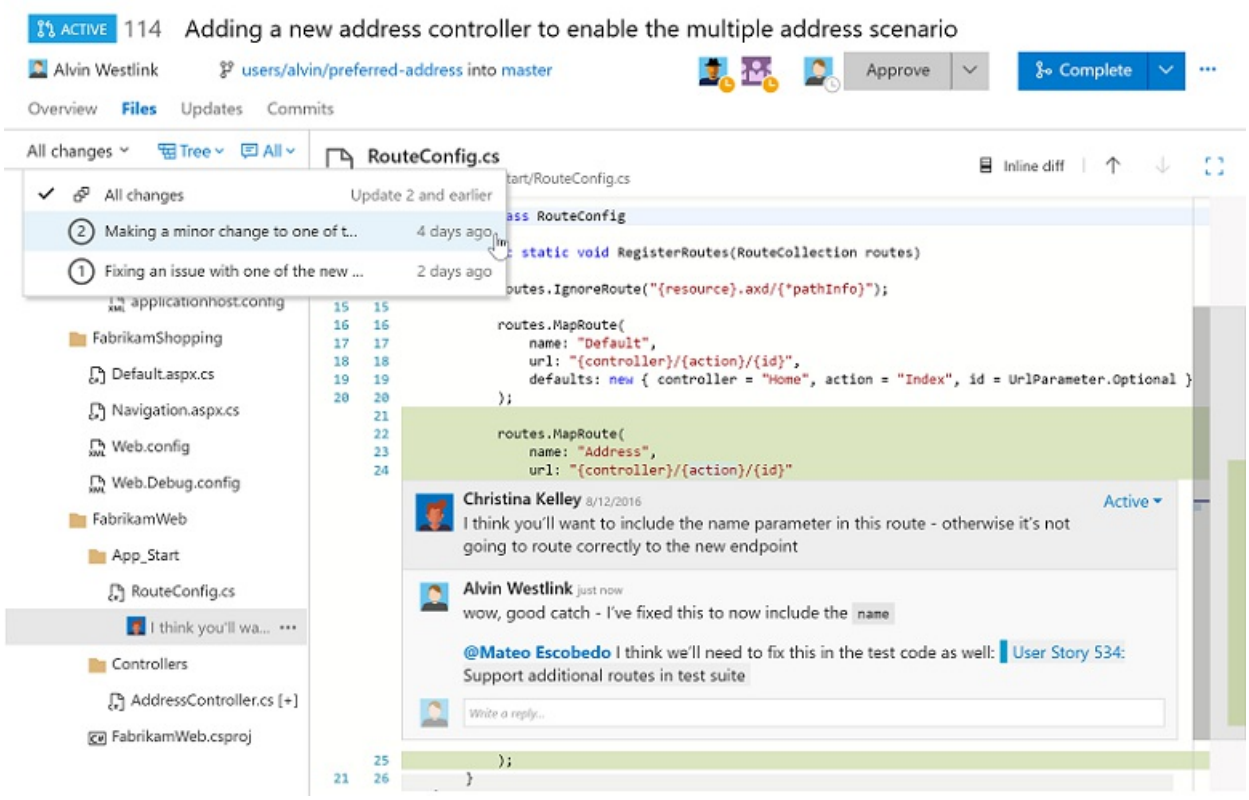
The Overview now highlights the PR Description and makes it easier than ever to give feedback *(Figure 22)*. Events and comments are shown with the newest items on top to help reviewers see the latest changes and comments front and center. Policies, work items, and reviewers are all provided in detail and reorganized to be more clear and concise.



(Figure 22) Pull request overview

Files

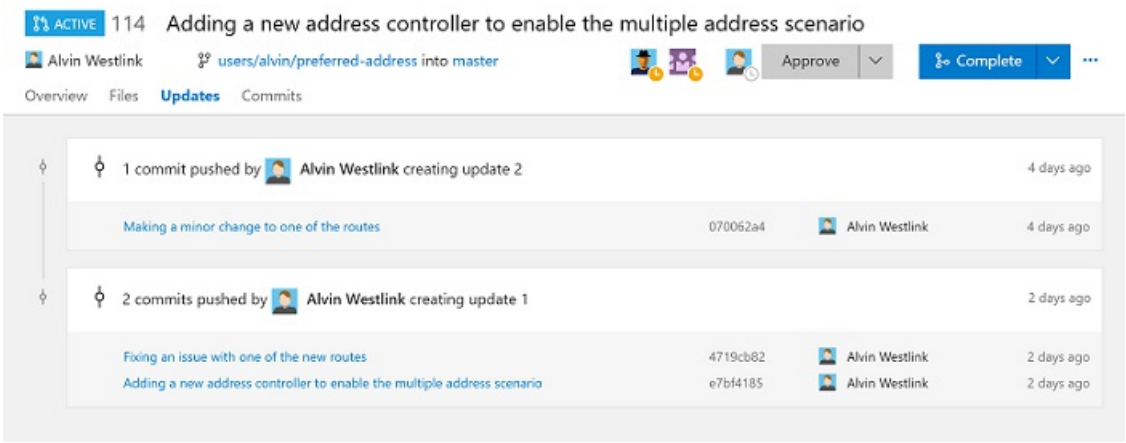
The biggest new feature in this release is the ability to see past updates made to a pull request *(Figure 23)*. In previous previews, we released the ability to properly track comments as a PR is updated with changes. However, it's not always easy to see what's between updates. In the Files view, you can now see exactly what changed each time new code is pushed to your PR. This is very useful if you've given feedback on some code and want to see exactly how it changed, isolated from all the other changes in the review.



(Figure 23) Pull request files

Updates

The new Updates view shows how the PR is changing over time *(Figure 24)*. Where the Files view shows how the files have changed over time, the Updates view shows the commits added in each update. If a force push ever happens, the Updates view will continue to show the past updates as they occurred in history.



(Figure 24) Pull request updates

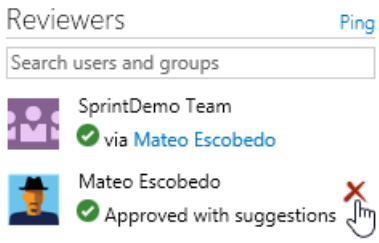
Comments, now with markdown and emoji

Use the full power of markdown in all your discussions, including formatting, code with syntax highlighting, links, images, and emoji *(Figure 25)*. The commenting controls also have a more user friendly editing experience allowing multiple comments to be edited (and then saved) at one time.

(Figure 25) Pull request comments

Add and remove reviewers in pull requests

It is now easier to add and remove reviewers from your pull requests. To add a reviewer or group to your pull request, simply enter their name into the search box in the Reviewers section. To remove a reviewer, hover over their tile in the Reviewers section and click the X to remove them *(Figure 26)*.



(Figure 26) Add reviewers in pull requests

Improved build and pull request traceability

The traceability between builds and pull requests has improved, making it easy to navigate from a PR to a build and back. In the build details view for a build triggered by a pull request, the source will now show a link to the pull request that queued the build. In the Build Definitions view, any build triggered by a pull request will provide a link to the pull request in the "Triggered By" column. Finally, the Build Explorer view will list pull requests in the source column.

Comment tracking for pull requests

Pull requests in VSTS have been improved to show comments left in files on the proper line, even if those files have been changed since the comments were added. Previously, comments were always shown on the line of the file where they were originally added, even if the file contents changed—in other words, a comment on line 10 would always be shown on line 10. With the latest improvements, the comments follow the code to show what the user expects—if a comment is added on line 10, and two new lines were subsequently added to the beginning of the file, the comment is shown on line 12.

Here is an example change with a comment on line 13 *(Figure 27)*:

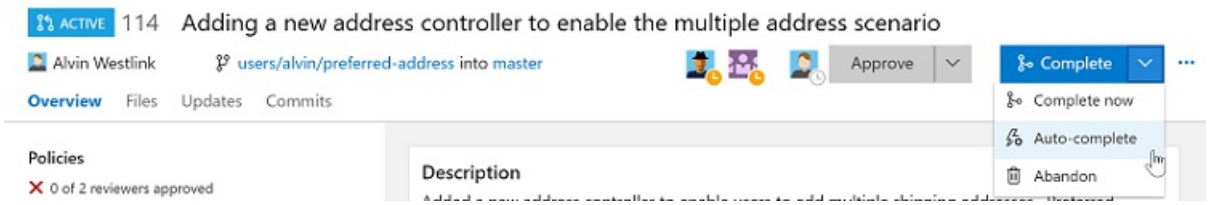
(Figure 27) Comment tracking

Even after the code has changed to shift the line with the original comment from 13 to 14, the comment is appearing in the expected place on line 14 *(Figure 28)*.

(Figure 28) Comment tracking with change

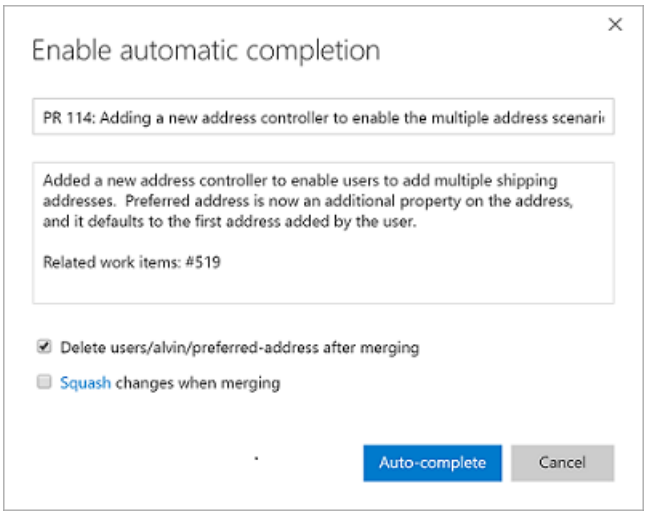
Auto-complete pull requests waiting on policies

Teams that are using branch policies https://docs.microsoft.com/vsts/git/branch-policies?view=vsts to protect their branches will want to check out the auto-complete action. Many times, the author of a pull request is ready to merge their PR, but they are waiting on a build to finish before they can click Complete. Other times, the build is passing, but there is one reviewer that has not given the final approval. In these cases, the auto-complete action lets the author set the PR to automatically complete as soon as the policies are all approved *(Figure 29)*.



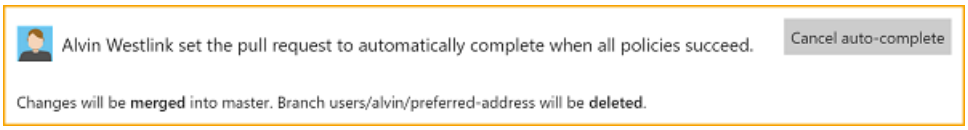
(Figure 29) Auto-complete

Just like the manual complete action, the author has a chance to customize the message of the merge commit and select the appropriate merge options *(Figure 30)*.



(Figure 30) Autodialog

Once auto-complete has been set, the PR will display a banner that confirms that the auto-complete is set and waiting for policies to complete *(Figure 31)*.

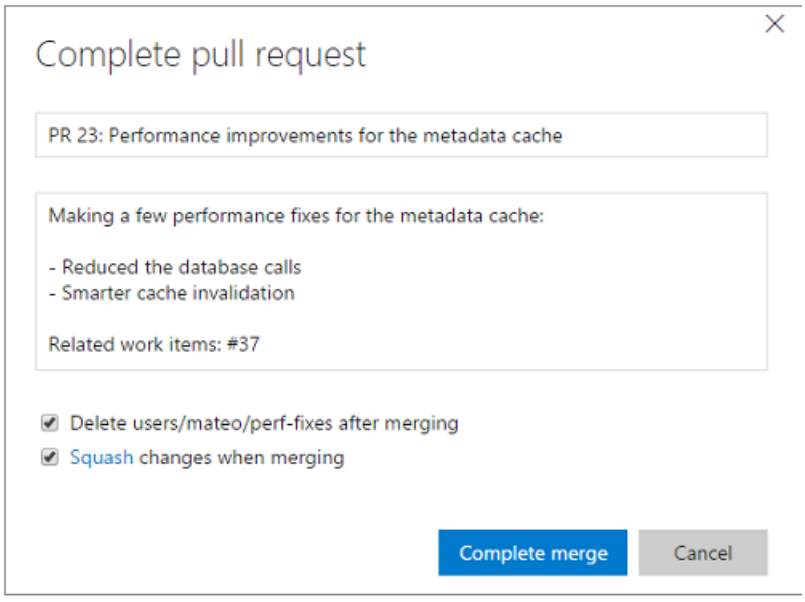


(Figure 31) Auto-complete confirmation

When all the policies are met (e.g., the build completes, or that final approval is granted), the PR is merged using the options and comments specified. As expected, if there is a build failure or the reviewer does not approve, the PR remains active until the policies are passing.

Squash merge pull requests

When completing a pull request, you now have the option to squash merge *(Figure 32)*. This new option produces a single commit containing the changes from the topic branch that is applied to the target branch. The most notable difference between a regular merge and a squash merge is that the squash merge commit will only have one parent commit. This will mean a simpler history graph, as any intermediate commits made to the topic branch will not be reachable in the resulting commit graph.

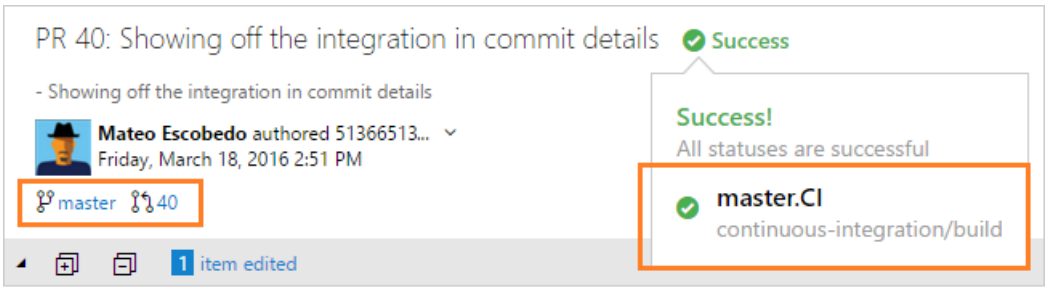


(Figure 32) Squash merge pull request

You can find more information at Squash merge pull requests.

Commit traceability

Build status (success or failure) is now clearly visible in the Code Explorer and Commit Details views *(Figure 33)*. More details are just a click away, so you will always know if the changes in the commit passed the build or not. You can also customize which builds post status in the repository options for the build definition. Additionally, the latest changes to the Commit Details view provide deeper insights about your changes. If you're using pull requests to merge your changes, you will see the link to the pull request that introduced the changes into the master branch (or in the case of a merge commit, the PR that created it). When your changes have reached master, the branch link will appear to confirm that the changes have been included.



(Figure 33) Commit Traceability

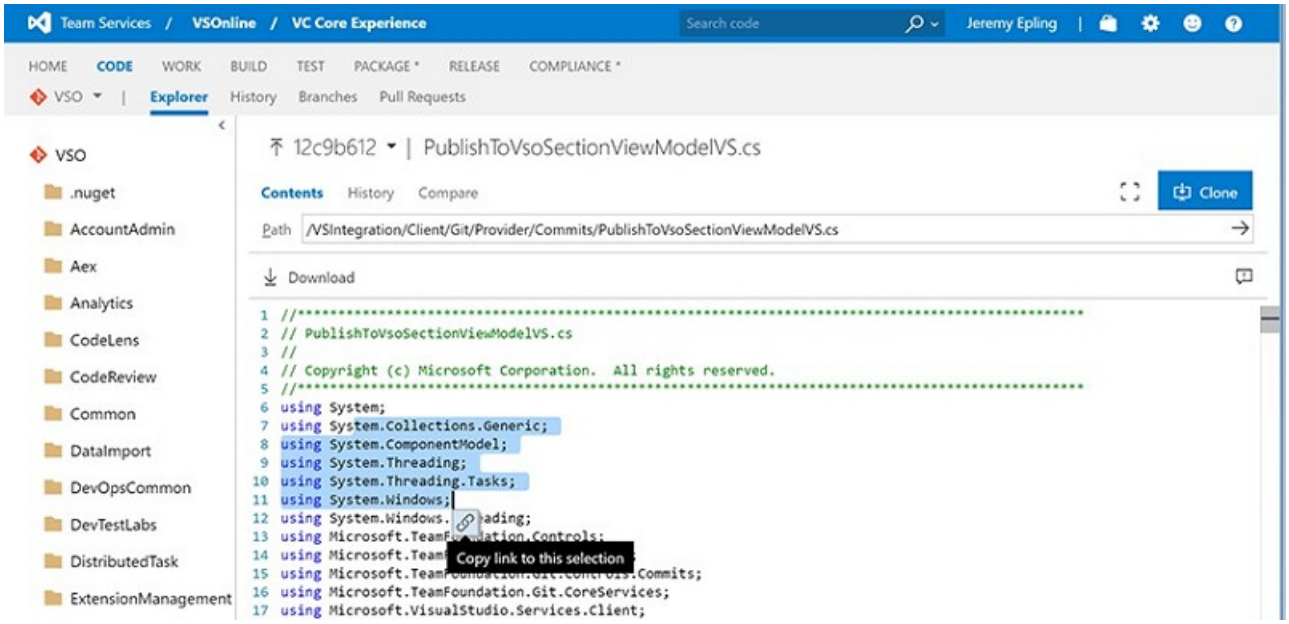
View Git LFS files in the Web

If you're already working with large files in Git (audio, video, datasets, etc.), then you know that Git Large File Storage (LFS) replaces these files with pointers inside Git, while storing the file contents in a remote server. This makes it possible to view the full contents of these large files by simply clicking the file in your repo.

For more information, see Manage large files with Git.

Create and send links to specific sections of code

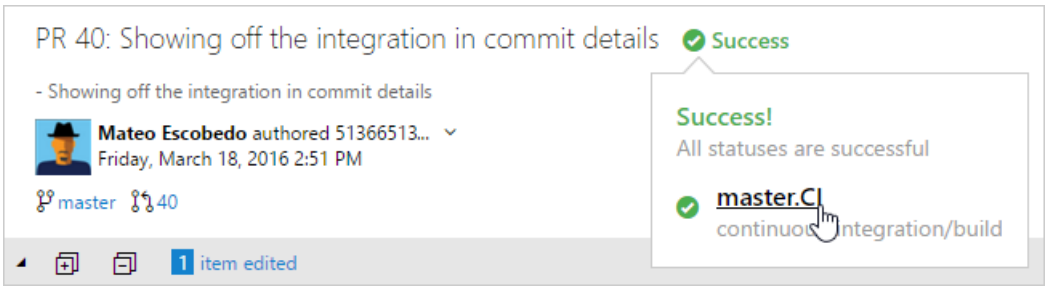
Share code references easily with code links *(Figure 34)*. Just select text in a file and click the Link icon. It will copy a link to the selected code. When someone views that link, the code you highlighted will have a gold background. It even works for partial line selections.



(Figure 34) Send links to code

Status API

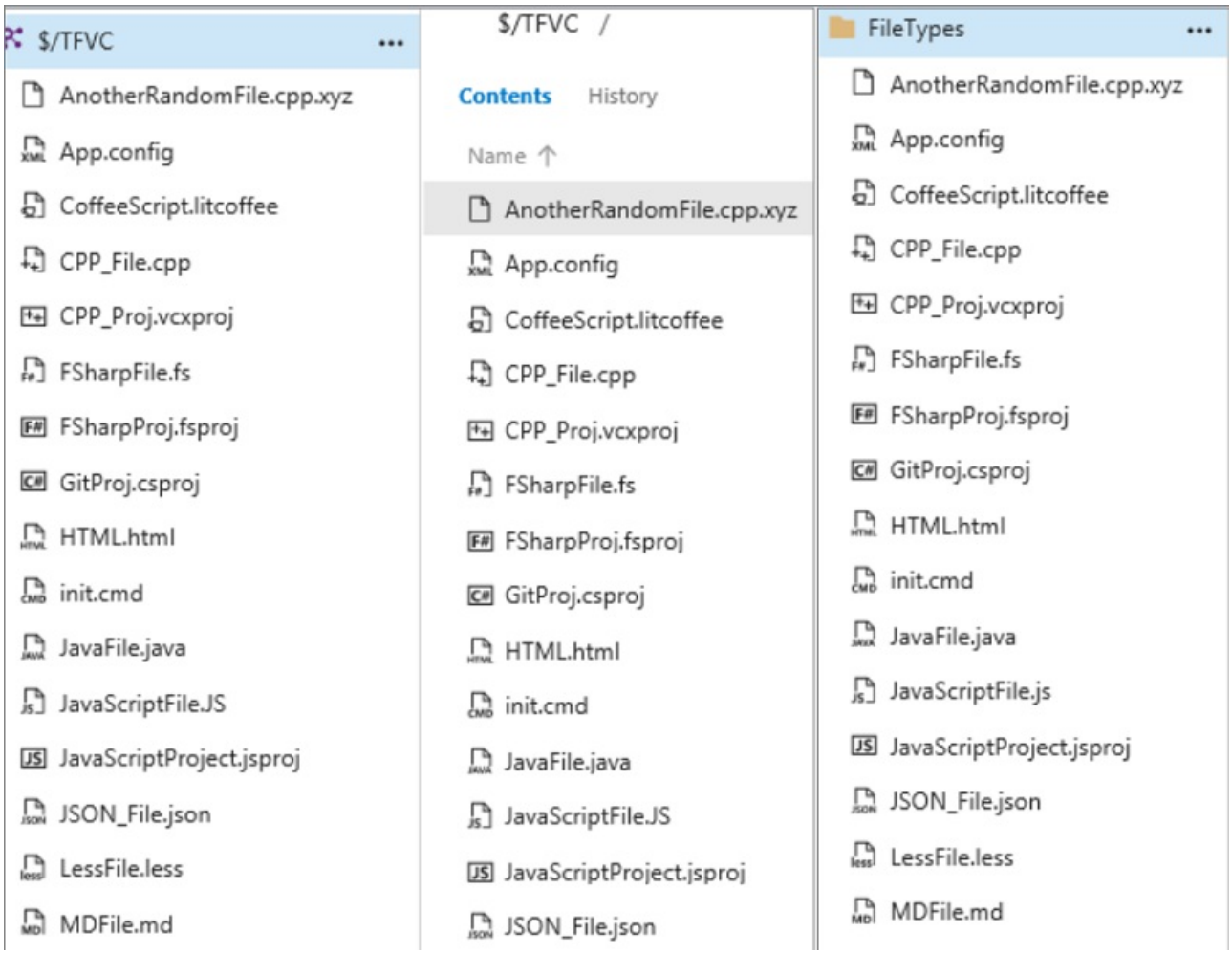
Success or failure of the build is now clearly visible in the code explorer and commit details views *(Figure 35)*. More details are just a click away, so you always know if the changes in the commit passed the build or not. You can also customize which builds post build status in the repository options for the build definition.



(Figure 35) Status API

File type icons

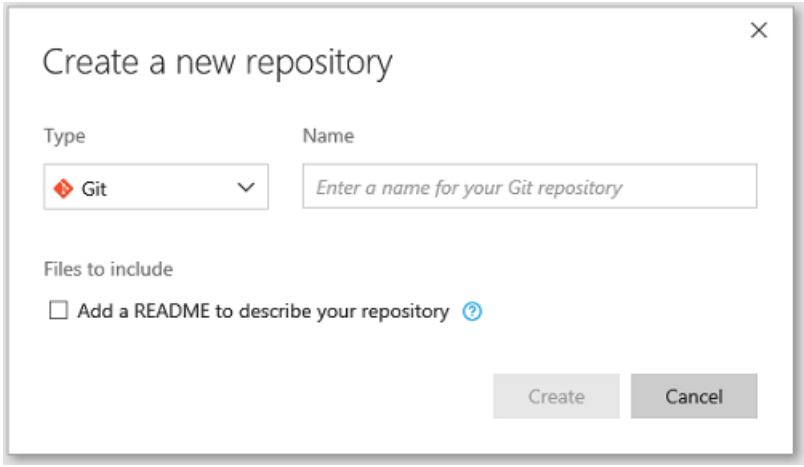
You will see new file icons matching the extension of the file in the explorer, pull requests, commit details, shelveset, changeset or any other view that shows a list of files *(Figure 36)*.



(Figure 36) File type examples

Add a ReadMe during repo creation

The new Git repository creation has been improved by providing users the ability to add a ReadMe file *(Figure 37)*. Adding a ReadMe to the repository not only helps others understand the purpose of the codebase, but also allows you to immediately clone the repository.



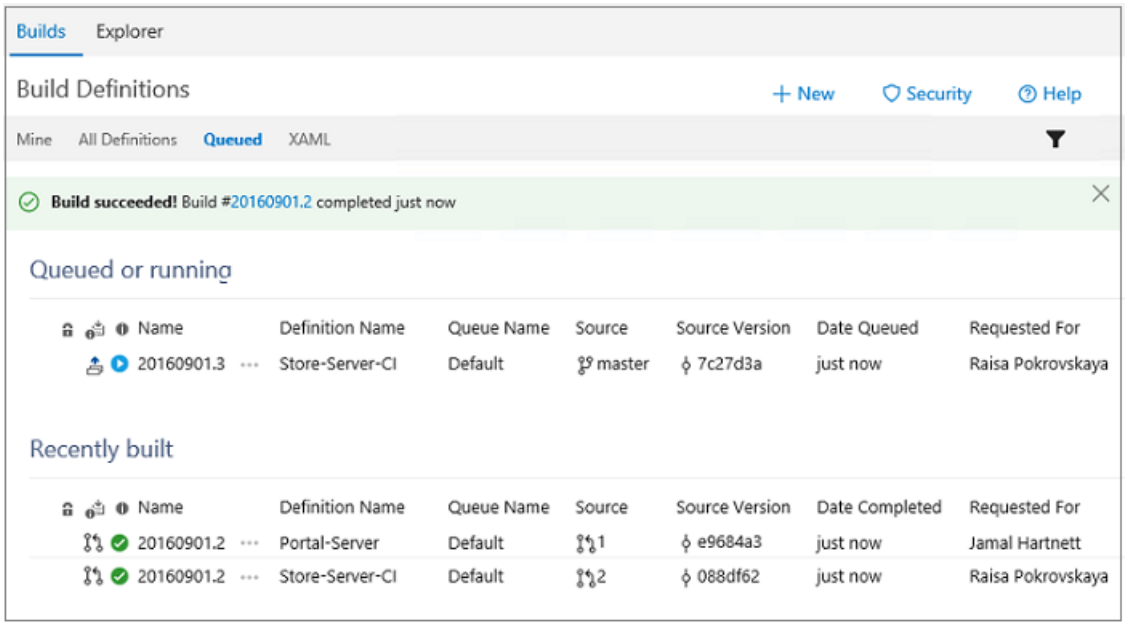
(Figure 37) Add a ReadMe file

Build Improvements

In this release, we have increased the size of the logs, added Java build templates, and improvements to our Xamarin support to name a few changes.

Redesigned build queue tab

We have implemented a new design for the Queued builds page that shows a longer list of queued and running builds, and in a more intuitive fashion *(Figure 38)*.



(Figure 38) Build queue tab

For more information, see Administer your build system.

Enable build result extensions to specify order and column

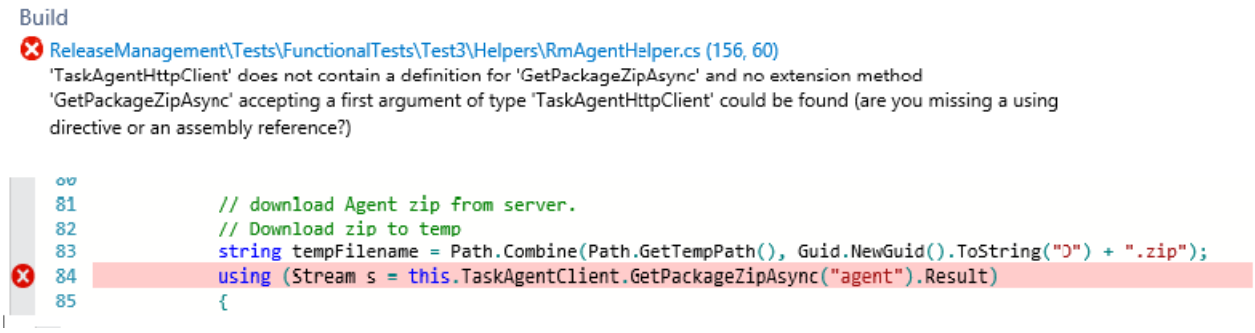
Build result section extensions can now specify which column and the order in which they appear *(Figure 39)*. The result view has two columns, and all extensions are in the first column by default. Note: All third-party extensions will appear after the build result sections we include.

```
{
  "id": "test-result-summary",
  "type": "ms.vss-build-web.build-results-section",
  "targets": [ "ms.vss-build-web.build-results-summary-tab" ],
  "properties": {
    "name": "i18n:Test Results",
    "order": 10,
    "column": 1,
    "height": "100%",
    "content": {
      "require": [ "TestManagement/Scripts/TestResults/TFS.TestManagement
      "initialize": "testResults.summary"
    }
  }
}
```

(Figure 39) Build order and column

Build to line number

Now you can jump from a build error to the line of code that caused it. Looking at the latest error on the primary build we use as a pull request policy internally, you see this *(Figure 40)*:



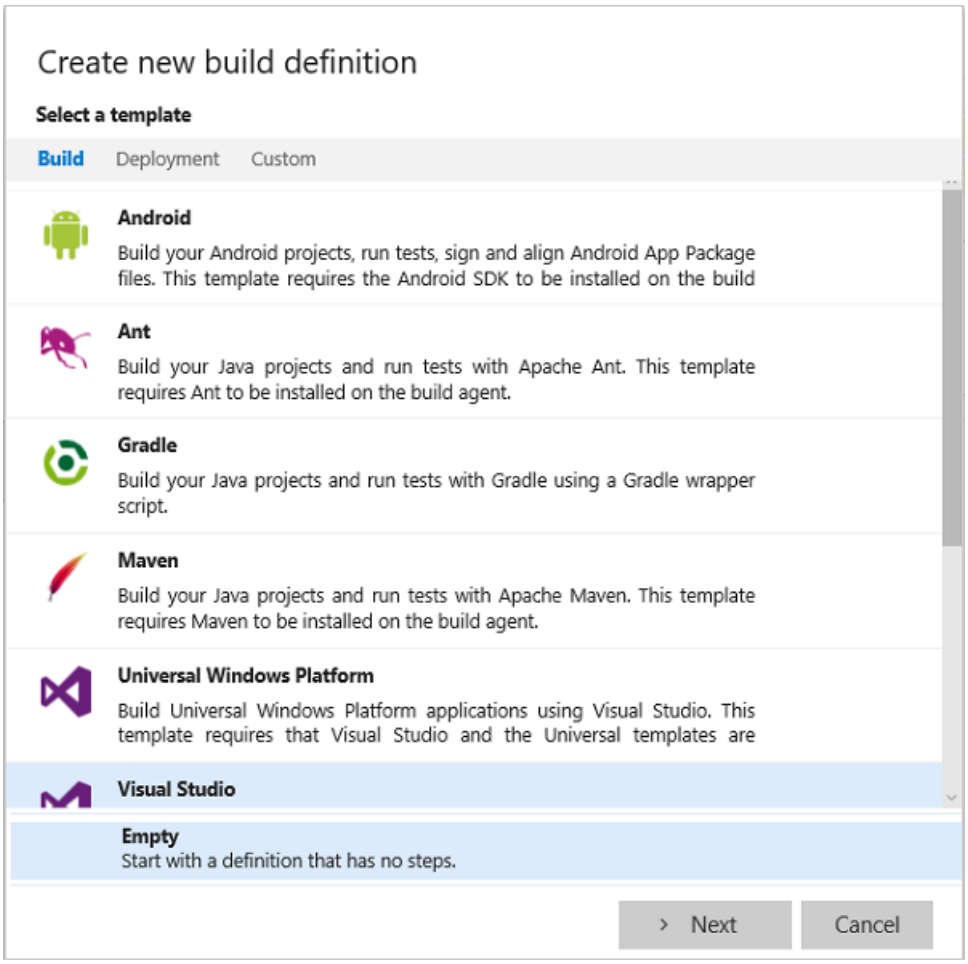
(Figure 40) Build to line number

Build log view supports much larger logs

The previous log view only supported logs up to 10,000 lines. The new viewer is based on the Monaco editor used in VS Code and will support logs up to 150,000 lines.

Java build templates

We have made it even easier for Java developers to get started with build by adding build templates for Ant, Maven, and Gradle *(Figure 41)*.



(Figure 41) Java build templates

For more information on templates, see Build steps.

Xamarin build tasks

We made some significant improvements to our Xamarin support:

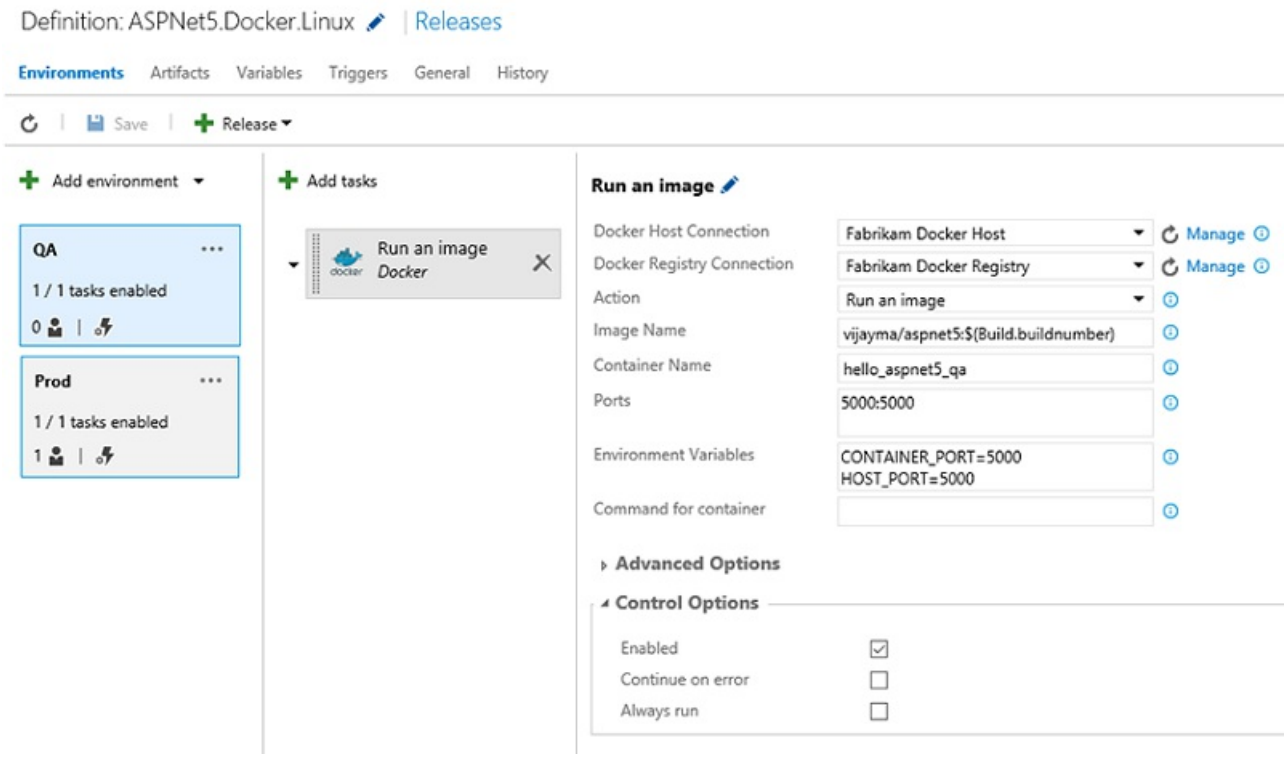
- The Xamarin.Android step now supports Mac and Linux.
- The Xamarin.iOS step now supports signing and packaging.
- Xamarin Test Cloud results can be displayed on the build summary page.
- A new Xamarin component restore step.
- The NuGet Installer step now supports Mac OS.

The Xamarin License step is no longer necessary and has been removed from the build templates. As part of this effort we are deprecating the task. All build definitions that use this task should be updated to remove it in order to prevent any disruption when the task is finally removed.

Finally, we have enhanced the Xamarin build definition templates to use these new tasks. Build your Xamarin app.

Docker integration for build and release management

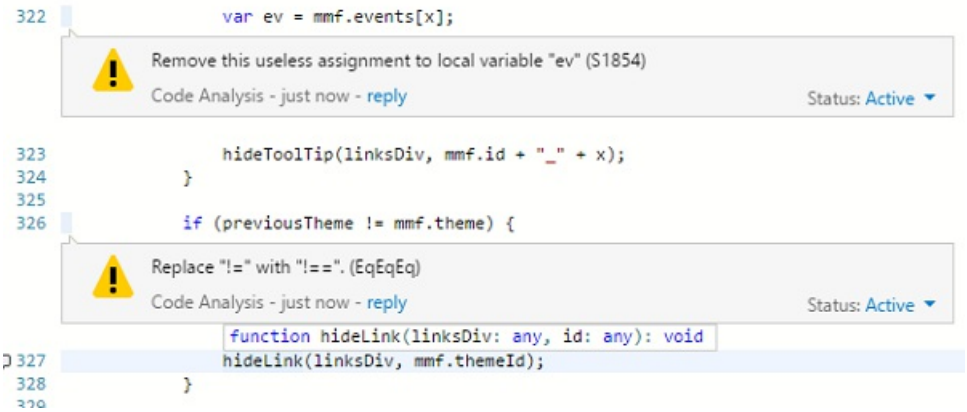
Take advantage of the build capabilities to build your Docker images and upload them to the Docker Hub as part of your continuous integration flow *(Figure 42)*. Then, deploy those images to a number of Docker hosts as part of Release Management. The Marketplace extension adds all the service endpoint types and tasks necessary for you to work with Docker.



(Figure 42) Docker images

SonarQube results in pull request view

If the build run to merge a pull request contains SonarQube MSBuild tasks, you will now see new code analysis issues as discussion comments in the pull request *(Figure 43)*. This experience works for any language for which a plug-in is installed on the SonarQube server. For more information, see the SonarQube Code Analysis issues integration into Pull Requests blog post.



(Figure 43) SonarQube pull requests

Configure status API reporting for a build definition

You can now choose which build definitions report their status back to the Git status API. This is particularly useful if you have many definitions that build a given repository or branch, but only have one that represents the real health.

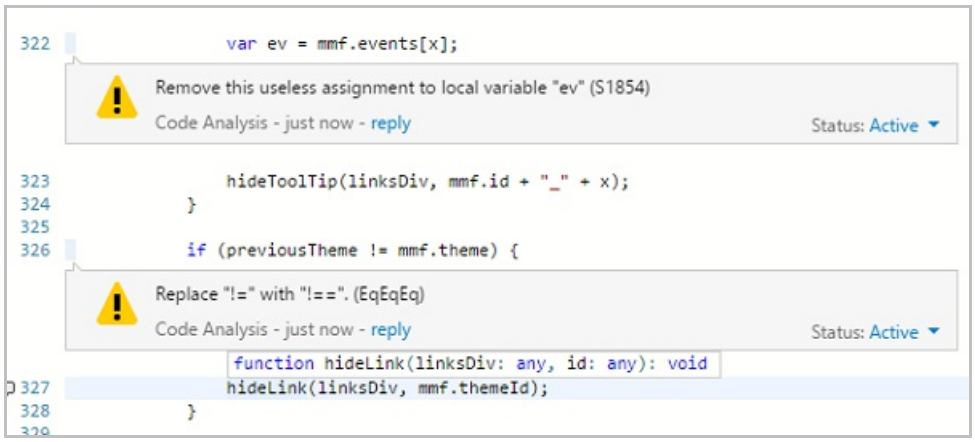
For more information, see the Build REST API reference.

Build vNext support in team rooms

It has been always possible to add notifications of XAML builds in the team room. With this sprint, users can also receive notifications from Build vNext completions.

Enable path filters for Git CI triggers

CI triggers for hosted Git repositories can include or exclude certain paths. This enables you to configure a build definition to run only when files in specific paths have changed *(Figure 44)*.



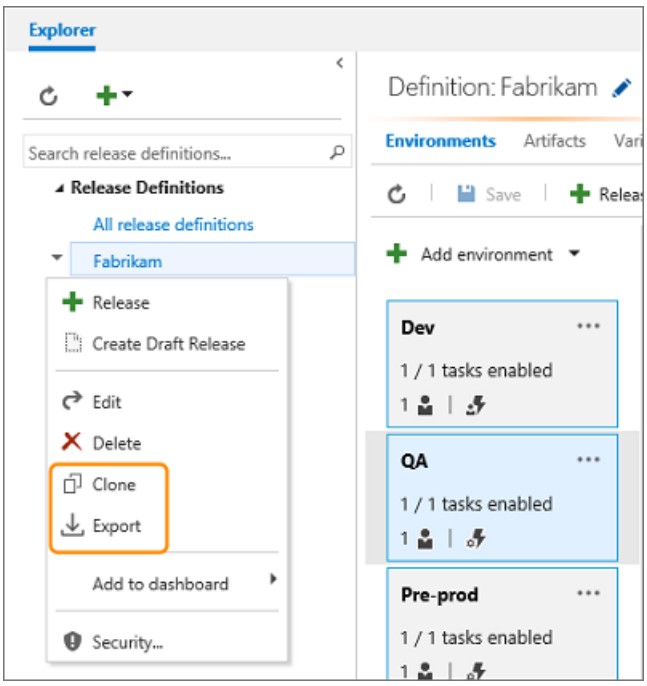
(Figure 44) Git CI Triggers

Release Management Improvements

Since the introduction of integrated web-based Release management in Team Foundation Server 2015, we have made several enhancements in this version.

Clone, export, and import release definitions

We have incorporated the ability to clone, export, and import release definitions within Release hub, without requiring installation of an extension *(Figure 45)*.



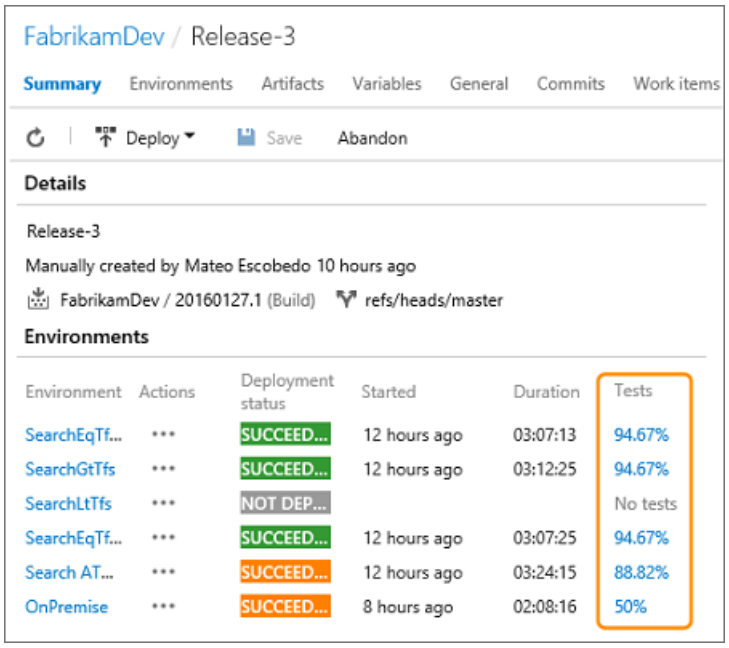
(Figure 45) Clone and export commands on release summary page

For more details, see Clone, export, and import a release definition documentation.

Test results displayed in the release summary

In the release summary page, we have enabled a contribution point for an external service to show environment-specific information.

In Team Services, this functionality is used to display a summary of test results when tests are run as part of a release environment *(Figure 46)*.



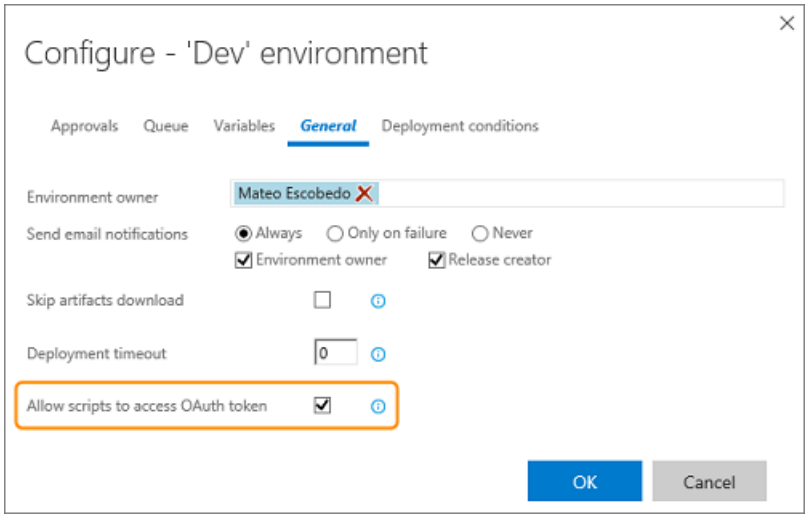
(Figure 46) Test results displayed in the release summary

For more details, see Understand the summary view of a release documentation.

Pass OAuth tokens to scripts

If you need to run a custom PowerShell script that invokes the REST APIs on Team Services, perhaps to create a work item or query a build for information, you need to pass the OAuth token in the script.

A new option when you configure an environment allows scripts to run as tasks in the environment to access the current OAuth token *(Figure 47)*.



(Figure 47) Pass OAuth tokens to scripts

For more details, see Environment general options documentation.

This is a simple example showing how to get a build definition *(Figure 48)*:

```
$url = "$($env:SYSTEM_TEAMFOUNDATIONCOLLECTIONURI)$env:SYSTEM_TEAMPROJECTID/_apis/bui
ld/definitions/$($env:SYSTEM_DEFINITIONID)?api-version=2.0"

Write-Host "URL: $url"

$definition = Invoke-RestMethod -Uri $url -Headers @{

      Authorization = "Bearer $env:SYSTEM_ACCESSTOKEN"

}

Write-Host "Definition = $($definition | ConvertTo-Json -Depth 1000)"
```

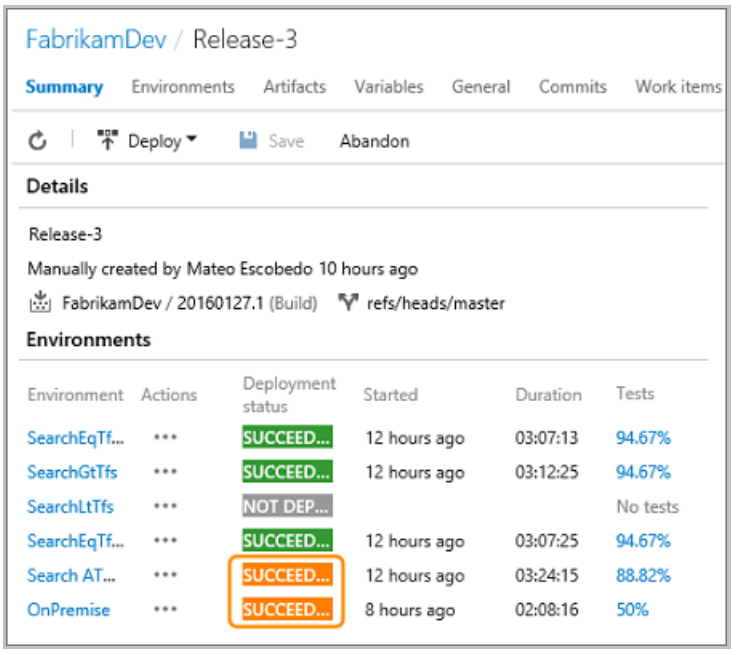
(Figure 48) Example script using passed oAuth token

Trigger on partially successful deployments

Build and release tasks have an option to **Continue on error** in the **Control Options** parameters for each task.

In a build definition, this results in a **Build partially succeeded** result if a task with this option set should fail.

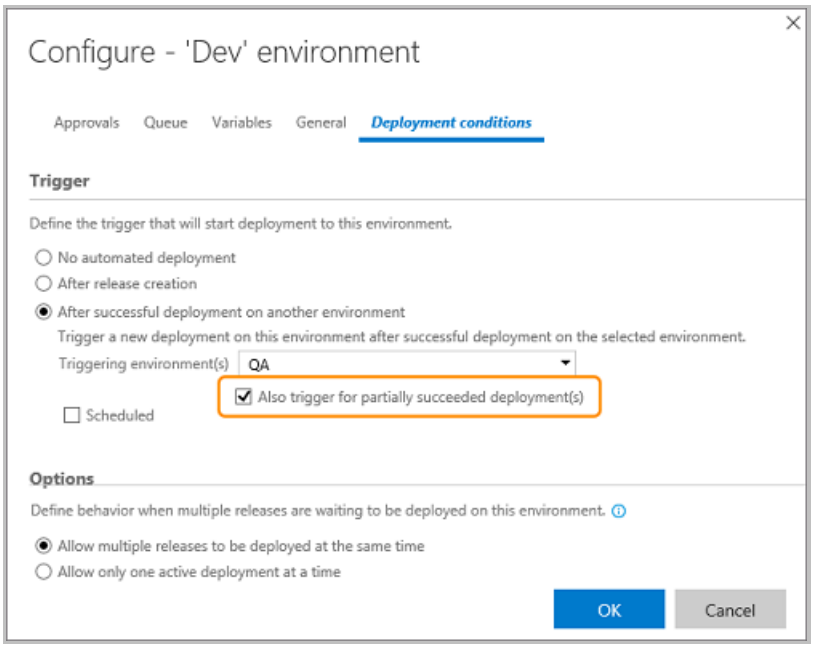
The same behavior is now available in release definitions. If a task fails, the overall release result will show as "Release partially succeeded" *(Figure 49)*.



(Figure 49) Release summary shows partially successful releases in orange color

By default, a partially successful release will not automatically trigger a release to a subsequent environment, even if this behavior is specified in the environment deployment options.

However, a new option can be set in each release environment that instructs Release Management to trigger a release to a subsequent environment when the previous release is partially successful *(Figure 50)*.



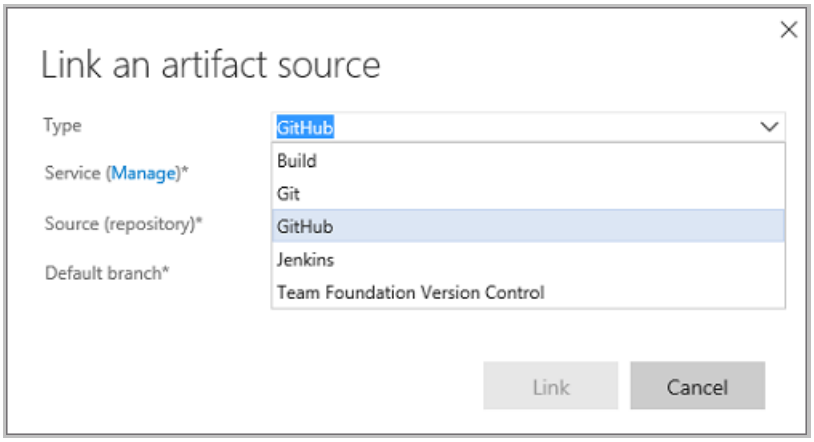
(Figure 50) Setting the option to trigger from a partially successful release

For more details, see Environment deployment triggers documentation.

Consume artifacts stored in GitHub directly

Sometimes you may want to consume artifacts stored in a version control system directly, without passing them through a build process, as described in this topic.

You can now do the same if your code is stored in a GitHub repository *(Figure 51)*.



(Figure 51) Linking code in a GutHub repository to a release definition

For more details, see TFVC, Git, and GitHub sources documentation.

Web App Deployment using ARM

A new version of the Azure Web App Deployment task is available, called AzureRM Web App Deployment.

It uses MSDeploy and an Azure Resource Manager service endpoint connection. Use this task to deploy Azure Web Jobs and Azure API apps, in addition to ASP.NET 4, Node, and Python based web apps.

The task also supports common publishing options such as the ability to retain app data, take an app off-line, and remove additional files at the destination.

More features, such as configuration transformations, may appear in forthcoming versions *(Figure 52)*.

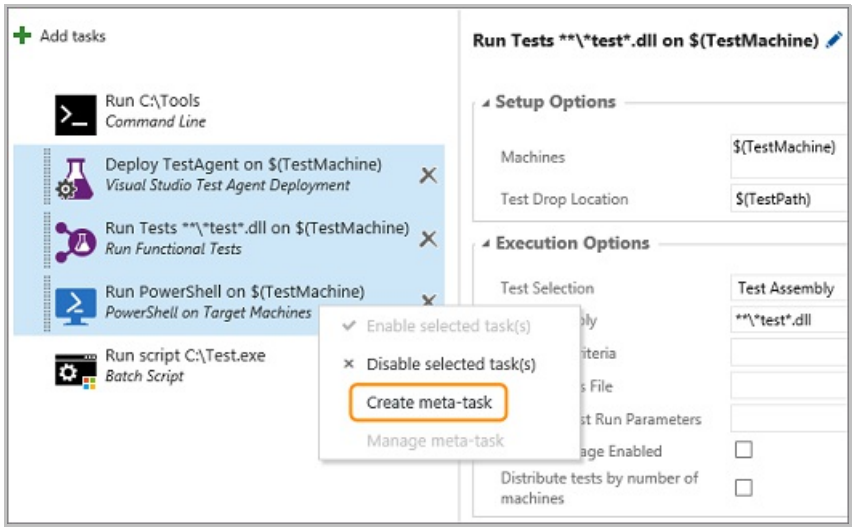
(Figure 52) Web app deployment using ARM

Task groups

A *task group* lets you encapsulate a sequence of tasks already defined in a build or a release definition into a single reusable task that can be added to a build or release definition just like any other task *(Figure 53)*.

You can choose to extract the parameters from the encapsulated tasks as configuration variables, and abstract the rest of the task information.

The new task group is automatically added to the task catalogue, ready to add to other release and build definitions.



(Figure 53) Linking code in a GutHub repository to a release definition

For more details, see Task Groups documentation.

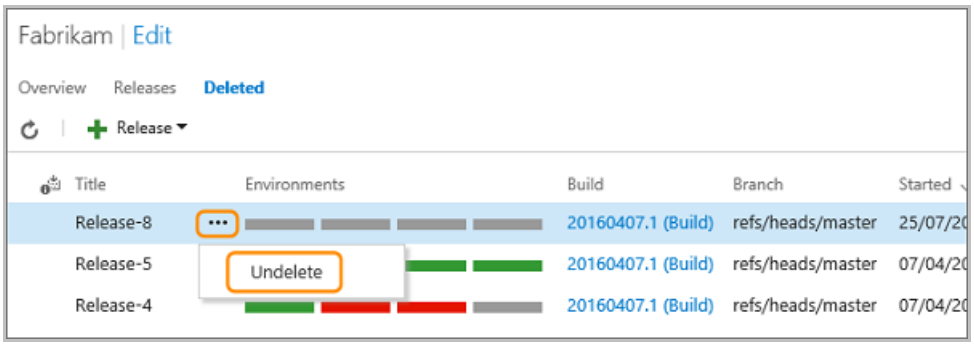
Soft delete of releases

When you delete a release, or it is automatically deleted by a retention policy, the release is removed from the overview and details lists.

However, it is retained with the release definition for a period (typically 14 days) before it is permanently deleted.

During this period, it is shown in the **Deleted** tab of the overview and details lists.

You can restore any of these releases by opening the shortcut menu and choosing **Undelete** *(Figure 54)*.



(Figure 54) Undelete releases

For more details, see Restore deleted releases documentation.

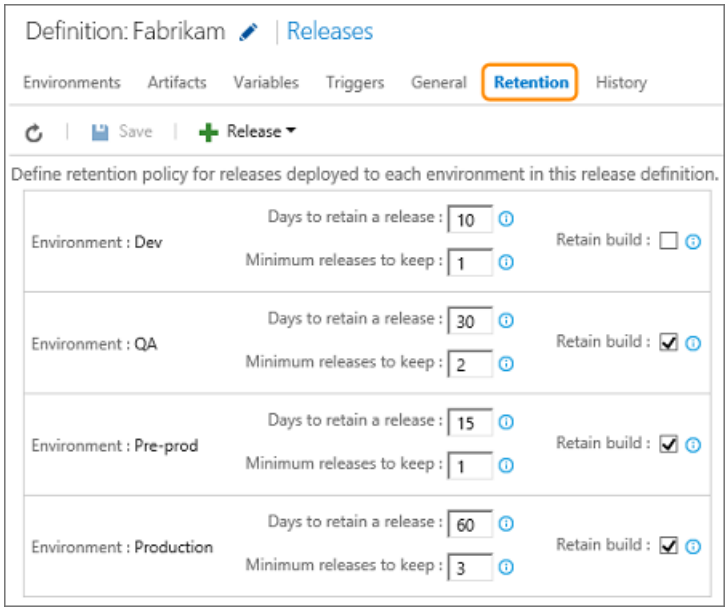
Retain releases and builds for each environment

The release retention policy for a release definition determines retention duration for a release and linked build.

By default, a release is retained for 60 days. Releases that have not been deployed or modified during that time are automatically deleted.

However, you may want to retain more releases that have been deployed to specific environments, such as your production environment, or retain them longer than those that were just deployed to other environments such as test, staging, and QA.

You can also retain the build linked to a release for the same period as the release to ensure that the artifacts are available if you need to redeploy that release *(Figure 55)*.



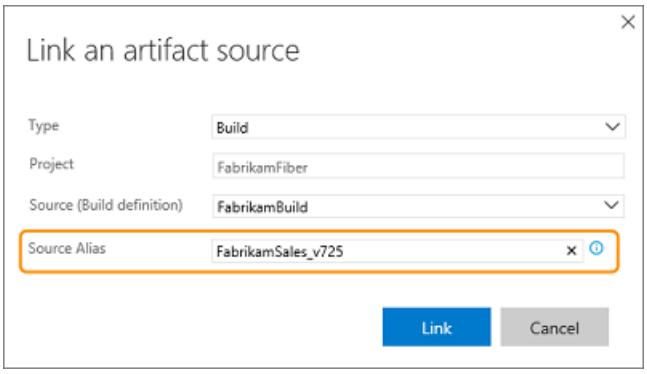
(Figure 55) Retain releases

For more details, see Release and build retention documentation.

Linked artifact improvements

Two new features make it easier to work with artifacts and artifact sources:

- You can link multiple artifact sources to a release definition *(Figure 56)*. Each artifact is downloaded into a folder on the agent called the source alias. You can now edit the source alias of a linked artifact. For example, when you change the name of the build definition, you can edit the source alias to reflect the name of the build definition.



(Figure 56) Linked artifact improvements

```
For more details, see [Artifact source alias](https://docs.microsoft.com/vsts/pipelines/release/artifacts?view=vsts#source-alias) documentation.
```

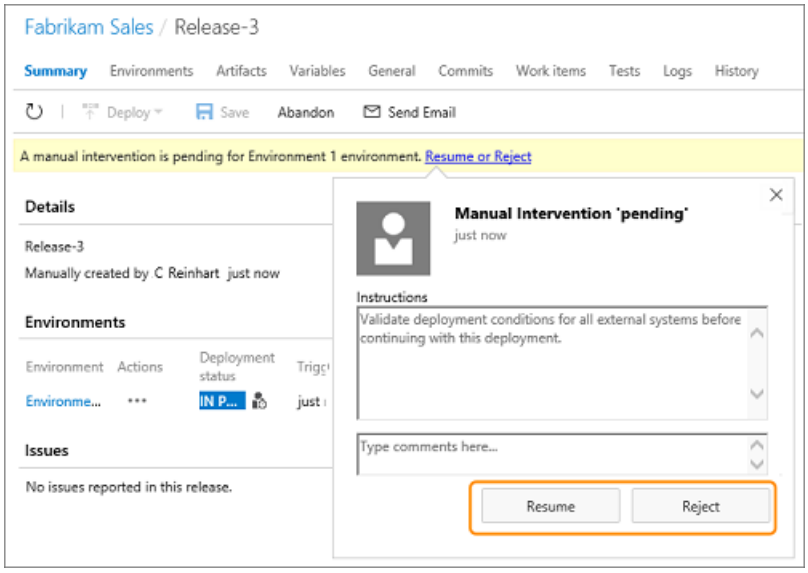
- A number of variables of the format Build.* (such as Build.BuildId and Build.BuildNumber) are exposed for use in task parameters. When multiple sources are associated with a release, these variables now populate with values from the artifact source you specify as the primary source. For more details, see Artifact variables documentation.

Deployment - Manual Intervention task

You can now pause execution during deployment to an environment.

Including a Manual Intervention task in an environment enables you to temporarily halt a deployment, perform manual steps, and then resume further automated steps.

You can also reject the deployment and prevent further steps from executing after a manual intervention *(Figure 57)*.

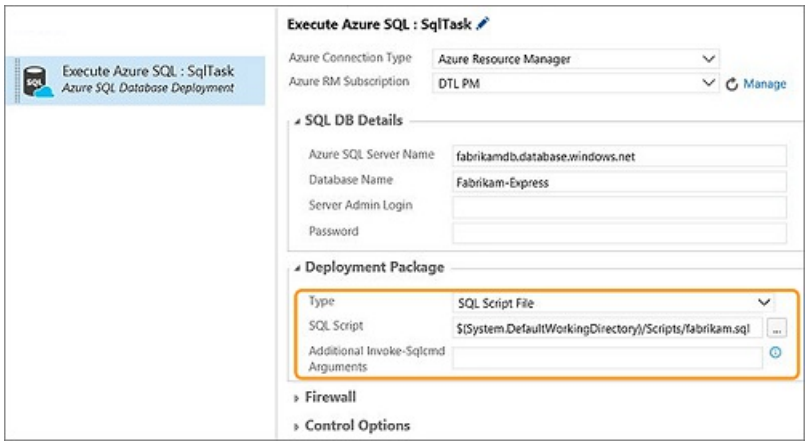


(Figure 57) Manual intervention task

For more details, see Manual intervention documentation.

SQL Database deployment task scripts

The **Azure SQL Database Deployment** *(Figure 58)* task has been enhanced to run SQL scripts against an Azure SQL Database. The scripts can be provided as a file, or inline within the task.



(Figure 58) SQL database deployment task scripts

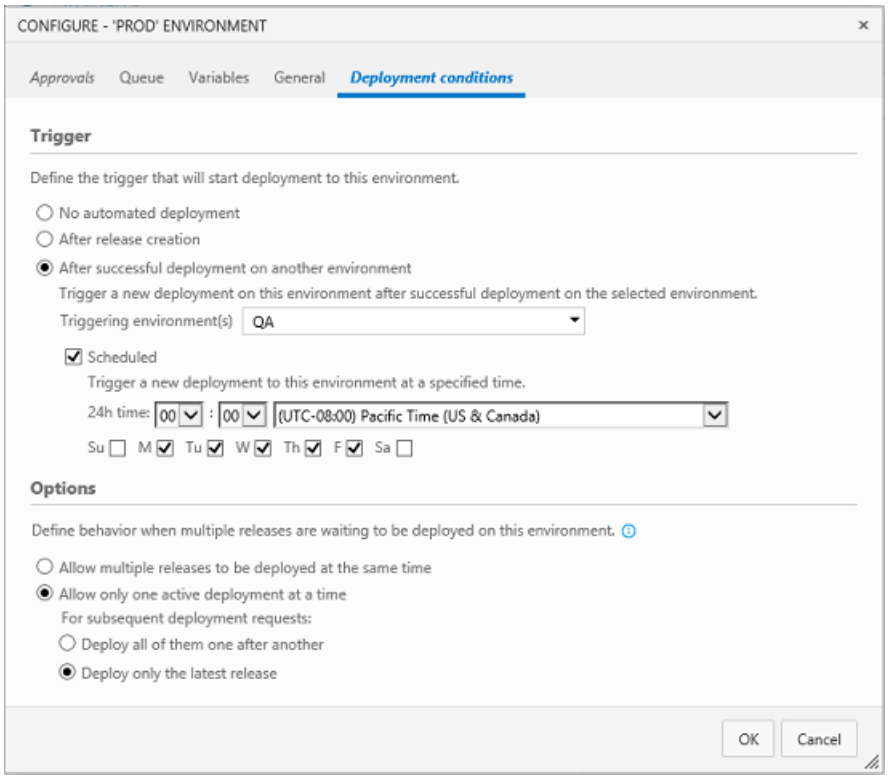
Release definition summary - dashboard widget

Pin a release definition to the dashboard - an easy way to make a summary of releases for that definition visible to all your team.

For more details, see Add release information to the dashboard documentation.

Promote releases to an environment at a specific time

Want all your production deployments to happen at midnight? You can configure a condition on an environment that selects a successful deployment (or just the latest one) from another environment, and deploys it at the specified time *(Figure 59)*.



(Figure 59) Schedule release to an environment

Deploy based on conditions in multiple environments

Until the previous version, you could do parallel deployments (*forkdeployments*), but you could not start a deployment to an environment based on the status of multiple environments (*join deployments*). Now you can.

For more details, see Parallel forked and joined deployments documentation.

REST APIs for release management

You can use the REST APIs for the Release Management service to create release definitions and releases, and manage many aspects of deploying a release.

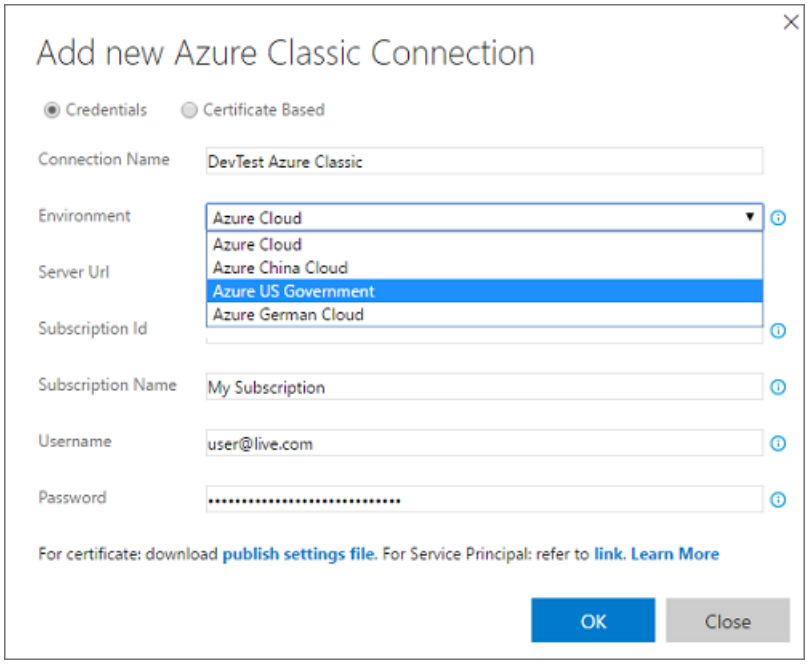
For more information, see the API reference documentation. You will find some basic examples that use the APIs in this blog post, Using ReleaseManagement REST API's.

Service hooks integration

Send release notifications when new releases are created, deployments are started or completed, or when approvals are pending or completed. Integrate with third party tools such as Slack to receive such notifications.

Deployment to national Azure clouds

Use the new Environment setting in an Azure Classic service endpoint to target a specific Azure cloud, including pre-defined national clouds such as Azure China cloud, Azure US Government cloud, and Azure German cloud *(Figure 60)*.



(Figure 60) Deployment to national Azure clouds

For more details, see Azure Classic service endpoint documentation.

Test Improvements

Key test improvements have been added in Team Foundation Server 2017.

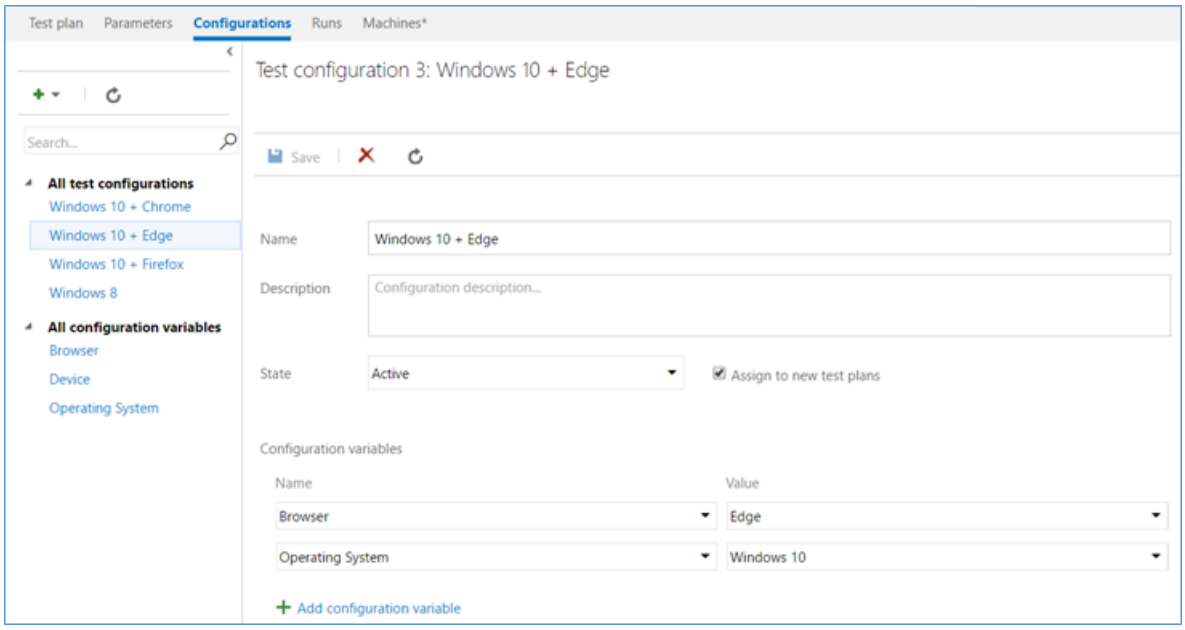
Updated test result storage schema

In this release, we are migrating the test result artifacts to a new compact and efficient storage schema. Since test results are one of the top consumers of storage space in TFS databases, we expect this feature to translate into reduced storage footprint for TFS databases. For customers who are upgrading from earlier versions of TFS, test results will be migrated to the new schema during TFS upgrade. This upgrade may result in long upgrade times depending on how much test result data exists in your databases. It is advisable to configure the test retention policy and wait for the policy to kick in and reduce the storage used by test results so that the TFS upgrade is faster. After installing TFS, but before upgrading the TFS instance, you can use the TFSConfig.exe tool to clean up test results. See TFSConfig.exe help for more details. If you do not have the flexibility to configure test retention or clean up test results before upgrade, make sure you plan accordingly for the upgrade window. See Test result data retention with Team Foundation Server 2015 for more examples about configuring test retention policy.

Test Hub improvements

Test configuration management in Test Hub

We have brought test configuration management to the web UI by adding a new Configurations tab within the Test Hub *(Figure 61)*. Now you can create and manage test configurations and test configuration variables from within the Test hub.

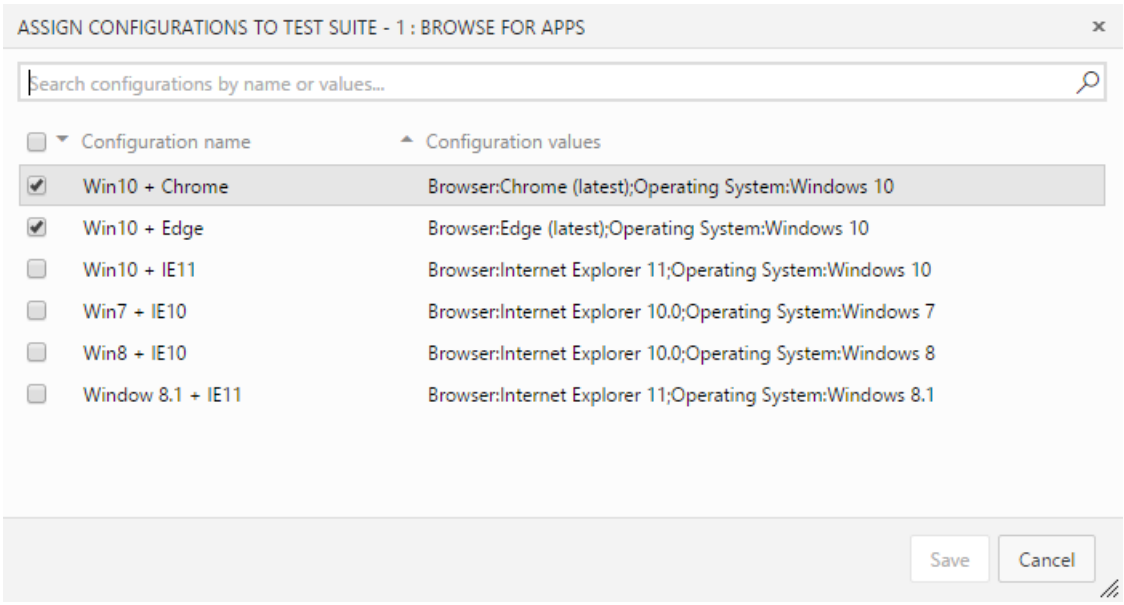


(Figure 61) Configurations hub

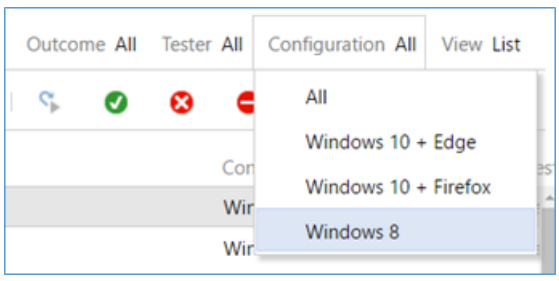
For more information, see Create configurations and configuration variables.

Assigning configurations to test plans, test suites, and test cases

Assigning configurations just got easier. You can assign test configurations to a test plan, test suite, or test case(s) directly from within the Test hub *(Figure 62)*. Right-click an item, select **Assign configurations to ...**, and you're off and running. You can also filter by Configurations in the Test hub *(Figure 63)*.



(Figure 62) Assign Configurations

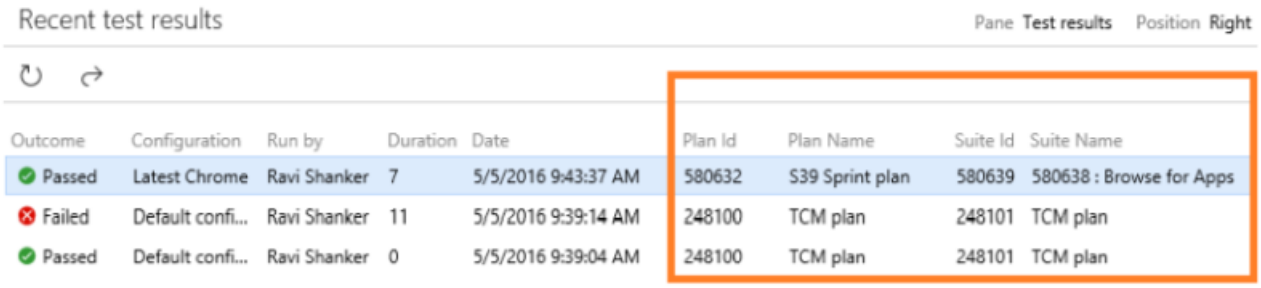


(Figure 63) Configurations Filter

For more information, see Assign configurations to Test plans and Test suites.

View test plan/test suite columns in test results pane

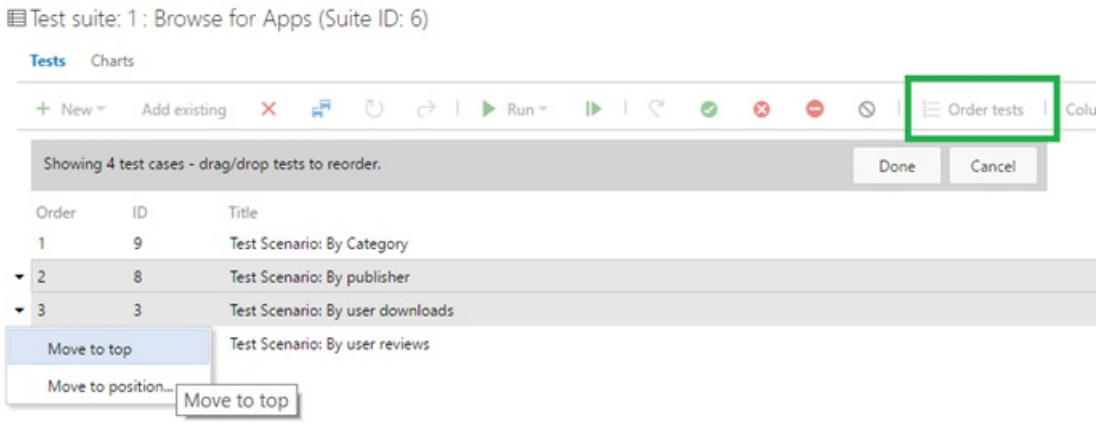
We have added new columns to the Test results pane that show you the test plan and test suite under which the test results were executed in. These columns provide much-needed context when drilling into results for your tests *(Figure 64)*.



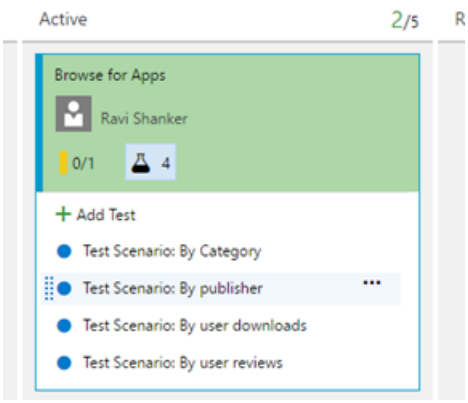
(Figure 64) Test results pane

Ordering of tests in Test Hub & on cards

You can now order manual tests from within the Test Hub *(Figure 65)*, irrespective of the type of suite in which they are included: static, requirement-based, or query-based suites. You can simply drag and drop one or more tests or use the context menu to reorder tests. Once the ordering is completed, you can sort your tests by the Order field and then run them in that order from the Web runner. You can also order the tests directly on a user story card on the Kanban board *(Figure 66)*. This completes one of the long-pending user voice items (with 495 votes) under manual testing.



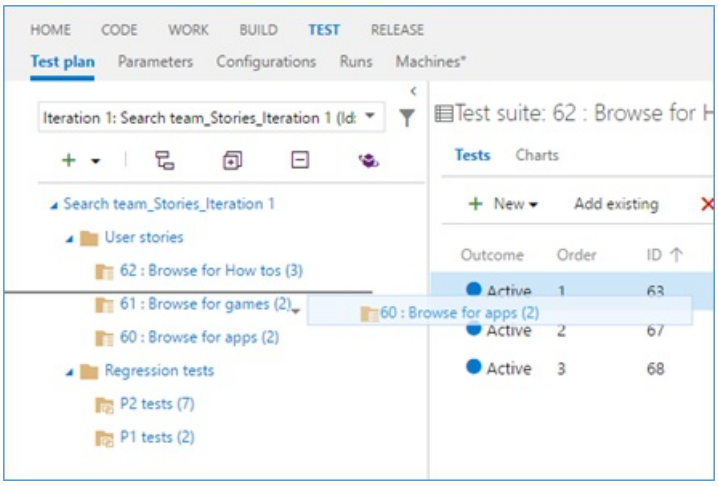
(Figure 65) Order tests



(Figure 66) Order tests on card

Order test suites in Test Hub

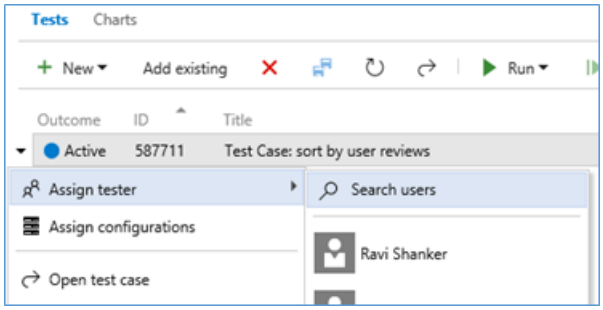
Test teams can now order the test suites as per their needs. Prior to this capability, the suites were only ordered alphabetically. Now, using the drag/drop capability in Test hub, suites can be re-ordered among the peer suites or moved to another suite in the hierarchy *(Figure 67)*. This addresses the following user voice item under manual testing/test case management.



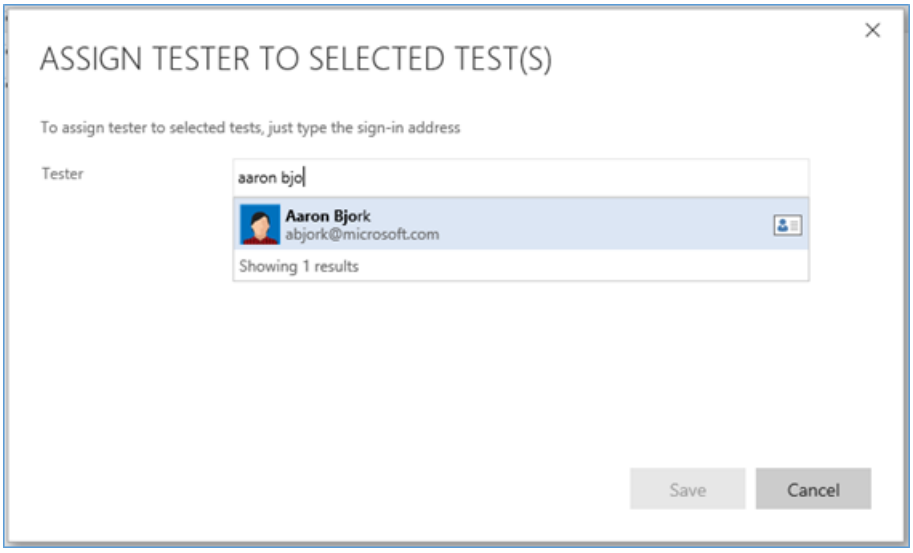
(Figure 67) Order Test suites

Search for users as part of assigning testers

As part of the rollout of new identity picker controls across the different hubs, in Test hub, we have also enabled the option to search for users when assigning testers to one or more tests *(Figure 68)*. This is extremely useful in scenarios where the number of team members is large, but the context menu only shows a limited set of entries * (Figure 69).



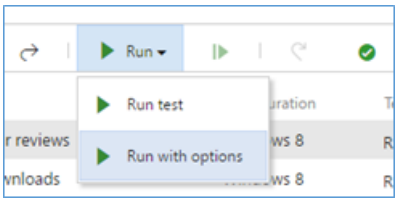
(Figure 68) Search users



(Figure 69) Assign Users

Pick a build to test with

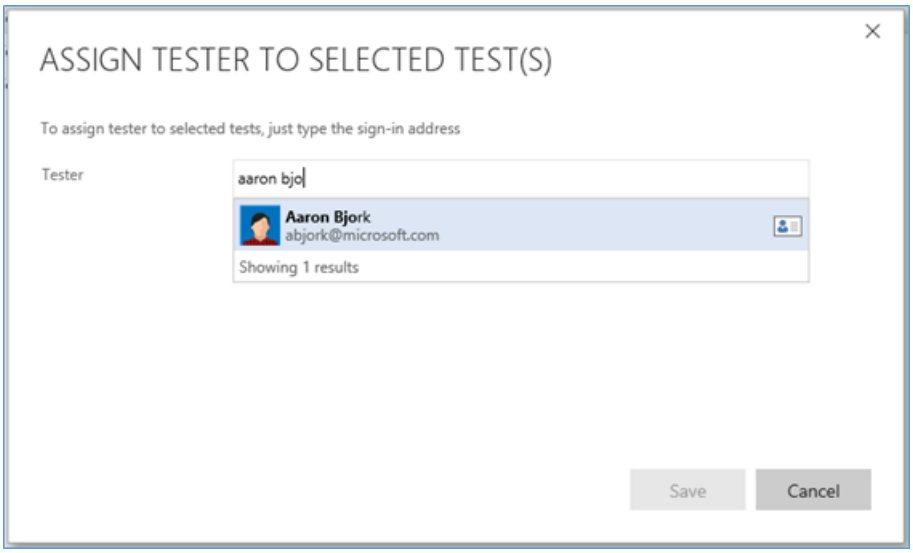
You can now pick the "build" you want to test with and then launch the Web runner, using 'Run with options' in Test hub *(Figure 70)*. Any bug filed during the run is automatically associated with the build selected. In addition, the test outcome is published against that specific build.



(Figure 70) Pick a build

Launch Microsoft Test Runner client from Test Hub with data collectors

You can now choose your data collectors & build to associate with the test run *(Figure 71)*, and launch the Microsoft Test Runner 2017 (client) in a performant way from Test hub, without having to configure them in Microsoft Test Manager client. The Microsoft Test Runner launches without opening the entire Microsoft Test Manager shell and will shut-down on completion of the test execution.

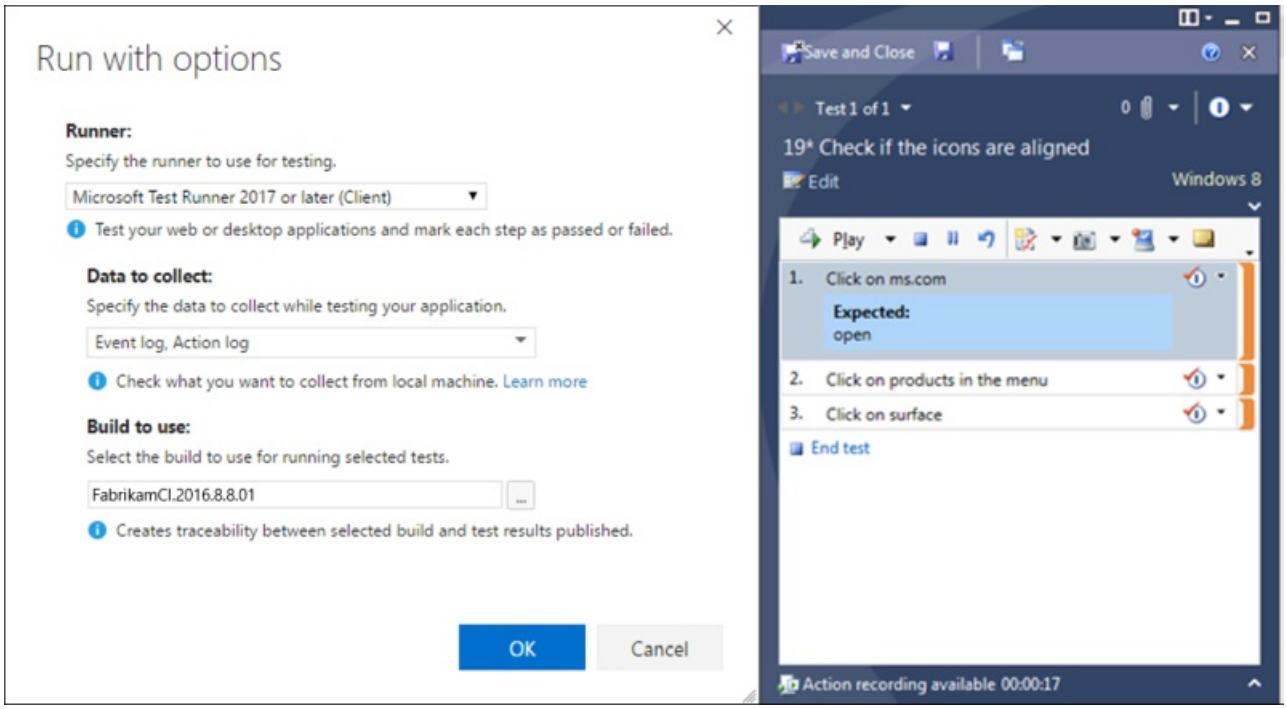


(Figure 71) Run with options

For more information, see Run tests for desktop apps.

Choose data collectors and launch Exploratory Runner client from Test hub

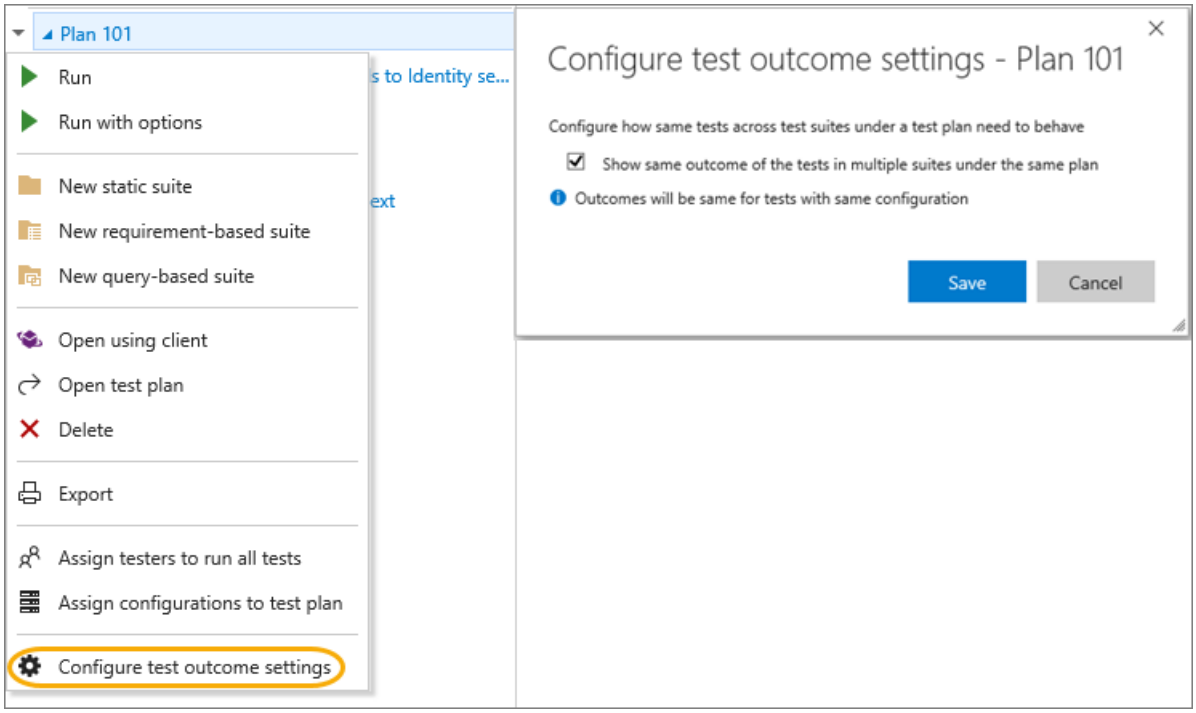
You can now choose your data collectors and launch the Exploratory Runner 2017 (client) in a performant way from Test hub, without having to configure them in Microsoft Test Manager client. Invoke 'Run with options' from the context menu *(Figure 72)* for a Requirement based suite and choose Exploratory runner and the data collectors you need. The Exploratory runner launches similar to Microsoft Test Runner as described above.



(Figure 72) Run with Options - XT

Configure test outcomes for tests across different test suites

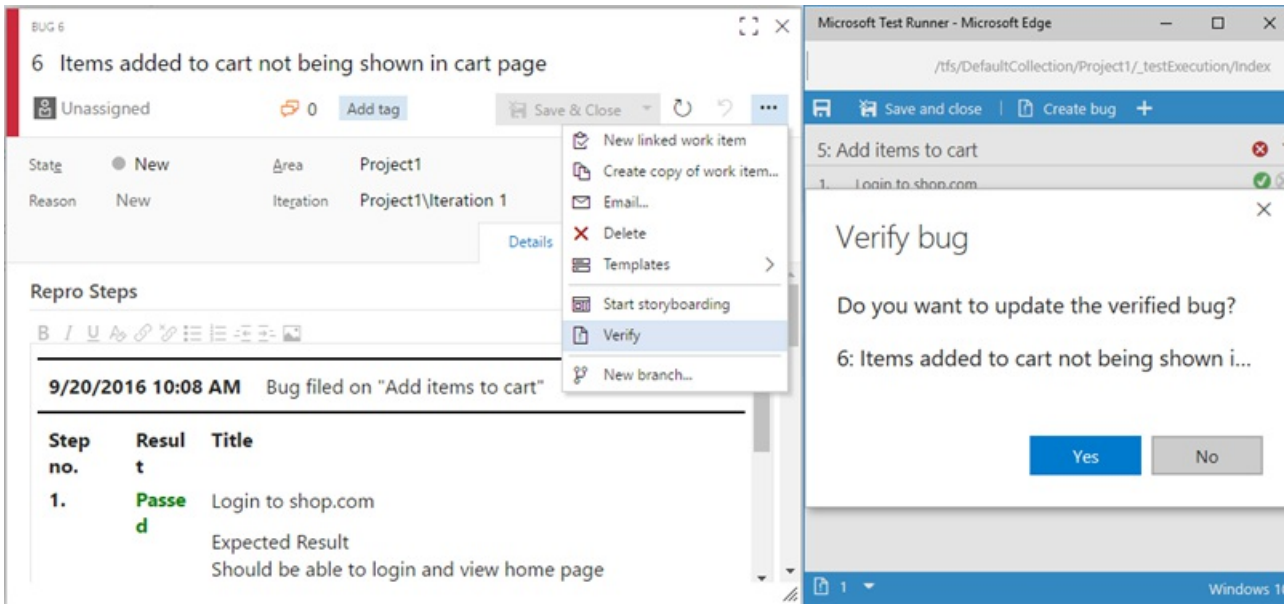
We have now added the ability to configure the behavior of test outcomes for tests shared across different test suites under the same test plan *(Figure 73)*. If this option is selected, and you set the outcome for a test (mark it as Pass/Fail/Blocked either from the Test hub, Web runner, Microsoft Test Runner, or from cards on Kanban board), that outcome will propagate to all the other tests present across different test suites under the same test plan, with the same configuration. Users can set the "Configure test outcomes" option for a particular test plan either from the Test hub test plan context menu or from the Kanban board test page in the common settings configuration dialog. This option is turned off by default and you will have to explicitly enable it to take effect.



(Figure 73) Configure test outcomes

Verify bugs from work item

You can now verify a bug by re-running the tests which identified the bug *(Figure 74)*. You can invoke the Verify option from the bug work item form context menu to launch the relevant test case in the web runner. Perform your validation using the web runner and update the bug work item directly within the web runner.



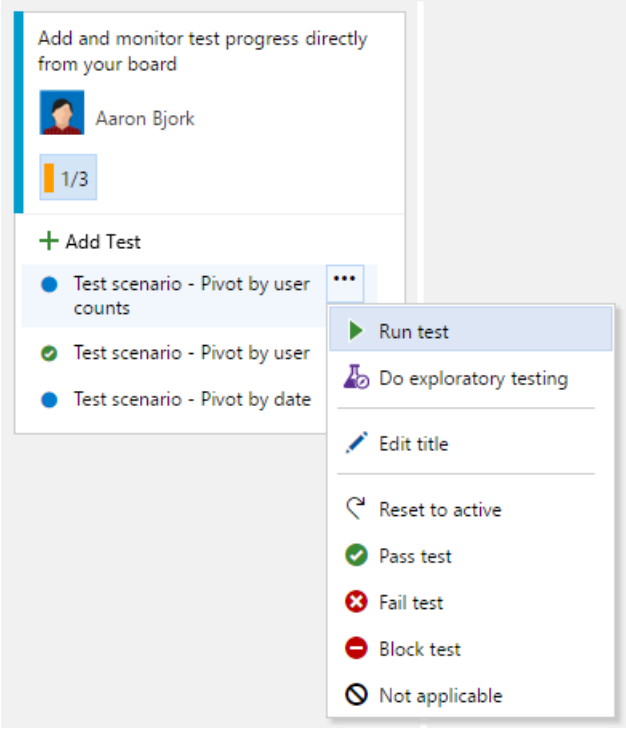
(Figure 74) Verify bugs

REST APIs for test plan / test suite clone

We have added REST APIs for cloning of test plans and test suites. You can find them under the Test Management section on our Team Services Integrate site.

Test progress from your Kanban cards

You can now add, view, and interact with test cases directly from your stories on the Kanban board. Use the new **Add Test** menu option to create a linked Test case, and then monitor status directly from the card as things progress *(Figure 75)*.



(Figure 75) Inline tests

With this new capability, you can now perform the following actions directly from a card on your board:

- Add tests.
- Open tests.
- Reparent a test by dragging/dropping from one user story to another.
- Copy the same test to another user story using CTRL+Drag/Drop (for scenarios where the same test case tests

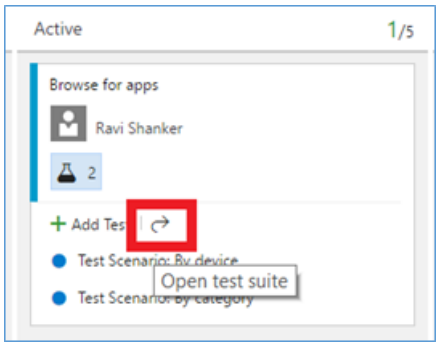
more than one user story).

- Update the test status by quickly marking it Pass/Fail/etc.
- Run the test by launching it in the Web Test Runner, from which you can pass or fail individual steps, file bugs, etc.
- View a summary of the roll-up status indicating how many tests have passed and how many remain for that story.

If you need advanced test management capabilities (like assign testers, assign configurations, centralized parameters, exporting test results, etc.), you can then switch over to Test Hub and start using the default test plan/requirement-based suites that have been auto-created for you. For more information, see Add, run, and update inline tests.

Traverse to a test plan/test suite from the card

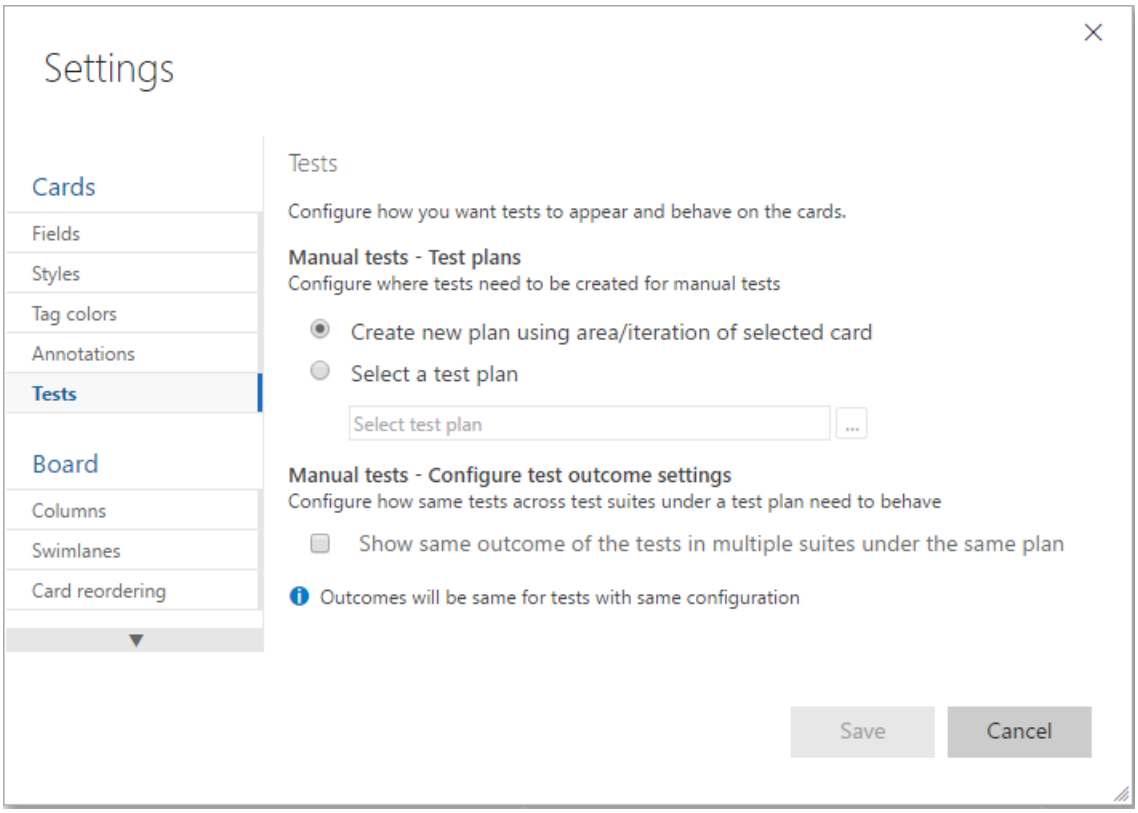
You can now easily traverse to the underlying test plan/test suite under which the tests are created, directly from a card on the Kanban board. Clicking on this link *(Figure 76)* will take you to the Test hub, open the right test plan, and then select the specific suite that controls those inline tests.



(Figure 76) Traverse to plan/suite

Test page in common settings configuration of Kanban board

Use the new Tests page in common settings configuration dialog on Kanban board to control the test plan where the inline tests are created *(Figure 77)*. Previously, any tests created on a card would automatically be added to a newly created test plan, provided no test plans existed that matched the area & iteration paths of the card. Now, you can override this behavior by configuring an existing test plan of your choice – all the tests are added to the selected test plan. Note that this functionality is only enabled if the Test annotation is turned on.

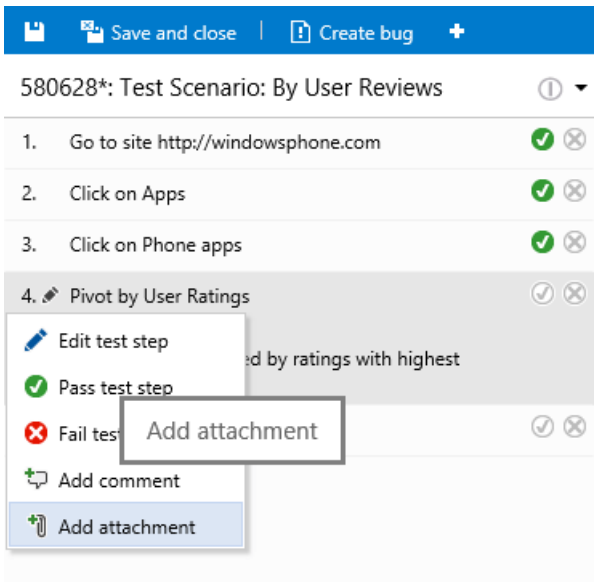


(Figure 77) Common settings

Web runner enhancements

Add test step attachments during manual testing

We have enhanced the Web test runner to give you the ability to add test step attachments during manual testing *(Figure 78)*. These step result attachments automatically show up in any bugs you file in the session and subsequently in the Test results pane.

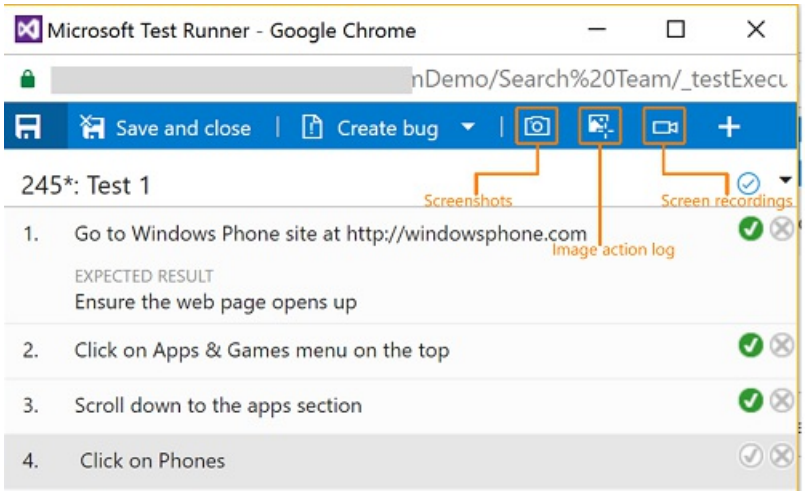


(Figure 78) Test Step attachments

Screenshot, screen recording, image action log and system info support in Web runner (using Chrome browser)

You can now take screenshots and annotate them inline when you use Web runner in Chrome *(Figure 79)*. You can also capture on-demand screen recordings of not just the web apps, but also your desktop apps. These screenshots and screen recordings are automatically added to the current Test step. In addition to screenshots & screen recordings, you can also capture on-demand image action log from your web apps. You need to specify the browser window on which to capture your actions – all actions on that window (any existing or new tabs you open in that window) or any new child browser windows you launch, will automatically be captured and correlated against the test steps being tested in the Web runner. These screenshots, screen recordings and image action logs

are then added to any bugs you file during the run and attached to the current test result. Similarly, the system information data is automatically captured and included as part of any bugs you file from the Web runner. All these leverage the capability from the Chrome-based Test & Feedback extension.



(Figure 79) Web runner using Chrome browser

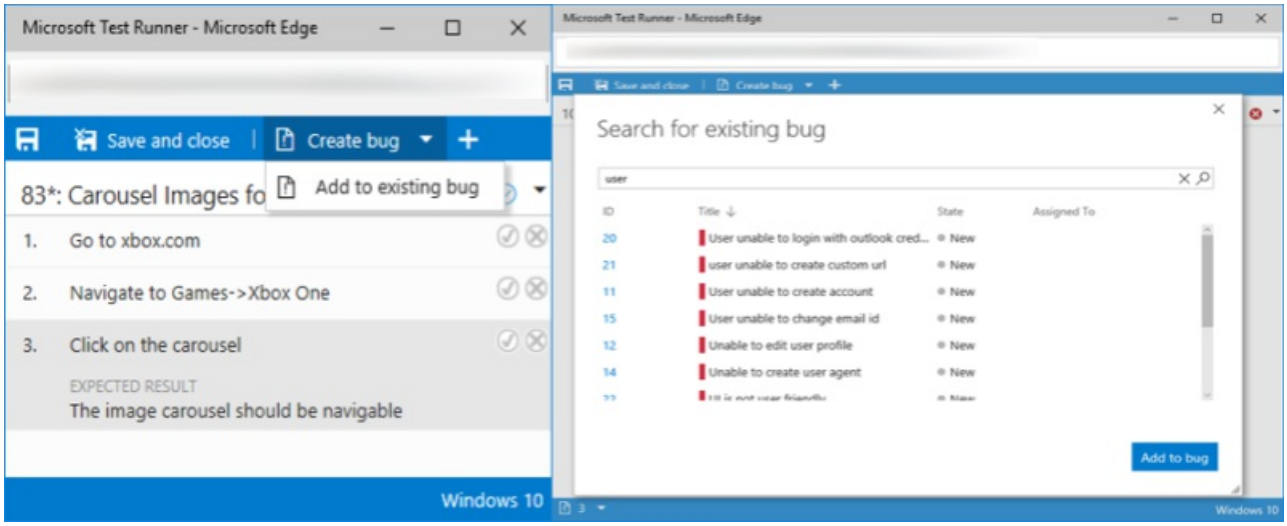
For more information, see Collect diagnostic data during tests.

Bugs filed as children – Web runner/test & feedback extension

When running tests in Web runner, launched either from a card on the board or from a requirement-based suite in Test hub, any new bugs filed will now be automatically created as a child to that user story. Similarly, if you are exploring a user story from the exploratory testing extension, any new bugs you file are also created as a child to that user story. This new behavior allows for simpler traceability across stories and bugs. This is applicable only if the "Working with bugs" settings in the Common Settings Configuration page is set to "Bugs do not appear on backlogs or board" or "Bugs appear on the backlogs and boards with tasks". For all other settings for "Working with bugs" option and in certain other scenarios, such as adding to an existing bug that already has a parent defined, a Related link is created instead.

Update existing bugs from Web runner

In addition to creating new bugs from the Web runner, now you can also update an existing bug *(Figure 80)*. All the diagnostic data collected, repro steps, and links for traceability from the current session are automatically added to the existing bug.



(Figure 80) Add to existing bug

Test & feedback extension - enhancements

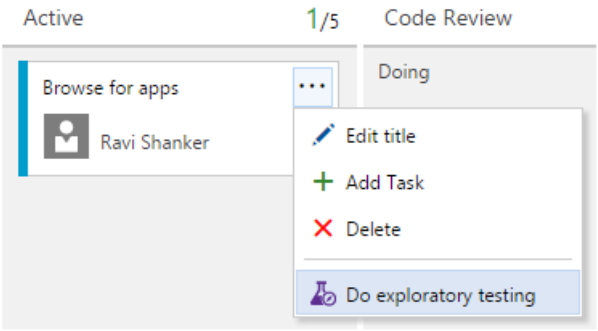
The browser-based Test & Feedback extension can be installed from the Visual Studio Marketplace. It supports both Visual Studio Team Services and Team Foundation Server (2015 or later).

Explore work items

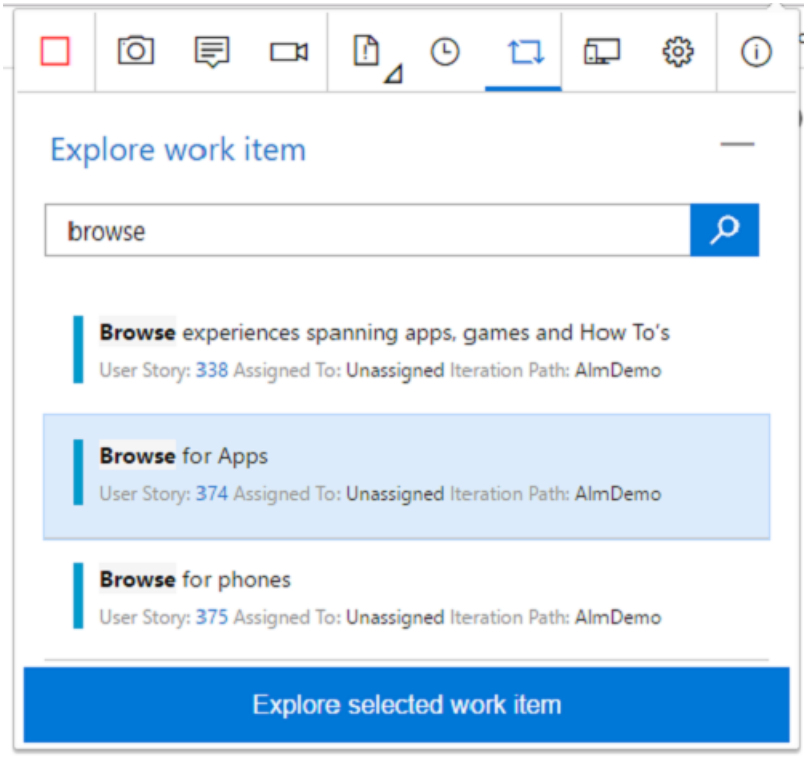
You can now do exploratory testing for a specific work item *(Figure 81)*. This lets you associate the selected work item with your ongoing testing session, and view the acceptance criteria and description, from within the extension. It also creates end-to-end traceability between bugs or tasks that you file on the selected work item. You can explore the work item either directly from a work item, or from within the extension:

• Directly from a work item *(Figure 81)*: Launch exploratory testing session for a specific work item directly from within the product using the "Do exploratory testing" option in the context menu. We have added entry points on all cards, grids, and in the Test hub.

• Within the extension *(Figure 82)*: Search for a work item from within the XT session and then associate it with the ongoing session.



(Figure 81) XT from work item



(Figure 82) XT from extension

For more information, see Explore work items with the Test & Feedback extension.

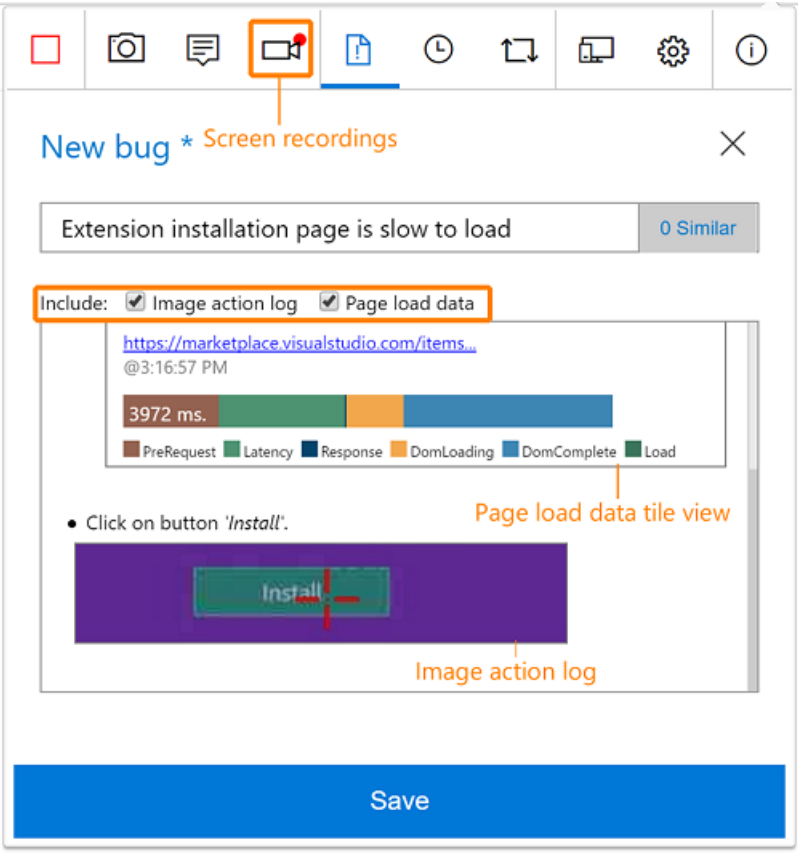
Capture image action log, screen recordings and web page load data using test & feedback

Image Action Log: The extension gives you a new option to add the steps that lead you to the bug automatically with just one click. Select the "Include image action log" option *(Figure 83)* to capture the mouse, keyboard, and touch actions, and add the corresponding text and images directly into the bug or task.

Screen recording as video: You can also capture on-demand screen recordings using the extension. These screen recordings can be captured not just from the web apps, but also your desktop apps. You can configure the extension to automatically stop screen recordings and attach them to a bug being filed using the extension's

"Options" page.

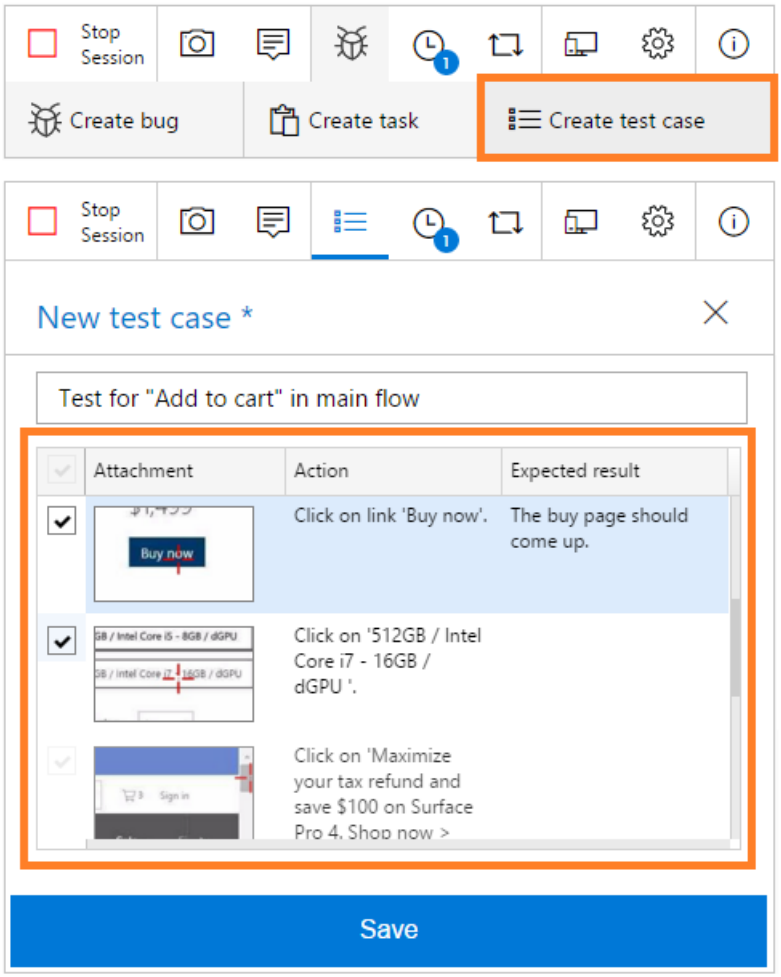
Page Load Data: We have added a new background capture capability to the extension – capturing of "web page load" data. Just like the "image action log" captured your actions performed on a web app being explored, in the form of images in the background, the "page load" functionality automatically captures details for a web page to complete the load operation. Instead of relying on subjective/perceived slowness of web page load, you can objectively quantify the slowness in the bug now. Once the bug is filed, in addition to the tile view, a detailed report is also attached to the bug, which can help the developer with their initial set of investigations.



(Figure 83) XT Image Action Log

Create test cases based on image action log data

When you create test cases during your exploratory session, the test steps with images are automatically filled in for you *(Figure 84)*. Simultaneous test design and test execution is the basis of true exploratory testing, and this new capability makes this a reality. You can edit the text captured, add the expected result, uncheck rows not relevant, and save it for upcoming test passes/runs.



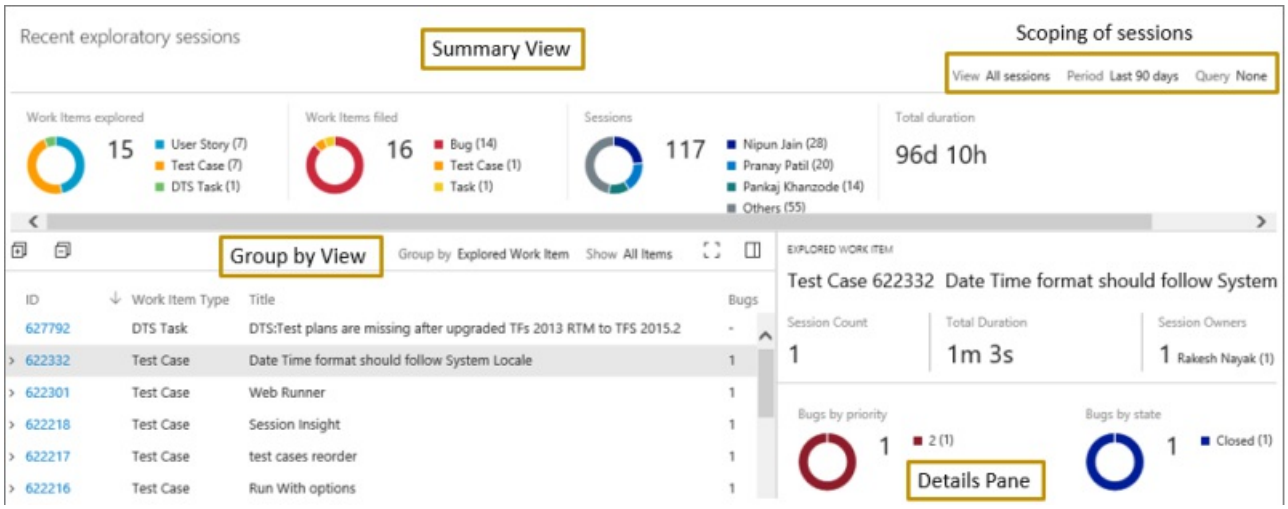
(Figure 84) XT Create Test Cases

For more information, see Create test cases based in image action log data.

Exploratory testing session insights

You can now view the completed exploratory testing sessions, either at a team or individual level, for a given time period created using the Test & Feedback extension. You can get to this insights page by clicking the "Recent exploratory sessions" link in the Runs hub within the Test Hub group in web access. This new view helps you derive meaningful insights, including:

- The summary view that shows a breakdown of the work items explored, the work items created, and the session owners, along with the total time spent on these sessions *(Figure 85)*.
- The group-by view can be pivoted by either explored work-items, sessions, session owners, or none. For any pivot, you can either view the list of all work items (bugs, tasks, test cases) created or scope the list down to a specific work item type.
- The details pane view that displays information based on selection in the group-by view. For a selected pivot row (say explored work items), you can view its summary information in the details pane, such as the total number of sessions, the total time spent across these sessions, the session owners who explored it, and the bugs/tasks/test cases created against it, along with their state and priority. For a selected work item row, you can view its work item form inline and make changes as appropriate.

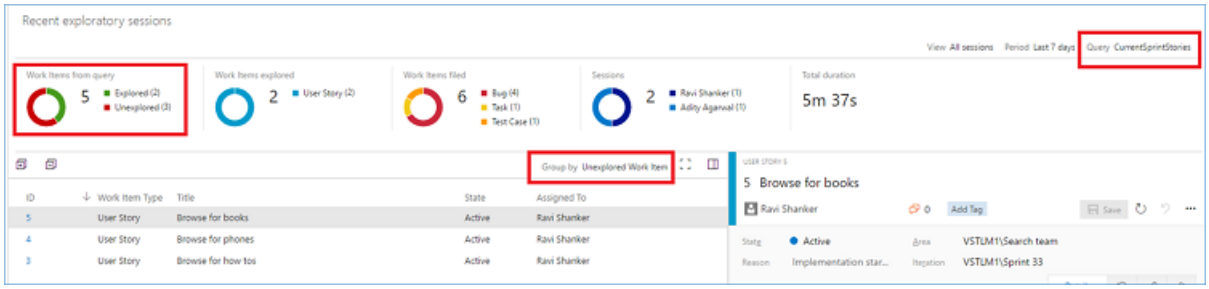


(Figure 85) XT Session insights

For more information, see Get insights across your exploratory testing sessions.

Exploratory testing sessions: View unexplored work items

In addition to seeing the details of all the explored work items in the "recent exploratory sessions" view, filtered by all/my sessions for a given date range, we have now added the ability to also see a list of all work items that have *NOT* been explored, in the same view *(Figure 86)*. You start by specifying a shared query for work items that you are interested in and the sessions page shows a list of all the work items from the query, with a breakdown of both explored and unexplored items in the summary section. In addition, using the "Unexplored Work Item" group by pivot, you can see the list of items that have not been explored yet. This is extremely useful to track down how many stories have not been explored or gone through a bug-bash yet.

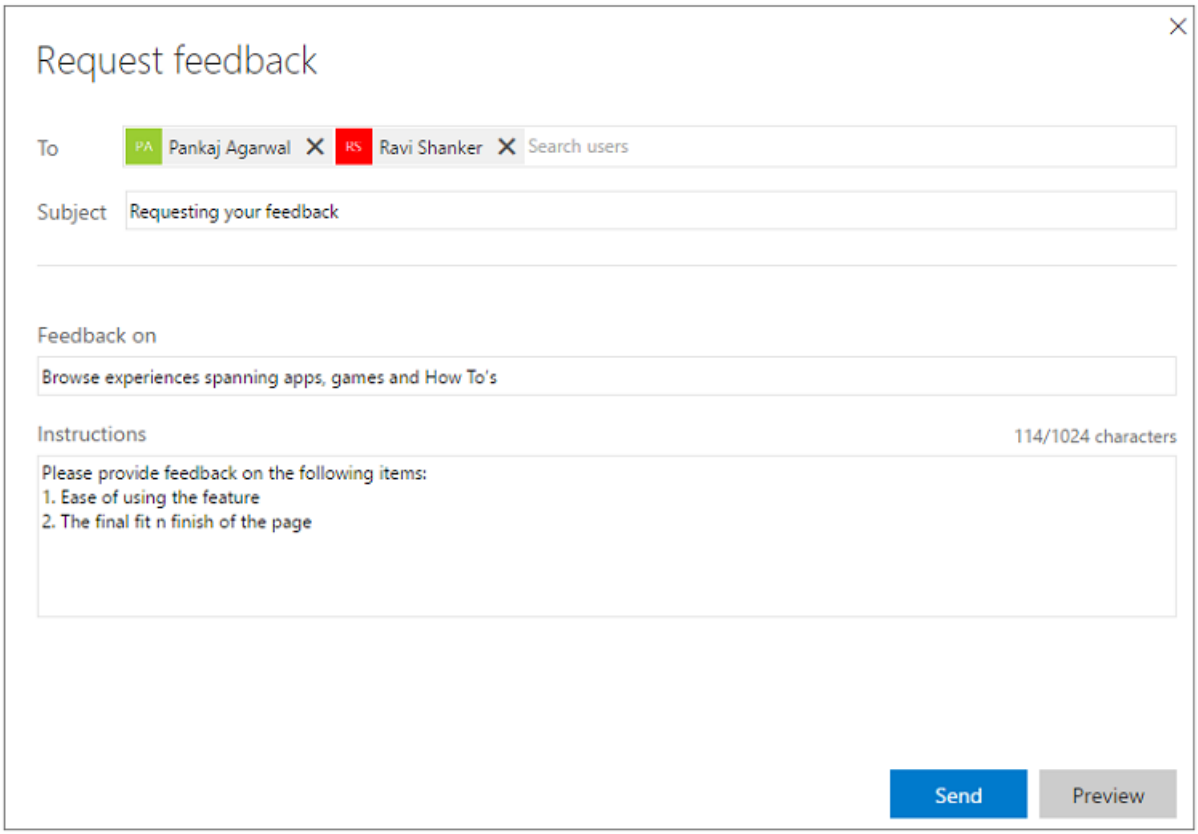


(Figure 86) View unexplored WIT

End to end stakeholder feedback flow

Request feedback

Users with basic access level can now request feedback from stakeholders directly for ongoing or completed features/stories using the Request Feedback option in the work item menu *(Figure 87)*. This opens the Request feedback form where you can choose the stakeholders you want feedback from and optionally provide a simple set of instructions prompting for the areas of the product you would like input. This will send off individual mails to the selected stakeholders along with the instructions provided, if any.

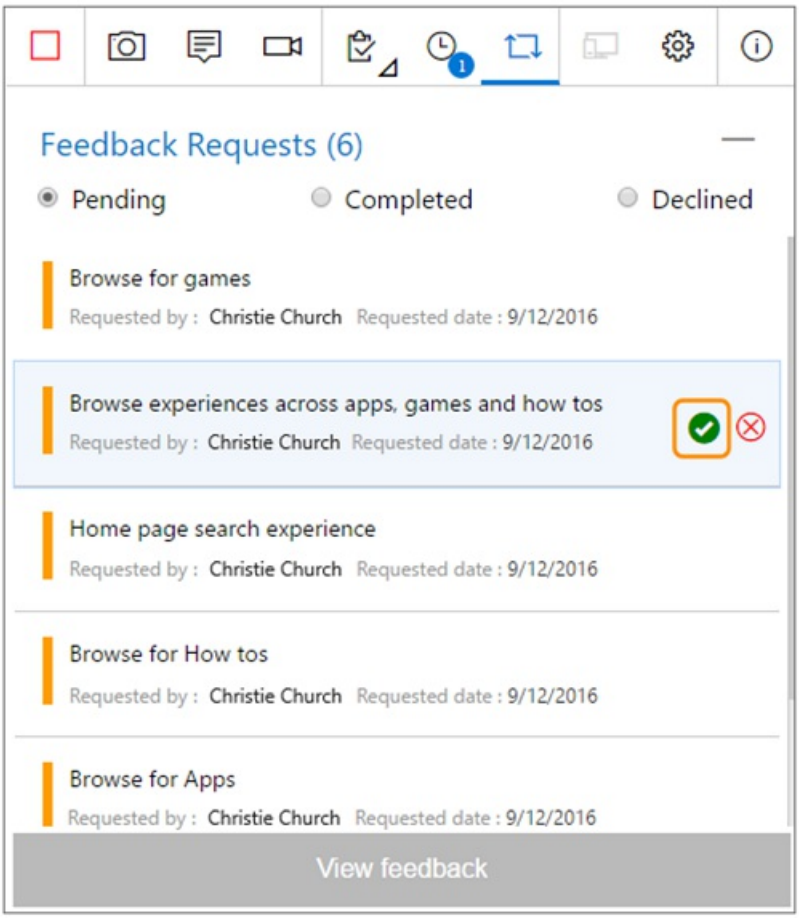


(Figure 87) XT Feedback Flow

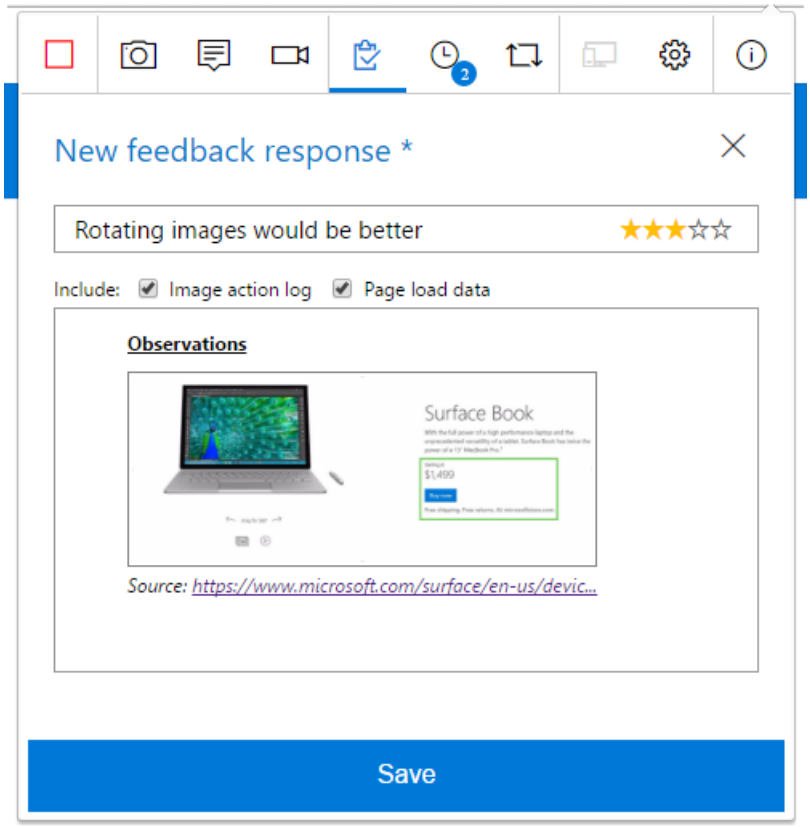
For more information, see Request stakeholder feedback using the Test & Feedback extension.

Provide feedback

Stakeholders can respond to the feedback request by clicking the Provide feedback link in the mail they received, which automatically configures the Test & Feedback extension (formerly Exploratory Testing extension) with the selected feedback request (it will prompt to install the extension, if not already installed). Stakeholders can then use the full capture capabilities of the extension to capture their findings and submit their feedback in the form of feedback response/bug/task work items. Additionally, stakeholders can navigate to the "Feedback requests" page to view in one place all feedback requests received by them. From the list, they can select the feedback request they want to provide feedback on, manage their "Pending feedback requests" *(Figure 88)* by marking them as complete or by declining them and can switch between different types of feedback requests by clicking on the desired radio button *(Figure 89)*.



(Figure 88) Provide feedback link



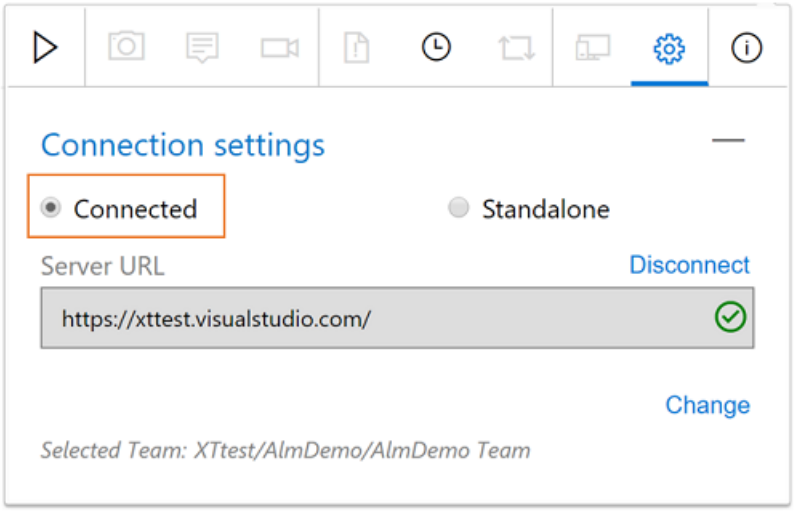
(Figure 89) XT Feedback Flow

For more information, see Provide feedback using the Test & Feedback extension.

Voluntary feedback

In addition to the solicited flow mentioned above, stakeholders can also use the extension to provide voluntary

feedback *(Figure 90)*. They can open the extension, select the "Connected" mode in the Connection settings page, and connect to the account and Project/Team to whom they wish to provide feedback. They can then use the extension to capture their findings and submit their feedback in the form of feedback response/bug/task work items.



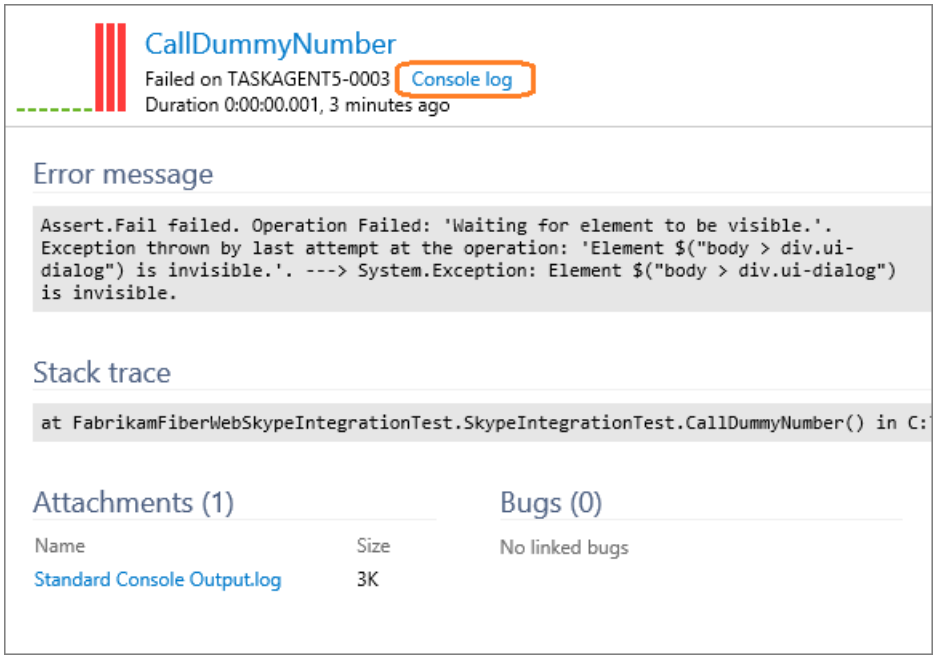
(Figure 90) Voluntary Feedback

For more information, see Provide voluntary feedback using the Test & Feedback extension.

Automated testing improvements

Console logs and test duration in tests tab in build/release summary

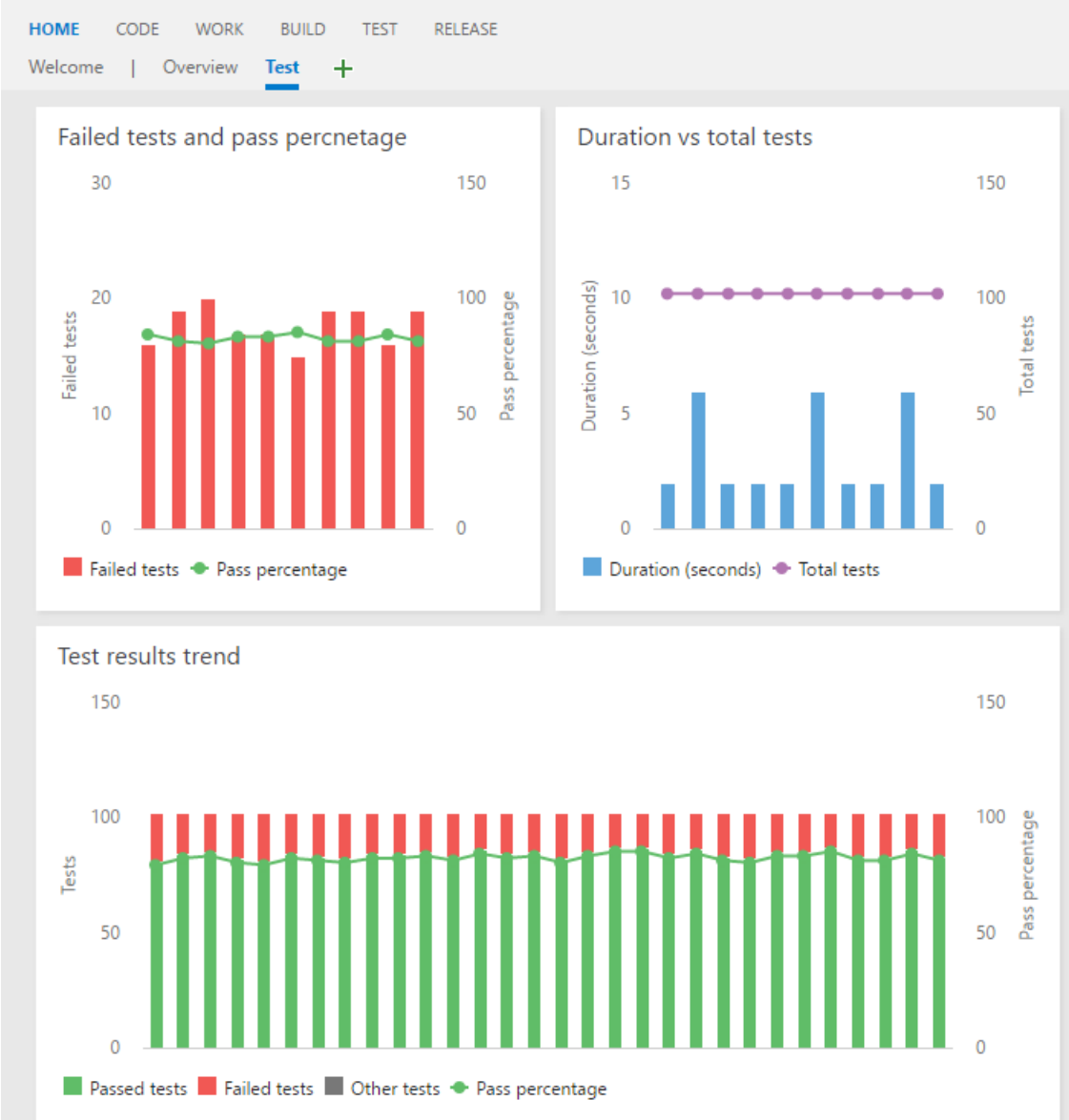
Test result console logs that are captured in .trx files are extracted and published as test result attachments *(Figure 91)*. You have an option to preview them in Tests tab, and do not need to download the trx file to view logs anymore.



(Figure 91) Console logs and duration

Test trend widget for builds

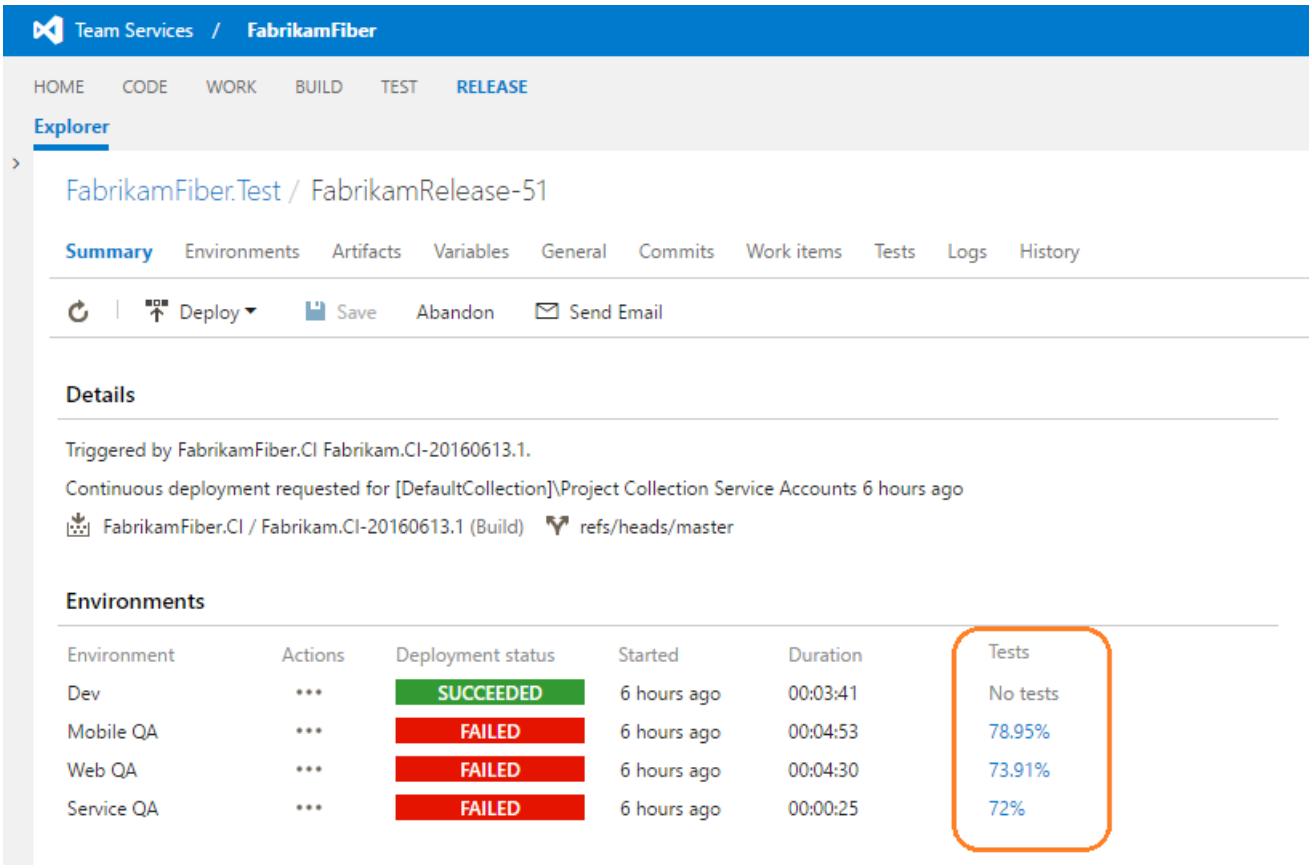
We have added a new 'Test result trend' widget to the Widget Gallery *(Figure 92)*. Use this widget to add a test result trend chart of up to 30 most recent builds for a build definition to the dashboard. Widget configuration options can help you customize the chart to include pivots like passed test count, failed test count, total test count, pass percentage, and test duration.



(Figure 92) 'Test result trend' widget

Test status with Release Environment summary

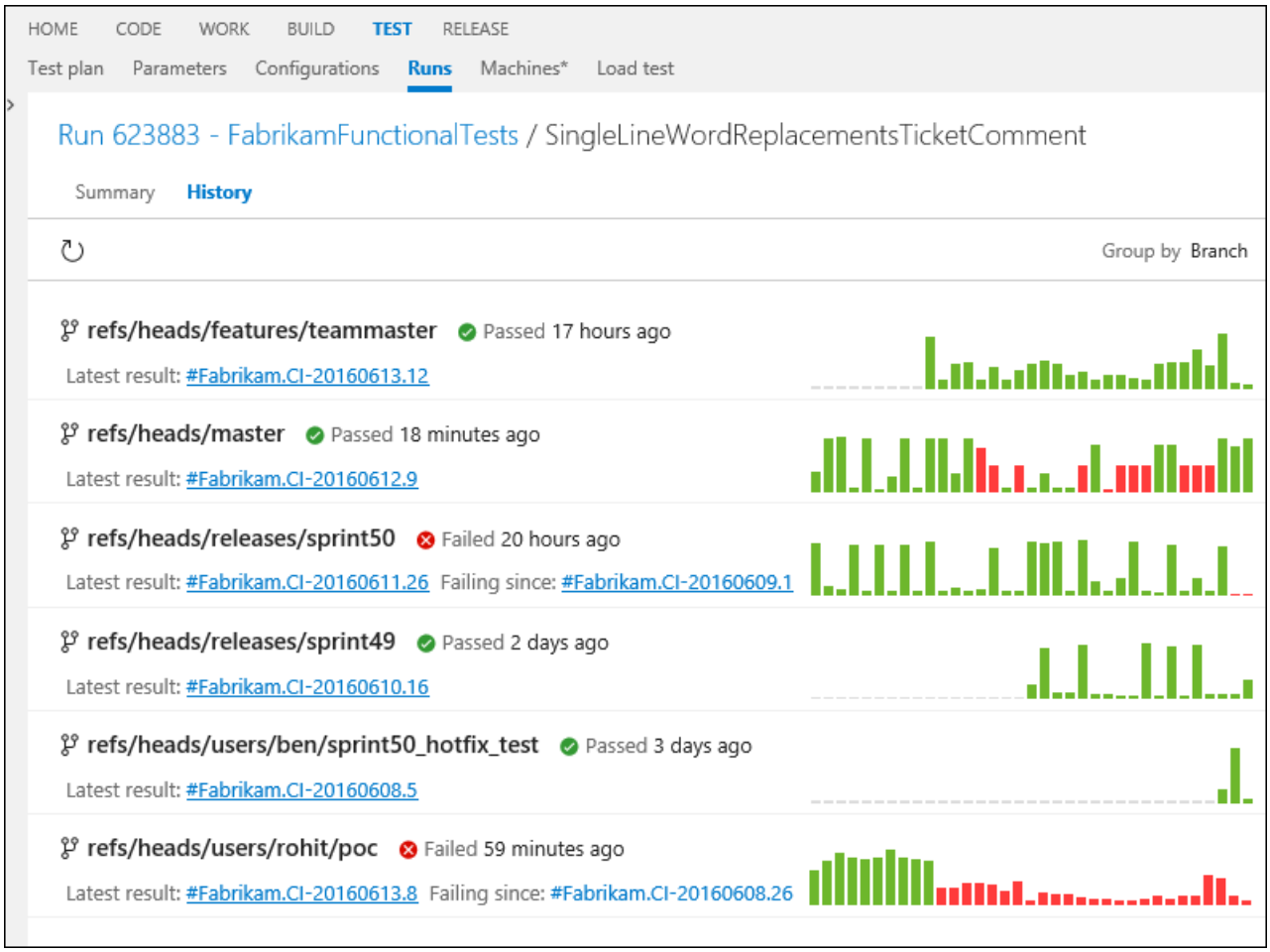
It is a recommended practice to use Release Environments to deploy applications and run tests against them. With this release, we have integrated test pass rate of Release Environments in the Environments section of the Release summary page *(Figure 93)*. As shown in the screenshot, if an Environment has failed, you can quickly infer if the failure is because of failing tests by looking at the Tests column. You can click on the pass rate to navigate to the Tests tab and investigate the failing tests for that Environment.



(Figure 93) Test status with Release Environment summary

Automated test history for branches and release environments

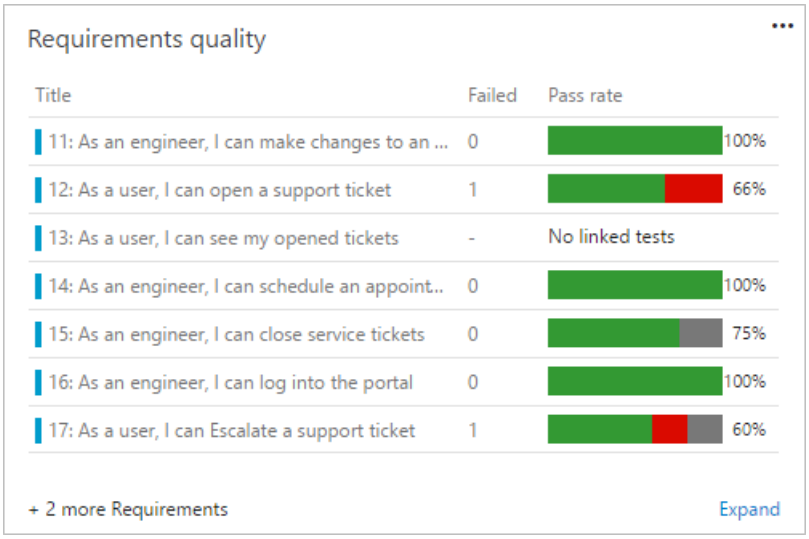
It is a common scenario for an individual test to run on multiple branches, environments, and configurations. When such a test fails, it is important to identify if the failure is contained to development branches like the master branch or if failures also impact release branches that deploy to production environments. You can now visualize the history of a test across various branches that it is testing by looking at the History tab in Result summary page *(Figure 94)*. Similarly, you group by the Environment pivot to visualize the history of a test across different Release Environments in which its run.



(Figure 94) Test status with Release Environment summary

Traceability with continuous testing

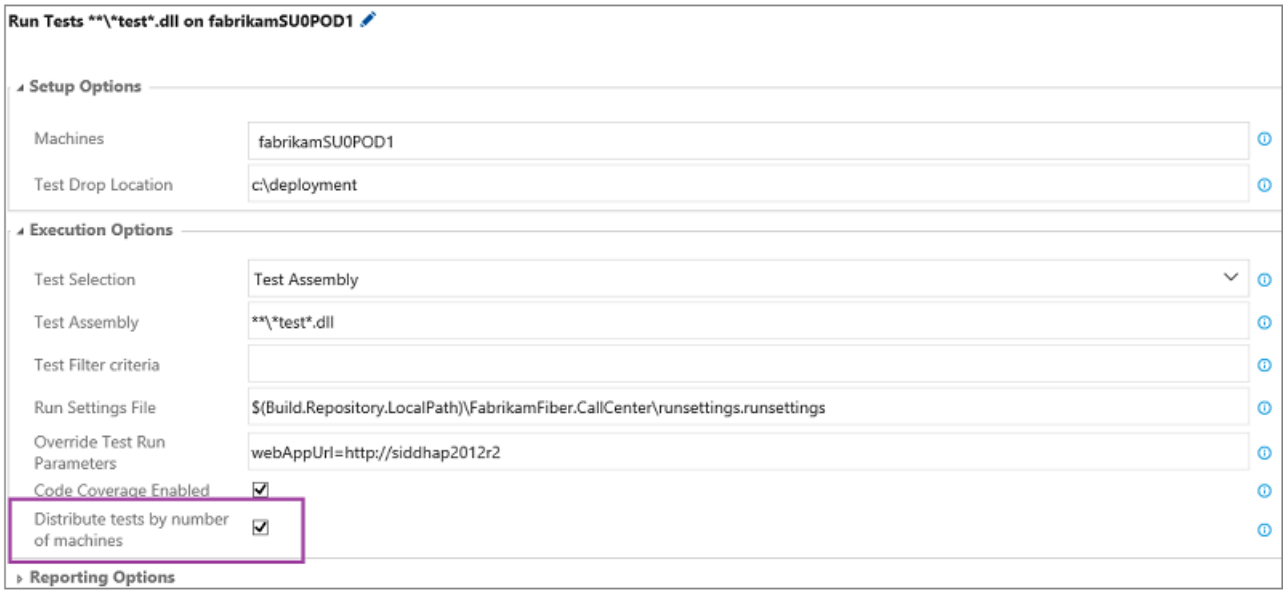
Users can now track the quality of their Requirements right on their Dashboard *(Figure 95)*. We already have a solution for Requirements quality for our Planned testing users and we are bringing it to our users who follow Continuous Testing. Users are able to link automated tests directly to Requirements and then use Dashboard widgets to track the quality of Requirements you are interested in tracking, pulling the Quality data from Build or Release.



(Figure 95) Requirement Quality Widget

Remote testing – distribute tests based on number of machines

We have enabled tests from within an assembly to be distributed to remote machines using the Run Functional Tests task *(Figure 96)*. In TFS 2015, you could distribute tests only at the assembly level. This is enabled using the check box in the task as below.



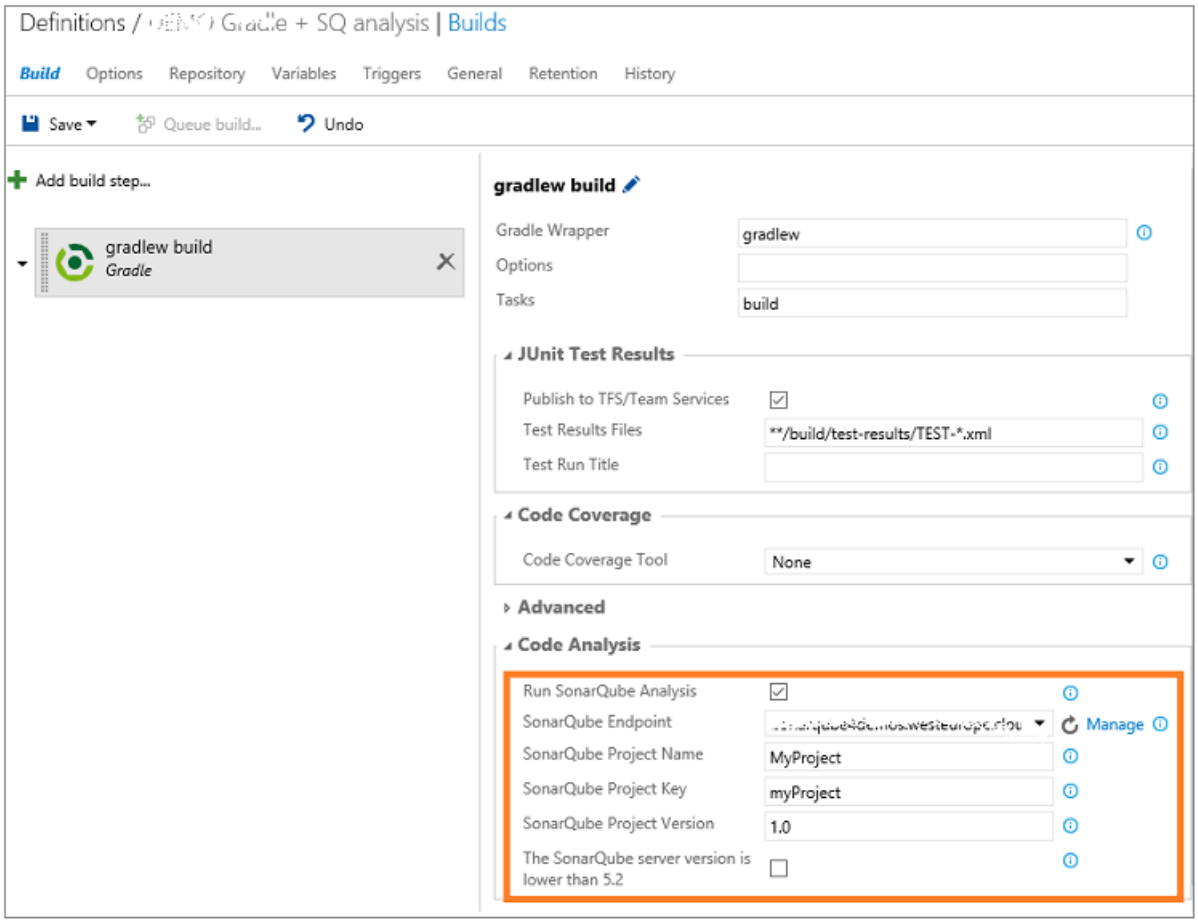
(Figure 96) Task Setting

Automated testing for SCVMM and VMWare

Users can dynamically set up test machines in the cloud with Azure, or on premises using SCVMM or VMWare, and use these machines to run their tests in a distributed manner. Users can use one of the machine provisioning tasks— Azure, SCVMM, or VMWare—followed by the Run Functional Tests task to run tests.

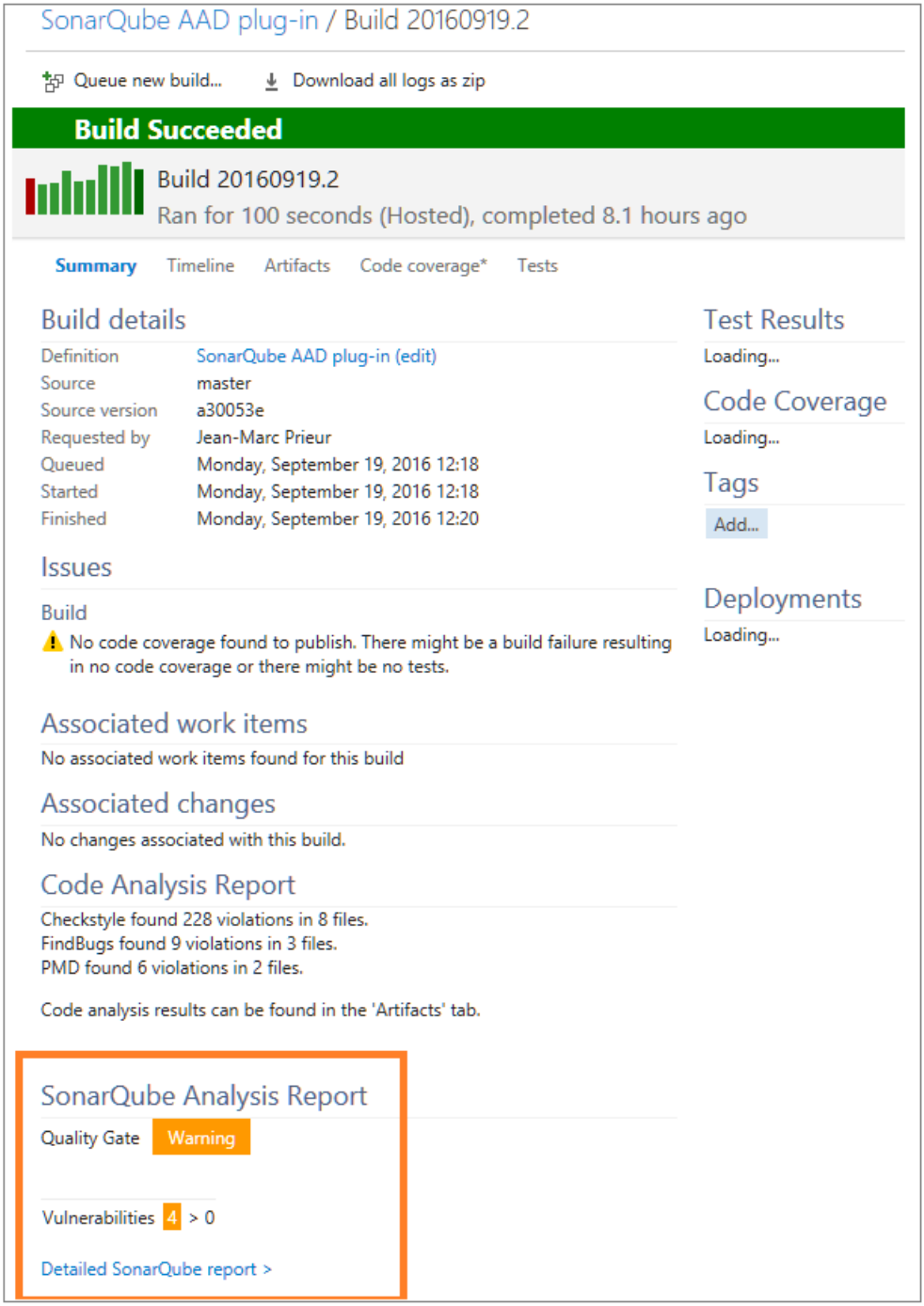
SonarQube analysis in Maven and Gradle tasks

You can now trigger a SonarQube analysis in the Maven and Gradle build task by checking 'Run SonarQube Analysis', and providing the endpoint, the SonarQube project name, the project key, and the version *(Figure 97)*.



(Figure 97) Run SonarQube Analysis

You will also now get a link on the SonarQube project *(Figure 98)*. You can request a full analysis to see the quality gates details, and choose to break the build if they are not met.

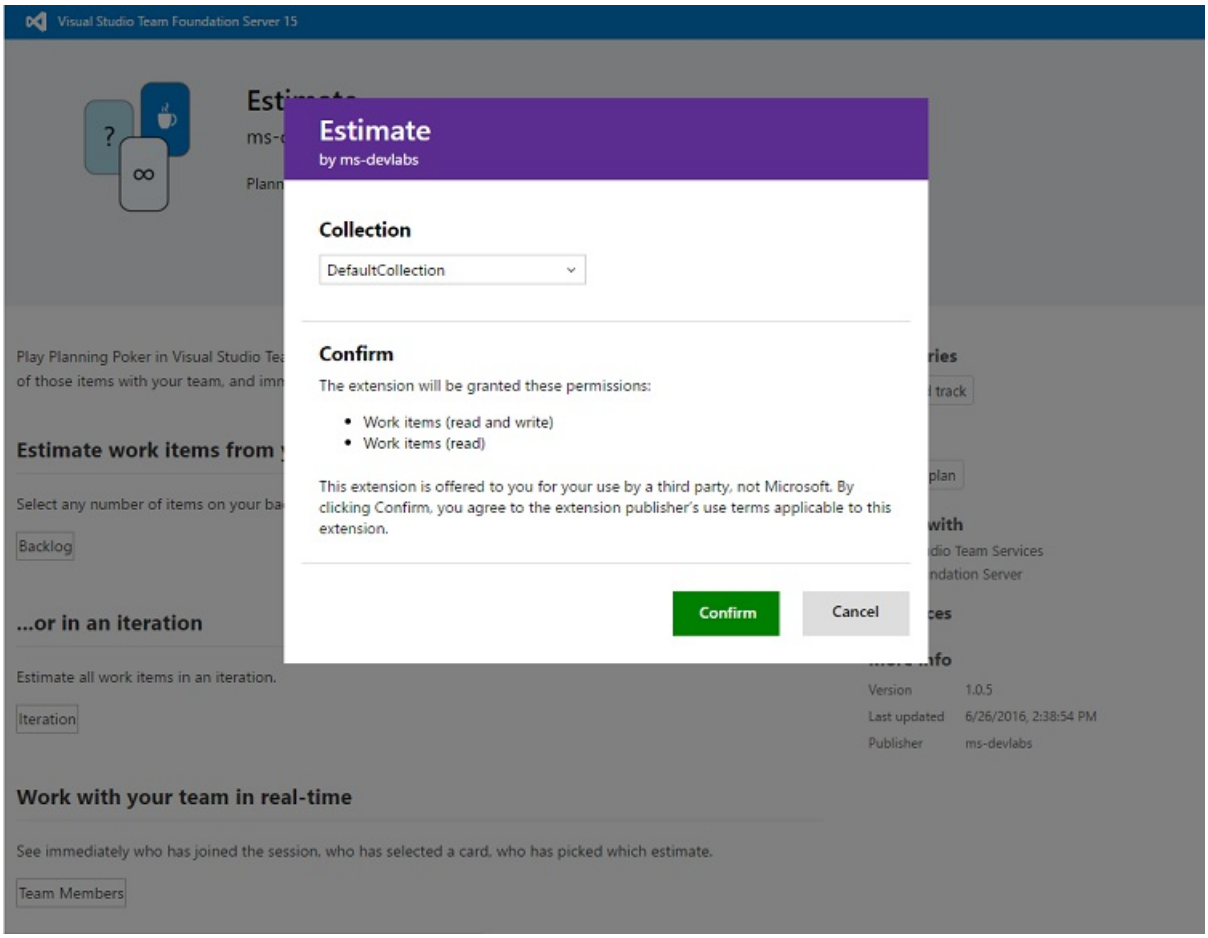


(Figure 98) Run SonarQube Analysis

For more information, please see The Gradle build task now supports SonarQube analysis.

Marketplace Improvements

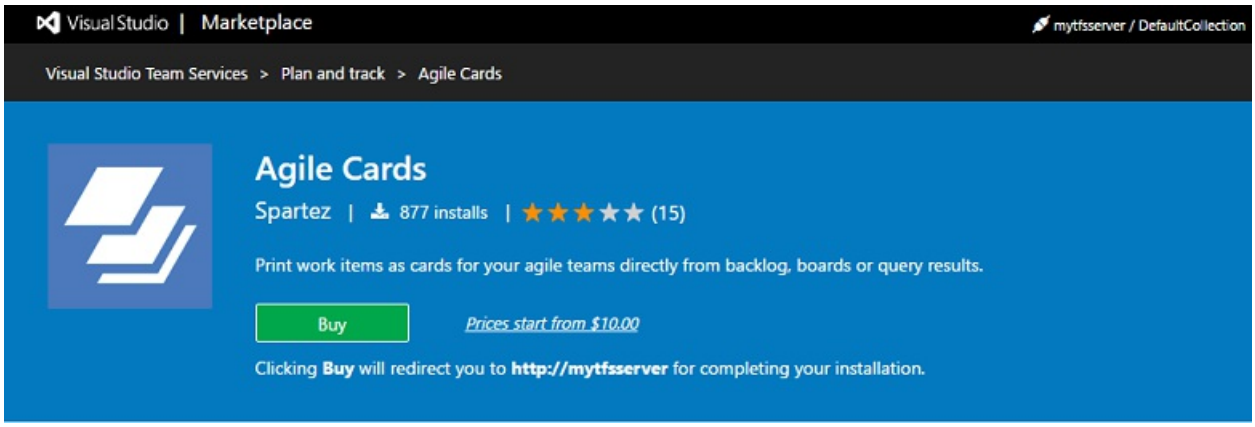
Project collection administrators can now browse to the Visual Studio Marketplace from a Team Foundation Server and install free extensions in a team project collection. The extensions are automatically downloaded from the Visual Studio Marketplace, uploaded to the Team Foundation Server, and installed in the selected team project collection *(Figure 99)*.



(Figure 99) Install Free Extension

Purchase and install paid extensions

Project collection administrators can now browse to the Visual Studio Marketplace from a Team Foundation Server, buy paid extensions, and install them in a selected team project collection *(Figure 100)*. The administrator can pay for extensions with an Azure subscription and select the number of users to assign these extensions. These extensions are automatically downloaded from the Visual Studio Marketplace, uploaded to the Team Foundation Server, and installed in the selected team project collection.



(Figure 100) Purchase Paid Extension

For more details, see Get extensions for Team Foundation Server documentation.

Administration Improvements

Windows Authentication

In previous releases, you needed to decide between NTLM and Negotiate security support providers for Windows Authentication when configuring a domain-joined TFS deployment. In 2017, we removed this setting from the configuration experience. If you want to continue using NTLM authentication in 2017, you do not need to take any

action. If you had been using Kerberos authentication and want to continue doing so in 2017, you do not need to take any action. TFS 2017 now always configures both the Negotiate and NTLM security support providers, in that order. With this configuration, Kerberos authentication is used where possible, providing enhanced security. When Kerberos cannot be used, NTLM authentication is used. We did extensive testing to ensure that there would not be any impact on existing TFS deployments using NTLM authentication due to this change.

A Modern navigation experience

In this release, we are enabling a new and improved top navigation bar. There are two core goals for the new nav:

- Increase navigation efficiency across product areas by quickly allowing you access any of the hubs with one click.
- Bring a modern visual aesthetics and user experience to the product.

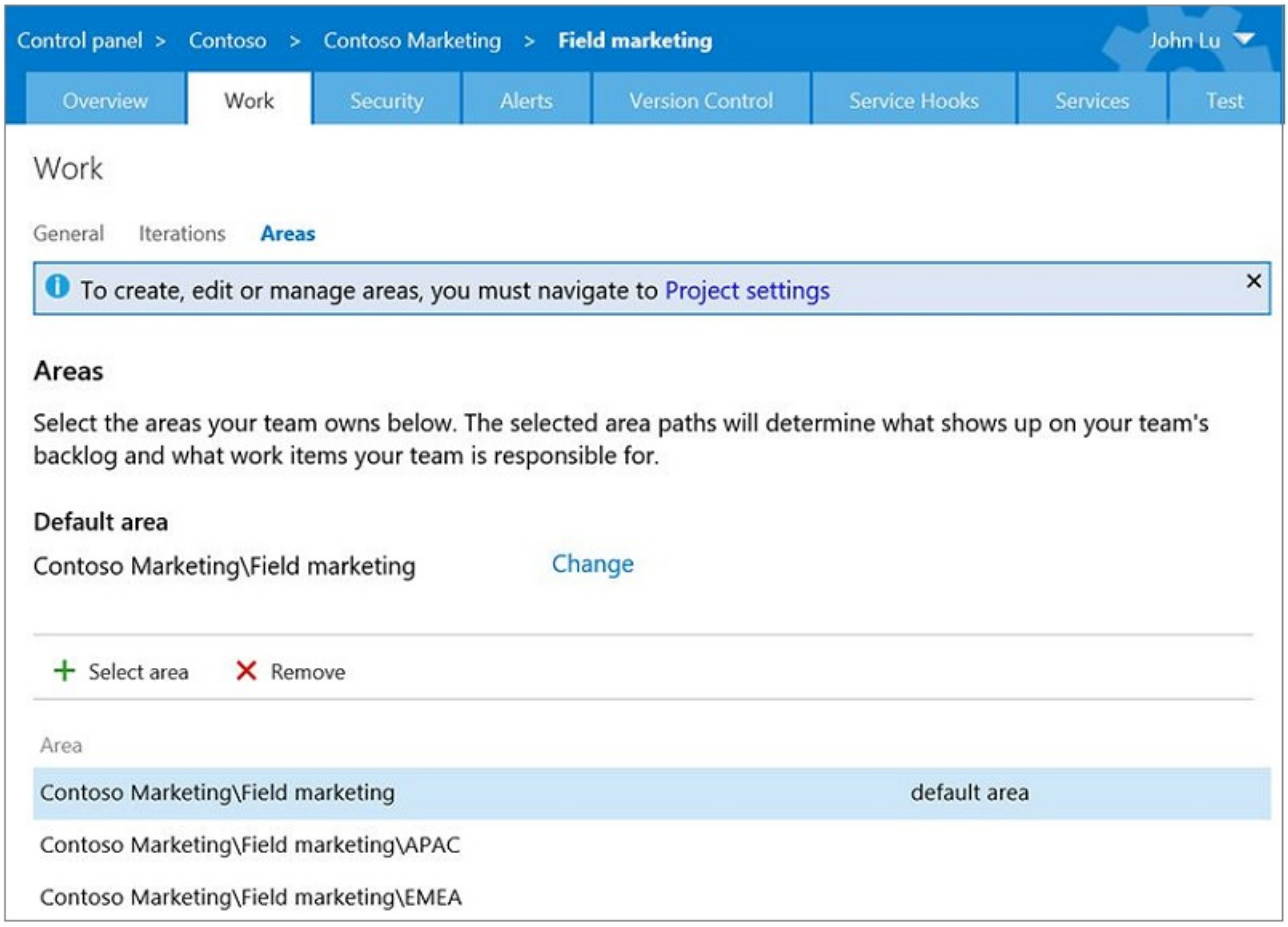
Since this is a big change for our users, and the feature is still being iterated on, we decided to have the new navigation UX off by default. If you want to play with it, you can enable it by going to the Team Foundation Server admin area Control Panel and choosing to "Turn on new navigation". Please note that it enables it for all users in the server.

Team project rename permission

The permission controlling which users can rename a team project has changed. Previously, users with Edit project-level information permission for a team project could rename it. Now users can be granted or denied the ability to rename a team project through the new Rename team project permission.

Admin settings Work hub

We have introduced a new "Work" hub in the Admin settings page that combines general settings *(Figure 101)*, Iterations, and Areas in a single tab. With this change, users will see clear differences between project-level settings and team settings. For team settings, users will only see areas and iterations that are relevant to their team. At a project level, the settings page will enable admins to manage areas and iterations for the entire project. Additionally, for project area paths, a new column called "Teams" has been added to make it convenient for admins to tell quickly and easily which teams have selected a specific area path.



(Figure 101) Admin work hub

Process configuration REST APIs

This public API allows users to get the process configuration of a given project. The process configuration contains the following settings:

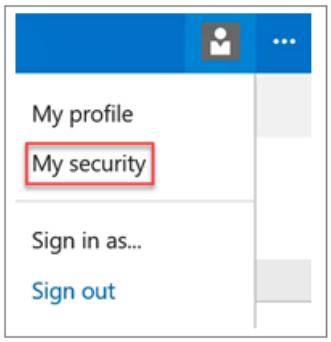
- **TypeFields:** abstractions of customizable fields that are used in the agile tooling. For example, the type of the "Story points" field is "Effort".
- **Backlog definitions:** define what work item types are on each of the backlogs. This is a frequently requested API from customers building extensions. With this data, an extension can know how to leverage process-specific fields to enable common scenarios in the agile tools (such as changing the activity or effort of a work item, knowing what work items are included at a given backlog level, or determining whether teams are identified by area path or a custom field). Please refer to Work Overview for more information.

New admin experience with prefix based AD search

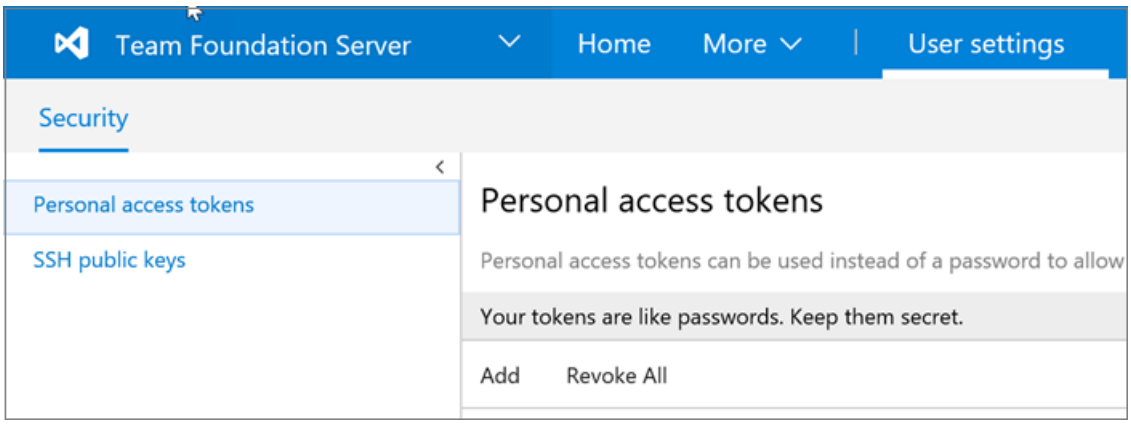
Team Foundation Server 2017 introduces a new experience to manage groups and group membership. You can search in active directory or local machine users/groups using prefix based search criteria on user/group name(s). For example, 'John D' as well as samaccountname (e.g. 'businessdomain\johbdnd') and see the contact card of a user/group.

User security settings

You can manage your personal access tokens and SSH in the new "My Security" experience *(Figure 102)*. Users who were using "My Profile" to manage SSH will now need to manage their SSH public keys in the user security settings *(Figure 103)*.



(Figure 102) My security



(Figure 103) My profile

Unified configuration wizard

In previous releases, you would pick one of multiple configuration wizards for your TFS deployment depending on what you were trying to do. The Basic and Full wizards could be used to configure a new deployment; the Upgrade wizard could be used for production and pre-production upgrades; and the Application-Tier Only wizard could be used for a variety of scenarios, including scaling out an existing deployment, replacing an application tier with new

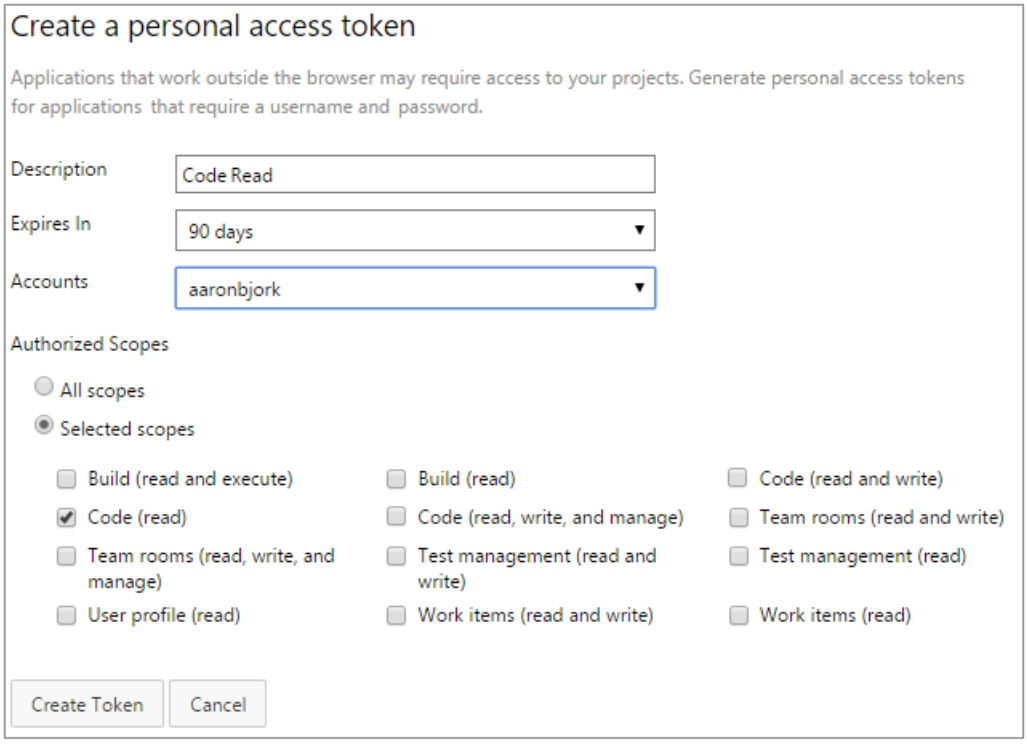
hardware, and so forth. In TFS 2017, all of these scenarios have been unified into a single Server Configuration Wizard, which guides you toward and then through each of these scenarios by asking you to make simple choices. Additionally, advanced configurations like pre-production upgrades and clone existing deployment now automate actions that used to be done through tfsconfig.exe, including changing server IDs, remapping database connection strings, and removing references to external dependencies (which used to be done with tfsconfig.exe PrepareClone).

New access level

With the new Visual Studio Enterprise group added to the Access Level admin portal in Team Foundation Servers, you can now quickly identify who has a Visual Studio Enterprise subscription. Once identified, these users will gain full access to all first party TFS extensions installed from the Visual Studio Marketplace at no additional charge.

Personal Access Tokens

You can now connect to any Team Foundation Server using a personal access token in addition to SSH *(Figure 104)*. This is helpful if you develop on Linux or Mac and would like to use in any automation tools and GIT. You can manage your personal access tokens from the user security settings page.



(Figure 104) Personal Access Tokens

Known Issues

This is a complete list of known issues in this release.

There are no Power Tools for Team Foundation Server 2017

- **Issue:**

  No Power Tools have been released for TFS 2017.

- **Workaround:**

  We are excited to let you know that most of the previous Power Tools have been integrated into TFS 2017. Unfortunately, the Process Template Editor has not been integrated, but you can get it in the Visual Studio Marketplace.

Updating custom control extensions

- **Issue:**

The schema for fields on the work item form has changed. The documentation for custom control extensions also changed.

- **Workaround:**

  See the new documentation: Add a custom control to the work item form.

Error when importing work item type definition

- **Issue:**

  Customers that have a work item page extension installed, who export a work item type definition then import that definition, will get the following error: "The 'LayoutMode' attribute is not declared".

- **Workaround:**

  There is an extra LayoutMode attribute on the PageContribution element each time you export a work item type definition. Before importing the definition, search for the PageContribution mode and remove the LayoutMode attribute. For example, remove LayoutMode="FirstColumnWide".

Customers should update to Git LFS version 1.3.1 or higher

- **Issue:**

  Git LFS versions before 1.3.1 will not be supported in future releases.

- **Workaround:**

  Customers using Git LFS are strongly encouraged to update to Git LFS version 1.3.1 or higher. Older versions of the LFS client are not compatible with authentication changes in this version of TFS. In order to give customers time to migrate, we implemented a short-term workaround for RTW. The workaround will be removed in Update 1, at which point Git LFS clients below 1.3.1 will no longer work.

NuGet Restore is not finding packages that exist in nuget.org

- **Issue:**

  When using NuGet 3.4.3 or greater, the NuGet Restore task does not restore packages from NuGet.org unless it is an explicit source in the NuGet.Config.

- **Workaround:**

  Ensure NuGet.org is in NuGet.Config.

  <packageSources> <add key="nuget.org" value="https://api.nuget.org/v3/index.json" protocolVersion="3">
  </packageSources>

NuGet build and release tasks do not authenticate

- **Issue:**

  When using Team Foundation Server / Package Management, NuGet build and release tasks will not authenticate to feeds if the agent is running as a NETWORK SERVICE user, which is the default when the build agent runs as a service. This happens because versions of NuGet before 3.5 use the credentials of the user account running the build agent, not the credentials provided by the build task.

- **Workaround:**

  To use NuGet build/release tasks with TFS feeds using an agent that is running as a NETWORK SERVICE, you must use NuGet 3.5 or higher.

NuGet build and release tasks use agent's credentials

- **Issue:**

  Versions of NuGet before 3.5 use the credentials of the user account running the build agent, not the credentials provided by the build task. This may result in unexpected access or lack of access to feeds.

- **Workaround:**

Use NuGet 3.5 or higher on TFS build agents.

External extensions do not automatically upgrade when upgrading TFS

- **Issue:**

  If you downloaded an extension from the Visual Studio Marketplace, published it to your TFS 2015 installation, and then upgraded to TFS 2017, the extension is not automatically updated when new versions of the extension are published to the Marketplace.

- **Workaround:**

  After upgrading to TFS 2017, uninstall the extensions you had installed in TFS 2015. Then reinstall the latest extensions. In TFS 2017 we added a feature to automatically check for updated external extensions once a day and upgrade them.

The Jenkins Queue Job task cannot be run in release definitions

- **Issue:**

  When running the Jenkins Queue Job task in a release definition, customers get a 500 server error.

- **Workaround:**

  Currently, the Jenkins Queue Job task can be run as part of TFS build definitions, but not release definitions. This ability will be added in a future release.

Custom TFS server plugins need to be rebuilt against TFS 2017 DLLs

- **Issue:**

  Custom TFS server plugins do not work after upgrading to TFS 2017.

- **Workaround:**

  Rebuild your custom server plugins against the TFS 2017 assemblies.

The Server Object Model for Custom TFS server plugins has changed since TFS 2015 RTM

- **Issue:**

  Custom TFS server plugins do not compile.

- **Workaround:**

  Fix up source code as described in this blog post.

When using administrator actions, an exception is thrown

- **Issue:**

  In the **Alerts Administration** page, when Team Administrators use the **Find Alerts for a specific user** search to find subscriptions for a team, they might get an exception.

- **Workaround:**

  - **Option 1:** Click on the **All Alerts** node and set the **All My Teams Alerts** filter to show. This will show all alerts for all groups that the user has access to.
  - **Option 2:** In case the group is a team, instead of searching by team name, navigate to this team's **Alerts Administration** page to manage subscriptions.

Issue using tasks for running functional tests in Team Build / Release Management

- **Issue:**

  Running functional tests in Team Build / Release Management using 'Visual Studio Test Agent Deployment' and 'Run Functional Tests' tasks from the task catalog currently uses Agents for Visual Studio 2015 Update 3 and can only be used to run tests built using Visual Studio 2013 and Visual Studio 2015. These tasks cannot be used for running tests built using Visual Studio 2017 RC. For more details, please refer to this blog post.

- **Workaround:**

  There is no workaround. Support for using Test Agent 2017 and running tests built using Visual Studio 2017 will be added in the TFS 2017 Update 1 timeframe.

Extensions are not being auto-updated

- **Issue:**

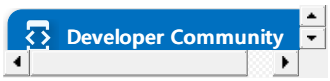
  If you upgrade a prior version of TFS to reach TFS 2017 and are running TFS 2017 in connected mode then your extensions will not be auto-updated as they should be.

- **Workaround:**

  There is no workaround at this time. We have fixed the issue and the auto update behavior will reach you through TFS 2017 Update 2. If for any reason you cannot wait for Update 2 then reach us through the Support channel and we shall share the fix earlier.

If you encounter issues that are preventing you from deploying in a production environment (Go-Live), please contact Microsoft product support. (English only) U.S. business hours only (M-F 6a-6p PST), 1 business day response.

See customer-reported issues reported for Team Foundation Server 2017.



Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2017 Release Notes History

9/7/2018 • 2 minutes to read

Release Notes History

- Team Foundation Server 2017 Update 3 Release Notes
- Team Foundation Server 2017 Update 2 Release Notes
- Team Foundation Server 2017 Update 1 Release Notes
- Team Foundation Server 2017 Release Notes

Team Foundation Server 2015 Update 4 Release Notes

9/7/2018 • 4 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

This is not the latest version of Team Foundation Server. To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2015 Update 4. Click the button to download.

Team Foundation Server 2015

To learn more about Team Foundation Server 2015, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: February 28, 2018

Details of What's New in Team Foundation Server 2015 Update 4.1

This update fixes potential cross site scripting (XSS) and other security vulnerabilities in Team Foundation Server 2015 Update 4. See the blog post for more information. It is a full upgrade, so you can upgrade directly to TFS 2015 Update 4.1.

Release Date: April 11, 2017

Summary of What's New in Team Foundation Server 2015 Update 4

- Agile Bug Fixes
- Version Control Bug Fixes
- Build Bug Fixes
- Release Management Bug Fixes
- Testing Bug Fixes
- Administration Bug Fixes
- Marketplace Changes

Details of What's New in Team Foundation Server 2015 Update 4

Agile Bug Fixes

- The @Today and @Me macros do not work correctly in non-English in the Kanban board card style rules.

- The inline **add card** experience on the Kanban board does not work correctly. For example, the Title field cannot be edited.
- If a user switches between two work items of the same type on the queries page before the HTML fields finish loading, the HTML field may become empty and the work item will become dirty.
- The Batch API, such as WorkItemStore.GetWorkItemIdsForArtifactUris(), may return incorrect results when called with many strings.
- When a customer has rules in the global workflow and tries to move them to a work item type definition, the following error occurs, "TF237090: Does not exist or access is denied".
- If a TFS instance has a collection with a space in the name and has a public URL that is different from the internal URL, inline images may be missing in work items when opened by another user.
- Work item tracking warehouse sync fails with a name conflict when field names only differ by a space replacing a "." or "_" (i.e. "My Field" and "My_Field").
- Work item tracking warehouse sync fails when a work item has a link comment that contains special characters, such as 0x0B.

Version Control Bug Fixes

- Destroying a very large team project or a very large TFVC source tree will time out and rollback.
- Renaming branch objects across projects may lose the parent relationships.

Build Bug Fixes

- The first check-in after configuring a Gated Check-in trigger for a build definition fails.
- The error, "An item with the same key has already been added", is shown while loading build tasks or queuing builds.
- The Windows build agent cannot build from Subversion repositories when running on 32-bit Windows.
- Build tasks are not updated when the extension is updated.

Release Management Bug Fixes

- In a release environment, if the **All users in sequential order** option is chosen and the approver order is changed, the definition is not marked dirty and cannot be saved.

Testing Bug Fixes

- Users are unable to deploy a standalone test agent.
- When selecting a test plan, the source filter is null.
- When you mark a test case as paused then save and close the test runner, you cannot continue the test case when you return to the test.
- **New for RTW**: When link between a bug and a test result is deleted, the deletion date does not get updated in the warehouse.
- **New for RTW**: Customers who do not have permissions on the default area path will get an error "Sequence contains not elements" when viewing a build's test results.
- **New for RTW**: In the MTM tool, customers get an error "Failed to initiate clone operation" when trying to clone a test plan.
- **New for RTW**: Extensions are not able to access **Test** hub REST APIs.

Administration Bug Fixes

- The admin console (TfsMgmt) may close unexpectedly during an upgrade.
- Code reviewers in non-English versions of TFS do not get an email when added to code reviews.
- The upgrade to TFS 2015 may fail with duplicate workspace names if an orphaned user has a workspace with same name as valid user.
- **New for RTW**: There can be performance problems when receiving notifications.
- **New for RTW**: The warehouse connection string points to the configuration database, instead of the warehouse database, after an upgrade, if the databases are on different SQL instances.
- In previous releases, customers using SSL offloading needed to add the X-Forwarded-Proto header on their

load balancer. With this update, they can simply configure the public URL in TFSMgmt.exe to generate https:// URLs.

- The Jenkins Service Hook was incompatible with newer versions of Jenkins because of a new authentication pattern. With this release, newer versions of Jenkins are compatible.

Marketplace Bug Fixes

- Installations of MS.TFS.Server are now supported.
- Paid Preview extensions are now supported on TFS 2015 through this update. Once installed they become free forever since out-of-box Commerce integration for extensions does not exist for TFS 2015. Additionally, any flag in the extension manifest not understood by the system is ignored and install will not be blocked.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2015 Update 3 Release Notes

9/7/2018 • 11 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

NOTE

This is not the latest version of Team Foundation Server. To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2015 Update 3. Click the button to download.

Team Foundation Server 2015

To learn more about Team Foundation Server 2015, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: June 27, 2016

Summary of What's New in Team Foundation Server 2015 Update 3

- SSH Support for Git repos
- Dashboard Widget SDK
- Testing - New features & Bug Fixes
- Agile Bug Fixes
- Build Bug Fixes
- Version Control Bug Fixes
- Administration Bug Fixes
- Extensibility Bug Fixes
- Release Management

Details of What's New in Team Foundation Server 2015 Update 3

SSH Support for Git Repos

With TFS 2015 Update 3, you can now connect to any Team Foundation Server Git repo using an SSH key. This is very helpful if you develop on Linux or Mac. Just upload your personal SSH key and you're ready to go.

Dashboard Widget SDK

In Update 3, not only can you use the out-of-the box dashboard widgets, you can also create your own widgets by using the SDK. For more information, see the Add a dashboard widget page on VisualStudio.com.

Testing - New Features & Bug Fixes

Testing - New Features - Support for Azure, SCVMM and VMWare

You can now dynamically set up test machines in the cloud with Azure, or on-premises using SCVMM or VMWare and use these machines to run tests in a distributed manner. You can use one of the machine provisioning tasks - Azure, SCVMM or VMWare followed by the Run Functional Tests task to run tests. For more information, please see the Install and configure test agents page.

Testing - Bug Fixes

Bugs reported through Connect:

- Test settings file is ignored when "Run in Parallel" is selected.
- TEMP folder is not cleaned after Test Agent Deployment is completed.
- Source filter string is required even with Test Selection set to Test Plan. User gets error "Cannot bind argument to parameter 'SourceFilter' because it is an empty string" if string is empty.
- Email/print test artifacts feature hangs and throws JavaScript TypeError.
- Web test runner window no longer wraps text.

Other bug fixes:

- "DistributedTests: Exception occurred while parsing buildId" is thrown in Release.
- Remote Test Execution gets aborted abruptly with error - Access to the path is denied.
- Test results cannot be uploaded from Ant, Maven or Gradle tasks in Release.
- VsTest task fails if full path of 2 DLLs are given separated by semicolon.
- No Test results are shown in Release when results are grouped by 'Test Suite' and Environment selected is 'All.'
- Visual Studio Test task will not upload test results if results folder is configured in runsettings file.
- Feedback request hyperlink is incorrect in email request.
- Query based test suites do not properly reflect the tests when assigned all the test cases in this test suite to be run by multiple testers.
- Exception Microsoft.TeamFoundation.TestManagement.Server.InvalidStructurePathException: The structure path CEBIS FWK is not valid.
- Error in test hub after upgrading of TFS to 2015.1.
- MTM 2015 | 2013 - TFS 2015.2 | Analyze test runs -results , Plan tabs comes up as empty for specific users.
- MTM Screen capture file upload retries after the failure with file not found error.

Agile Bug Fixes

Bugs reported through Connect:

- Setting styles in the sprint board cards may cause an error if the locale is set to French.
- Setting styles in the sprint board cards may cause an error if the locale is set to German.
- Unable to create a query when there is a clause with an Area Path with non-standard characters, such as an underscore or single quote.
- The links label control does not show hyperlinks in web access.
- Creating new team projects causes a TF30177 "Cannot insert duplicate key row in object 'dbo.Constants" error.
- The Add Widget dialog respects the browser language over the language selected in "My Profile."
- In the Build Chart widget, the most recent bar in the chart shows green, even if the build fails.
- The Stakeholder banner is missing so users are not aware they are logged in as a stakeholder and do not have access to all features.
- Readme files do not always display on the Team Project welcome page.
- When setting a part of a time in work item tracking, the month and day values may get switched.

Other bug fixes:

- A Work Item Tracking Web Page control referencing an identity field as Param with through an error when the value is empty.

- Error when changing the name of the Query Result widget.
- The Remaining Hours input is not big enough on the card.
- Backlog will not load when the user does not have permissions to a parent work item.
- Navigating to the WORK hub after changing team projects results in a TF400483 error.
- The Dashboard Manager icon has no visual cue on focus.
- The Add Dashboard icon in Dashboard Manager has no clear visual cue on focus.
- The add and delete Dashboard buttons in Dashboard Manager do not work on pressing ENTER.
- In the Query Tile and Work Item Chart widgets, when tabbing through the configuration blade, the input will get stuck on the Query Selector with an error that no query is selected.
- When upgrading from Team Foundation 2013 Update 1 or earlier, the contents of the project homepage will not be migrated.
- When licensed as a Stakeholder, you cannot navigate between dashboards.
- In the markdown widget, if the markdown references an image in source control, it will not display.
- If a third party widget is in an error state, the entire dashboard fails to load.
- If a third party widget is in an error state, adding new widgets are added as blank.
- If a third part widget is in an error state and then removed from the dashboard, the error banner is not cleared.
- When dashboard widgets are added and conflict with one another, such as in different browser sessions at the same time, the error is not descriptive.
- Avatars do not load in the Pull Request widget.
- In the Build Chart widget, the last completed status icon is incorrect when compared to the build chart.
- When in Edit Mode of a dashboard, the error banner is covered up with the dashboard background.
- In the Visual Studio Links widget, the "Open in Visual Studio" image is plain purple.
- When making changes in the configure widget blade, there is no prompt about discarding changes when cancelling out.
- If a widget has an error, the user can still save configuration changes.
- When previewing a widget in the dashboard, it is zoomed in and blurry.
- Tabbing in the Dashboard edit mode tabs through the widget instead of the delete and configure icons.
- When in the Dashboard edit mode, ESC should exit out of edit mode.
- When creating a new Work Item Chart widget in Firefox, the chart types are of varying sizes.
- In the Work Item Chart widget, the chart options are not displayed until a query is selected.
- In the Sprint Overview widget, setting the iteration dates does not refresh the widget.
- In the Sprint Burndown widget, tabbing to the graph and hitting Enter does not open the lightbox.
- In the Conditional Query Tile, the input field for a rule allows a five digit number but only displays four digits.

Build Bug Fixes

Bugs reported through Connect:

- Unable to filter builds by tags on Firefox.
- When setting the permissions of a user on a build, there is an error when saving.
- If a build is scheduled to run in the late evening, it runs on the previous day.
- Build fails with "TF14044: Access Denied: User Project Collection Build Service needs the AdminWorkspaces global permission(s).".
- The time formatting from My Profile is not used in the Build hub.
- Build fails with "curl was not found in the path" error when running a curl task in Build.
- Gated build gives an error of "Shelveset not found."
- There are formatting problems when creating a new build definition in Chrome.
- When a XAML Build has a large number of warnings, it shows an error of "An undefined error occurred while attempting to connect to the server. Status code 0.".

- When resizing the Reason column in the Build page, the entire icon array is displayed.
- Build fails with "Invalid solution configuration and platform.".
- When including an npm (Node.js package manager) install task, builds fail with an error that it cannot find the npm install.
- Error of "Invalid source label format" when editing a build definition that labels a Git repository with a build number.
- Continuous Integration does not always trigger when using an external Git repository.
- On upgraded project collections, gated checkins fail due it using the build account instead of service account.

Other bug fixes:

- getBuildBadge vso-node-api fails if using a PAT without the "All Scopes" permission.
- If a build definition name contains square brackets, the revision number is not calculated correctly.
- When splitting a Team Project Collection, there are duplicate build service identities.
- When entering a shelveset name when queuing a new build, you get a misleading error of "There are issues with the request or definition that will prevent the build from running: The value specified for SourceVersion is not a valid version spec.".
- Extensions with cross platform build tasks do not work.
- Build fails to connect to Subversion when using SSL port 8443.
- When using an SVN repository for a build that does not have mappings, the Source Version is not set.
- Cannot queue a Team Foundation Version Control build from a source label.

Version Control Bug Fixes

> **NOTE**
> These are bug fixes for Version Control in Team Foundation Server. For Version Control fixes in Visual Studio, see the Visual Studio Release Notes.

Bugs reported through Connect:

- When using Git LFS, there may be problems with functions such as cloning the repo.
- There are hourly Git pull request event log errors of "TF53010: The following error has occurred in a Team Foundation component or extension.".

Other bug fixes:

- Adding a Latest Version link type to a work item does not work.
- The Team Foundation Version Control warehouse adapter fails after upgrading from Team Foundation Server 2010.
- There is a limit of 25 commits when linking to work items during pull request creation.
- If a repository has multiple build definitions configured, the Build Explorer may show one definition's name but link to the last build on another.
- In Pull Requests, the identity picker is cut off on the right side.
- Team Foundation Version Control files show that there is an encoding change even if there was no change.
- On a Git push over SSH, there is an error "TF401030: The Git pack header is invalid.".

Administration Bug Fixes

Bugs reported through Connect:

- When splitting a team project collection, after cloning the collection and deleting a team project in the first collection, the other collection may not show the project that was deleted in the other collection. The direct URL will work, but the user cannot browse to the team project.

Other bug fixes:

- When upgrading, the readiness check may fail with errors that Port 8080 is unavailable and "TF401147: The previously configured ports for the Application Tier Web Service site are currently in use.".
- In the Admin Console, the Proxy Server URL is blank.
- When configuring TFS, the port and vdir may incorrectly fall back to the default mappings.
- The Admin Console may crash when loading the Collections tab.

Extensibility Bug Fixes

Bugs reported through Connect:

- "TF400367: The request could not be performed due to a host type mismatch" error when omitting the collection in the URL when using the TFS SDKs.
- Deleting a branch triggers a build when using Jenkins service hooks.
- When clicking Manage Events in a team room, there is an error "Invalid Navigation Level".
- When working with Alerts, fields may have unexpected allowed values.
- Emails are not always received for alerts.
- Alerts for team projects with spaces in the name include invalid links.
- There is no link to All Alerts in the Alerts administration page.
- In the Chinese version of TFS, there is no Slack option in service hooks.

Release Management

We have fixed some of the reported issues in the web-based version of Release Management. Here are some of the key issues that were fixed:

- Undefined error is shown while browsing the Release hub, when network is flaky.
- Downloading server drop artifact creates additional file under Build artifacts directory.
- Duplicate service endpoints are created from endpoint creation dialog.
- Nuget Installer task fails with Release Management.
- Auto-refresh: Pending approvals yellow bar is not displayed after starting deployment on an environment.
- Email option in approvals is not enabled if there are multiple approvers for an environment.

We have also fixed a few reported bugs in the WPF version of Release Management.

- When there is an api-version mismatch, releasemanagementbuild.exe should show proper error message instead of 403 error.
- Intermittent network failures when copying files to Deployer.

Known Issues

For a complete description of known issues in this release, see the following MSDN article: Known Issues in Team Foundation Server Update 3.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2015 Update 2 Release Notes

9/7/2018 • 15 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

> **NOTE**
> **This is not the latest version of Team Foundation Server.** To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2015 Update 2.

> **IMPORTANT**
> Team Foundation Server 2015 Update 2.1 has had some bug fixes since Update 2.0. Please refer to KB3160281 for the list of fixes. If needed, for configurations such as multiple application tiers that require matched versions, you can still download Team Foundation Server 2015 Update 2.0.

To learn more about Team Foundation Server 2015, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: May 5, 2016

Summary of What's New in Team Foundation Server 2015 Update 2

Feature updates:

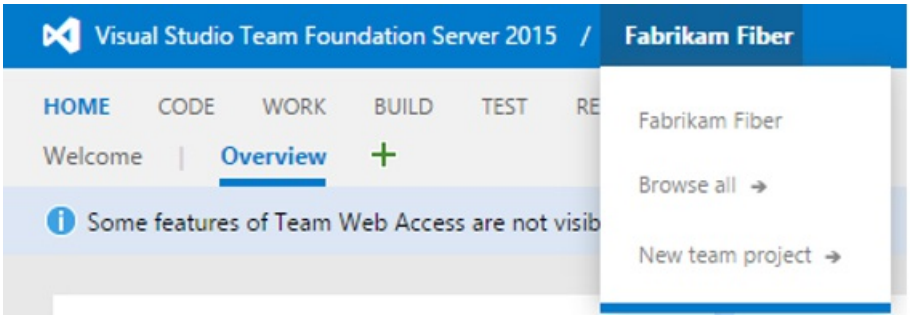
- Team project creation and deletion experience improvements
- Release Management: new features and improvements
- Test results available in the Release summary
- Reordering cards on boards
- Deleting work items
- Global shortcut keys
- Dashboards edit mode
- Auto-refresh dashboards
- Build widgets in the dashboard catalog
- Work item query charts in the dashboard catalog
- @mentions and #ID
- Pull request widget
- Markdown widget lets you choose an existing file in a code repository to display
- Common identity picker

- Gated check-in for Team Foundation Version Control
- Version control web
- Team Foundation Server extensions
- Keyboard shortcuts for Kanban board
- Improved linking between code and work items
- Build-related features and improvements
- Testing: new features and improvements
- Test in Java

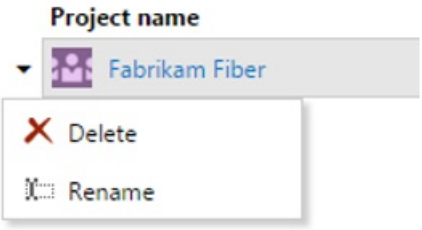
Details of What's New in Team Foundation Server 2015 Update 2

Team project creation and deletion experience improvements

We are giving you the option to create and delete Team projects from the web experience and through the REST APIs. These improvements help to bring more parity between Visual Studio Team Services and Team Foundation Server for project management operations. Users who have permission to create team projects can do so from the collection administration view or the navigation drop-down list. See creating a team project for more details.



Similarly, users who have permission to delete a team project can do so from the collection administration view.



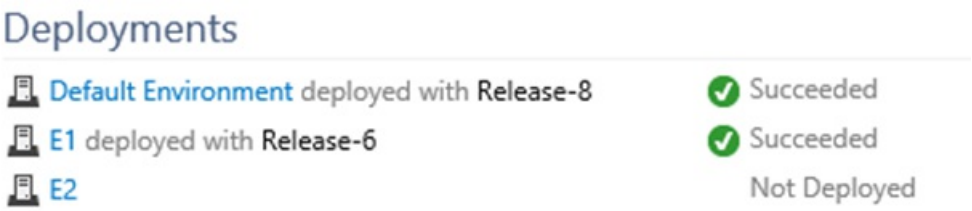
Release Management: new features and improvements

- **New version of web-based Release Management.** Release Management is now integrated into TFS. You can access Release Management features directly from the RELEASE hub in TFS web access without having to install a separate server or client. This new version of Release Management is simpler to use than the earlier version and is based on the same infrastructure and experience as the new Build system in TFS.
  - Simplify your application deployment to on-premises servers, SCVMM- or VMware-managed virtual machines, or to Azure.
  - Control your deployments by setting up release definitions that trigger deployments to environments in sequence or in parallel. Track how far your work items, commits, builds, and releases are deployed.
  - You can create bugs for failed tests. Debug information, like error message and stack trace, will be auto-populated in the bug.

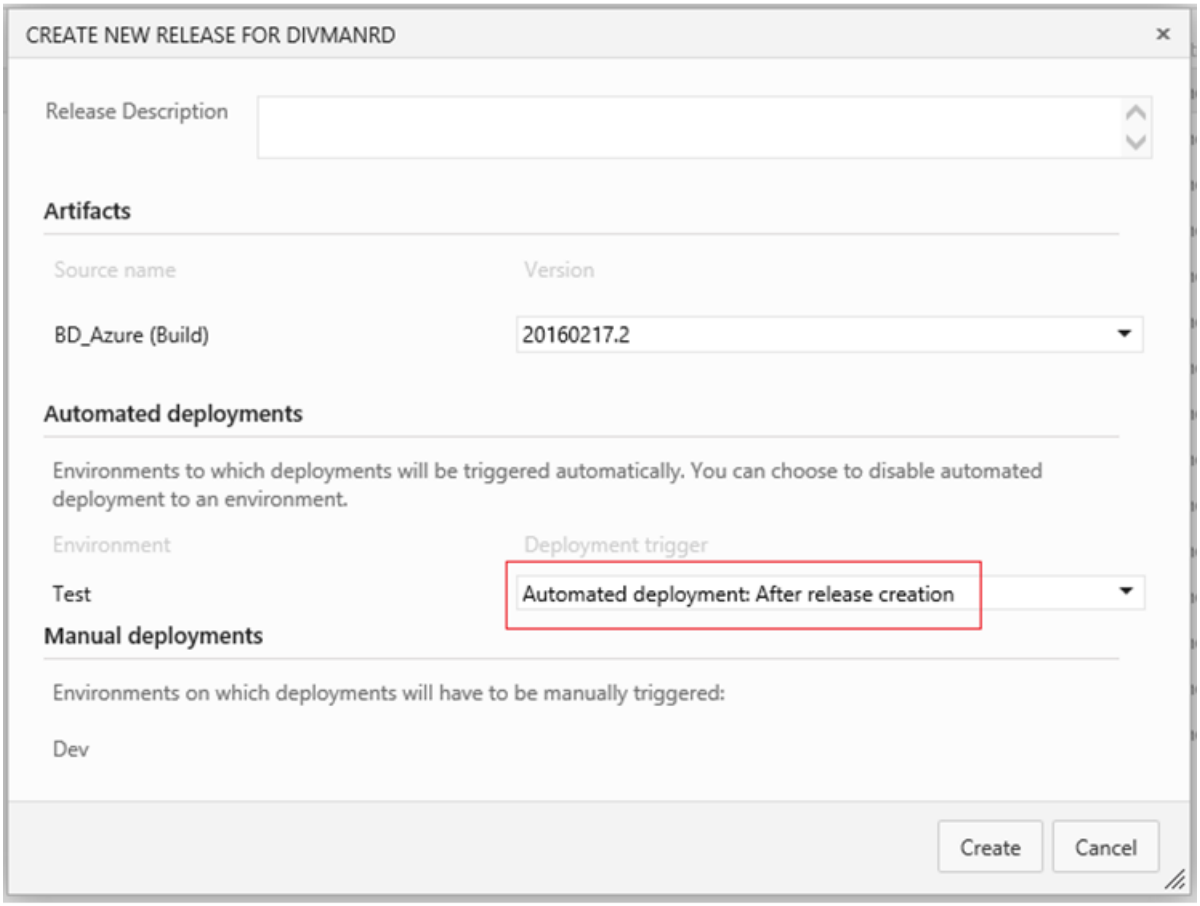
There is documentation about this new version and some recent blog posts.

- **A new extension for integration with VMWare.** Connect to a VMWare vCenter server from Visual Studio Team Services or from TFS and create, delete, and apply snapshots on virtual machines that are managed in vCenter. For virtual machines that are managed in VMWare, dynamically provision or restore them to a clean snapshot, then create build and release definitions to deploy. Get this new extension from the Marketplace.

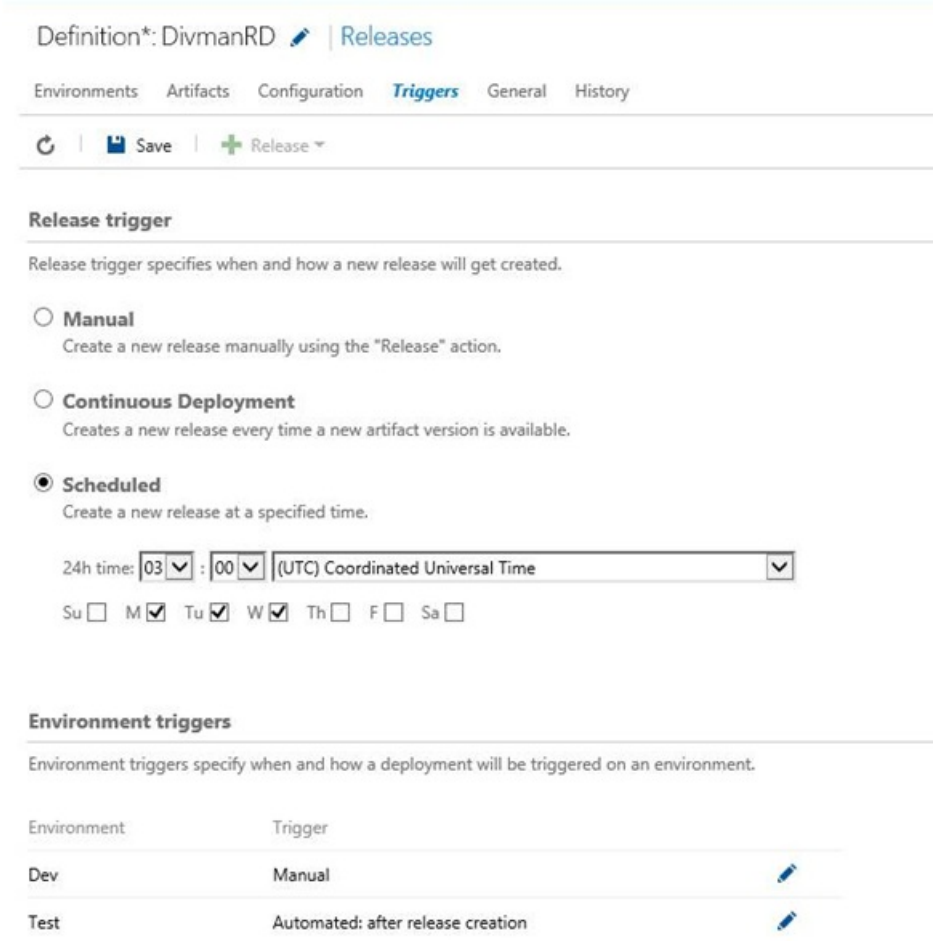
- **A new extension for System Center Virtual Machine Manager (SCVMM).** Connect to a SCVMM server and create, delete, and apply snapshots on virtual machines that are managed in SCVMM administration console. Now you can replicate one of the key features from TFS Lab Management: create build and release definitions to deploy to virtual machines that are managed in SCVMM after you have restored them to a clean snapshot. Get this new extension from the Marketplace.
- **A deployments section in build summary.** This new section shows a list of environments where a build was deployed through Release Management. You can also create a new release from this section.



- **Set target environments when you create and deploy a release.** We have added the ability to override automatic deployment. When you want to turn off automatic deployment to specific environments, go to the Automated deployments section in the Create New Release dialog box.A note on security: The flexibility to edit the deployment flow graph across environments is still restricted to release definition authors only.

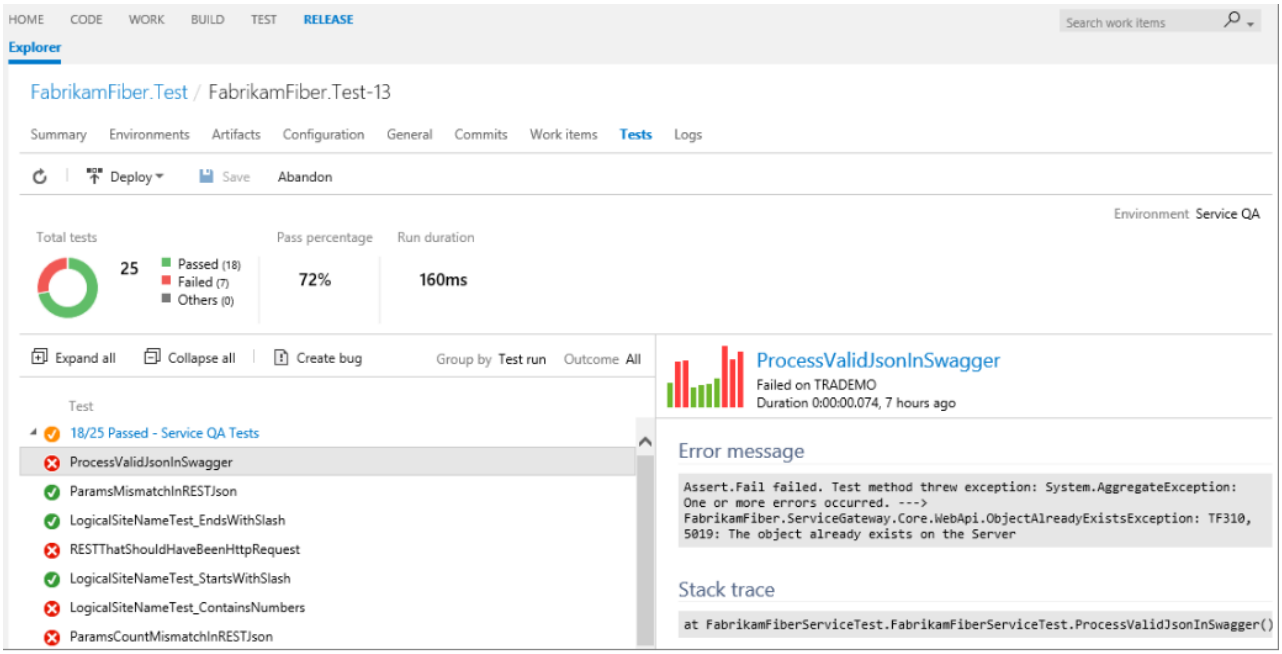


- **Create new releases on a schedule.** We have added the ability to set up a schedule for creating new releases. The latest version of successful build artifacts is picked up at the scheduled time. Once you create the release, automatic deployment to environments is determined by the deployment conditions configured on those environments. To configure a scheduled trigger for creating new releases, go to the Triggers tab in the release definition.



Test results available in the Release summary

To view test quality and analyze test results, the Tests tab in the Release summary will show you test status of each environment in which tests have run. The status includes count of passed and failed tests, pass percentage, and test duration for a particular environment or for the entire release, across all environments. You can drill down into the error message, stack trace, and test attachments to look for failed tests - without leaving the Release summary page.



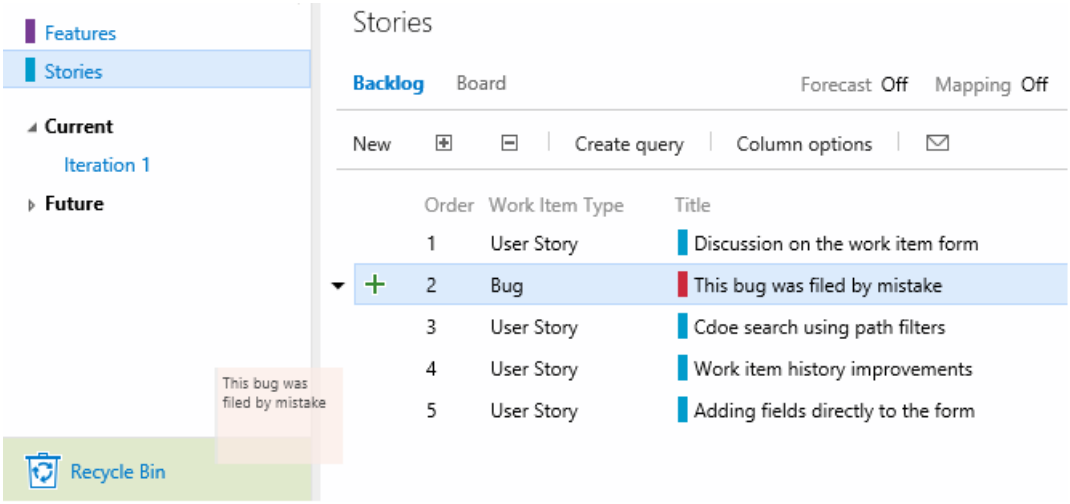
Reordering cards on boards

Optimize boards for the way your team works. You can now specify how cards are reordered when you change

columns: reorder with no restrictions or strictly follow the backlog order.

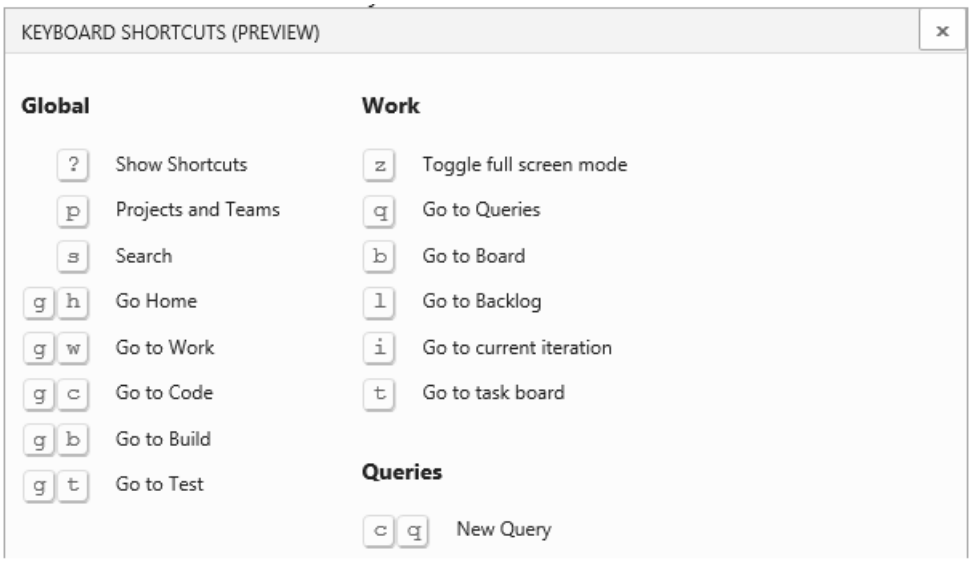
Deleting work items

You now have the ability to delete work items, complete with a recycle bin experience! This capability replaces the **Removed** state. We have implemented **delete** as a unique permission, so you have fine-grained control over who has access to this capability.



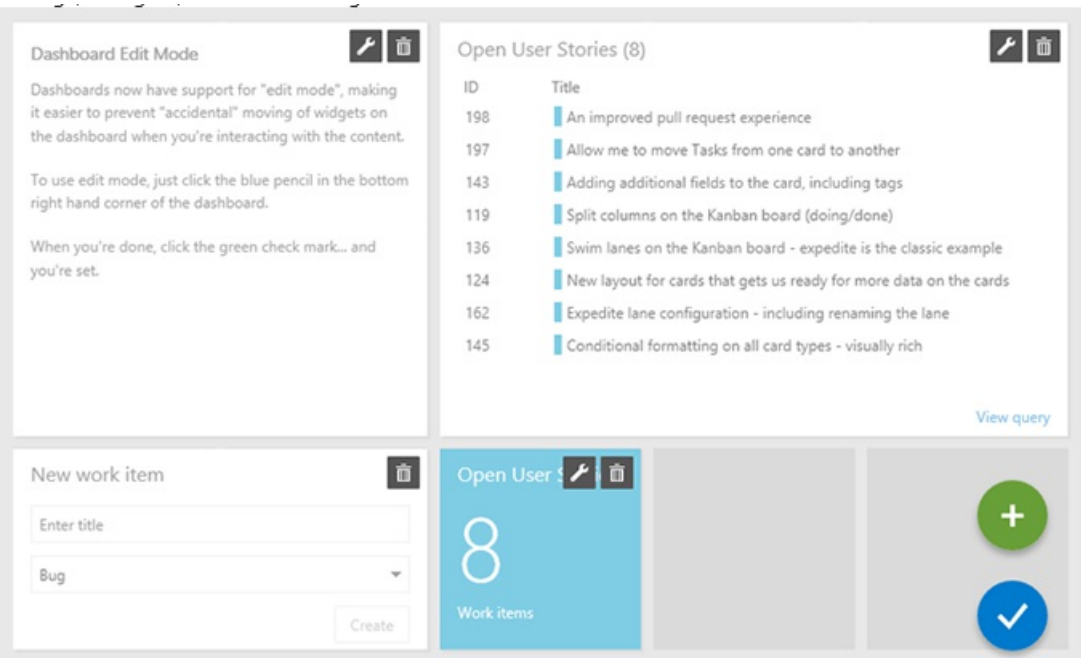
Global shortcut keys

Introducing keyboard shortcuts for the Code and Work hubs! Navigate without ever taking your hands off the keyboard.



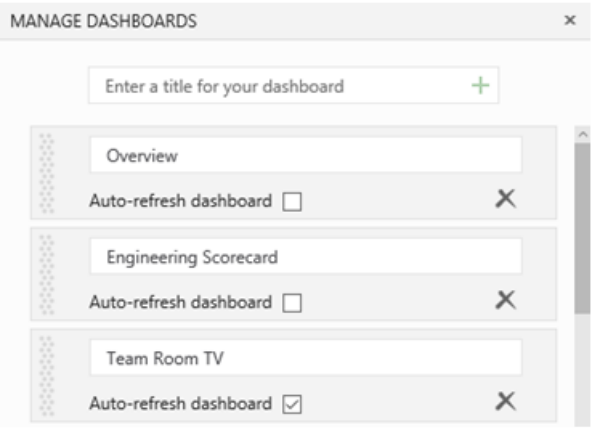
Dashboards edit mode

We have added a new mode called **edit mode** to all dashboards. Widgets will not move unless you select edit mode, reducing the chances of "accidentally" moving a widget. While you're in edit mode, you can remove, rearrange, and configure widgets, as well as add new widgets.



Auto-refresh dashboards

We have added an option to auto-refresh a specific dashboard every 5 minutes. This is a great way to put the dashboard on your TV monitor or hallway screen and keep it updated.

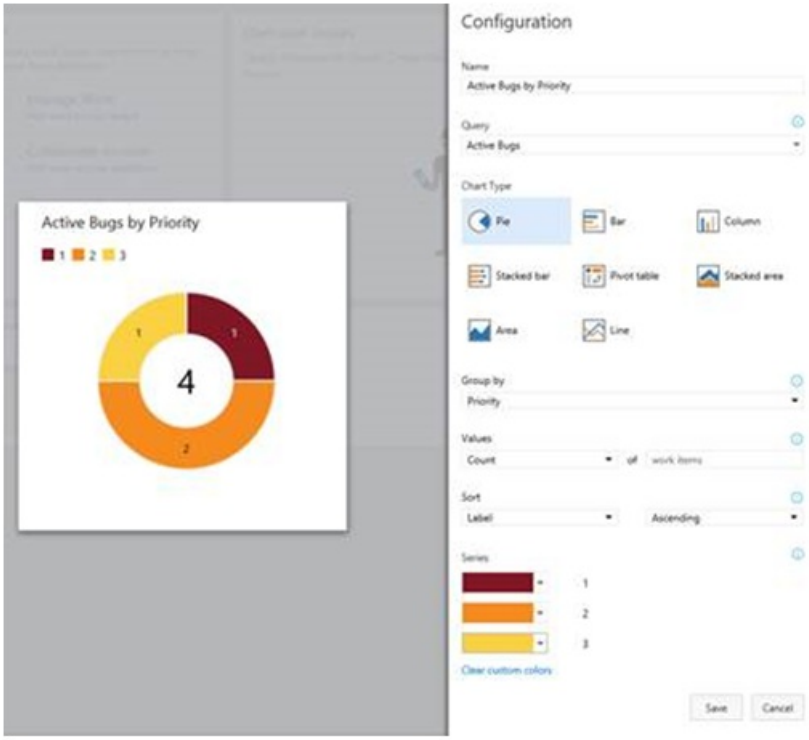


Build widgets in the dashboard catalog

A new option to add a build history chart from the dashboard catalog lets you configure the build definition directly from the dashboard. Read the dashboards futures blog to learn how we are improving the discoverability and ease in bringing different charts to your dashboard.

Work item query charts in the dashboard catalog

You can now add charts for work item queries to the dashboard from the widget catalog. These charts are configurable. You can choose a query, a chart type, and visualization options while staying in the dashboard context. Chart types include pie, bar, column, stacked bar, line, area, and stacked area. You can also display data in a pivot table. Additionally, you can still add charts from the Charts tab in the Work hub and configure them just like you have always done.



@mentions and #ID

Now you can use the **@ symbol** to mention people in pull request discussions, including commit changeset and shelveset discussions.

When you mention a person in a discussion, they receive an email alert that includes the discussion and a link to the pull request, commit, changeset, or shelveset. People mentions are clickable and display an identity card, with additional details. If you mention individuals who do not have permission to the pull request, commit, changeset, or shelveset, you'll receive an email indicating that the @mention has failed.

In addition, you can now use the **# symbol** to mention work items in pull request discussions, including commit changeset and shelveset discussions.

These work items are clickable and open a new tab, where you'll see the work item experience that you're already familiar with.

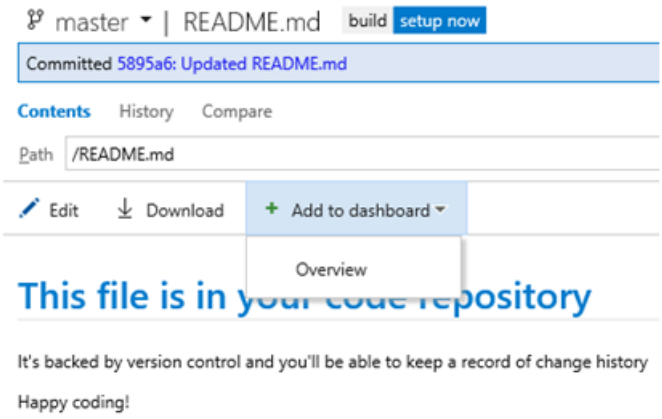
Note: We do not currently support @/# mentions in work item discussions and code comments. This is something we are planning on supporting in a future release.

Pull request widget

Team admins can now add the pull request widget from the dashboards catalog. For a specific repository, the pull request widget lets users see pull requests that are assigned to the team, assigned to me, and created by me. Users can quickly navigate to an individual pull request or they can view the summary of pull requests in the repository.

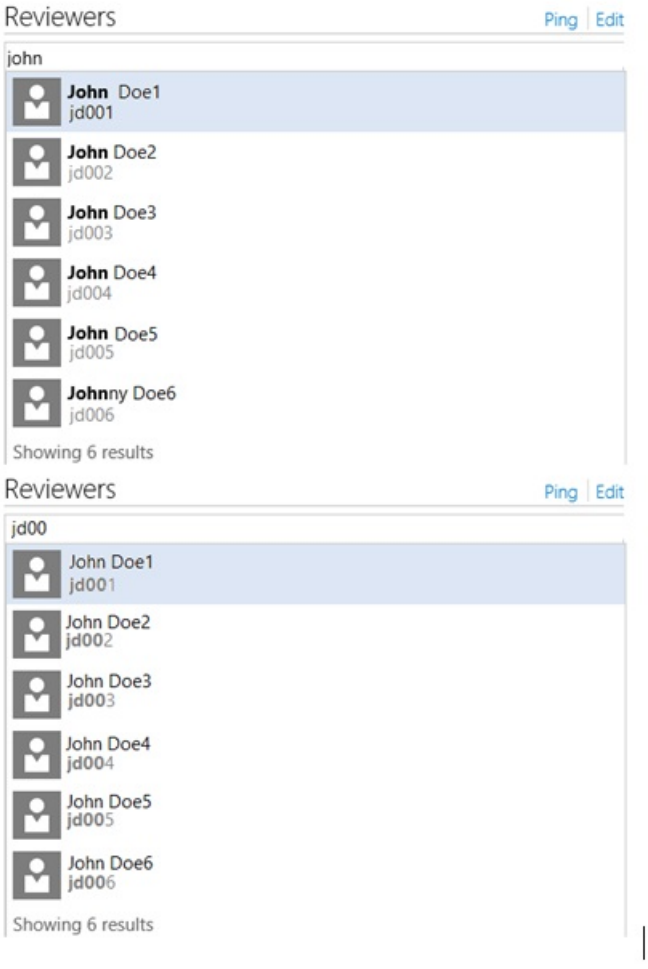
Markdown widget lets you choose an existing file in a code repository to display

You can now choose to display any markdown file in your existing repository. You can also add the file to any dashboard in your team project directly from the Code Explorer.



Common identity picker

We are rolling out an improved experience to search and discover users and groups in version control, release management, and @mention areas. Users can now do prefix-based searches on attributes like Display Name (such as 'john doe') and SAM-Account-Name (such as 'jdoe').

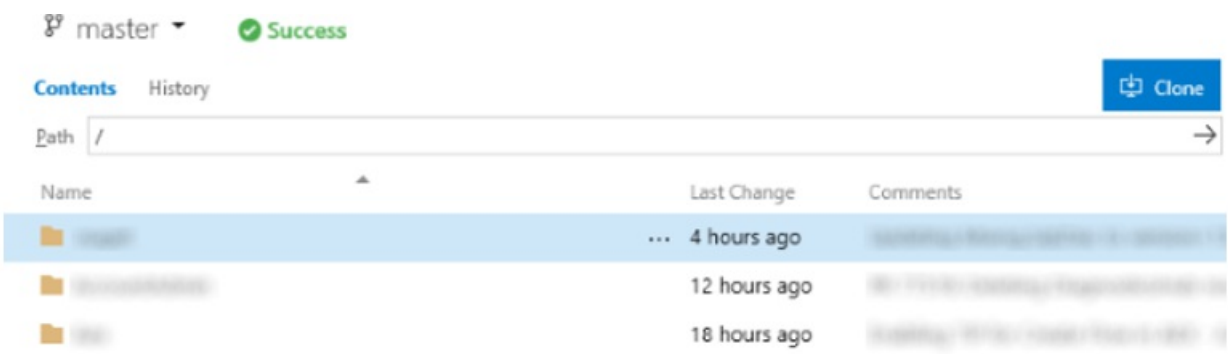


Gated check-in for Team Foundation Version Control

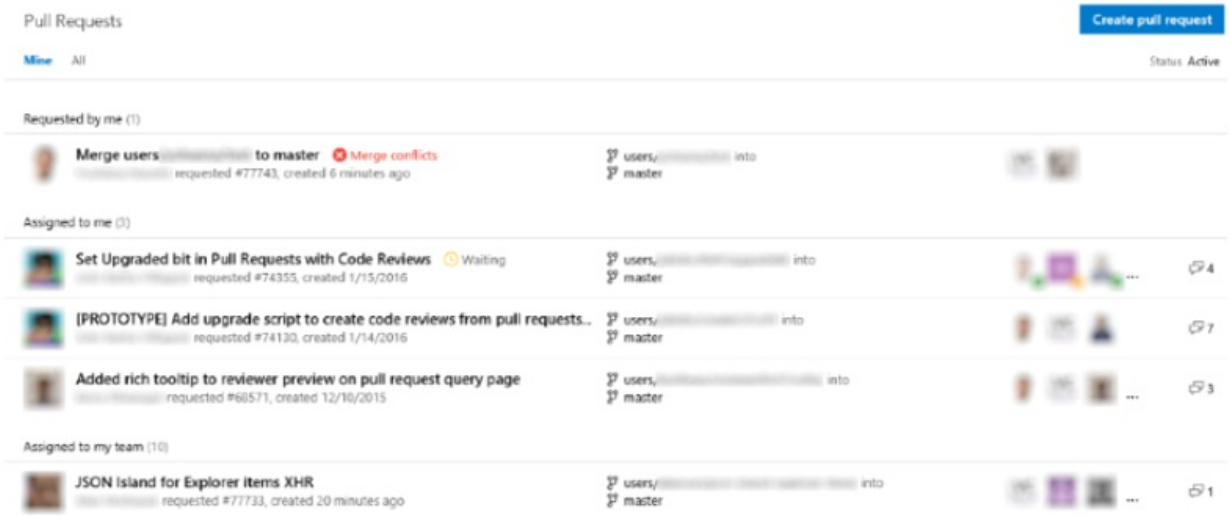
TFVC projects can have branch policies to require a successful build before any code is submitted into a branch.

Version control web

The CODE hub has several new features and numerous usability and style improvements. The Code Explorer now shows the status of the latest build for the selected branch, and has all new, modern icons to improve readability on high resolution displays.



The Pull Requests hub has improved views to highlight the pull requests you have created and those assigned to you and your team. Traceability between pull requests, work items, and commits were improved by new linking experiences.

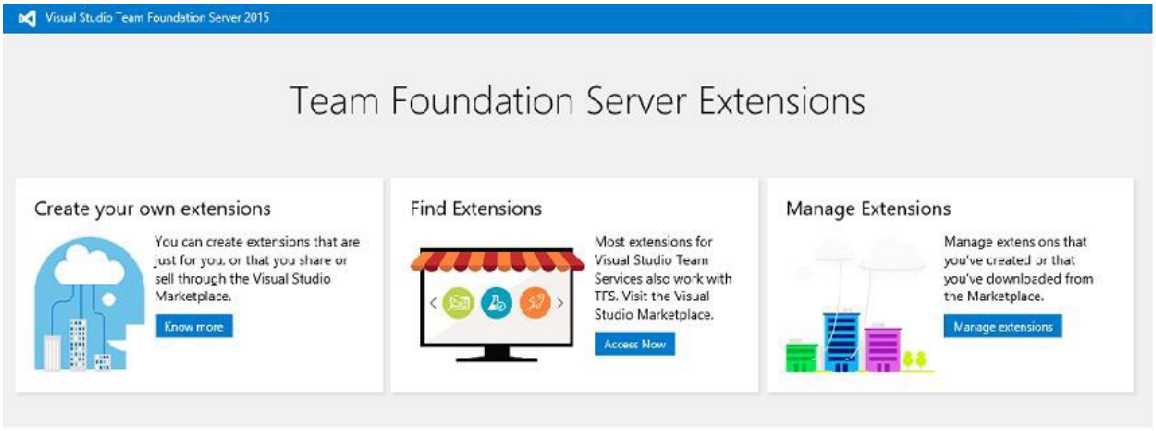


Team Foundation Server extensions

Extensions enable integrated experiences that extend and enhance the Team Foundation Server web experience, either by adding new capabilities or integrating with existing services.

On-premises extensions can be uploaded to Team Foundation Server and installed on specific team project collections.

Extensions can also be downloaded from the Visual Studio Marketplace and uploaded to a Team Foundation Server.



Keyboard shortcuts for Kanban board

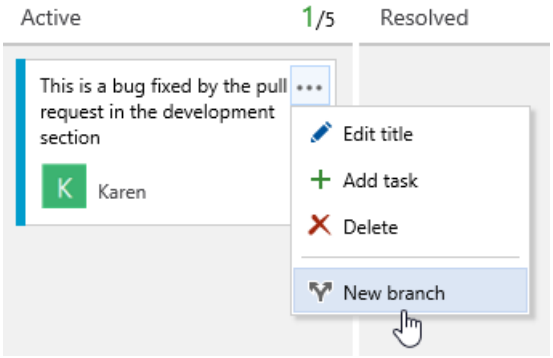
You can now click a work item on the Kanban board and use keyboard shortcut keys for common tasks. Here's a

sample of what you can do:

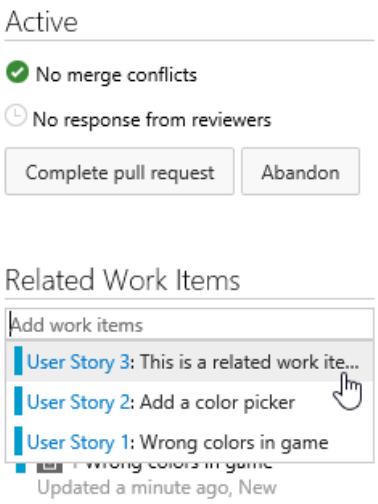
- create new work items
- move work items within or across columns and swimlanes
- expand and collapse work items

Improved linking between code and work items

In addition to commit linking, we have added new linking capabilities. You can link work items to Git branches and pull requests and also create branch links directly from backlog items or the Kanban board.



When you create a pull request, any work items linked to the source branch or the commits are automatically linked to the new pull request. The pull request details view has also been updated to include a Related Work Items section. The Related Work Items section lists all of the work items that are linked to a given pull request and provides a simple experience for adding links to more work items.



Build-related features and improvements

- **Build administration.** Administrators for build queues can now control who is able to create build and release definitions that use a given queue. This allows a team to have its own build resources within a given account or project collection.
- **Historical statistics.** Historical statistics for build and release agents are available from the queue and pool view. This can give build administrators a better understanding of how much usage there is for a given build resource.
- **Build interface improvements.** We have simplified the interface for creating new build definition. You can set key information about your sources and queue after selecting a template.
- **Extend build results view.** The build result (sometimes called build summary) view can be extended with custom information and views by using the extension framework. You can also extend the view by publishing a simple markdown file using this command:
  `##vso[task.addattachment type=Distributedtask.Core.Summary;name=myattachmentname;]c:\myattachment.md`.
- **Publish tasks as extensions.** Using the gallery, build and release tasks can be published as part of an

extension.

Testing: new features and improvements

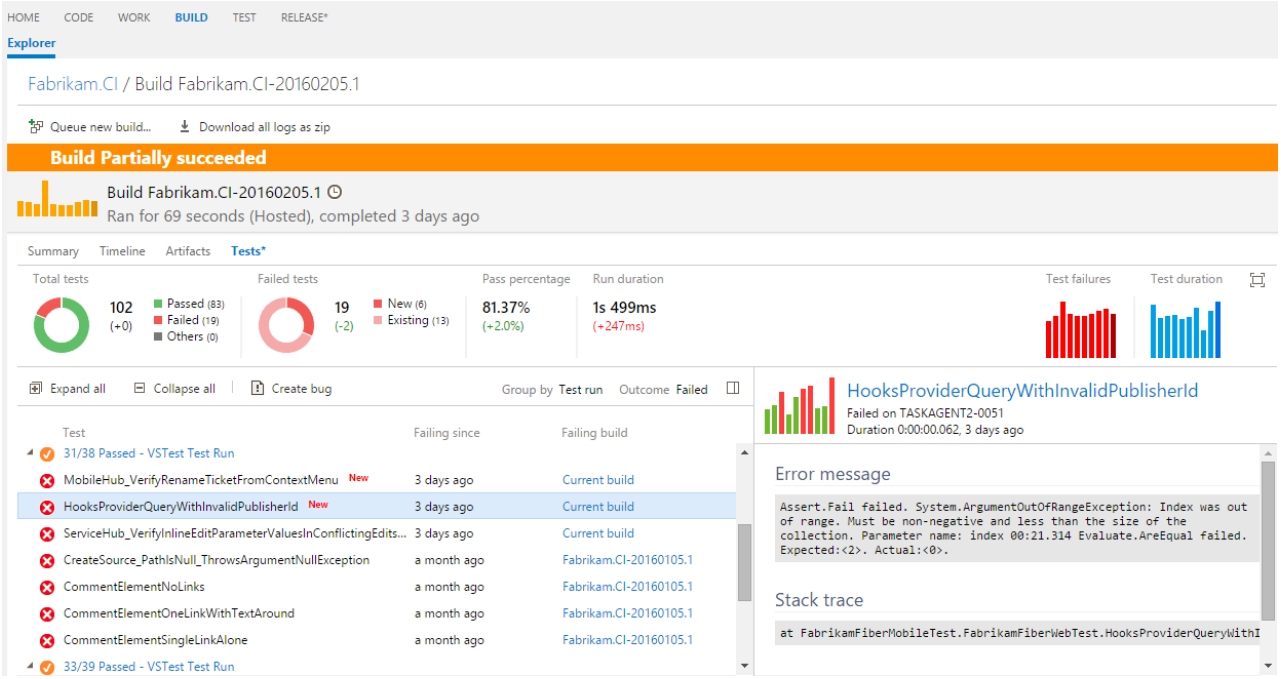
Planned testing in Build vNext

If you have invested in planned testing using test plans and test suites, you can trigger automated runs in Build vNext by using the **Run Functional Tests** task. Results are displayed in the Tests tab on the Build Summary page, as well as in the Test Hub. We have added a new Group By option on the Tests tab. Grouping by Test Suite gives you a quick view of all suites that were run and their quality.

Test results in build

There is help for you to analyze test results in the Build summary page:

- An aggregate summary of test results across all test runs in the build.
- Tests failing for the first time in the bug are flagged as **New failures** to help you identify regressions. For tests that continue to fail across multiple builds, you can see how stale the failures have become and navigate to the build in which a failure was introduced.
- Trend charts that show the count of failed tests and test duration for the last 10 builds. You can add these charts to the dashboard.
- Code coverage reporting was enhanced to show a pictorial view with build-to-build deltas.
- Test status is now available the build completion email notification.



Task improvements:

- Visual Studio Test using Test Agent task was renamed to Run Functional Test.
- The Visual Studio test task and the Run Functional Test tasks are faster, more intuitive, and user-friendly.
- We have added support for xUnit, NUnit, Junit and the .trx file format to the Publish Test Results task.
- A new task, Publish Code Coverage, uploads data from Jacoco and Cobertura code coverage tools.
- In addition, we completed a lot of work to give our Java users the ability to publish results and code coverage information to TFS. Read about it in the section, "Test in Java."

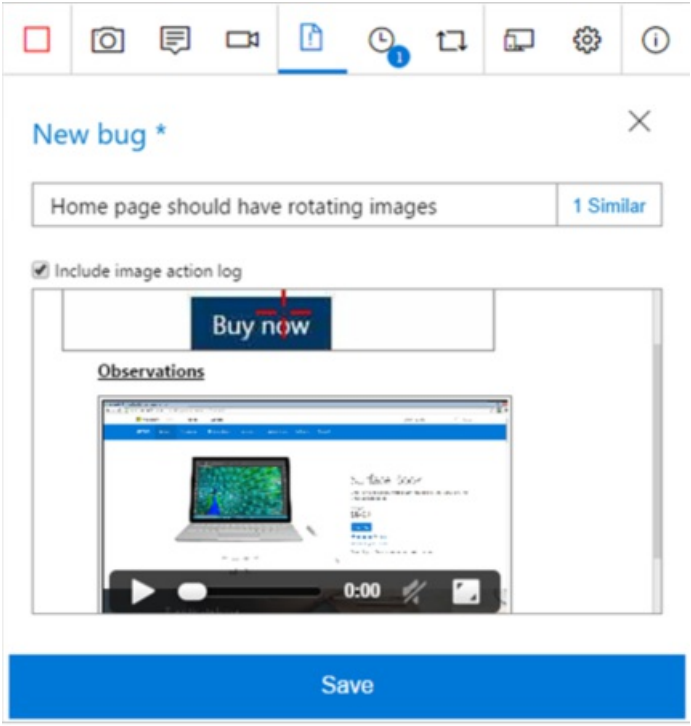
Manual testing improvements:

- **Filter test plans.** Set a work item query based filter for test plans.
- **Show test from child suites.** See all tests that are present in the current suite and their child suites, with one click.
- **Delete test plans.** One of the top UserVoice items for test is available in this release. Now you can delete plans directly from the Test Hub.

Exploratory testing (XT) on the web

Learn about exploratory testing on the web and get started by installing the Chrome browser plug-in from the Marketplace. Features include:

- **Capture screenshots and notes.** Simplified capture of notes and screenshots with inline annotation capability.
- **Create bugs and tasks.** Capture the issues found during the exploratory testing as either Bugs or Tasks. The notes, annotated screenshots, videos, image action logs, team area and iteration paths, and system and browser information are automatically captured.
- **Search and update existing bugs.** As you create a bug, the extension will automatically search and list existing bugs based on the Title match. You have the option to update an existing bug with the new information so you can avoid duplicate bugs.
- **Explore work items.** Do exploratory testing for a specific work item. This lets you associate the selected work item with your ongoing testing session and view the acceptance criteria and description from within the extension. It also creates end-to-end traceability between bugs or tasks that you file and the selected work item. You can explore the work item either directly from a work item or from within the extension:
  - **Directly from a work item.** Launch exploratory testing session for a specific work item directly from within the product. We have added entry points on all cards, grids, and in the Test hub.
  - **Within the extension.** Search for a work item from within the XT session and then associate it with the ongoing session.
- **Data collection - image action log.** Now you have the option to add the steps that lead to the bug automatically with just one click. Select the Include image action log option to capture the mouse, keyboard, and touch actions and add the corresponding text and images directly into the bug or task.
- **Create test cases based on image action log.** Simultaneous test design and test execution is the basis of true exploratory testing. Create test cases during your exploratory testing session and the test steps, with images, are automatically filled in for you. You can edit the captured text, add the expected result, exclude rows that aren't relevant, and save the text for upcoming test passes and runs.
- **Support for screen recording.** You can now capture your findings for all those problematic issues by using on-demand screen recording, which automatically attaches to the task or bug that you file. Support for audio is coming next.
- **Connect to devices.** Test your application on devices by using browser-based emulators or device cloud providers like Perfecto.
- **Standalone mode.** Now you can jot down notes, take screenshots (and annotate them if you want to), and create bugs without connecting to Visual Studio Team Services or to TFS. Share your findings with your team in a report that includes details of all the bugs you created, screenshots, and your notes.



Test in Java

- **Support for JUnit.** You can upload JUnit results to TFS from Ant, Gradle, and Maven tasks.
- **Support for JUnit in Publish Test Results task.** Run tests using a different command line tool and upload results to TFS.
- **Code Coverage support.** Choose between Jacoco and Cobertura as Code Coverage tools for your Java project. The build file is modified on the fly to use the correct tool and the results are uploaded to TFS.
- **Publish Code Coverage task supports Jacoco and Cobertura.** If your build file is already set to collect code coverage information, or if you're using a different Code Coverage tool (such as Istanbul) that supports outputting results in Jacoco or Cobertura format, you can use the Publish Code Coverage task to upload code coverage data to TFS.

Bug Fixes & Known Issues

For a complete description of technology improvements, bug fixes, and known issues in this release, see the following MSDN article, Description of Team Foundation Server 2015 Update 2.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2015 Update 1 Release Notes

9/7/2018 • 14 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

> **NOTE**
>
> **This is not the latest version of Team Foundation Server.** To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2015 Update 1.

To learn more about Team Foundation Server 2015, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: November 30, 2015

Summary of What's New in Team Foundation Server 2015 Update 1

Feature updates:

- Git and Team Foundation Version Control (TFVC) in the same project
- Version control on the web - history and getting started improvements
- Query on Kanban columns
- SonarQube works for Java programs built with a Maven Build task
- SonarQube Analysis build tasks work with on-premises and hosted agents
- Multi-select on all backlogs
- Tag coloring on Kanban
- Tasks as a checklist
- Dashboards: customizable; create multiple dashboards; new widgets
- Rename Kanban columns and swimlanes inline
- Pull requests in Visual Studio
- #ID in pull requests
- Branch policy to require linked work items
- Add/remove users from capacity
- Multiple activities per team member
- Drag any item to an iteration from anywhere
- Add panel on the iteration backlog
- Line on the burndown indicates actual capacity
- Reorder cards when changing columns
- Configure settings directly from backlogs/boards

- Hide empty fields on cards
- Card coloring on task board and Kanban board
- When creating a query, limit the values shown for Work Item Type
- Manual Testing: Export test outcome
- Manual Testing: Manual test step results and iterations for data driven tests
- Improved access control for build resources
- Improved source control integration in Team Build
- Fixed small usability issues in Build Explorer
- Test result retention policy
- XAML build parity progress
- Commit details summary is easier to read
- Improved experience for empty Git repositories and cloning existing repositories
- Azure Resource Group deployment support in build workflow
- Continuous delivery: Provisioning of resources in Azure
- Visual Studio error list filtering governs modified files
- Standalone Office integration installer

Details of What's New in Team Foundation Server 2015 Update 1

Git and TFVC in the same project

Team Foundation Server now supports adding Git repositories to Team Foundation Version Control (TFVC) Team Projects or adding TFVC repositories to Git team projects. This makes it easier to adopt a new version control system while keeping all your current team project data. Team members need to run Visual Studio 2015 Update 1 to switch between TFVC and Git in the same team project.

Version control on the web - history and getting started improvements

We have improved the commit details page for merge commits making it easier to read the commit message and see the important details of a merge commit. Merge commits default to showing a diff against parent 1 because that is the most likely diff you want, especially if you use pull requests. In addition, the empty Git repo and clone Git repo experience make it easier to clone your repo through Visual Studio, XCode, Eclipse, or the command line.

Query on Kanban columns

Data from your Kanban board, including the board column, column done indicator, and board swim lane, are all now available. You can create queries, charts, and styling rules based on them. These fields show up in all column option pickers, and can be exported as read-only fields in Excel. You will see three new fields in work item experiences:

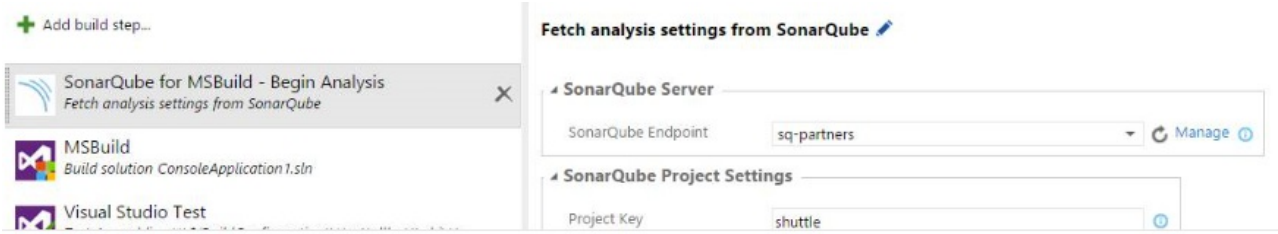
- Board Column: current Kanban board column the work item is in.
- Board Column Done: False (Doing) or True (Done) value depending on whether the work item is in the Doing or Done split column on the Kanban board.
- Board Lane: current Kanban board swimlane.

SonarQube works for Java programs built with a Maven Build task

You can now perform a SonarQube analysis using the Maven Build Task.

SonarQube Analysis build tasks work with on-premises and hosted agents

SonarQube build tasks that embed the sonar-msbuild-runner 1.0 work with on-premises and hosted agents. You can now use two build tasks to execute a SonarQube analysis in conjunction with MSBuild. Before the build steps that execute the actual MSBuild, insert a **SonarQube for MSBuild - Begin Analysis** task to specify the SonarQube project parameters, the connection to the SonarQube server, and (until SonarQube 5.2 is available) the connection to the SonarQube database. After the build and any test tasks, append a **SonarQube for MSBuild - End Analysis** task to complete the analysis and send the data to SonarQube.



Multi-select on all backlogs

Multi-select is available on all backlogs and supports:

- Drag-and-drop reordering and re-parenting
- Dragging items to an iteration
- Dragging items to the mapping pane
- Support for **move to top** and **move to position** on the context menu
- Bulk edit
- Bulk assign to

Tag coloring on Kanban

You can now change the color and style of title text as well as add color to specific tags. To try this, click the gear icon. You can format title text under **Styles** and color tags under **Tag colors**.

Tasks as a checklist

Tasks can now be added, edited, and marked as done directly from the parent card on your board. A summary of the roll-up status indicates the number of completed tasks and how many remain.

Dashboards

Dashboards are now available in Team Foundation Server Update 1. Dashboards provide visibility to your team and stakeholders on progress of work, code, tests, and builds.

We have added the following:

- **Customizable dashboards.** The existing project home pages are now a completely customizable experience, giving you full control of all the data on your screen. As a team admin, you can rearrange, remove, or add the widgets that you want and use. A new widget catalog is accessible to team admins through the large green plus (+) button at the lower right side of the dashboard.
- **Create multiple dashboards.** Until now, the Team Overview page was the only page that teams could use to pin widgets. Now, as a team admin, you can create multiple dashboards using the plus (+) button next to an existing dashboard or through the dashboard manager. The dashboard manager is accessible through the settings icon on the dashboard.
- **New dashboard widgets.** All the pieces of data that used to be available on the previous Team Overview page are now available through the widget catalog. You can still add multiple charts to the dashboard through the Work, Test, and Build hubs. In addition, we've brought new widgets into the catalog: a conditional query tile (that allows you to set red or green if the number of work items crosses a threshold); a code tile (recent commits in the last 7 days); a query results widget (allows you to see all work items from a query); and a markdown widget (enables custom text, images, and links).

Rename Kanban columns and swimlanes inline

You can now rename columns on your Kanban board in place, without having to jump into the configuration dialog.

Pull requests in Visual Studio

Team Explorer has a new Pull Requests hub where users can see the list of pull requests they have created as well as the pull requests others have assigned to them. We have improved the Create Pull Request experience enabling users to publish branches and create pull requests in a single action.

#ID in pull requests

You can now mention work items using the "#" symbol in pull request discussions, pull request code comments, and commit code comments. Those work items are clickable and present the work item light box experience that you're familiar with.

Branch policy to require linked work items

We now have a branch policy to require associated work items for any pull request. Like the code reviewer and build policies, any code submitted to the branch must be submitted through pull request. When you create a pull request, the associated commits are inspected for work item links, and if there is at least one link, the policy is fulfilled. Also, you can now link work items to pull requests directly, and if the pull request is directly linked to at least one work item, the policy is fulfilled. If no work items are linked to the pull request or the associated commits, the policy will not succeed.

Add/remove users from capacity

Two new buttons on the toolbar let you add any member to your sprint plan (including those outside your team), as well as add any missing team member.

Multiple activities per team member

When planning capacity for a sprint, you can now assign multiple activities to a single team member. Just hover over or click on a row, then click the ellipses (...) to add a new activity. If a team member does both development and testing, you can fine tune the capacity to match your team's skill sets.

Drag any item to an iteration from anywhere

You can now drag any work item from any board or any backlog to a sprint.

Add panel on the iteration backlog

We have made a change that allows you to add new backlog items directly to your sprint backlogs. Previously, the only way to add new work to requirements or bugs to a sprint was to add them to the product backlog, and then drag each item one at a time to the correct sprint.

Line on the burndown indicates actual capacity

The burndown chart now has a line that shows available capacity during a sprint, in addition to the existing ideal trend line. Now you'll have a better idea of whether you're on track to finish all your work during the sprint. This line builds on the capacity data, including individual and team days off.

Reorder cards when changing columns

You can now reorder the card when you change the column, so the card stays where you drop it. We also eliminated auto-scrolling, so the board keeps its location after you drop a card.

Configure settings directly from backlogs/boards

The gear icon appears on all the backlog and board pages. Click it to show all the settings available for that page. As new settings become available, you'll find them here.

Hide empty fields on cards

We have added the ability to hide empty fields on the card. This means less whitespace, smaller cards, and more cards on your board. To enable this feature, click the gear icon, select **Fields**, and clear **Show empty fields**.

Card coloring on task board and Kanban board

You can now configure boards to include custom formatting on your work items, based on any value. Setting this up is easy. Click the gear icon on the top of your board, select **Card styles**, and create a rule for the color you would like to add.

When creating a query, limit the values shown for Work Item Type

You can now scope queries to limit the drop–down list for states and work item types, eliminating states and types from projects outside the scope of your query.

Manual testing: Export test outcome

The Export Test Plan feature in Test Hub lets you email or print a test plan in a review–friendly format. Until now, you could export test–authoring–centric information such as test plan, test suite hierarchy, test configurations, and test cases. Now there is support for exporting test–execution–centric information. You can use this feature to share

a detailed status of test execution progress with stakeholders. In the export selection dialog, you can export the latest test outcome, which exports the latest test result for all tests of a test case, including fields like the time when the test was run, the build that was tested, the configuration that was tested, the tester assigned the test, and the identity who actually ran the test.

Manual Testing: Manual test step results and iterations for data driven tests

Test Hub supports running manual tests and viewing test pass/fail information. Until now, viewing detailed test result information like test step details, comments, and attachments was possible only with Microsoft Test Manager (MTM). With this update, you can browse which test steps passed or failed and the comments entered by the tester while running the manual tests. In Test Hub, click the **Runs** tab. If you have run tests using MTM and have captured screenshots, video, and other attachments such as system information, you can view them as well. If your tests are data driven with parameters, you can browse the details for all the test iterations. To browse this detailed test result information, open the **Recent Test Results** pane and double-click a test result. You can also view these test results by navigating to Runs tab in Test Hub.

Improved access control for build resources

Build administrators can now add permissions to agent queue, which restricts who can use that queue in a build definition.

Improved source control integration in Team Build

You can control the client–side mappings for source control folders from Team Foundation Version Control (TFVC). We have reduced the number of characters in the default working directories for Team Build, which should help mitigate some of the path length issues in Team Build.

Fixed small usability issues in Build Explorer

The My Builds context menu item is available again in the Build Explorer. This opens the list of your builds in Visual Studio instead of always taking you to the web. Clicking on an individual build to see the details will still launch the browser.

Test result retention policy

We have enabled in-product support to simplify test result data cleanup and remove dependency on tools like the Test Attachment Cleaner. The test result retention policy feature adds the capability to clean up all test result data in the system including test runs, test results, and test attachments that are older than a specified number of days. All test results in the system, including results from automated test workflows, manual test workflows, and test results published using REST APIs are cleaned up using this feature. This retention policy is configurable at a team project level. You can specify separate policies for automated and manual test results, offering the flexibility to retain manual test results longer than automated test results.

We are also integrating test retention with the retention policy of the new build system (Build vNext) by adding functionality to delete test results when builds are deleted. More information about this feature is coming on the Visual Studio ALM blog.

XAML build parity progress

Team Build in Visual Studio Team Services and Team Foundation Server now support the following features from the XAML build system:

- Label sources in both Git and TFVC
- Client–side workspace mappings for TFVC

Commit details summary is easier to read

The commit details summary is easier to read because the commit message is at the top and the extended message is below. It also defaults to showing a diff with parent 1, which works well with the new pull request experience that always creates a merge commit.

Improved experience for empty Git repositories and cloning existing repositories

It is easier to clone a repository on Visual Studio Team Services and Team Foundation Server. You can **clone in**

Visual Studio or create a personal access token directly from the empty Git repo page or clone button.

Azure Resource Group deployment support in build workflow

With the addition of the Azure Resource Group deployment task, along with the support of ARM (Azure Resource Manager), you can manage the lifecycle of an Azure Resource Group. This includes support for provisioning and deployment of Azure resources. The Azure Resource Manager lets you create reusable deployment templates that declaratively describe the resources that make up your application (for example, a website and a SQL database). This simplifies the process of creating complex environments for development, testing, and production in a repeatable manner. You can find Azure templates at GitHub and Azure Gallery.

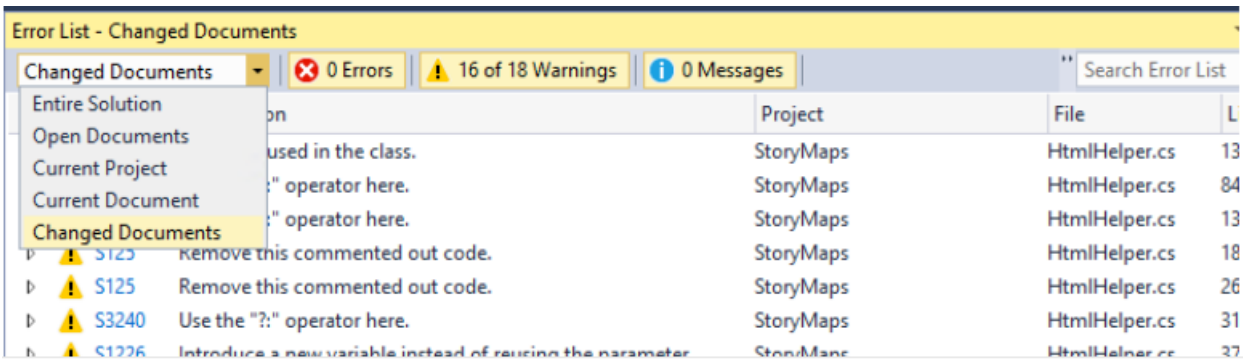
In addition, Azure service connection endpoint now supports Azure AD Security Principals

Continuous delivery: Provisioning of resources in Azure

Using Azure Resource Manager, you can now include a step in your build definition to provision resources. The new "Azure Resource Group Deployment" task lets you provision resources, like virtual machines, by using a template file and then deploy the latest build into those resources by using resource extensions, such as DSC resource extension.

Visual Studio error list filtering governs modified files

You can now filter the error list to see only errors, warnings, and information in files that have pending changes. This way, you are not overwhelmed by errors in the files that you did not touch, and you can focus on the errors that you have an opportunity to fix.



Standalone Office integration installer

The standalone Team Explorer installer was removed in Team Foundation Server 2015 because many non-developer features were moved into the TFS web experience. One inconvenience this introduced was that non-developers who wanted to use TFS Office integration capabilities could no longer access them unless they installed a full version of Visual Studio. In the TFS 2015 Update 1 release, a standalone Office integration installer is now available. It includes Excel integration Project client integration and the PowerPoint–based storyboarding tool.

Bug Fixes & Known Issues

For a complete description of technology improvements, bug fixes, and known issues in this release, see the following Knowledge Base (KB) article.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2015 Release Notes

9/7/2018 • 20 minutes to read

| Developer Community | System Requirements and Compatibility | License Terms | TFS DevOps Blog | SHA-1 Hashes |

> **NOTE**
>
> **This is not the latest version of Team Foundation Server.** To download the latest release, please visit the current release notes for Team Foundation Server 2018 Update 3. You can change the language of this page by clicking the globe icon in the page footer and selecting your desired language.

In this article, you will find information regarding Team Foundation Server 2015.

To learn more about Team Foundation Server 2015, see the Team Foundation Server Requirements and Compatibility page.

Please see the TFS Install page for more information.

Release Date: August 6, 2015

Summary of What's New in Team Foundation Server 2015

SKU changes:

- Basic license expanded
- Changes to the schema in databases
- Project Server Extensions
- SharePoint Extensions

Feature updates:

- Identity control and avatars
- Taskboard: Bugs on your backlogs and boards
- Product backlog updates
- Sprint backlog and task board updates
- Customize and configure your cards
- Kanban board updates
- Turn off the first column on the CFD chart
- SAFe support for Process Templates
- Process Templates renamed
- Current iteration query token
- Query progressive disclosure
- Branch policies
- Branch policies - Gated build
- Branch policies - Code review
- Branch history (pushes & pull requests)
- Web history view for Git projects

- Quick code editing
- History control
- View history on a folder
- Build automation system
- Team project rename
- REST APIs
- Service hooks
- Improved merge performance
- Assign multiple testers
- Cloud-based load tests
- Automated testing
- Change in API behavior

Details of What's New in Team Foundation Server 2015

Basic license expanded

The following features are now available to all Team Foundation Server users with a "Basic" license:

- Web-based test execution
- Agile portfolio management
- Work item chart authoring
- Team Rooms

What this means: All teams of five or fewer members with a "Basic" license have access to these features using Team Web Access for free, while larger teams can access this functionality at a much lower price point.

Changes to schema in databases

Team Foundation Server 2015 includes a large number of changes to the schema used in its databases, and as such, an upgrade from TFS 2013 and older releases is expected to take a significant amount of time. Because upgrades are offline, Microsoft is providing a tool, TfsPreUpgrade.exe, which can be used to perform the most time-expensive portions of the upgrade online against TFS 2013 QU4 and QU5 deployments. The Upgrade wizard includes a readiness check, which warns you if your database is large enough that running TfsPreUpgrade.exe is recommended.

Project Server Extensions

Project Server Extensions are now a separate download. Check out the TFS section of the Downloads page for more information.

SharePoint Extensions

In the past, if you wanted your Team Foundation Server integrated with a SharePoint instance that was on a different machine, you could either run the Team Foundation Server installer on the SharePoint server and then configure TFS Extensions for SharePoint *or* run a special installer (tfs_sharePointExtensions.exe) that put only the bits necessary for configuring TFS Extensions for SharePoint.

We have removed this special installer, so to integrate your Team Foundation Server with SharePoint, you must run the Team Foundation Server installer on the SharePoint server and then configure TFS Extensions for SharePoint.

Identity control and avatars

This new control includes a user's full name, avatar, and email address.

We've designed this control to be very intuitive to use. When you put focus on the control, it starts by giving you an MRU (most recently used) list of people to whom you've most recently assigned work items. If the person you're after is not in the list, just click Search and the list will populate with matching results from the users in your account. And, not only are we providing a new identity control, but we've also refactored many of the places where we display a user's name so that it now includes their avatar. You'll see avatars on cards on your work items,

boards, and more.

Taskboard: Bugs on your backlogs and boards

We've enabled the ability for teams to choose if they want to show bugs on their backlogs, regardless of process template. We have expanded the functionality of this setting. Teams can now choose to show bugs on the backlogs and taskboard with requirements (user stories or product backlog items), with tasks, or not at all.

Product backlog updates

Backlog navigation updates

We've overhauled the navigation of backlogs. From every backlog, you can drill down into more levels, all the way down to Tasks. In addition, from every backlog, you can toggle the levels above your backlog on or off by using the Parents filter. Items not owned by the team, but pulled in based on relationship, display with a hollowed-out color bar. At a glance, you can differentiate between items that your team owns from those owned by other teams.

Finally, you can reorder and re-parent in every view! Just drag and drop in any of the views to reorder items and change relationships.

Opt-in to portfolio backlog levels

Related to the navigation updates, you can now turn off backlog levels that your team isn't using. Prior to this update, all backlog levels were forced on every team. Each backlog level is now "opt-in", allowing you to configure the levels appropriate for your team. Click the gear at the top of the page, select the team you're configuring, and then choose the backlog levels you need.

Re-ordering in a filtered backlog

The context menu now provides an option to move an item to the top or to a specific position, even when a filter is applied on the backlog.

Text filtering on backlogs & queries

Now you can quickly filter backlogs and query results by using the new filter textbox we've placed on the toolbar. Simply type in text from the items you're looking for and the backlog/result is immediately filtered to show only those items with matching text. This feature is really handy when you're scanning a long backlog or query result for a specific item (or set of items). Note that the matching is done on data in the displayed columns—including tags.

Sprint backlog and task board updates

Show un-parented tasks

The tasks in the sprint that do not have a parent story will now show up on the sprint backlog and task board, under an "Un-parented" category. The un-parented row is highlighted with a grey-colored bar. You can move tasks from an un-parented row to any user story, and vice versa. (Note: Drag & drop of an un-parented row is not allowed; it will always appear on the top of sprint backlog as well as the task board.)

Collapse completed stories

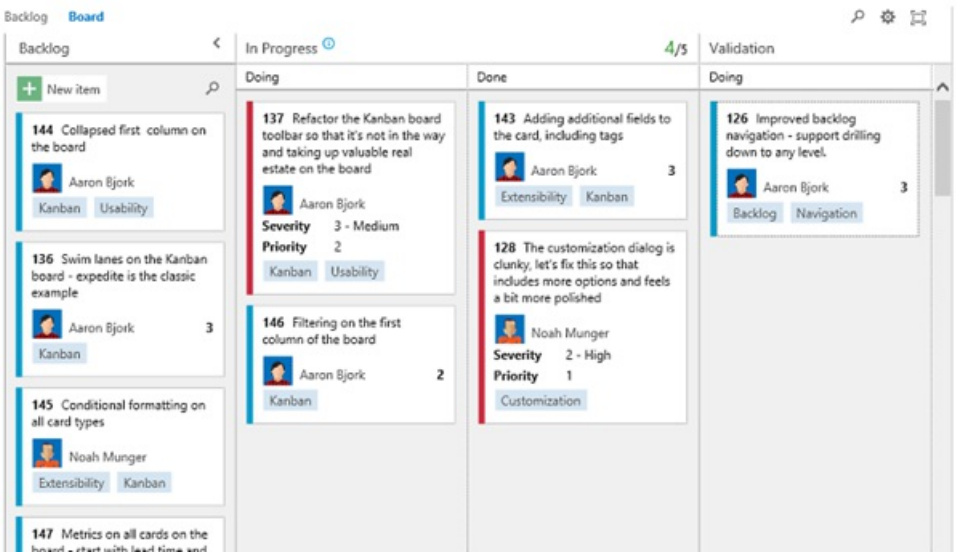
Completed stories collapse automatically when the task board is opened. All stories on the sprint backlog collapse by default. Stories that are collapsed but have pending work will show a warning on the task board. Collapsed rows on the task board will also show the summary of pending work for that user story. And, PBIs on task board will now appear as cards just as tasks do.

Customize and configure your cards by adding fields & tags

Not only can you customize how your cards look on your Kanban board, but you now have configuration options for the data showing up on your cards in the Customize Cards dialog box.

(A similar customization dialog box is also available for the Taskboard.)

You can turn the ID on or off, select how the assigned to field is displayed, and choose to show tags directly on your cards. Most people want fields like "title" and "assigned to" on every card, but it's a time saver to bring a bit more information to the cards so you can take action on them without having to open them for more details. For example, notice that we've added both "Priority" and "Severity" to the following bug cards:



The Custom fields added to your cards are directly editable from the board. And, these options are per team (or backlog) and per work item so that you have maximum flexibility.

Kanban board updates

Adding and editing directly from the Kanban board

We've updated all boards to support adding new cards and inline editing. The Kanban board now has a New item button at the top of the first column that adds a new card. After you add a new card, all data on the card can be edited directly from the card itself. Learn more about Kanban inline adding and editing.

Reordering on the Kanban board

We've turned on reordering items on the board. You can now move items up and down in priority within each column on your board. Any changes made on the board are also reflected directly on the backlog. In fact, with this change, many of you may choose to use the board over the backlog, as the board now supports adding, inline editing, and reordering. Learn more about Kanban reordering.

Filtering across all data displayed on the Kanban board

You can now filter the entire board. Enter a filter term in the all-board filter and it will filter by any information displayed on your card, including any fields that you might have added, tags, or ID. We've also provided a filter on the first column so you can find that item on your backlog to pull onto your Kanban board.

Split columns on the Kanban board

We've added a new feature to our Kanban board called "Split Columns". Kanban teams use a pull model to move work through the board. To do this effectively, each column on your board is split into two subcolumns—Doing and Done. Moving a card into the Done column provides a clear signal that work is ready to progress, and that the card can be pulled by the person/team who owns that next stage.

To split any of the columns on your board, simply click the Customize Columns link on the toolbar.

Learn more about Kanban split columns.

Swimlanes on the Kanban board

We added the ability for teams to create horizontal swimlanes to track different classes of work. A classic example is the Expedite lane. So now, each team can create their own lanes, and make the board look just the way they want.

Kanban definition of done

As work moves through your board, it's critical that you and your team are on the same page about what "done" means for each column. This release brings a new capability that lets you specify a definition of done for each column on your board. We even support markdown, so you can format the text or include hyperlinks to other locations.Columns with a definition now include a small icon in the header that communicates the agreed-upon definition.

Learn more about Kanban DoD (definition of done).

Turn off the first column on the CFD chart

You can now omit the first column of the Kanban board and get a more meaningful CFD chart. (The first column often represents the long backlog of items a team is working on, but not the items actively on the kanban board.)

To do this, select "Edit" from the CFD chart, and then uncheck the "include first column" checkbox. (And note that for all existing backlogs/CFD charts, this box is checked by default.)

SAFe support for Process Templates

We are very excited to provide built-in support for the Scale Agile Framework (SAFe) using our existing Scrum, Agile, and CMMI templates.

Support for Epics

- We have added an Epic work item type and a backlog/board to track Epics. Epics are hierarchically above Features. Features are mapped to Epics, like Backlog items are mapped to Features.
- Full backlog and board functionality are available. You can manage the Epic backlog like any other backlog as well as customize your Kanban columns and cards to match your needs. (The Epics backlog is not enabled by default. To enable this feature, check the "Epics" checkbox from the Team Settings page.)
- The Epics backlog can be turned on or off at the team level. As per our whitepaper, Portfolio Teams should enable the Epics backlog. Program and Feature teams can disable the Epics backlog if they don't manage Epics in your organization.

Support for Architectural vs. Business backlogs

We have added a "Value Area" field to all work items that appears on a backlog, that is: Epics, Features, and (depending on your process template), the field also appears on Product Backlog Items, User Stories, and Requirements.

The Value area has two values: Business and Architectural. By default value, all Epics, Features, and Stories are Business types. To create an Architectural Epic, Feature, or Story, set the value to Architectural.

With this functionality, you can define Architectural Epics, which in turn break down into Architectural Features and Stories, allowing you to track your architectural roadmap across your organization.

Learn more about SAFe support for process templates.

Process Template renamed

The name of the templates will change from the verbose names that includes the version name (for example, "MSF for Agile Software Development 2013.4") to simply "Scrum", "Agile" and "CMMI". The templates are now locked, which means that you cannot make changes to the shipped templates. To create a custom process template based on a shipped template, simply export an existing template, give it a new name and version, and then reimport it by using the Process Template Manager. Existing projects are unaffected by this change, which means that they can continue to have their process customized by using witadmin.

Current iteration query token

This feature gives you the ability to specify a token that represents the current iteration in iteration-based queries. As you know, iterations have dates associated with them. As you move from iteration to iteration, it's very tedious to update all the queries used to track work for the next iteration. This release brings the addition of a new query token, @CurrentIteration, that returns the current iteration based on today's date. There are some limitations with this new token, however. For example, it doesn't work in Excel. The token relies on understanding your team context, and unfortunately Excel doesn't have all the information needed to determine which iteration is current. Learn more about the current iteration query token.

Query progressive disclosure

Now, large query lists are not opened every time the query pane is shown. Only the first two levels are loaded, and then you can load the remaining levels on demand.

Branch policies

To help teams using Git improve the quality of code going into their repo, we've added a new capability to set policies on branches. These new policies enable teams to configure requirements for their development branches

that are enforced by the server when pushing or merging pull requests. You can prevent build breaks by using the build policy to require that all changes entering a branch must pass a configured build. You can also use code review policies to set a minimum number of reviewers for pull requests, or even require specific users to review changes made to specific portions of your codebase.

Branch policies – Gated build

Git projects can now set branch policies to require a successful build before any code can be submitted into a branch. Enabling the build policy will require the use of pull request to submit changes into the configured branch, and completion of the pull request is gated upon the successful outcome of the configured build.

Branch policies – Code review

Git projects can now set branch policies to require code reviews for any code submitted into a branch. Enabling the code review policies for a branch will enforce that all code must be submitted to that branch using pull requests. The policies provide options to require a minimum number of code reviewers, as well as to require specific reviewers for particular paths and/or file types.

Branch history (pushes & pull requests)

In the web portal, the History hub under CODE has been updated to support a new view for Git projects. The new "Branch Updates" view shows all of the updates for a given branch, and groups commits by Push and Pull Request activity. This view provides developers a new insight into how their Git repo is being updated over time, and provides traceability from History to Pull Requests.

Web history view for Git projects

The code history hub has a new view: Branch Updates. This view—which is available only for Git projects—shows all updates for a given branch, and groups commits by push-and-pull request activity. This view also provides developers with a new insight into how their Git repo is being updated over time, and provides traceability from history to pull requests.

Quick code editing

In this release, we added the ability to make a quick edit to a file in version control directly from your web browser and then commit those changes straight back to the service. When browsing a source file, you now have an Edit command that puts the file into editing mode. Changes can then be made inline, complete with color coding and formatting support. As soon as you click the Save command, we create a new commit/changeset with your changes. Use the diff view to see exactly what changes you're making before committing the changes. If the file is a Markdown or HTML file, you can also preview your changes before you save them.

Not only can you edit files, but we also added the ability to add, delete and rename files directly from the web. To add a new file (or files), right-click a folder in your repository, select

Add file(s), enter your check-in/commit comment, and you're ready to go. The days are gone when you have to download your entire codebase just to rename or delete a file.

The new editing capabilities also show up in the Welcome hub, making it easy to create documentation for your projects. If you don't have a README.md file, you can start with our template guide and commit your own.

We also enabled the ability to create links to existing (or new) markdown files by following the syntax. Don't worry if the page doesn't exist, because you can edit and commit the new file when you click the link, wiki-style.

With these new features, we hope that you find creating and editing your project documentation to be easy and fast!

Learn more about the quick code editing.

History control

We have optimized the history control to make discussions easier to read. Specifically, we've reduced the vertical space required so that you can get to the discussions you want to see more quickly, and we've done this without reducing functionality.

View history on a folder

Now, you can right click on any folder in Solution Explorer, the Changes page, or the Commit Details page, and get the history of changes to files within that folder.

Build automation system

Team Foundation 2015 includes our new build automation system. To learn more about the new build automation system, click the link in the message header, which is in the Build tab.

Team project rename

We've enabled the ability to rename a team project. All of your version control paths, work items, queries, and other team project artifacts are updated to reflect the new name. Team projects can be renamed multiple times and older names can be reused as well.

Learn more about renaming a team project.

REST APIs

This is the first release that brings REST APIs to on-premises TFS. JSON REST APIs enable a lightweight way to work with Team Foundation Server from virtually any device, platform, or technology stack, including Windows, Android, iOS, Node.js, and others. You can create and query work items, queue a build, get recent team room messages, access source code, and accomplish almost any team or code management task.

Service hooks

You can use service hooks to let your app or service get notified instantly when an event happens in Team Foundation Server. With service hooks, your app or service can avoid continuously polling to check for changes, such as completed builds, commits/check-ins, or work item changes. Now you can create powerful integration scenarios where Team Foundation Server can inform another service of a change, thereby enabling the use of both services together. You can find services hooks as a new hub in project administration.

How service hooks work: A service hook subscription controls what action to perform on a target, external service when a specific type of event happens. Similarly to an e-mail alert subscription, a service hook subscription is associated with the user who created it. When an event occurs and a service hook attempts to match a configured subscription to an event, a permission check is performed to ensure the user who created the subscription has permission to access to the resource associated with the event. For example, a user (likely a project administrator) creates a service hook subscription that is triggered on all "work item created" events. When a new work item is created under an area path that this user does not have access to, the permission check will prevent the subscription from matching and therefore avoid any external notification from being sent through this subscription.

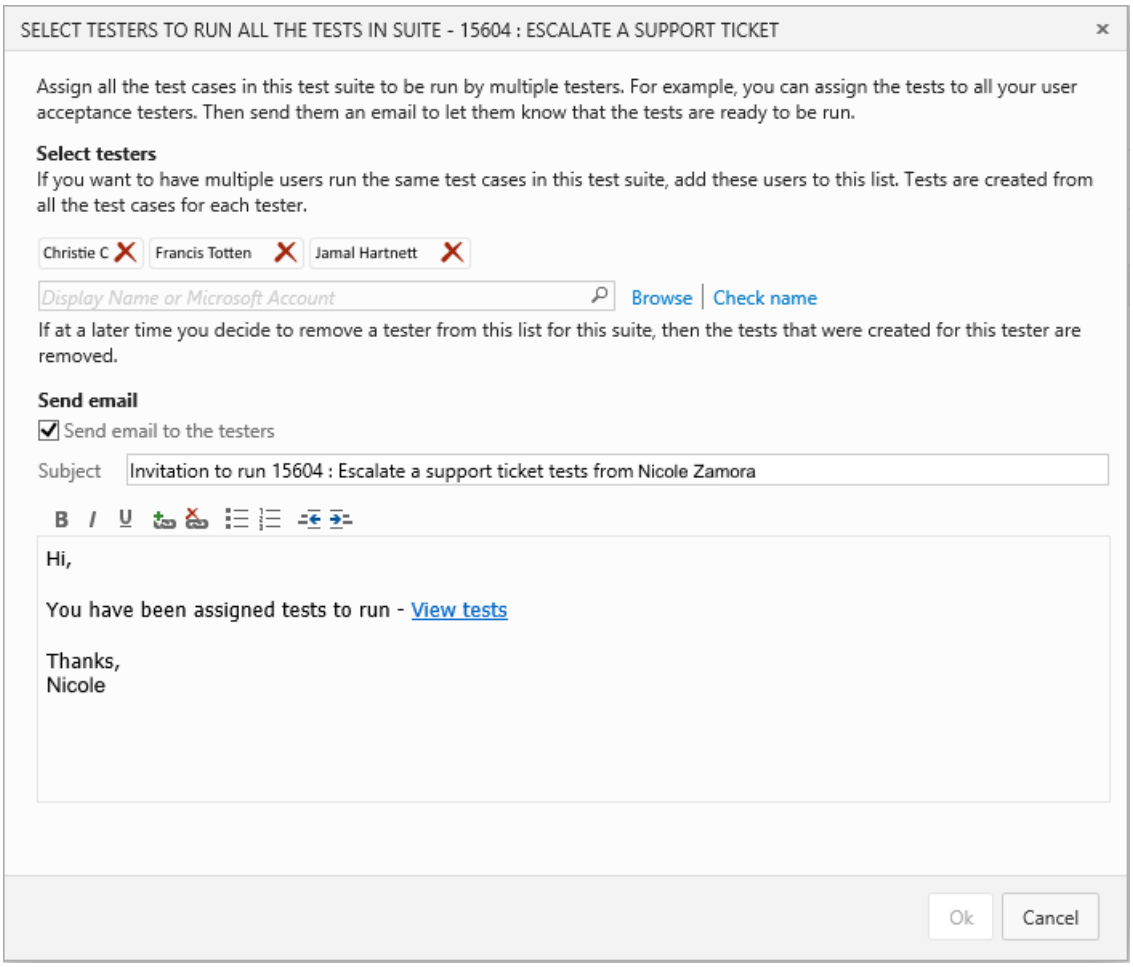
However, because service hooks make it easy to integrate with external services (like Trello or Campfire), you should make sure that the data that the creator of a subscription has access to is not made available to other users who might not have the same level of access. For example, a subscription that is defined to send all "code push" events to a Campfire room could result in information being improperly disclosed to users who do not have access to the repository associated with the event (but would be able to see the information because they have access to the Campfire room).

Improved merge performance

We have improved merge performance, which is especially apparent on large repos.

Assign multiple testers and invite them for testing

If you have a scenario in which you must invite multiple sign-off owners to run the same set of test cases, you can now assign multiple testers to a test suite. Doing so will pick each test case in the test suite and create a test for each tester who you add to the test suite. You can also send an email inviting them to run tests. When a tester clicks the "View tests" link in the email, a test plan opens that includes a filtered list of tests assigned to that tester.

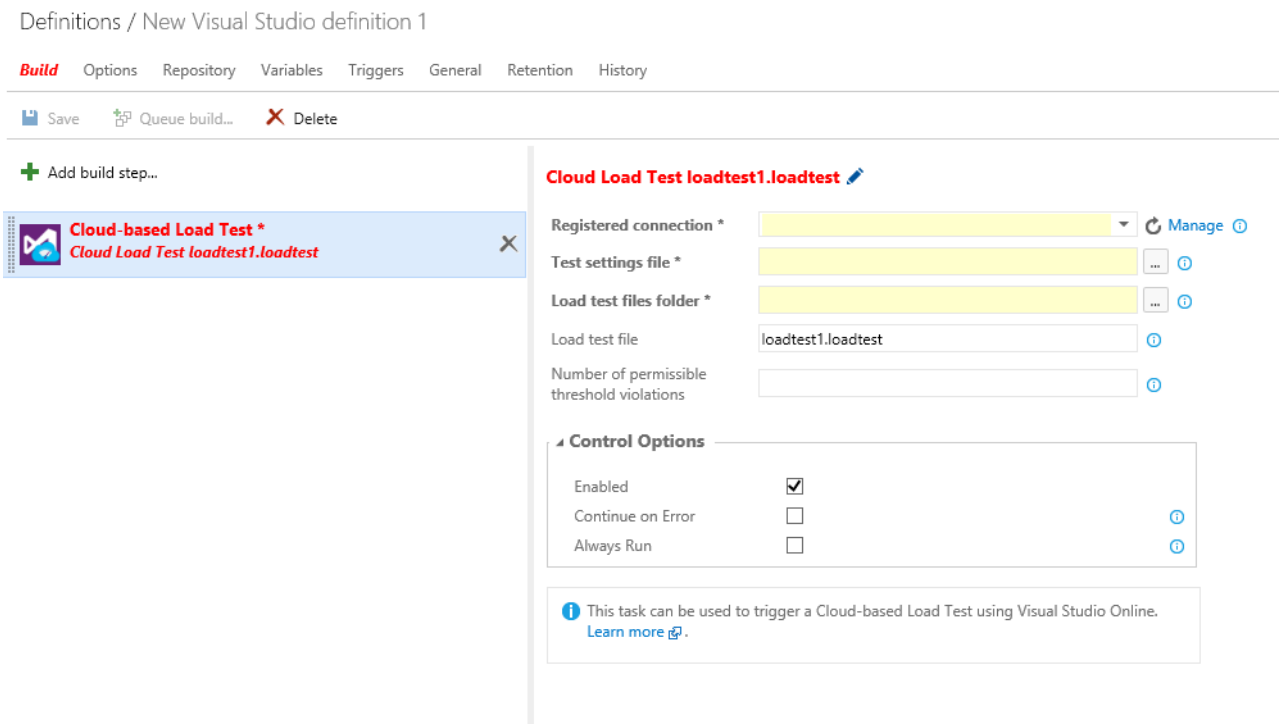


Cloud-based load tests

We are announcing new capability for running Cloud-based load tests as part of the new Build system. There are two parts: A cloud-based Load Test and a Cloud-based Web Performance Test:

list-style-type: none;

- **Cloud-based Load Test** allows you to execute an existing Load test project as part of your CI/CD pipeline.
- **Cloud-based Web Performance Test** executes a simple load test against an App URL, with the basic load test parameters being configured in the task itself.



Automated testing

We are announcing new capabilities for running unit tests on the build machine, functional automation runs on remote machines, and browse test results as part of the new Build system. Read more details in this blog.

There are now two test tasks - Visual Studio Tests (VSTest Task) and Visual Studio Tests using Tests Agents (VSTest Remote Task).

- **VSTest Task** runs tests locally on the build machine, and typically, you'll run only unit tests with this task.
- **VSTest Remote Task** runs tests on remote machines in a distributed fashion by using test agents, and you can run either functional, unit tests or UI tests.

Additionally, we are announcing a Machines hub, and a Runs tab in the Test hub.

- **Machines hub**. You'll use the Machines hub to create and manage remote machines.
- **Runs tab**. This tab in the Test hub acts as a a single repository for all test results in the system. You'll not only be able to browse automated test results from VSTest and VSTestRemote tasks, but also from legacy workflows such as XAML Builds and Build- Deploy-Test workflows. Additionally, you can choose to integrate publishing test results into you custom tasks by leveraging REST APIs to publish Test Results. The 'Runs' hub today supports querying on test runs and test results, assigning owners to test failures, tracking their analysis, and filing bugs.

Change in API behavior

The name, description, and metadata arguments values of the API for IProcessTemplates.AddUpdateTemplate Method are now overridden by the process template data specified in the zipFileName. The reason for this change is to avoid conflicts between what's in the ZIP file and what's passed in as a parameter to the API.

IProcessTemplates.**AddUpdateTemplate** Method

```
void AddUpdateTemplate(
        string name,
        string description,
        string metadata,
        string state,
        string zipFileName
)
```

The following screenshot shows where these properties are defined in ProcessTemplate.xml.

```
<?xml version="1.0" encoding="utf-8"?>
<ProcessTemplate>
  <metadata>
    <name>Agile</name>
    <description>This template is flexible and will work great for most teams using Agile planning methods, including those practicing Scrum.</description>
    <version type="ADCC42AB-9882-485E-A3ED-7678F01F66BC" major="14" minor="1" />
    <plugins>
      …
    </plugins>
  </metadata>
  <groups>
```

Bug Fixes & Known Issues

For a complete description of technology improvements, bug fixes, and known issues in this release, see the following KB article.

Feedback

We would love to hear from you! You can report a problem and track it through Developer Community and get advice on Stack Overflow. As always, if you have ideas on things you would like to see us prioritize, head over to UserVoice to add your idea or vote for an existing one.

Top of Page

Team Foundation Server 2015 Release Notes History

9/7/2018 • 2 minutes to read

Release Notes History

- Team Foundation Server 2015 Update 4 Release Notes
- Team Foundation Server 2015 Update 3 Release Notes
- Team Foundation Server 2015 Update 2 Release Notes
- Team Foundation Server 2015 Update 1 Release Notes
- Team Foundation Server 2015