



Microsoft

# Visual Studio 2012 Product Guide

 Visual Studio





This document is provided "as-is". Information and views expressed in this document, including URL and other Internet website references, may change without notice. Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred. You may copy and use this document for your internal, reference purposes.

This document is provided for informational purposes only, and Microsoft makes no warranties, expressed or implied, with respect to this document or the information contained in it.

© 2013 Microsoft. All rights reserved.

Microsoft, Direct3D, DirectX, Excel, Hyper-V, IntelliSense, IntelliTrace, PowerPoint, SharePoint, Silverlight, SQL Server, Visual Basic, Visual C#, Visual Studio, Windows, Windows Vista, Xbox, Xbox 360, and XNA are trademarks of the Microsoft group of companies. All other trademarks are property of their respective owners.

Microsoft Visual Studio 2012 is an integrated solution that enables individuals and development teams of any size to turn their ideas into exceptional, compelling applications. It enables all stakeholders involved in software delivery to take advantage of state-of-the-art tooling to create outstanding experiences that delight the end users of your consumer and business applications with a simplified development solution that enables quality through all the tasks and roles involved in a software project.

With Visual Studio 2012, you enhance your team's ability to implement state-of-the-art developer practices that are adapted to your team's optimal rhythm. It provides workflows and tools that shorten delivery cycles, includes customers and operations in software construction, and eliminates waste. As a result, you reduce risks, solve problems faster, and continuously deliver value that exceeds customers' expectations.

This Visual Studio 2012 Product Guide provides a comprehensive overview of scenarios and features regarding:

- The new integrated development environment (IDE)
- Windows 8 development
- Web development
- Cloud development
- SharePoint development
- Application Lifecycle Management tools

This guide also provides information about the benefits that Visual Studio 2012 offers to diverse stakeholders in software delivery projects, including testers and operations staff.

# Contents

## Introduction to Visual Studio 2012 8

Visual Studio 2012	11
Deploying Visual Studio 2012	13
Visual Studio 2012 Product Family	14
What's new in Visual Studio 2012	18
Developing to standards	18
Summary	19

## Visual Studio 2012 Developer Experience 20

New User interface	21
Project Compatibility	21
Managing the Environment	22
Getting Started with Community Code and Extensions	30
Platform Experience	35
Services Experience	45
Summary	49

## Developing for Windows 8 50

Windows 8 Applications are Simple to Use	51
Windows 8 Applications Can Easily Connect and Communicate	53
Windows 8 Applications Can Adapt to Their Environment	54



Windows 8 Applications are Device Friendly	54
Windows 8 Applications are Mobile and Scalable	55
Windows 8 Applications are Trustworthy and Safe	56
Windows Runtime and application model	57
Building Windows Store applications with C++ by Using Visual Studio 2012	71
Designing Games for Windows 8	75
Summary	77

## Developing for the Web 78

Develop to Standards	79
Find Problems Faster	86
Develop for a Modern Platform	89
Develop Web services	93
Summary	99

## Developing for the Cloud 100

Azure Roles	101
Coding for Azure	107
Windows Azure SQL Database	112
Publishing to the Cloud	113

Building Hybrid Cloud Solutions with Windows Azure	117
Profiling and Tracing Web Applications and Services in the Cloud	119
Enabling Continuous Delivery in the Cloud	120
Building and Deploying Applications to the Cloud by Using Visual Studio LightSwitch	121
Summary	127



## Developing for SharePoint 128

Accelerate SharePoint Site Development	129
Easily Incorporate Silverlight Web Parts into SharePoint Sites	133
Quickly Reuse Items from Existing SharePoint Sites	135
Optimizing and Profiling SharePoint Code	137
Unit Testing Your Code and Mocking SharePoint Components	138
Summary	141



## Application Lifecycle Management with Visual Studio 2012 142

The Benefits of Visual Studio 2012 and Team Foundation Server 2012	144
Increasing Efficiency and Reducing Waste	145
Application Lifecycle Management Features in Visual Studio 2012	149

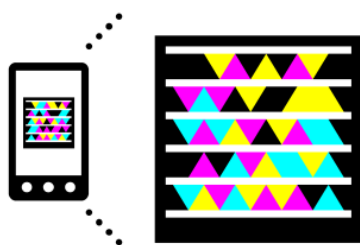


Visual Studio Team Foundation	
Server 2012 in the Cloud	186
Support for Heterogeneous Environments	191
Conclusion	196

## Visual Studio 2012 for Other Project Stakeholders 197

Product Managers and Designers	198
Project Managers	199
Team Leaders and Architects	200
Customers	200
Testers	202
End-user Developers	204
Conclusion	207

## Feature Matrix 208



**Get the free mobile app at**  
**<http://gettag.mobi>**

This product guide includes Microsoft Tags to make it easy for you to access links from your smartphone or tablet device. You can scan the tags by using the free reader from <http://gettag.mobi>. Microsoft provides a tag reader which reads both Microsoft Tags and QR codes for most mobile platforms, including Android™, Blackberry®, iPhone®, Java® ME, Symbian®, Windows Mobile, and Windows Phone.



# Introduction to Visual Studio 2012

After releasing Visual Studio 2010, the Visual Studio team took a step back to look at the significant trends in the software construction world and to absorb ongoing feedback from customers. Some important trends became apparent. In a bid to differentiate themselves and increase their competitive advantage, businesses were investing more in custom software development than in off-the-shelf solutions. There has been an explosion of innovative software targeted at consumers. Development has also become even more of a team sport, involving many more non-developer stakeholders and requiring more advanced development practices. Finally, there has been an explosion of new platforms, even as web development becomes more critical to successful software delivery.



Consequently, the Visual Studio team wanted to ensure that Visual Studio 2012 included the tools needed to help users to quickly develop amazing applications for both businesses and consumers.



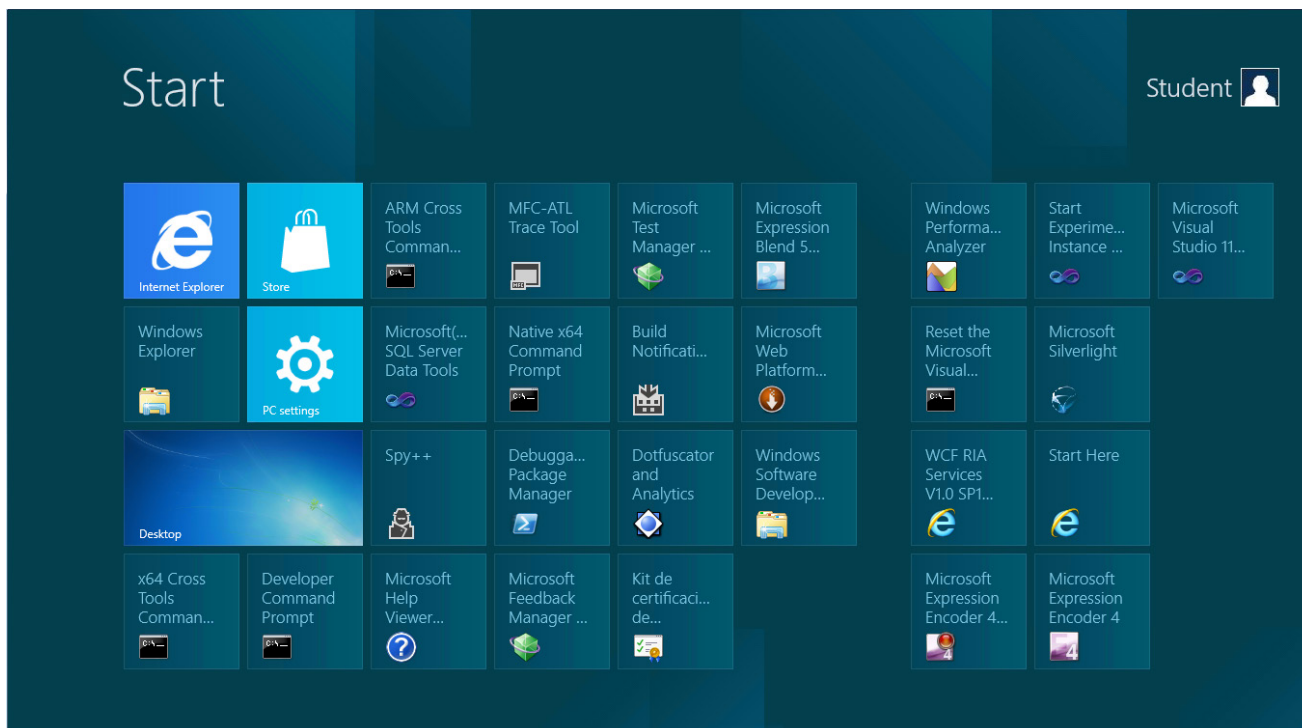
Businesses increasingly use custom software to stay competitive. Stakeholders at every level are now critical to ensuring that an application is successful. End users also must be involved in order to ensure that the application is functional, efficient to use, and easy to learn. Visual Studio 2012 and Visual Studio Team Foundation Server together make it easy to include participants from within and outside of the business during the development process.

Software development is shifting from being primarily an enterprise-driven process to incorporating a consumer focus, resulting in the emergence of a plethora of applications aimed at the consumer market for new device platforms. The explosion of these new device platforms is another major new challenge for developers.

Previously, most applications ran either on a server or on a desktop. Today, many other platforms are common; smartphones and tablet devices are becoming ubiquitous. Developers must either create applications that work across multiple platforms with a consistent user experience, or make applications tailored to each platform but

implementing consistent business logic. They also need to create a connected experience that allows end users to move seamlessly among platforms and devices.

Visual Studio 2012 makes it easy to develop applications for multiple platforms, including Windows Phone smartphones, slate devices running Windows (including the new full-screen tailored applications for Windows 8), the Xbox 360 console, and of course on traditional desktop computers. Portable libraries enable developers to write code once, compile it once, and deploy the compiled assembly everywhere. With Team Foundation Server, it is easy to work on alternative platforms and still maintain control of source code and project timelines, and to integrate a disparate set of third-party development tools.

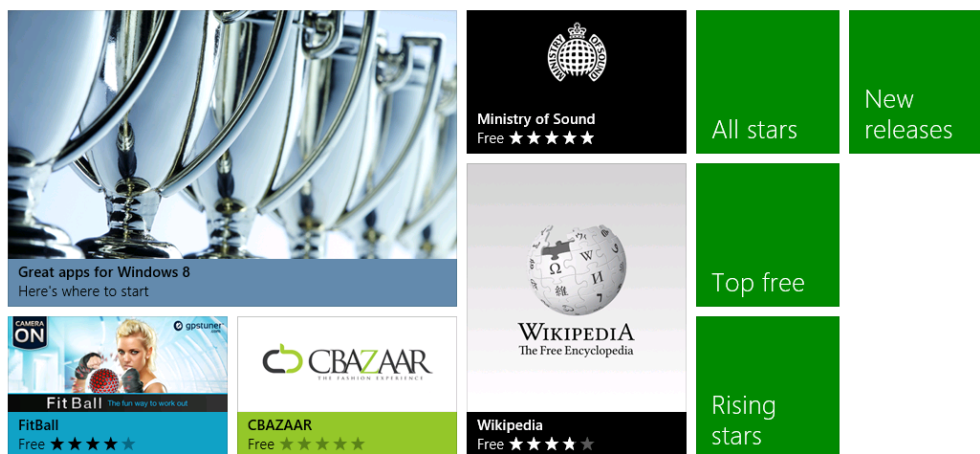


Another major change is the introduction of app stores. Often tied to a particular platform (or at least to a specific platform manufacturer), app stores enable consumers to quickly search for and acquire an application. For example, Windows Phone has the Windows Phone Marketplace and Windows has the Windows Store. These stores provide significant opportunities for software developers to reach markets that had previously been difficult to penetrate.

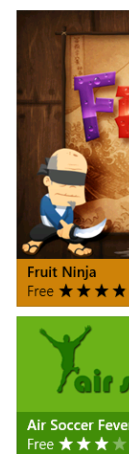


## Store

### Spotlight



### Games



Applications can be monetized, either by charging for the application or by including advertising in the application. Visual Studio 2012 provides integrated tools to publish software to these stores and to expose applications to a large and potentially lucrative market.

## Visual Studio 2012

Visual Studio has been designed to ensure that developers can provide a continuous flow of value to the business. The interface has been revamped to remove significant clutter from the screen, while still providing fast access to frequently used features. Removing unnecessary distractions helps developers remain focused on the key development tasks.

The software development ecosystem is evolving, and Visual Studio supports not only developers but also many other people involved in the software

development process. Now stakeholders become part of the process early on, and as requirements change, Visual Studio enables you to capture those changes and to rapidly update your application. Productivity enhancements help developers to work faster. Visual Studio now builds applications more quickly and includes new and enhanced tools that make debugging easier.

With the advent of new platforms and new distribution channels, new challenges and new opportunities arise. Applications often must be developed for many platforms, each with subtly different requirements. Visual Studio helps developers to create applications that span device boundaries. Project templates, debugging tools, and portable code libraries all help to make this process easier. Visual Studio even enables you to upload your application directly to the Windows Store.

Another significant change is the increase in applications that use and implement services to access data. From games with scoreboards to weather forecasts and traffic reports, many applications use services to provide up-to-date and relevant information. Applications can easily go viral and very literally be an overnight success. If this happens, you need to ensure that your services providing the data can cope with the sudden influx of requests. Visual Studio supports developers by enabling them to create solutions that are highly scalable, supporting a range of models from durable full duplex synchronous services to high-throughput asynchronous services.

Developing a scalable solution is only half the challenge: You must also provide a suitable environment to act as a host. With Visual Studio you can develop for a range of infrastructure choices. Whether you choose to target a physical server, a virtual server, or a private or public cloud, Visual Studio is your solution. If you decide to use the Windows Azure public cloud offering, you can deploy directly from Visual Studio to the cloud. As your application matures, you can perform in-place updates, and when your application goes viral you can easily scale out your application.

By enabling developers to focus on the code, involving stakeholders both within and outside the business, and by making deployment easy, Visual Studio helps to ensure that your software development team continuously delivers business value.

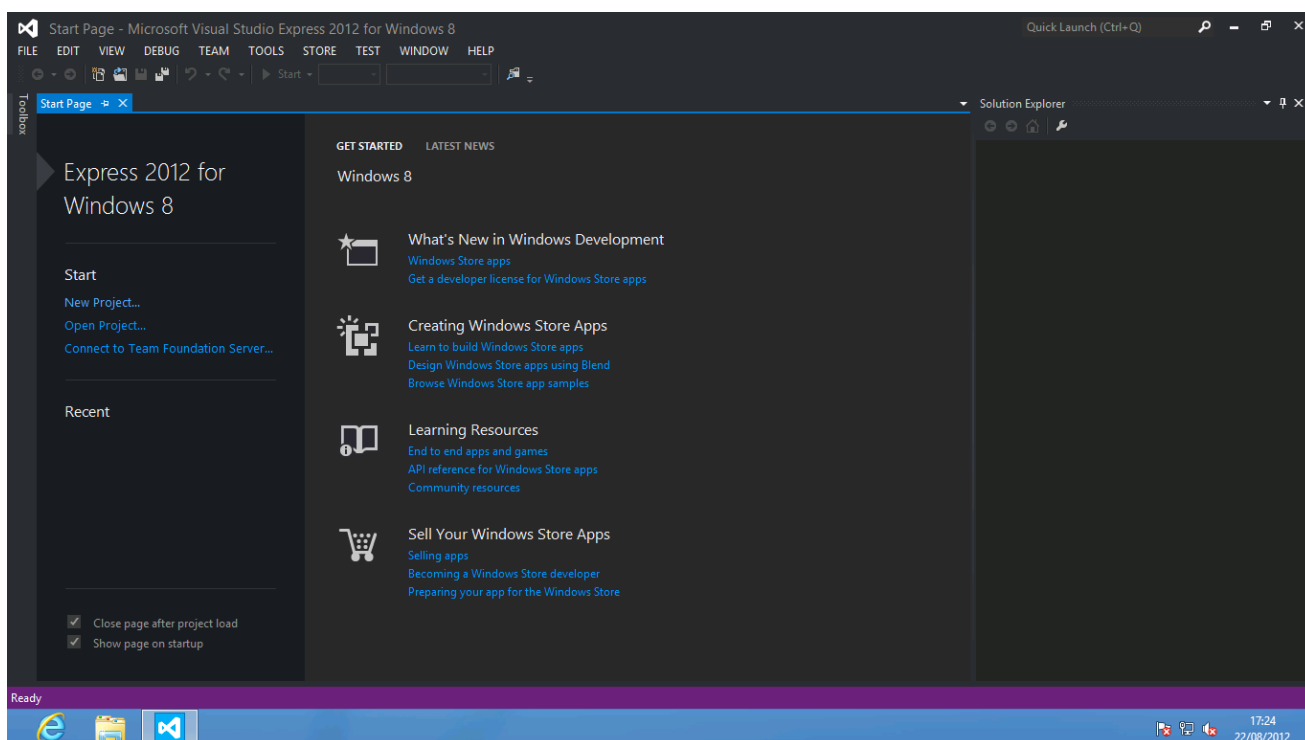


# Deploying Visual Studio 2012

Visual Studio 2012 is now easier than ever to purchase, install, and upgrade.

We have worked hard to make Visual Studio 2012 projects backwards compatible, making Visual Studio easier to deploy. When you open an existing project by using Visual Studio 2012, in many instances no changes are made to the project. Where changes are necessary, they are backwards compatible with Visual Studio 2010. This enables members of the team to continue using Visual Studio 2010 while they collaborate on a project. Rather than forcing all developers on the team to migrate simultaneously, they can upgrade to Visual Studio 2012 when it is most convenient.

Developers wishing to create applications for Windows 8 can take advantage of the free version of the product, Visual Studio Express for Windows 8.



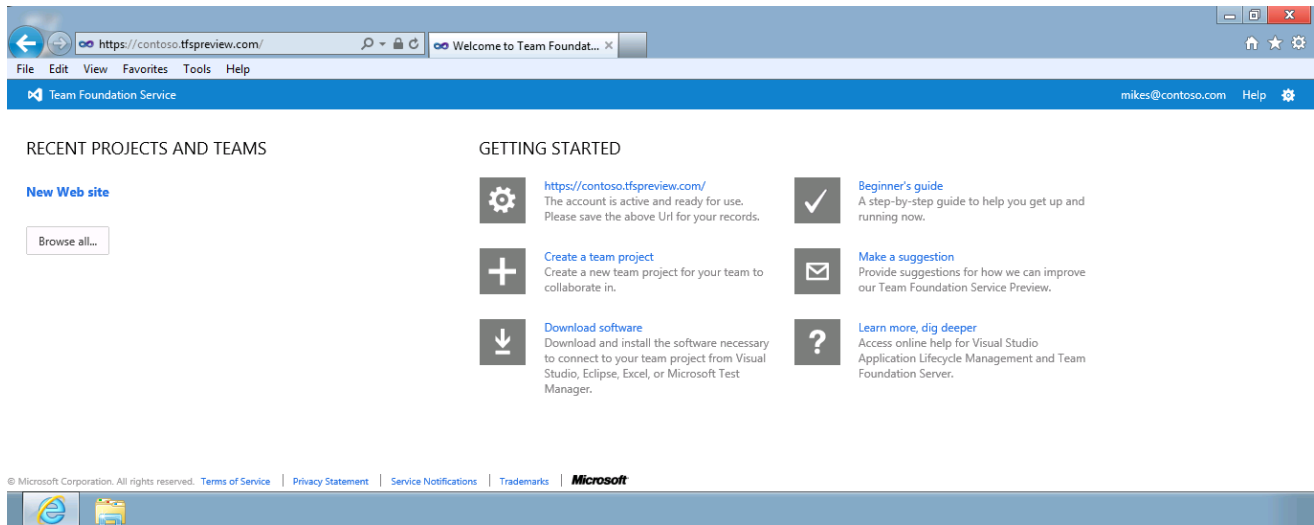
In addition to making Visual Studio faster, this release includes new tools that enable teams to create high-quality software, and ensure that all stakeholders remain involved in the development process. Visual Studio, together with the other tools in the Visual Studio 2012 product family, encourages agile development for rapid stakeholder feedback.

## Visual Studio 2012 Product Family

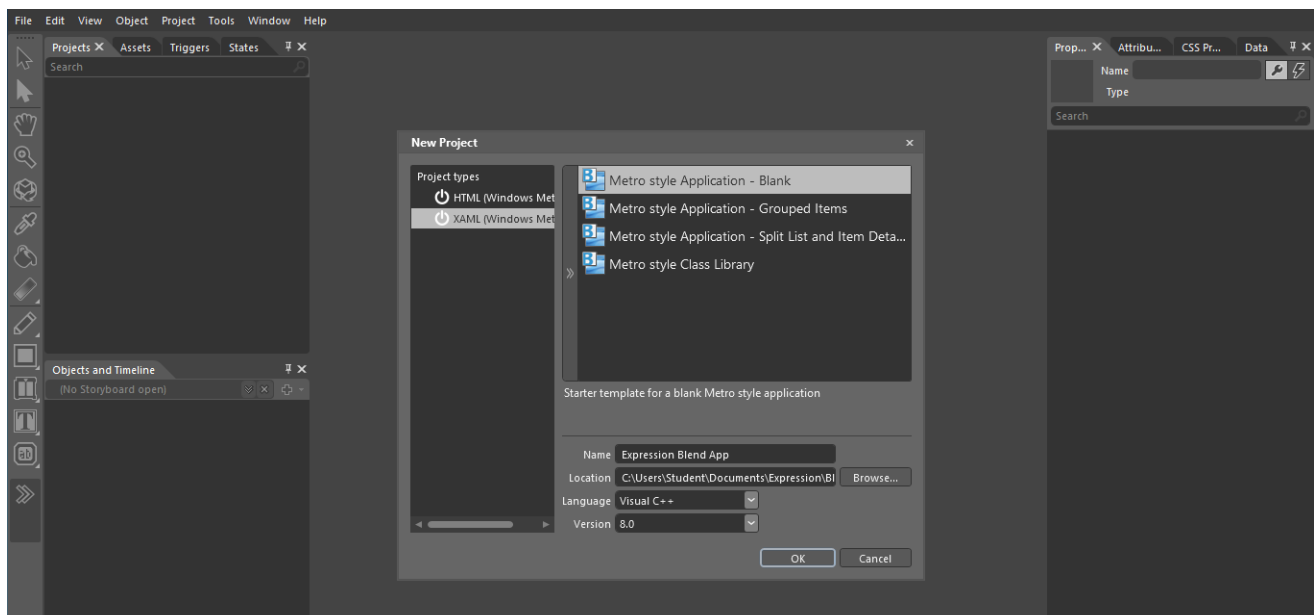
Creating software involves many other team members besides developers. Agile methodologies enable stakeholders such as architects, product managers, designers, program managers, testers, operations personnel, and end users to have much greater input into the application development process.



[Visual Studio Team Foundation Server 2012](#) enables all stakeholders to participate as equal members of the development team throughout the application lifecycle. New connectors and integration points enable operations staff and project managers to work more closely with the development team to realize faster delivery cycles and to add real business value. Visual Studio Team Foundation Server 2012, in common with previous versions of Visual Studio Team Foundation Server, is available for installation on a local server. However, there is now another option. Team Foundation Service is available as an online software-as-a-service (SaaS) version of Team Foundation Server.



It is a fully hosted solution offering easy configuration and a relatively small upfront capital expenditure. This gives organizations that previously did not want to invest in or maintain a local server the opportunity to take full advantage of the capabilities of Visual Studio Team Foundation Server 2012.

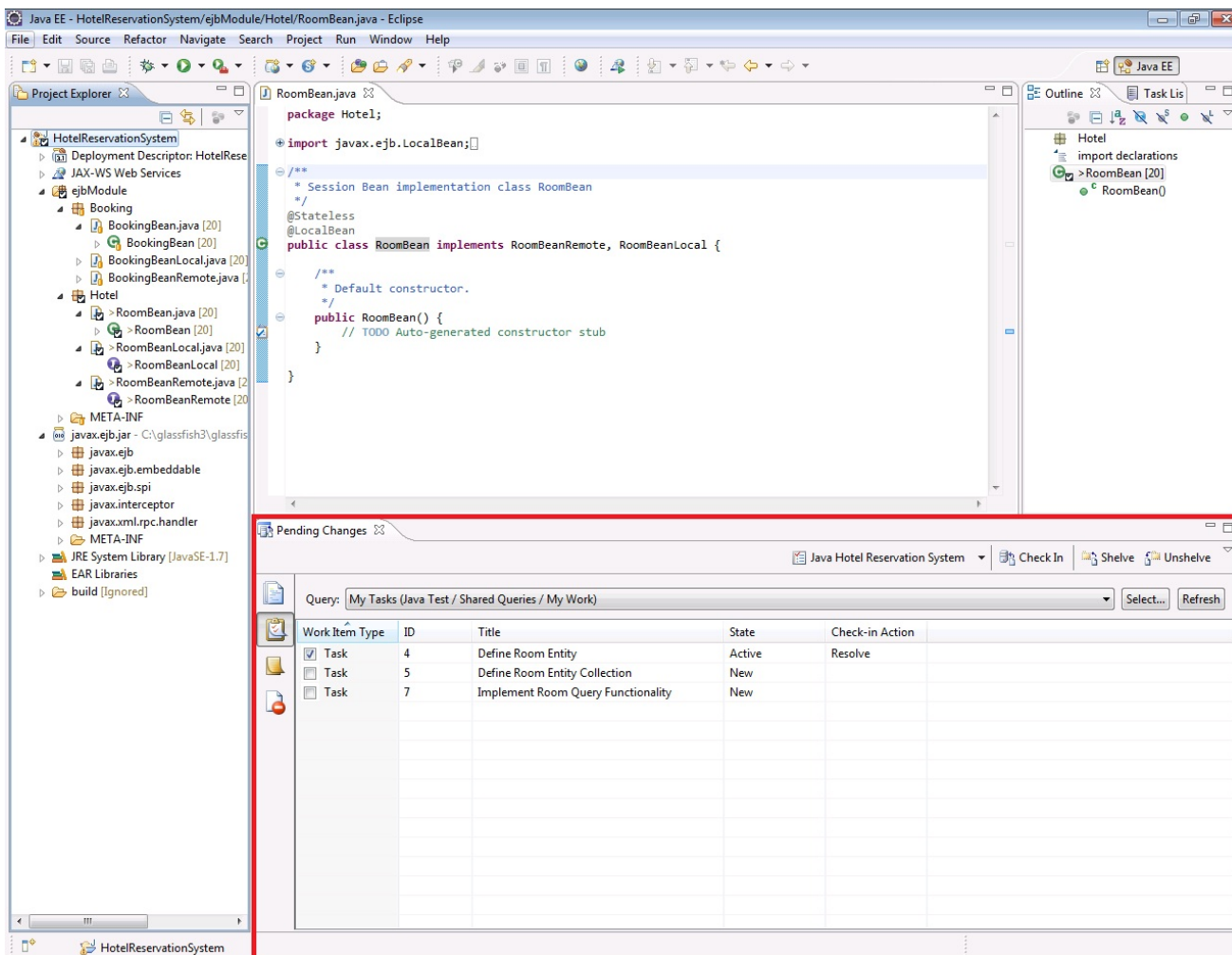


Blend for Visual Studio 2012 enables designers to create visually stunning applications. Applications tailored to Windows 8 can use XAML to define the user interface, ensuring that designers can continue to use their existing design skills. Portability between Visual Studio 2012 and Blend has been improved, and the design surface in Visual Studio has



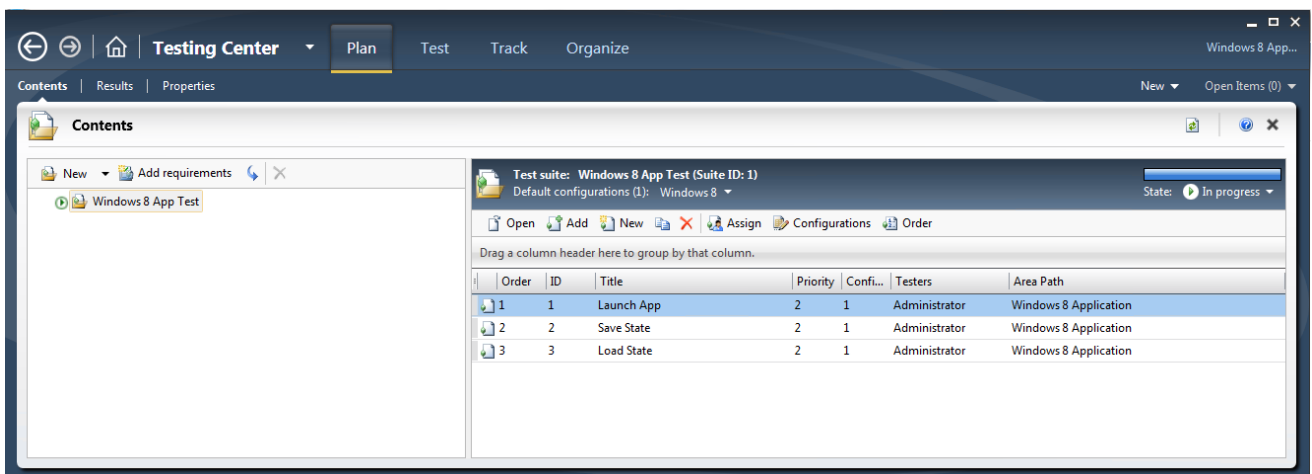
been upgraded to more closely match the experience in Blend. These improvements make it easier for designers and developers to work together to make visually stunning, highly functional applications.

Visual Studio 2012 and Visual Studio Team Foundation Server 2012 are designed to work together. However, some teams may be using a different development environment, such as Eclipse™. Visual Studio Team Foundation Server 2012 provides adapters to connect to supported IDEs so that developers can work in their preferred development environment. These adapters also work with Team Foundation Service.



Visual Studio Test Professional 2012 helps to ensure that the testing process is fully integrated into the software development lifecycle. With Test Professional, high-quality software testing is now a preeminent part of the development process. Test plans and testing can be performed more quickly by using the same tools as developers. Test Professional provides many features to make testing applications easy, including:

- Support for application testing on remote devices without installing Visual Studio 2012 or Test Manager on the device itself (a small test agent must be installed on the device).
- Reusable software test plans that include tests and configurations.
- Manual test plans that test managers can use to collect data for bug reports and for visualizing project progress against the test plan.
- Exploratory testing to enable testing without requiring a formal test plan, by simply using the application and Test Manager to record and document actions, attach screenshots, record steps, and create a test case to ensure that any bugs that are discovered are addressed.
- Automated testing in a virtual or physical environment, and performed on a variety of devices, spanning platforms as necessary.



## What's new in Visual Studio 2012

Visual Studio 2012 includes support for many of the platforms and environments in which code now runs.

### What's new for Windows 8

Windows 8 provides many new opportunities for developers. Developers who are familiar with the Visual C# or Visual Basic languages can use their skills to develop the new style of tailored applications for Windows 8. Web developers can employ JavaScript™, and the JavaScript libraries in Windows 8 enable web developers to leverage the full power of the new Windows Runtime available in Windows 8.

### What's new for the web

Developers can use HTML5, CSS3, and JavaScript to create modern web applications that render perfectly on many devices and platforms. Visual Studio 2012 includes full support for all of these languages, supporting script loaders for JavaScript and enabling the CSS workarounds used to support older browsers. In addition, ASP.NET web forms now provide client-side validation and object binding to make development faster and the user experience better.

For developing and debugging web applications, developers can take advantage of Internet Information Server (IIS) Express. This is a lightweight version of IIS, configured in Visual Studio 2012, which is suitable for testing your applications. Web applications that store data can use the new LocalDB, a lightweight version of SQL Server, with full fidelity to both SQL Server and SQL Azure, ensuring that you do not need to change any code when you deploy your application.

## Developing to standards

As platforms have evolved, software developers have faced many challenges to ensuring that their applications are portable. Developing to industry standards helps to ensure that applications work. Visual Studio 2012 provides first-class support for many development language standards.



## Summary

Visual Studio 2012 gives you the power you need to provide a continuous flow of value to your business. You can use the Visual Studio product family to involve stakeholders in projects early in the process. By developing in an agile style, you ensure that you continue to meet the needs of the business and therefore continue to add value. Visual Studio enables you to develop applications for a multitude of devices, which expands your potential base of users. You can also use Visual Studio to create highly scalable services for your applications and web sites. Support for a flexible infrastructure ensures that your applications and supporting services can grow with your user base.





# Visual Studio 2012 Developer Experience

Visual Studio 2012 incorporates thousands of improvements for developers that make it easier to be productive, to collaborate seamlessly within teams, and to focus on creating value for end users. The core development environment has been revamped and refined based on developer feedback. New technologies have been added to the .NET Framework that bring critical improvements to familiar languages, including Visual C#, Visual Basic, Visual F#, and ASP.NET, as well as adding new standards-based support for C++ and JavaScript. Of course, Visual Studio continues to provide a support environment for Python™, Ruby™, and many other languages. Visual C# and Visual Basic developers also benefit from an additional project template, enabling them to create Windows Runtime

components. Project templates have also been updated to support Process Lifetime Management (PLM) behavior, a behavior that enables applications to automatically save and load state upon being suspended or resumed.

Revamped MSDN subscription benefits mean that it is easier than ever to test your software on any target Microsoft platform. Redesigned developer centers provide a complete on-ramp to creating any kind of application and turning your ideas into great software.

## New User interface

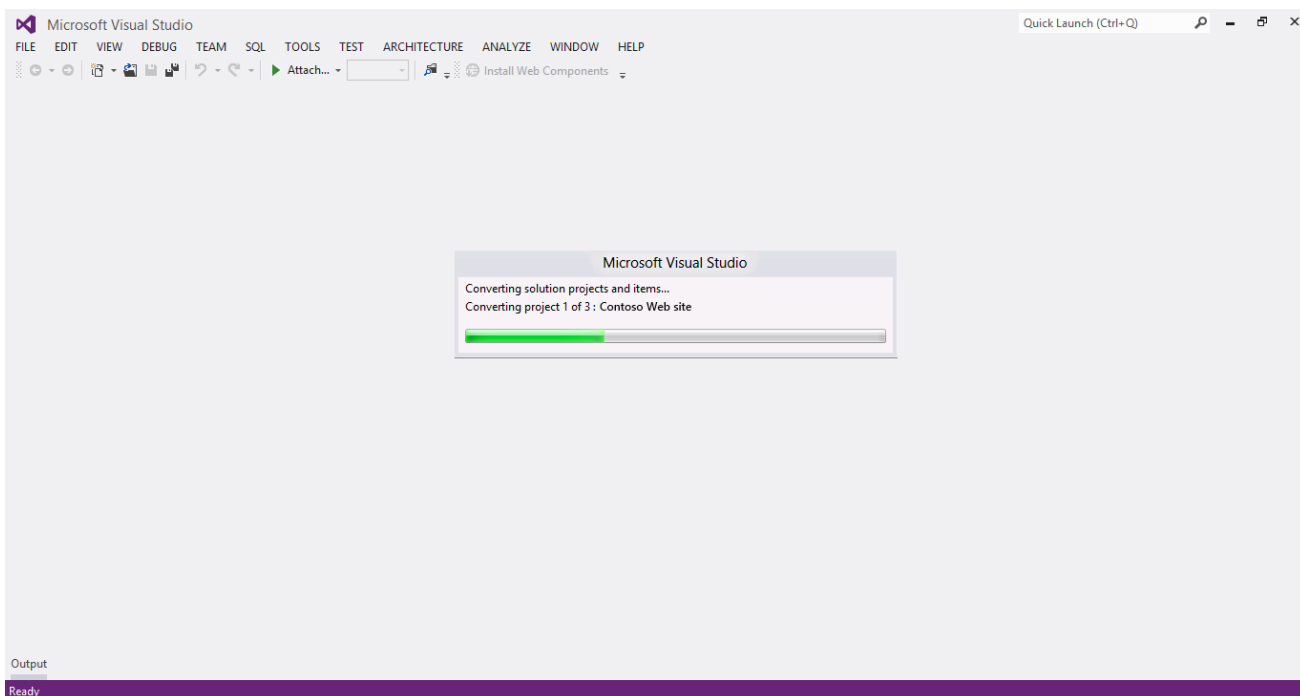
Visual Studio provides an environment in which developers can be more productive and work without interruptions caused by their tools.

The first change you will see in Visual Studio is the new developer user experience. The minimalist style of Visual Studio puts more emphasis on your code and subtly blends the application controls into the background. New dynamic toolbars and reduced UI chrome mean more workspace is available for viewing and exploring your code. Core elements such as Solution Explorer, Preview tabs, and History now have even more functionality to help you understand your code without opening up additional windows. When new extensions are available, subtle notifications give you that information without breaking the flow of productive work.

## Project Compatibility

Project compatibility enables developers working in different versions of Visual Studio to work together. Developers on a team do not need to upgrade simultaneously to the latest version of Visual Studio, and project compatibility ensures that all developers can continue working together. Moving to Visual Studio 2012 is a seamless operation that in most cases does not require you to modify your projects. Simply open a Visual Studio 2010 project in Visual Studio 2012 and begin working on your application. Furthermore, Visual Studio does not change the format of your solution or project files, which means that developers using Visual Studio 2012 can collaborate on the same projects as developers who continue to use Visual Studio 2010.

Some project types may require a conversion operation when migrating from Visual Studio 2010. In these cases, the first time you open the Visual Studio 2010 project, Visual Studio 2012 automatically performs the conversion. Thereafter, the project loads seamlessly and silently. This operation is fully backwards compatible, so that projects opened in Visual Studio 2012 will continue to work in Visual Studio 2010.



## Managing the Environment

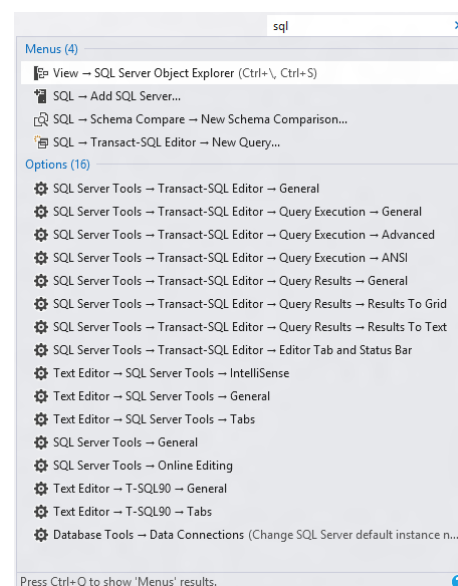
Managing your environment can help you become more productive. More demanding hardware and additional displays give you the power to work faster, but the ability to effectively manage your resources is critical. Visual Studio now includes features that help you take advantage of the power of modern computers, enable you to work seamlessly across multiple monitors, and quickly find what you are looking for.

## Search Everywhere

Visual Studio adds search functionality throughout the environment (code, menus, dialog boxes, and controls) to make finding what you are looking for fast. Searching is not limited to just your code, but extends across the entire environment to help you quickly locate the commands and functions you need. Below are just a few examples of how search is now integral to the IDE.

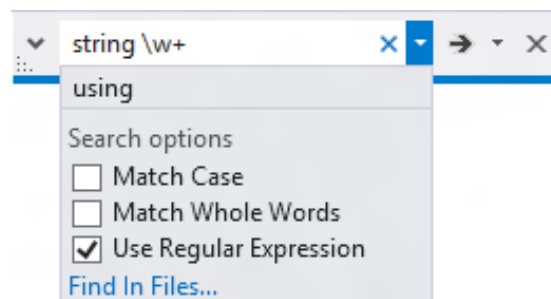
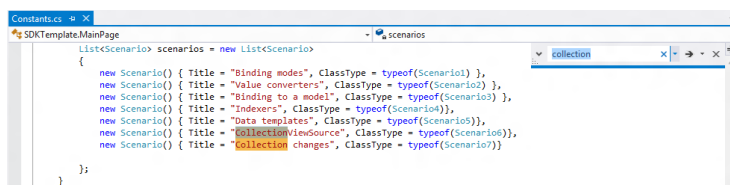
### Quick Launch

Visual Studio 2012 Quick Launch helps you locate commands. The Quick Launch dialog box is always available in the toolbar and is accessible via the Ctrl + Q shortcut. Specify a keyword in the search box to access the command that you want to use, and the Quick Launch dialog box displays all menu items in the Visual Studio IDE that match this keyword. To make searching for regularly used features even faster, recently used items are grouped at the top of the search results.



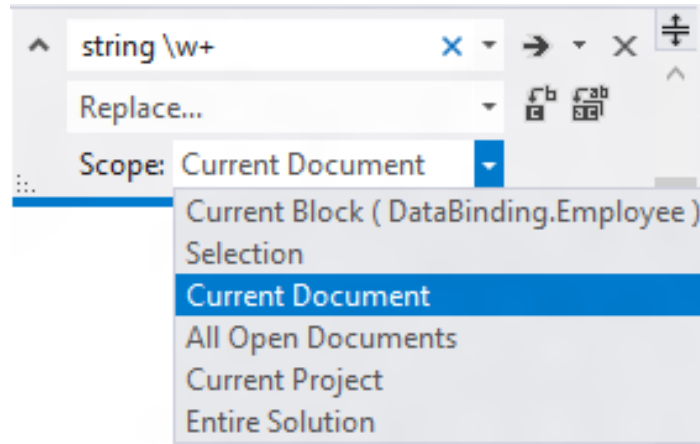
### Quick Find

Quick Find lets you search through your code. You can specify the search criteria as a string to perform simple matching, and you can also modify the search options to perform pattern matching by using regular expressions. Matching strings are highlighted in the Editor window, enabling simple movement between matches.





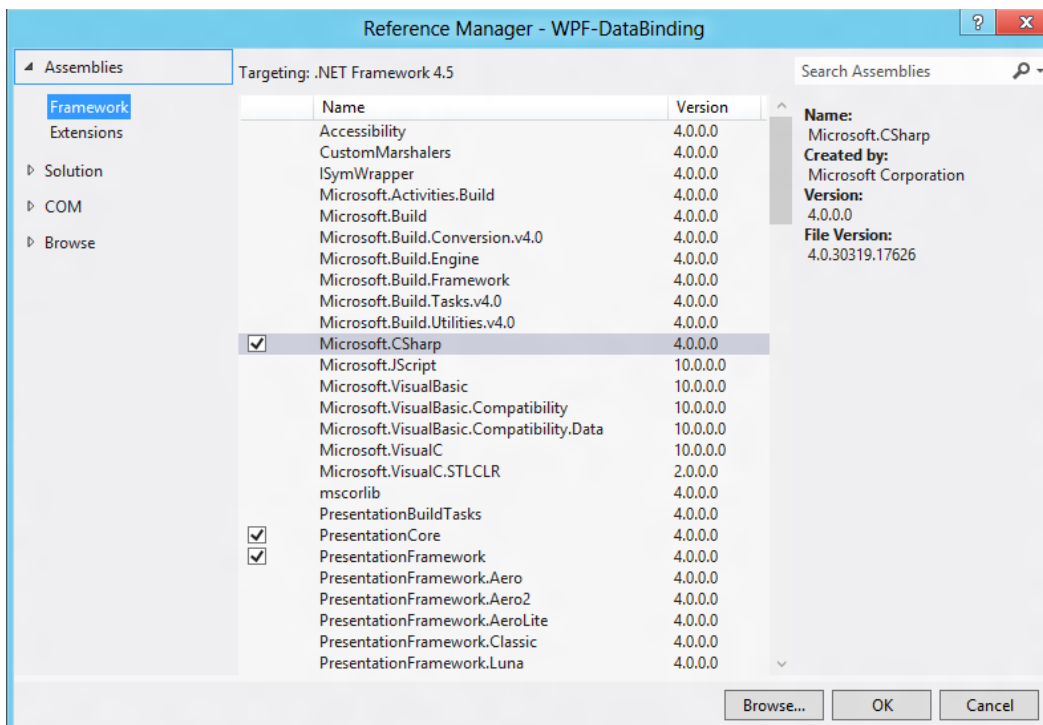
By default, Quick Find searches the current file. You can easily broaden the scope of the search to encompass all files. You can also perform a Find and Replace operation from Quick Find.



### Updated Add Reference Dialog

The Add Reference dialog box is used for adding assembly references to projects. It also intelligently caches references to .NET Framework assemblies and COM libraries for faster startup. Indicators identify which assemblies are already referenced in a project.

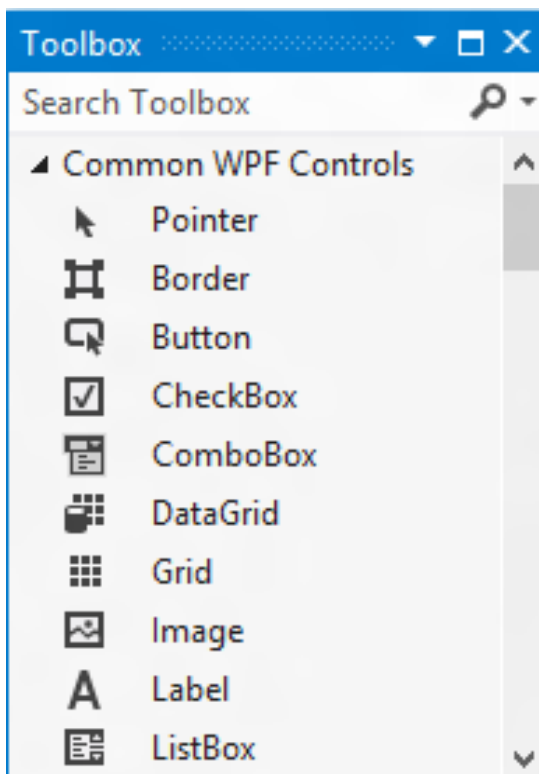
You can use the Search Assemblies dialog box to quickly search all assemblies.



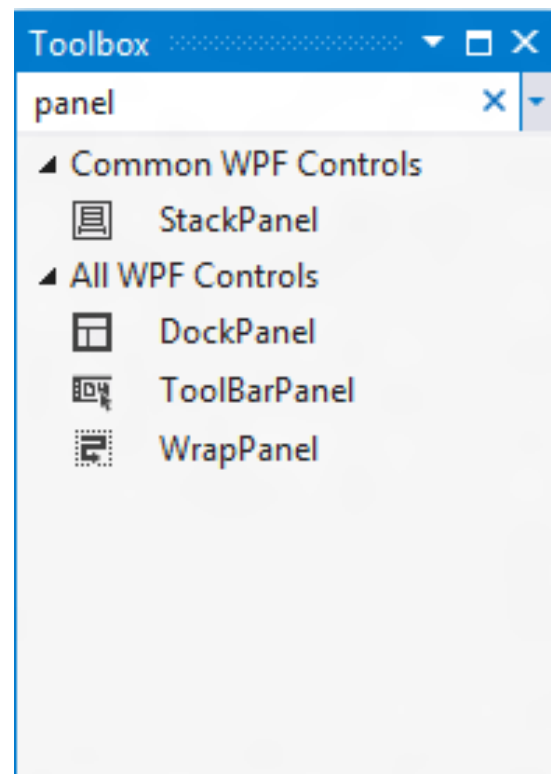
## Search Enabled Toolbox

The toolbox is searchable to provide fast access to controls during development.

Before:

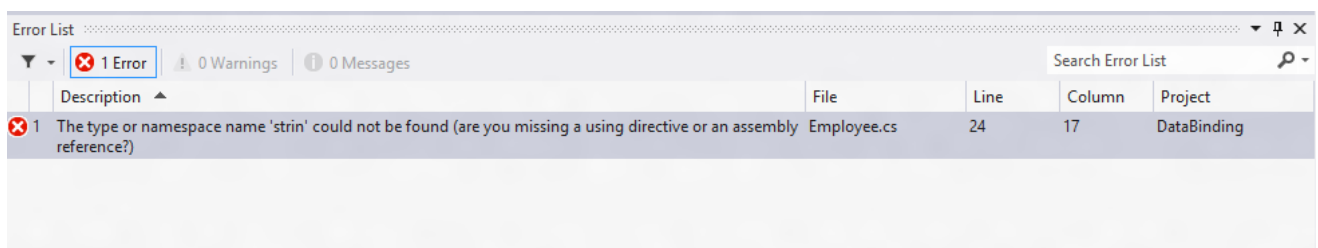


After:



## Error List

An enhanced error list enables developers to find related compile-time errors or to quickly check for a specific error.

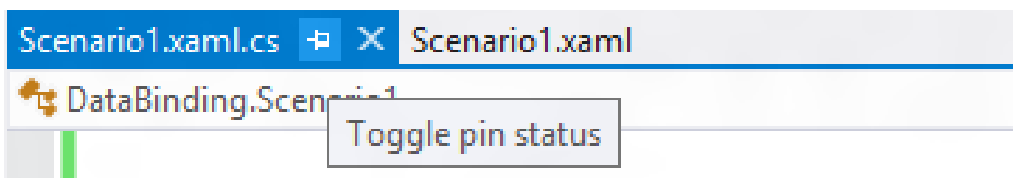


## Navigating and Understanding Solutions

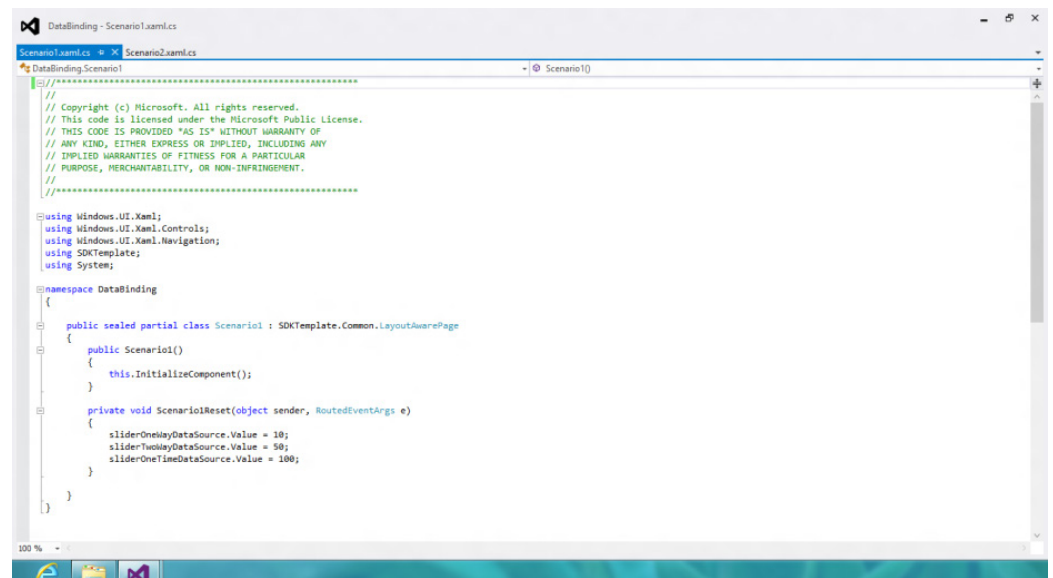
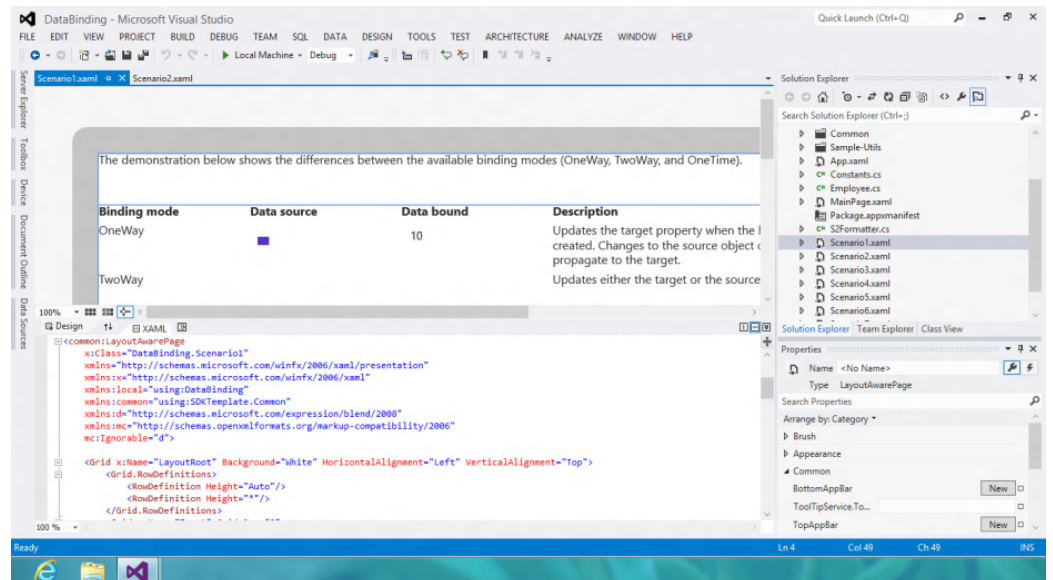
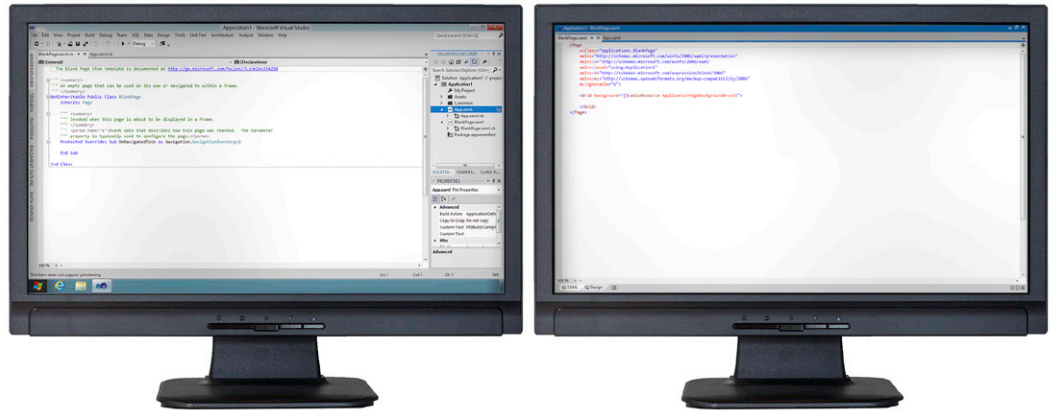
As hardware becomes more powerful and user expectations increase, applications become more complex and an increasing number of resources are used in projects and solutions. It is axiomatic that complexity increases as more time is invested in an application and more features are added. As a result, most code bases tend to expand. The Visual Studio IDE includes powerful features to help developers to navigate and understand ever more complex solutions.

### Document Tab Well

A typical Visual Studio project consists of many files, and developers often will have a number of files open at the same time; it can become difficult to keep track of them all. Visual Studio provides more control over how it organizes open files displaying XAML or HTML markup, code, resources, properties, configuration data, and graphics by enabling you to pin documents to the "tab well." The tabs for all unpinned documents appear to the right side.



The tab well also works with multiple monitors. If you drag a document from Visual Studio to a second monitor, it includes its own tab well. You can drag other documents into the same window, and their tabs will appear in the tab well. Using this approach, you can keep your primary documents and code open on your main monitor while having any additional documents displayed and accessible on a second monitor.



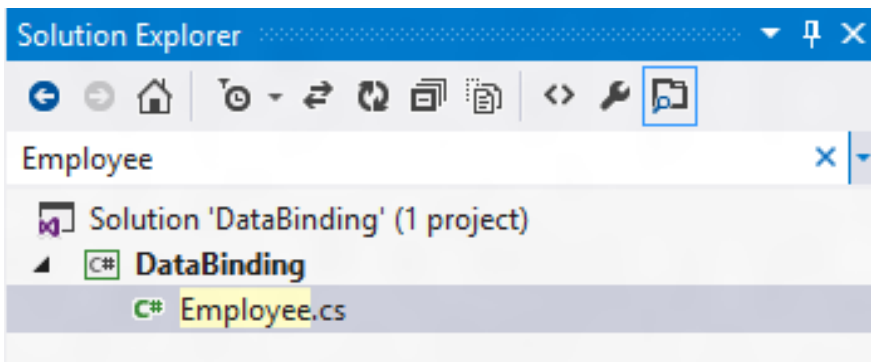


## Solution Explorer

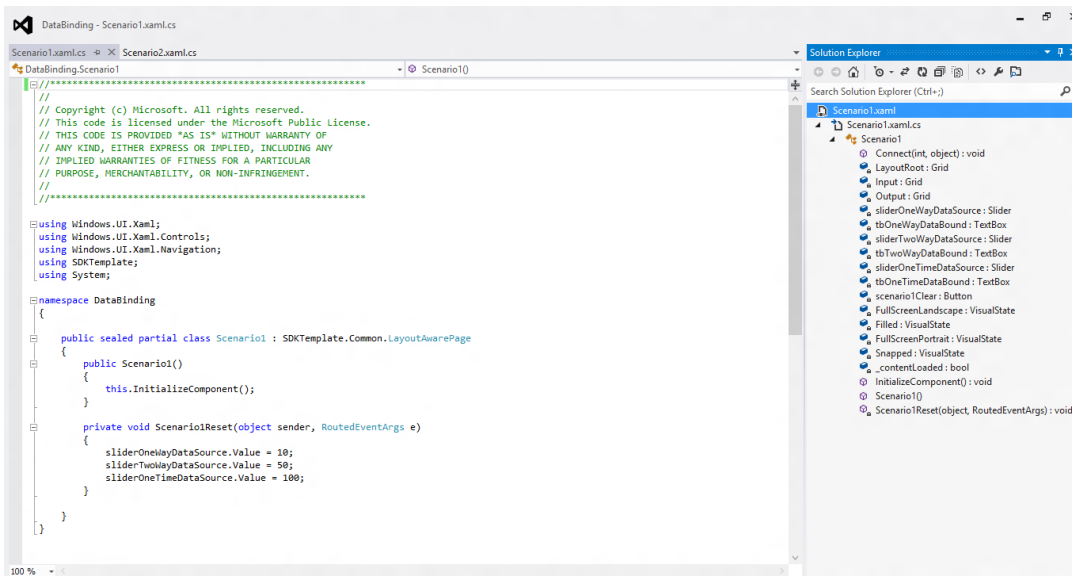
Solution Explorer has been extended to more easily understand the structure of complex projects, enabling you to visualize the elements of your applications, and helping you to locate items much more easily.

For example, expanding an XAML file in Solution Explorer also opens the code-behind file. Visual Studio displays the methods, properties, fields, events, and delegates in an object browser. Required information is in one easy-to-find place, and can be further explored to reveal type information for objects and methods.

The search bar in the Solution Explorer window enables you to locate items in your solution that match a specified string, and you can also limit the items displayed in Solution Explorer to the files that are currently open, unsaved, or edited.

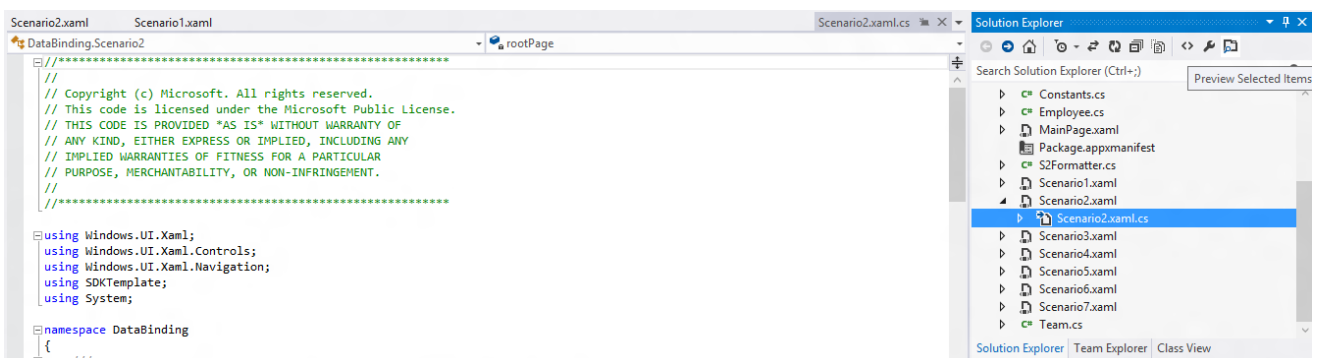


The [Create new window containing a copy of the content of this window](#) button in the Solution Explorer toolbar opens a new Solution Explorer window with the focus on the currently selected item. If you have a second monitor and your operating system supports it, you can choose which monitor displays a window. You can even group multiple windows together in “rafts” on other monitors. These settings persist when you close Visual Studio.



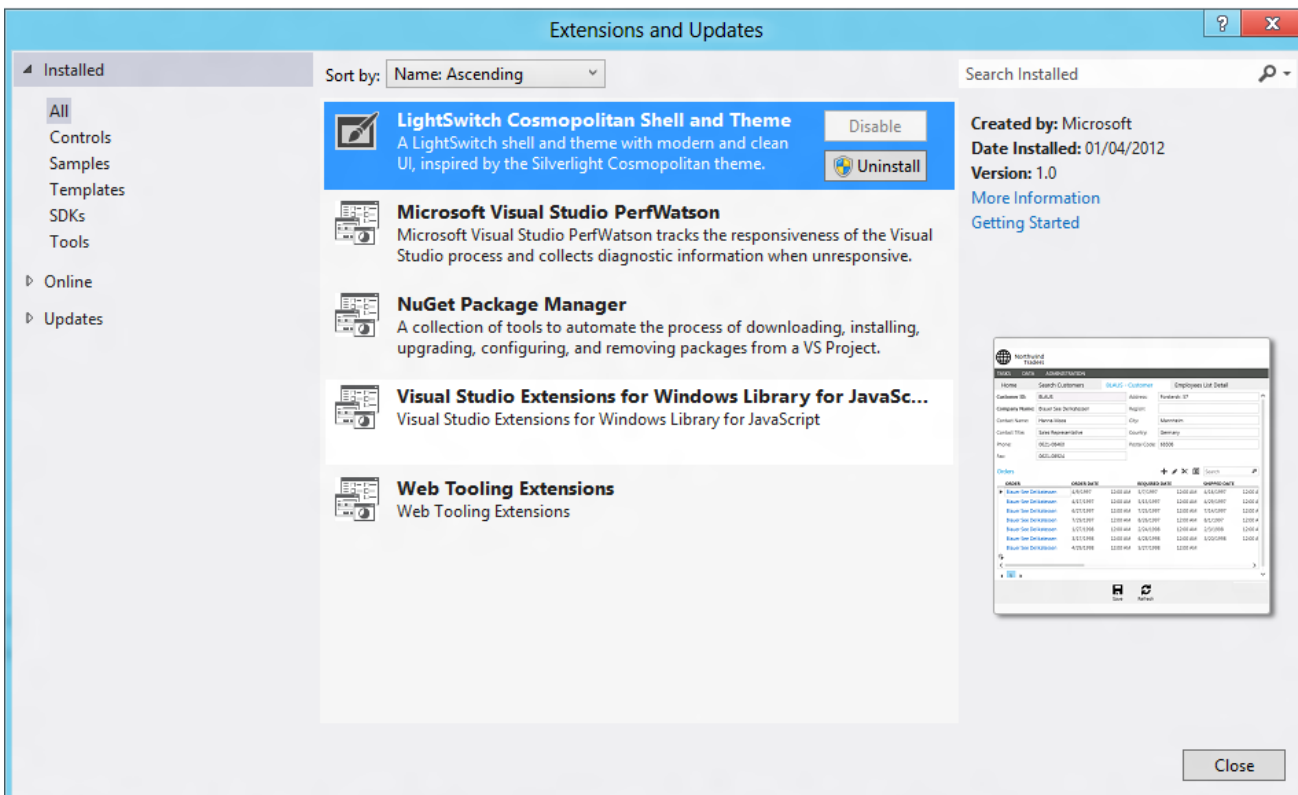
## Preview Tabs

Solution Explorer makes finding the right file much easier by displaying the file contents in a preview tab prior to opening it. As you move between files in Solution Explorer, the preview tab recycles. When you find the file you are looking for, you simply start editing the file and Visual Studio automatically converts your preview tab to a normal tab. This significantly reduces clutter in the Document Tab Well and can help to make you more productive.



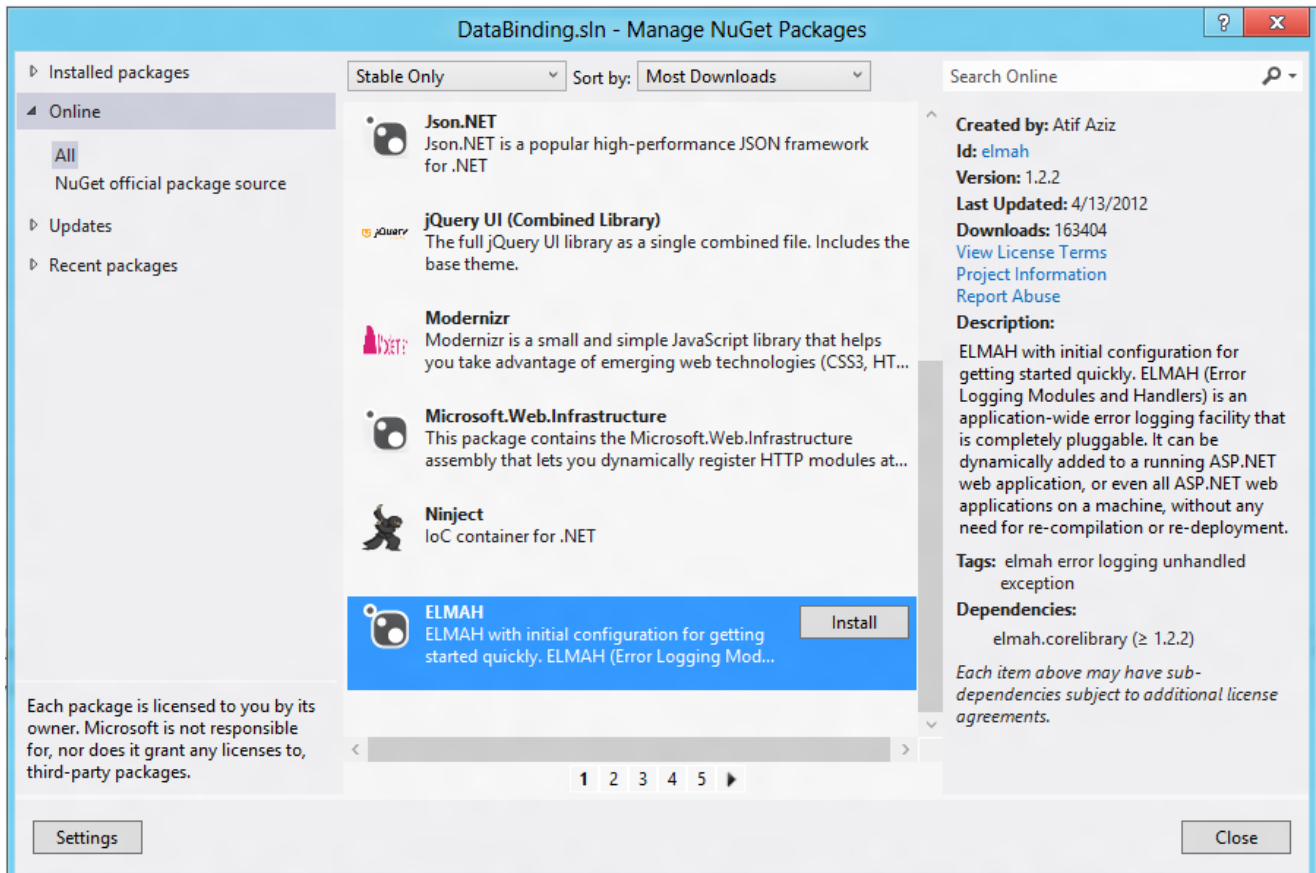
# Getting Started with Community Code and Extensions

Visual Studio 2012 makes it easy to access community code. You use the Extension Manager to install, manage, or access the online gallery to which Microsoft and third-party developers can upload simple-to-install add-on extensions. In Visual Studio 2012, if an extension developer updates an extension that you are using, the IDE notifies you of the update to ensure that you always know you are using the most recent version.



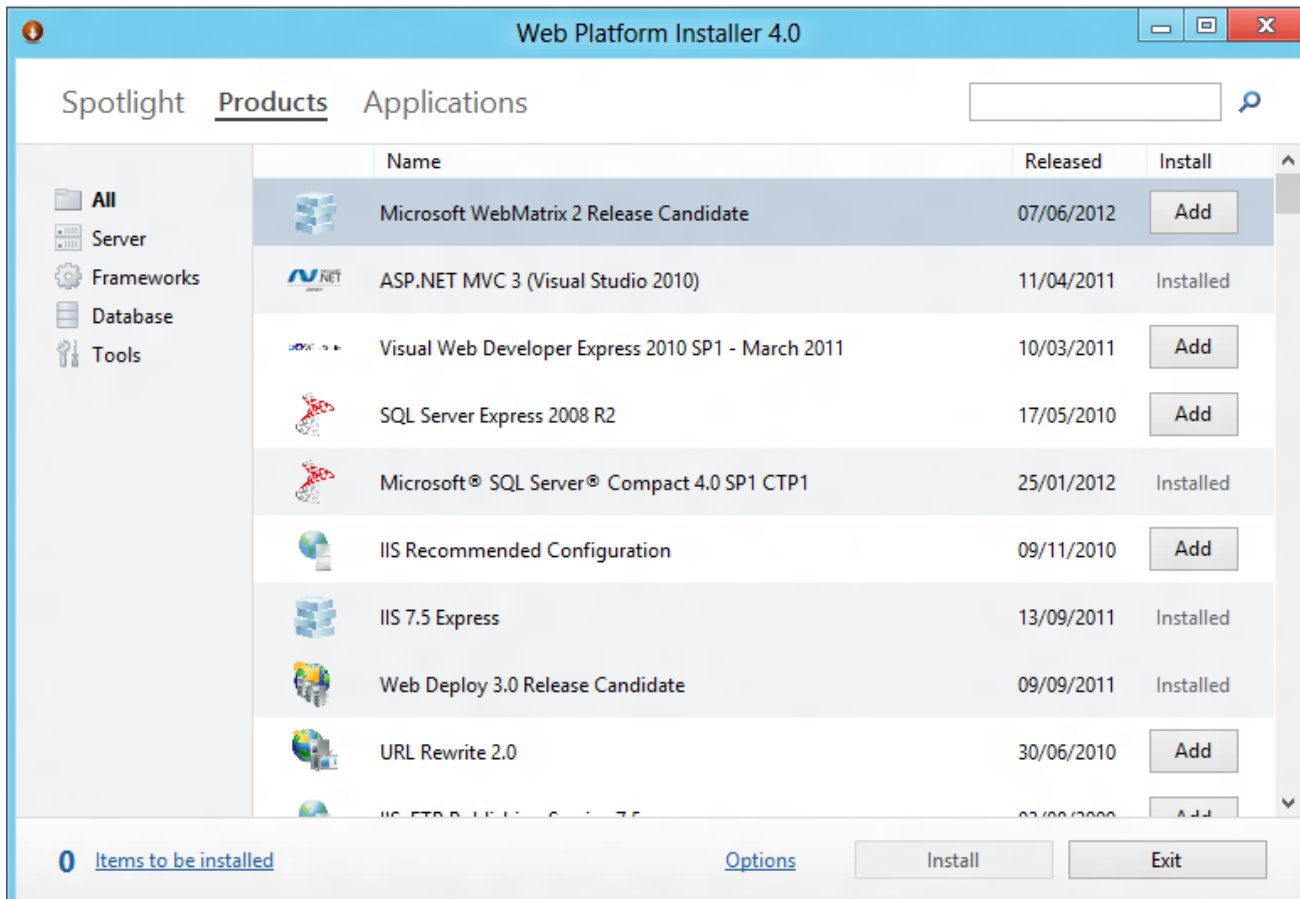
One useful new extension included with Visual Studio is the NuGet Package Manager, which you can use to add packages to your projects. For example, the Error Logging Modules and Handlers (ELMAH) logging library is often used by

web developers. Installing the ELMAH logging library includes adding assemblies and requires specific changes to configuration files. The NuGet package installer adds the necessary files and makes the appropriate changes to configuration files.



The Web Platform Installer (Web PI) is a stand-alone tool that makes it easy to get the latest components of the Microsoft Web Platform, including IIS, SQL Server Express, .NET Framework, and Visual Web Developer.

In addition to web platform components, the Web PI also enables you to install several common, free web applications.



## Performance Improvements

Visual Studio includes significant improvements to make it faster and more responsive than previous versions. These include:

- Using less virtual memory, which reduces the footprint and resources required.
- Loading projects asynchronously and loading key parts of the project first, so that developers can start working faster.
- Making better use of multi-core processors when compiling applications.



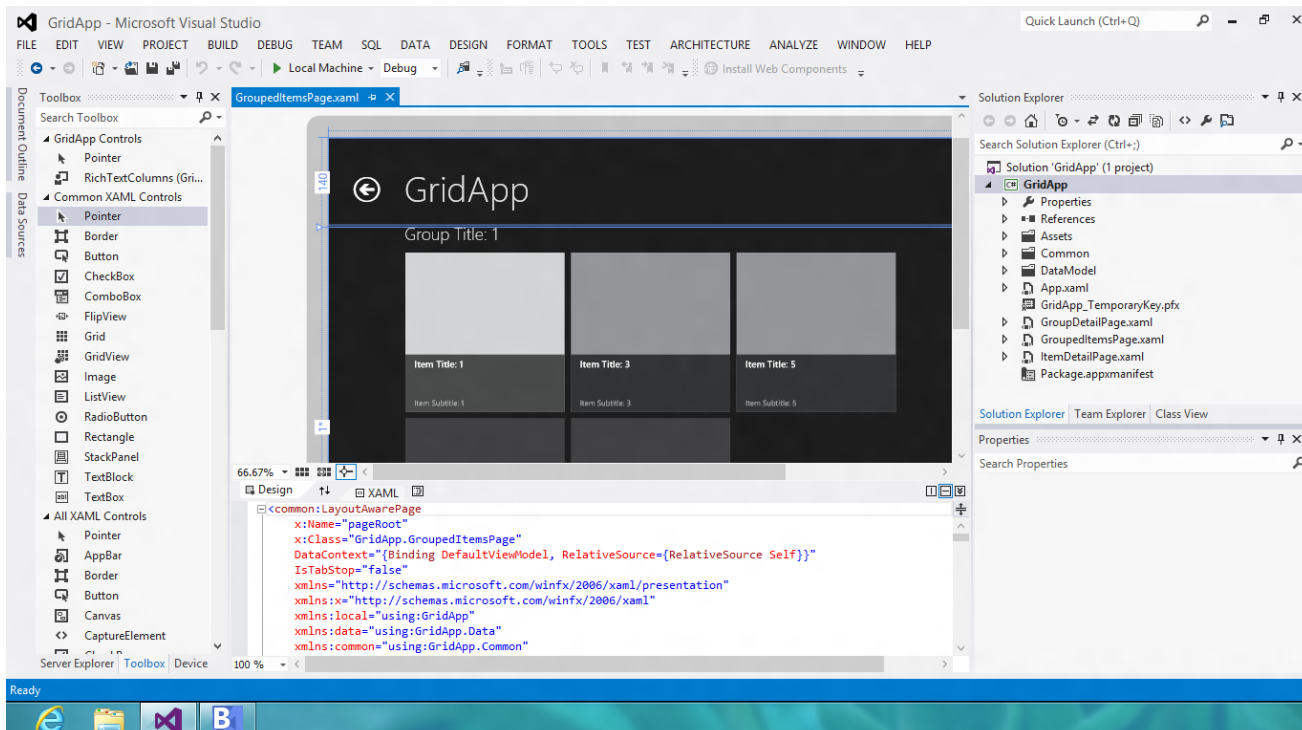
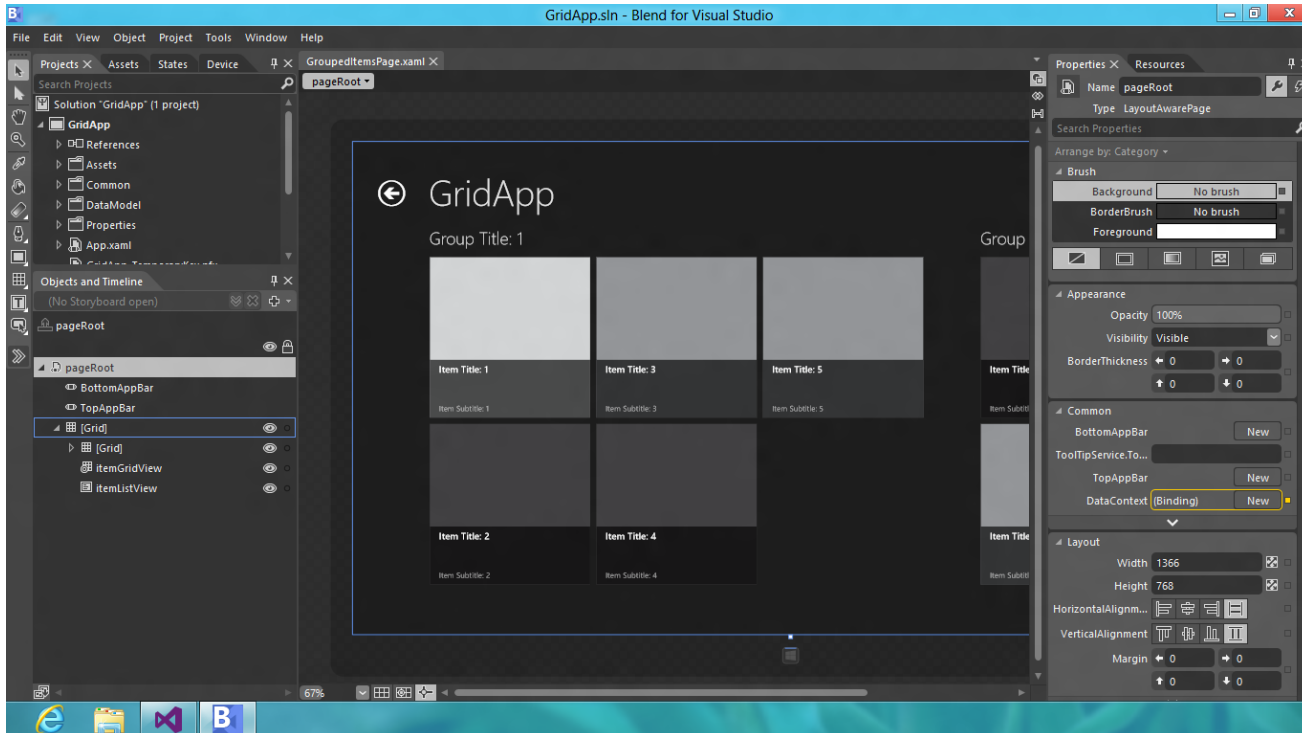
## Language Support

Visual Studio now supports Language Interface packs, making it much easier to change the language of the user interface without installing additional versions of Visual Studio. Personalization settings are preserved when you switch languages.

## Designer Enhancements

Visual Studio includes a new design surface for creating great looking applications and user interfaces. Project formats are now consistent between Blend for Visual Studio 2012 and Visual Studio 2012, and both IDEs use the same designer surface. This ensures that the application renders identically in both environments. Blend provides powerful design tools that help developers to optimize the look and feel of their applications. Blend offers a number of improvements, which include:

- DOM manipulation improvements. Designers can now identify elements that have been moved or reparented by JavaScript.
- Artboard improvements. The Artboard now supports Grid lines and Snap to Grid lines, the ability to turn adorners on or off, the ability to select all elements, and margin and padding value indicators.
- Improvements to CSS support. IntelliSense now indicates syntactically invalid CSS selectors, and users are notified when properties are unrecognized.



## Team Explorer

Team Explorer enables developers and other team members to browse project source code, work items, builds, and other data stored in Team Foundation Server. With Team Explorer, team members can connect to Team Foundation Server, Team Foundation Server Express, or Team Foundation Service (the SaaS version of Team Foundation Server). All members of the Visual Studio 2012 family, including the Visual Studio 2012 Express versions, include Team Explorer.

# Platform Experience

## Asynchronous Programming

Users expect applications to remain responsive. Performing long-running tasks asynchronously can help to achieve this. Visual Studio 2012 together with Windows 8 includes key enhancements to help developers create applications that incorporate asynchronous code.

The C# and Visual Basic languages have been extended with the `async` method modifier and the `await` operator, enabling a programmer to easily specify the code that initiates a long-running task and to indicate which code to run when the task completes. This dramatically simplifies the code required to create an asynchronous method. Developers can use the Task Parallel Library within a method labeled with the `async` modifier to perform an action asynchronously.

<CODE>

```
Task.Run(() =>
{
    TimeConsumingMethod();
});
```

</CODE>

At compile time, the compiler decorates asynchronous methods with an attribute that can help improve toolability support, such that via reflection you can find the compiler-generated state machine that underpins a particular asynchronous method.

Alternatively, developers can use many of the new methods added to the Windows Runtime (and made available to managed applications) to perform common system operations asynchronously. The Windows Runtime exposes asynchronous APIs for all tasks that may take some time to perform, such as reading from or writing to a file.

<CODE>

```
StorageFolder folder = ApplicationData.Current.LocalFolder;  
folder.CreateFileAsync("MyFile.txt");
```

</CODE>

In addition to providing this asynchronous functionality, developers can also use the new keywords to provide asynchronous support to Windows Communication Foundation (WCF) services. When a developer needs to call an asynchronous operation in a service, Visual Studio generates a proxy class that correctly invokes the operation with little additional effort from the developer.

## Portable Libraries

Visual Studio 2012 provides tooling to develop for several different platforms, including the desktop, web, Windows Azure cloud, Windows Phone, and Xbox 360. For developers who needed to develop across these platforms by using previous versions of Visual Studio, code had to be recompiled for each target platform, thus creating many versions of the same application logic, each of which needed to be maintained separately. This duplicated the effort required when fixing bugs or changing the business logic of an application.

Visual Studio 2012 includes support for portable libraries that can help to resolve this problem. A developer can create a library based on the **Portable Class Library** project template. Using the template, the developer creates a

project and chooses the target platforms. The code is automatically restricted to the functions that are common across these platforms. For example, string concatenation is the same across platforms, so it is permitted in a portable library; whereas file system access can be different, so it is not permitted.

Portable libraries are binary portable assemblies, meaning that you can copy the same assembly to multiple platforms without recompiling. In the event of an update, you simply recompile the portable library once and deploy it into all of your applications. In this way, developers can build and reuse a portable core of their code and then for each platform build only what is necessary to conform to platform-specific conventions. This enables you to consolidate key investments and fixes in a single piece of code.

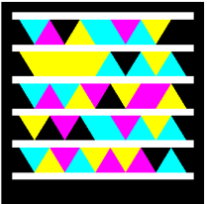
## What's new in the .NET Framework 4.5

The .NET Framework 4.5 includes many new features to make it easier for you to develop more powerful applications. The following list includes just a few of these new features and modifications:

- Integration with the Windows Runtime, enabling you to build managed applications that can take advantage of the new features of the Windows Runtime (such as the many new asynchronous APIs) in a seamless manner.
- Console UTF 16 support, which enables you to use UTF 16 encoding in console applications.
- Application domain culture setting, which enables you to set the application domain culture from within your application.
- Managed Extensibility Framework (MEF) improvements, which enable support for generic types, defining multiple scopes, and support for convention-based programming.
- Customizable regular expression timeouts, which enable you to configure timeouts for resolving regular expressions, helping to ensure that your application remain responsive.



- Improved compression, which enables you to reduce file size by more than in previous versions.



The .NET Framework 4.5 is compatible with Windows Vista, Windows 7 and Windows 8. For more information on the new features in the .NET Framework 4.5 see the What's New in the .NET Framework 4.5 page at [http://msdn.microsoft.com/en-us/library/ms171868\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ms171868(v=vs.110).aspx).

### What's new in Visual C#

Visual C# has been extended and enhanced to take advantage of the new features provided by the .NET Framework 4.5. Examples include

- Built-in support for writing methods that can perform tasks asynchronously through the `async` method modifier and the `await` operator.

<CODE>

```
async Task<int> GetSizeAsync(string url)
{
    var client = new HttpClient();
    string data = await client.GetStringAsync(url);
    return data.Length;
}
```

</CODE>

- Caller information attributes, which identify the source code path, source code line, and member name of the caller during development and debugging, or to create a diagnostic application.

<CODE>

```
public string GetCallerDetails([CallerFilePath] string file = "",
                              [CallerLineNumber] int line = 0,
                              [CallerMemberName] string member = "")
{
    return string.Format("File path: {0}, Line: {1}, Member name:
{2}", file, line, member);
}
```

</CODE>

- Parallel compilation, which helps to shorten project build times.

## What's new in Visual Basic

Visual Basic has been similarly enhanced and its feature-set brought more in line with that of Visual C#.

- Support for writing asynchronous methods through the **Async** modifier and **Await** operator.

<CODE>

```
Async Function GetSizeAsync(url As String) As Task(Of Integer)
    Dim client = New HttpClient()
    Dim data As String = Await client.GetStringAsync(url)
    Return data.Length
End Function
```

</CODE>

- Iterators, which enable developers to return (yield) multiple items of the same type from a method.

<CODE>

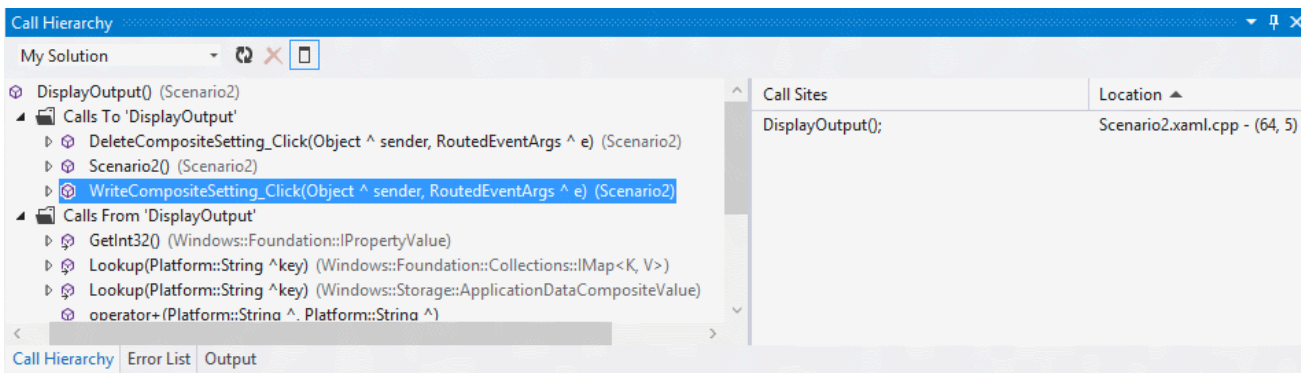
```

Iterator Function GetCustomers() As IEnumerable(Of String)
    Yield "Contoso"
    Yield "Fabrikam"
End Function

```

</CODE>

- Call hierarchy, which enables you to locate code that calls a particular member, types that implement a particular interface, or members that override a virtual or abstract member.



- Caller information attributes, to identify the source code path, source code line, and member name of the caller during development and debugging.

<CODE>

```

Function GetCallerDetails(<CallerFilePath> Optional file As String = "",
                        <CallerLineNumber> Optional line As Integer = 0,
                        <CallerMemberName> Optional member As
                            String = "")As String
    Return String.Format("File path: {0}, Line: {1},
    Member name: {2}", file, line, member)
End Function

```

</CODE>

- The **Global** keyword, which you can use to create types in a namespace outside of the project defined root namespace.

<CODE>

```
Namespace Global.MyGlobalNamespace
    Class SimpleClass

End Class
End Namespace
```

</CODE>

## What's new in Visual F#

Visual F# continues to evolve to become a fully featured language. The new enhancements include:

- Type providers, which you can use to generate types that represent structured data. Visual Studio includes type providers for the following data types:
  - o Open data (OData), a network service protocol that enables data to be transmitted over a network or the Internet.
  - o Database connections, for example, connections to a SQL Server database.
  - o Database schema (DBML) files that are generated by using LINQ to SQL.
  - o Entity data model (EDMX schema) files that are generated by using the entity framework.
  - o Web services that expose WSDL-enabling client applications to more easily consume these services.

- o Resource (resx) files,  
used in .NET Framework applications.

<CODE>

```
// Use a type provider to access a SQL Server database.  
[<Generate>]  
type dbSchema = SqlDataConnection<"Data Source=ServerName\  
InstanceName;Initial Catalog=CustomerDatabase;Integrated Security=SSPI;">  
let db = dbSchema.GetDataContext()
```

</CODE>

- Query expressions (LINQ), which you can use to write LINQ queries in code in syntax similar to SQL. This can save you time when writing code and make code more understandable for a reviewer, thus helping to ensure a quality product.

<CODE>

```
// Use LINQ to select customer from the customers table in a database  
let selectCustomers = query { for customer in db.Customers do  
    select customer }
```

</CODE>

- Auto-implemented properties, which enable you to add properties without manually implementing the backing store for that property.

&lt;CODE&gt;

```
// Class with automatic property.
type SampleClassType() =
    // Declaring an automatic property.
    member val autoProperty = 3 with get, set

let class1 = new SampleClassType()
// Accessing an automatic property.
System.Console.WriteLine("{0}", class1.autoProperty)
// Setting an automatic property.
class1.autoProperty <- 10
```

&lt;/CODE&gt;

## What's new in ASP .NET

The features available to ASP.NET have been extended to incorporate the new application models and development technologies for building web applications.

- Client-side validation. Visual Studio also adds support for JavaScript form validation to ASP.NET web forms. This feature previously existed for Model View Controller (MVC) developers.

&lt;CODE&gt;

```
public class SimpleModel
{
    public SimpleModel() { }

    [Required]
    [StringLength(40)]
    public string Name { get; set; }
}
```

&lt;/CODE&gt;



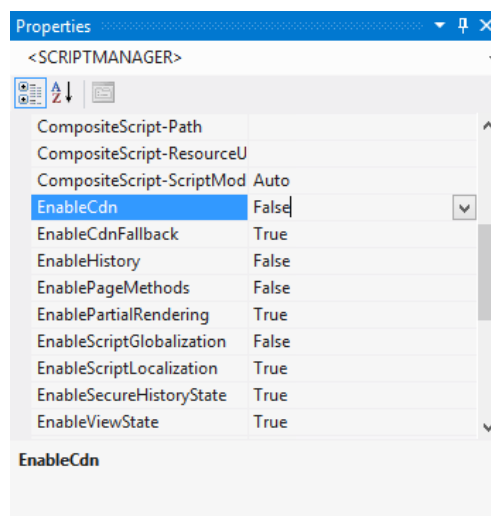
- Model binders. Visual Studio adds support for binding controls to properties of a model in ASP.NET web forms applications. This is similar to the functionality previously available to MVC developers.

<CODE>

```
<asp:GridView ID="customerDeatils" runat="server"
ModelType="SimpleModel" SelectMethod="GetModel"
AutoGenerateColumns="false">
    <Columns>
        <asp:BoundField DataField="Name" HeaderText="Customer Name" />
    </Columns>
</asp:GridView>
```

</CODE>

- CDN fallback support. Visual Studio adds support for loading resources from a content delivery network (CDN) in order to improve page load time. CDNs typically have very high bandwidth, and users may have copies of files served by an application cached locally. Developers can specify files hosted on CDNs. However, developers often have no control over a CDN, and files may change. Developers can use the **EnableCdn** property of the **ScriptManager** control to verify that resources loaded from a CDN function correctly, but to load a local copy of the resource if it fails.



- Full HTML5 support, including semantic elements such as `email`, `tel`, and `url`.
- Page Inspector. Page Inspector enables you to open a web page within Visual Studio. By using the Page Inspector, developers can make live changes to the DOM and edit CSS rules. When they achieve the desired result, developers can easily copy these changes to the correct source file.
- OAuth and OpenID support. Using these features, you can create sites and permit users to sign in with credentials from other sites, including Google, Yahoo, Facebook, Twitter, and Microsoft account.
- IIS Express. IIS Express provides a hosting environment based on IIS that developers can use to test and debug web applications. Visual Studio 2012 includes IIS Express as the default web server for debugging web applications. Using IIS Express ensures that developers can use features such as SSL while debugging.
- Local DB, a lightweight database engine that runs in the context of the current user, reducing configuration effort in the development environment. LocalDB provides a feature-set that is fully compatible with SQL Server and SQL Azure, making it simple to migrate from the development environment to production.
- Support for unhandled exceptions in JavaScript.
- Support for debugging T4 text templates.

## Services Experience

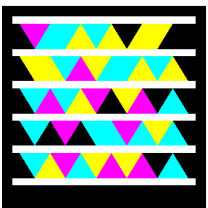
### MSDN Services and subscription benefits

As the world of software development becomes more complex, developers need a wide range of skills and information to ensure that the solutions they design, develop, and deploy will operate in many environments.

MSDN subscriptions give your development team access to virtually every Microsoft product and technology, helping to ensure that they make the right technology choices when producing custom solutions. In addition, MSDN provides high-quality resources, training, and support that will help enhance the skills and productivity of development teams.

MSDN Subscriptions include:

- Development and test use of Microsoft tools, servers, and clients.
- Exclusive access to Visual Studio feature packs.
- Access to new versions of software (including the next pre-release version) as well as previous versions for development and testing purposes, or for migrating legacy applications.
- Access to the Windows Azure Platform and introductory offers.
- Professional technical support
- Online training resources



A range of MSDN subscriptions are available, with different prices and benefits. For a full list, see the Buy an MSDN Subscription page at <http://msdn.microsoft.com/en-us/subscriptions/hh442902.aspx>.

Subscription Level	MSDN Operating Systems	MSDN Embedded	Visual Studio Professional with MSDN	Visual Studio Test Professional with MSDN	Visual Studio Premium with MSDN	Visual Studio Ultimate with MSDN
<b>Software and Services for Production Use</b>						
<b>Primary Development/Testing Tool</b>		<a href="#">Visual Studio Professional</a>	<a href="#">Visual Studio Professional</a>	<a href="#">Visual Studio Test Professional</a>	<a href="#">Visual Studio Premium</a>	<a href="#">Visual Studio Ultimate</a>
Visual Studio Team Foundation Server 2010			✓	✓	✓	✓
Visual Studio Team Explorer Everywhere 2010						✓
<a href="#">Microsoft Expression® Studio 3</a>					✓	✓
Microsoft Office Professional Plus 2010, Project Professional 2010, Visio Premium 2010					✓	✓
<a href="#">Windows® Azure™ platform</a> <sup>1</sup>			✓		✓	✓
<b>Software for Development and Testing</b>						
<a href="#">Windows®, Windows Server®</a>	✓	✓	✓	✓	✓	✓
<a href="#">Microsoft SQL Server®</a>			✓	✓	✓	✓
Toolkits, Software Development Kits, Driver development kits	✓	✓	✓	✓	✓	✓
<a href="#">Microsoft Office</a>					✓	✓
<a href="#">Microsoft Dynamics®</a>					✓	✓
All other Servers (Microsoft Office SharePoint Server, Microsoft Exchange, etc.)					✓	✓
<a href="#">Windows Embedded</a>		✓			✓	✓
<b>Additional Benefits</b> <sup>2</sup>						
<a href="#">Technical support incidents</a>	2	2	2	2	4	4
<a href="#">Priority support in MSDN Forums</a>	✓	✓	✓	✓	✓	✓
<a href="#">Microsoft E-Learning</a> course collections (typically 20 hours each; per calendar year)	1	1	1	1	2	2
<a href="#">MSDN Magazine</a>	✓	✓	✓	✓	✓	✓
<a href="#">MSDN Flash newsletter</a>	✓	✓	✓	✓	✓	✓
<a href="#">MSDN Online Concierge</a>	✓	✓	✓	✓	✓	✓
<a href="#">Special Offers from Microsoft and partners</a>	✓	✓	✓	✓	✓	✓

## Team Foundation Service

Team Foundation Service is a SaaS offering from Microsoft that provides the rich functionality of a local installation of Team Foundation Server without the additional effort involved in managing, licensing, and backing up a local server. For more information about Team Foundation Service, see the chapter Application Lifecycle Management with Visual Studio 2012.

Team Foundation Service / New Web site

HOME WORK SOURCE BUILD

backlog board work items

Product Backlog

- Past
- Current
  - Sprint 4
- Future
  - Sprint 5
  - Sprint 6

Sprint 4

23 February - 29 February  
5 work days

work details on

contents capacity

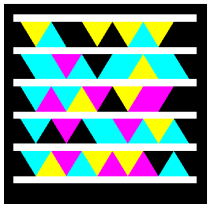
Create Backlog Query Column Options

Effort	Title	State	Assigned To	Remarks
30	Add log in page. Create login page interface	To Do	mikes@cont...	
30	Add purchase history page Develop purchase history page	Done	paulb@cont...	
	Test bug	New	mikes@con...	

Work

- Work By: Activity
- Work By: Assigned To

© Microsoft Corporation. All rights reserved. Terms of Service Privacy Statement Service Notifications Trademarks Microsoft



You can find more information about Team Foundation Service at <http://tfs.visualstudio.com>.



## Summary

Visual Studio 2012 provides a powerful new developer experience that enables developers to be more productive than ever before. The IDE ensures that developers are highly efficient by providing easy access to commonly used tools and a streamlined user interface that enables developers to remain focused on their code. Visual Studio enables developers to build applications for Windows 8, and the new asynchronous programming features ensure that developers can easily produce responsive and scalable code, while portable libraries help developers to implement applications and libraries once for use across multiple platforms and devices. MSDN subscriptions provide developers with access to a range of Microsoft resources and software for testing and development purposes, which makes creating development and test environments easier and less expensive.

In short, Visual Studio 2012 helps developers code faster, write more efficient code, and deploy to more platforms and devices with minimal extra effort.





# Developing for Windows 8

Great applications begin with great developers. Windows 8 gives you the platform and tools to create rich application experiences that enable your customers to focus on the tasks that are important to them. Applications are at the center of the Windows 8 experience. They are alive with activity and vibrant content. Your customers immerse themselves in your full-screen, Windows Store applications, enabling them to focus on their content rather than on the operating system.

Signing up to sell your application through Windows Store means you can tap into the broad customer reach of Windows, which spans markets around the world. You will draw from a wealth of new tools and services that enable you to develop

and deploy applications faster and more profitably. Of course, you can also reach and engage new customers by using the commerce platform of your choice.

With Windows 8 you can leverage your existing skills and code assets to create great experiences for your customers:

- Web developers can use their HTML5, CSS3, and JavaScript skills to build native applications for Windows.
- .NET developers can use XAML, C#, and Visual Basic to build beautiful Windows Store applications.
- Game developers can use the power of DirectX 11.1 to build amazing, immersive gaming experiences.
- Driver developers can use the new, integrated Visual Studio 2012 development environment to increase productivity.

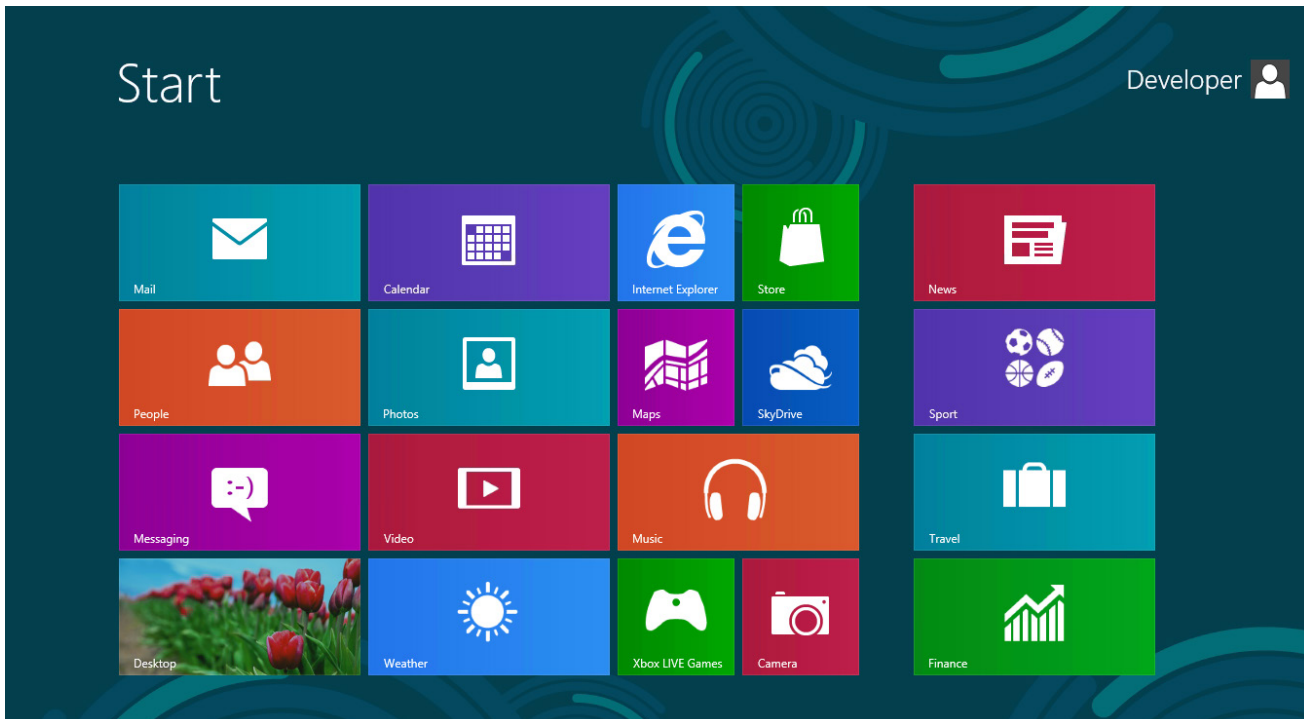
With Windows 8, you are ready to imagine, build, and sell the next great application.

## Windows 8 Applications are Simple to Use

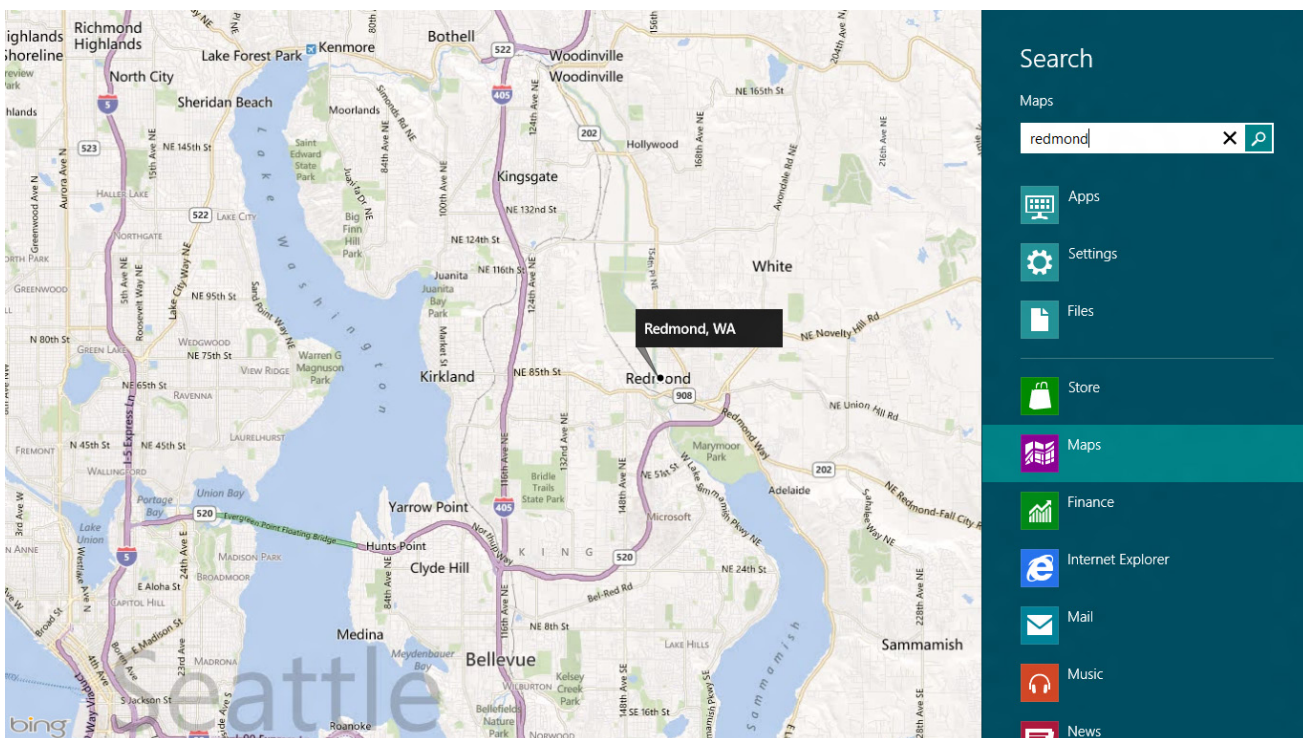
The user interacts with the operating system shell by utilizing a consistent and easy-to-learn set of gestures. For example, a user can tap an application tile to start it running, and stretch and pinch elements in the user interface (UI) to zoom in and out. The user can also slide the UI of a running application off to one edge of the screen to reveal the shell, and then start another application.

Applications can present live tiles that dynamically surface their most important or recent content, providing users with essential information at a glance and without the need to start the application or bring it to the foreground. Applications running in the background can also send notifications to the

Windows 8 shell, which can then inform the user of significant events. For example, an e-mail application can indicate that a new message has arrived, and the user can then re-engage with the application to read the message.



Windows 8 applications can also make use of **application contracts**, which enable applications to communicate with each other and to integrate with common operating system features, freeing the user of your application from the need to learn specific procedures for performing common tasks. For example, the Search contract enables a developer to supply the search results from an application to the Windows Search experience. The Share contract enables one application to share the content it produces with another; the content could be photographs, images, text, or a whole range of other types of data.



Other functionality available in this manner includes the PlayTo contract, which provides the ability to play content on the device if the user clicks the Windows 8 PlayTo button; the Print contract, which enables an application to output data to a connected printer if the user clicks the Windows 8 Print button; and the Send contract, which enables an application to send content to a connected device.

## Windows 8 Applications Can Easily Connect and Communicate

Applications can take advantage of Windows 8 connectivity to efficiently communicate with the outside world. The operating system automatically switches to the network connection with the best quality, and applications can throttle the speed at which they send and receive data in order to deliver the best possible user experience for an application, while simultaneously balancing the requirements of other running applications. You can also build applications that transmit and receive SMS text messages by using a broadband adapter or a connected mobile phone. Windows 8 can also update the live tile of an application based on network state or data received.



## Windows 8 Applications Can Adapt to Their Environment

Windows 8 applications can take advantage of the sensors built into many Windows 8 devices in order to adapt themselves to their environment. The operating system makes available data from accelerometers, gyros, compasses, GPS devices, ambient light sensors, and human presence sensors, among others. Devices with near-field proximity sensors enable applications to publish and subscribe to near-field communications events, enabling you to build solutions that can easily connect and communicate across devices, perhaps performing tasks such as automatically copying pictures and videos from mobile phones and cameras, or sharing audio with friends.



## Windows 8 Applications are Device Friendly

Windows 8 can conserve resources such as memory or battery life by suspending applications running in the background. A developer can write code to trap the notifications that Windows 8 sends to an application and save any required state information. The application can be resumed at a later time, either by the user bringing it back to the foreground or as the result of a notification received by the Windows 8 device. The developer can include logic to quickly reapply the saved state and enable the application to continue running. To the user, this transition from running in the background to being suspended and then being reactivated is totally transparent. This strategy is useful for applications that are expected to be inactive when they do not have the focus.

In other situations, an application may need to continue running if, for example, it is carrying out a long-running operation such as



uploading or downloading a large file, or performing a complex calculation. To handle these scenarios, the developer can include code that enables the program to continue with its operations.

## Windows 8 Applications are Mobile and Scalable

Applications can roam. Windows 8 enables applications to synchronize their state with another device via the cloud. For example, a user can install the same application on their computer and their smartphone, run the application on the computer, and perform some work. The user might then leave the office, but transition to using the application on their smartphone while accessing the same data. At home, the user may then switch to another computer and continue running the application, again with the same state and data. The different form factors and capabilities of devices such as PCs and smartphones are not a barrier to well-designed Windows 8 applications. The controls available to Windows 8 developers enable you to build applications that can operate equally well whether they are running on a touch-based device or a computer that relies on a keyboard and mouse to provide input.





When you use the Windows 8 controls, animations, behaviors, and layouts, you can design applications that retain their functionality and appearance regardless of the aspect ratio, resolution, or orientation of the device screen, or the user input method. The Windows 8 scaling system keeps your applications working perfectly on devices with different display sizes and pixel densities.

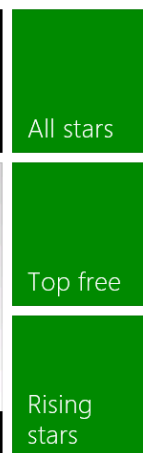
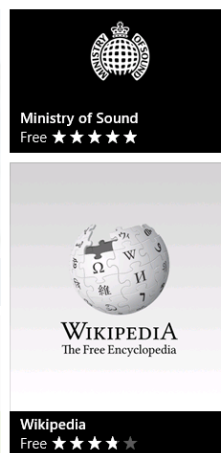
## Windows 8 Applications are Trustworthy and Safe

A Windows 8 application runs in a security context that controls the resources that the application can access. A developer can adjust the limits of this security context—for example, by enabling an application to perform potentially sensitive operations—but the developer must explicitly document these capabilities by providing an appropriate security policy. A user installing the application can query this policy and choose not to install the application if he or she is uncomfortable with its capabilities.

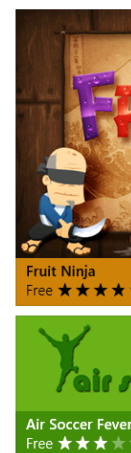
If you are building commercial applications, a key advantage of Windows 8 is the way in which you can distribute software by using the Windows Store.

### Store

#### Spotlight



#### Games



You distribute an application by submitting it to the Windows Store. Prior to your application being made generally available for download and sale, it undergoes rigorous checks and tests to verify that it does not contain malware or exhibit suspicious behavior, and that it meets the security policy provided with the application. You also need to make your applications available for user ratings and reviews. As a result, users can trust applications that appear in the store, making them more likely to download your software than they would be from a traditional e-commerce site.

Additionally, Windows 8 applications implement a safe, clean installation procedure that consists primarily of copying application files to specific folders in the AppData folder. When an application is uninstalled, these folders are deleted, leaving no vestige of the application on the computer. As a developer, you do not have to write the application installer or uninstaller because this functionality is built into the operating system. Consequently, Windows 8 guarantees that installing and uninstalling applications from the Windows Store will not gradually degrade your system or leave behind any unnecessary clutter. This in turn makes users more likely to download your software and install it on a “trial or buy” basis, increasing the likelihood that they will buy your application if they like it.

## Windows Runtime and application model

The Windows Runtime (WinRT) is a straightforward set of APIs used to build Windows Store applications. WinRT lets you build applications that look great and are intuitive for your customers to use. WinRT APIs are available to developers in multiple languages, including JavaScript, C++, C#, and Visual Basic. The new Windows SDK for Windows Store applications also includes a subset of traditional Win32, COM, and .NET Framework APIs, as well as HTML5 and CSS3 APIs that are accessible to Windows Store application developers.

## Choose from a broad range of application programming languages

To build fully native and robust applications that make the most of a computer running Windows 8, you can develop by using web programming technologies or by using familiar Windows programming techniques.

Windows Store applications using JavaScript leverage the combination of HTML5 and CSS3 to define the UI, along with JavaScript for implementing the application logic.

Windows Store applications using C++, C#, or Visual Basic use XAML markup for the UI, with C++, C#, or Visual Basic for application logic. Game developers can build Windows Store games by using C++ and DirectX 11.1 to take full advantage of graphics hardware, or build casual games by using HTML5 or XAML.

## Create immersive games using the power of DirectX

The new Windows 8 graphics stack is better integrated, making Direct2D, Direct3D, and DirectCompute API components easier to use together, and requiring fewer duplicated resources than before. Capabilities previously available only in XNA, such as DirectXMath, XAudio2 API, and XInput API, are now available. For the ultimate experience in gaming and video, use DirectX 11.1 to bring stereoscopic 3D to your applications.

## Compatibility with existing frameworks

The WinRT APIs are directly accessible to other Microsoft programming frameworks, such as the .NET Framework or Windows C Runtime Library. In most cases, accessing WinRT functionality from another framework is as simple as importing the right namespace to your code and taking advantage of simple object-creation semantics that are similar to what you see in JavaScript, Visual C#, and Visual Basic .NET.

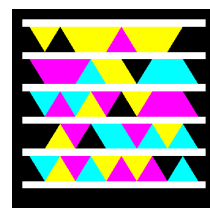
WinRT features are represented by an Application Binary Interface (ABI) that exposes functionality at a binary level rather than an object level, and includes additional details, such as how to call a function, and how exceptions are handled or propagated. Any programming language that can produce the appropriate binary objects can use an ABI, which makes Windows APIs accessible from other programming languages. The ABI is exposed in metadata assemblies (.WinMD files), enabling static languages like C# and dynamic languages like JavaScript to understand the shape and structure of the WinRT API.

The new Windows 8 application models feature native extensibility, which lets you build your own reusable component libraries. You first build your own custom components with C++, C#, or Visual Basic. You can then use JavaScript or any other supported language to call these components from your applications.

## Tools and Resources for Developers

Get started writing applications quickly with the Windows 8 SDK, which includes new versions of the free Visual Studio Express and Blend for Visual Studio 2012. You also get the latest sample applications, templates, and tutorials, as well as guidance from the experts.

The new Windows 8 SDK and the online Windows Developer Center (available at <http://msdn.microsoft.com/en-us/windows/br229518>) provide everything you need to start building your applications, including the latest tools, APIs, compilers, debuggers, sample applications, and documentation.



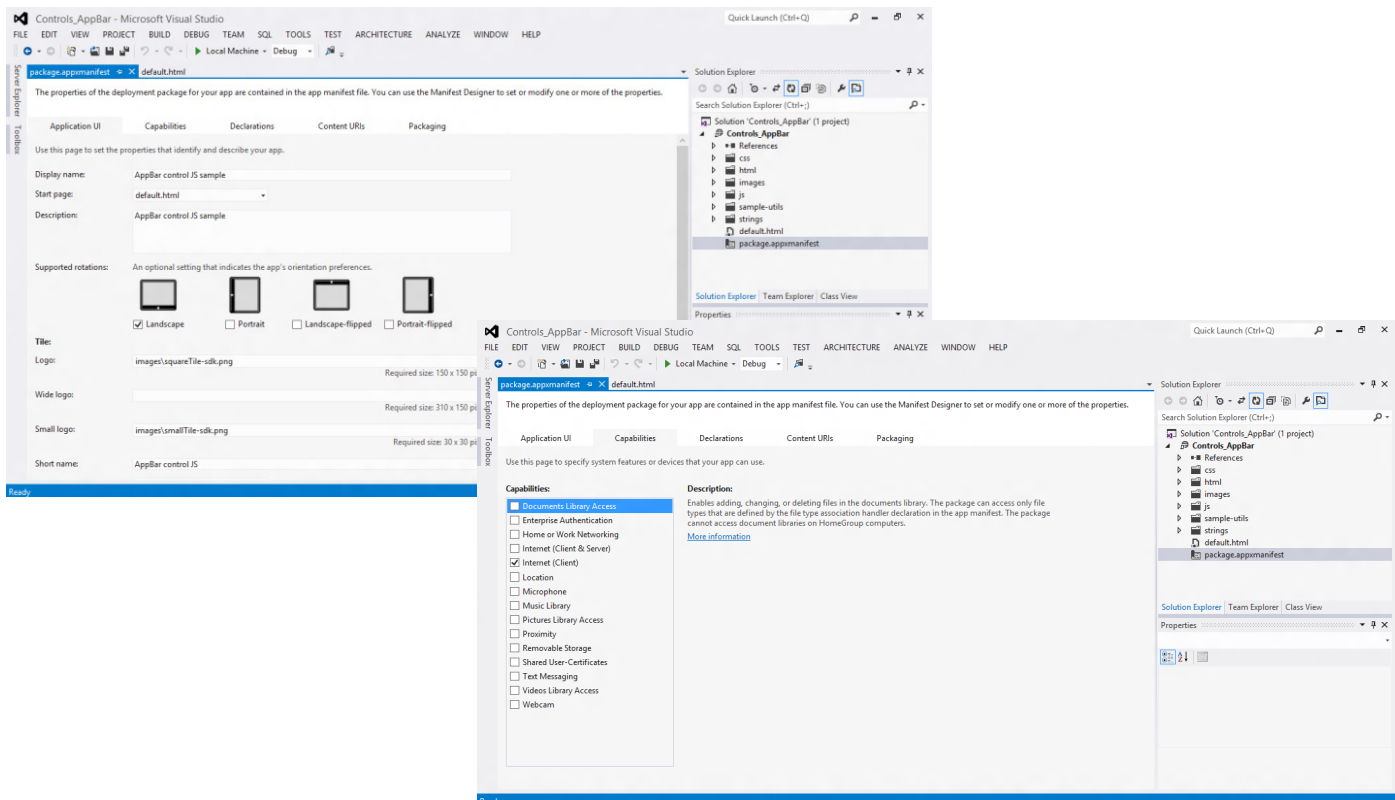
Windows 8 and Windows Store make it simple for millions of customers from practically anywhere in the world to find, try, and buy useful, high-quality applications. You will find it easy to distribute, update, and get paid for the applications that you develop. The site provides a Dashboard that enables you to improve your applications by monitoring their success. You can also view reports on downloads, revenue, usage, in-application transactions, customer ratings, and market trends.

The Windows Developer Center provides you with clear and definitive guidance for developing beautiful Windows Store applications. Other features include:

- Tool downloads, including the Windows 8 SDK (which includes Visual Studio Express and Blend)
- Hundreds of sample applications
- Access to experts in the community
- Tutorials and guidance written by experts

## Configuring, Packaging, and Publishing

With Windows 8, you no longer need to build a custom installer to enable users to install your application. Instead, you provide an application manifest that defines the properties and policies of the application, and then package it up. Visual Studio 2012 provides the Manifest Designer to facilitate this task.



After you have tested and packaged your application, you can submit it to the Windows Store for verification. Visual Studio 2012 makes submitting applications easy and includes tools to help you complete this task.

## **Making Native Code Libraries Available to Windows Store Applications**

Visual Studio 2012 makes it easy to add a class library with C# or Visual Basic to your application, and to create Windows Runtime types that you can call from JavaScript. Internally, Windows Runtime types can use any .NET Framework functionality that is allowed in a Windows Store application. Externally, the members of such a type can expose only WinRT types for their parameters and return values. When you build your solution, Visual Studio builds your .NET Framework project and then executes a build step that creates a WinMD file, which is included as part of your application.

## **Designing better applications: Visual Studio + Blend**

Creating compelling and differentiated applications is not easy, but Visual Studio 2012 and Blend for Visual Studio 2012 provide you with the tools and flexible workflow you need to be successful. Making the user experience (UX) an integral part of the development process is critical to the success of a development project. UX problems found late in the development cycle are a common source of cost overruns.

The combination of Windows 8, Visual Studio 2012, and Blend enable you to create great user experiences, while structuring your application in the tools of your choice. Whether you start developing your application in Visual Studio or Blend is not important—either way, you will be creating standards-based code that will make you productive right from the start.

Blend provides a modern design environment that enables you to drag and drop elements, then move, style, and refine them via the interactive design surface of Blend. The code that Blend generates is concise and tightly written, giving you confidence that the final application will look and feel exactly the same when deployed to a user's device as it does when building it by using Blend.

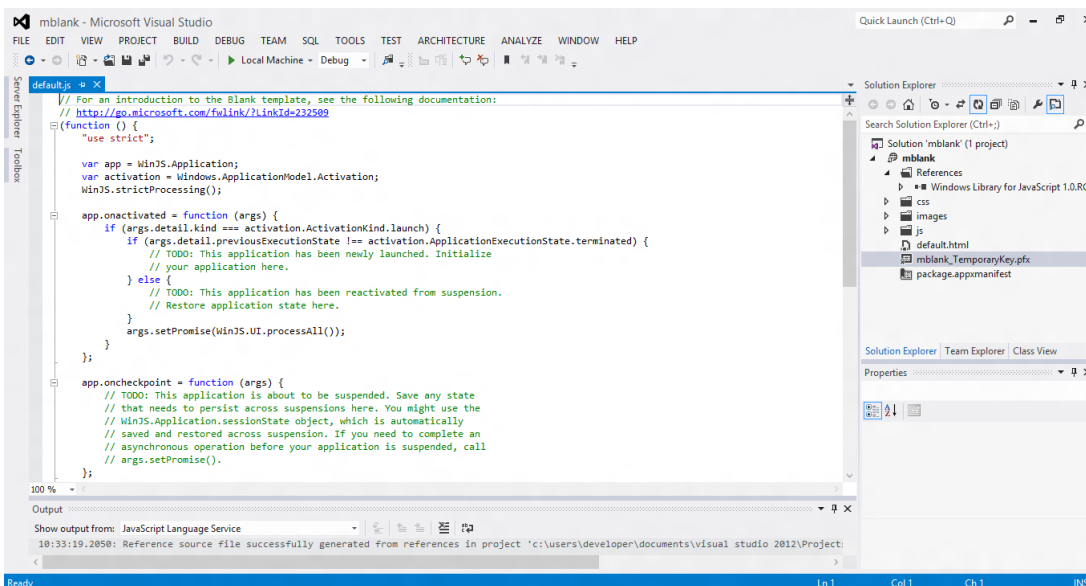
Visual Studio 2012 and Blend for Visual Studio 2012 share exactly the same project and solution formats. In addition, they share the same designer surface, ensuring that designs render identically in both tools. The tight integration between Blend and Visual Studio 2012 enables you to use both tools in parallel, seamlessly switching between them and giving you the flexibility to work the way you want to work at any given point in time. This approach not only eases the task of the developer and the graphics specialist, it also makes it simpler to manage the entire development lifecycle of the application.

## Building Windows Store Applications with JavaScript by Using Visual Studio 2012 and Blend for Visual Studio 2012

Semantic markup is the most productive way to structure Web applications, providing a clean separation of logic and presentation. With Windows 8, you can apply your HTML, CSS, and JavaScript skills to build Windows Store applications. Visual Studio 2012 and Blend for Visual Studio 2012 enable you to make the most of your current skills.

## Creating a new Windows Store JavaScript application

Visual Studio 2012 includes a set of templates to get you started building Windows Store applications quickly by using JavaScript.



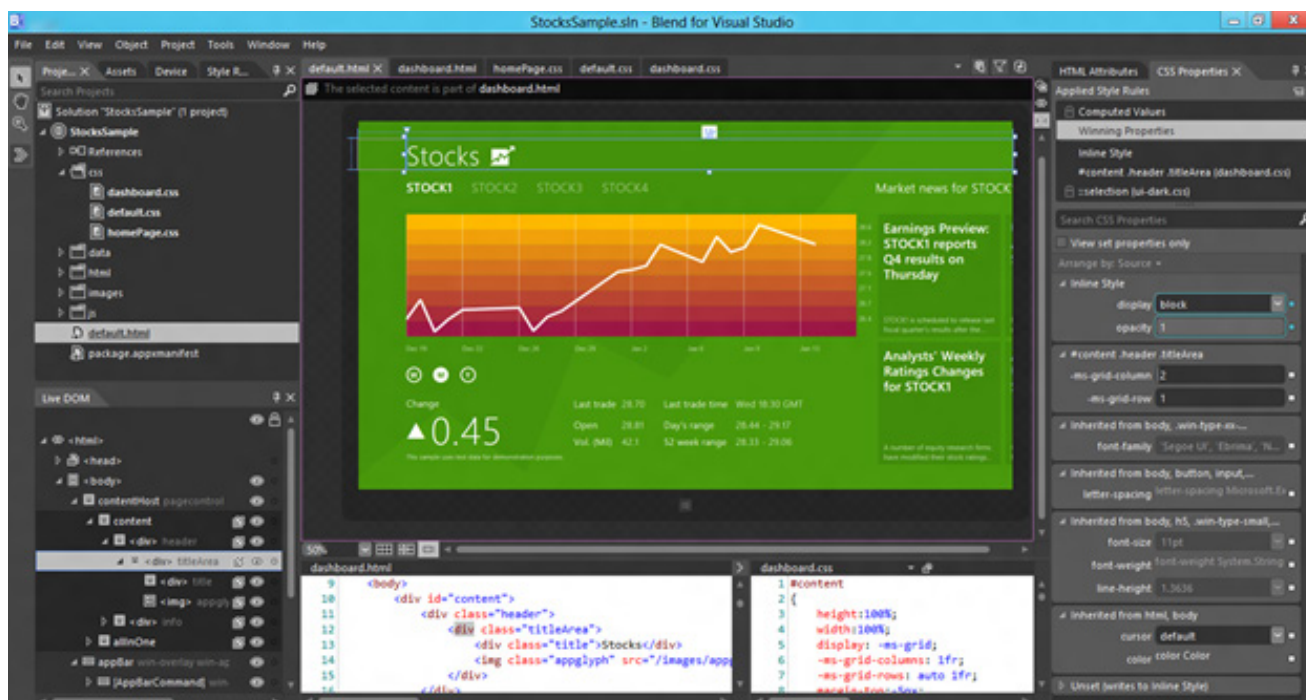


The Blank Application template provides the simplest starting point, providing a basic, default project structure together with sample resources and images.

The Fixed Layout Application, Grid Application, Split Application, and Navigation Application templates are designed to provide a starting point for more complex user interfaces that address a variety of more advanced scenarios.

## Designing and Styling the User Interface with HTML and CSS

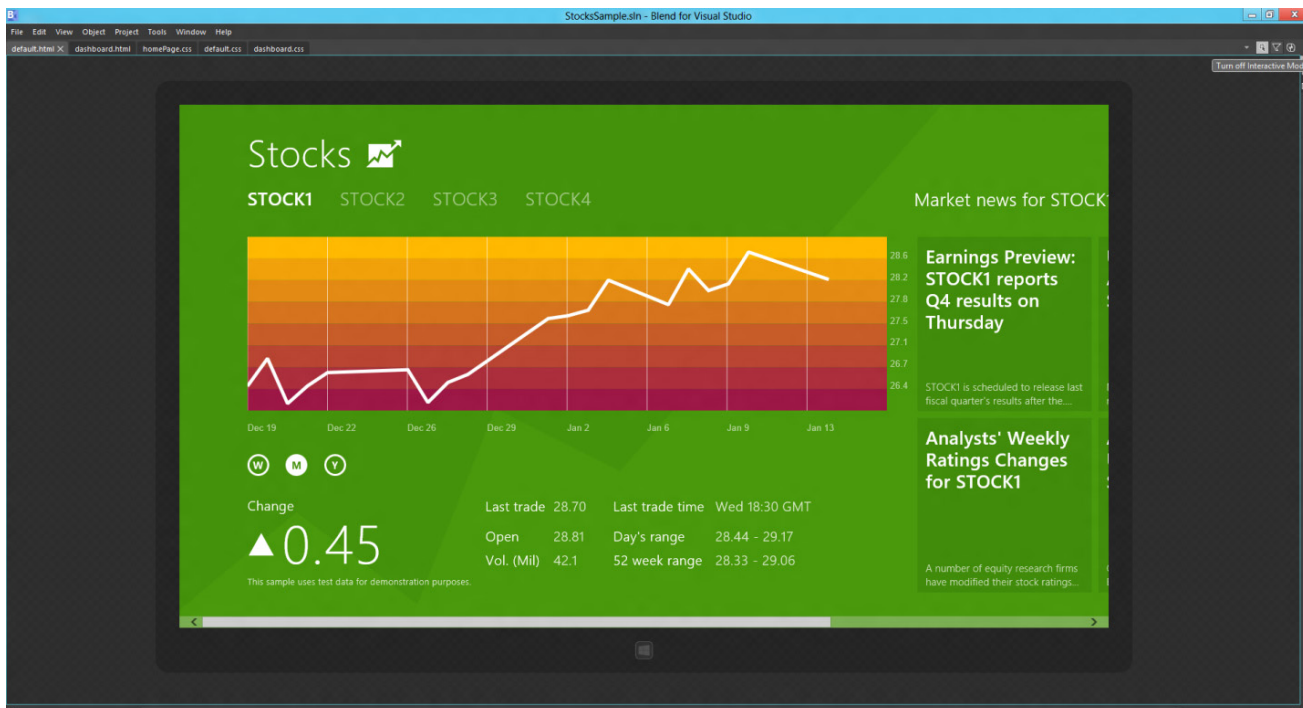
Blend for Visual Studio 2012 has been updated to support development of HTML and CSS user interfaces. The Blend design surface takes full advantage of CSS, enabling you to create, change, and manipulate CSS styles in your applications.



Blend enables you to quickly select items and change their CSS properties. The Property Inspector saves you time by showing only the properties set on a particular element, and reduces frustration by identifying which CSS style is actually being applied to a particular element.

As you make changes, Blend treats markup differently, giving you the control you need. No longer do you click the Bold button and wonder what markup was generated. Instead, you simply assign an ID or Class to an element and then use the property inspector to set properties as a Rule, creating the type of clean, uncluttered code you would write by hand.

With drag-and-drop access to the Windows Libraries for JavaScript, Blend accelerates the delivery of beautiful Windows Store applications. The innovative interactive design mode of Blend enables you to view your application on the design surface running as a live application. This mode lets you visualize different application states and view the generated JavaScript content. When you leave this interactive mode, the application remains paused, while the generated JavaScript elements remain visible and in place. Content that appeared as the application was running remains in view and the property inspector clearly shows which attributes are read-only and which can be styled by using the same CSS and tools used to style the original static content. This workflow accelerates the development process for Windows Store application developers. The ability to quickly experiment with and refine the UI of an application—without the need to continually rebuild—saves a lot of time. Instant visual confirmation encourages developers to experiment more freely and try new ideas.



You can also add your own HTML markup and JavaScript code directly to a project in Visual Studio 2012.

Visual Studio 2012 treats JavaScript, HTML5, and CSS as first-class members of the development language community, providing IntelliSense for a great code-focused experience.

## Testing your Windows Store applications with JavaScript

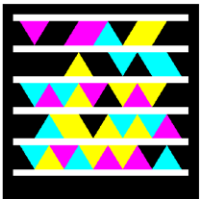
Visual Studio 2012 ships with an updated debugging and diagnostic experience for JavaScript applications, including:

- Simulator. Windows Store applications run in a full-screen, immersive context, and need to be able to respond to hardware events such as screen rotation or calls to geolocation APIs. The application simulator enables a developer to simulate common touch, location, and rotation events.



- **DOM Explorer and Console.** JavaScript developers depend on browser-based tools that enable them to make changes to the DOM of a running web application. With Visual Studio 2012, this experience is part of the IDE, adding a DOM explorer and console window that give JavaScript developers fine control over their running applications.
- **Debugging Support.** The Visual Studio 2012 debugger supports JavaScript. You can define watch lists based on JavaScript variables, set breakpoints, single-step through JavaScript code, and examine and set the values of JavaScript objects.

## Building Windows Store Applications with C# and Visual Basic by Using Visual Studio 2012



If you are familiar with the .NET Framework, Silverlight, Windows Presentation Foundation (WPF), or Windows Phone 7 development, you can use your existing skills and code assets by building Windows Store applications with C# or Visual Basic. Your UI is built declaratively by using XAML markup. For a more complete introduction to using the Visual Studio 2012 IDE, see the Exploring the IDE page at:

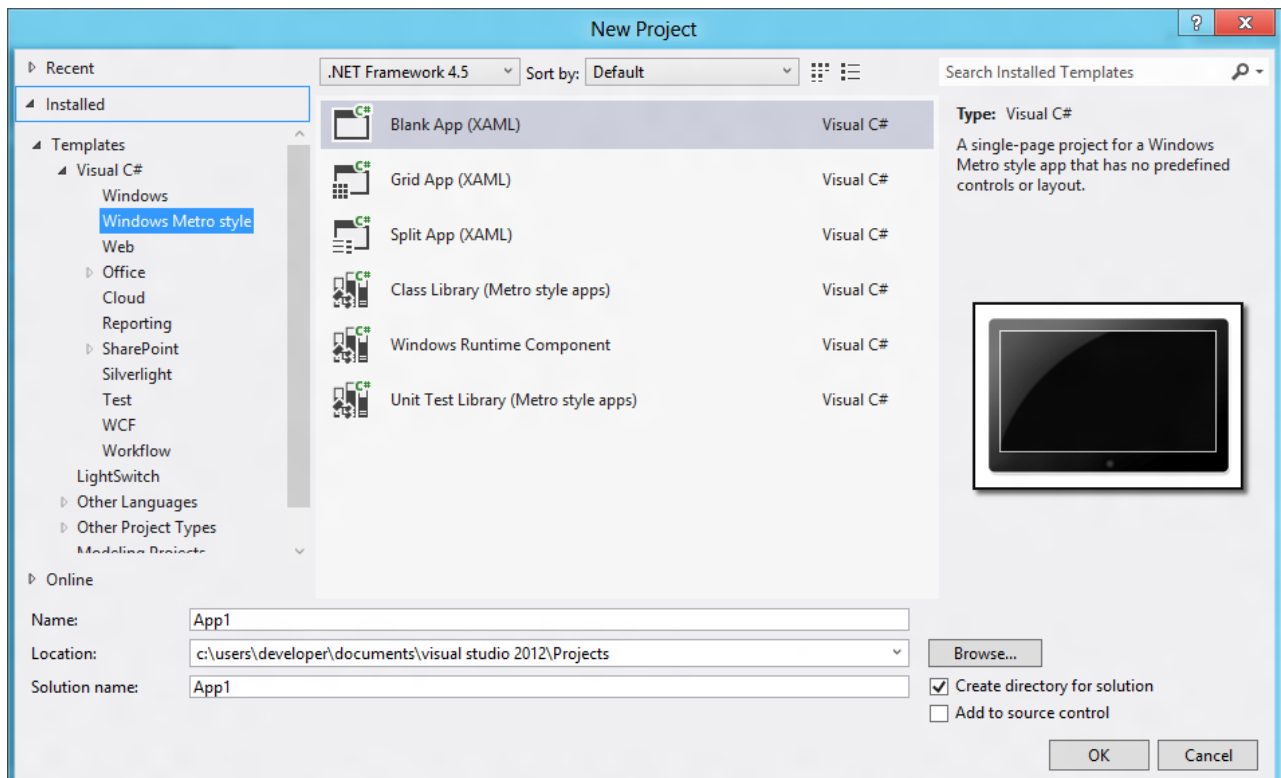
[http://msdn.microsoft.com/library/jj620916.aspx\(v=vs.110\).aspx](http://msdn.microsoft.com/library/jj620916.aspx(v=vs.110).aspx).

## Creating a new Windows Store application with C# or Visual Basic

Visual Studio 2012 can be used to generate the XAML markup for your UI.

When you create a new project, you can start with a blank project or a template.

The templates provide you with a selection of Windows 8 UI designs in order to save you the time of creating your own. The following templates are available to developers creating applications by using Visual C#, Visual Basic, and Visual C++:

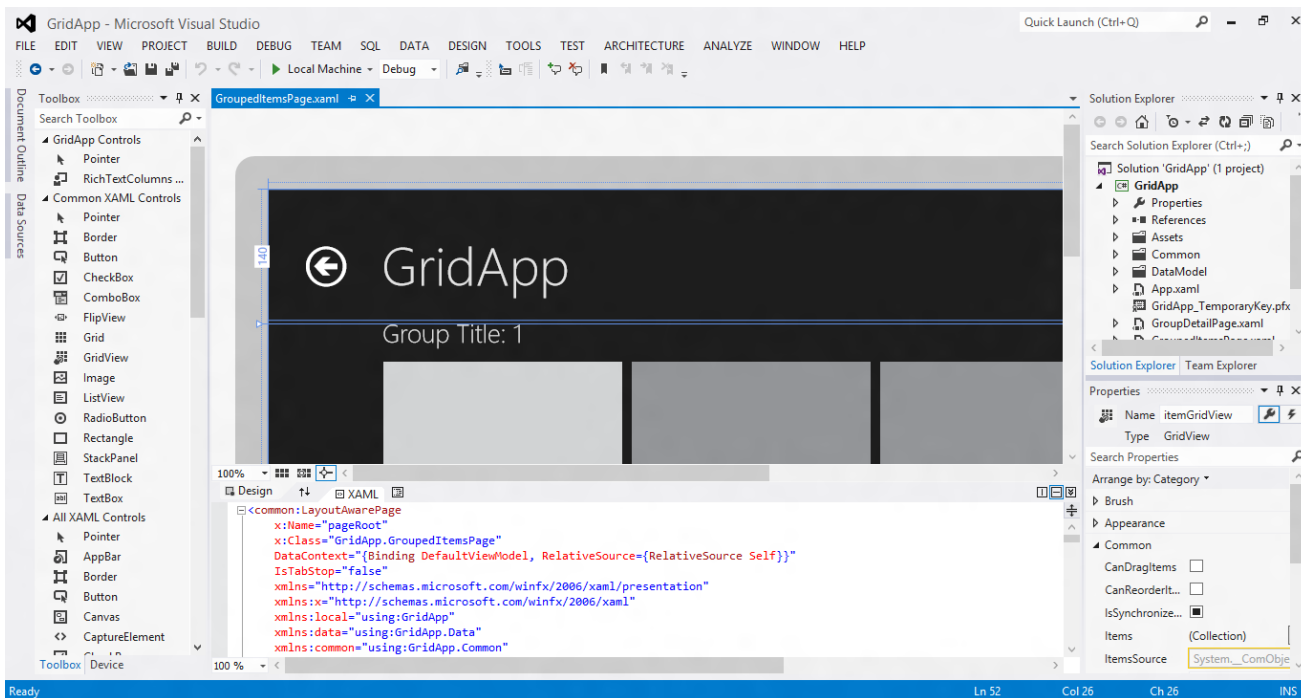


- **Application.** This template enables you to create a unique interface for the application by starting with no defined layout or controls. You must define the layout for the application and add any necessary controls.
- **Grid Application.** This template includes a page with controls laid out in a grid pattern. You can add additional controls or change the content of the controls in the grid according to their requirements.
- **Split Application.** The split application template includes a page split in a master detail format. The left side of the page contains a list of items; clicking or touching an item causes the right side of the page to update with the corresponding information.
- **Class Library.** This template enables you to create a reusable class library for use in Windows Store applications.
- **Unit Test Library.** This template enables you to create unit tests for your Windows Store applications.

## Designing and Styling the User Interface with XAML

Visual Studio 2012 has an integrated visual designer for XAML-based user interfaces for Windows Store applications. You can use the designer to generate the XAML that defines your UI, or you can edit the XAML directly.

The following illustration shows the XAML designer surface, the XAML markup in the code editor, and the Toolbox window:



As a bonus, many of the inconsistencies that occurred between earlier versions of Visual Studio and Blend are no longer present, because both tools now share the same XAML design surface. You can therefore be certain that applications will render in exactly the same way in both tools. This is great for developers, who can now use Blend and Visual Studio interchangeably and without worry, focusing on the strength of each tool: Blend to lay out the structure of the UI and to add the textures, nuance, and beauty demanded by successful modern applications, and Visual Studio for developing the business logic of the application and making the UI respond to customer input.

Developers and designers can also take advantage of the Device Panel to author full screen landscape, full screen portrait, filled, and snapped application view states. The Device Panel will now automatically change visual states when the View is changed. Turning on recording mode enables changes made on the Design Surface or Property Inspector to alter the currently selected visual state, rather than always making changes to the base.

## New features

In general, C# or Visual Basic developers can make use of any new features in the Visual Studio 2012. For more information, see the Exploring the IDE page at: [http://msdn.microsoft.com/library/jj620916.aspx\(v=vs.110\).aspx](http://msdn.microsoft.com/library/jj620916.aspx(v=vs.110).aspx).



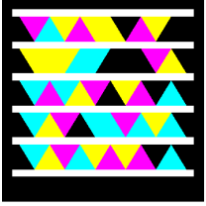
The following sections describe some new feature highlights in Visual Studio 2012.

## Debugging

Visual Studio includes a powerful debugger with many new features to support debugging Windows Store applications, including:

- Robust unit test support by using the Microsoft Test framework.
- Windows 8 simulator (for example, for testing at various resolutions, orientations, layouts, etc. with always-on-top mode for natural usage in single-monitor scenarios).
- Improved remote debugging (for example, by using a tethered device over a wired or wireless network).
- Edit and Continue enhancements (faster bug fixing with ability to EnC around lambda expressions).
- Parallel Watch window, which evaluates arbitrary expressions across all threads.
- Pervasive “flagging” support to enable debugging parallelism at scale.

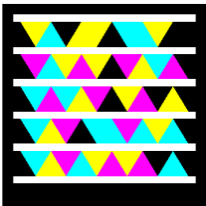




For more information, see Debugging and testing Windows Store applications at [http://msdn.microsoft.com/en-us/library/windows/apps/hh441481\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/windows/apps/hh441481(v=vs.110).aspx)

## Deep asynchronous programming support in WinRT

The model implemented by Windows Store applications is geared towards performing tasks asynchronously, to help ensure that applications remain responsive. Indeed, many WinRT APIs support only asynchronous operations. For example, all UI operations involving dialogs and menus, file pickers, and all file and web I/O operations are asynchronous. In short, anything that can take an indefinite amount of time is handled in such a way that it prevents the application from freezing or becoming unresponsive. Visual Studio 2012 includes debugger support for asynchronous programming tasks, including enabling you to step through and out of async methods as if those methods were synchronous.



To learn more about asynchronous programming with C# or Visual Basic, see the Quickstart: Calling asynchronous APIs in C# or Visual Basic page at <http://msdn.microsoft.com/en-us/library/windows/apps/hh452713.aspx>

## Hybrid applications

You can utilize code written with a compiled language like C++, C#, or Visual Basic in your JavaScript applications for Windows Store. Visual Studio 2012 provides you with a means of authoring an API in a DLL just by writing a class in C# or Visual Basic.

## Concurrency Visualizer

To profile your parallel .NET applications and find performance bottlenecks, Visual Studio 2012 includes the updated Concurrency Visualizer. This tool shows how parallel code uses resources as it runs, including how many

cores it uses, how threads are distributed among cores, and the activity of each thread. This information helps you to confirm that your parallel code is behaving as you intend, and it can help you diagnose performance problems.

## Building Windows Store applications with C++ by Using Visual Studio 2012

Visual Studio 2012 supports C++ developers with a set of easy to use C++ templates for building Windows Store applications. These templates are focused on ensuring that C++ developers can easily build Windows Store applications that take advantage of the touch-first experience of Windows 8.

### Designing the User Interface

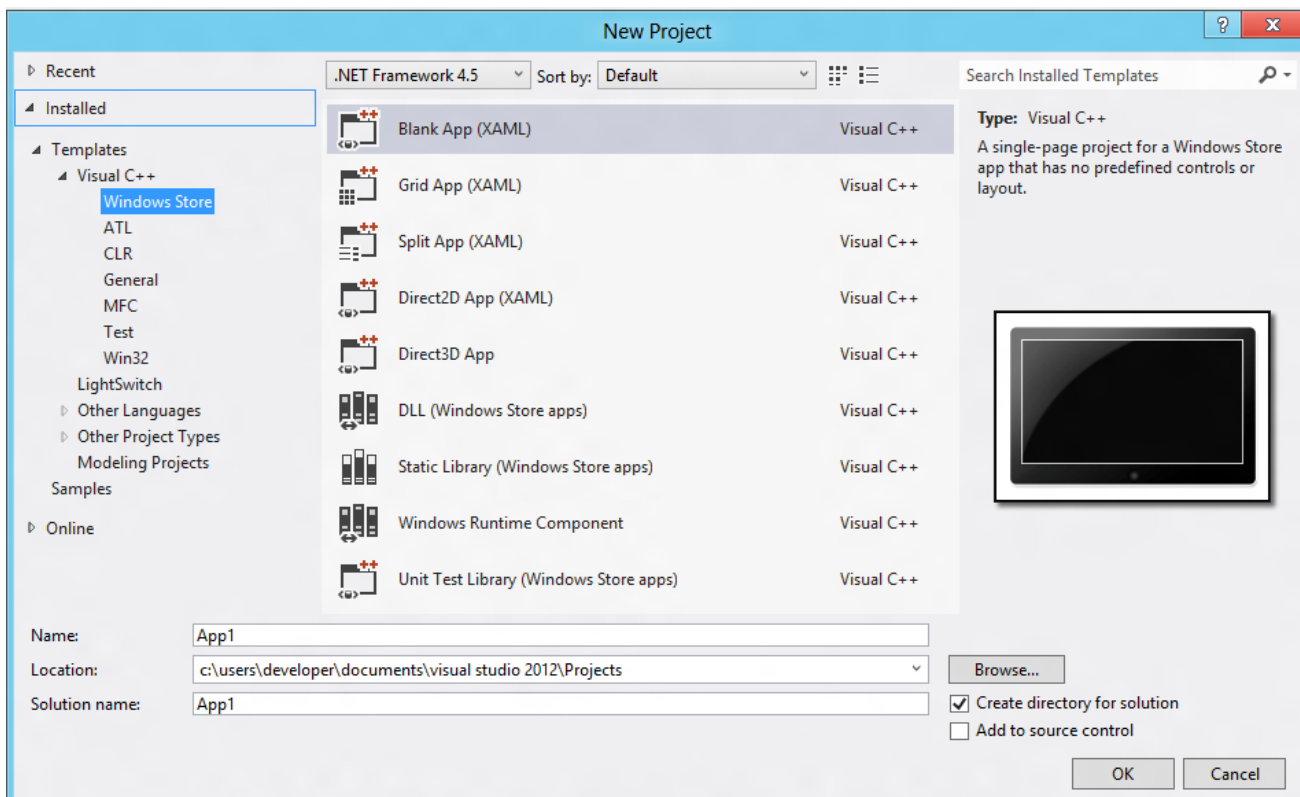
Visual Studio 2012 provides developers with a design surface they can use to drag and drop controls and see the design surface update in real time. This design surface supports XAML for native C++ applications. IntelliSense is available for the properties and events exposed by controls.

C++ developers can use many of the templates available to Visual C# and Visual Basic developers as a starting point for their applications. Visual Studio 2012 also includes a number of additional C++ templates, enabling developers to take advantage of features not available through managed code.

Developers can also take advantage of the following improvements to the standard templates:

- Virtualization support by default. The GridView and ListView controls now take advantage of a header region allowing the bound items to be virtualized.

- Displaying only a subset of items in the GridApp template. Rather than displaying the entire collection of items, only a subset of items is displayed, which will help improve the scalability and performance of your applications.
- Keyboard and mouse navigation, so users can navigate your applications with back and forward buttons, and keyboard shortcuts.



In addition to the standard templates, these additional templates are available to C++ developers:

- WinRT Component DLL. This template enables you to create a dynamically linked server library that you can use in your Windows Store applications.
- DirectX Application. This template enables developers to create applications that use DirectX graphics.

You can use Blend to lay out the UI and generate the XAML code for your C++ application.

## Implementing Application Logic with C++

Visual Studio 2012 provides the same facilities for rapid C++ code development that are available to other languages. Comprehensive editing and IntelliSense support for Windows Store applications with C++ means that developers experience an IDE that is fully aware of new platform capabilities such as objects, interfaces, and types. When designing applications, you can move seamlessly between your XAML that defines the UI and C++ code that implements logic. IntelliSense for your XAML objects is available within the C++ code editor.

The parallel programming support in the C++ concurrency namespace enables developers to take advantage of multi-core and many-core architectures.

In Visual Studio 2012, the Parallel Patterns Libraries have been extended to provide better performance, more control, and richer support for the parallel patterns that developers need most. These include:

- The Parallel Patterns Library, which supports fork-join parallelism, asynchrony, and task continuations.
- Agents and Messaging, which enable developers to express dataflow pipelines that naturally decompose into concurrent units.
- Concurrency safe containers package, which provides common Standard Template Library data structures such as queue, vector, and map.
- A Customizable scheduler and resource manager, to facilitate the smooth composition of the components that implement parallelization.

Graphics Processing Units (GPUs) are increasingly used for performing general purpose computations. C++ Accelerated Massive Parallelism (C++ AMP) delivers support for this approach and helps developers speed up the data parallel algorithms implemented by their applications. C++ AMP is provided in

the concurrency namespace. Developers can simply insert or wrap their data in the new `concurrency::array` or `concurrency::array_view` template classes, convert their code to use the new overload for the `concurrency::parallel_for_each` method to iterate through this data, and then apply the functions available in the data parallel API to implement their algorithms.

## Testing and Debugging C++ Applications

Producing quality applications means testing earlier in the development cycle. Visual Studio 2012 introduces a native framework for unit testing in C++. This helps developers ensure that changes they make to their code deliver the correct experience, and also helps developers find issues during the code-build-debug cycle rather than during software testing.

Visual Studio ships with an updated debugging and diagnostic experience for Windows Store applications built with C++. This support includes:

- A Windows Store application simulator. Windows Store applications are full-screen and need to be able to respond to hardware events like screen rotations or calls to geolocation APIs. The simulator enables a developer on a single computer to run applications and simulate common touch, location, screen resolution, and rotation actions.
- Improved support for remote debugging. You can now perform remote debugging by simply tethering your computer to another computer with a single Ethernet cable or by using Wi-Fi.
- The Parallel Watch window. This window enables you to observe the values of an expression across all threads and processes, and perform sorting and filtering on the result.
- Updated concurrency Visualizer. The visualizer enables you to profile parallel C++ applications to locate performance bottlenecks.
- C++ AMP Debugging. Debugging a C++ AMP application is as straightforward as debugging any other C++ application.

## Designing Games for Windows 8

Windows 8 provides a high-performance platform that can take full advantage of the speed and power of modern computers. With DirectX 11 and the appropriate hardware, your applications can use DirectCompute shaders to perform parallel data processing on the GPU and obtain as much as an order of magnitude more computing power than provided by a CPU alone. Windows 8 is an ideal environment for running real-time, processor-intensive applications such as games.

Visual Studio 2012 with Visual Studio Graphics (VSG) tools has a renewed focus on gaming, letting games developers be more productive and creative. The VSG toolbar in Visual Studio 2012 provides access to resource editing, visual design, and visual debugging tools for writing 2D and 3D games.

### Viewing and basic editing of 3D models.

You can use VSG as a level design tool for game engines or as a code generation tool for game development. You typically create game models by using a third-party 3D modeling program. You then incorporate these models into a content production pipeline that involves modeling, texturing, rigging, animation, and level creation. VSG supports viewing and basic editing of standard 3D model formats. This allows VSG to load models created by other tools, or for other tools to edit models created with VSG.

### Viewing and editing of images and textures with support for alpha channels and transparency.

VSG includes powerful image and texture editing tools that upgrade the existing image editing capabilities of Visual Studio with support for alpha transparency, additional image formats, textures, and extensibility. The existing Visual Studio 2010 image editor is retained in order to support legacy icon and cursor file formats.

VSG tools also support alpha blending or overdraw. This applies to pixel editing operations such as moving/resizing a region, pasting an image onto the current image, and image filters such as rotating a region by 90 degrees. For example, when moving a selected region, that region can be blended with the portion of the image under it or the region can overwrite all data under it.

### Visually designing shader programs and effect files

Game projects typically involve a large number of materials, effects, and shaders. These items are used to color, texture, and apply special visual effects to objects. VSG includes a graphical design tool that you can use to create effect files. You can then use these files as materials on 3D models. VSG also enables you to define pixel shaders.

### Debugging and diagnostics of DirectX based output

Visual Studio 2012 provides a built-in implementation of functionality based on the existing PIX for Windows tool for debugging DirectX-based games and applications. It includes and enhances many of the features in PIX for Windows in order to help you diagnose rendering problems.

Visual Studio 2012 also includes a first-class High Level Shader Language debugging experience, fully integrated into the Visual Studio debugger, to help you identify and correct DirectX bugs.

You can extend the VSG tools by using JavaScript programs. For example, you can introduce new commands and capabilities to the various tools; add support for new import and export file formats; generate code snippets, project templates, and code-behind files; and integrate with popular third-party game engines that provide special effects generation tools.





## Summary

From the perspective of a developer, Windows 8 is a powerful and adaptive platform for your next generation of applications. We have endeavored to implement a modern and extensible framework to support your development efforts, regardless of which language and programming paradigm you prefer.

Windows 8 and Visual Studio 2012 provide the ideal development toolset, enabling an end-to-end workflow that takes you from your initial idea all the way through to your application being deployed to the Windows Store. Tools that enable you to rapidly create the structure of your applications and test them help you to discover user-experience and application-flow issues early in the development cycle, when these issues are easy to fix. The flexible workflow between the development tools allows you to function to your strengths, seamlessly switching from one tool to the other, and enables you to experiment and deliver the kinds of compelling and differentiated application experiences your customers expect on modern devices.

Whatever your development background, we have the tools, platform, and development languages to set you on the path to building amazing applications.



# Developing for the Web

Web developers use a range of technologies to create interesting, attractive, and interactive sites. HTML5, CSS3, and JavaScript are the most common technologies used for this purpose because they support the widest range of web browsers. As JavaScript has matured as a technology, frameworks such as jQuery have been created to help developers to code faster. Visual Studio 2012 supports many of these technologies and simplifies the development process to help developers code faster and to ensure that their code conforms to the latest standards.

Visual Studio 2012 is the tool for web developers who want to create a simple web application quickly. Equally, Visual Studio 2012 is ideal for a team of professional developers working on a large, rapidly evolving web application, because it offers full support for source control, testing, code reviews, and deployment.

In addition to supporting the latest standards, developers need a modern web platform. Visual Studio 2012 delivers just that by providing:

- Support for working with data, including simplified development tools that make development, testing, and debugging easier.
- Support for authentication services from external providers through Open Authorization (OAUTH).
- Support for mobile devices, including mobile-optimized views and pages.
- Support for hosting sites and services in the cloud with Windows Azure.

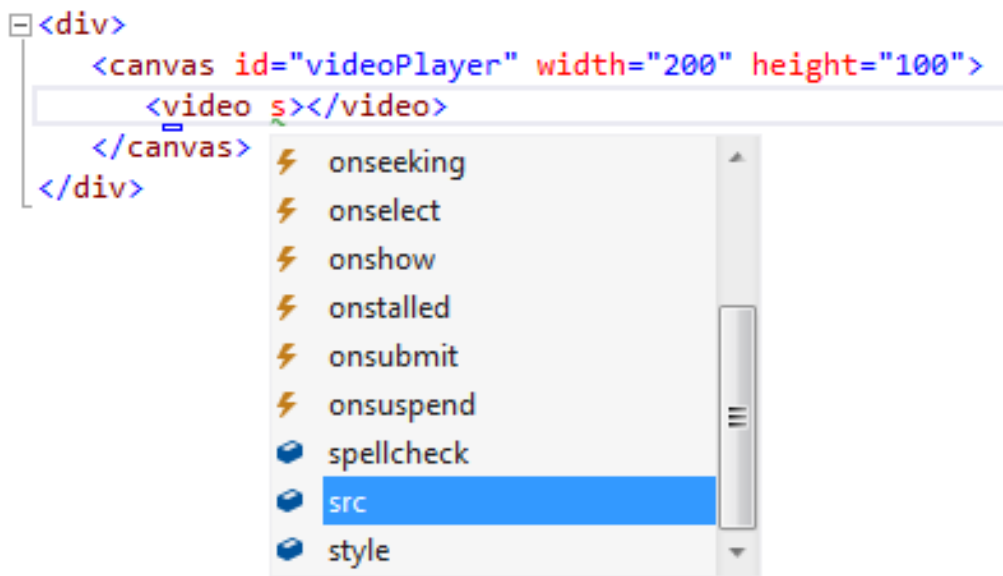
Debugging and resolving problems is often a significant part of the development process. Visual Studio 2012 helps you to find problems faster by offering numerous improvements to debugging tools, plus new tools that bring browser DOM inspection right into the IDE.

## Develop to Standards

### Use the latest HTML, CSS, and JavaScript with jQuery

Users accessing a web application employ a variety of browsers. Some browsers support plug-ins while others don't, and all browsers have quirks that affect how they display a site. Web standards exist to help ensure that web applications look and work correctly regardless of the browser being used. HTML, CSS, and ECMA scripts are the standards for developing cross-platform web sites. Previous versions of Visual Studio supported developing web applications by using these standards, but Visual Studio 2012 extends and significantly improves this support.

Visual Studio 2012 includes full IntelliSense support for HTML5, making it easier to develop conforming web sites. In addition, the code editor has new productivity features that simplify web development. For example, Visual Studio 2012 can help you correct simple coding mistakes. The following figure shows how IntelliSense supports the new HTML5 video element:



## Use CSS to Style Your Site

All but the most basic sites require styling, normally accomplished by using CSS. The CSS editor in Visual Studio 2012 has been updated to support the latest CSS3 standard and incorporates full IntelliSense.

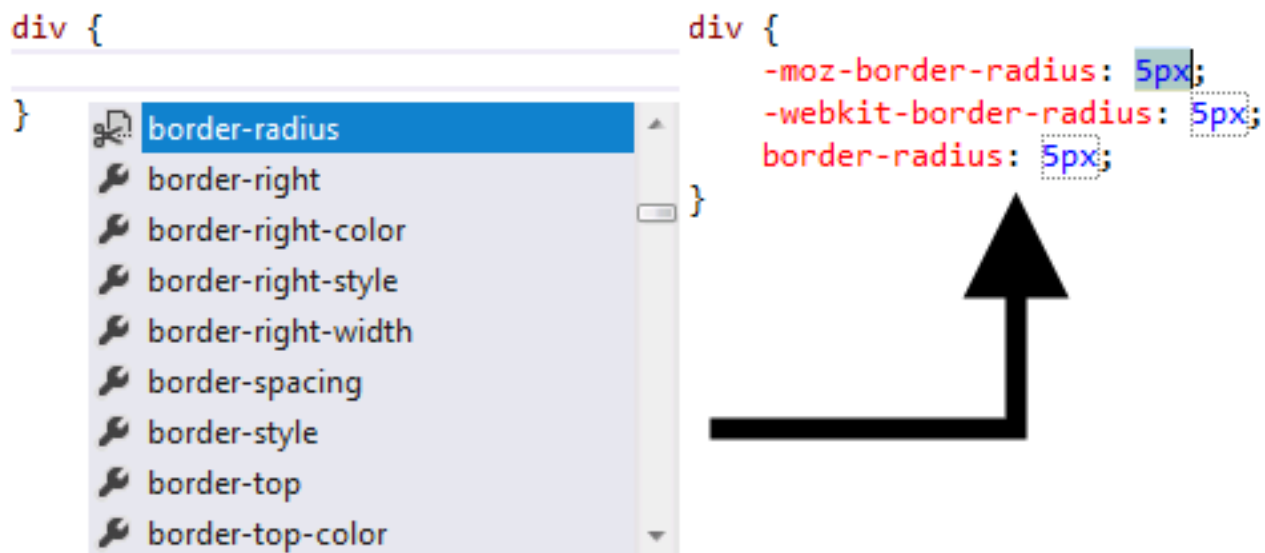
Older browsers typically do not fully or properly support the W3C standards. Developers expect to include various browser-specific workarounds to ensure that their sites work correctly in these older browsers. Previously, such workarounds would be marked as errors in the editor, but Visual Studio 2012 recognizes common browser workarounds and parses them correctly.

Visual Studio 2012 also includes many other productivity improvements, such as:

- Color picker. Use the new color picker to choose colors while editing CSS. The color picker includes support for opacity and comparison of two colors.



- Snippets. Snippets make it faster for developers to create CSS. Visual Studio 2012 includes many snippets that can be used out of the box, and a developer can easily define their own snippets.



- Commenting. Use commenting to highlight a section of CSS and place comments within the code.

```
div {
    /*-moz-border-radius: 5px;
    -webkit-border-radius: 5px;
    border-radius: 5px;*/
}
```

- Automatic indentation. The editor automatically indents the CSS hierarchically as it is developed.

```
div {
    -moz-border-radius: 5px;
    -webkit-border-radius: 5px;
    border-radius: 5px;
}

div a {
    color: red;
}
```

- Region support. Define regions of markup similarly to easily show or hide focus for a specific section of CSS.

```
/*#region Customizations */
div {
    -moz-border-radius: 5px;
    -webkit-border-radius: 5px;
    border-radius: 5px;
}

div a {
    color: red;
}

/*#endregion */
```

- Outline support. Outline sections of code to identify or hide code blocks.

```
div {  
    -moz-border-radius: 5px;  
    -webkit-border-radius: 5px;  
    border-radius: 5px;  
}
```

## Use jQuery and JavaScript to Make Your Site Interactive and Responsive

In addition to creating visually compelling web applications, developers often need to create highly interactive and responsive sites. JavaScript is the standard scripting language used by all modern browsers. Using JavaScript on a web site has many advantages, including:

- Faster responses to user interaction by removing the need for a page refresh every time the user changes an option or clicks a button.
- Reducing load on a server by pre-validating data in the browser. Data that has already been validated on the browser often passes any validation required by the web server. This reduces repeat requests for submitting web forms.

Visual Studio 2012 incorporates the JavaScript engine used in Internet Explorer 9. This parser significantly improves the accuracy of JavaScript parsing when compared to previous versions of Visual Studio, and ensures that the behavior in Visual Studio 2012 matches that of the browser as closely as possible. Visual Studio 2012 also provides support for script loaders that can improve page performance and reduce the complexity of developing a site that contains many JavaScript files. Developers simply select which JavaScript files they are using, and Visual Studio provides full IntelliSense support regardless of whether the file is loaded by using a master/layout page or a script loader.

In addition to the other changes, the JavaScript editor in Visual Studio 2012 contains a number of productivity improvements that align JavaScript with other .NET languages such as Visual C#, including:



- Go To Definition. Choose **Go To Definition** to move the cursor to the object or function definition, even if it is located in a different file.
- Function overloads. Use JavaScript function overloads to add multiple functions with the same name. IntelliSense continues to parse code without displaying an error.
- Brace matching. Select an opening or closing brace to automatically highlight the corresponding closing or opening brace.
- Outlining. You can expand or collapse functions and code blocks to highlight the structure of your code.
- Smart indenting. The editor automatically indents the script hierarchically, as it is being written.
- XML documentation comments. Use commenting to add comments to object and function definitions, and to place comments within the script. Comments are displayed by IntelliSense when writing code.

## Use IntelliSense to Develop with Ease

IntelliSense is a key Visual Studio feature that has been enhanced in Visual Studio 2012.

IntelliSense can make a developer significantly more productive by reducing the time spent typing and by providing a list of options from which a developer can choose. This reduces the need for the developer to refer back to the object definition for the correct property name and casing. In some cases, the list of options can be overwhelming. An auto-reduce feature (as they type, developers are prompted with a shrinking list of appropriate possibilities for matching property names) has been extended to support HTML, JavaScript, and CSS.

Visual Studio 2012 includes more triggers for IntelliSense, helping to ensure that coding help is displayed whenever it may be useful. Also, shortcuts in IntelliSense further reduce effort by providing fast access to common properties.

## Easily Develop Sites

.NET developers building a web site typically choose between an ASP.NET Web Forms project or a Model View Controller (MVC) project. In MVC applications, developers define classes that describe the data being used by the page (this is the model). Views define the interface seen by the user, and a developer or designer can change a view without changing any other code. Application logic is handled by the controller. This approach separates application logic from views, making for a maintainable web application. Controls in the view are bound to properties of the model at runtime. The developer does not need to write code to perform the binding because it is handled by the framework. Developers using Web Forms previously had to write code to bind controls to properties. However, in Visual Studio 2012 developers and designers can use object binding in Web forms applications, simplifying the code and reducing repetitive development work.

While developers could always write their own JavaScript code to validate user input on the client, MVC provides a simple syntax to annotate a control without needing to write custom code. Visual Studio 2012 brings this power to ASP.NET Web Forms. With previous versions of Visual Studio, MVC developers could simply annotate properties in a model and the framework would add client-side validation automatically. Visual Studio 2012 has brought this functionality to ASP.NET Web forms developers, who can now also annotate properties for automatic client-side validation. Validating input on the client can reduce the number of invalid submissions to your site and provides a more responsive interface for a user.

In addition to the new features added to Web Forms, Visual Studio 2012 includes several IDE enhancements to make the development process easier, including:

- Chrome tasks in the editor. Perform chrome tasks while in the code editor window, enabling you to configure data sources and similar chrome tasks without switching to design view.
- Event generation in the source view. Generate a method stub for an event when declaring the appropriate attribute in the code editor window, without the need to switch to design view or to generate method stubs manually.

- Renaming ASPX pages. More easily change the name of an ASPX page. Visual Studio 2012 automatically updates all references.
- Extracting to a user control. Select and extract the required markup to create a new user control. Visual Studio 2012 updates the page to include all necessary tag prefixes and registrations, and includes the control in the correct place on the page. Re-use the control elsewhere on the site as required.

## Find Problems Faster

### Test, Debug and Deploy with Confidence

A developer creating a web application must test and debug before deploying the application to a production server. Because developer computers can be very different from production servers, developers must be confident that testing and debugging on their computers will match the results on the production server. One issue facing developers is differing versions of the .NET Framework. To address this issue, Visual Studio 2012 enables developers to create applications that reference earlier versions of the .NET Framework while ensuring that they do not use features added in later versions.

When developing a web application, developers typically host the application on their workstations for testing and debugging. Previously, the simplest approach for this was to use the Development Web server provided with Visual Studio. While this was easy to configure, there were significant differences between the Development Web server and Internet Information Services (IIS) software typically used on production servers. One solution to this problem is to install IIS on the developer's computer. While this strategy closely matches the production environment, it is subject to limitations. For example, the version of IIS is controlled by the operating system.

Visual Studio 2012 ships with IIS Express, a lightweight web server that includes all of the core functionality of the current version of IIS. Using IIS Express, Visual Studio 2012 enables developers to debug and test sites with minimal configuration—including the power to perform a custom configuration, if

necessary—and without the added requirement of using a full IIS installation. Developers can configure IIS Express through Visual Studio. In addition, administrative privileges are not required to configure IIS Express, making it a viable option in environments where using administrative rights is not possible.

Visual Studio 2012 also ships with a new Publish Web wizard, providing not only a new user interface, but a number of features that help ease the complications and stresses of deployments.

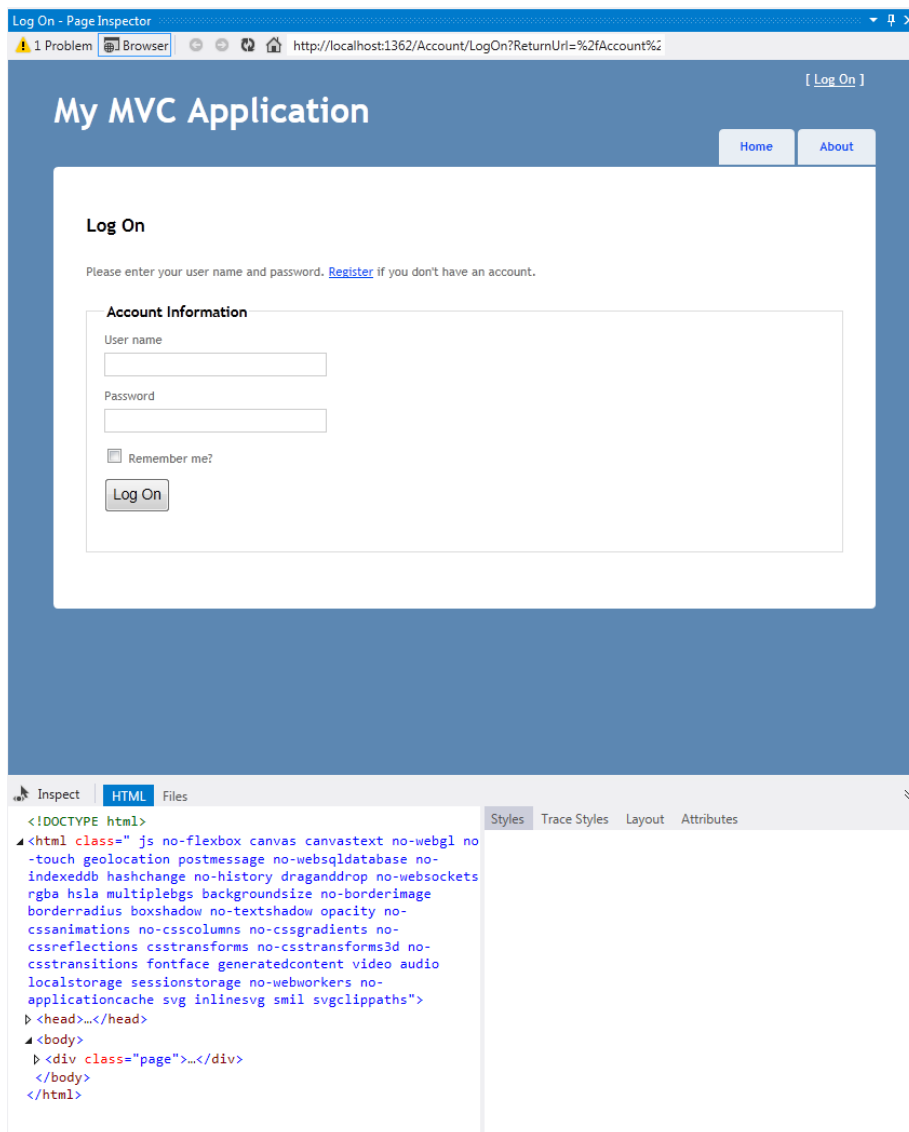
The screenshot shows the 'Publish Web' wizard in Visual Studio, specifically the 'Production' tab and the 'Connection' step. The wizard is titled 'Publish Web' and has a subtitle 'Publish web application'. On the left, there is a navigation pane with four items: 'Profile', 'Connection' (which is highlighted with a blue arrow), 'Settings', and 'Preview'. The main area is titled 'Production' and contains the following fields and controls:

- Publish method:** A dropdown menu set to 'Web Deploy'.
- Service URL:** A text box containing 'production.contoso.com'.
- Site/application:** A text box containing 'www.contoso.com'.
- Destination URL:** A text box containing 'http://www.contoso.com'.
- User name:** A text box containing 'administrator'.
- Password:** An empty text box.
- ☐ Save password
- Validate Connection:** A button.

At the bottom of the wizard, there are four buttons: '< Prev', 'Next >', 'Publish', and 'Close'.

## View and Edit Web Pages in the IDE

To debug their sites, web developers often use tools in browsers to view the (DOM) of the rendered page and to make changes to rectify issues. Typically, the developer then copies these changes back to their development environment. Visual Studio 2012 includes a new tool, the Page Inspector, to simplify this process. By using the Page Inspector, developers can make live changes to the DOM and edit CSS rules, and when they achieve the desired result they can easily copy these changes to the source file. The Page Inspector also enables developers to reverse engineer—with just a single click—the rendered HTML, CSS, or JavaScript to the source file.



Browsers typically support a dynamic approach to debugging in which the developer can interact and run commands on an ad-hoc basis. Visual Studio 2012 adds the JavaScript console to the development environment, enabling developers to set break points and single-step through their code. These break points are persisted when the application stops, and are available the next time the developer runs the application in Debug mode.

## Develop for a Modern Platform

### Develop for Mobile Devices

In almost all cases, developers want their sites to be accessible on as many devices as possible, ranging from desktop and tablet computers through smartphones. In the past, developers commonly created custom versions of their sites to target various devices. This approach resulted in a significant challenge, requiring complex, costly maintenance. Recent updates to web standards, however, make it easier to develop cross-platform sites, and most modern devices support the latest versions of these standards.



Using the latest web standards helps to ensure that a site displays properly on most devices. Sometimes developers must create a custom view of a page for mobile browsers. This can be achieved in two ways. The first way is to have the server return a correctly formatted page. The second method is to have the page include styles that are only applied on devices that meet particular criteria (for example, a screen size smaller than a specified value).

Visual Studio 2012 helps make development for mobile devices easier by supporting tools that optimize a page for a mobile device. For example, Visual Studio now supports jQuery Mobile, which includes controls optimized for mobile devices. By using a combination of these technologies, developers can create sites that look great and perform well on mobile devices.

Visual Studio 2012 also supports user agent detection. Developers can use this feature to detect the browser and platform of a requesting device and to then return an appropriate page version. For example, the application may return a version of a page with larger buttons if the request comes from a touchscreen device.

## Easily Work with Data and State Information

Managing data and handling information about application state is one of the major differences between developing a desktop application and developing a web application. With applications that run on a local computer, state information and data can be stored on the local disk in the context of the current user. For web applications, many users make requests to the same web server. Data must be stored between requests and identified to ensure that each user retains the appropriate state information. For large web applications hosted by a web farm, this data must be shared among servers. This is a common challenge for web developers, and there are many approaches to meeting this challenge.

One approach is to save data in cookies that are transmitted to the server with each request. The cookie identifies the request and is used to retrieve data on the server. Visual Studio 2012 includes a new integrated development database called SQL Express LocalDB, a lightweight database engine that installs and starts quickly and provides full compatibility with both SQL Server and SQL Azure. This ensures that developers can develop, test, and debug



sites on their development computers with confidence that the sites will work as expected when deployed to a production web server using a SQL Server database. A site developed by using LocalDB will work in the cloud or with an instance of SQL Server with only minimal configuration changes. To make the process easier, Visual Studio 2012 provides tools that enable developers to seamlessly deploy to the cloud or to an on-premises instance of SQL Server.

When developing a site that uses a SQL Express database, another common problem has been that the database file would get locked by Visual Studio or by the web site, and a database update may fail as a result. LocalDB runs in the context of the current user and avoids this problem.

## Integrate with Other Technologies

Writing JavaScript can be a time-consuming process, but developers can save a lot of time during development by leveraging existing tools. For example, the jQuery library includes shortcuts that can save significant time and effort. Visual Studio 2012 fully supports jQuery and makes it easy to integrate with web sites.

Authentication is another area that can be time consuming to implement. While ASP.NET has always had a powerful authentication provider, sometimes developers do not want to implement it, if, for example, the developer did not want to host a custom database. In Visual Studio 2012, ASP.NET web pages support OAuth, which enables developers to integrate with external authentication providers, including Google®, Yahoo®, Facebook®, Twitter®, and Windows Live.

Using an external authentication provider can simplify the registration process for the users of a web application and can save the cost of implementing an authentication provider. This can improve the user experience, which in turn can help to achieve better sign-up rates. While this feature may be useful to some sites, ASP.NET authentication is still available to developers and still fully supported. ASP.NET authentication configuration benefits from some of the IDE improvements in Visual Studio 2012.

Visual Studio 2012 includes an extension manager that can be used to add new libraries and capabilities from trusted third parties. Additionally, the NuGet third-party package management system (available in the gallery

as an extension) enables developers to easily access other useful code. For example, NuGet enables developers to include the Error Logging Modules and Handlers library (ELMAH) in their project with minimal effort.

## Improve Performance

Modern web sites may present large amounts of complex content, images, audio, and other information. With a wired Internet connection, a user browsing such a site typically experiences reasonable performance. However, more and more users are viewing sites on mobile devices whose wireless connections are not as fast or reliable as wired connections. Additionally, as the web has matured, performance expectations have increased. Web sites are now expected to function as quickly as desktop applications. Reducing the volume of data being transferred is one way that developers aim to improve the performance of their web applications. Visual Studio 2012 includes features to help ensure that sites load rapidly and with minimal bandwidth use. Visual Studio 2012 supports asynchronous requests using jQuery and JavaScript; these requests can update just a small part of the page, which can be significantly faster than completely reloading a page.

Visual Studio 2012 also supports deferred loading, enabling developers to create pages in which the content loads first and the scripts are loaded afterwards. While the total time to download the page may not change significantly, the perceived load time can be much faster because the page renders while JavaScript files are still being downloaded. In some circumstances, deferred loading may enable scripts not to be loaded at all, saving bandwidth and delivering a real performance increase.

In addition ASP.NET now supports optimization, bundling, and minification for script and CSS files. These features help developers to further reduce the bandwidth requirements of their web application.

## Develop Web services

Web services provide a mechanism for developers to expose business operations to the outside world. Developers can create applications to consume a service, and these applications can run in the web, on the desktop, on a mobile device, or on almost any platform that has the required connectivity. Service developers can create services by using Windows Communication Foundation (WCF). Visual Studio 2012 makes it easier than ever to develop WCF services. For developers choosing between ASP.NET and the WCF Web API, there is now one choice: The Web API is now part of ASP.NET MVC.

### Develop Asynchronous Services

Services can contain long-running operations. Service consumers often prefer to invoke asynchronous operations to ensure that calling an operation does not make the calling application unresponsive. Visual Studio 2012 makes developing asynchronous operations in a web service easy. Service developers can use the `async` and `await` keywords in the Visual C# and Visual Basic languages to define asynchronous methods that implement these operations. For consuming these operations, developers can simply create a service reference and Visual Studio generates a proxy class to correctly handle the asynchronous behavior.

Defining an asynchronous operation:

<CODE SNIPPET>

```
[OperationContract (AsyncPattern=true)]  
Task<string> GetData(int value);
```

</CODE SNIPPET>

Consuming the service:

<CODE SNIPPET>

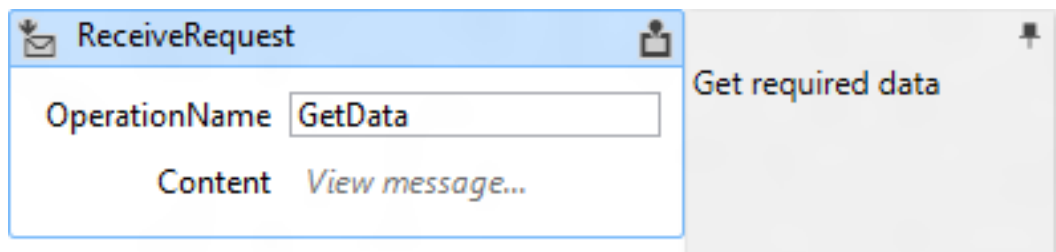
```
Service1Client client = new Service1Client();  
return await client.GetDataAsync(3);
```

</CODE SNIPPET>

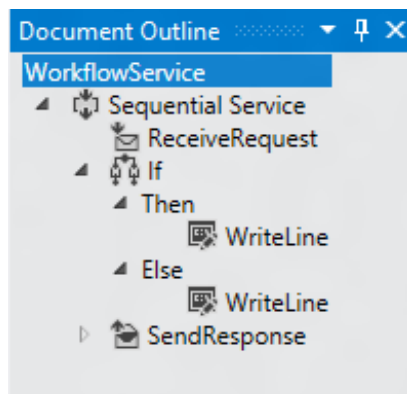
## Use Workflows to Model Services

Business processes can often be modeled as workflows. In Visual Studio 2012, you can easily create services based on workflows by using the workflow designer. The designer is highly visual, making it simple to include other stakeholders in the development process. In Visual Studio 2012, the workflow designer has been improved and benefits from many new features, including:

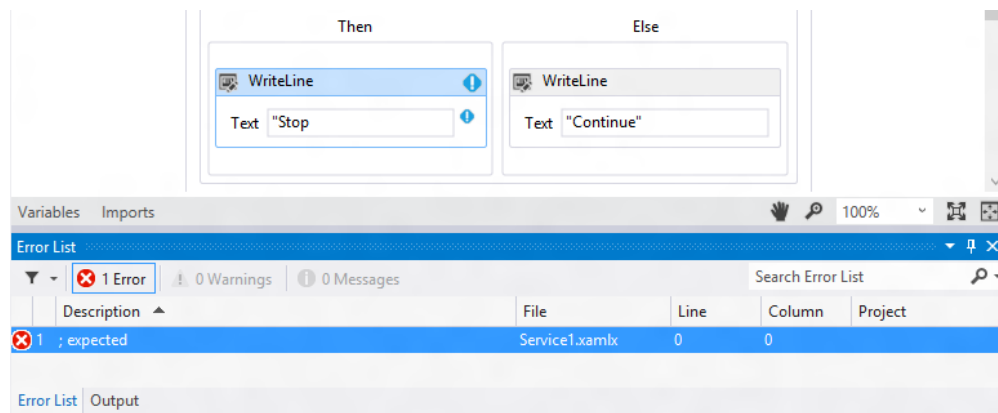
- Activity annotations. You can add annotations to provide additional developer documentation.



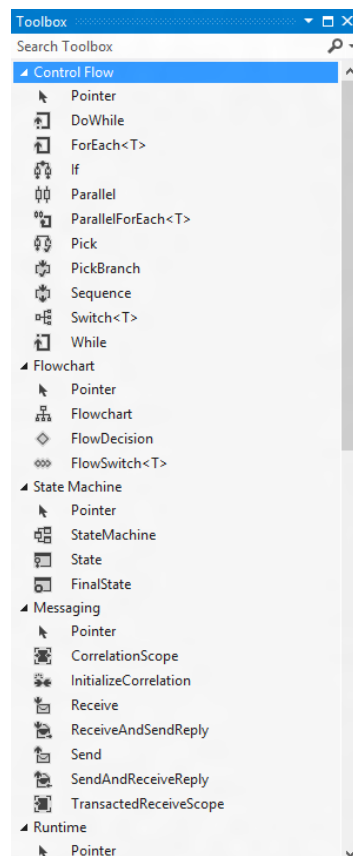
- Tree view. You can navigate large workflows more easily by viewing the structure.



- Pan gestures. You can easily pan through the designer if the workflow does not fit on a single page.
- Error identification. You can find and directly navigate to errors in the workflow designer from the error list.



- Expanded activity pallet. You can create workflows faster by using the new activities available in Visual Studio 2012.

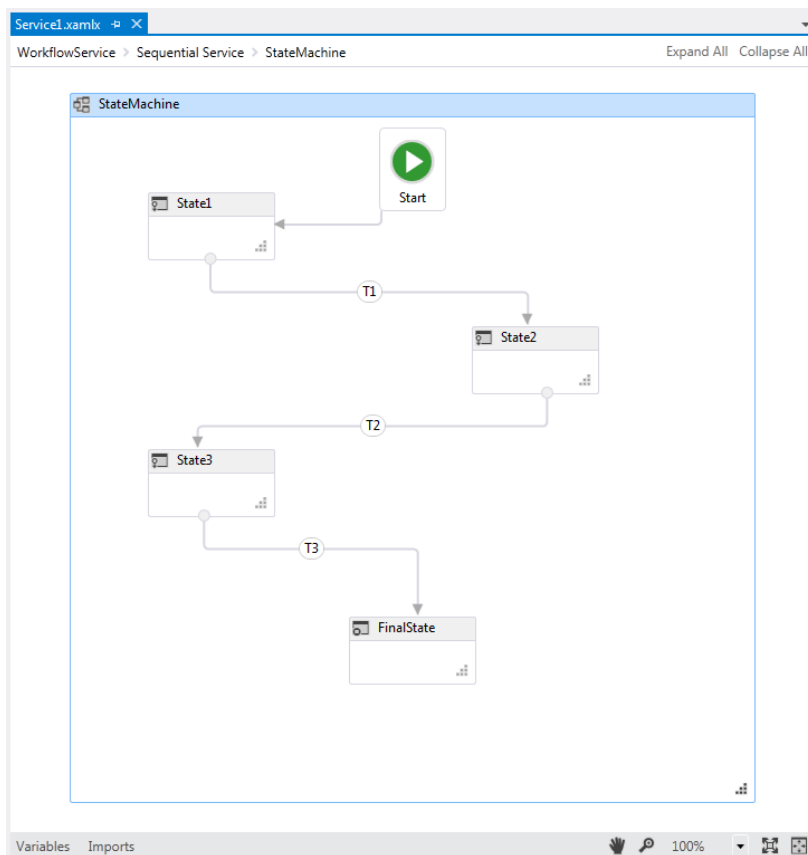


- Consistent syntax for workflow expressions. If you are using Visual C# to build workflows, you can now define expressions by using C# syntax. If you are using Visual Basic to build workflows, you can continue to use Visual Basic syntax in expressions.

## Repurpose an Existing Service in a Workflow

A developer sometimes needs to create a workflow based on an existing service contract, for example, to upgrade an existing service or if the service contract has been defined by another department or company. Visual Studio 2012 supports contract-first services, enabling the developer to create workflows based on existing contracts.

State machines can often be used to model business processes. Visual Studio 2012 adds support for finite state machine workflows. A developer can choose to use either code or the visual canvas to create a finite state machine workflow, making it easy to include such a workflow in any service.



## Maintain and Manage Workflows

A typical workflow evolves over time. Business processes change and consumer demand follows the latest trends. Developers must adapt their applications to ensure that they remain relevant. For short-running workflows, the developer may be able to simply update the workflow without impacting current users (or by causing minimal downtime). For long-running workflows, this may be more of a challenge. Workflow instances may be active or persisted (the durable capture of a workflow instance's state, independent of process or computer information), and stopping them may not be an option. Visual Studio 2012 provides two options to solve this problem: First, workflow versioning enables developers to track which variant of a workflow is being used by a particular running or persisted instance. The developer can run workflows side-by-side (SxS), enabling current workflows to complete by using the old version of the workflow, but creating all new workflows by using the new version. Or second, the developer can choose to upgrade running instances of a workflow to the new version, ensuring that any new requirements are correctly applied.

### <SCENARIO IN PRACTICE>

Contoso is a fictional company that sells online software and uses a workflow to model the purchasing process. During a recent promotion, every customer received a free download of an e-book. When the promotion ended, Contoso management decided that anyone who started an order while the promotion was running should still receive the free e-book. Developers used Visual Studio 2012 to deploy an updated version of the workflow that did not include the free e-book. They used a side-by-side deployment, which meant that customers who had already started the purchase process got the free e-book, but other customers did not.

Several months later, a regulatory change required every customer to accept new terms and conditions. Contoso developers used Visual Studio to deploy an updated version of the workflow that included the new step. They chose to upgrade running workflows to ensure that every customer accepted the terms and conditions.

### </SCENARIO IN PRACTICE>



## Use Full-Duplex Communications

Services are often developed to WS-\* standards, and operations are usually implemented by using request-response messaging in which the consumer always initiates transactions. There is, however, growing demand for full-duplex communication, where either the consumer or the service may control the transfer of data. Visual Studio and the WCF Web API include support for developing duplex services exposed over HTTP by using the new WebSockets standard. Developers can use the WCF Web API to create Representational State Transfer (REST) services that provide real-time services to clients on virtually any platform.

## Deploy Services Quickly

Visual Studio 2012 also makes deploying and consuming WCF services easier. Default project configurations have been updated with smart defaults that often are suitable for deployment to production with no or minimal changes. If the developer needs to edit the configuration, the editor now includes better IntelliSense and improved validation to help ensure configurations are created correctly.

To protect a service, the service can now inherit settings directly from IIS. This feature enables configuring security for the service only once, including HTTPS and authentication, and the service automatically uses these settings with no additional configuration.

For consuming applications, WCF publishes the service metadata as a single WSDL document, making the service easier to consume. Developers can choose to use lightweight messaging by using SOAP over UDP, according to their needs.



## Summary

Customers expect web applications to look great on any platform. They should be functional and fast. Web developers can help to ensure the applications they develop are responsive, fully featured, and attractive by developing for the latest standards. Web services enable developers to expose business logic that developers can integrate with web applications across platforms and devices. Visual Studio 2012 includes all of the tools necessary for web developers to create applications that conform to the appropriate standards, meet user expectations, and function perfectly. From initial planning through development, testing, and deployment, Visual Studio 2012 includes the tools to make creating web applications easier and faster.

Visual Studio 2012 also makes it easy to develop services to use in a service-oriented architecture. Developers can easily create services based on workflows and they can create asynchronous services, helping to ensure that consuming applications remains responsive during service calls. For services providing real-time information, developers can implement full duplex services and exploit Web Sockets.



## Developing for the Cloud

Microsoft Windows Azure provides a highly scalable and reliable hosting infrastructure and toolset for applications and services running in cloud. Windows Azure is an open platform and developers can choose to build applications by using a variety of tools and web application frameworks with programming languages such as Java or PHP. However, most developers are likely to use Visual Studio 2012 as the development environment, to use C#, Visual Basic or F# as the programming language, and to use the .NET Framework to provide the runtime services. Developers working in this environment will find that Visual Studio 2012 makes developing, deploying, and testing applications and services for Windows Azure very easy. Additionally, the tooling the Visual Studio 2012 provides enables development teams to implement comprehensive diagnostics and tracing, while Visual Studio Team Foundation Server support Azure development throughout the entire application lifecycle.

Deploying an application in the cloud ensures that the application can scale as the business grows. Windows Azure runs application instances, each of which is a virtual server running in a datacenter managed by Microsoft. The infrastructure provided by the datacenters helps to ensure balanced and responsive performance. An administrator can choose the number of instances and the volume of resources such as memory and computing power to allocate when the application is deployed. As demand increases, an administrator can upgrade an instance to have more resources or add additional instances. Microsoft also provide utilities that enable an administrator to configure automatic scaling; this feature can dynamically start further application instances during peak working hours, and then stop surplus instances at off-peak times. Elastic scalability helps to ensure that applications are responsive while still minimizing running costs.

Development teams wanting to upgrade to the latest Azure 1.7 SDK can do so on a developer by developer basis.

For detailed information on the architecture of applications that run by using Windows Azure, visit the Introducing Windows Azure page at <http://www.windowsazure.com/en-us/develop/net/fundamentals/intro-to-windows-azure/>.



## Azure Roles

An application deployed to Windows Azure runs as one or more roles. A role is simply a collection of application files and configuration information that is deployed as an application instance. A Windows Azure solution contains one or more roles, and each role can be configured and managed independently. Windows Azure supports several different types of roles, and each type has a specific purpose.

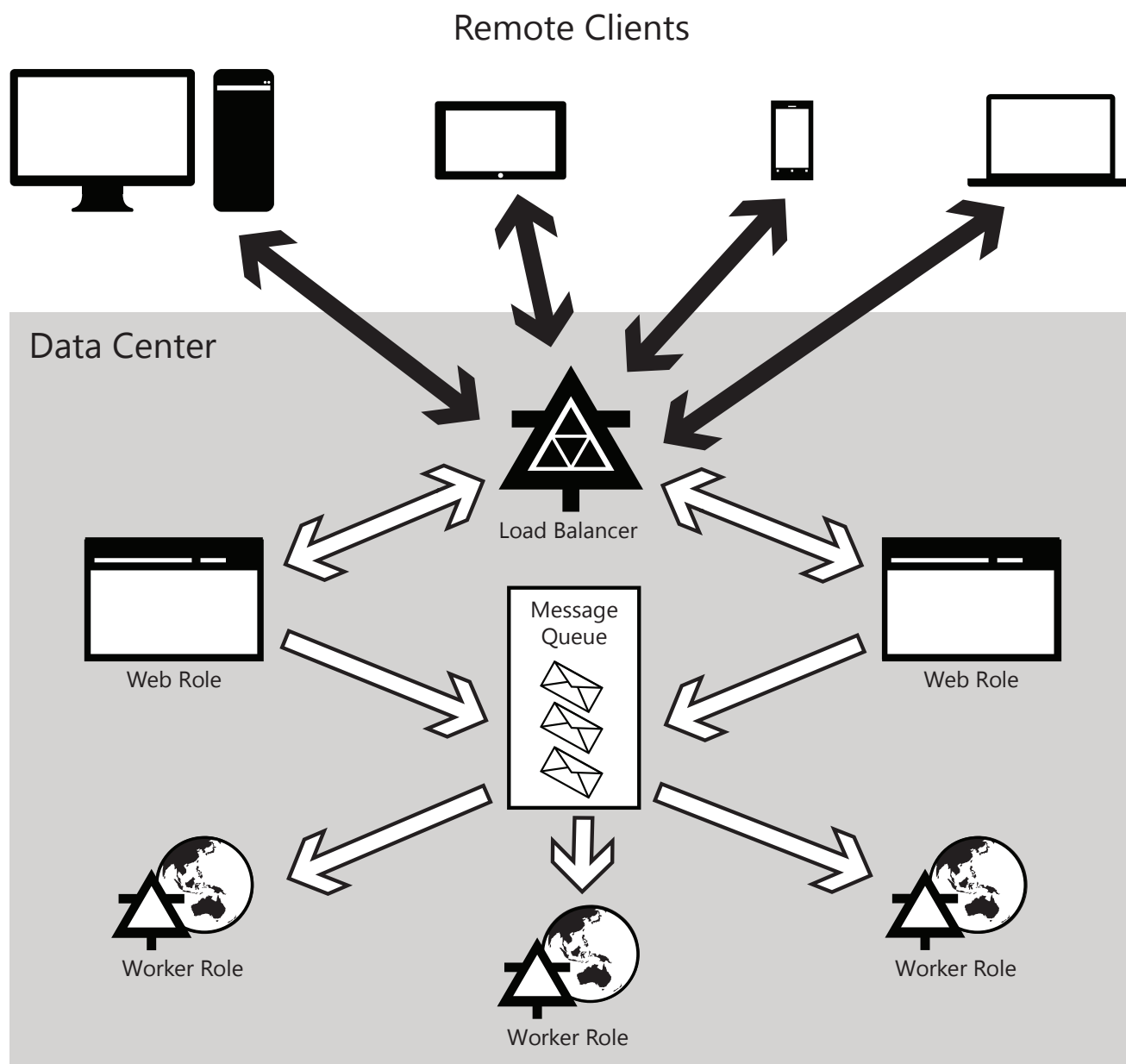
## Web Role

A Web role contains a web application or web service. Typically, a developer uses a Web role to build the user interface for an application running in the cloud. Web role instances in Windows Azure provide a dedicated Internet Information Services (IIS) web server, making it easy to create applications by using popular web technologies such as ASP.NET (including ASP.NET MVC), Windows Communication Foundation (WCF), PHP, and Node.js.

## Worker Role

A Worker role is a general purpose container for running custom code for a web application or service. Worker role instances typically run asynchronous, long-running, or perpetual tasks independent of user interaction or input. Most Azure solutions separate the background processing of an application into a Worker role and run the front-end in a Web role to better distribute application logic and improve control over how the application scales. Azure provides communication mechanisms in the form of queues and structured storage, controlled by the Azure Storage Service to enable Web and Worker roles to share data. By default, the code in a Worker role will execute by using IIS as the application server, but Worker roles can be built by using other application platforms, including Apache™ Tomcat® and the Java Virtual Machine (JVM).

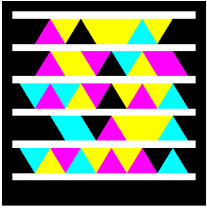
Windows Azure instances in Web and Worker roles are automatically maintained and managed by Microsoft staff with no involvement required by the developer or IT department.



### Cache Role

Performance is important for many applications. One way to improve the performance of an application is to store application data in a high-speed, in-memory cache. Windows Azure Caching (Preview) supports the ability to host caching services on Windows Azure roles. There are

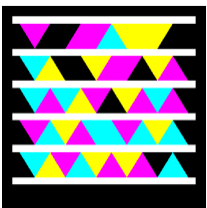
two main deployment topologies for caching (preview): co-located and dedicated. Co-located roles also host other non-caching application code and services. Dedicated roles are only used for caching.



Visual Studio 2012 enables you to configure Windows Azure Caching (Preview) for your cloud service. For more information on Windows Azure Caching (Preview), see the Overview of Windows Azure Caching (Preview) page at <http://msdn.microsoft.com/en-us/library/windowsazure/hh914161.aspx>.

## Virtual Machine Role

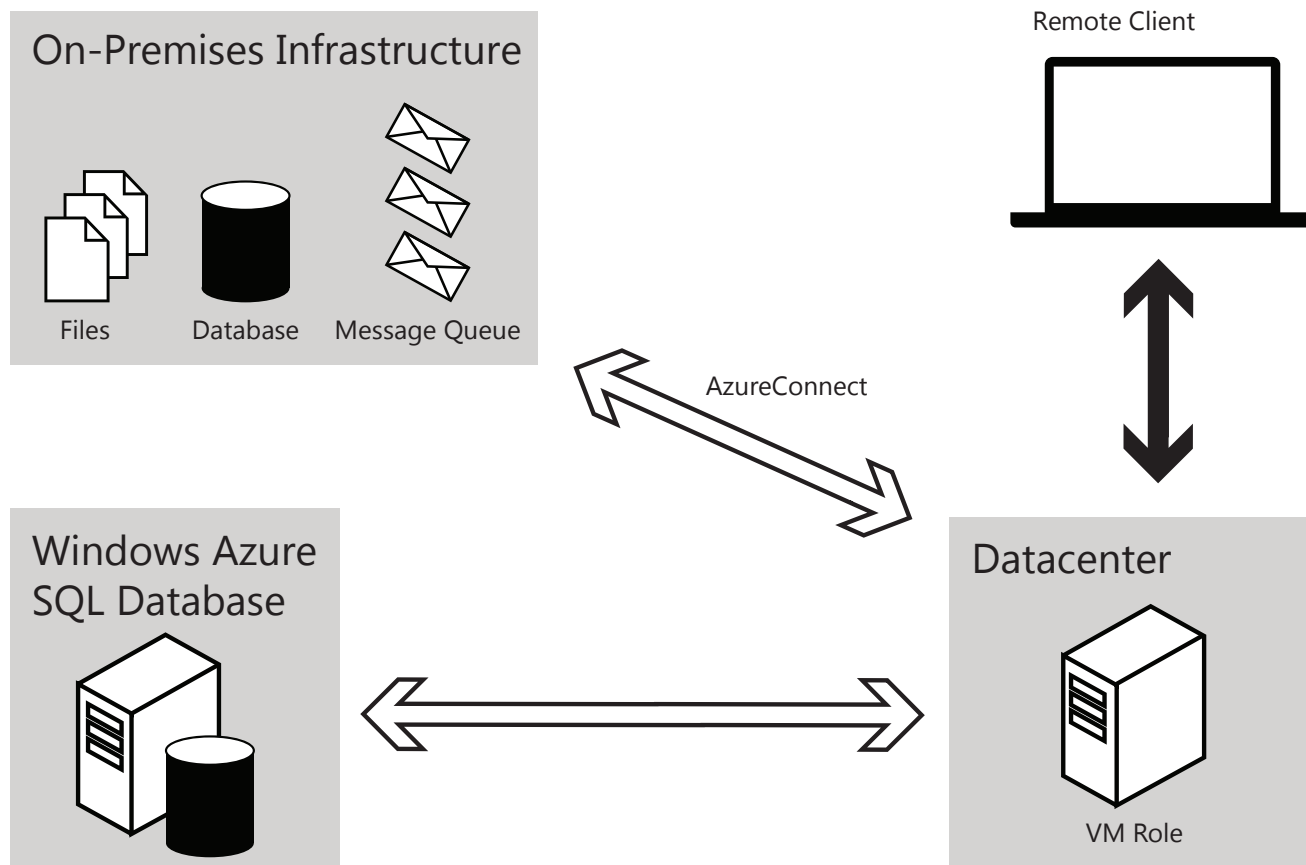
Not all applications are suitable for structuring as a Web or Worker role. For example, an organization may have an existing application that they simply wish to run in the cloud, or that may require significant customization of the machine on which it is running, possibly including the installation of a variety of dependent software components. The Virtual Machine (VM) role is designed for such scenarios. A VM role is a custom Windows Server 2008 R2 image running in a datacenter. Using a VM role, a developer can prepare a custom virtual machine image containing the applications and any dependent software, and then upload this virtual machine image to an Azure datacenter.



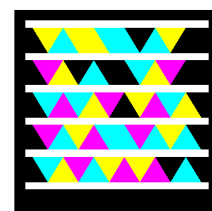
A VM role is ideal for hosting stateless compute-intensive applications. Code running in a VM role can store and access resources such as files, databases and message queues held on-premises within an organization by using Azure Connect to establish a secure virtual network connection. Azure Connect is described in more detail on the Overview of Windows Azure Connect page at <http://msdn.microsoft.com/en-us/library/windowsazure/gg432997.aspx>.



Alternatively, it can employ SQL Database if it needs to store and maintain structured information in a database.

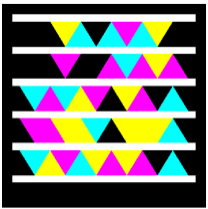


For more information about Azure roles, visit the Overview of Creating a Hosted Service for Windows Azure page at <http://msdn.microsoft.com/en-us/library/windowsazure/gg432976.aspx>.



## Azure Stateful Virtual Machines

VM roles are stateless, enabling them to be stopped and restarted automatically by the datacenter in which they are running without losing data. However, Azure now also supports completely customizable, stateful virtual machines. A developer has full control over the configuration and installation of the software, but application state and other data can be maintained in a persistent manner on the virtual machine. The developer's organization takes on the responsibility for monitoring the health of a virtual machine deployed in this manner, and for maintaining the state information held by each virtual machine. For more detailed information on Windows Azure virtual machines, visit the Virtual Machines page at, <http://www.windowsazure.com/en-us/home/scenarios/virtual-machines/>

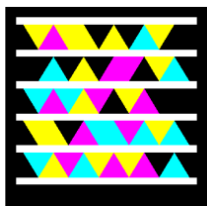


## Azure Storage

Each Windows Azure instance runs in a virtual machine located in a datacenter managed by Microsoft. However, these virtual machines are dynamic, and they can be stopped and restarted at any time. Consequently, any data stored in memory or on disk in the virtual machine is volatile. Windows Azure provides Azure Storage for holding non-transient data. This data can be held in a variety of formats; Azure provides table storage for holding structured data, blob storage that can contain large objects such as video and audio data, and queue storage for passing small messages between roles.

Azure Storage enables an application to store non-transient structured data and access it from any instance of a role. It is accessed through the Azure Storage Service provided by the Azure platform, and implements a secure, scalable, highly available and easily accessible storage space. The Azure Storage Service also supports NTFS emulation, enabling an application to use blob storage as a virtual disk drive in the cloud.

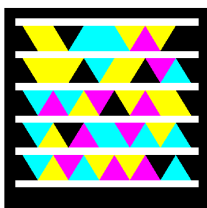
Visual Studio 2012 makes using Windows Azure Storage easy. Where previously it was necessary to manually create storage accounts by using the Windows Azure online portal, a developer can now complete this task completely within Visual Studio. This saves time and simplifies the workflow for developing a Windows Azure application.



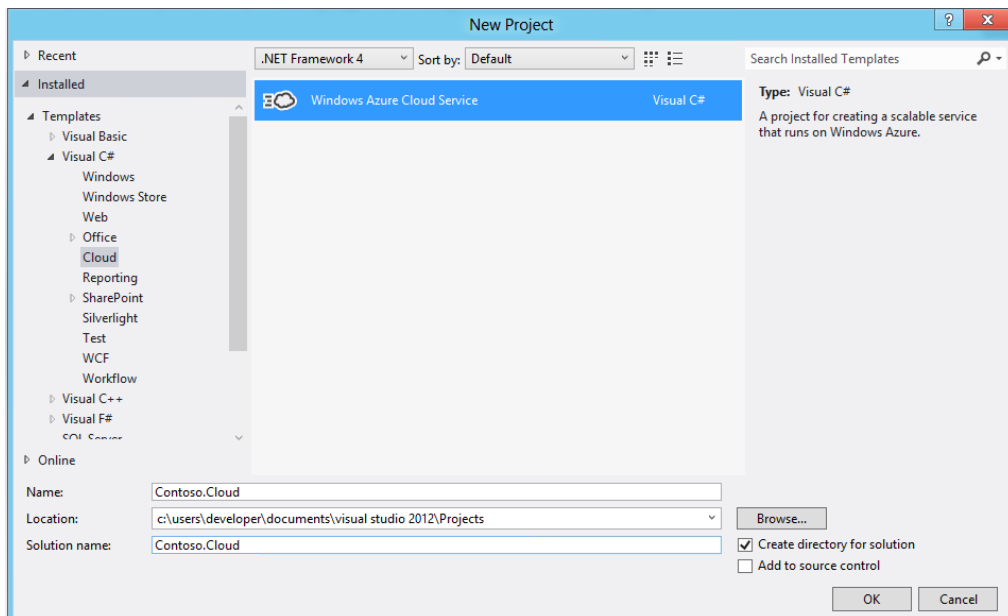
For more detailed information on Azure Storage, visit the Using the Windows Azure Storage Services page at <http://msdn.microsoft.com/en-us/library/windowsazure/ee924681>.

## Coding for Azure

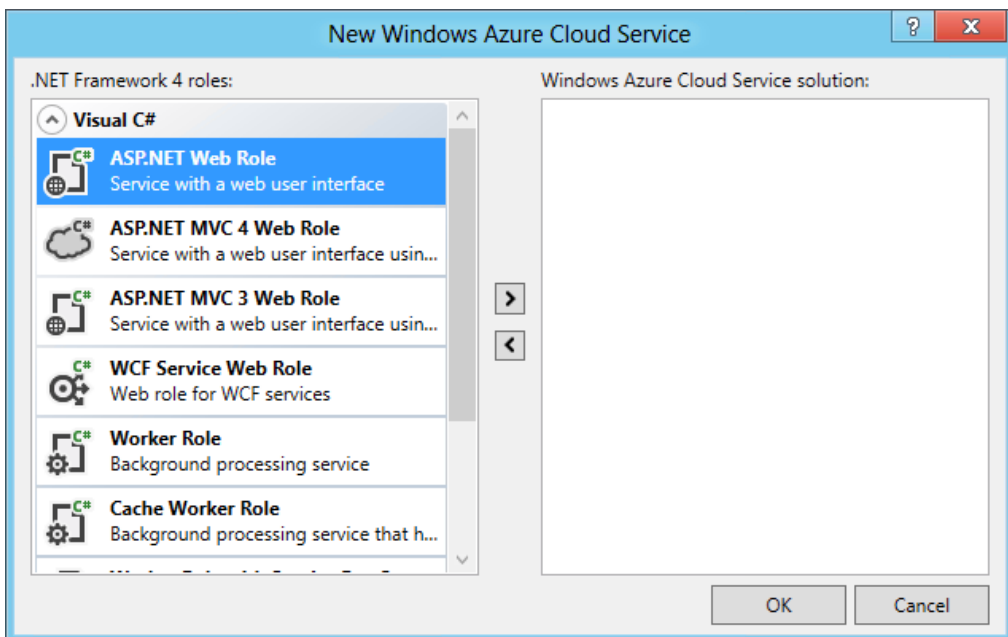
Developers building applications for Windows Azure use the Windows Azure SDK for .NET. This SDK integrates into the Visual Studio 2012 IDE, and includes a collection of templates, tools, and other utilities that enable a developer using Visual Studio 2012 to implement the code for Web and Worker roles. A developer can specify default configuration settings and test and debug the application locally by using the Azure Emulator. The emulator simulates the way in which Web and Worker roles run in the cloud, and enable a developer to take full advantage of the comprehensive debugging features of Visual Studio 2012. In fact, building and testing a Web or Worker role is very similar to creating an ordinary web application, and developers can leverage their existing skills to develop applications for Windows Azure. For more information about the Azure SDK and the features that it includes for use with Visual Studio 2012, visit the Windows Azure Tools for Microsoft Visual Studio page at <http://msdn.microsoft.com/en-us/library/windowsazure/ee405484.aspx>.



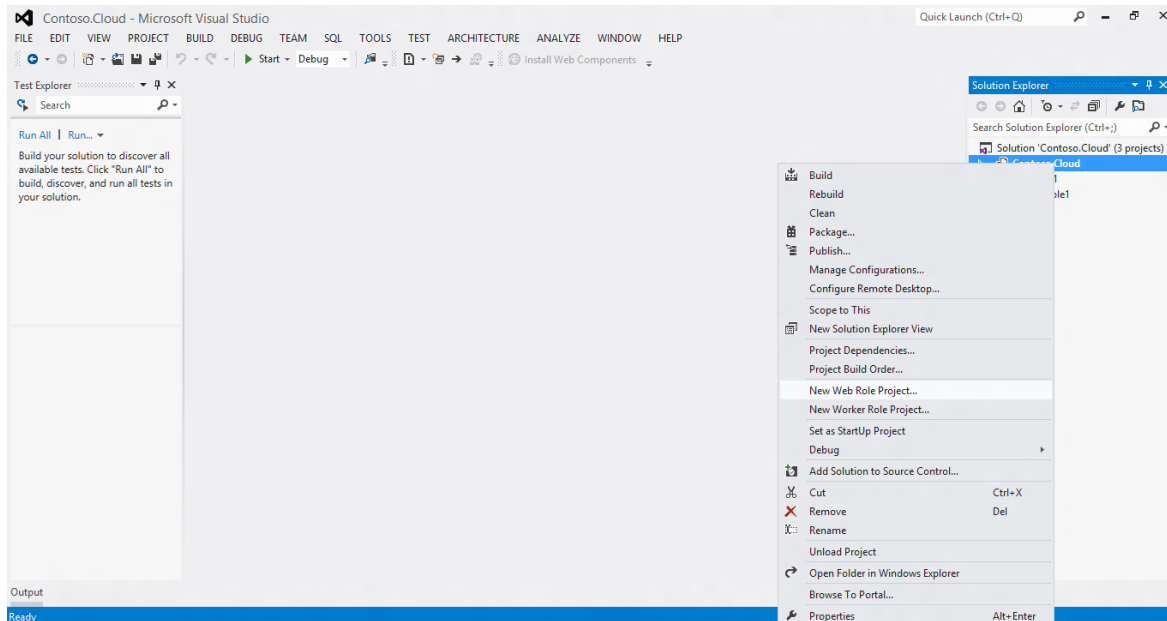
A developer creates a new cloud application by selecting the Windows Azure Project template:



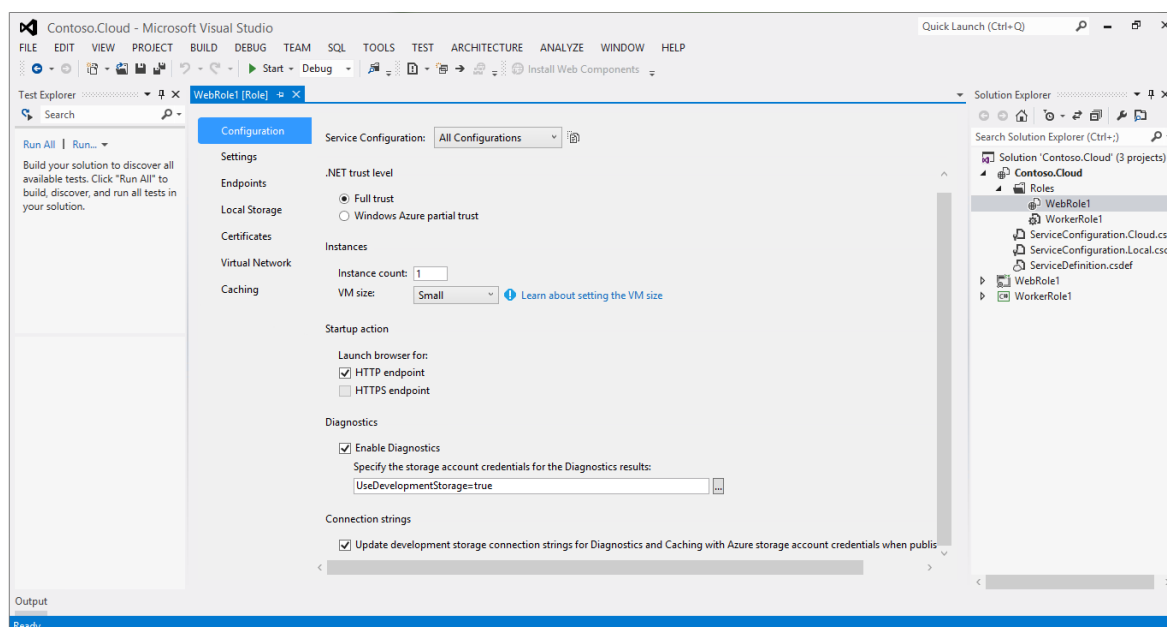
This template runs a wizard that enables the developer to select the roles to be implemented by the cloud application. Notice that Web roles can be constructed by using a variety of technologies. It is also possible to implement multiple Web and Worker roles in the same cloud application and these roles can be developed by using different programming languages.



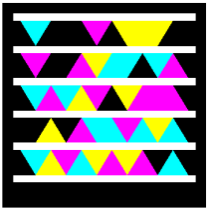
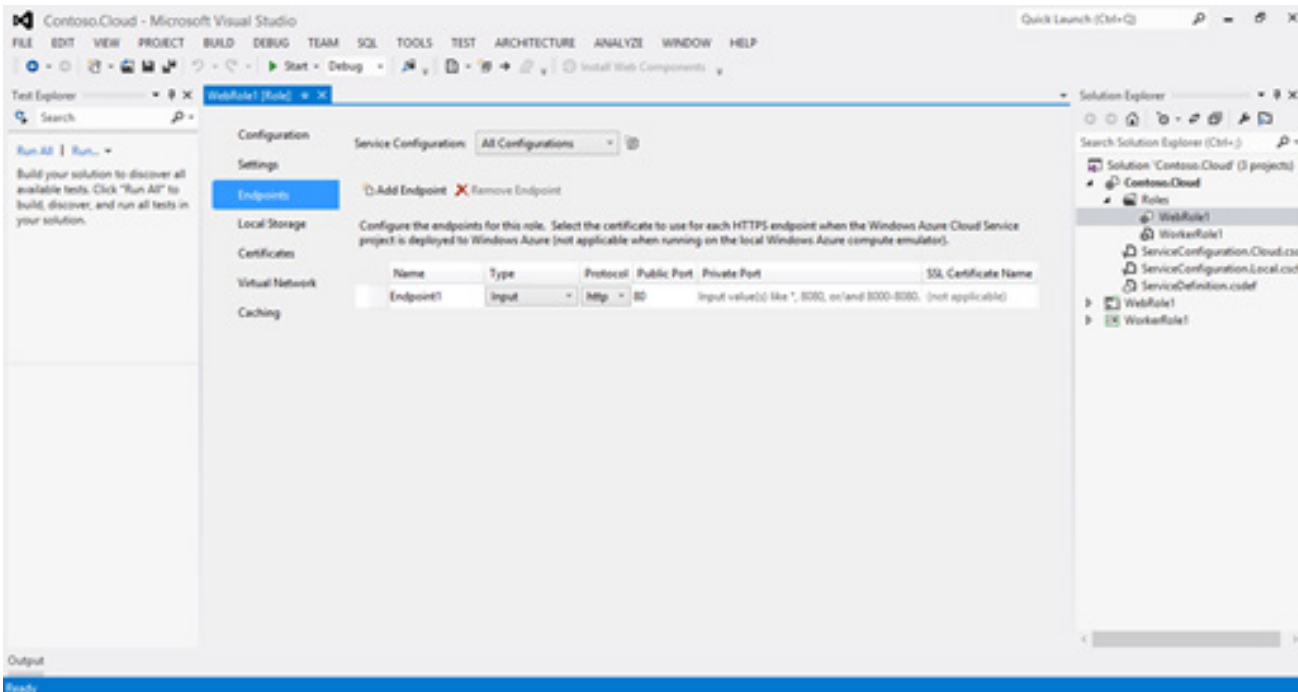
A cloud solution contains separate projects for each role, together with a project that defines the configuration of the solution. The configuration project provides wizards that enable the developer to add further roles to the project.



The Roles folder enables a developer to quickly configure each role in a cloud solution, specifying the volume of resources that the role has access to when it is deployed, the number of instances of the role to deploy, and the account keys to use for accessing Azure Storage.

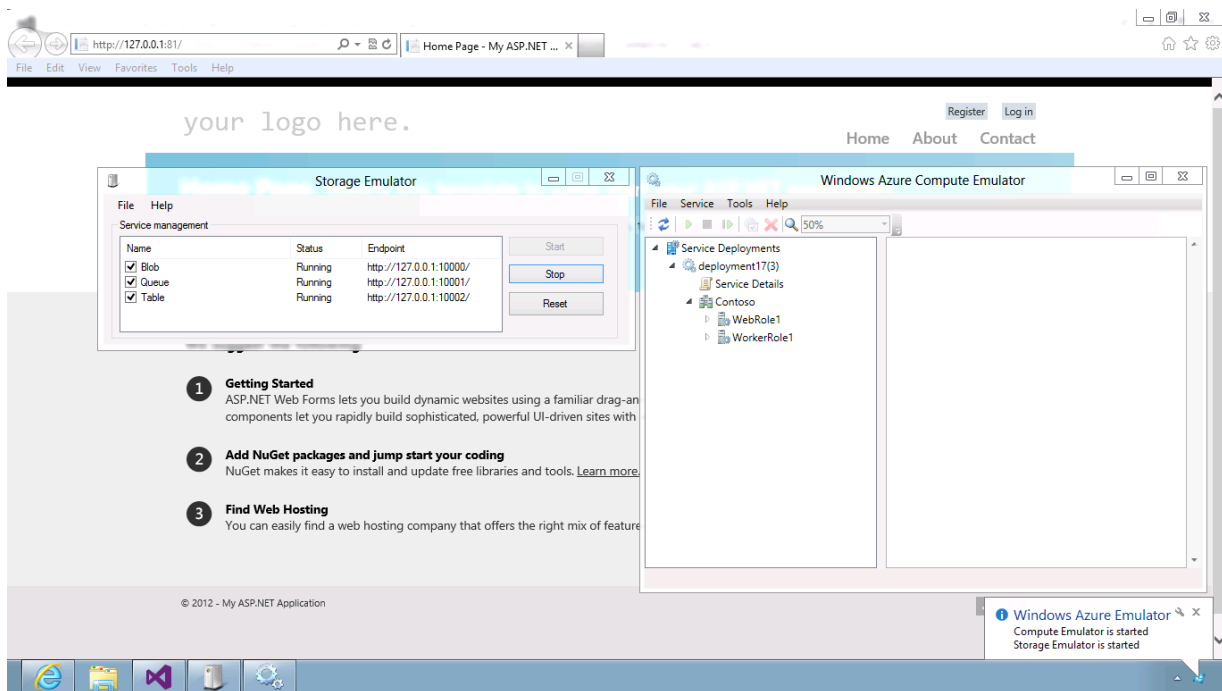


The Configuration screen also enables a developer to specify items such as role-settings, and the URLs of endpoints that the cloud application should expose.

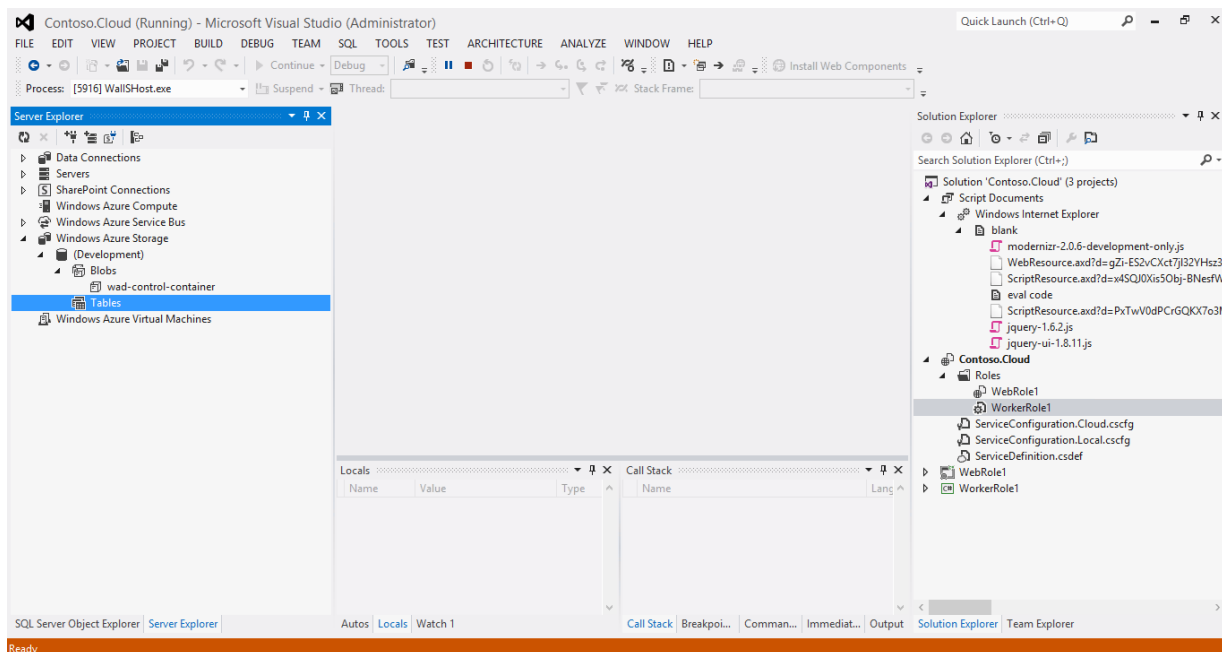


For further information about using Visual Studio 2012 to configure cloud application, visit the Configuring a Windows Azure Project page at <http://msdn.microsoft.com/en-us/library/windowsazure/ee405486.aspx>.

For development and testing purposes, a developer can deploy and debug roles locally on the development machine. The Windows Azure SDK includes a collection of tools and services that can emulate the cloud environment, including simulated Web and Worker roles, and a local service that mimics the features of Azure Storage. A developer can use this environment for fast and easy, feature testing and then hand the project over to an administrator when testing is complete for deployment to the cloud.

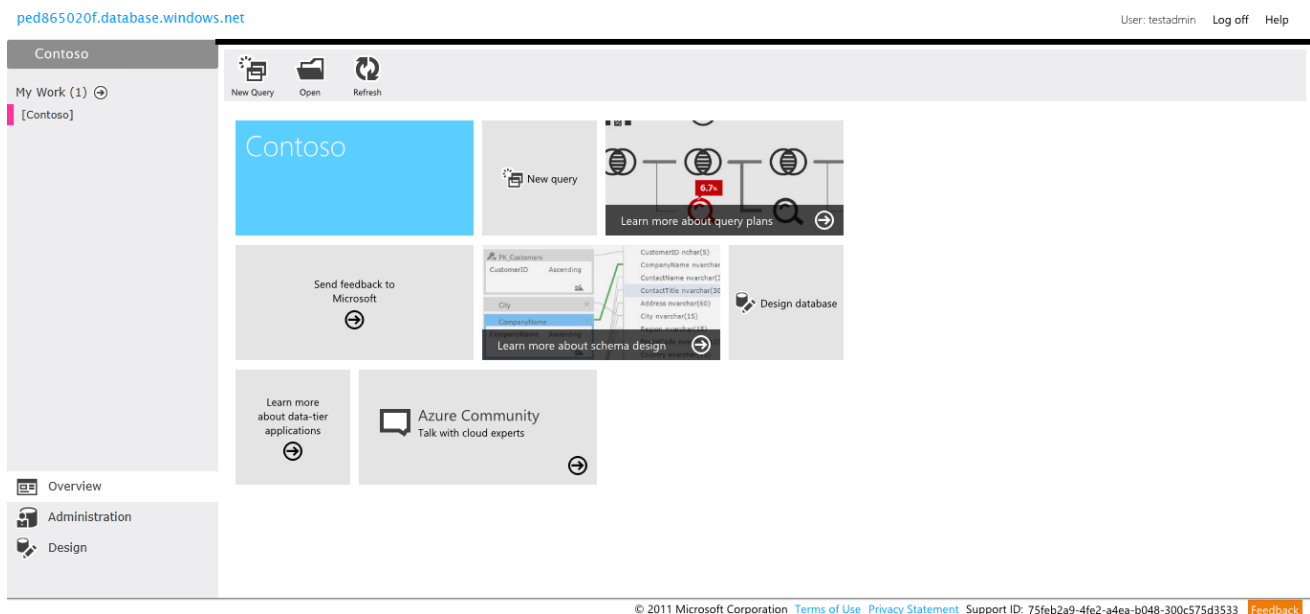


The Server Explorer in Visual Studio 2012 also provides access to local Azure Storage, as well as the Azure Storage Service in the cloud if the developer has the appropriate access keys. A developer can use this utility to quickly examine the contents of Azure Storage and verify that code running in the Web and Worker roles is storing the expected information.



## Windows Azure SQL Database

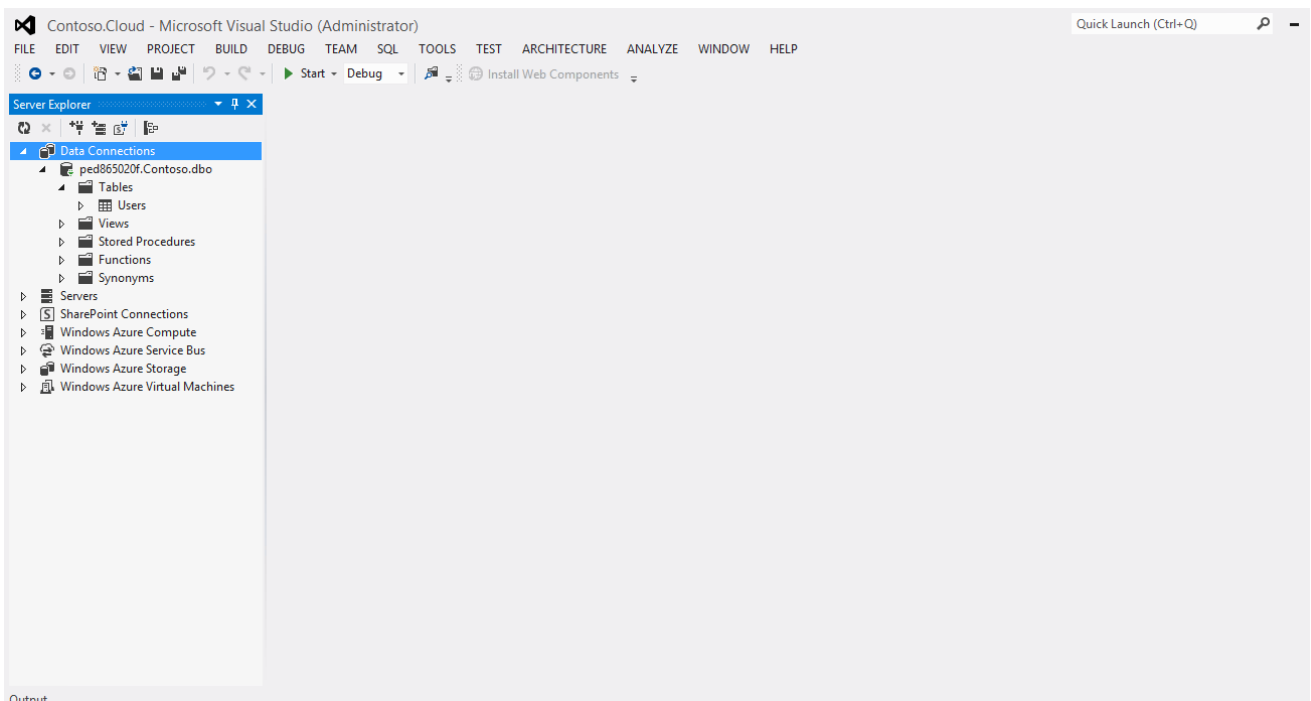
Many applications use databases to store related data. Windows Azure SQL Database is the cloud-based database that forms part of the Windows Azure platform. An administrator can easily provision SQL Database databases from the Windows Azure web portal. SQL Database is fully scalable, is reliable, and is familiar to developers who have used SQL Server. For development purposes, Visual Studio 2012 includes Local DB, a lightweight database engine that makes it easier to build data-driven applications. Local DB has full fidelity to SQL Database databases. Developers can create an application with confidence that it works correctly when they deploy the application to the Windows Azure platform.



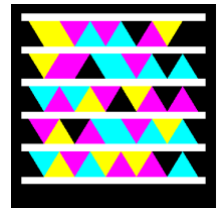
Once the application logic for a cloud application has been completed and tested locally, and administrator can reconfigure the application to connect to the live SQL Database in the cloud.



Using Server Explorer in Visual Studio 2012, a developer with the appropriate access rights can connect directly to SQL Azure in the cloud to view data.



For more information about Windows Azure SQL Database, visit the Introducing Windows Azure SQL Database page at <http://msdn.microsoft.com/en-us/library/windowsazure/ee336230.aspx>.



## Publishing to the Cloud

An administrator packaging and deploying an Azure application can use the same Configuration wizard available to a developer. An administrator can modify any of the configuration settings, and specify the necessary access keys and endpoints for the solution, and also configure features such as Remote Desktop, which enables an administrator to connect directly to the

virtual machines running the various roles in the cloud. An administrator can also enable and configure IntelliTrace, enabling support staff to debug issues that may occur while the application is running in the cloud.

**Publish Windows Azure Application**

**Windows Azure Publish Settings**

Sign in  
**Settings**  
 Summary

Common Settings | Advanced Settings

Deployment label:  
 Contoso.Cloud

☒ Append current date and time

Storage account:  
 ContosoService

☐ Delete deployment on failure

☒ Deployment update [Settings...](#)

☒ Enable IntelliTrace [Settings...](#)

☐ Enable profiling [Settings...](#)

[Online privacy statement](#)

< Previous   Next >   Publish   Cancel

**Package Windows Azure Application**

Service configuration: Cloud

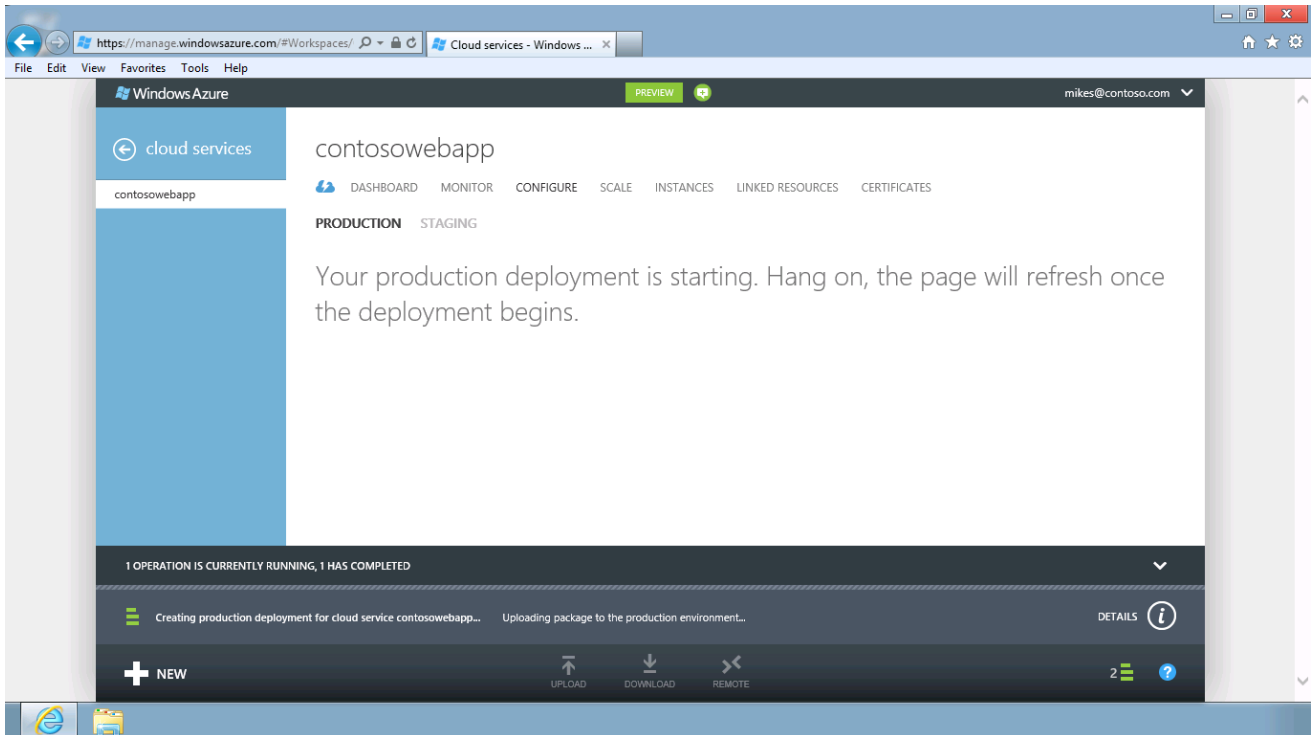
Build configuration: Debug

☐ Enable Remote Desktop for all roles [Settings...](#)

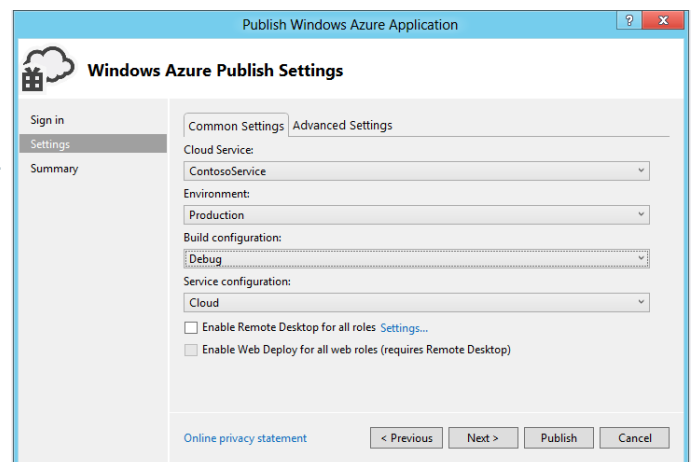
Package   Cancel

The Configuration wizard converts the various settings into a set of XML files that define the configuration for each service implemented by the Web and Worker roles, together with the endpoint and connectivity configuration for each service. The Package wizard enables an administrator to build an installable package for the solution, including the executable code and the configuration information. An administrator can also enable and configure a remote desktop connection for the solution, enabling direct access to each role instance when it is running. This is useful for monitoring purposes.

After the package has been generated, the administrator can log into the Azure account for the organization and upload the package to service in the cloud by using the Azure Management Portal.



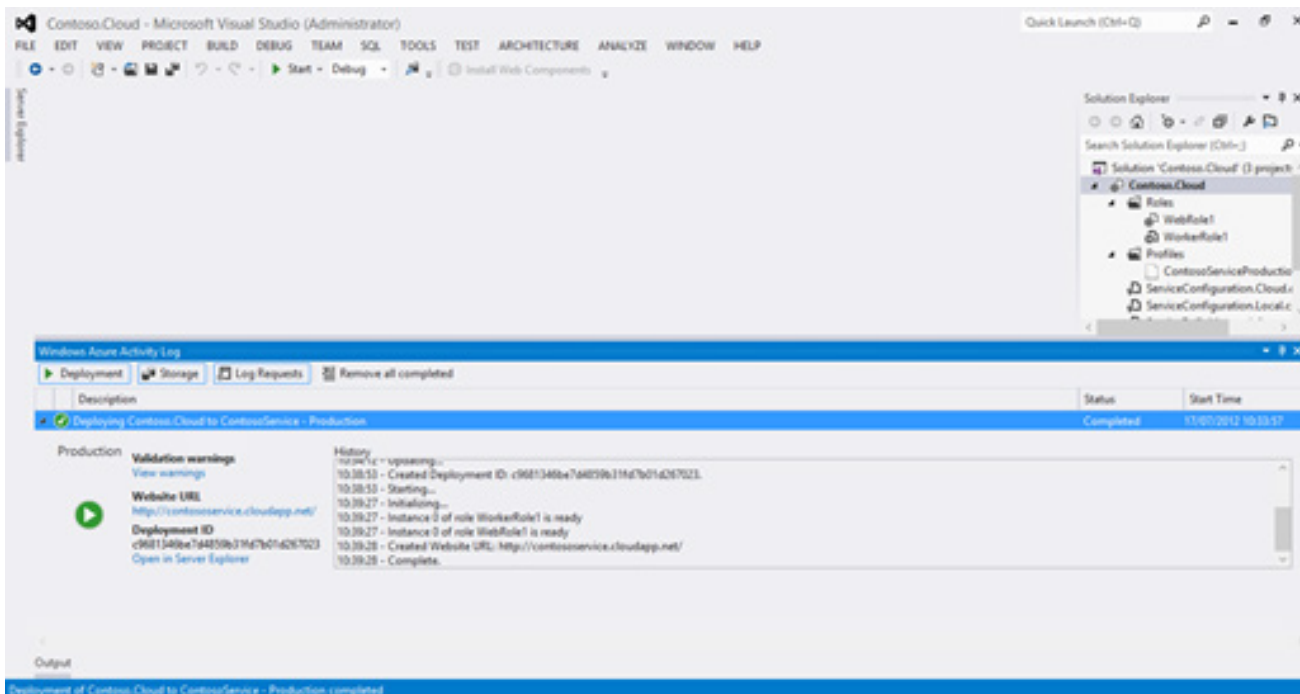
The Publish Windows Azure Application wizard combines the packaging and deployment phases, enabling an administrator to perform both tasks directly from Visual Studio 2012. This wizard prompts the user for the details of the Azure account and the cloud service to use, before building and deploying the application to this service.



Note that the account details can be saved locally, to prevent an administrator from having to enter them repeatedly when deploying or reconfiguring services.

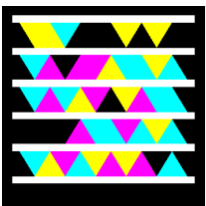
The wizard performs consistency checking to verify that the configuration is valid and to minimize the possibility of failure due to configuration errors when the application runs in the cloud.

As an application is published and deployed, the activity log in Visual Studio 2012 displays the deployment progress and the initialization status of virtual machines.



Most applications evolve as business requirements change or customer demand varies. Visual Studio 2012 makes updating applications easy with in-place updates. Updates can be made to code and to many configuration settings without taking the deployed solution offline.

To update the configuration settings, an administrator can simply make the necessary changes in the configuration files and use the publish wizard to re-publish the application. Visual Studio 2012 and the Windows Azure platform perform an in-place update automatically.

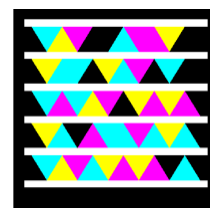
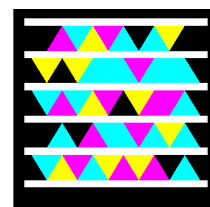
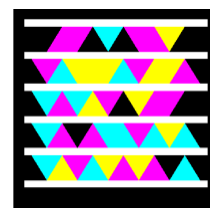


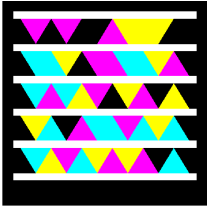
For more information about using Visual Studio to publish and deploy a cloud application, see the Publishing Windows Azure Applications to Windows Azure from Visual Studio page at <http://msdn.microsoft.com/en-us/library/windowsazure/ee460772.asp>

# Building Hybrid Cloud Solutions with Windows Azure

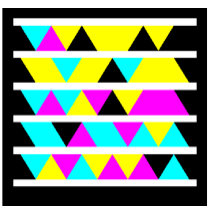
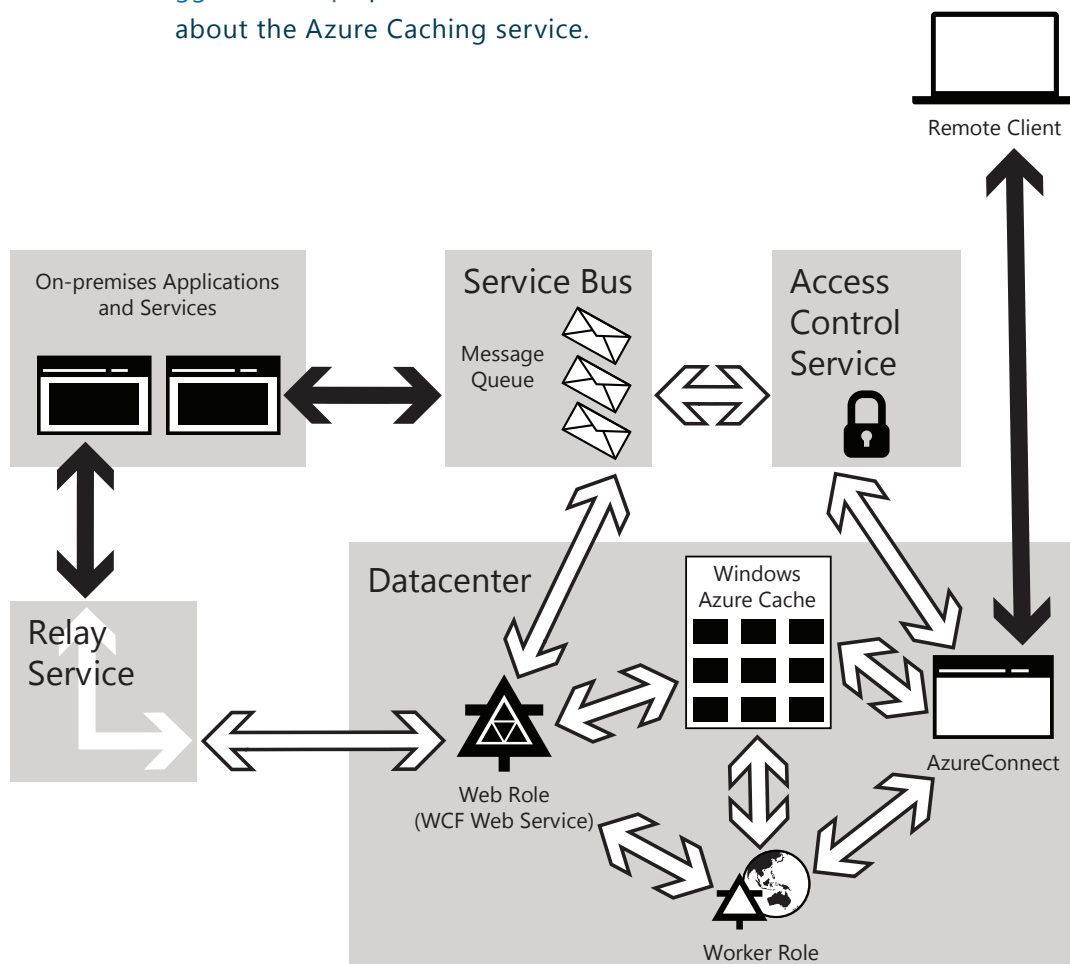
A hybrid cloud solution consists of a combination of application elements, some of which run on-premises within an organization, while other components are located in the cloud. Building hybrid cloud solutions is easy with Visual Studio 2012 and the Windows Azure SDK for .NET. Using the tools provided by Visual Studio and the Azure SDK, a developer can create new web applications and services, or adapt existing applications and services for the cloud. The key issues are those associated with connecting the various components in a reliable and highly secure fashion. Windows Azure provides a collection of technologies aimed at optimizing communications and data access in a cloud application. These technologies include:

- The Windows Azure Service Bus, which offers scalable cloud-based message queuing. More information about the Service Bus is available on the Messaging page at <http://www.windowsazure.com/en-us/home/features/messaging/>
- The Azure Relay Service, for protecting and optimizing connections between services running in the cloud or on an on-premises server and remote client applications. This is part of the Azure Service Bus designed specifically for supporting services hosted in the cloud. For more information about this feature, see the Windows Azure AppFabric: An Introduction to Service Bus Relay page at <http://www.microsoft.com/en-us/showcase/details.aspx?uuid=395930db-6622-4a9f-8152-e0cb1fc5149c>
- The Access Control Service which enables an organization to implement claims-based authentication, integrating the facilities of common authentication providers such as Windows Live, Google, and Facebook, into their own solutions, or to implement a federated security mechanism that incorporates the security services of partner organizations. For more information, visit the Access Control Service 2.0 page at <http://msdn.microsoft.com/en-us/library/windowsazure/gg429786.aspx>.





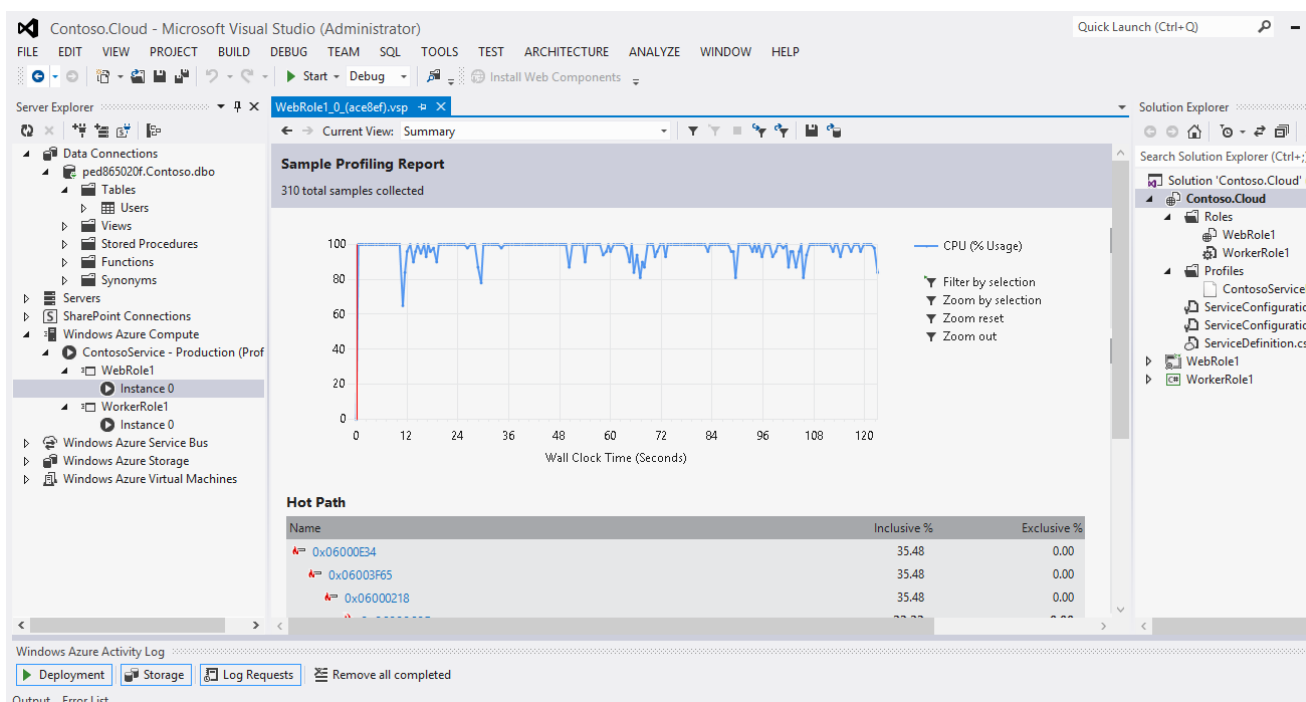
- In-role caching, which provides a reliable mechanism for enabling an application to retain frequently used data in-memory in the virtual machine in which it is running, reducing the network latency that commonly occurs when connecting to services running across the Internet. The Service Bus also provides a separate caching service that enables multiple role instances to cache shared data locally in the same datacenter. The Caching in Windows Azure page at <http://msdn.microsoft.com/en-us/library/windowsazure/gg278356.aspx> provides more information about the Azure Caching service.



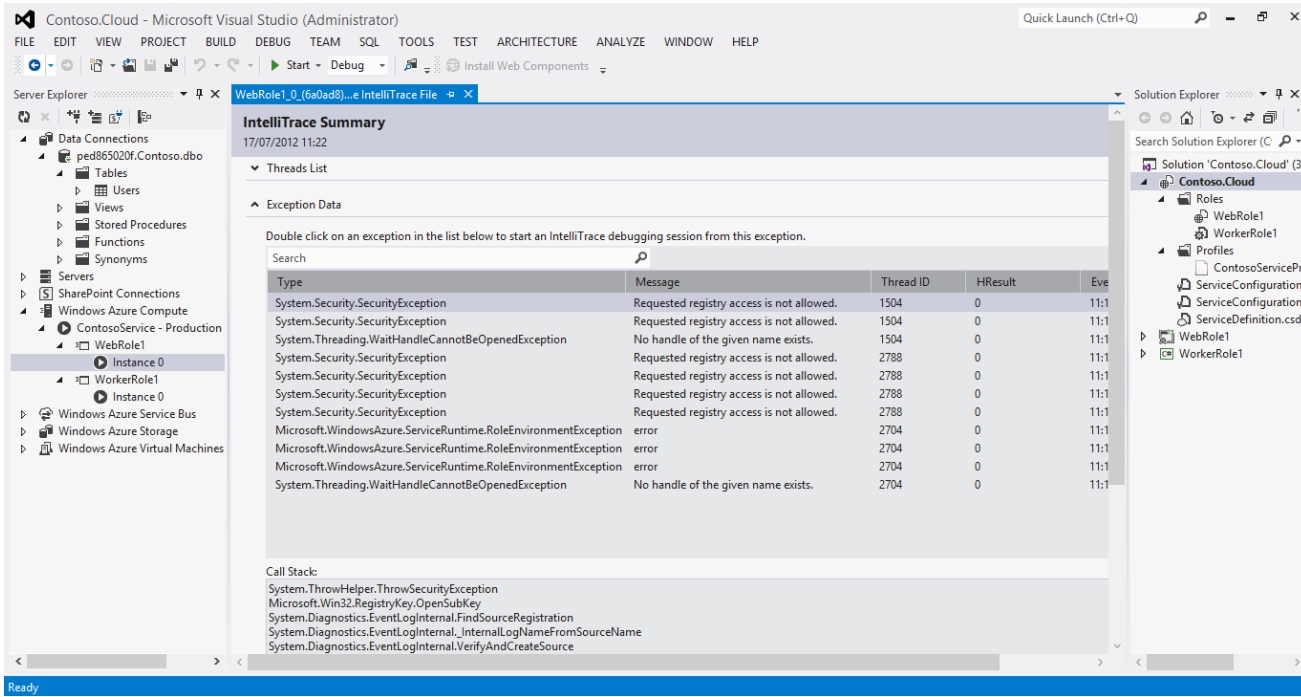
For more information about building hybrid cloud applications and the features that Service Bus provides, visit the Building Hybrid Solutions with Windows Azure page at <http://www.windowsazure.com/en-us/develop/net/fundamentals/hybrid-solutions/>.

# Profiling and Tracing Web Applications and Services in the Cloud

When building web applications for the cloud, or when integrating existing on-premises applications with cloud-based services, new challenges arise. Characteristics such as network latency and resource constraints can cause unexpected performance issues. Visual Studio 2012 includes profiling tools that developers can use to identify the possible causes of poor performance. An administrator or developer can enable profiling of selected parts of an application, and the Windows Azure platform can capture and store this profiling information. Developers can download this data and analyze it by using the profiling tools in Visual Studio 2012.

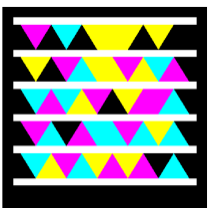


An administrator can configure IntelliTrace for cloud applications and services, enabling a cloud application to capture and record debugging information that can be downloaded and replayed inside Visual Studio 2012. Using IntelliTrace enables a developer to shorten the debugging cycle, and also to capture errors and trace events caused by circumstances that are difficult to reproduce locally.



## Enabling Continuous Delivery in the Cloud

Visual Studio Team Foundation Server running on-premises can integrate with Azure SDK tools and Microsoft PowerShell scripts to implement a tight feedback loop between the project team building cloud services and the customers that use them. A developer can modify code to add new features or fix bugs and check the code in to the Team Foundation Server (TFS) repository. TFS can execute a continuous build process to detect new versions of source files, rebuild the solution, and deliver a new version of the application for testing and deployment to the cloud.

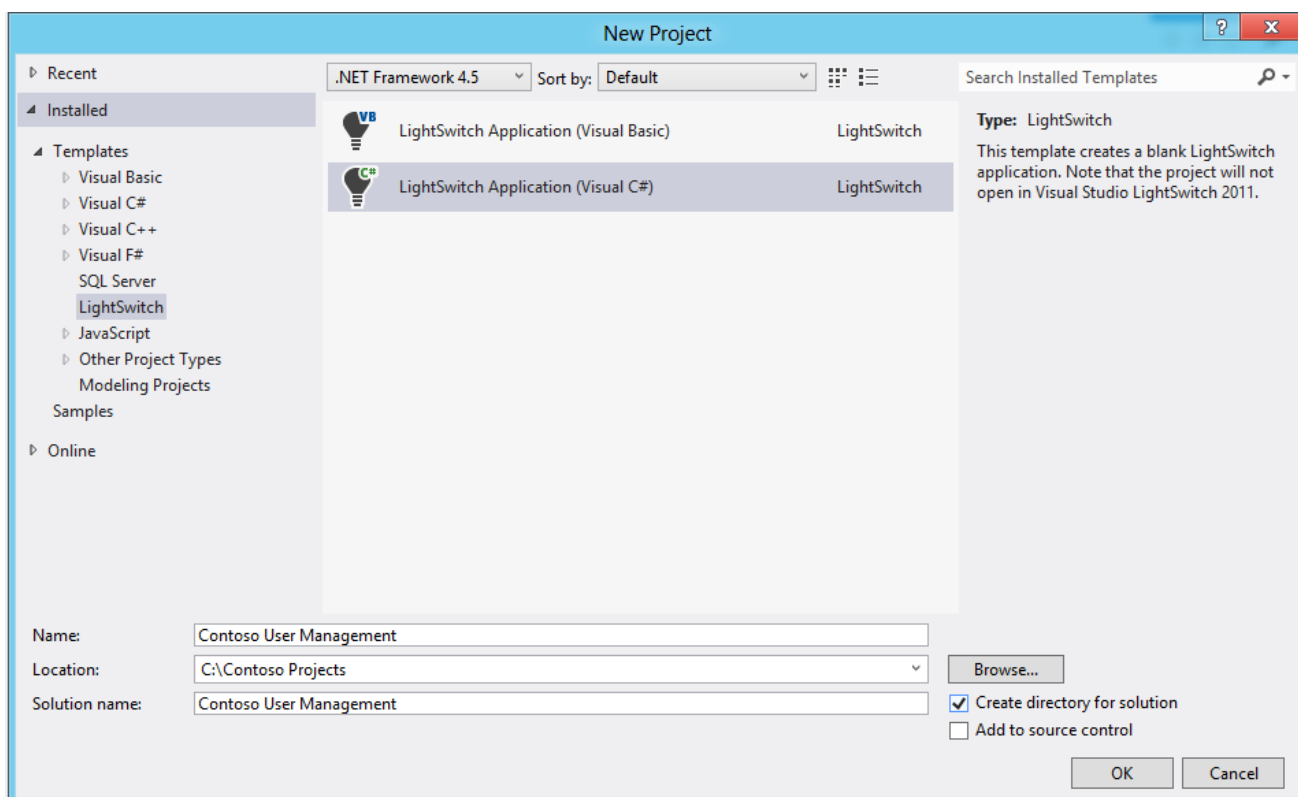


The same process also works with Visual Studio Team Foundation Service, enabling a project team to manage its operations in the cloud and deploy applications to the cloud. For more information about Team Foundation Service, visit: <http://tfs.visualstudio.com/>.

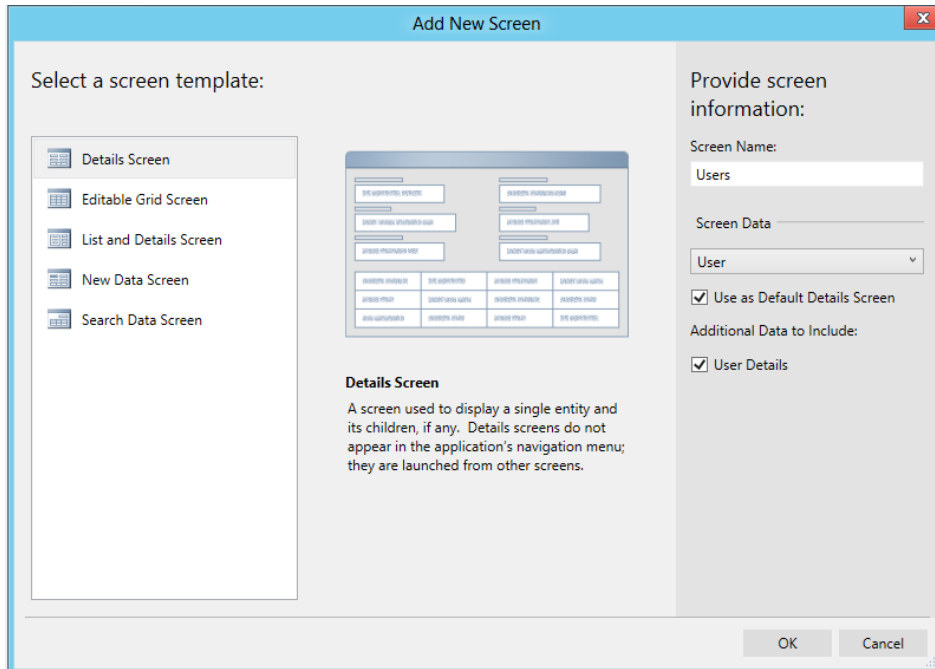


# Building and Deploying Applications to the Cloud by Using Visual Studio LightSwitch

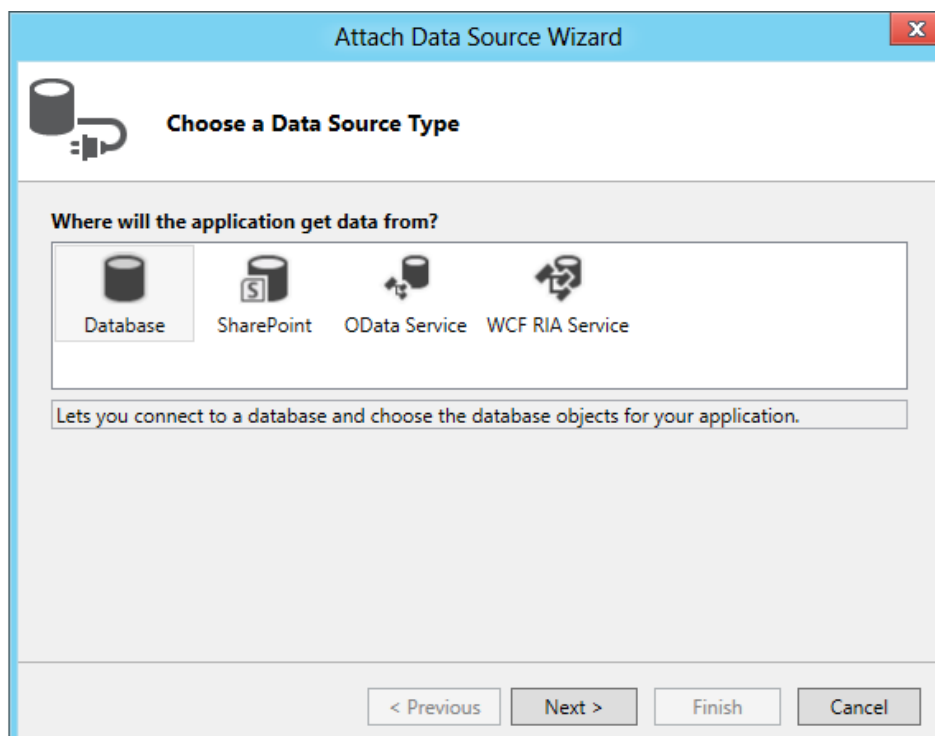
Visual Studio LightSwitch gives end-user developers a fast and simple toolset for building high-quality, data-driven, line-of-business applications. It is available as a separate, standalone product but the LightSwitch templates are also integrated into Visual Studio 2012.



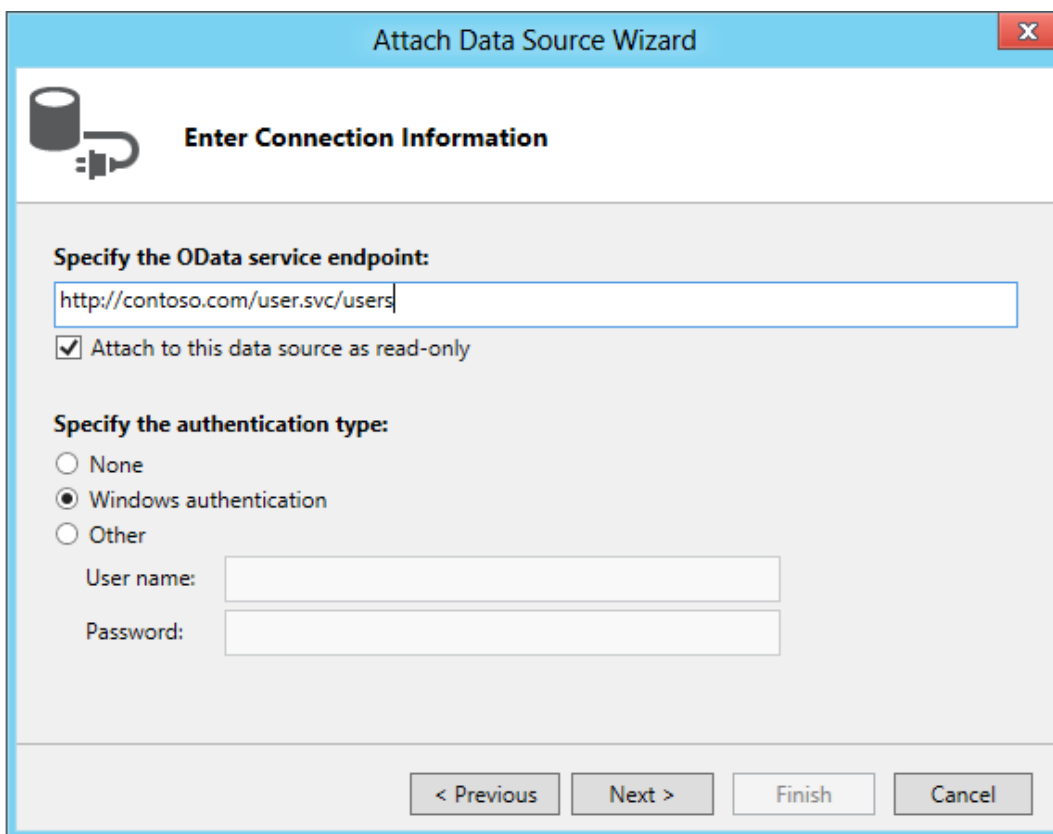
These templates enable a developer to be productive immediately by providing a collection of pre-configured screen templates that implement the common application logic required by most business applications.



These screens can be connected to a variety of data sources such as a database located locally or in the cloud, a Microsoft SharePoint site, an Open Data Protocol (OData) Service, or a WCF service.



Visual Studio 2012 provides Wizards for each of the data sources. For example, the OData Service wizard prompts the developer for the service endpoint to connect to and the credentials to use.



The image shows a screenshot of the 'Attach Data Source Wizard' dialog box in Visual Studio 2012. The dialog has a blue title bar with the text 'Attach Data Source Wizard' and a close button (X). Below the title bar is a header area with a database icon and the text 'Enter Connection Information'. The main content area is divided into two sections. The first section is titled 'Specify the OData service endpoint:' and contains a text box with the URL 'http://contoso.com/user.svc/users'. Below the text box is a checkbox labeled 'Attach to this data source as read-only' which is checked. The second section is titled 'Specify the authentication type:' and contains three radio buttons: 'None', 'Windows authentication' (which is selected), and 'Other'. Below the radio buttons are two text boxes labeled 'User name:' and 'Password:'. At the bottom of the dialog are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'.

Subsequent pages in the wizard enable the developer to select the entities to use from this endpoint. Visual Studio 2012 generates code that connects to this endpoint and retrieves the data, returning it as a set of custom types. A developer can customize the data source and include additional code and queries to modify the way in which the data is presented and updated.

The built in functionality generated for a screen includes default logic for adding new data items to a data source, searching through data, and displaying the details of data items in list or grid form. Screens can be linked to enable a user to drill down through data, displaying information related to selected items. A developer can easily customize screens and the navigation paths between them to meet the exact requirements of the application.

```

LightSwitchApplication.ApplicationDataService Users_Validate(User entity, EntitySetValidationResi
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Microsoft.LightSwitch;
using Microsoft.LightSwitch.Security.Server;

namespace LightSwitchApplication
{
    public partial class ApplicationDataService
    {
        partial void Users_Validate(User entity, EntitySetValidationResultsBuilder results)
        {
            if (this.Users.Where(u => u.UserName == entity.UserName).FirstOrDefault() != null)
            {
                results.AddPropertyError(
                    "The specified username has already been used, please specify a unique username..!!",
                    entity.Details.Properties.UserName);
            }
        }
    }
}

```

The extensibility features of LightSwitch enable developers to build and apply custom shells and themes to applications, enabling them to blend in with the look and feel of existing corporate applications. With Visual Studio 2012, LightSwitch includes the new Cosmopolitan shell and theme, which provide a more immersive feel, simple and clean control styling, inclusion of branding, and better use of screen space.

Contoso User Management

MENU

Search Users

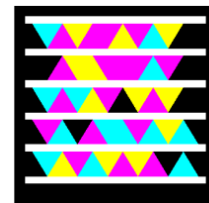
Users

USER NAME	FIRST NAME	SURNAME	EMAIL	AGE
GrigoryP	Grigory	Pogulsky	GrigoryP@Contoso.com	54
JamesA	James	Alvord	JamesA@Contoso.com	19
▶ SanjayJ	Sanjay	Jacob	SanjayJ@Contoso.com	34

1

Save Refresh Design Screen

A developer can also modify the code behind any screen to include the appropriate business logic to be performed when the user queries or edits data. For more information about building applications with Visual Studio LightSwitch, visit the Quickly Build Business Applications page at <http://www.microsoft.com/visualstudio/en-us/lightswitch/overview/build-quickly>.



A LightSwitch application can be deployed as a local desktop application on each user's machine, but a more common scenario is to publish the application to a web server and enable users to connect to it from a browser. The application can be run by using an instance of IIS located within the organization, or it can be published to Azure. Also, users have the option to only deploy the OData service created by LightSwitch, and not the client application.

The LightSwitch Publish Application wizard enables an administrator to configure the application and to upload it directly to Azure.

**LightSwitch Publish Application Wizard**

**Application Services**

Application Type  
Application Server Configuration  
Subscription  
Service Configuration  
Security Settings  
Data Connections  
Summary

**Where will the application's services be hosted?**

☐ Local  
Run application services on the end user's machine.

☐ IIS Server  
Host application services on an Internet Information Services (IIS) Server. A separate server is required.  
[Import Settings...](#)

☒ Windows Azure  
Host application services using Windows Azure.  
[Learn more about hosting using Windows Azure](#)

< Previous   Next >   Publish   Cancel

A LightSwitch application actually comprises two principal areas of functionality: data sources and screens. When an administrator publishes a LightSwitch application the data sources are deployed as data services, while the screens are packaged within a Microsoft Silverlight component that can be run in a web browser. With Visual Studio 2012, an administrator can elect to perform service-only publication, in which case only the data sources are packaged and deployed as an OData feed, with separate endpoints for each data source. Client applications built by using other languages and technologies can connect to this feed and consume the data from each source.

The screenshot shows the 'LightSwitch Publish Application Wizard' dialog box. The title bar includes a close button (X). The main content area is titled 'Welcome to the LightSwitch Publish Wizard'. On the left, a vertical list of steps is shown: 'Application Type', 'Application Server Configuration', 'Subscription', 'Service Configuration', 'Security Settings', 'Data Connections', and 'Summary'. The 'Application Type' step is currently selected. The main area contains the following text: 'This wizard helps you publish a LightSwitch application and make decisions about how it will be installed.' followed by the question 'What kind of application do you want to publish?'. There are three radio button options: 'Desktop' (with a description: 'This application runs on the user's desktop and can access Microsoft Office and other programs on the user's computer.'), 'Web' (with a description: 'This application runs in the user's browser and will not be able to access other programs on the user's computer.'), and 'Service only' (which is selected, with a description: 'This application will run only as a web service and no client will be published.'). Below these options is a warning icon (yellow triangle with an exclamation mark) and a message: 'The active configuration is set to Debug so you may be publishing debugging utilities with your application. To avoid publishing debugging utilities, exit the wizard, go to the configuration manager and change the active configuration to the Release configuration.' At the bottom right, there are four buttons: '< Previous', 'Next >', 'Publish', and 'Cancel'.

LightSwitch Publish Application Wizard

**Welcome to the LightSwitch Publish Wizard**

Application Type  
Application Server Configuration  
Subscription  
Service Configuration  
Security Settings  
Data Connections  
Summary


This wizard helps you publish a LightSwitch application and make decisions about how it will be installed.

**What kind of application do you want to publish?**

☐ Desktop  
This application runs on the user's desktop and can access Microsoft Office and other programs on the user's computer.

☐ Web  
This application runs in the user's browser and will not be able to access other programs on the user's computer.

☒ Service only  
This application will run only as a web service and no client will be published.

 The active configuration is set to Debug so you may be publishing debugging utilities with your application. To avoid publishing debugging utilities, exit the wizard, go to the configuration manager and change the active configuration to the Release configuration.

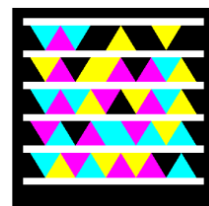
< Previous   Next >   Publish   Cancel

## Summary

Windows Azure provides a comprehensive platform for hosting applications in the cloud. Windows Azure is secure, scalable, and reliable making it the perfect platform to host applications without the added effort and cost of managing servers. An organization can use Visual Studio 2012 to easily create, publish, manage, and update applications in Windows Azure while minimizing the downtime of these applications. Visual Studio Team Foundation Server provides extended facilities for project teams, enabling them to respond rapidly to user requests by implementing a continuous delivery process.

By using Visual Studio 2012 with the Windows Azure SDK for .NET, developers can build, test, and debug cloud applications by using a local emulator. This strategy helps to identify any application-specific errors that may be present, before deploying the application to the cloud. Additionally, the monitoring and logging capabilities of Azure, together with the IntelliTrace features of Visual Studio 2012, enable administrators and developers to capture performance and other trace data, and to analyze this information to identify the causes of any bottlenecks in the system. The Windows Azure page at <http://msdn.microsoft.com/en-us/library/dd163896.aspx> provides more information on using Azure to design and implement cloud-based applications and services.

Visual Studio LightSwitch provides a fast and convenient way to build and deploy data-driven line-of-business applications. These applications can also be deployed and tested locally, while the deployment wizard in LightSwitch enables an administrator to quickly package an application and upload it to the cloud.





# Developing for SharePoint

Collaboration is a key factor in the success of effective organizations, whether this collaboration is between individual employees working as part of a small team, between departments needing to liaise and coordinate with each other, or between separate organizations acting as partners to achieve a common goal. In this environment, the ability to share information quickly and more securely is paramount. Microsoft Windows SharePoint helps you to build solutions that can accomplish these aims. By using Windows SharePoint, you can build custom web sites for your organization that help you share information and cooperate with your colleagues and business partners.

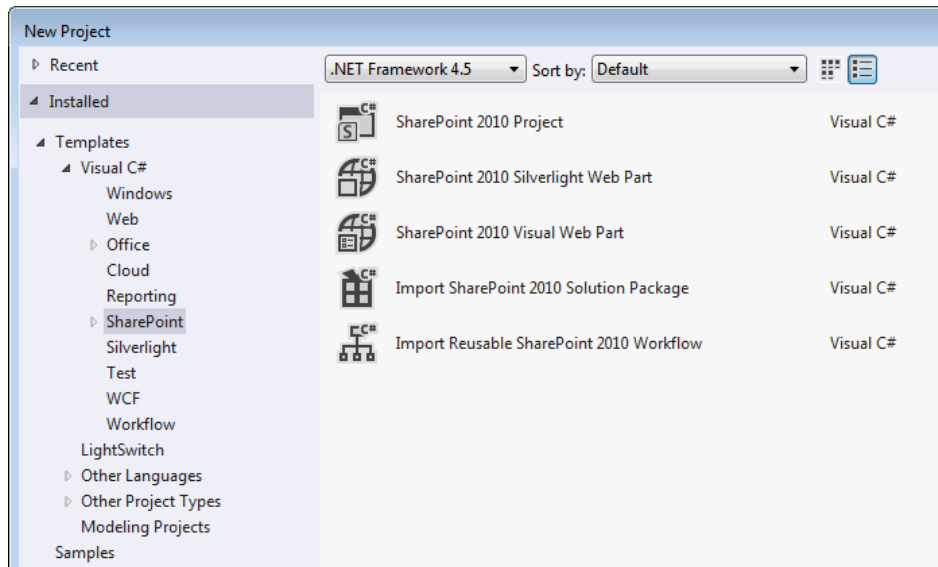


Visual Studio 2012 makes it easy to implement SharePoint solutions. The IDE it incorporates significant performance and reliability improvements over earlier versions of Visual Studio that are designed to help developers get SharePoint sites up and running quickly. Specifically, Visual Studio 2012 provides:

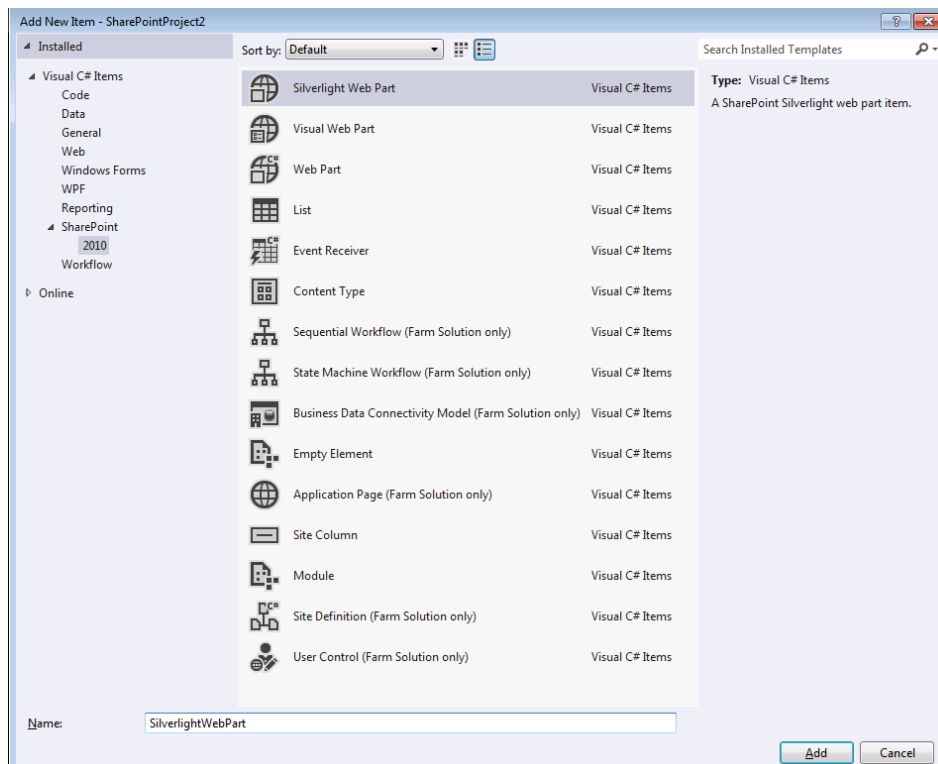
- Updated project templates and design tools to help accelerate project development.
- Enhanced support for and integration with SharePoint content types and other SharePoint elements within the Visual Studio IDE.
- Easier deployment to remote SharePoint servers.
- Full support for performance profiling.
- Unit testing your code and mocking SharePoint components by using the Microsoft Fakes framework.
- Full support for JavaScript components when building SharePoint sites.

## Accelerate SharePoint Site Development

Visual Studio 2012 includes simplified SharePoint project and item templates, together with a comprehensive collection of code samples that are closely matched to the business problems that users typically need to address. These new templates and samples enable developers to swiftly design and implement SharePoint sites that meet users' requirements.

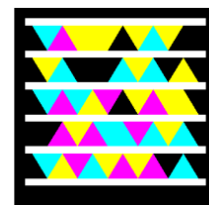


The SharePoint 2010 Project template provides a clean starting point for developing a new SharePoint application. A developer can quickly add an extensive collection of items such as Web Parts, lists, site columns, user-defined content types, modules, and pages to a SharePoint application by using the item templates provided with Visual Studio 2012.

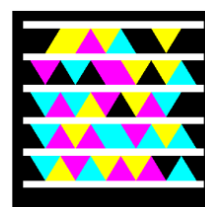
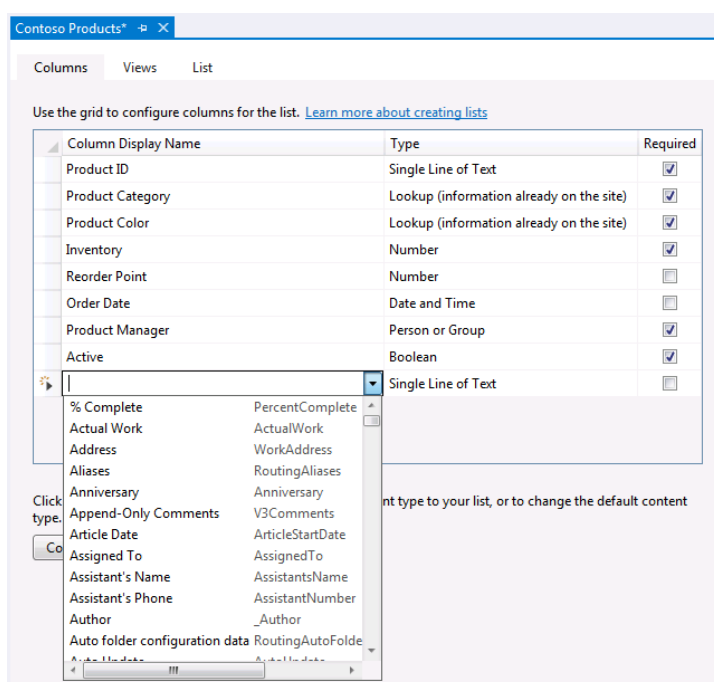


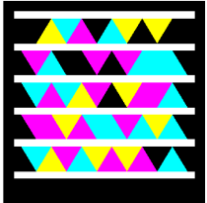
Most item templates provide designers and wizards, enabling a developer to concentrate on the functionality of the solution rather than being concerned about the format of the XML markup contained in the various SharePoint content and configuration files. The most commonly used items in a SharePoint site are Site Columns, Lists, and Content Types.

A Site Column represents an abstract data type displayed by a page in a SharePoint application, such as a telephone number, a comment, or the city name of a contact in a contact list. The Site Column wizard in Visual Studio 2012 makes it simple to define new site columns, providing a visual designer that generates the appropriate XML code for the Elements.xml file. For more information about Site Columns, visit the Introduction to Columns page at <http://msdn.microsoft.com/en-us/library/ms450825.aspx>.

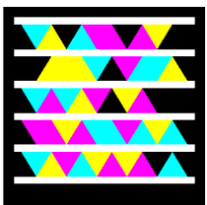
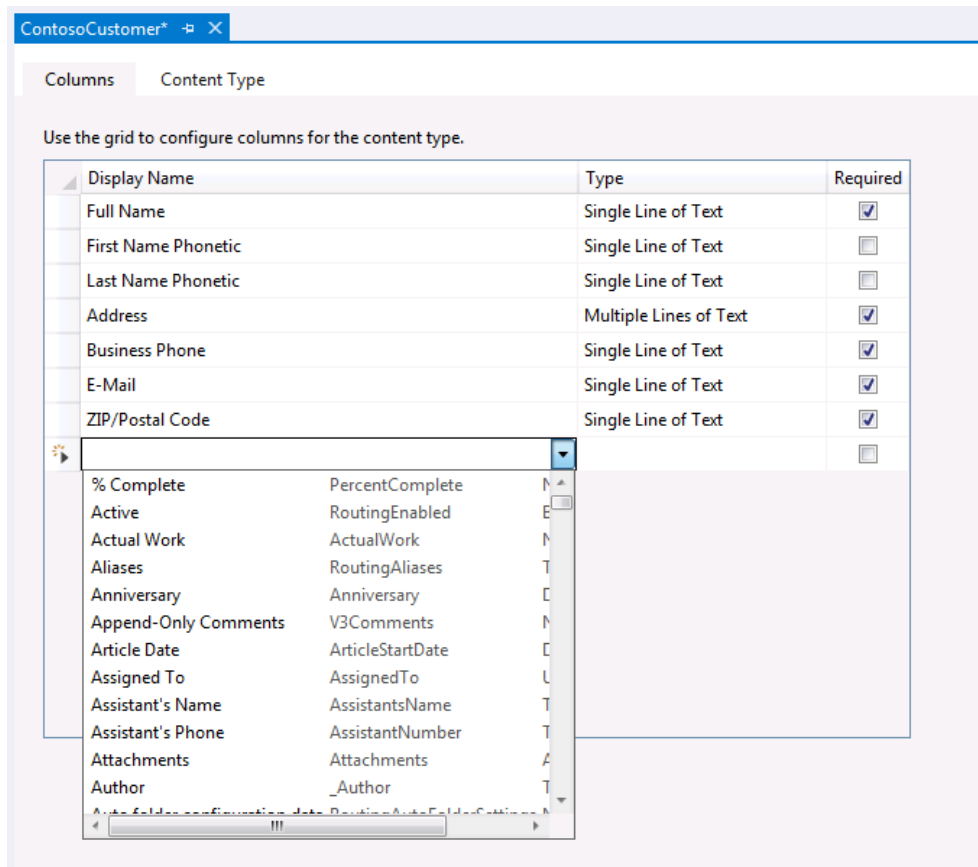


The new List Designer in Visual Studio 2012 enables developers to more easily define lists, and to specify the columns and views that appear on a SharePoint page. Again, this wizard automatically generates the XML markup for the list. The items in a list can be subject to custom validation rules to ensure that a user only enters valid data. For more information about defining and using lists, visit the List Forms page at <http://msdn.microsoft.com/en-us/library/aa543232.aspx> and the List Views page at <http://msdn.microsoft.com/en-us/library/ff604021.aspx>.

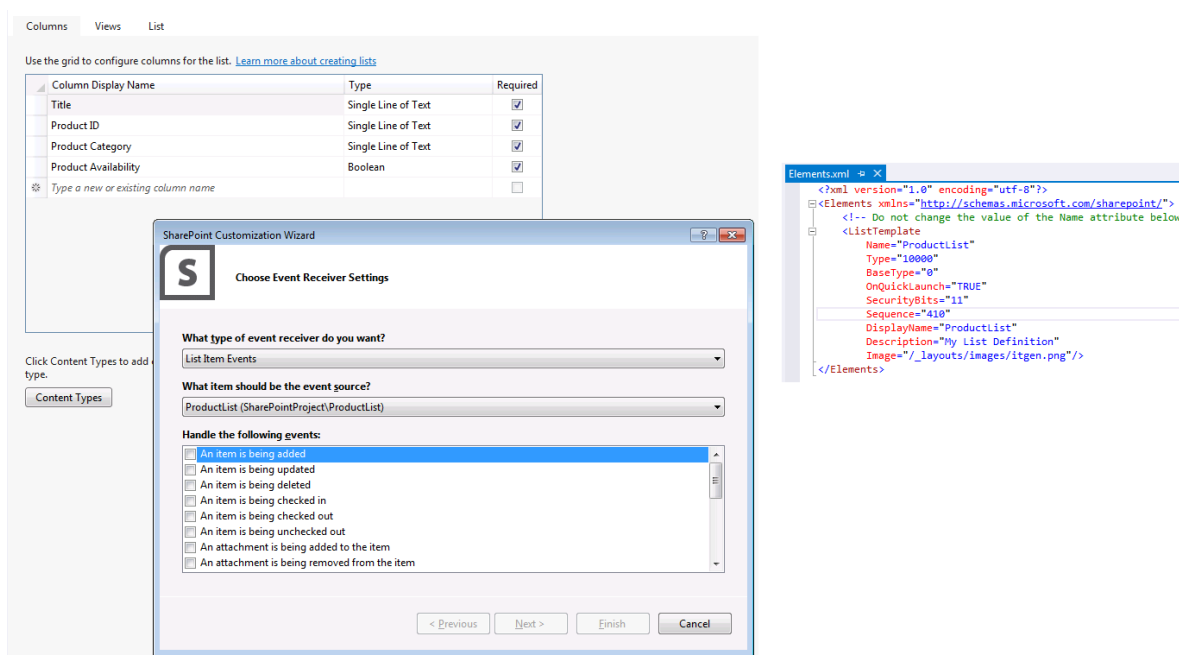




A Content Type represents a collection of resources. For example, a content type could contain contact information used by a contacts lists or task information referenced by a task list. A developer can reuse the same content type in different lists and document libraries in a SharePoint site. The Content Type designer in Visual Studio 2012 provides a visual tool for implementing content types. For more information, visit the Content Types page at <http://msdn.microsoft.com/en-us/library/ms479905.aspx>.



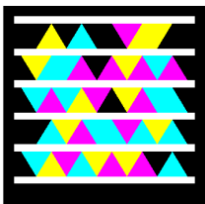
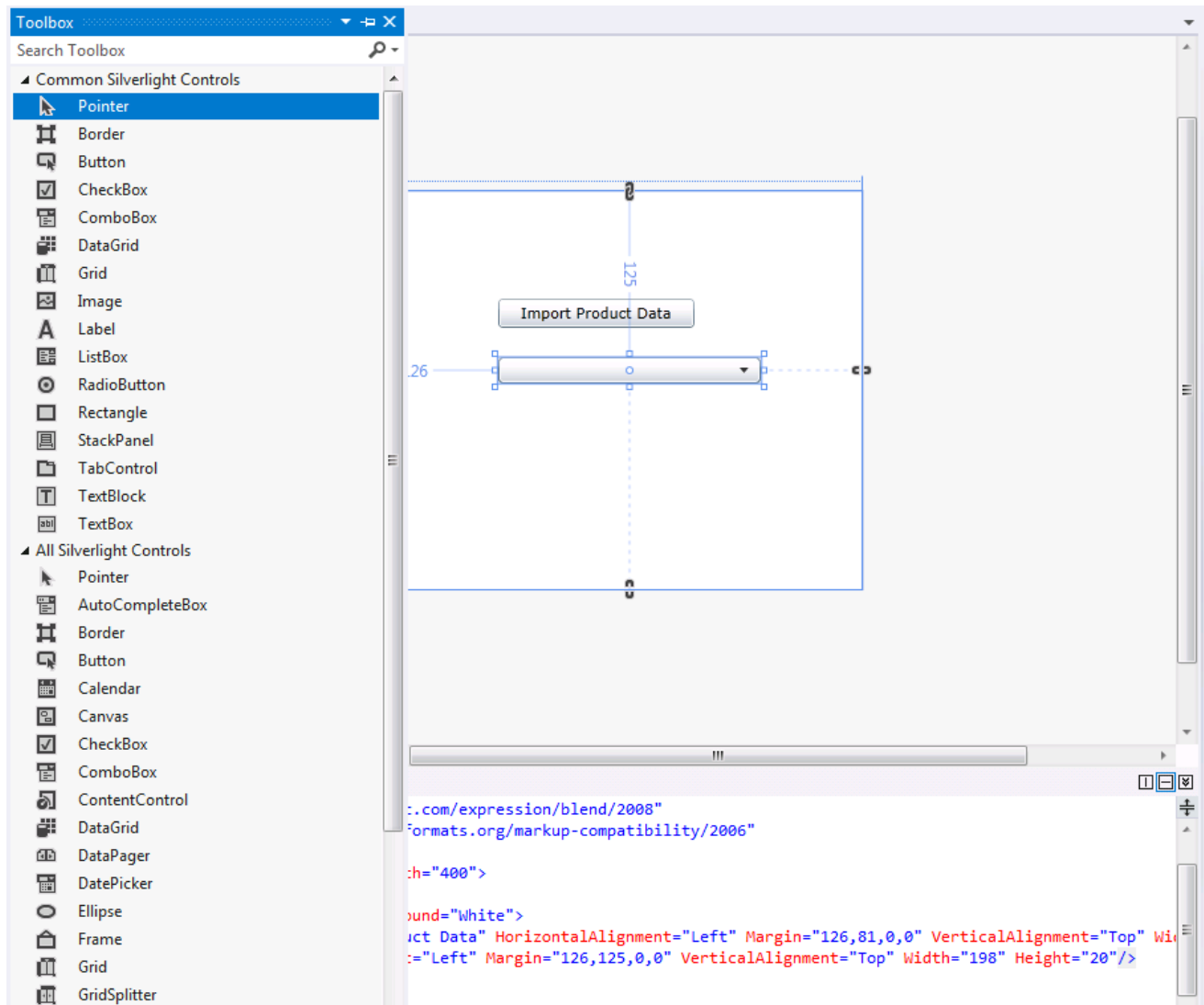
For a full list of the SharePoint templates supplied with Visual Studio 2012, see the SharePoint Project and Project Item Templates page at [http://msdn.microsoft.com/en-us/library/ee231554\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ee231554(v=vs.110).aspx).



## Easily Incorporate Silverlight Web Parts into SharePoint Sites

Using SharePoint Web Parts, a developer can build components that users can configure to customize the functionality, appearance, and behavior of their SharePoint pages. Visual Studio 2012 provides a fully integrated environment for creating, designing, packaging, deploying, and debugging Web Parts. Developers can lay out the user interface and implement the functionality behind the elements in this Web Part. Developers can create ASP.NET Web Parts, HTML and JavaScript Web Parts, or Silverlight Web Parts. The Designer window in Visual Studio 2012 enables a designer to visualize the appearance of the Web Part and add controls from the toolbox.

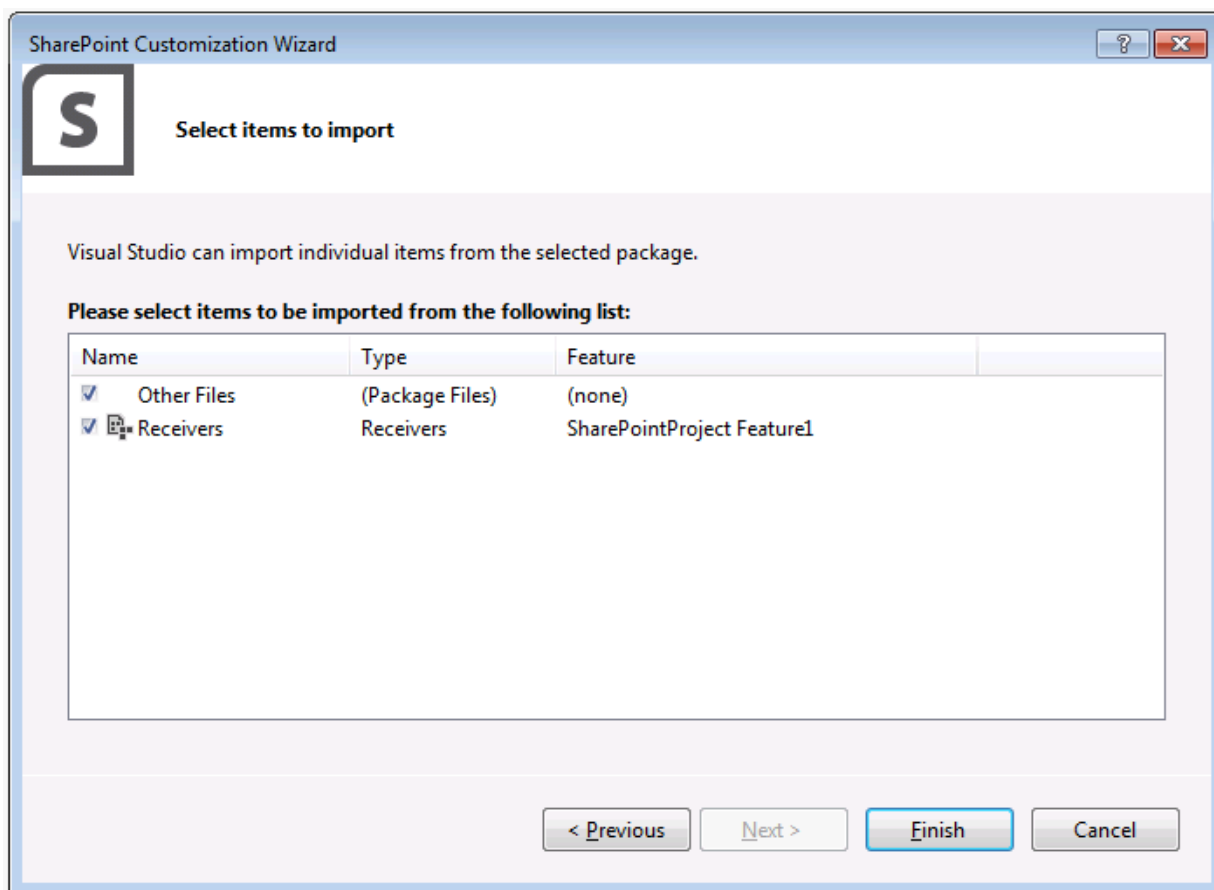
The Designer window in Visual Studio 2012 enables a designer to visualize the appearance of the Web Part using the controls provided in the Toolbox:



For detailed information about building Web Parts with Visual Studio 2012, visit the Creating Web Parts for SharePoint page at [http://msdn.microsoft.com/en-us/library/ee231579\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ee231579(v=vs.110).aspx).

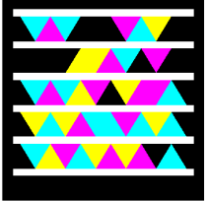
## Quickly Reuse Items from Existing SharePoint Sites

If an organization has existing SharePoint 2010 sites, a developer can import all or part of this site into Visual Studio 2012 by using the Import SharePoint 2010 Solution Package project. The project runs a wizard that prompts the developer for the location of the existing solution package file. The developer can then specify the items in this package to be imported.



The developer can customize the imported items and then redeploy them.

Similarly, the Import Reusable SharePoint 2010 Workflow template enables a developer to import and tailor a declarative workflow created previously by using SharePoint Designer 2010. The developer can deploy the modified workflow to a SharePoint site.



For more information, visit the Importing Items from an Existing SharePoint Site page at [http://msdn.microsoft.com/en-us/library/ee231584\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ee231584(v=vs.110).aspx).

## Access Resources Held on a Local SharePoint Server at Design Time

At design time, earlier versions of the Visual Studio IDE did not resolve references to some resources that were located on a local SharePoint server but that were outside the scope of your solution. These references were only resolved at runtime. This meant that useful design-time features such as IntelliSense for JavaScript files located on a local SharePoint server were unavailable, and the Design View window was unable to correctly display image files located on the local SharePoint server.

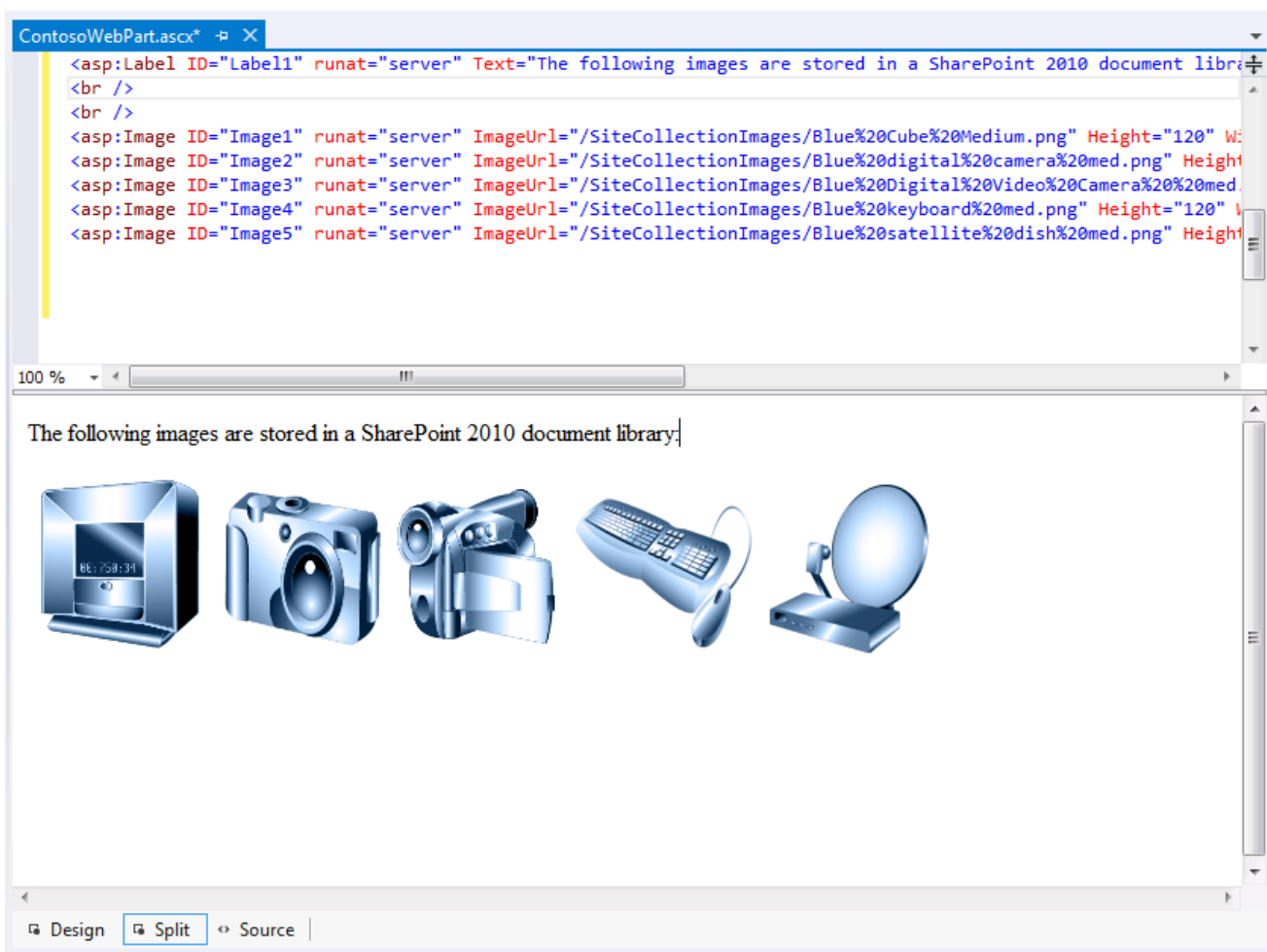
Visual Studio 2012 changes the way in which it handles references to these items. Visual Studio 2012 now provides IntelliSense for JavaScript files deployed to the SharePoint server and correctly displays images in the Design View window.

```

ContosoWebPart.ascx*
<%@ Register Tagprefix="Utilities" Namespace="Microsoft.SharePoint.Utilities" Assembly="Microsoft.SharePoint, Version=14.0.0.0, Culture=neutral, PublicKeyToken=71e9bce4c190c876" %>
<%@ Register Tagprefix="asp" Namespace="System.Web.UI" Assembly="System.Web.Extensions, Version=3.5.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" %>
<%@ Import Namespace="Microsoft.SharePoint" %>
<%@ Register Tagprefix="WebPartPages" Namespace="Microsoft.SharePoint.WebPartPages" Assembly="Microsoft.SharePoint.WebPartPages, Version=14.0.0.0, Culture=neutral, PublicKeyToken=71e9bce4c190c876" %>
<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="ContosoWebPart.ascx.cs" Inherits="SPPProject.ContosoWebPart" %>
<script type="text/javascript" src="/_layouts/MicrosoftAjax.js" ></script>
<script type="text/javascript" src="/_layouts/SP.debug.js" ></script>
<!-- Add a reference to a JavaScript file on the SharePoint site-->
<script type="text/javascript" src="/_catalogs/masterpage/JavaScript/ContosoUtilities.js"></script>
<script type="text/javascript" lang="javascript">
    ExecuteOrDelayUntilScriptLoaded(exampleFunction, "sp.js");
    //Use intellisense to browse the functions in your custom JavaScript file
    function exampleFunction() {
        Get
    }
</script>
  
```

- \$get
- getComputedStyle
- GetContosoCustomerDetails
- GetContosoCustomerList
- GetContosoProductDetails**
- GetContosoProductList
- getSelection
- XMLHttpRequestEventTarget



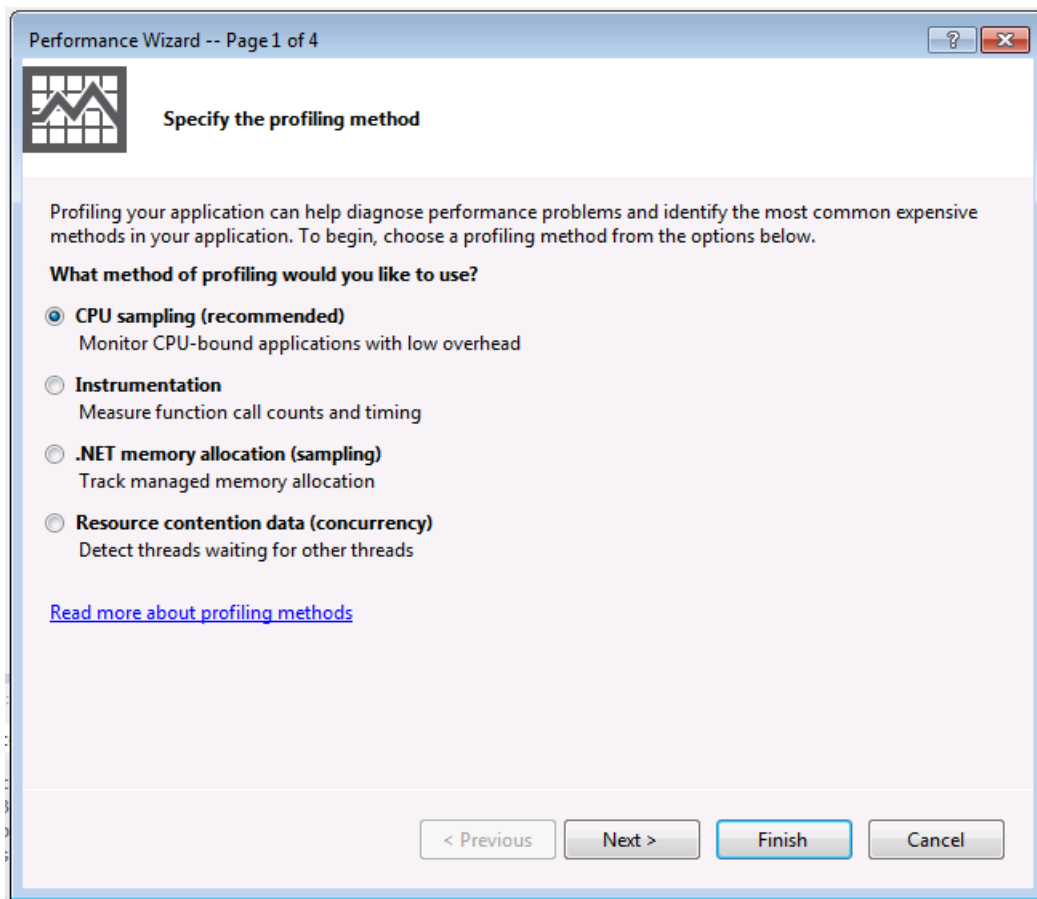


These enhancements improve the development experience and increase confidence that a SharePoint site will function as expected when it is deployed.

## Optimizing and Profiling SharePoint Code

SharePoint solutions can contain a variety of items, including Web Parts, lists, event receivers, and other items. Optimizing a complex mixture of these elements to ensure that they provide the best performance can also be a challenging task. Visual Studio 2012 includes features that can assist a developer in pinpointing the items in a SharePoint solution that

may require attention. Specifically, Visual Studio 2012 enables a developer to perform sampling, instrumentation, and tier integration profiling of SharePoint code by using the SharePoint Performance wizard.

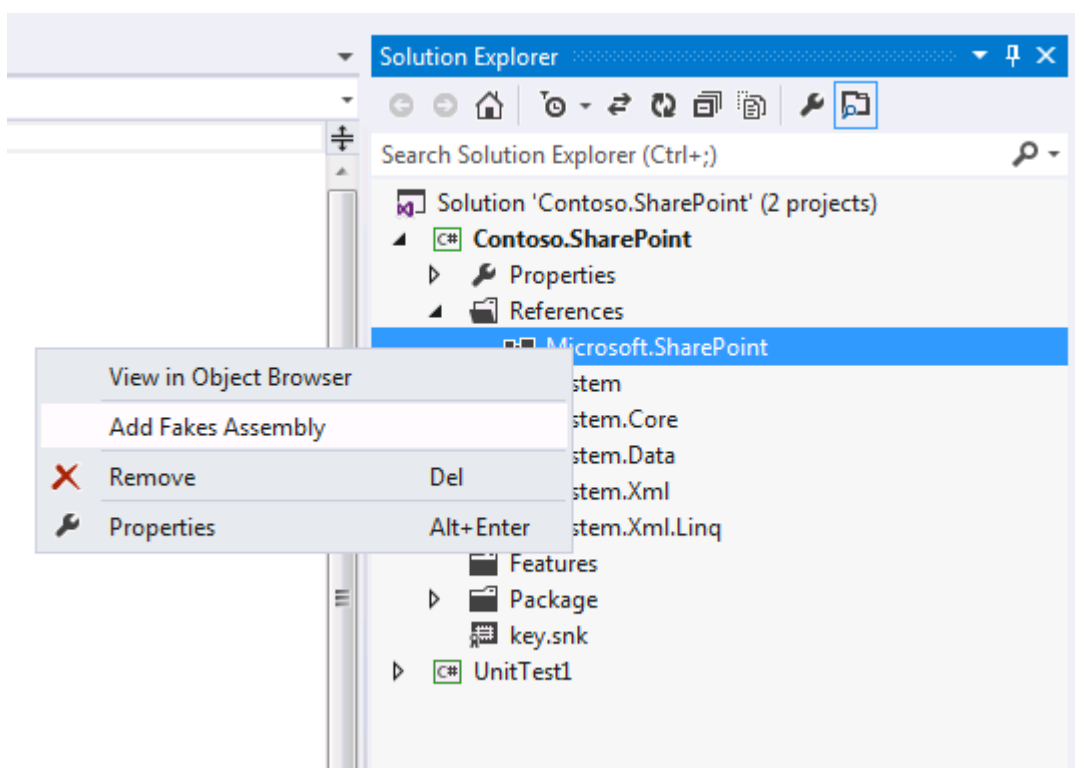


## Unit Testing Your Code and Mocking SharePoint Components

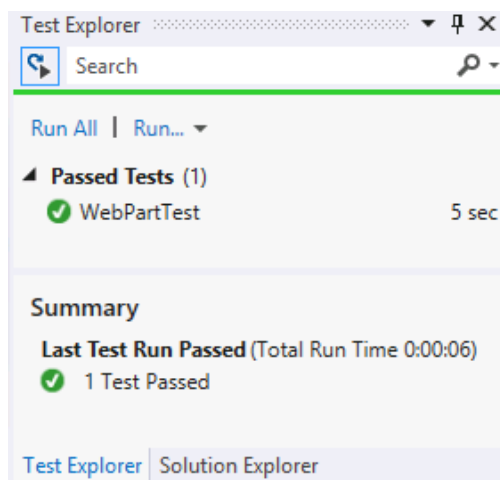
It is also important to thoroughly test the elements of a SharePoint site before making it publicly available.

Testing SharePoint code often requires assemblies provided on the SharePoint server by the SharePoint runtime. This can make isolated unit testing difficult.

Visual Studio 2012 includes the new Microsoft Fakes feature that makes it easy for you to test your SharePoint code without requiring a SharePoint server. Developers can use the Microsoft Fakes feature to create fake versions of SharePoint runtime assemblies that are then used with unit tests. Developers have full control over which assemblies they want to create fake versions of: Simply right-click an assembly and then click **Add Fakes Assembly**.



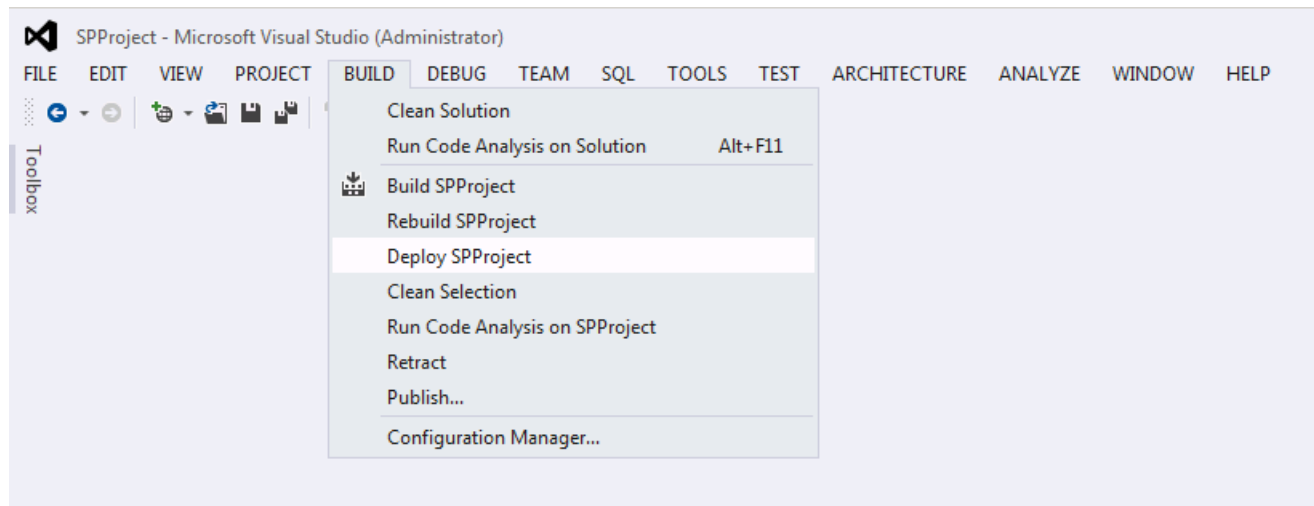
The Microsoft Fakes feature in Visual Studio 2012 ensures that unit tests that rely on the SharePoint runtime, for example, code that uses an instance of the `SPContext` object, run correctly. Developers can run the unit tests for their SharePoint application, step through their code and see the calls from the code to the fake SharePoint objects. No code changes are necessary when you migrate your solution to a SharePoint server.



## Quickly Deploy SharePoint Solutions

It just takes a couple of clicks in Visual Studio 2012 for you to publish SharePoint solutions and deploy them to a remote SharePoint server. This is a substantial improvement over the process implemented by earlier versions of Visual Studio.

An administrator can quickly upload a SharePoint solution to a SharePoint server, and a developer can also quickly package and deploy a solution locally for further development and testing.



## Summary

SharePoint offers a platform for building web sites that enable employees and partner organizations to collaborate and to share applications and data in a safer and more secure manner. Visual Studio 2012 includes a collection of tools, templates, and wizards that make it easy for developers to design and implement highly functional SharePoint sites. Additionally, the testing and optimization facilities in Visual Studio 2012 help to ensure that SharePoint sites function correctly and efficiently, while the deployment wizard enables an administrator to package a SharePoint site and upload it to a web server.

For a complete list of the new features provided by Visual Studio 2012 for developing SharePoint solutions, visit the What's New for SharePoint Development in Visual Studio 2012 page at [http://msdn.microsoft.com/en-us/library/ee290856\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/ee290856(v=vs.110).aspx).





# Application Life Cycle Management with Visual Studio 2012

With great software increasingly defining the ability of many businesses to compete and succeed, more attention is being paid to making software deliver on business value. To do this, businesses must take into account the whole lifecycle of software, from the moment it is conceived to when it is deployed and eventually retired.

Designing and constructing a modern application frequently involves the efforts of a range of participants with specific skills: the customers and business owners who derive value from the software, project managers who make sure the process meshes with other business deliverables, designers who create captivating user interfaces, developers who implement efficient business logic, testers who thoroughly test the application in a timely and comprehensive manner, and operations personnel who deploy and maintain the application. The expanded Application Lifecycle Management capabilities of Visual Studio 2012 enable even more stakeholders, including developers, to take advantage of their skills to build high-quality solutions focused on business value.

Visual Studio simplifies the interactions among team members and roles by streamlining the flow of information across the team. Microsoft Visual Studio Team Foundation Server 2012 acts as the central repository and integration hub that brings together tools, processes, and project artifacts to simplify the in-context collaboration of team members across functional disciplines and platforms. By providing an extensible and open connector model, Team Foundation Server ensures that over time, more and more stakeholders can closely collaborate with the development team to deliver value. With Team Foundation Server, teams have the freedom to choose which development process to adopt or customize; they can get started with a number of proven development methodologies, including a new Agile process template.

Teams often use many different tools and platforms to deliver successful solutions. By standardizing on a single integrated infrastructure, development teams that use a range of platforms and tools can improve collaboration and reduce operational costs. Team Foundation Server enables members of diverse development teams to focus on a single, integrated view of the development process, regardless of the platform they are targeting or using for development. Visual Studio Team Foundation Server works seamlessly with familiar client tools such as Visual Studio, Microsoft Test Manager, and Visual Studio Team Explorer. With Team Explorer Everywhere, Team Foundation Server seamlessly integrates team members who are using IDEs based on Eclipse™. Customers and business stakeholders can use simple tools such as Microsoft Excel and PowerPoint. Integration with Microsoft Project Server enables project managers to gain insight into the health of ongoing projects, understand how they support business needs, and help identify ways to

improve existing processes. In short, team members can focus on delivering value for customers rather than transitioning information between roles.

With Visual Studio Team Foundation Service, Microsoft has combined the power of Team Foundation Server with the simplicity and efficiency of the cloud. Team Foundation Service is built on the high availability and scalability of the Windows Azure platform and Windows SQL Database. Teams of all sizes can be up and running in very little time and are free to focus on delivering great software that delights their customers, while leaving the infrastructure management to Microsoft.

## The Benefits of Visual Studio 2012 and Team Foundation Server 2012

For all but the most trivial of solutions, application development is a multidisciplinary exercise that encompasses a wide variety of tasks and participants. The principal aim of any software development effort is to deliver a quality application that meets the customers' requirements in a timely and cost-effective manner. However, there are many obstacles that can get in the way of any development team. Many of the problems are caused by a lack of clear and effective communication among developers, project managers, and customers. Other problems may result from a lack of discipline that occurs when developers and project managers misuse or misunderstand the development process. Consequently, adaptable and easy-to-use tooling is critical to help overcome the issues that can arise when producing a potentially complex system.

Visual Studio 2012 and Team Foundation Server help to address these concerns by supplying a collection of tightly integrated tools to support and manage the entire application lifecycle.

The primary aims of Visual Studio 2012 and Visual Studio Team Foundation Server 2012 are to:

- Prioritize collaboration among everyone involved in developing an application, incorporating customers as equal members of the development team.



- Deliver timely and actionable feedback to reduce wasted effort and its associated costs.
- Provide natural and appropriate tools for the tasks involved in designing, developing, testing, delivering, and maintaining quality software.
- Support best practices for application development, while remaining independent of any specific methodology.

Collaboration and communication are essential elements in any successful software project. Earlier versions of Visual Studio Team Foundation Server focused on ensuring that developers and project managers had the tools and information that they needed to build and track the development of a project. However, your customers are a fundamentally important part of the team; if you do not deliver what they want, your efforts are wasted. In this release, we have extended the functionality in Visual Studio Team Foundation Server and provided additional features to enable developers and customers to work together more closely. These tools will help to close the communications gap by enabling customers to express their requirements more succinctly and to provide more effective and focused feedback.

## Increasing Efficiency and Reducing Waste

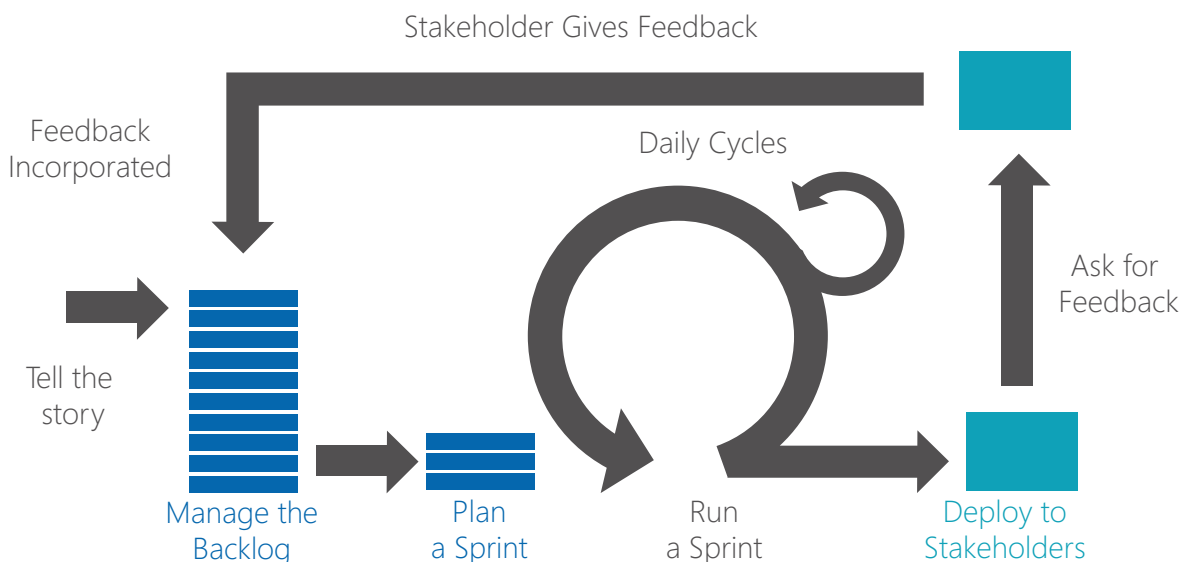
Traditional waterfall development approaches take the view that it is necessary to fully understand the users' requirements before attempting to design or build software. Although this approach is widely used and extremely valuable, it is sometimes difficult to react to user requirements that may change swiftly. Some of these modifications may be due in part to changes in the business environment, and others may be a result of misunderstandings that are caused by the differences between the lay vocabulary of the users and the technical vocabulary of the analysts and developers. Whatever the causes of change, the sooner that you can react, the less costly will be the impact.

Agile methodologies seek to address and accommodate change, primarily by breaking a large project down into smaller chunks, but many of the same issues can still arise, albeit earlier and on a smaller scale. The basic problem remains: Users must be able to express their requirements in a clear, concise, and unambiguous manner that can be understood by the developers. Customers' expectations may also grow unreasonably and quickly. For example, as technology advances, users frequently assume that any solution can easily include the latest and most desirable features. Technological advances frequently lead to more diverse solutions, and failure to understand this complexity—or the context in which it is best applied—can also cause much wasted effort.

### Including Customers in the Development Workflow

A key part of successful application development is to listen to what customers say and to build what they want. To assist with understanding and documenting users' requirements, Visual Studio 2012 provides storyboarding tools that enable the development team to discuss and visualize requirements with users and to understand how users expect to use the application. The storyboard can be saved to Microsoft SharePoint and linked to Visual Studio Team Foundation Server 2012 work items to provide context for developers.

In an Agile environment, developers must quickly demonstrate the results of their labors and obtain feedback from users, again in a clear, concise, manner. Visual Studio 2012 provides tools that enable a stakeholder to review the software and provide actionable feedback. The team can generate the appropriate tasks to deal with this feedback, plan for their implementation, and allocate the corresponding work to the appropriate members of the team. The feedback is stored in the Visual Studio Team Foundation Server 2012 database and is available to the developers who are responsible for incorporating any required changes.



### Working in a Familiar Environment

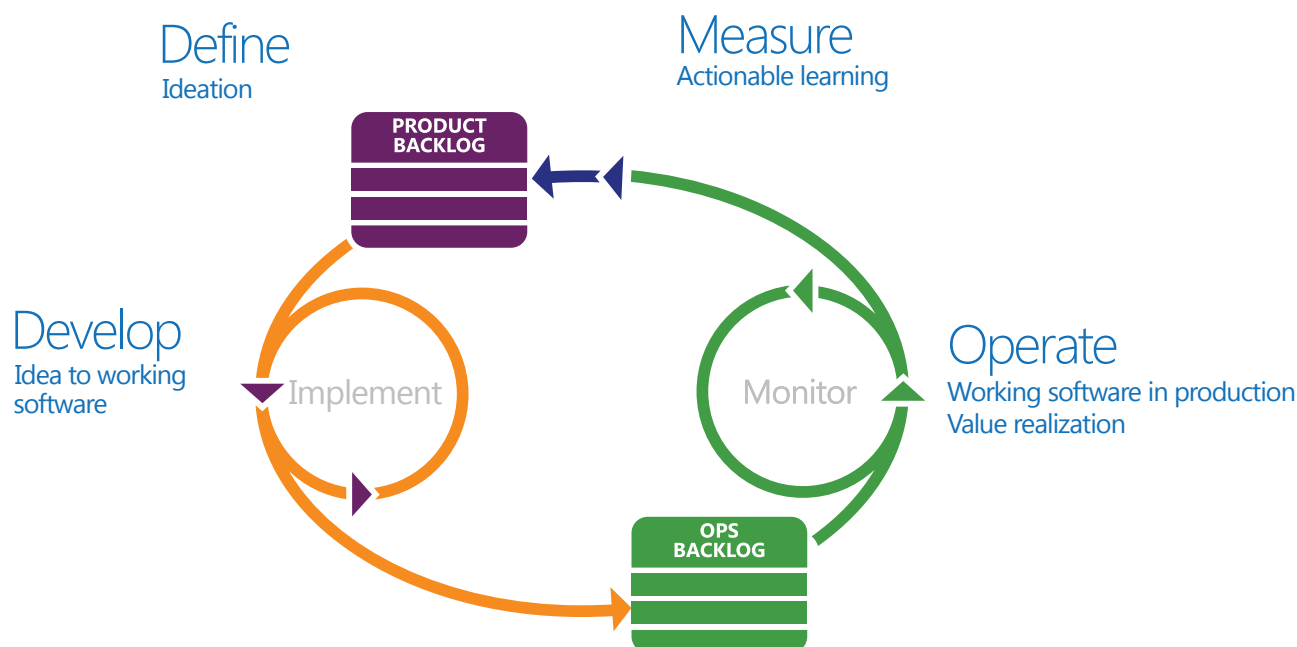
Another common source of waste is the tools themselves. Developers forced to use unfamiliar tools require time to learn how to use the tools properly. Additionally, developers cannot be efficient if they must use tools that do not fit into their team's development methodology. Team Explorer integrates the features that Visual Studio Team Foundation Server 2012 provides into Visual Studio 2012 in a natural manner, enabling developers to perform their tasks in a familiar environment. Visual Studio Team Foundation Server 2012 and Visual Studio 2012 do not mandate that you follow any specific development methodology: You can use them with the development approach with which your organization is most comfortable, whether it is Scrum, Agile, or a more formal method such as the Capability Maturity Model Integration (CMMI®).

Developers and project managers must communicate effectively with other team members who are working on the same project, and cooperating members should ideally have the same understanding of the solution and share the same priorities. Visual Studio 2012 enables developers and project managers to provide context as they communicate with each other, helping to ensure that they all have the same, complete understanding of the tasks on which they are working and how they fit together. Additionally, Visual Studio 2012 provides a live view of project status, ensuring that all participants are kept up to date.

## Managing the Lifecycle of a Modern Application

Modern software should be viewed as a living asset, with a lifecycle that spans the activities that represent the entire lifetime of the product, from identification of business requirements through development, testing, deployment, support, and maintenance, right up to the eventual retirement of the application. Consequently, the application lifecycle management toolset that you employ must be able to handle the development, testing, and deployment aspects of an application and be flexible enough to manage and incorporate feedback from operations staff. As the software is used, bugs are often discovered. So the toolset must provide facilities for recording issues, to enable the development team to resolve them, and to provide the operations staff with the information that they require to roll out the fixes. This interaction implies an inherent connection between the processes that the development team performs when it builds and maintains the software and those that the operations team uses to manage the software when it is deployed. Development and operations personnel must collaborate effectively because there is potential for bottlenecks to occur that will lead to delays and inefficiencies wherever handoffs between the two teams exist, especially when there is scope for miscommunication between them.

Visual Studio 2012 helps to align development with operations into a seamless workflow, reducing potential churn. For example, a common situation is that the operations team reports a bug in the live software that the development team is unable to reproduce outside the production environment. To help with situations such as this, Microsoft IntelliTrace has been extended to support the production environment. By using Visual Studio Team Foundation Server 2012, the operations team can record and capture the sequence of events that caused the bug and to store the trace information, together with the appropriate environmental data. The operations team can then hand this information, together with any other observations, to the development team for analysis and triage. Following an Agile approach, the project manager can subsequently create an appropriate work item and allocate it to a developer. This workflow helps to accelerate communications between the development and operations teams, ensuring that bugs are resolved and that updated software is deployed in a timely manner.



## Application Lifecycle Management Features in Visual Studio 2012

Visual Studio 2012 builds on the features of earlier versions, integrating with Visual Studio Team Foundation Server 2012 to implement a data store that houses the details of the tasks to be performed (the backlog) for the entire development team. This new implementation of Visual Studio retains the facilities of earlier versions, such as tools to support a highly customizable development process, integrated check-in and check-out facilities, and automated real-time reporting. It also includes workflows that actively incorporate the activities of the test team into the application lifecycle, thereby improving the quality assurance processes of the development effort.

We have updated the planning tools to provide transparency across the planning process and to enable the full participation of the development team. These tools scale as required, and you can apply them to match your organization's development processes and best practices.

With Visual Studio 2012, we have included new tools that improve the interaction with the key stakeholders of any project—the users. We provide features that fully integrate users into the development workflow, enabling them to participate in the design of an application and to provide clear and actionable feedback for the solution. As mentioned previously, we have also improved the support for operations staff who act as the users' advocates when the solution has been deployed; they can provide comprehensive information about bugs and other features in the system that must be dealt with.

We have optimized the developer experience that Team Explorer provides in Visual Studio 2012, freeing the development team from many of the distractions that can occur when it is working on a complex project and enabling the team to work more quickly and efficiently.

Finally, we have made testing a first-class citizen in the development process by integrating the test cycle into the development workflow, thereby enhancing the scope for automated and exploratory testing.

## Managing a Project

We have implemented an enhanced web experience for Visual Studio Team Foundation Server 2012. The Visual Studio Team Foundation Server 2012 Web Access application provides a rich experience that you can access by using a web browser through a web portal. We have also implemented touch controls to enable quick, natural interaction on touch-screen devices.

**Note:** The examples shown in this section are based on the Scrum process template for Visual Studio Team Foundation Server 2012. Visual Studio Team Foundation Server 2012 is designed to be methodology-neutral. We provide project templates for some common Agile methods that you can import and adapt to your own requirements. Other templates for different development processes are also available and, naturally, you are free to develop and publish your own custom templates.

The Web Access application gives the stakeholders who may not have access to Visual Studio 2012 an extremely easy way to interact with the rest of the team. For example, users can view their work items.

The screenshot shows the 'Assigned to me' view in the Team Foundation Service web interface. The browser address bar displays the URL: `https://contoso.tfspreview.com/DefaultCollection/New%20Web%20site/_workitems`. The page title is 'Team Foundation Service / New Web site'. The navigation bar includes 'HOME', 'WORK', 'SOURCE', and 'BUILD', with 'WORK' being the active tab. The left sidebar contains 'backlog', 'board', and 'work items', with 'work items' being the active view. The main content area shows a list of work items assigned to the user 'mikes@contoso.com'. The list has columns for ID, Work Item Type, Title, State, Area Path, and Iteration Path. The selected work item is ID 3, a Task titled 'Create login page interface'.

ID	Work Item Type	Title	State	Area Path	Iteration Path
3	Task	Create login page interface	To Do	New Web site	New Web site
22	Bug	Test bug	New	New Web site	New Web site
1	Product Backlog Item	Add log in page.	Committed	New Web site	New Web site
20	Code Review Request	User Details	Requested	New Web site	New Web site
5	Task	Conduct User Survey	In Progress	New Web site	New Web site
11	Feedback Response	Feedback Response from mikes@contoso.com for User Experience	Active	New Web site	New Web site
10	Feedback Request	User Experience	Active	New Web site	New Web site
7	Product Backlog Item	User Surveys	Committed	New Web site	New Web site

Below the list, the details for the selected work item 'Task 3: Create login page interface' are shown. The title is 'Create login page interface'. The Iteration is 'New Web site/Release 1/Sprint 4'. The STATUS is 'To Do'. The DETAILS section is currently collapsed.

Users can query the source control database and view the check-in status of items.

The screenshot shows the 'Source' view in the Team Foundation Service web interface. The browser address bar displays the URL: `https://contoso.tfspreview.com/DefaultCollection/New%20Web%20site/_VersionControl#path=/$/New Web site/RequirementsGathering/RequirementsGatheringTool/MainWindow.xaml.cs`. The page title is 'Team Foundation Service / New Web site'. The navigation bar includes 'HOME', 'WORK', 'SOURCE', and 'BUILD', with 'SOURCE' being the active tab. The left sidebar contains 'explorer', 'changesets', and 'shelvesets', with 'explorer' being the active view. The main content area shows the file history for 'MainWindow.xaml.cs'. The file path is '\$/New Web site/RequirementsGathering/RequirementsGatheringTool/MainWindow.xaml.cs'. The history shows a single entry: 'add' by 'mikes@contoso.com' on '24/02/2012' with the description 'Create initial version of survey tool'.

Path	Version	Author	Date	Description
\$/New Web site/RequirementsGathering/RequirementsGatheringTool/MainWindow.xaml.cs	5	mikes@contoso.com	24/02/2012	Create initial version of survey tool

The Web Access application is an ideal tool for geographically dispersed teams; all team members see the same views of the same data that is managed by the same instance of Visual Studio Team Foundation Server 2012.

If you are following a Scrum process, and you have the Scrum template installed, the Web Access application enables you to break a project down into work items and then assign these work items to sprints in the product backlog. Work items can include rich text and images to provide additional information and helpful feedback to developers.

The screenshot displays the TFS Web Access interface for a 'Product Backlog'. The left-hand navigation pane shows a hierarchy: 'Product Backlog' (selected), 'Past', 'Current' (containing 'Sprint 4'), and 'Future' (containing 'Sprint 5' and 'Sprint 6'). The main content area is titled 'Product Backlog' and features a 'board' tab. Below the tab, there's a 'Create Backlog Query' button and a 'Column Options' link. A modal window is open for adding a new item, showing 'Type: Product Backlog Item' and a 'Title' field. The main table lists work items with columns: Order, Title, State, Effort, and Iteration Path. One item is listed: Order 1, Title 'Test bug', State 'New', and Iteration Path 'New Web site\Release 1\Sprint 4'. The top navigation bar includes 'HOME', 'WORK' (selected), 'SOURCE', and 'BUILD'. The bottom status bar contains copyright text and links to 'Terms of Service', 'Privacy Statement', 'Service Notifications', and 'Trademarks'.

You can view the details of an individual sprint and edit it to decompose the items it contains into tasks.



Team Foundation Service / New Web site

HOME WORK SOURCE BUILD

backlog board work items

Product Backlog

Past

Current

Sprint 4

Future

Sprint 5

Sprint 6

Sprint 4

23 February - 29 February  
5 work days

work details on

Work

Work By: Activity

Work By: Assigned To

© Microsoft Corporation. All rights reserved. Terms of Service Privacy Statement Service Notifications Trademarks Microsoft

The Web Access application also enables you to drag items and reorder them if you need to reprioritize work. You can reassign items to different sprints by simply dragging them onto a sprint in the leftmost pane.

Team Foundation Service / New Web site

HOME WORK SOURCE BUILD

backlog board

Product Backlog

Past

Current

Sprint 4

Future

Sprint 5

Sprint 6

Task 3: Create login page interface

Create login page interface

Iteration New Web site\Release 1\Sprint 4

STATUS

Assigned To mikes@contoso.com

State To Do

Reason Additional work found

Blocked

DETAILS

Remaining Work

Backlog Priority

Activity Development

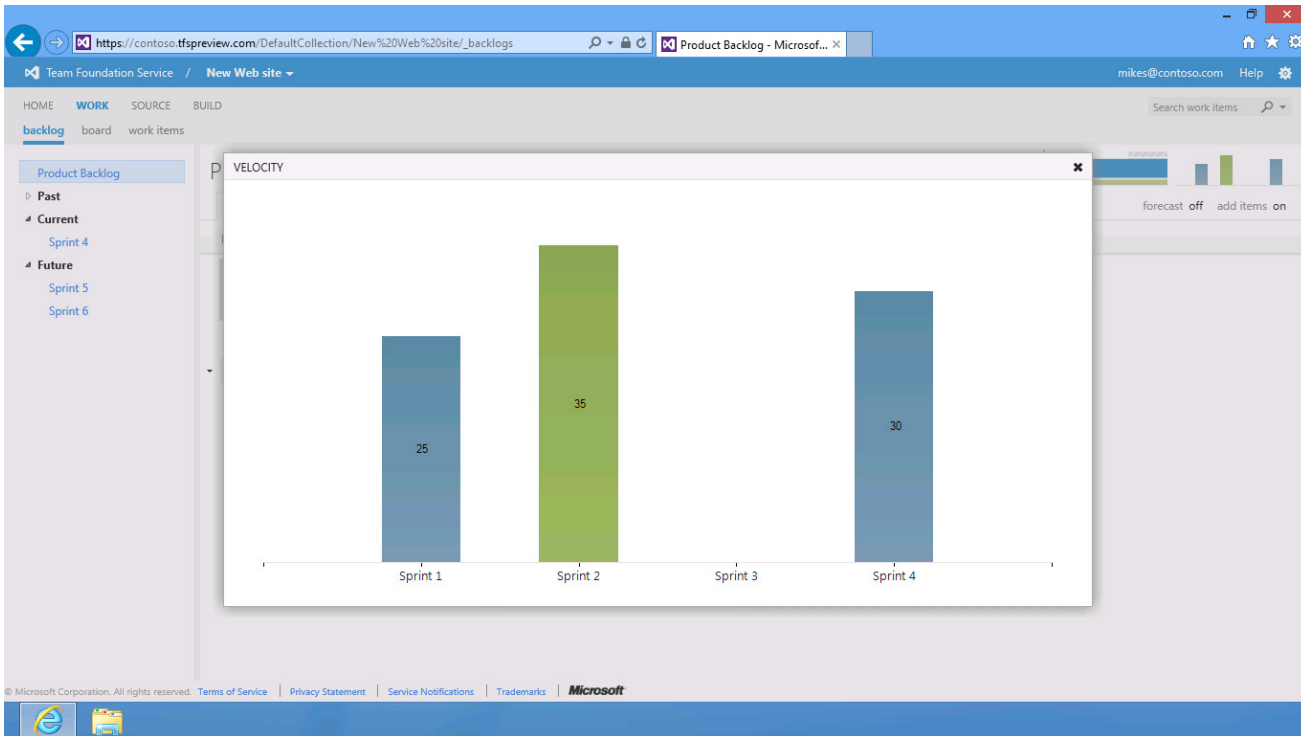
Area New Web site

DESCRIPTION

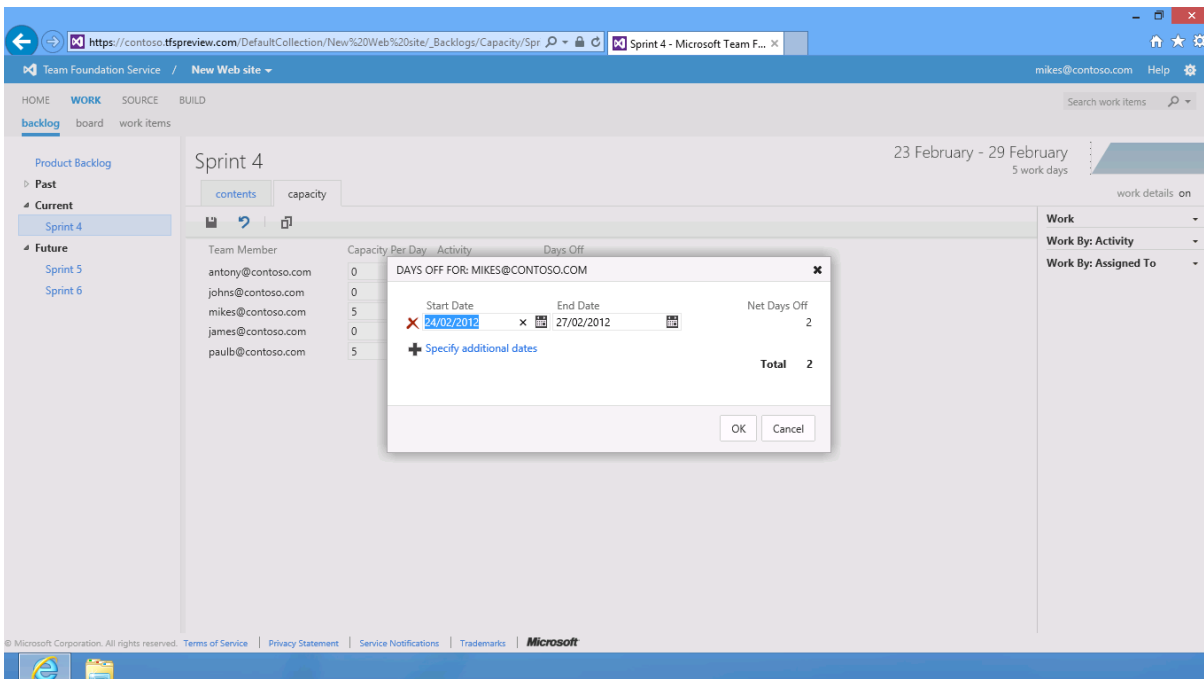
HISTORY LINKS (2) ATTACHMENTS

Save Save and Close Cancel

© Microsoft Corporation. All rights reserved. Terms of Service Privacy Statement Service Notifications Trademarks Microsoft



Of course, you can only assign work to team members if they are available to work on the project. The capacity-planning view in the Web Access application enables you to record and adjust the availability of any team member and to amend the team capacity accordingly.



The task board enables you to manage the status of the various tasks that make up the backlog items. For example, you can view which tasks are in progress, yet to be started, and completed. As team members start and complete tasks, you can perform drag-and-drop operations to move items around the task board so that you have a complete and up-to-date view of a project's progress.

The screenshot shows the TFS task board for 'Sprint 4' (February 23 - February 29, 5 work days). The board is divided into three columns: 'TO DO', 'IN PROGRESS', and 'DONE'. On the left, there is a list of backlog items with expand/collapse icons (+/-). The tasks are as follows:

Backlog Item	TO DO	IN PROGRESS	DONE
Add log in page.	Create login page interface mikes@contoso...		
Add purchase history page			Develop purchase history page paulb@contoso...
Test bug			

At the bottom of the page, there is a footer with copyright information: © Microsoft Corporation. All rights reserved. | Terms of Service | Privacy Statement | Service Notifications | Trademarks | Microsoft.

Another key feature of the task board is the set of workflow rules that help to prevent accidental errors. For example, suppose that you attempt to move a task from the **Done** column back to the **In Progress** column without specifying the additional effort that is required to complete the work? In this case, the task board highlights the error.

The screenshot shows the TFS web interface for a project named 'New Web site'. The 'WORK' tab is active, displaying a Kanban board for 'Sprint 4'. The board is organized into three columns: 'TO DO', 'IN PROGRESS', and 'DONE'. In the 'TO DO' column, there are two tasks: 'Add log in page.' and 'Add purchase history page'. The 'IN PROGRESS' column contains one task: 'Develop purchase history page'. The 'DONE' column is currently empty. The board is for 'Sprint 4' which runs from '23 February - 29 February' (5 work days). The user 'mikes@contoso.com' is logged in. The interface includes a top navigation bar with 'HOME', 'WORK', 'SOURCE', and 'BUILD' tabs, and a search bar for work items.

You can edit the properties of the task to specify the remaining required effort.

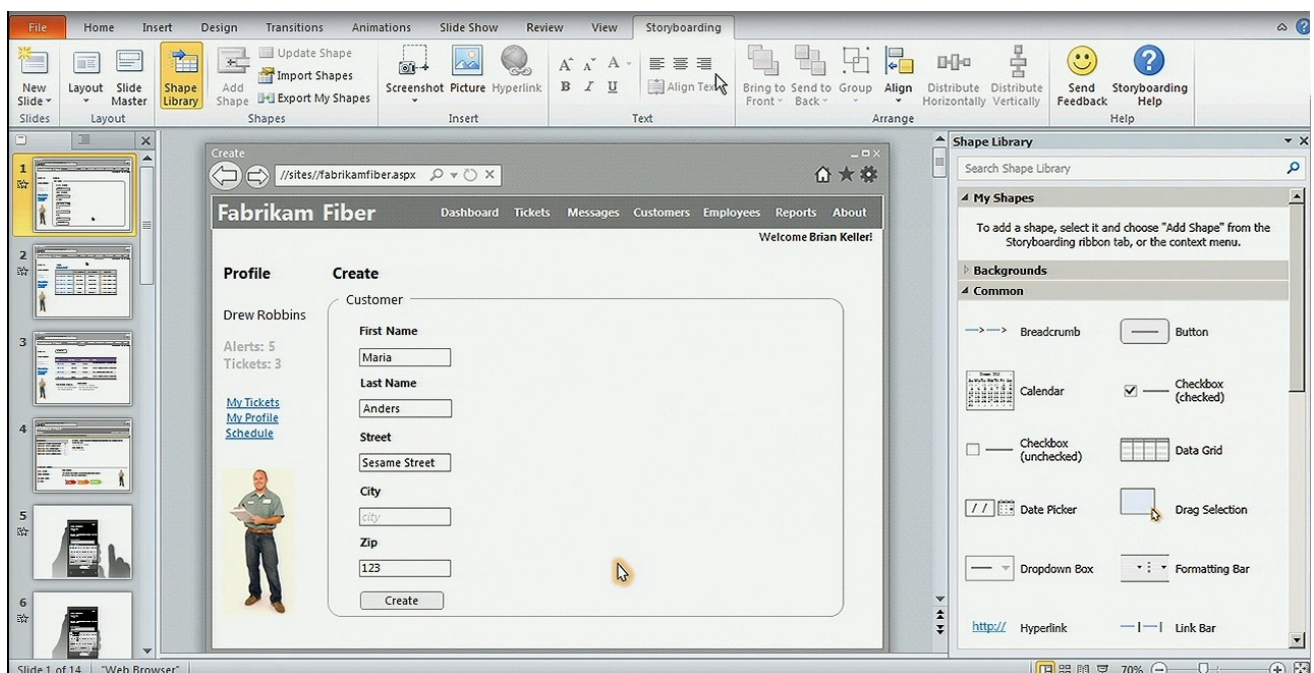
The screenshot shows the 'Edit Task' dialog box for the task 'Develop purchase history page'. The dialog box has a title bar 'Task 9\*: Field 'Remaining Work' cannot be empty.' and a close button. It contains fields for 'Assigned To' (paulb@contoso.com), 'State' (In Progress), 'Reason' (Additional work found), and 'Blocked'. The 'DETAILS' section includes 'Remaining Work' (a yellow input field), 'Backlog Priority', 'Activity' (Development), and 'Area' (New Web site). The 'DESCRIPTION' section has a text area with a rich text editor toolbar. The 'HISTORY', 'LINKS (1)', and 'ATTACHMENTS' sections are also visible. The dialog box has 'Save', 'Save and Close', and 'Cancel' buttons at the bottom.

The default view of the task board shows all of the tasks for each of the work items, but you can highlight a specific team member if you simply need to see the tasks that are assigned to that developer. You can also use the task board to reassign tasks among developers by using a drag-and-drop operation to move tasks between the developers' schedules.

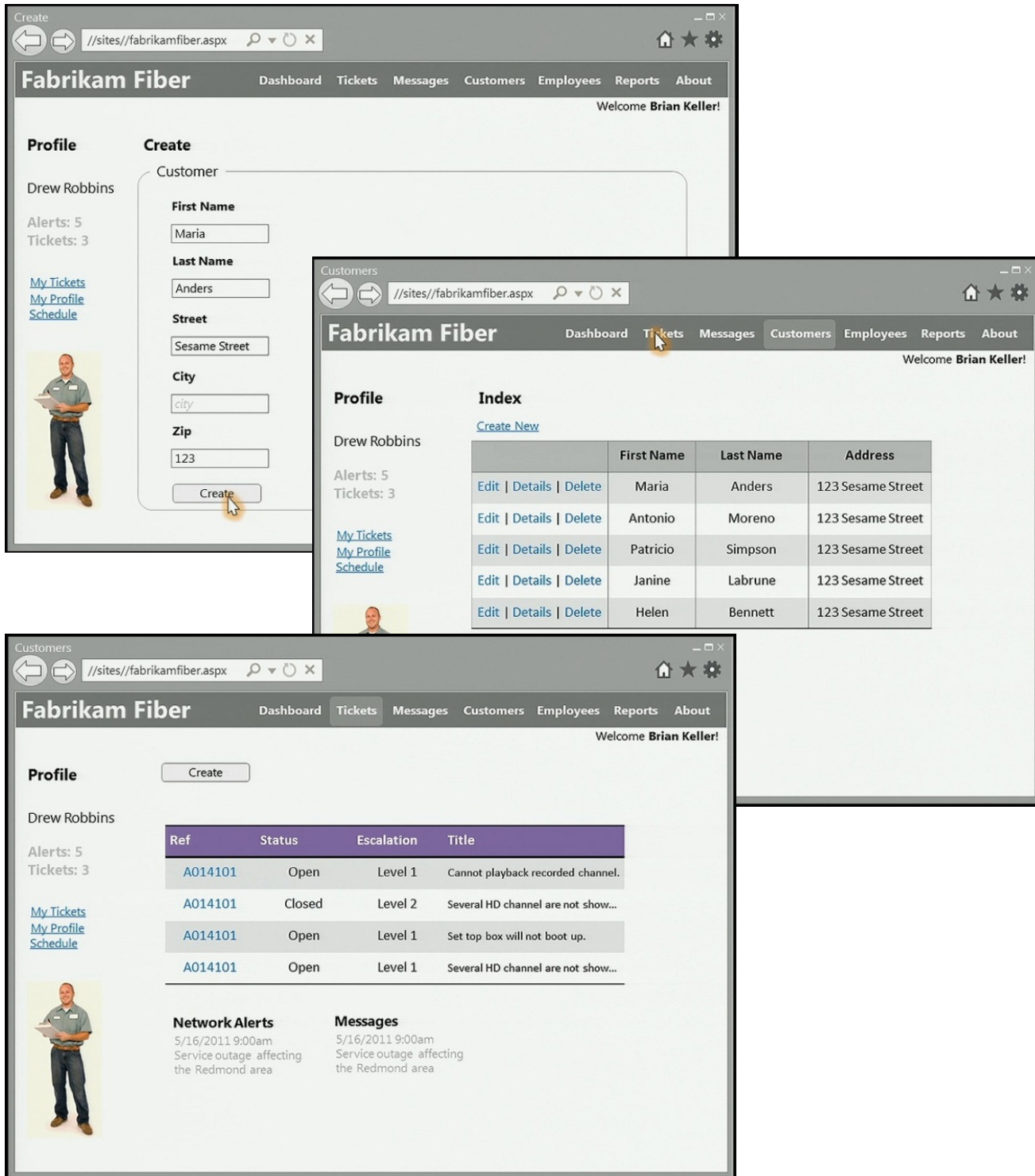
In short, by using the Web Access application you can easily manage the project, assess the capacity of the development team, prioritize work, manage the dependencies and relationships among work items and developers, monitor the status of the project, and generally keep track of the project. The project data is live, and any changes are reflected to developers who are using Visual Studio 2012.

## Gathering User Requirements

The Storyboarding Tool is an add-in for PowerPoint that enables a developer to work with a user to understand the business requirements and to plan how the application should function.

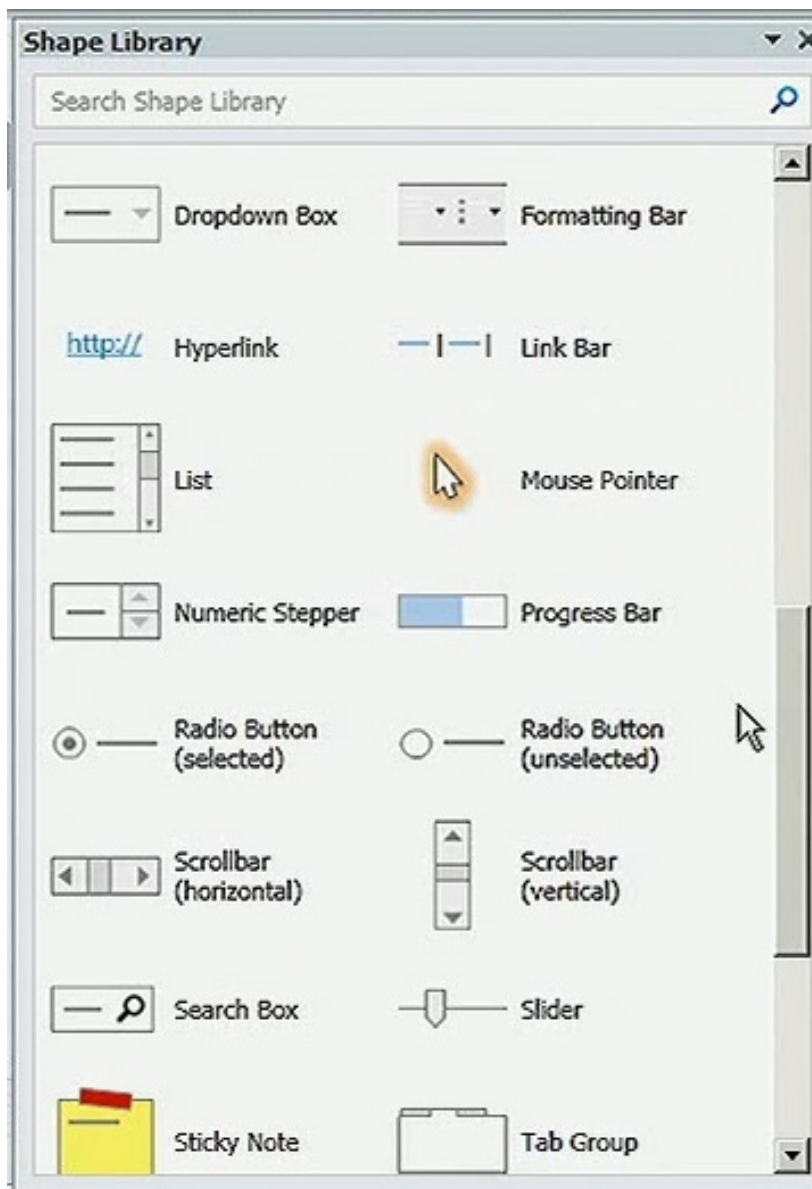


With this tool, you can quickly mock-up a user interface and create animations that illustrate how the user expects to navigate through the application, show sample data, simulate events, and generally get a feel for what the user wants.



It is easy to collaborate with a group of users, too, by displaying the storyboard through a projector and making live changes as users participate in the discussion. You can also e-mail the storyboard to users and gather their feedback offline; users simply use PowerPoint to view it.

The Storyboarding Tool provides a rich library of controls, but you can also create custom shapes, import them into the storyboard shape library, and share them with the rest of the team.



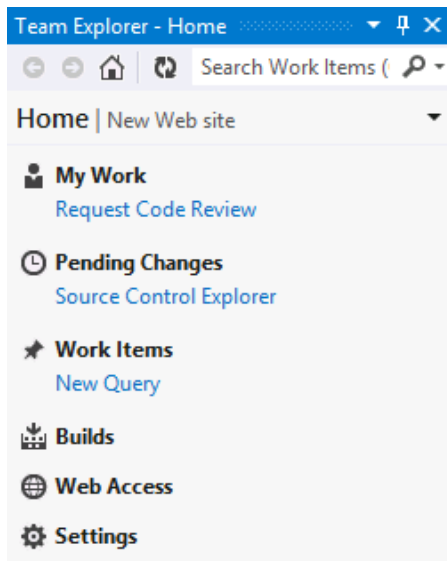
Using the Storyboarding Tool helps to ensure that you fully understand your users' requirements, and it enables your users to portray their vision for the application. All of this can happen very quickly, without requiring that you write a single line of code or waste time generating expensive prototypes.

## Improving Developer Productivity

A key aim of Visual Studio 2012 is to help improve developer productivity. With this in mind, Visual Studio now includes a number of new capabilities. These capabilities enable a developer to focus on the tasks that are necessary to develop quality software, and they supply a quick and easy mechanism for collaborating with other developers working on the same project.

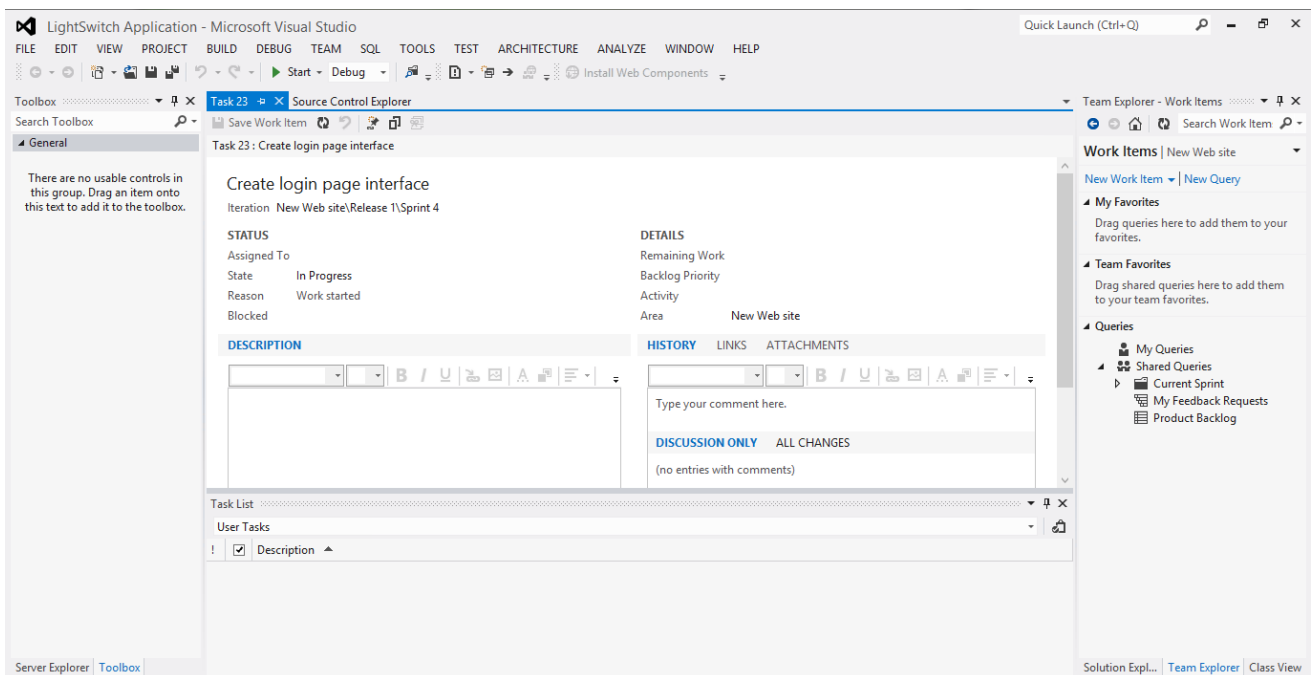
### Team Explorer

Team Explorer is the developer's primary interface for interacting with Visual Studio Team Foundation Server 2012. In Visual Studio 2012, we have changed Team Explorer to provide a more organized user interface that simplifies many of the tasks that team members perform. Team Explorer makes use of asynchronous communications with Visual Studio Team Foundation Server 2012 to retrieve and save data, resulting in shorter response times. Team Explorer also provides a Web Access link that quickly launches the Visual Studio Team Foundation Server 2012 Web Access application.

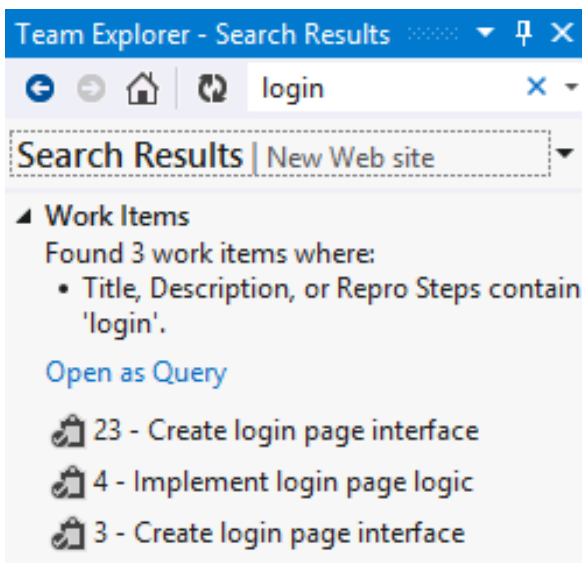




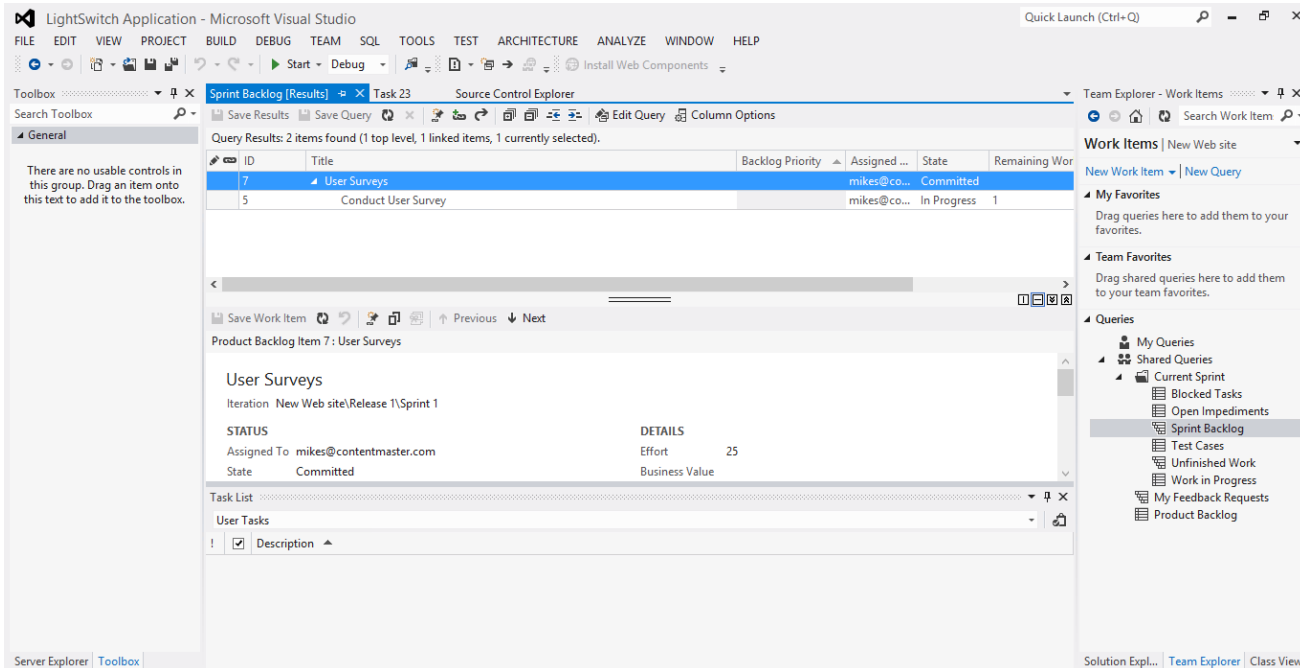
The My Work pane lists the developer's tasks and displays the status of these tasks. It is the view that a typical developer will leave open much of the time. Developers can double-click a task to get access to the details or to edit the task. In addition, the View All link displays the details of the tasks in Visual Studio 2012 in a format that is similar to the one used by the Web Access application.



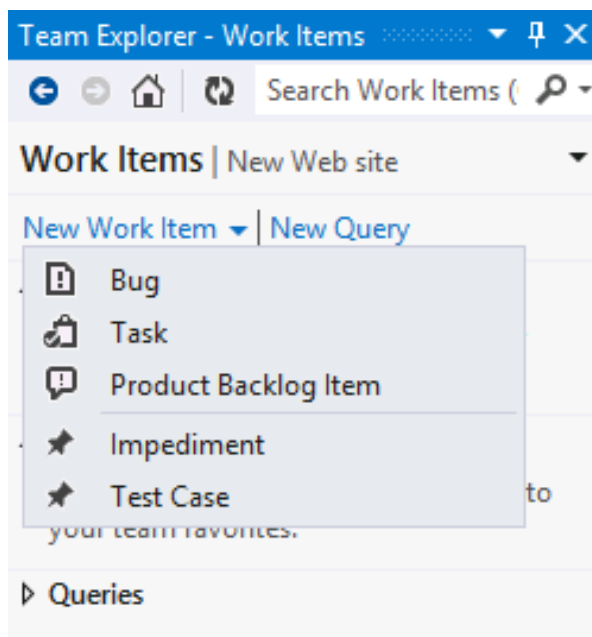
The integrated search feature helps a developer to quickly locate project items and other information.



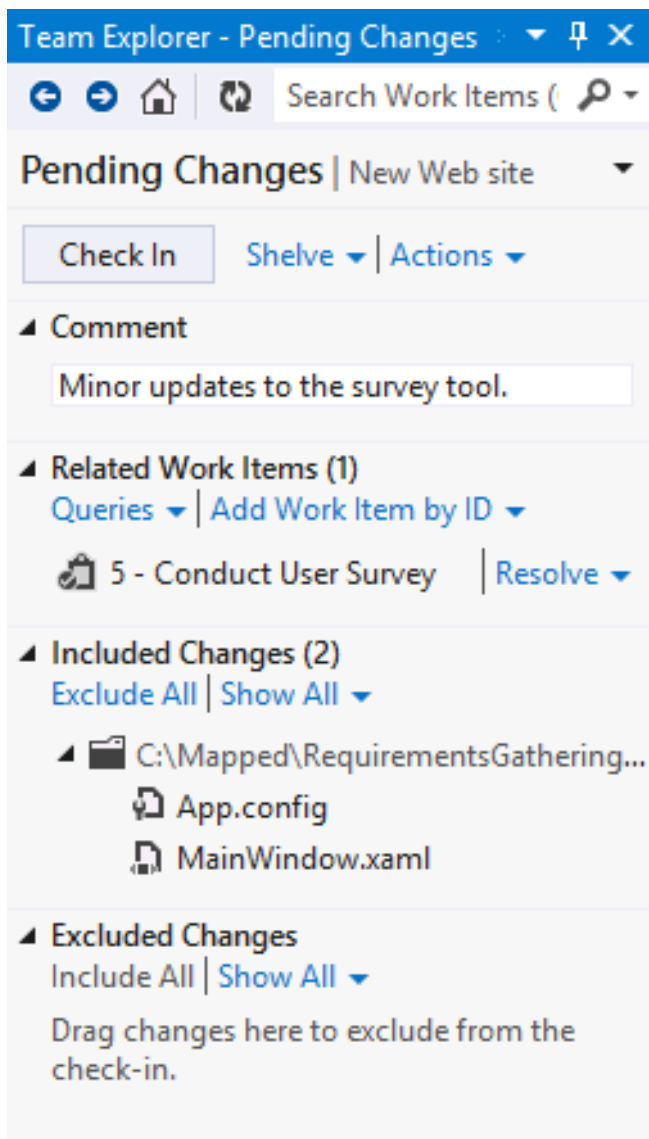
Other panes in Team Explorer provide access to the other important functions that a developer must perform and to the data that the developer requires. For example, the Work Items pane provides a broader view of all of the work items across the project.



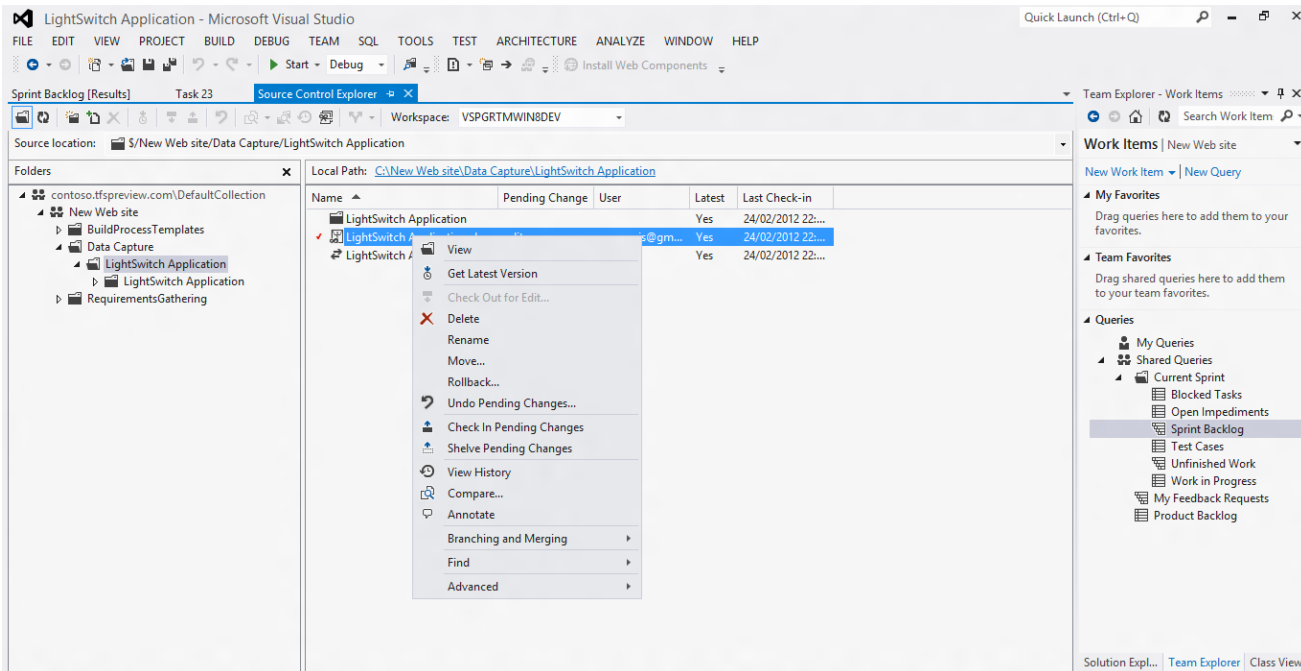
The **New Work Item** link in this pane enables a developer to quickly define additional work items and to add them to the project.



Using the Pending Changes pane, a developer can check source code in and out of the Visual Studio Team Foundation Server 2012 database. The Pending Changes pane displays the context of the check-in operation, enabling the developer to quickly verify that the correct files are saved and to specify that the corresponding work item has been resolved.



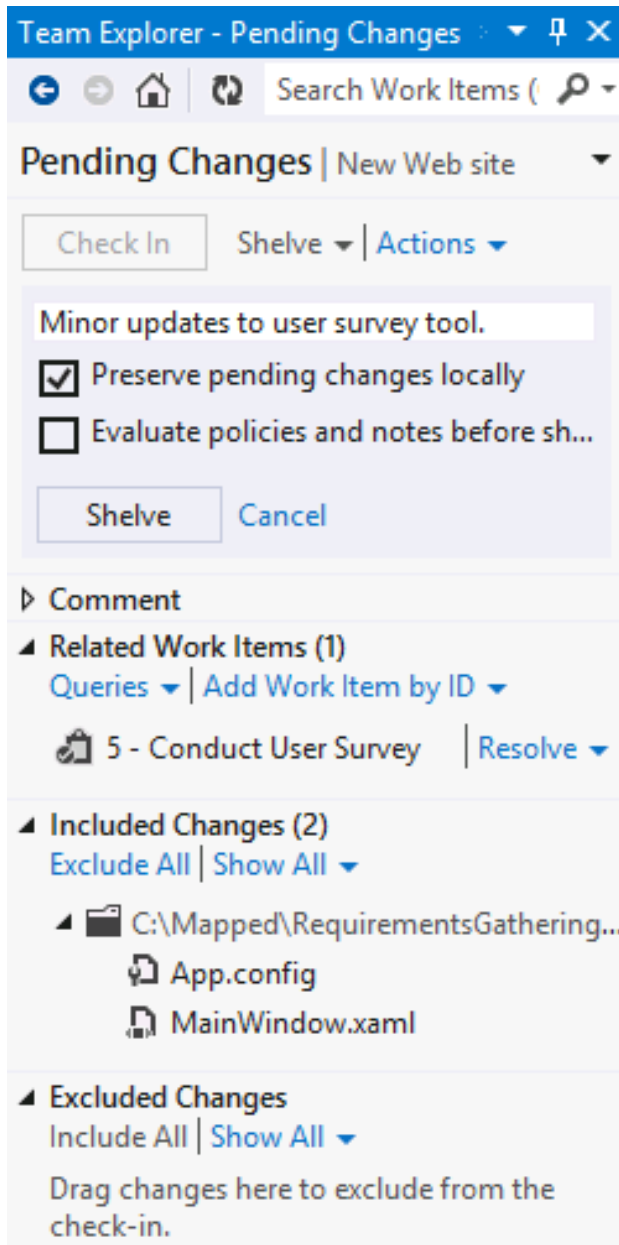
The Source Control Explorer window enables you to perform check-in and check-out operations at the source-file level.



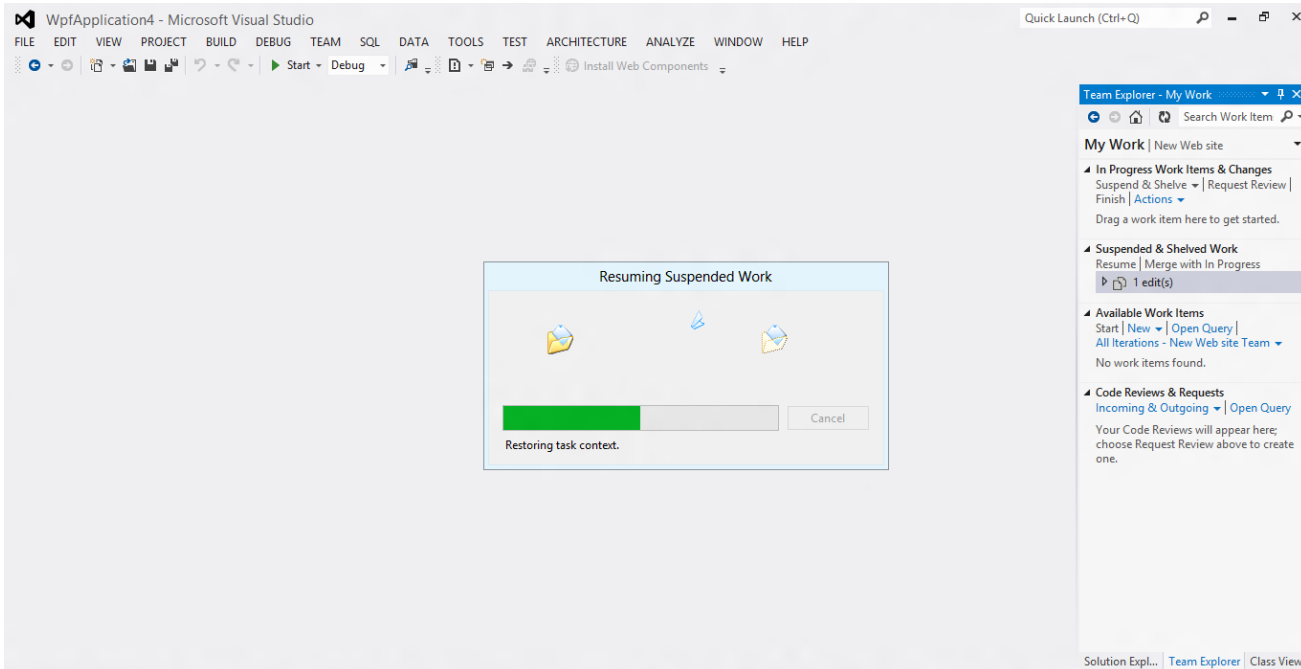
## Shelvesets

If you are working on a complex piece of code, you may have files open in several windows across multiple monitors, you may have various breakpoints set in your code, you may have watch windows set up, and so on. In the midst of all this, suppose that your project manager walks in and asks you to put everything on hold and work on another urgent task that concerns a different piece of code. In this situation, Visual Studio 2012 enables you to preserve not only the state of your work but also your entire development environment, so that you can switch to the urgent task and then resume your current work later, hopefully minimizing the effects of the interruption.

To save your current context, you can simply click Shelve in the My Work pane in Team Explorer. Visual Studio Team Foundation Server 2012 saves your code and settings to a shelveset in the Visual Studio Team Foundation Server 2012 database. A shelveset is a serialized representation of your environment, including the state and positions of any open windows.



To continue with your work, you can select your shelveset from the list of suspended activities and then resume it. Visual Studio 2012 downloads the code and settings from the Visual Studio Team Foundation Server 2012 database, restores your various windows, watches, breakpoints, and other configuration data, and enables you to carry on from exactly where you left off.



## Code Reviews

Collaboration is a fundamental feature of application lifecycle management that has been improved in Visual Studio Team Foundation Server 2012. One way in which Visual Studio 2012 facilitates collaboration is by enabling developers to request and perform code reviews by using Team Explorer. This feature defines a workflow in Visual Studio Team Foundation Server 2012 that saves project state and routes review requests as work items to team members. As always with Visual Studio Team Foundation Server 2012, these workflows are independent of any specific process or methodology, so you can incorporate code reviews at any convenient point in the project lifecycle.

The [Request Code Review](#) link in the My Work pane enables you to create a new code review task and assign it to one or more other developers.

Team Explorer - New Code Review

←

→

🏠

🔄

Search Work Items ( 🔍 )

New Code Review | New Web site

2 edit(s) | [View Changes](#)

Select one or more reviewers to review your changes and enter a comment for them if appropriate

[Add Recent Reviewers](#)

👤

paulb@contoso.com

✕

👤

Enter the name of a reviewer <op>

▼

Add Reviewer | Press Enter to add this review

📋

Minor updates to the survey tool.

🏷️

New Web site

▼

💬

Minor updates to the survey tool.  
Can you check the data access code?

Submit Request

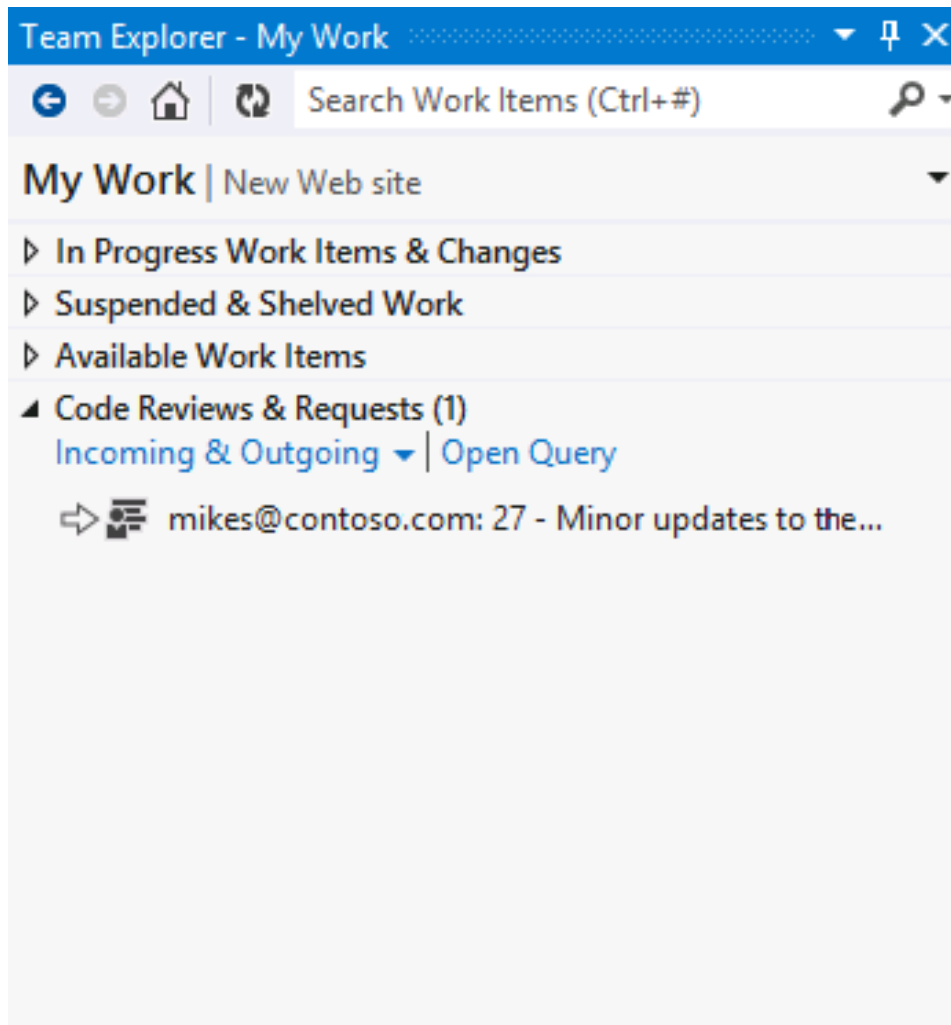
Cancel

Related Work Items (1)

📁

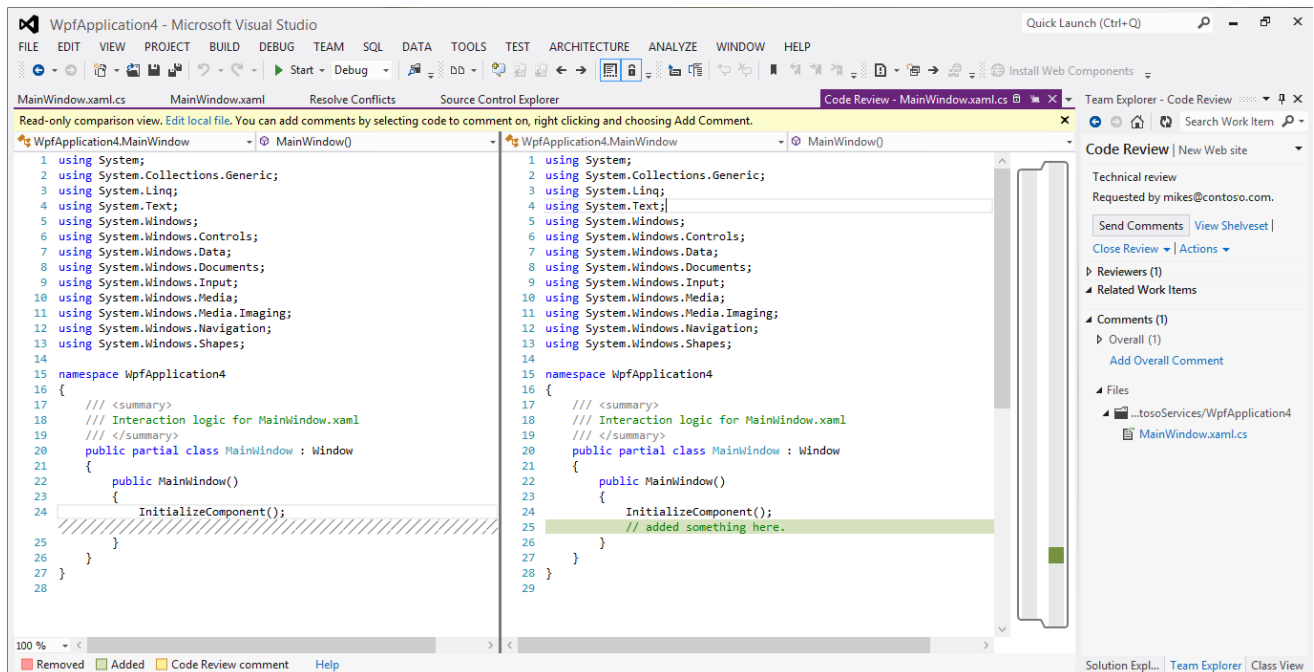
5 - Conduct User Survey

Visual Studio Team Foundation Server 2012 creates a shelveset for the project and stores it in the Visual Studio Team Foundation Server 2012 database. This code review request then appears in the My Work pane for each reviewer in Team Explorer.

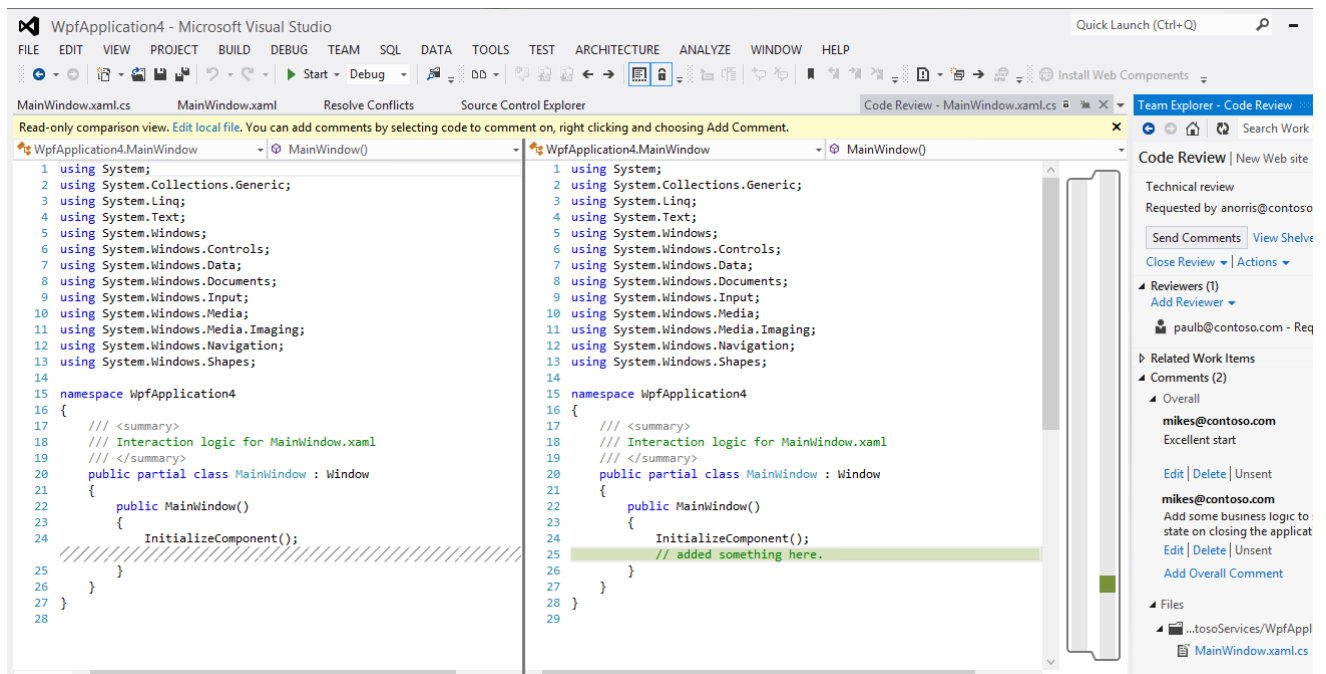


The reviewer can accept or decline the review and respond to any messages or queries associated with the code review. When the reviewer examines the code, Visual Studio 2012 displays the code by using a "diff" format, which shows the original code and the changes that were made by the developer who requested the review. This feature enables the reviewer to quickly understand the scope of the changes.



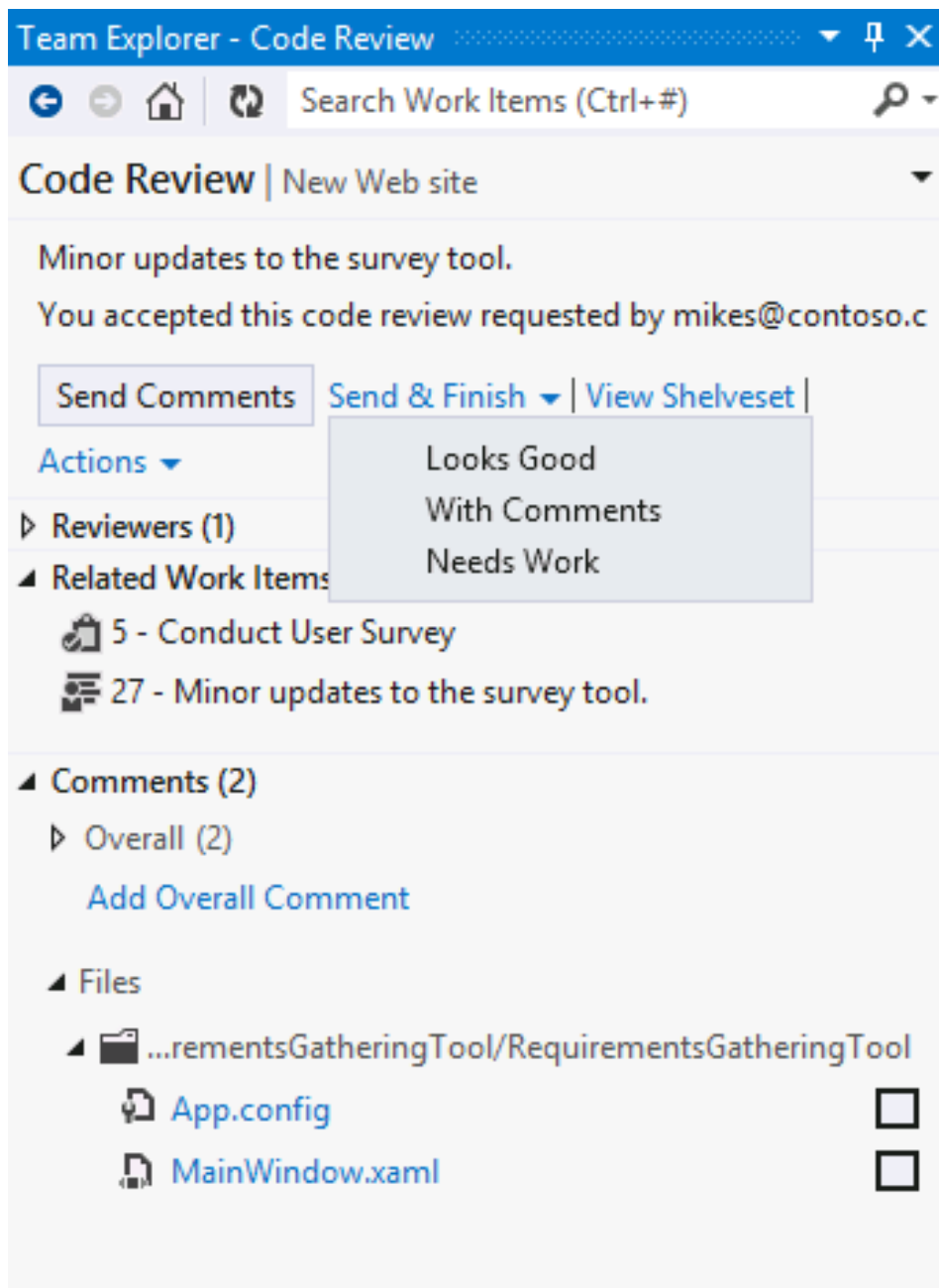


The reviewer can add comments to either the file or to specific parts of the code. These comments are sent back as part of the code review.

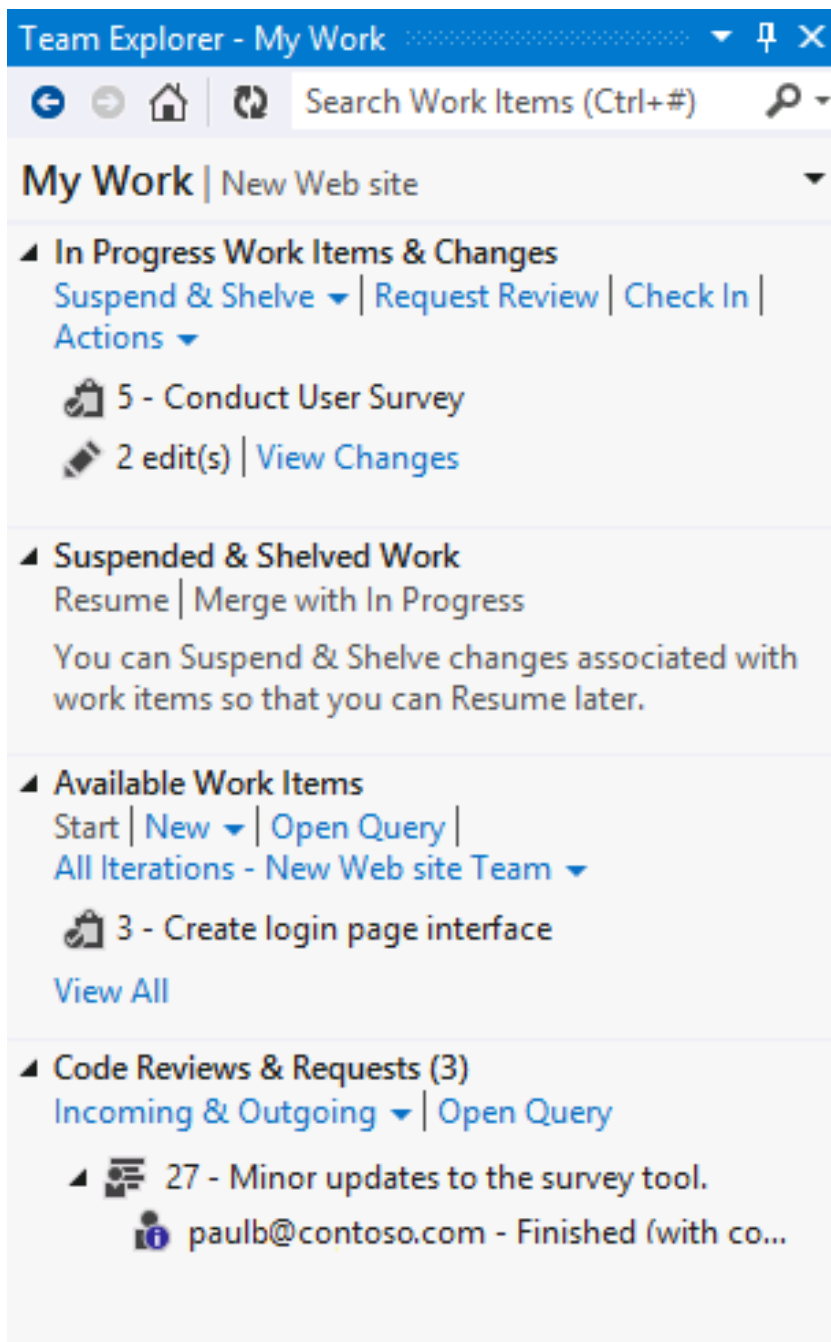


If the reviewer requires additional context or wants to suggest modifications to the code, the reviewer can unshelve the shelveset for the code review. This opens the entire solution and puts it in the state that the developer was using when the code review was requested.

When the review is complete, the **Send & Finish** button in the Code Review pane enables the reviewer to provide a quick summary of the feedback.



The results of the review appear in the Code Reviews & Requests section of the My Work pane in Team Explorer for the developer that requested the review.

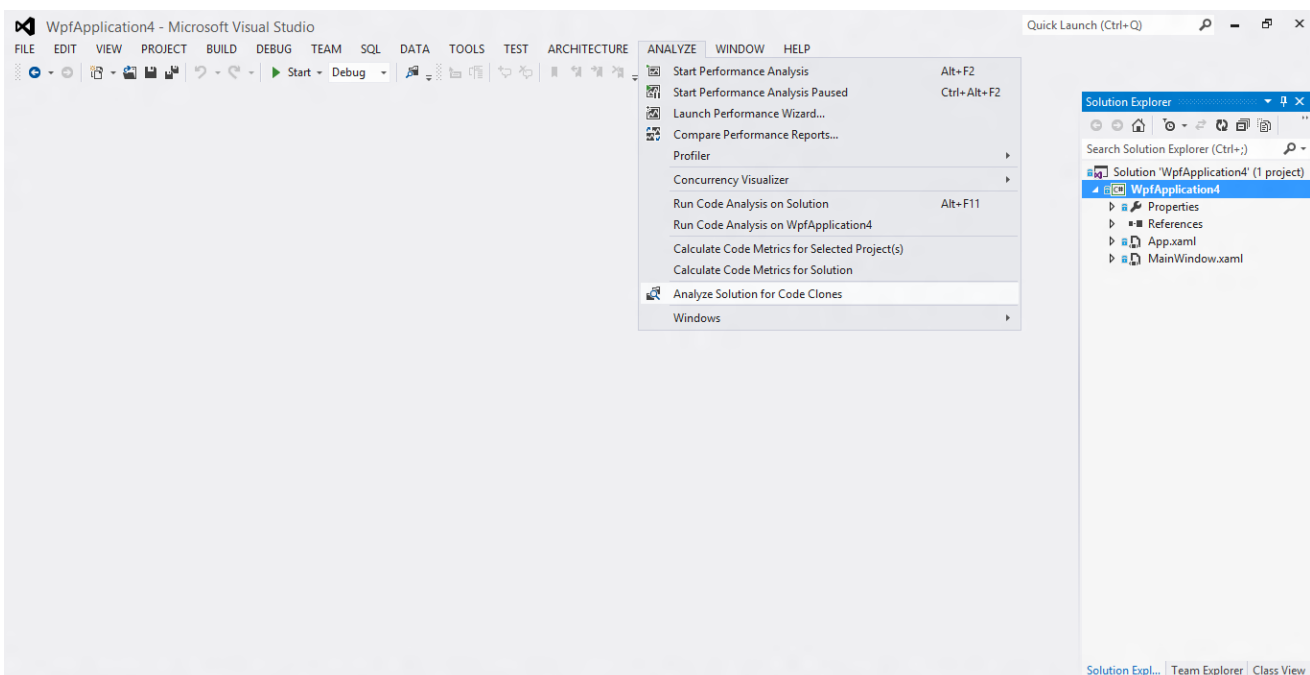


The developer can click the review work items to examine any comments that the reviewer made, and incorporate into the code any changes that result from the review.

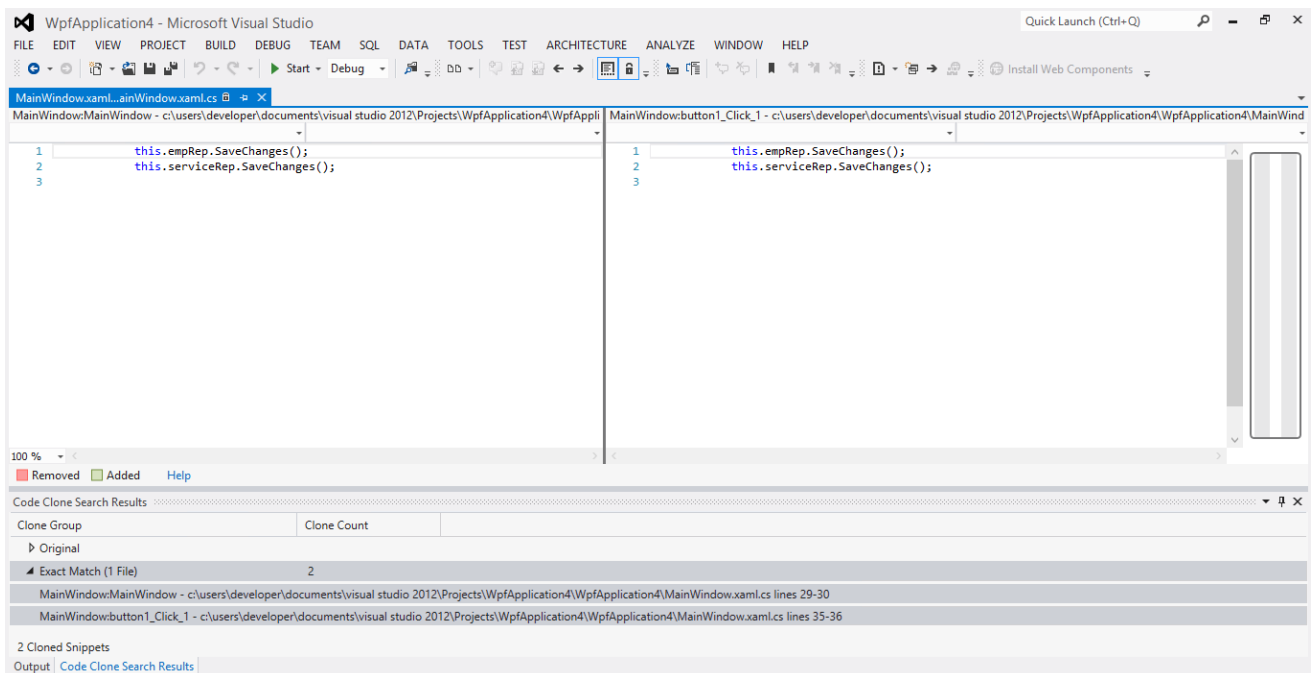
## Code-Clone Analysis

Visual Studio has historically provided tools that enabled a developer to improve code quality by refactoring code; for example, by extracting commonly occurring functionality into a method. However, this process depends on the developer to determine where such reusable code is likely to occur. Different developers working on the same project may not be aware of any such refactoring, resulting in duplicated code.

Another common phenomenon concerns the use of copy and paste; a developer may copy a chunk of code that performs some useful task from one project or file to another. If this occurs several times, it is possible that the same code is added to different parts of the same project. Furthermore, if the code that is copied contains a bug, the developer has probably duplicated this bug in multiple places. The result is code that takes longer to test and correct.

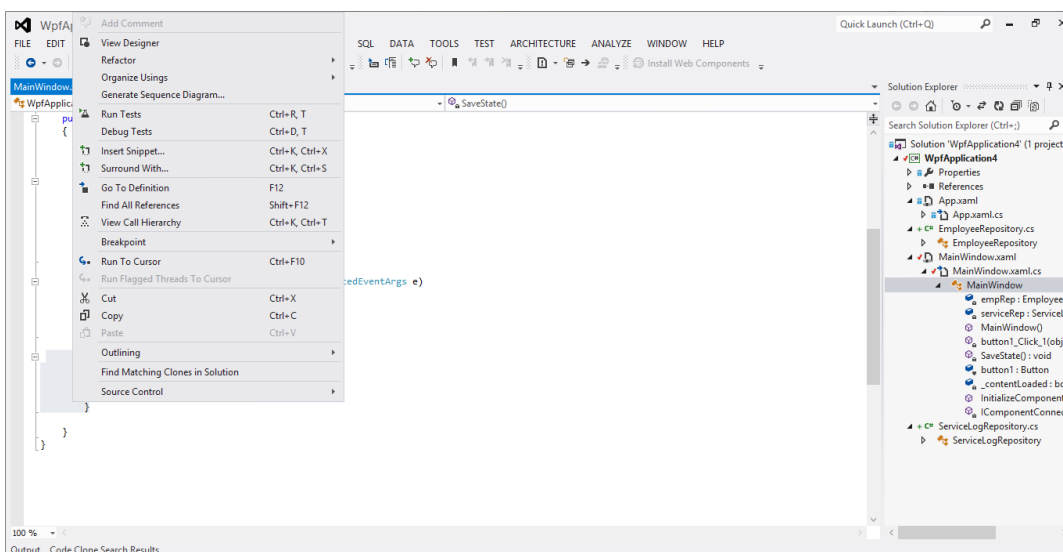


You can use the Code-Clone Analysis tool in a couple of ways. In the first scenario, the Code-Clone Analysis tool examines your



solution to look for logic that is duplicated, enabling you to factor this code out into one or more common methods.

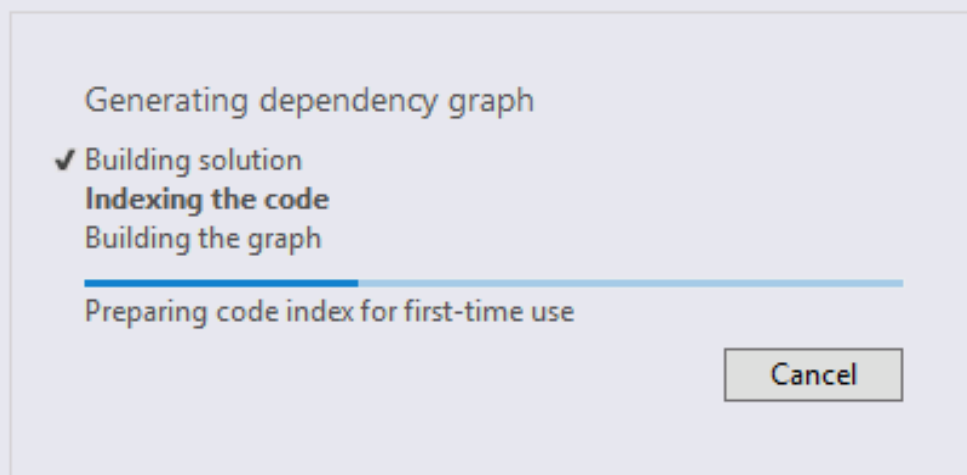
The Code-Clone Analysis tool is intelligent—it does not just search for identical blocks of code, it searches for semantically similar constructs by using heuristics.



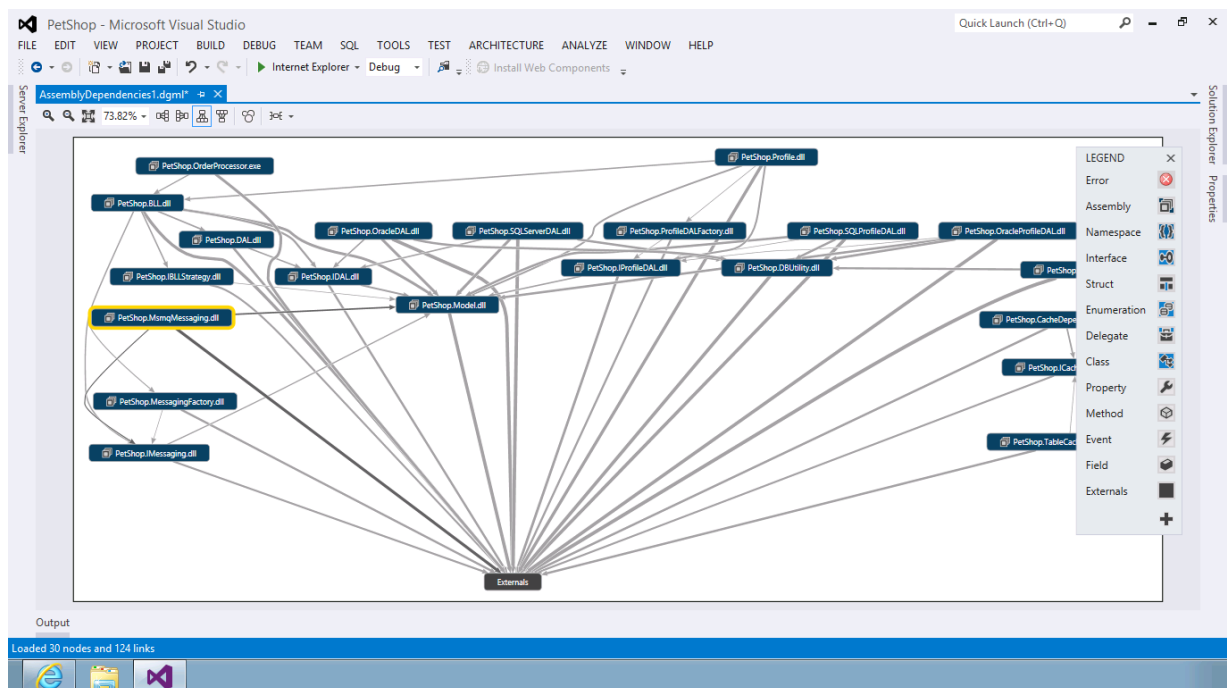
You can search through an entire solution to find all possible code clones, or you can highlight a block of code and have the Code-Clone Analysis tool find any code that is similar. This technique is useful if, for example, you are correcting a bug in a piece of code and you want to know whether the same bug resulting from a similar piece of code occurs elsewhere in the project.

Visual Studio includes strong architecture and code understanding tools that have been enhanced significantly in Visual Studio 2012. New capabilities include the following:

- Faster generation of code dependency graphs. Now dependencies are indexed, making it faster to create graphs and update them incrementally as your code changes and evolves.



- Graphs can now be created directly from solution explorer starting from any types or member and their related items. Creating a graph showing the related items and their parent container is now trivial. It even supports drag and drop from solution explorer to the dependency map.



- Full support for C or C++ projects for dependency diagrams, Architecture Explorer and layer and dependency validation.
- Full support for editing and annotating dependency diagrams graphically, including rearrangement, adding new nodes, deletion and comments.
- Once the graph is created, it can now be searched so a developer can easily hone in on specific sections in complex diagrams. It can also be panned and zoomed to focus on specific sections in greater relief.

Additionally, Unified Modeling Language (UML) tools have been integrated into Visual Studio 2012 that enhance visualization and modeling tasks such as generating code from models and creating UML class diagrams (reverse engineering) from existing code.

Notably, dependency graphs and layer diagrams continue to be extensible. In addition new architecture contextual commands in the solution explorer and architecture diagram also now have extension points.

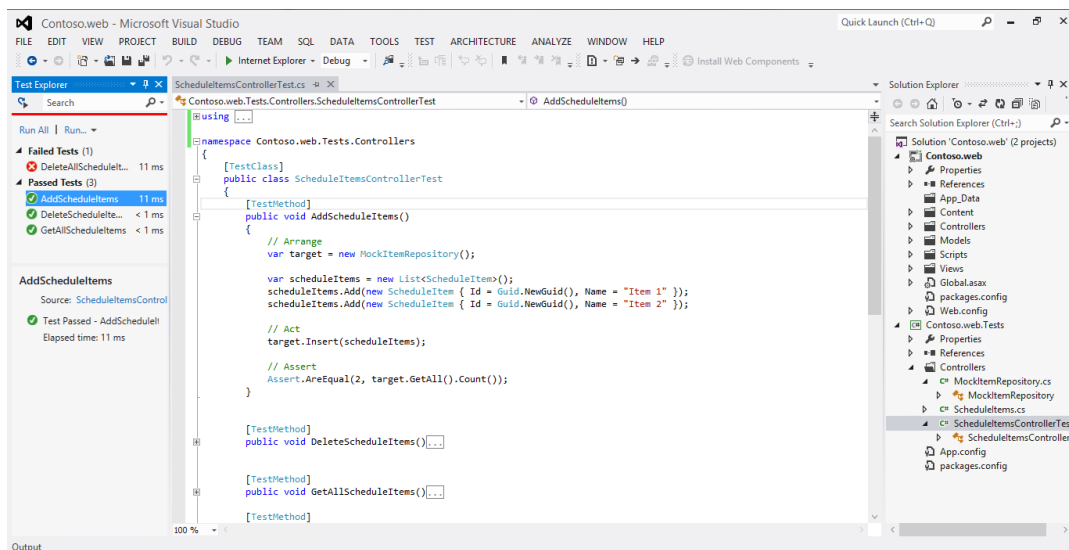
Architecture tools in Visual Studio help you get your software projects off to the right start, enable new developers to organically discover the project details in a shorter timeframe, and help you control technical debt by enforcing architecture restrictions for new code being checked into the project.

## Continuous Unit Testing

Visual Studio 2012 provides a unit test runner that can continuously execute unit tests in the background while you are working on your code. You can see the results immediately, so you do not have to pause and wait for test results. Test impact analysis ensures that the most important tests are run first, so any failures appear in seconds.

In Visual Studio 2012, we have updated the unit testing capability and made it easier to work with third-party unit testing frameworks such as NUnit and xUnit. This capability is extensible.

You can integrate unit testing seamlessly into your projects. The Unit Test Explorer in Visual Studio 2012 enables you to manually run the tests that are defined for your project and view the results.

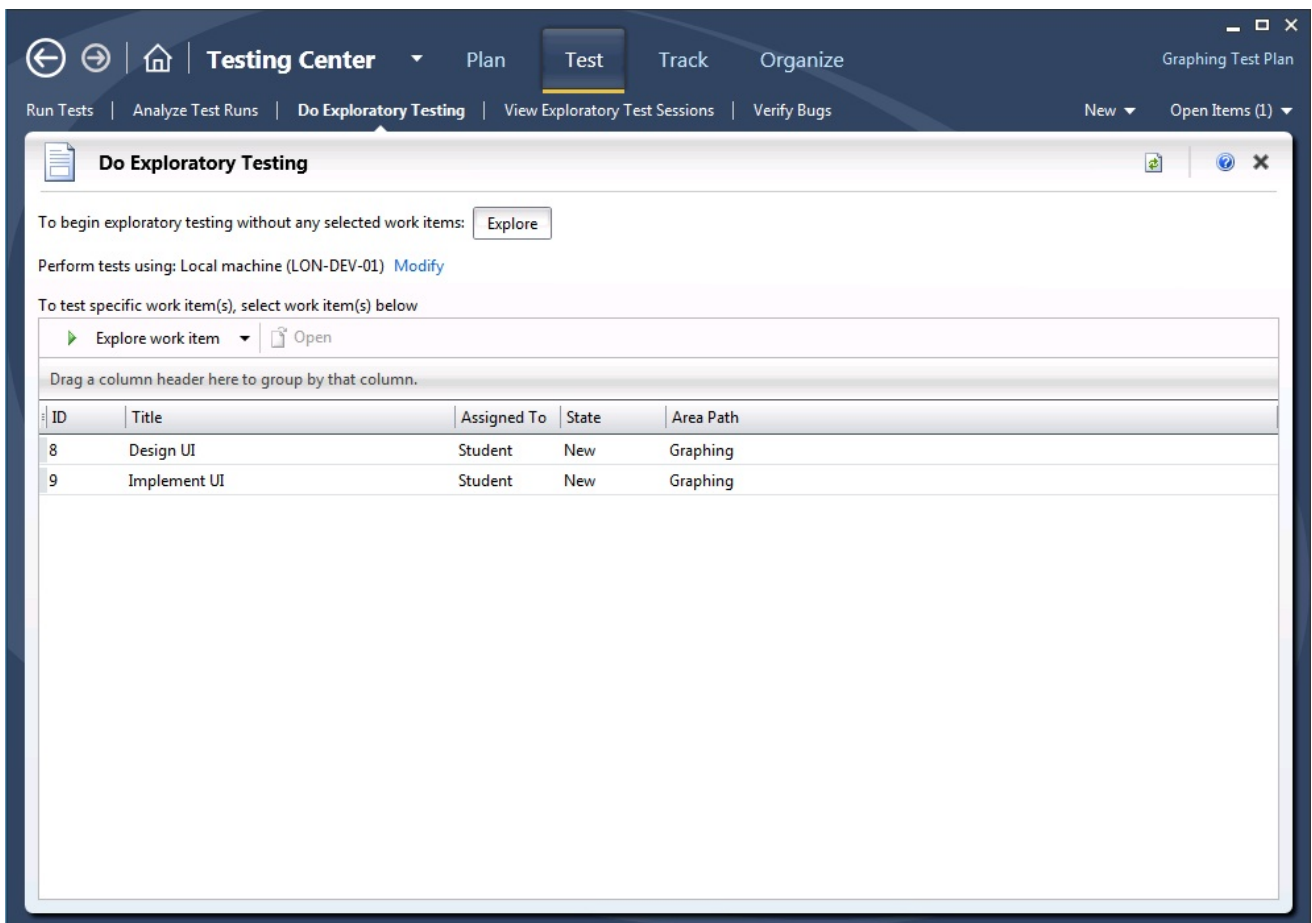




For tests that fail, Visual Studio 2012 displays the reason for the failure and a stack trace. You can click the link to the failing line of code and ascertain the cause, make any necessary corrections, and run the unit tests again.

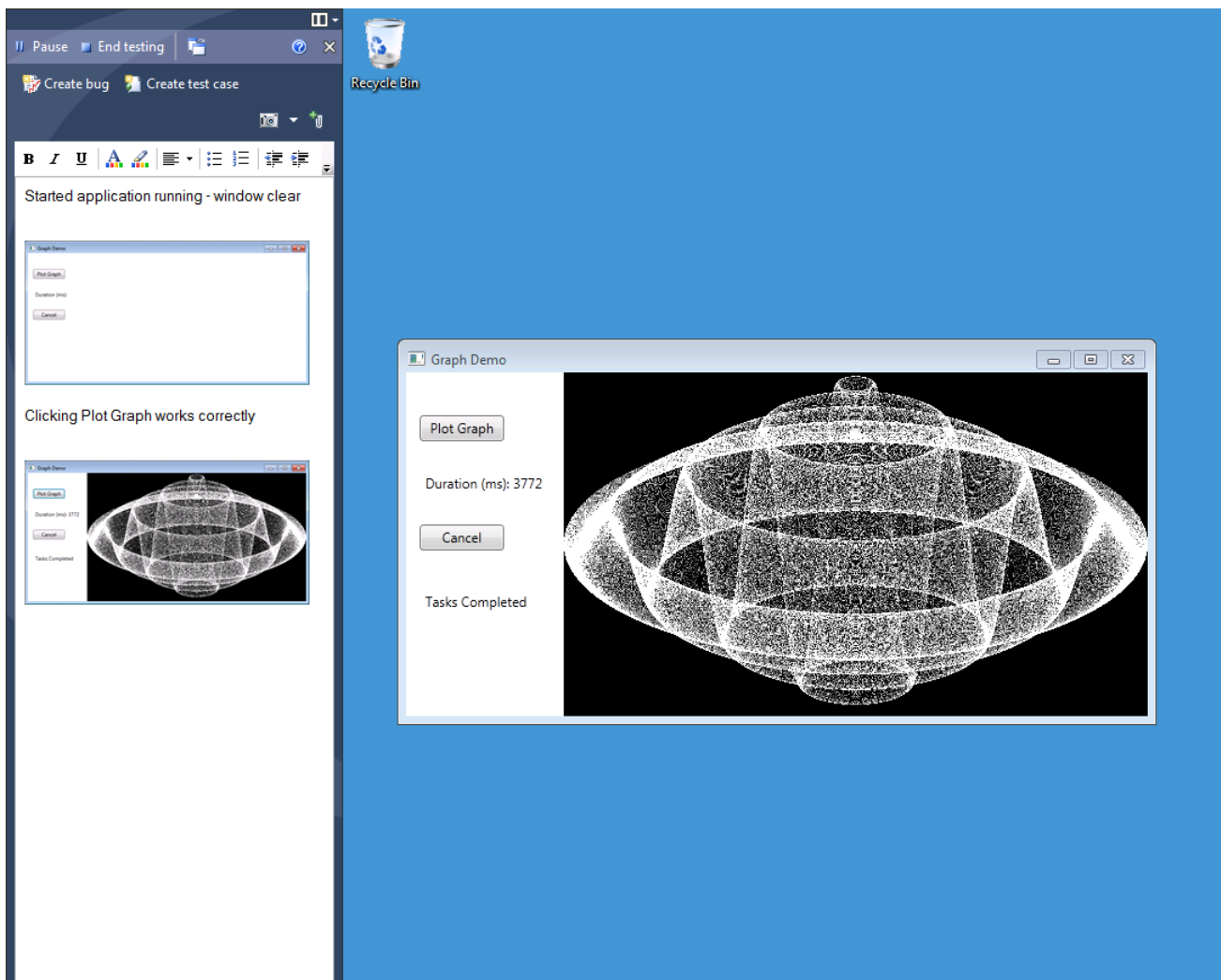
## Performing Exploratory Testing

Unit testing is a planned operation based on a specific and well-defined set of tests. However, the most revealing bugs are frequently discovered as the result of unanticipated exploration of an application. To support this method of working, Microsoft Test Manager enables you to perform exploratory testing and to record any bugs that occur.

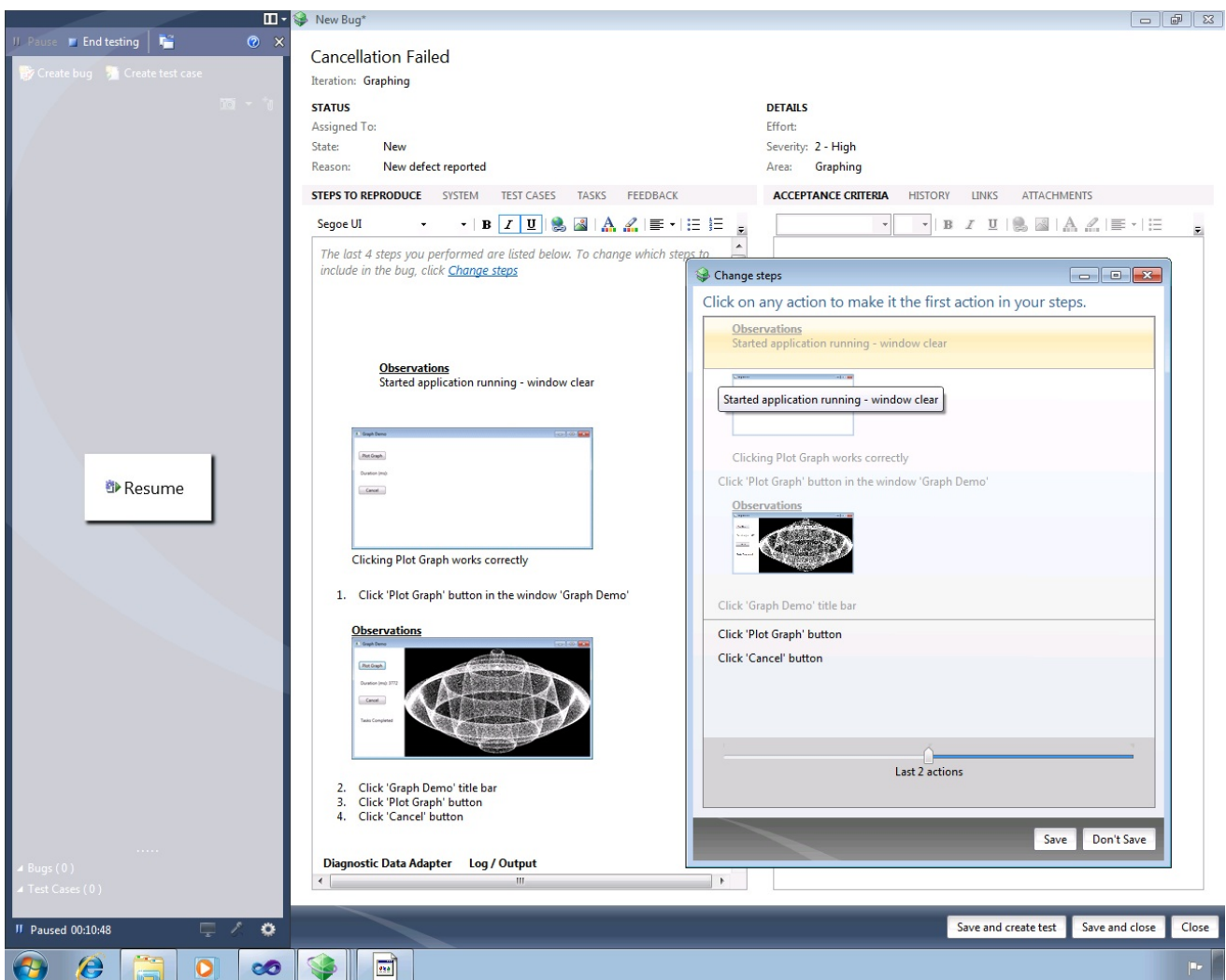


The Exploratory Testing tool is ideal for implementing Agile testing. Specifically, it enables you to:

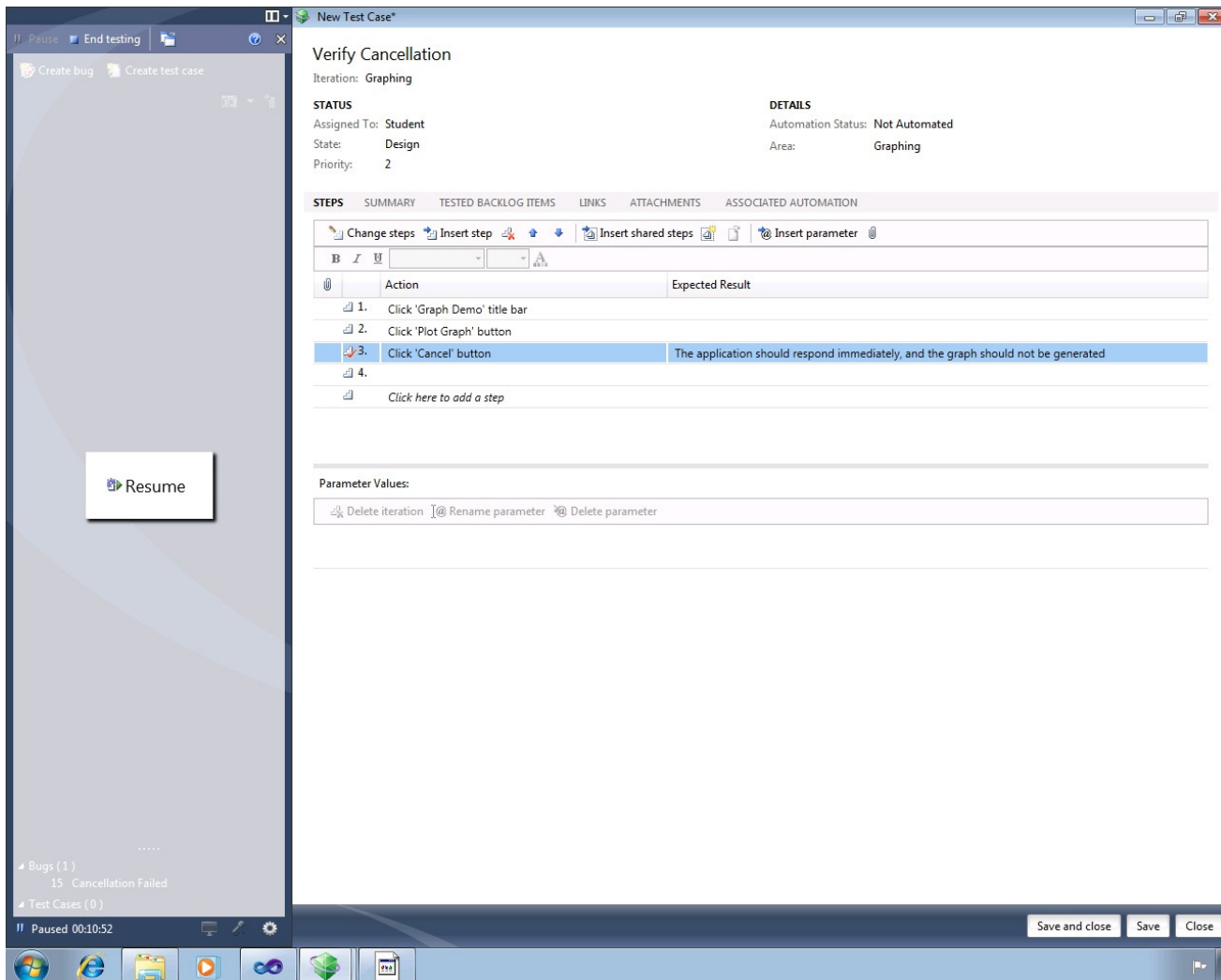
- Test without performing formal test planning. You can now start directly testing the product without spending time writing test cases or composing test suites. When you start a new testing session, the tool generates a full log of your interaction with the application under test. You can annotate your session with notes, and you can capture the screen at any point and add the resulting screenshot to your notes. You can also attach a file to provide any additional information that may be required.



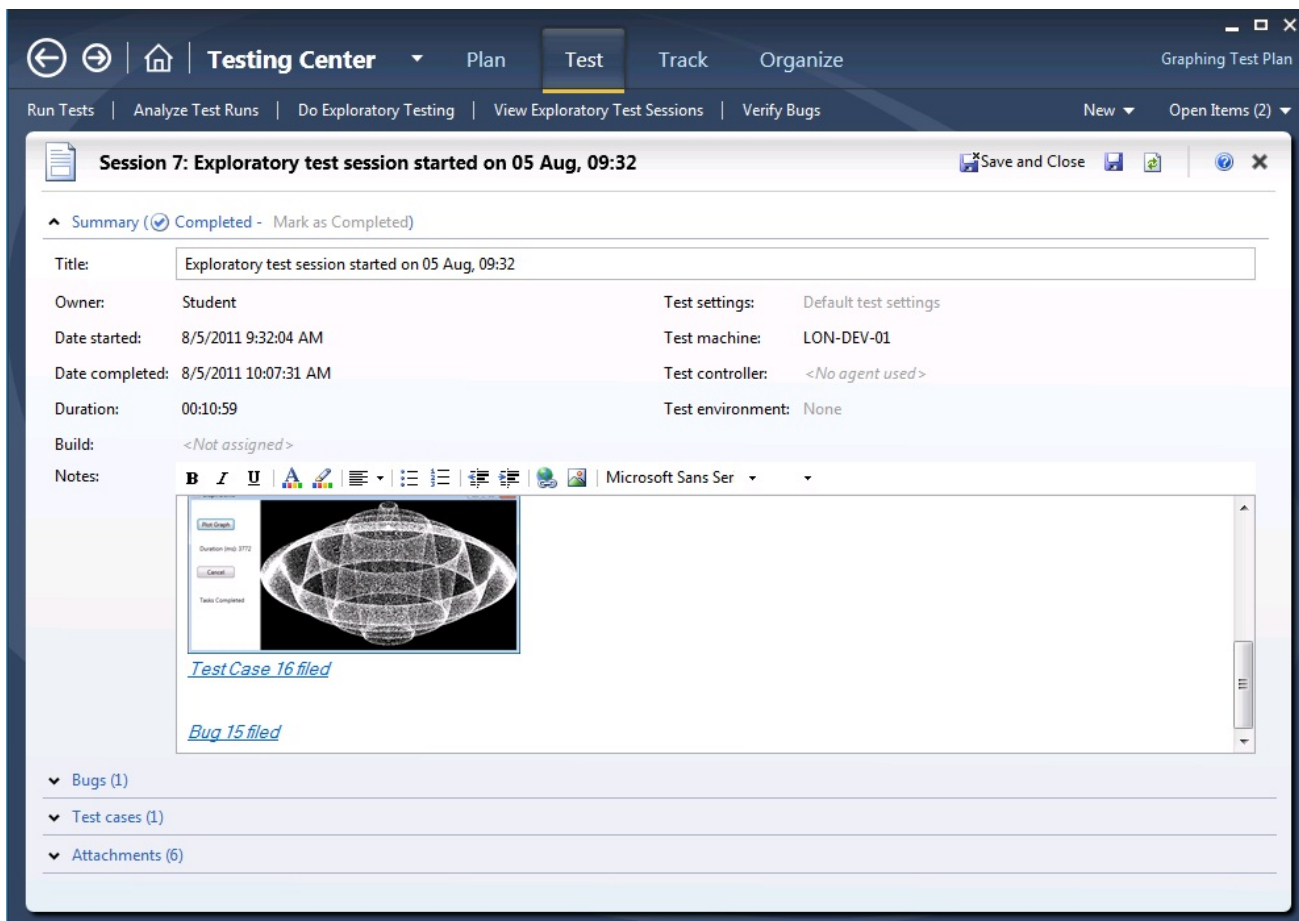
- File actionable bugs fast. While you are running the application, you may encounter some unexpected behavior. The Exploratory Testing tool enables you to generate a bug report, and the steps that you performed to cause this behavior are automatically included in the bug report. If necessary, you can edit these steps or reorder them before you save the bug. You can also link the bug to a task in Team Explorer, and then assign this task to a developer.



- Create test cases. In addition to recording bugs, you can generate test cases that are based on the steps that caused the bugs to appear. You can edit the steps, reorder them, and insert additional ones. This feature can help the test team to verify that a bug is no longer present when the developer updates the code. These test cases are stored in the Test Manager repository.



- Manage exploratory testing sessions. When testing is complete, you can return to Microsoft Test Manager, which saves the details of the testing session and includes information such as the duration, which new bugs were filed, and which test cases were created.



## Providing Customer Feedback

Your customers and end users are the reason that you build software. Therefore, obtaining their feedback is critical to the success of the project. With Visual Studio 2012, we have implemented the Feedback Manager tool, which enables these key stakeholders to provide timely and actionable feedback to the development team. Actionable feedback includes not just text comments but also videos, screenshots, and audio annotations that help the user to highlight any specific issues. These items help the developer to understand the context in which the user was performing the actions when a problem occurred or a feature was found to work incorrectly.

Using the Feedback Manager, the user can run the application and record the ways in which the application is used. The user can also take screenshots or record audio and video at any point, perhaps providing a running narrative of the tasks that he or she is performing.

The screenshot shows the Visual Studio Feedback Manager window. At the top, there are three buttons: "Start", "Provide" (highlighted in blue), and "Submit". Below these buttons is a label "ITEM: 1" with a blue square containing the number "1". The main area is titled "User Experience" and contains a large empty text box. Below the text box is a horizontal bar with four icons: a microphone, a screen, a camera, and a paperclip. Below these icons are four labels: "Screen with Voice", "Voice only", "Screenshot", and "Attach file". Below these labels are two buttons: "Record" and "Attach". Below the buttons is a status bar that says "Not recording" and a "Stop" button. Below the status bar is a rich text editor with a toolbar containing bold, italic, underline, text color, background color, bulleted list, numbered list, link, and unlink buttons. The text editor contains the text "Screen Recording 1 (00:38)" and "App looks OK. Some of the fonts ay be difficult to read for people who are color blind." Below the text editor is a rating section that says "Rate this item:" followed by five stars. At the bottom are two buttons: "Previous" and "Next".

Start Provide Submit

ITEM: 1

User Experience

.....

Screen with Voice Voice only Screenshot Attach file

Record Attach

Not recording Stop

**B I U** | A | [Color Picker] | [Background Color] | [Bulleted List] | [Numbered List] | [Link] | [Unlink]

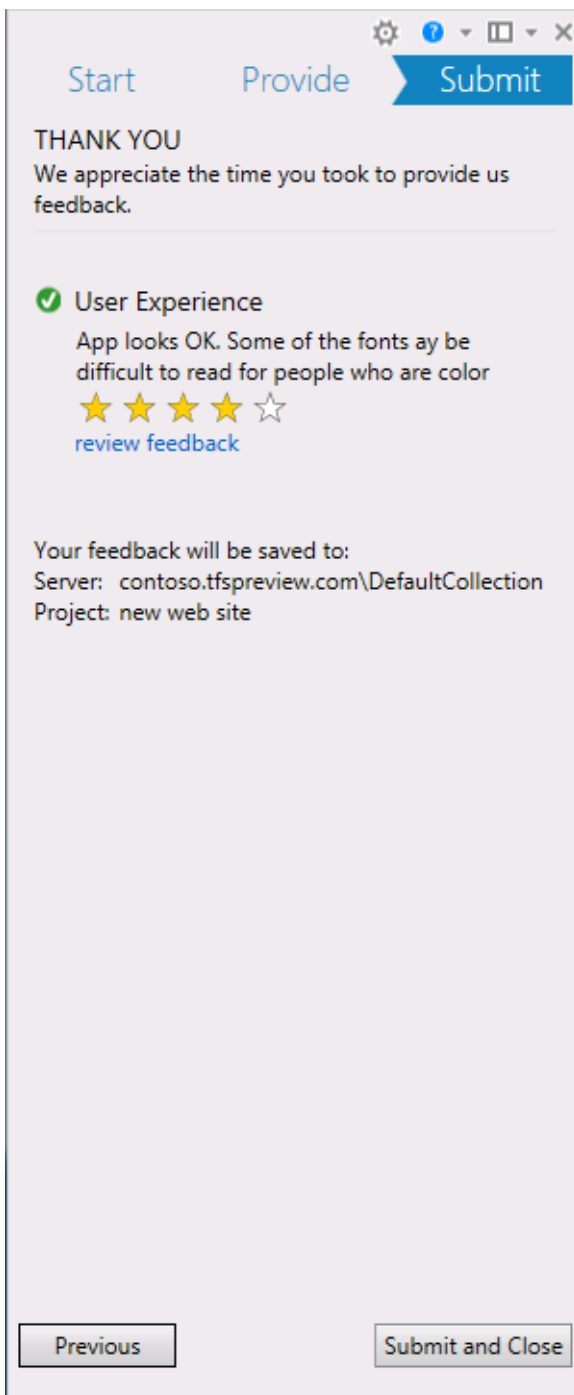
Screen Recording 1 (00:38)

App looks OK. Some of the fonts ay be difficult to read for people who are color blind.

Rate this item: ☆ ☆ ☆ ☆ ☆

Previous Next

The Feedback Manager records the user's interactions with your application and logs keystrokes, mouse clicks, and other events. When the user has finished the feedback session, clicking **Submit** saves the feedback information to the Visual Studio Team Foundation Server 2012 database. The user also can provide a rating and comments for the application.



The screenshot shows the 'Submit' dialog box of the Visual Studio Feedback Manager. At the top, there are three tabs: 'Start', 'Provide', and 'Submit' (which is active and highlighted in blue). Below the tabs, the text reads 'THANK YOU' followed by 'We appreciate the time you took to provide us feedback.' Below this is a section titled 'User Experience' with a green checkmark icon. The feedback text is 'App looks OK. Some of the fonts ay be difficult to read for people who are color'. Below the text is a five-star rating system with four yellow stars and one grey star. A link 'review feedback' is present. At the bottom, it states 'Your feedback will be saved to:' followed by 'Server: contoso.tfspreview.com\DefaultCollection' and 'Project: new web site'. At the very bottom, there are two buttons: 'Previous' and 'Submit and Close'.

Start Provide **Submit**

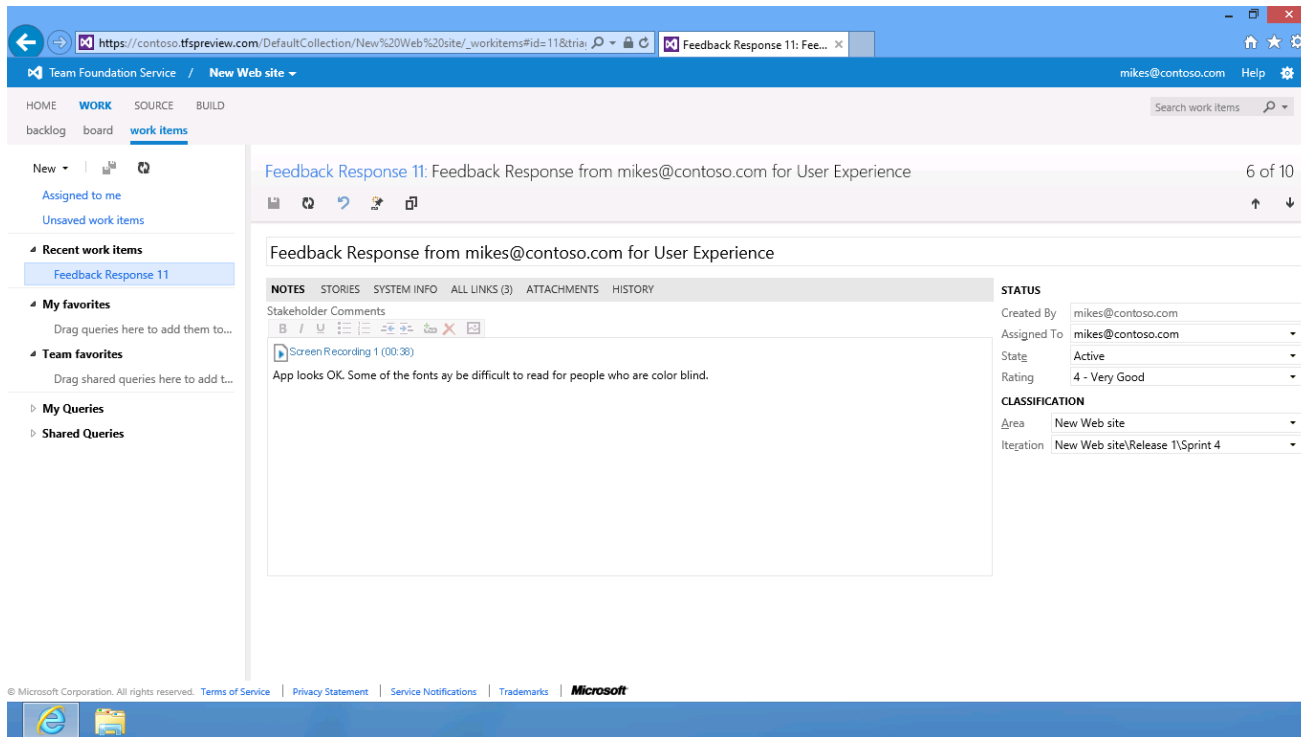
**THANK YOU**  
We appreciate the time you took to provide us feedback.

✓ **User Experience**  
App looks OK. Some of the fonts ay be difficult to read for people who are color  
★★★★☆  
[review feedback](#)

Your feedback will be saved to:  
Server: contoso.tfspreview.com\DefaultCollection  
Project: new web site

Previous Submit and Close

The product owner in the development team can assess this feedback, generate the appropriate work items and tasks that include the details that the user provided, and assign them to a developer.

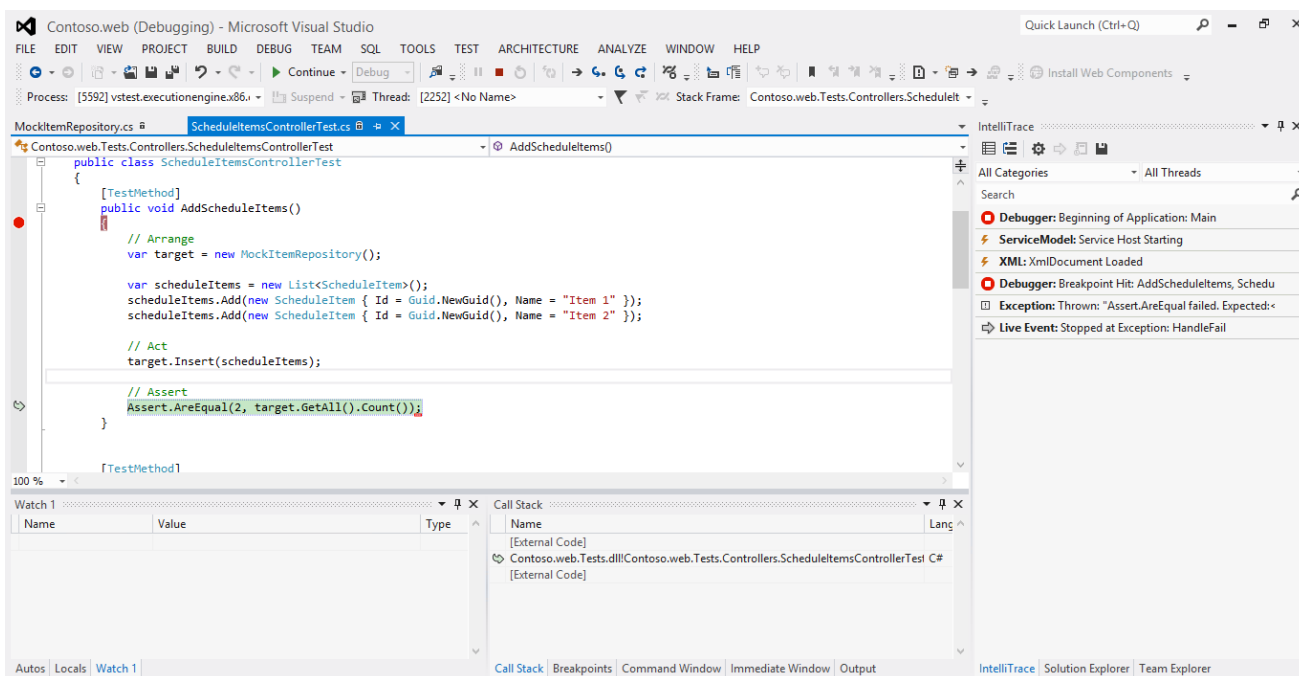


## Getting Feedback from Applications in Production

If your applications running in the live environment display unusual behavior or bugs, the operations staff can use IntelliTrace in production to capture the details of that behavior and report it back to the development team. Using IntelliTrace in this manner enables the operations staff to provide data that gives the developers greater insight into the causes of bugs, and it can greatly reduce the time it takes to fix such bugs in a production application.

IntelliTrace in production is an established debugging tool that records the sequence of events that led to the unexpected behavior. The information that IntelliTrace captures includes diagnostic data, details of exceptions, the call stack, and a trace of where the code failed. The data that IntelliTrace captures can be stored and then replayed by a member of the development team by using Visual Studio 2012. A developer with Visual Studio 2012 can use this information to debug the application as if it was running live on their development computer.



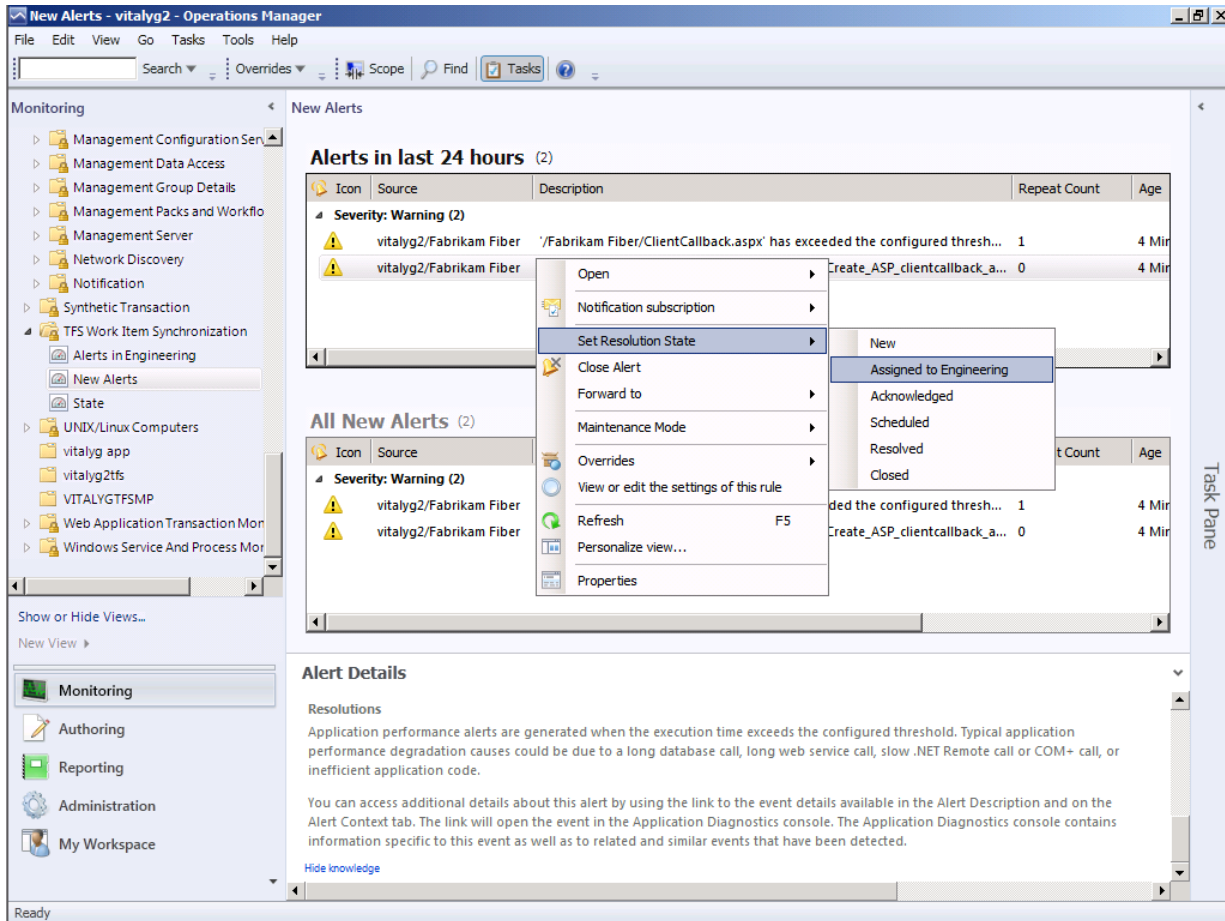


Visual Studio also includes **PreEmptive Analytics** that provide real-time feedback solutions for applications in production. This is perfect for situations in which your application is fully deployed with customers, with a limited access to each individual instance in order to debug it. This is very common for client-side applications and mobile applications. PreEmptive Analytics provides a system to send unhandled exceptions and diagnostic data to an end point from where it can be aggregated automatically as work items in TFS. Your team will have complete visibility into all unhandled exceptions for your customers (who have Internet access), with bugs filed automatically to be followed up and fixed for the next release.

## Working Closely with Operations Personnel

Visual Studio provides a server monitoring solution for teams that use or want to adopt System Center Operations Manager (SCOM). Visual Studio can deploy a monitoring agent for ASP.NET applications running on a web server. This agent collects rich data about exceptions, performance issues, and other errors. Using the Work Item Synchronization for TFS, operations staff can file these exceptions as work items in TFS and assign them to developers to investigate in order to improve and fix production web applications. Visual

Studio and the Work Item Synchronization for TFS, in conjunction with SCOM, provide a real-time improvement feedback loop for server-based applications deployed in production, leading to continuous improvements and high quality.



## Visual Studio Team Foundation Server 2012 in the Cloud

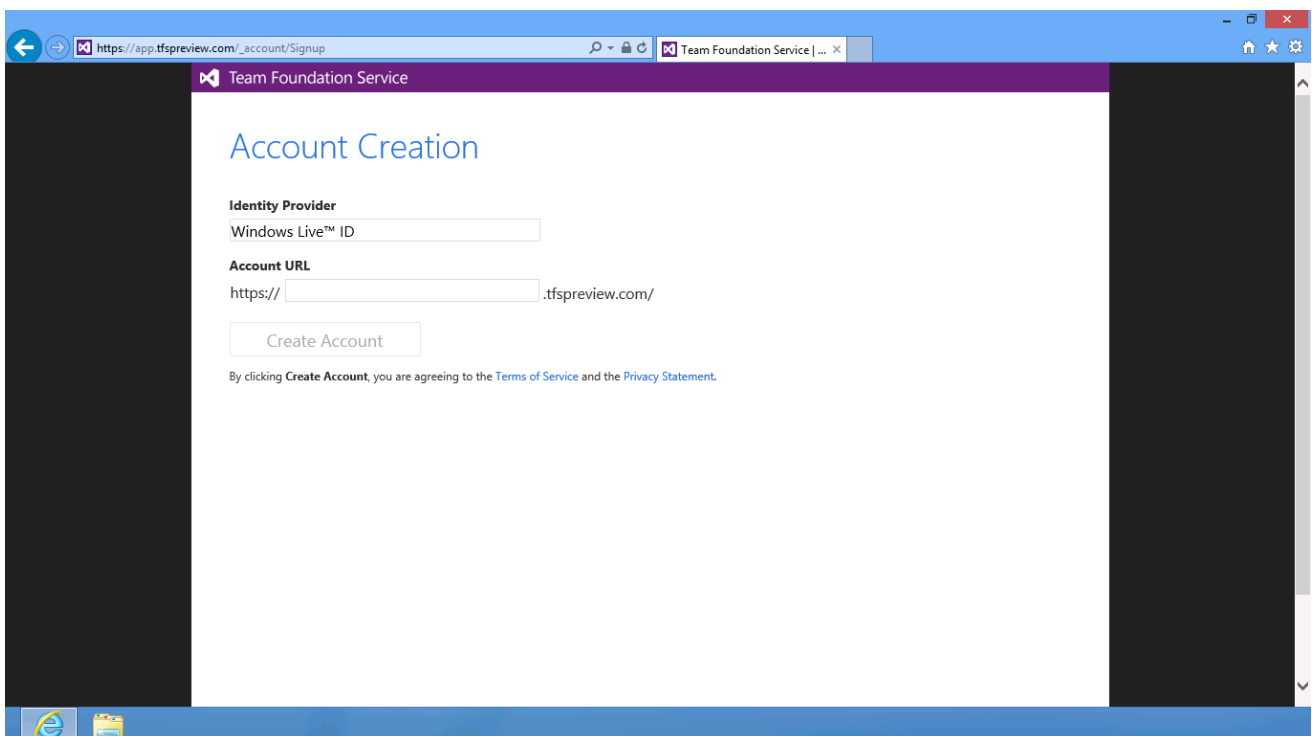
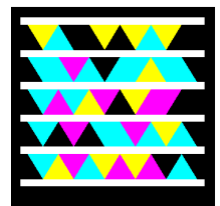
Visual Studio Team Foundation Server 2012 provides a powerful repository for storing project data, but as with all server-based software, there are costs associated with installing, configuring, and managing it. Additionally, as an increasing number of projects take advantage of its features, you may need to scale up the hosting platform to support the additional workload.

Furthermore, as Visual Studio Team Foundation Server 2012 becomes a critical part of the development infrastructure, you must be prepared to maintain it, including implementing an appropriate disaster-recovery strategy. These requirements can add up to a significant investment in time and money.

To address these issues, we have made Visual Studio Team Foundation Server 2012 available in the cloud, running under Windows Azure.

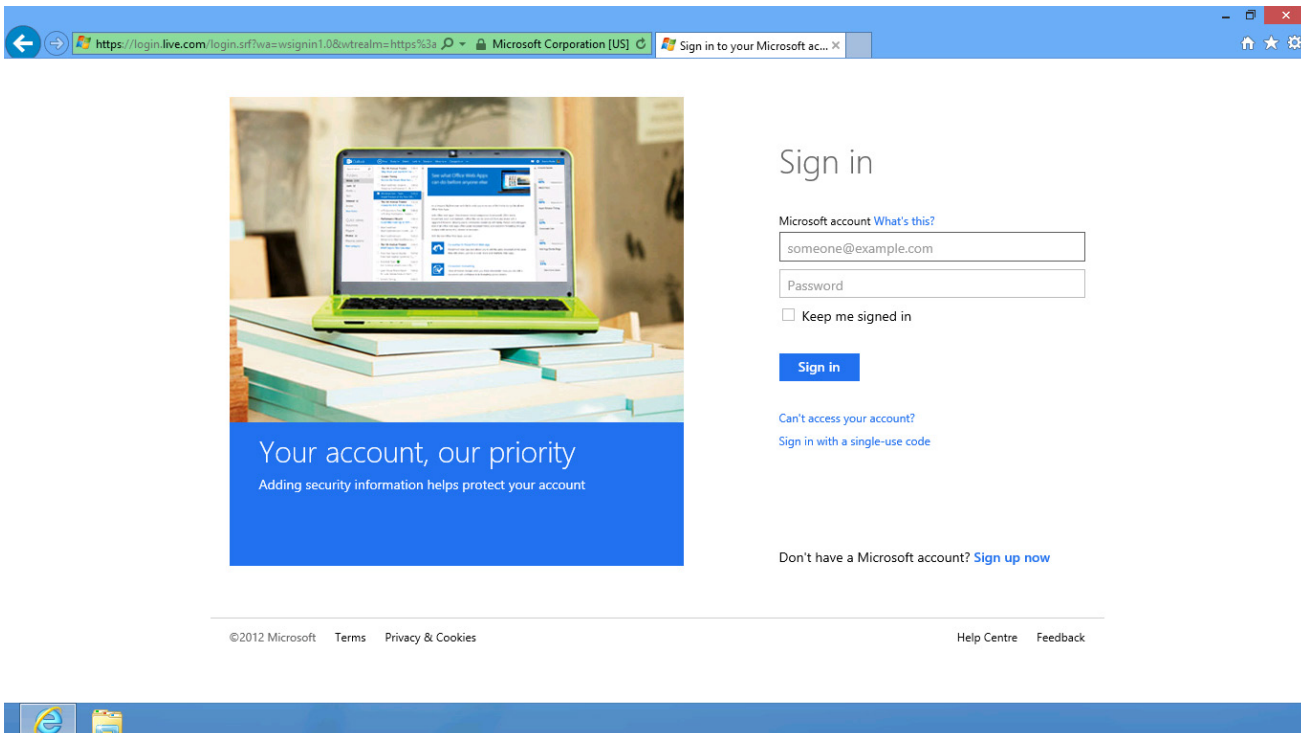
### Simple Sign-Up and Immediate Web Access

Visual Studio Team Foundation Service features a friction-free signup experience that enables teams to create and set up a complete Agile Application Lifecycle Management system in a matter of seconds. By using a web browser and providing an e-mail address, you can get your team up and running quickly.



You need to specify a URL for your instance of Visual Studio Team Foundation Server 2012, provide an invitation code, accept the Terms of Service, and click **Create Account**. Within seconds, your account is created, and the Agile Application Lifecycle Management system is set up and available for immediate use.

You can then enroll your team members and add them to the service. It is hosted in the cloud, so Visual Studio Team Foundation Service is available to all team members regardless of location—they simply need to sign in and provide their credentials.



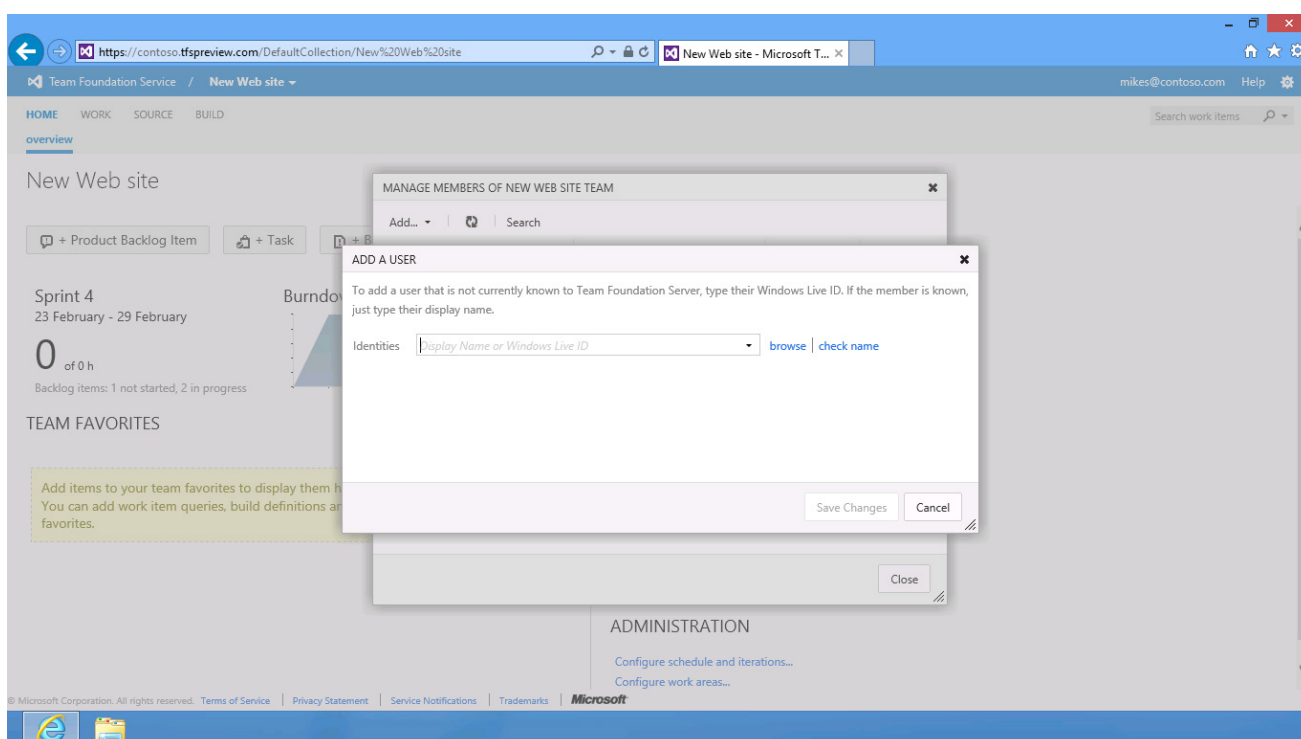
Consequently, this solution is ideal for geographically dispersed teams. If you have a network connection, you can access Visual Studio Team Foundation Service from anywhere.

## Easy Collaboration with External Contributors

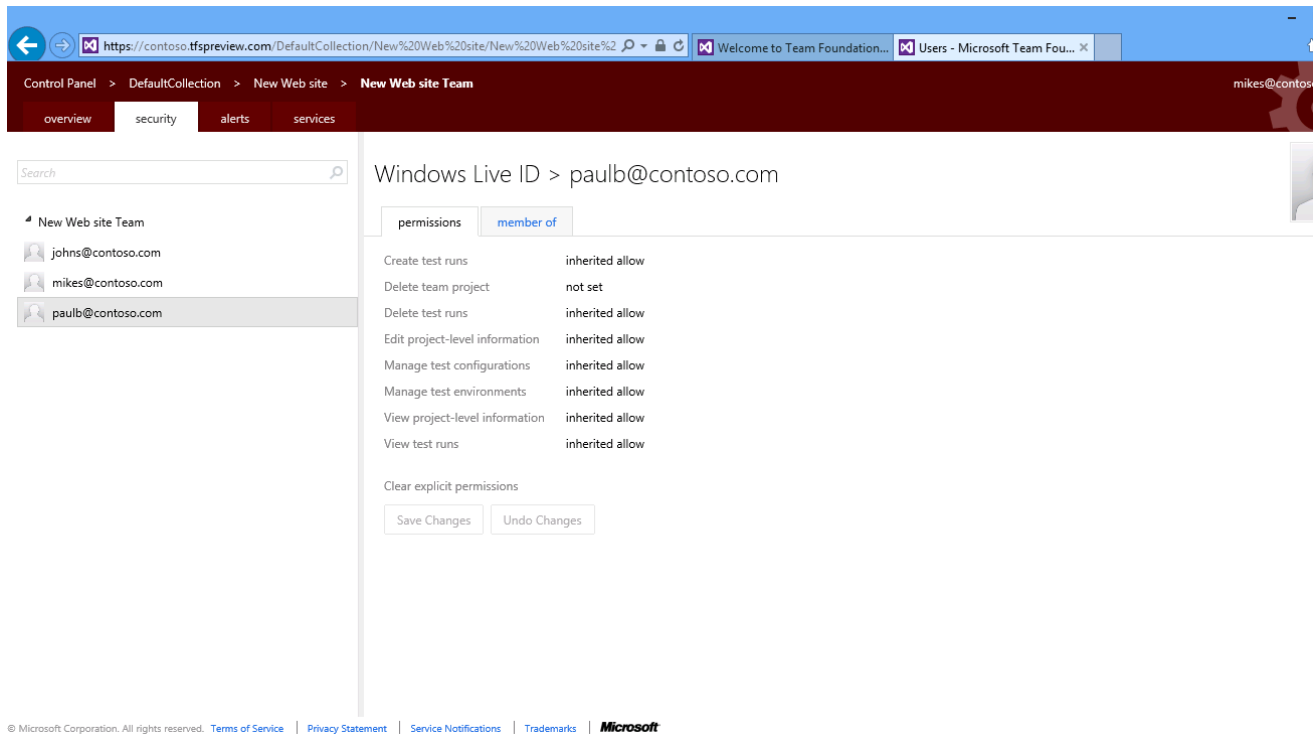
Collaboration and communication are essential elements in any successful software development project. At different phases of the project, you may need contributions from users or specialists who are external to your

company or located in different locations around the world. Historically, providing system and network access to these external stakeholders and contributors has been a challenging task that involved either creating those users as local users on your network and providing virtual private network (VPN) access, or setting up an instance of the server in a perimeter network. These tasks require skilled security professionals and, in many cases, dedicated infrastructure and rigorous security management and policies.

Even if you are fortunate enough to have access to the appropriate resources to enable this scenario, it still takes time. With Visual Studio Team Foundation Service running in the cloud, you can provide access to the system in seconds, and Microsoft takes on the burden of infrastructure security and monitoring. In the User Administration portal, you simply enter the e-mail addresses of the persons to whom you want to give access.



You can then specify the projects that they can access and the operations that they can perform.



## No Infrastructure Management

With Visual Studio Team Foundation Service in the cloud, you can spend your time building great software instead of managing hardware infrastructure or updating software. As the administrator of Visual Studio Team Foundation Service, you do not have to concern yourself with tasks that do not add value, such as performing operating system or software upgrades, making backups, patching the server, or responding in the middle of the night to hardware or power failures. Instead, Microsoft takes on that burden.

You have guaranteed access to world-class data centers and network infrastructure with the peace of mind that Microsoft ensures that you are always using the latest release of the software, that the system incorporates world-class security, that the system is protected, and that your data is backed up. You are free to focus on what is most important to you: building software that delights your customers.

## Support for Heterogeneous Environments

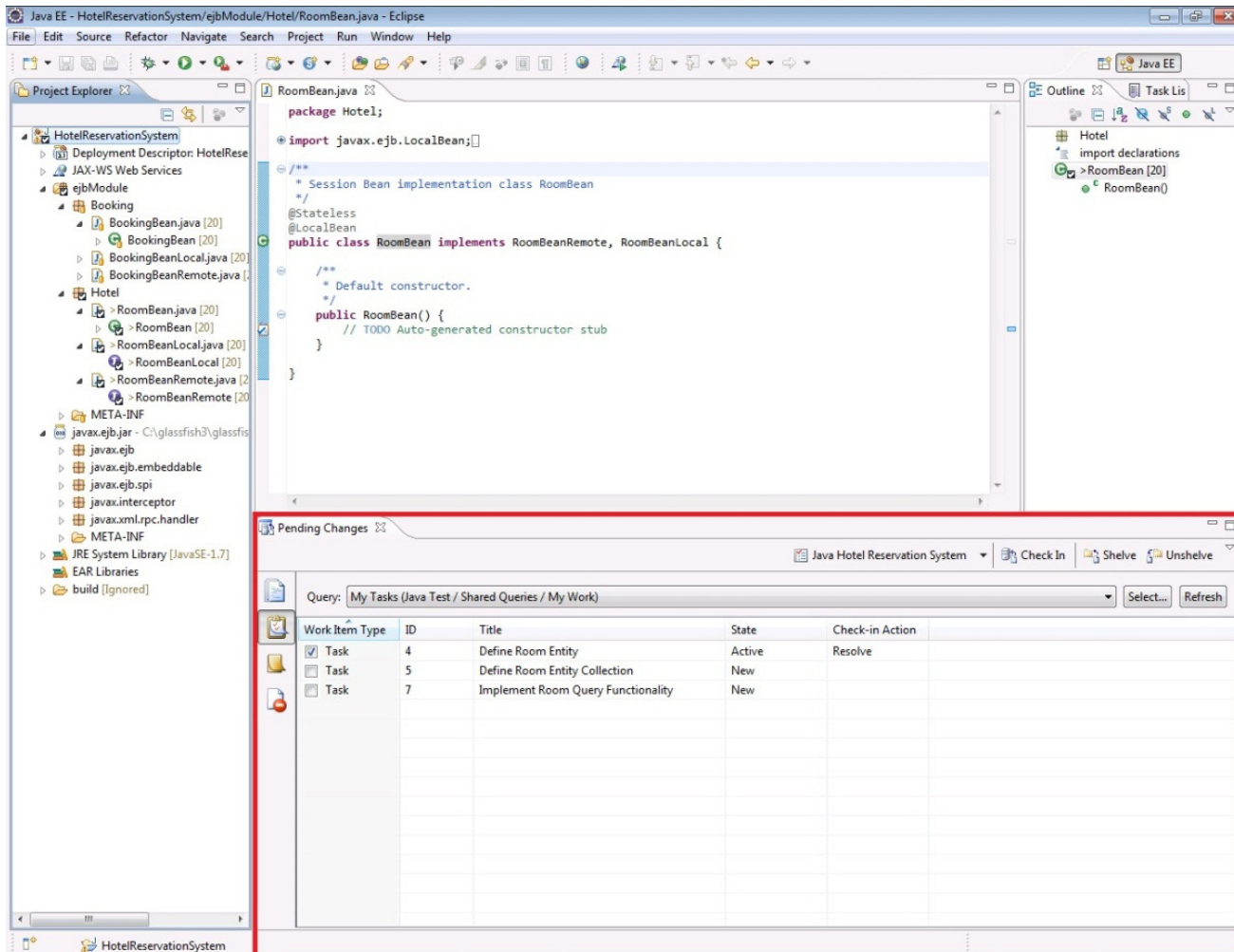
Although Visual Studio 2012 is the ideal partner for Visual Studio Team Foundation Server 2012, it is not the only development environment in the world. Many organizations use other popular IDEs that are based on Eclipse, and an important feature of Visual Studio Team Foundation Server 2012 is its ability to integrate with these tools, enabling developers to continue to use a familiar environment.

In a similar vein, although Hyper-V is the preferred platform for provisioning and managing virtual machines to test your applications, a number of enterprises have invested in alternative private cloud IT infrastructures such as VMware®. Visual Studio 2012 provides enhanced lab management capabilities that can integrate with the test infrastructure of your choice.

### Application Development

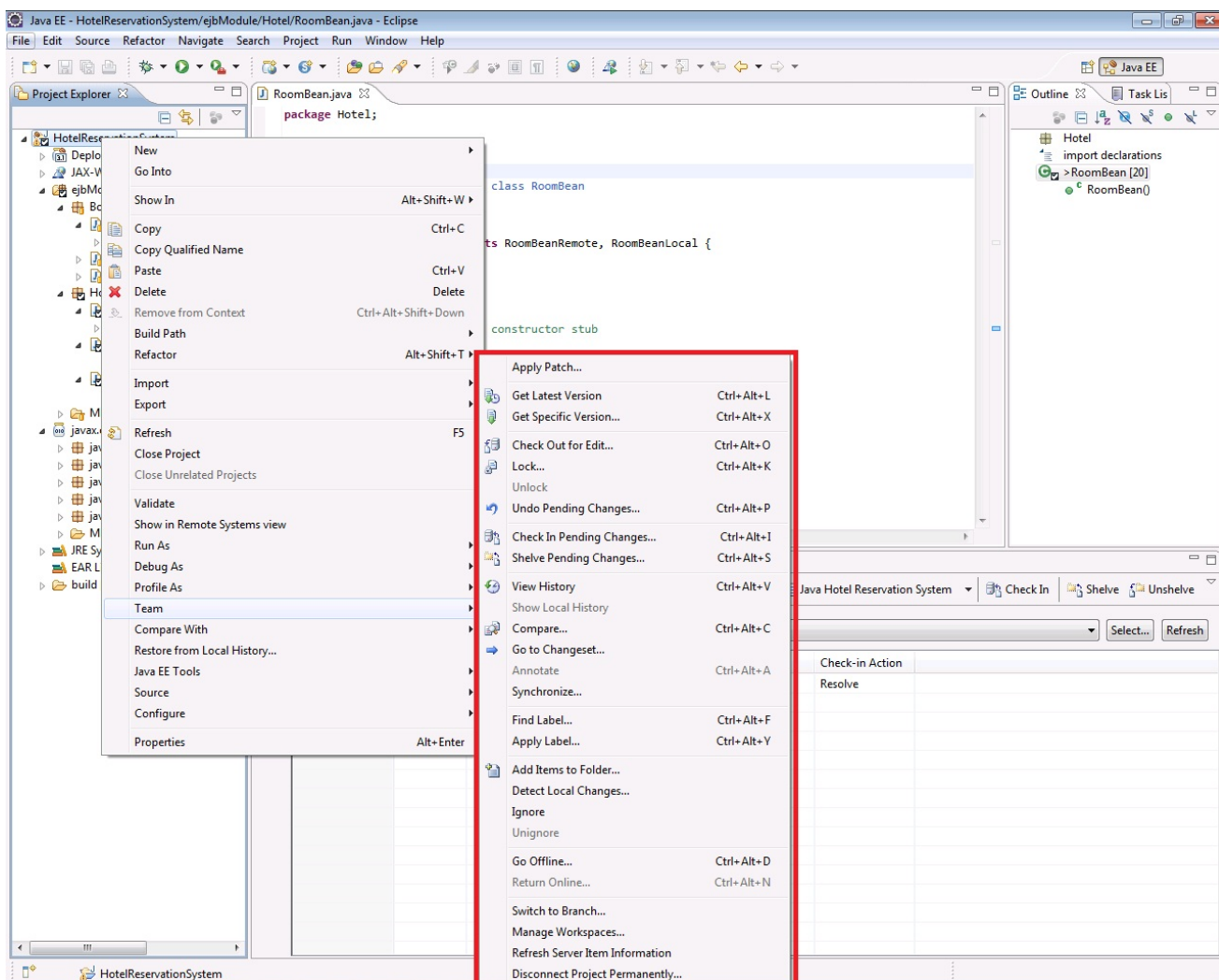
If you are familiar with an Eclipse-based IDE such as IBM® Rational® Application Developer, Adobe® Flex® Builder™, or Aptana® Studio, you can connect to Visual Studio Team Foundation Server 2012 and take advantage of its capabilities by using Team Explorer Everywhere.

Team Explorer Everywhere includes a plug-in for Eclipse that provides a significant subset of the facilities that are available to Visual Studio 2012. For example, you can view and edit your work items, employ version control over your application code, track bugs, generate reports, and get an up-to-date view of the entire project. The Team Explorer pane integrates directly into the Eclipse IDE.



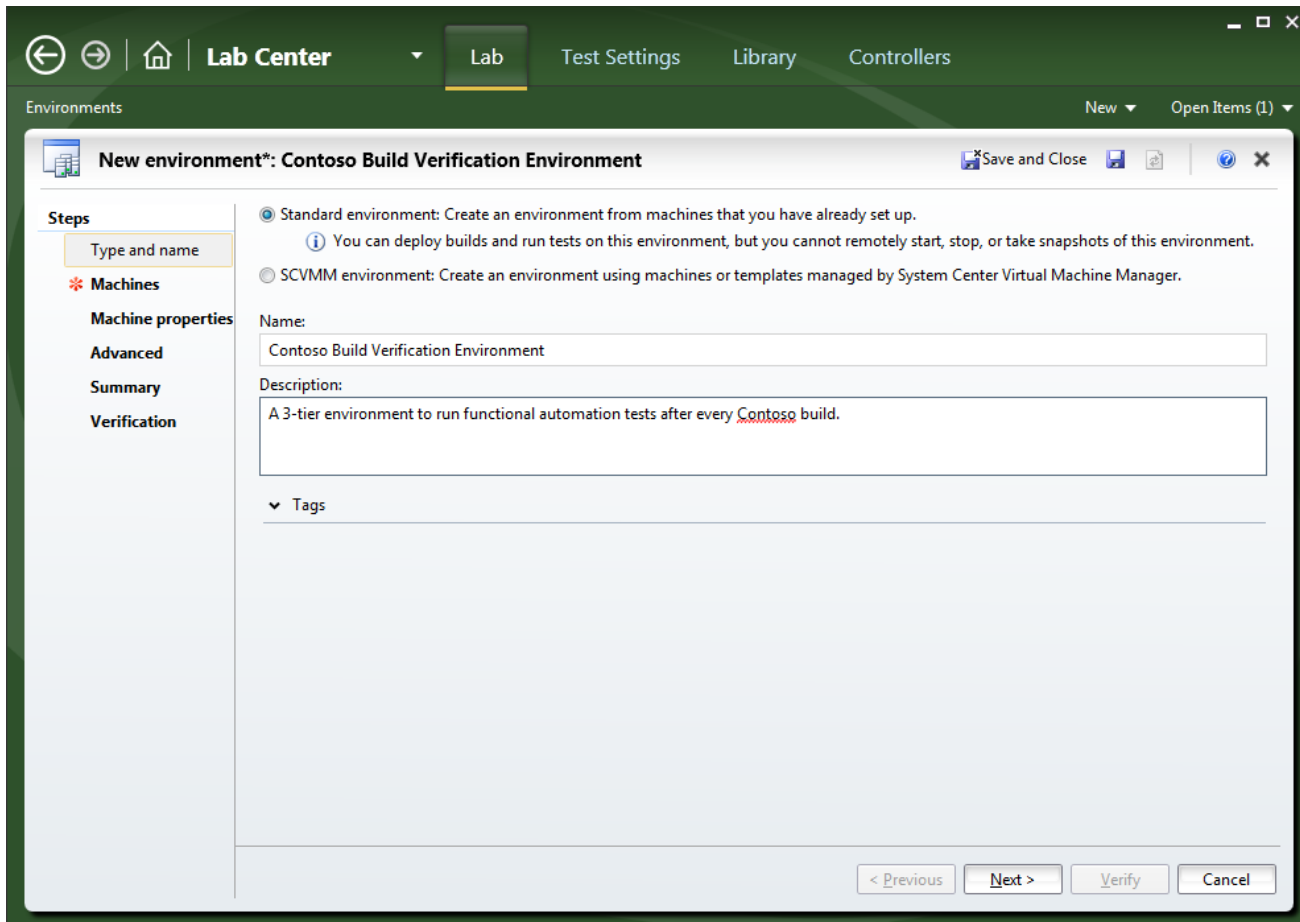
The plug-in also adds commands to the **Team** menu that support Team Foundation Server check-in, check-out, and version-control features.





## Lab Management

Visual Studio Team Foundation Server 2012 enables you to perform test lab management without requiring that you spend time setting up and configuring Microsoft System Center Virtual Machine Manager (SCVMM). Visual Studio Team Foundation Server 2012 provides zero-configuration, which you can use to create standard environments from Microsoft Test Manager by grouping the machines that you have provisioned in your private cloud infrastructure. You only need to run a single wizard to create your environment, and this wizard can create all of the necessary agents and automatically install and configure them within the test machines.



You can use standard environments as part of manual or automated testing and for build-deploy-test automation. To get additional benefits such as snapshots, you can continue to create environments manually by using SCVMM. Visual Studio Team Foundation Server 2012 supports both SCVMM 2008 R2 and SCVMM 2012, which has a number of enhancements to simplify the management of your test lab infrastructure.

Lab Workflow Parameters


Specify the environment where the application is deployed

Welcome  
Environment  
Build  
Deploy  
Test

Environment name: Contoso Build Verification Environment

☐ Revert to a specific snapshot of the environment

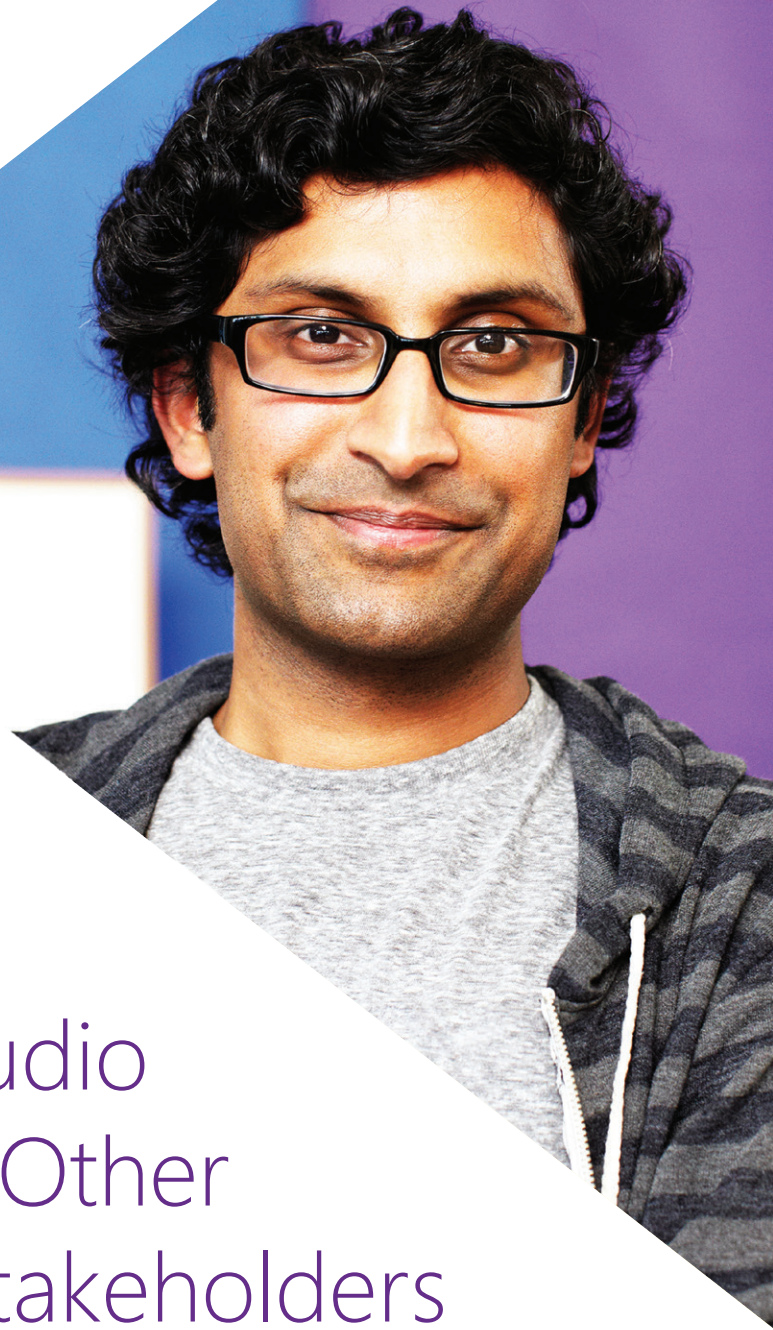
Snapshot name:  ...

 This option is available only for virtual environments.

< Previous   Next >   Finish   Cancel

## Conclusion

Designing, building, deploying, and maintaining successful applications is a collaborative effort that involves many people fulfilling a variety of roles. Visual Studio Team Foundation Server 2012 provides a scalable repository that facilitates active and meaningful communications among the different team members who are working on a project. The tools provided with Visual Studio Team Foundation Server 2012 enable developers, customers, and operations staff to capture highly descriptive and actionable data about the way in which an application behaves. Team Explorer in Visual Studio 2012 supplies a focused and up-to-date view of the project and the various tasks that a developer must perform. Together, all of these features help to improve the synergy that naturally exists between the members of the team, enabling them to collaborate more effectively and thereby increase the productivity of the entire development process.



# Visual Studio 2012 for Other Project Stakeholders

Visual Studio has long been an excellent tool for developers. The new version, Visual Studio 2012, incorporates the major enhancements for developers that are documented in this guide. And now Visual Studio also includes key enhancements that promote greater involvement by other stakeholders—product managers, designers, team leaders, architects, testers, customers, and others—in the development process. By bringing together key functionalities and integrating more closely with other technologies, Visual Studio 2012 enables these other stakeholders to interact more efficiently with developers, which ultimately helps to improve the quality of software solutions.

Managing development tasks has never been easier with Visual Studio 2012 Team Foundation Server (TFS), which now boasts a touch friendly web UI, enabling stakeholders to manage their product backlog and task board on their touch devices.

## Product Managers and Designers

In the past, product managers and designers typically had only intermittent exposure to Visual Studio. With Visual Studio 2012, product managers can use Visual Studio or the Web Access application. With Visual Studio 2012 TFS, product managers can create project backlog items, create associated tasks, and assign those tasks to sprints. For more information on the features available to product managers and designers, see the chapter Application Lifecycle Management with Visual Studio 2012 in this product guide.

The screenshot displays the TFS Web Access interface for a project named 'New Web site'. The top navigation bar includes links for HOME, WORK, SOURCE, and BUILD. The 'WORK' section is active, showing a 'backlog' and 'board' view. The 'board' view is selected, displaying a Kanban board for 'Sprint 4' running from '23 February - 29 February' (5 work days). The board is organized into three columns: 'TO DO', 'IN PROGRESS', and 'DONE'. The 'TO DO' column contains three items: 'Add log in page.', 'Add purchase history page.', and 'Test bug.'. The 'IN PROGRESS' column is empty. The 'DONE' column contains one item: 'Develop purchase history page.'. Each item is represented by a card with a title, a description, and an assignee. The 'Add log in page.' item is assigned to 'mikes@contoso...'. The 'Develop purchase history page' item is assigned to 'paulb@contoso...'. The bottom of the page features a footer with copyright information and links to Terms of Service, Privacy Statement, Service Notifications, and Trademarks.

	TO DO	IN PROGRESS	DONE
Add log in page. +	Create login page interface mikes@contoso...		
Add purchase history page +			Develop purchase history page paulb@contoso...
Test bug +			

# Project Managers

Project managers may not have access to Visual Studio, yet need quick, easy, anywhere access to key project metrics. With Visual Studio 2012 TFS, project managers can use the Web Access application to access project data. They can view project or individual utilization, and assign team members accordingly. They can track sprint progress and review the task board to see current assignments.

Visual Studio already provides native integration for high-quality project management tools such as Microsoft Project. Through TFS 2012, there is now native support for a connector to Project Server, which enables organizations to create a centralized view of all in-progress software projects.

For more information on the features available to project managers, see the section “Managing a Project” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

The screenshot displays the TFS web interface for 'Sprint 4' (February 23 - 29, 5 work days). A modal dialog titled 'DAYS OFF FOR: MIKES@CONTOSO.COM' is open, showing a table of days off.

Start Date	End Date	Net Days Off
24/02/2012	27/02/2012	2
<b>Total</b>		<b>2</b>

The background interface shows a list of team members and their capacity per day:

Team Member	Capacity Per Day
antony@contoso.com	0
johns@contoso.com	0
mikes@contoso.com	5
james@contoso.com	0
paulb@contoso.com	5

## Team Leaders and Architects

Architects play a key role in developing a software application, particularly if the application is large and involves many developers. Architects bring their great experience to bear in deciding the best way to build an application. Traditionally, architects also enforce an agreed upon design. Additionally, architects manage the valuable assets represented by the existing code base, work to reduce technical debt, and ensure that new projects work seamlessly with past projects.

Visual Studio 2012 introduces new robust architecture design, discovery, and enforcement tools for project architects in order to help teams deliver on the agreed design.

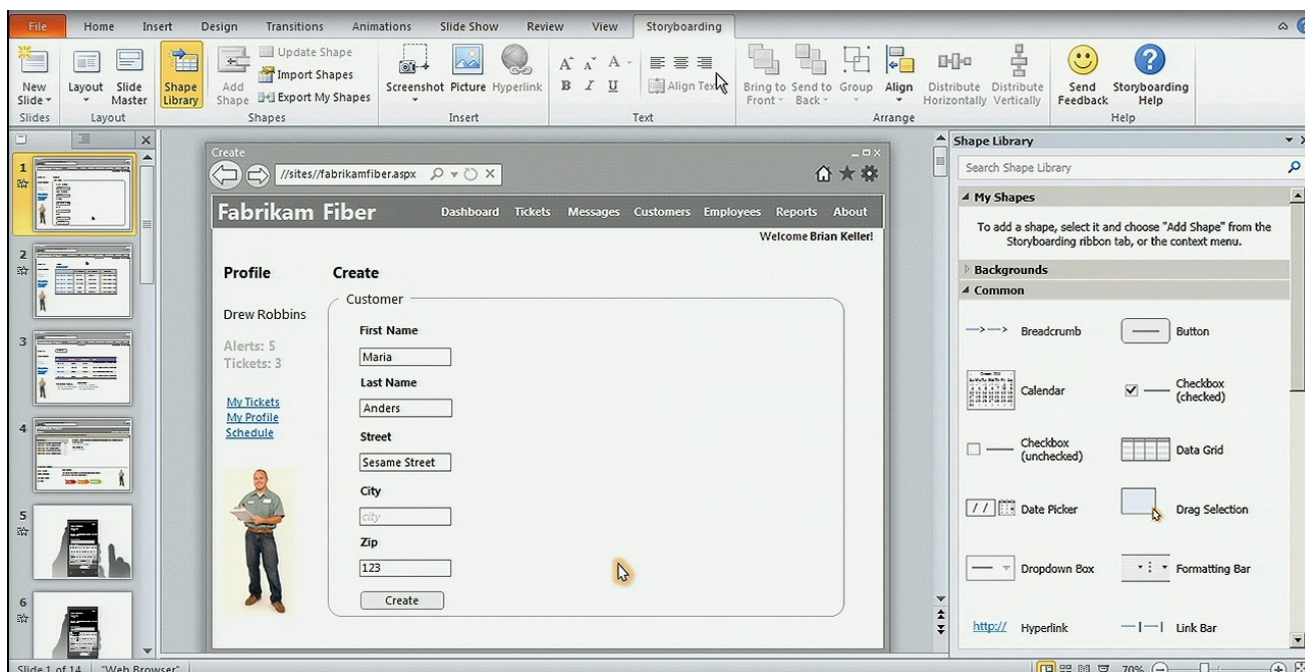
In addition, a code clone feature gives team leaders and architects tools to understand when the code base needs refactoring to make it more efficient and reusable, which reduces errors and improves quality and maintainability.

Code reviewer is another role that may be performed by a team leader, although in some organizations reviews are performed by peers. Visual Studio 2012 makes code reviews easy. Developers can request a review, associate the review with a work item, and choose who they want to perform the review. For more information on code reviews, see the section “Code Reviews” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

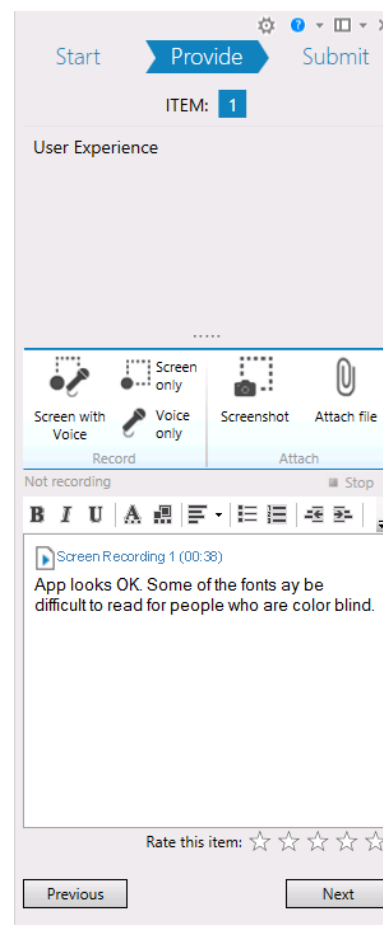
## Customers

Successful applications rely on input from external contributors who can help with the design of the application and provide feedback during and after development. Visual Studio 2012 includes the PowerPoint Storyboarding tool that product managers and designers can use to get rapid feedback from external contributors on the interface design. For more information on the PowerPoint Storyboarding tool, see the section “Gathering User Requirements” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

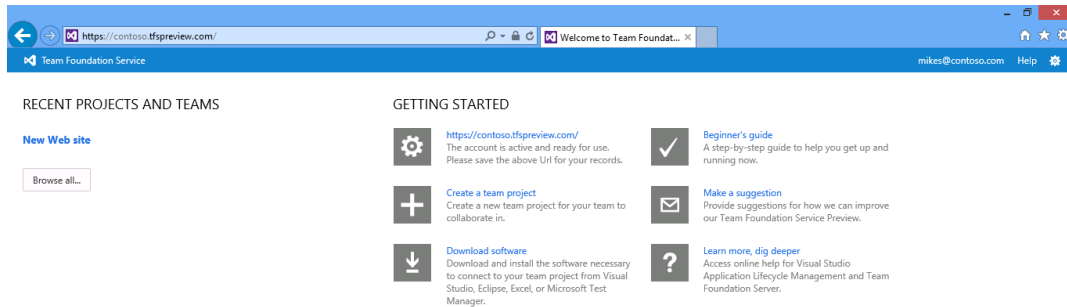




In addition to helping create interface mockups, external contributors such as end users can provide useful early feedback. Visual Studio 2012 includes the Feedback Manager tool. End users and other stakeholders can use the Feedback Manager tool to record themselves while using the application and to provide their feedback on the application. The feedback is saved directly to the TFS database, providing easy access for the developers, architects, testers, project and product manager, and other stakeholders. For more information on the Feedback Manager tool, see the section “Providing Customer Feedback” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

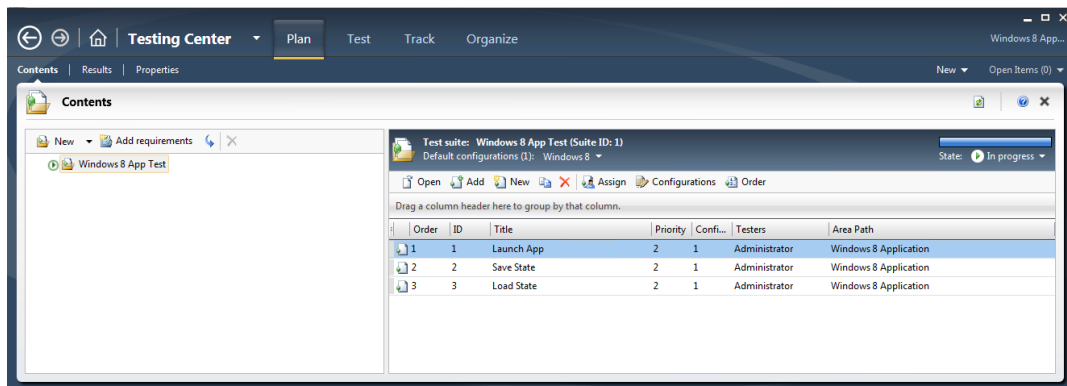


Contributors often are external to a company, or are not in the same domain or office. TFS Online simplifies the problems associated with enabling access to TFS for users beyond the perimeter network of an organization. With TFS Online, external users no longer require a VPN or credentials against the local domain. TFS Online enables you to easily add users from outside the company or in geographically dispersed locations simply by using their Microsoft account credentials. For more information on TFS, see the section “Visual Studio Team Foundation Server 2012 in the Cloud” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

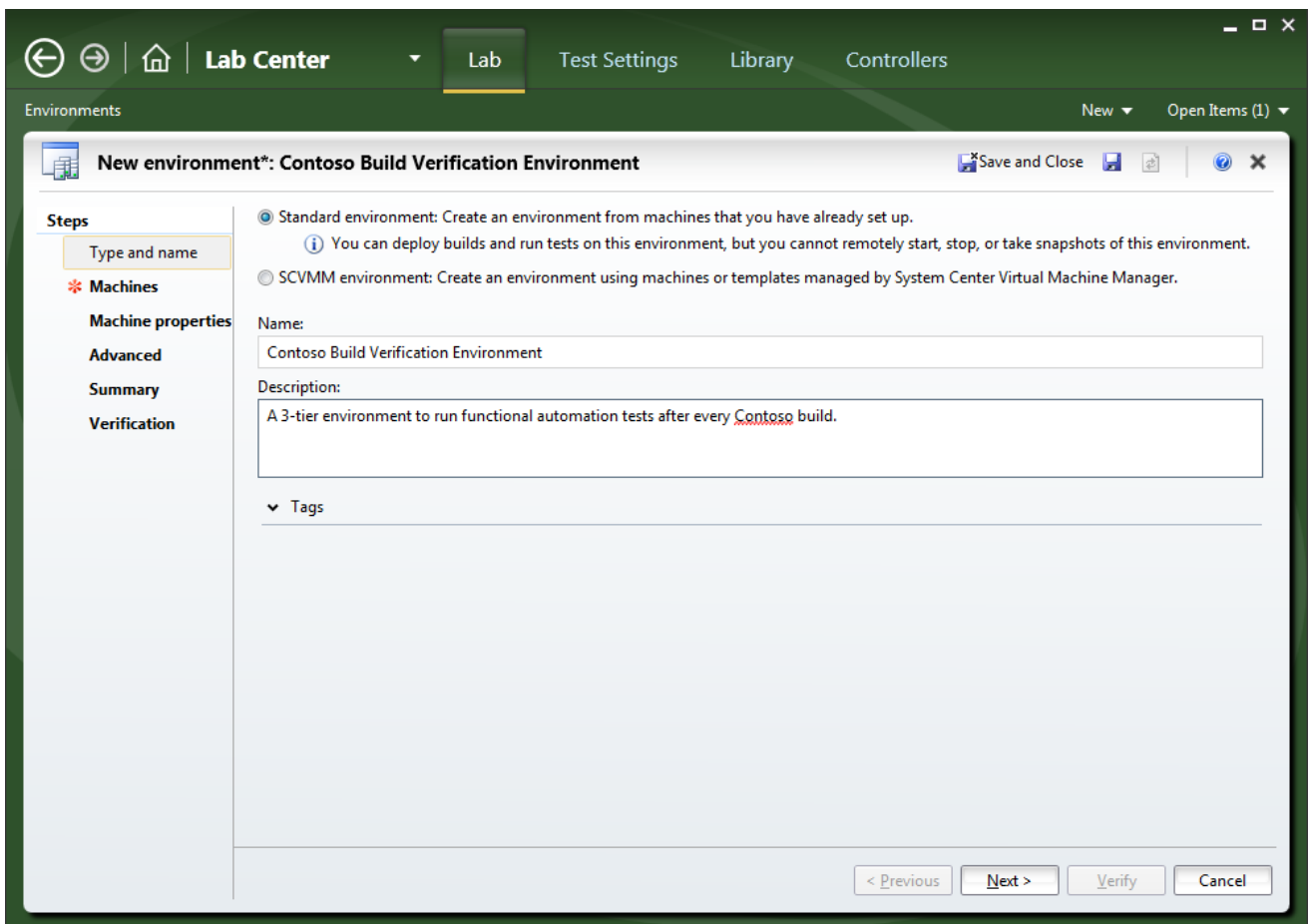


## Testers

Testers are key members of the development team. Visual Studio 2012 makes testers first-class members of the development process. Testers can use Test Manager to create test plans and perform tests. For more information on Test Manager, see the section “Visual Studio 2012 Product Family” in the Introduction to Visual Studio 2012 chapter of this product guide .



Testers create test labs in order to test applications in multiple environments. Lab Manager enables testers to create virtual test environments quickly, deploy the application, and then complete automatic tests. For more information on Lab Manager, see the section “Lab Management” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.



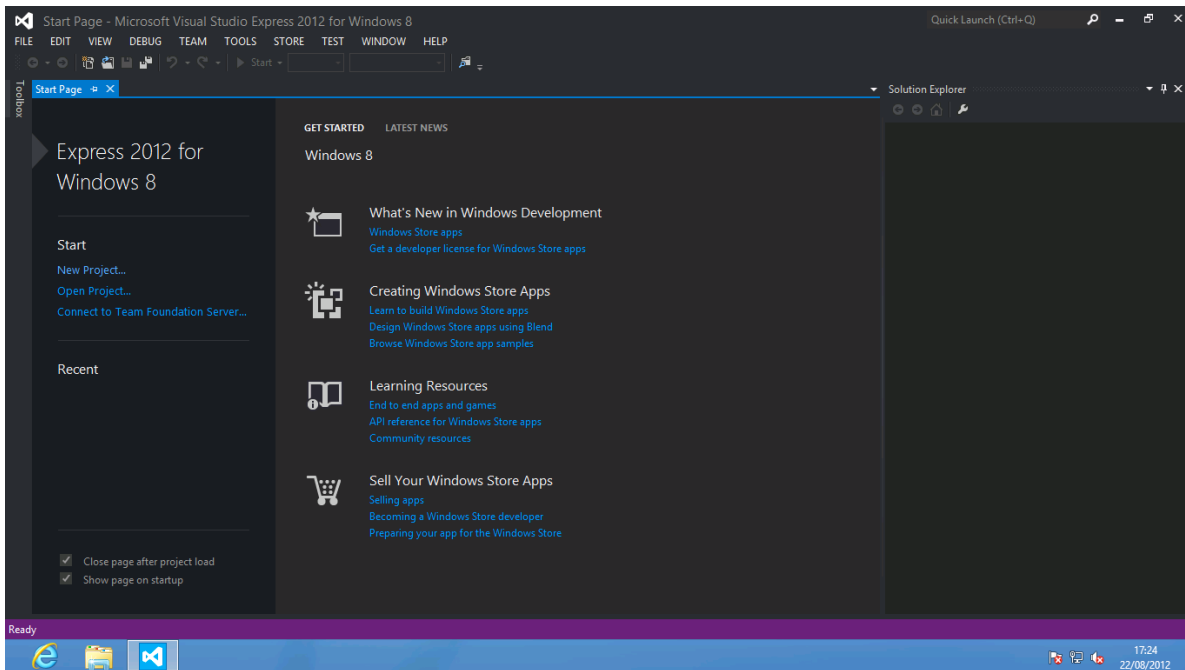
In addition to structured testing, application errors are often identified by performing unstructured exploratory testing. Visual Studio 2012 Test Manager supports exploratory testing. The tester can perform unstructured testing and the test manager will track the user’s actions and record the steps. For more information on exploratory testing, see the section “Performing Exploratory Testing” in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

Visual Studio 2012 ships with many enhancements that will help testers and developers ensure solutions are suitably tested. These include:

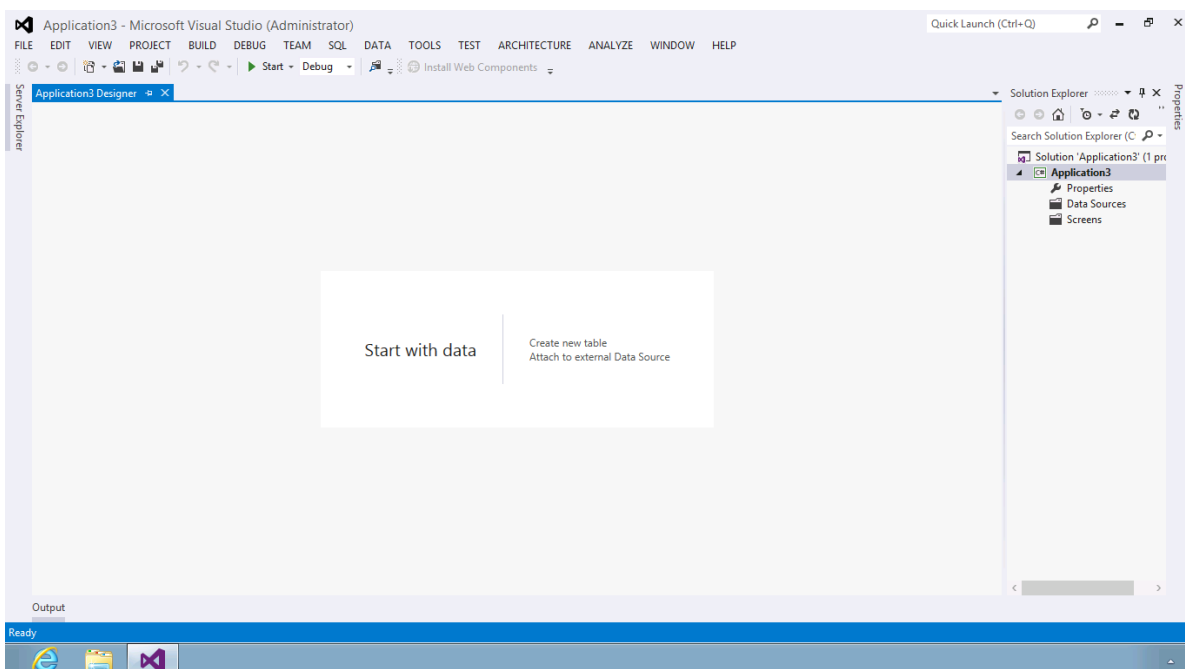
- Backward compatibility with Visual Studio 2010 test types.
- Unit test and code analysis support for Windows Store applications.
- HTML log files for analyzing Coded UI Test issues. Testers are also able to filter and display the most relevant and actionable information so that end-users can troubleshoot and fix Coded UI Test failures.
- Rich action logs containing text and images for Windows Store applications built with Visual C# and C++.
- Simplified test agent and controller communication in a cross-domain or workgroup environment with a lab service account
- Licensing simplification for load testing. You no longer need virtual user licenses to increase the number of virtual users that you can use in your load tests.

## End-user Developers

End-user developers create applications to meet specific business needs, often as a secondary role. End-user developers can use the Express edition of Visual Studio 2012 to create applications for a variety of devices. The Express edition supports integration with TFS, providing the integrated experience without the need to install the full version of Visual Studio.

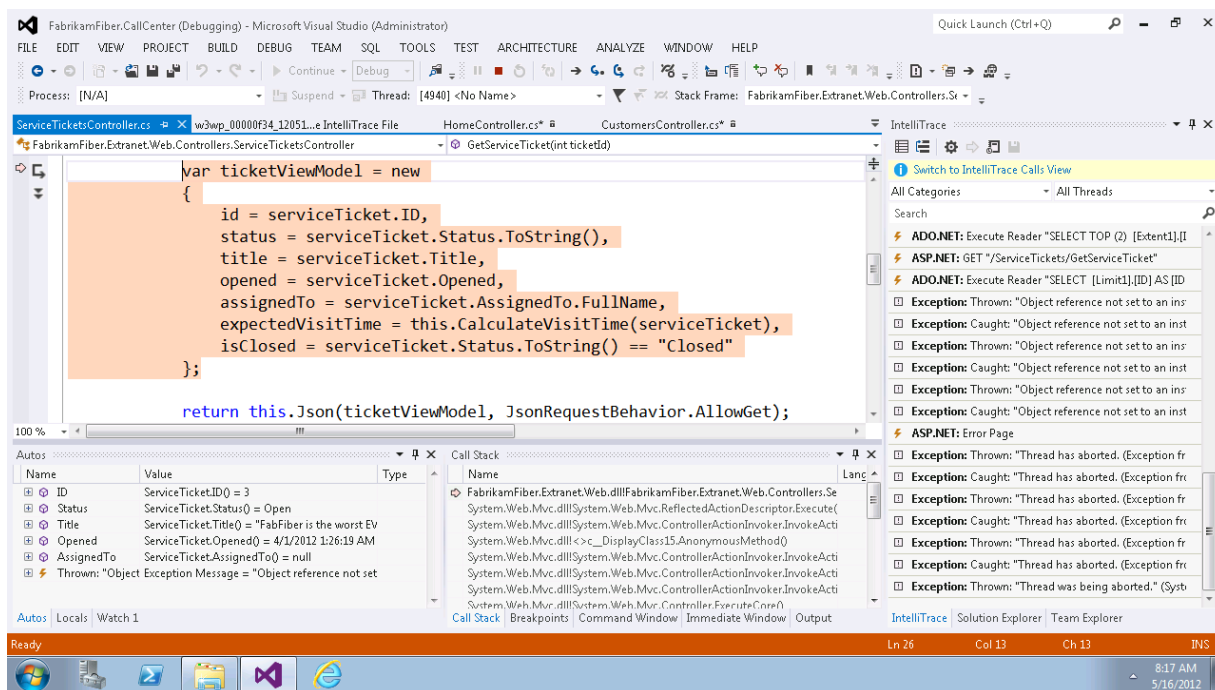


For data-driven applications, end-user developers can use Microsoft Visual Studio LightSwitch projects in Visual Studio 2012, which enables them to quickly and easily create data-driven applications by using visual tools and writing minimal code. This speeds up development and also helps to ensure more secure solutions.



## Operations Staff

Operations staff often encounters bugs not seen in testing. Inevitably, the production environment differs from the testing environment, causing unforeseen problems. With Visual Studio 2012, you can use IntelliTrace in Production to generate trace information about your application and, where appropriate, to automatically send that information to TFS. The PreEmptive Analytics feature enables you to automatically generate bug reports based on unhandled exceptions, which developers can fix in a future release. For more information on IntelliTrace in Production, see the section "Getting Feedback from Applications in Production" in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.



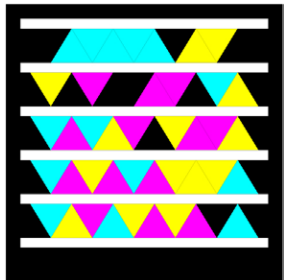
For web applications deployed by using System Center Operations Manager, you can deploy a monitoring agent that collects data about performance issues and exceptions. Operations staff can use this in conjunction with Microsoft TFS Connector to raise bugs and assign bugs to developers, providing a real-time feedback loop for deployed applications. For more information on server monitoring solutions, see the section "Working Closely with Operations Personnel" in the Application Lifecycle Management with Visual Studio 2012 chapter of this product guide.

## Conclusion

Professional software development is an engineering process that involves not just developers but also the whole range of planning, testing, management, and operations staff, together with the end-users for whom the software is being developed. The Visual Studio 2012 family provides a range of integrated tools that enable all stakeholders to cooperate to keep development projects on track, and that helps to ensure that the delivered software is timely, functional, and cost-effective.

## Feature Matrix

See which features and other benefits are included in the various Visual Studio 2012 editions and find which is the best for you at <http://www.microsoft.com/visualstudio/compare>







Microsoft

