Windows* 8 Development – Environment and Resources

Are you interested in developing applications for Windows 8? Or are you interested in adding some of the new features and APIs offered by Windows 8 to a current Windows 7 application? A lot of documentation exists on these subjects, but you may be feeling lost on where to begin. In this overview article, we will tell you about the tools and resources that will help you start coding right away.

The list below represents the best information about Windows 8 development, from features to coding, debugging, and simulating the final device. We will also talk about support for different languages and packaging.

Development Environment

- **Visual Studio* 2012**
The official development environment for Windows 8 is Visual Studio 2012 (VS 2012), an Integrated Development Environment (IDE) created by Microsoft, and this web site has the latest versions, updates, and information. VS 2012 is the first version to support the latest APIs and new features.

- **Develop Windows Store apps using Visual Studio 2012 (Windows)**
This is a great place to start if you don’t have experience with Visual Studio or software development. This article explains the process of creating applications using Visual Studio 2012, as well as highlighting many of its features.

- **Templates to speed up your app development**
Here’s another great resource to get started fast with Windows 8 development. This article explains how to use the templates available in VS 2012 for application development.

- **Windows Runtime - Windows 8 new APIs**
The newest sets of APIs introduced with Windows 8 are called Windows Runtime, or WinRT, and provide easy access to many resources in the operating system. You probably will also be happy to know that a subset of these are available for both Windows 8 Store apps and Desktop apps.

Games with Unity*

- **Developing Windows Store games with Unity**
A popular framework for developing games, Unity has an excellent set of resources popularly used by many developers. With backward compatibility with Windows 7, any game developed with Unity should be immediately compatible in the Windows 8 desktop mode. The presentation on this page indicates Unity is in “the process of leveraging existing Unity content to the Windows 8 platforms.”

Desktop Apps Development

- **Windows Store versus Desktop apps**
One of the first questions you may be asking is which environment to use—do I want to develop Windows Store or Desktop apps? This article can help you understand the differences.
• **Using WinRT from Desktop applications**
  If you want to create Desktop applications using WinRT, you will need to change a few settings in VS 2012. In this Intel Developer Zone article, you can find out how to do that. After changing these settings, you can use the WinRT APIs in the same way as you would when creating Windows Store applications.

• **Compatibility and Certification for Windows 8 Desktop**
  You can list your Desktop applications in the Windows Store. This page gives you information on how to certify, register, and submit your application.

### Programming Languages

• **WinRT Programming Languages:** C++/CX, .NET(C# or Visual Basic*) and JavaScript*
  Microsoft provides roadmaps for creating Windows Store applications in those programming languages that are officially supported for Windows 8 development. In these roadmaps, you can find information about developing for Windows 8 (guidelines for design and specific features like controls, layouts, and more), and you can download the tools.

• **Other Languages:** Adobe AIR* and Java*
  Some languages do not support WinRT. However, if you want to offer these applications, a good place to start is Intel AppUp*. In these articles, you can find information about distributing applications written in AIR and Java and uploading them to the Intel AppUp center.

### Testing and Debugging

• **Running Windows Store apps in the simulator**
  This MSDN article covers how to debug and test Desktop and Windows Store apps in Visual Studio. Using the Simulator mode, you can run your applications and even simulate touch and gesture, as well as location. You can simulate the Windows 8 Modern UI* environment in a sandbox to test your application’s behaviour without affecting the rest of the system. You can also find out how well your application handles suspend/resume situations, which must meet minimum standards enforced by Windows 8.

### Packaging

• **Packaging for Windows 8 Store apps**
  Visual Studio also helps you to package your Windows Store application for distribution through Windows Store. This applies only to Windows Store apps, not Desktop apps.

• **Packaging Desktop apps with the Intel AppUp SDK plugin**
  For Desktop applications, most modern IDE environments offer you help to create an MSI installer. This article tells you how to use Visual Studio 2008 or 2010 with an Intel AppUp plugin to create MSI packages. If your application has an EXE installer, it can also be submitted to the Intel AppUp center.

• **Packaging for AppUp – Java and Adobe AIR**
  With Intel AppUp you can also submit Desktop applications created in Java and Adobe AIR, and these links have specific information about packaging them. But make sure to read and
understand the packaging guidelines to have a faster and smoother process in submitting your applications!

We’ve told you about a few of the major sources of information available online, and we hope this helps to quickly get you up to speed. If you know other great articles or web sites about Windows 8 development environments, please share them with us in the comments!

**Notices**

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: [http://www.intel.com/design/literature.htm](http://www.intel.com/design/literature.htm)

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark* and MobileMark*, are measured using specific computer systems, components, software, operations, and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Any software source code reprinted in this document is furnished under a software license and may only be used or copied in accordance with the terms of that license.

Intel, the Intel logo, and AppUp are trademarks of Intel Corporation in the US and/or other countries.

Copyright © 2013 Intel Corporation. All rights reserved.

*Other names and brands may be claimed as the property of others.