Supporting Advanced Color in Print Drivers

December 17, 2007

Abstract

This paper provides information about incorporating Window Color System (WCS) support in both Microsoft version‑3 and XPSDrv print drivers for the Microsoft® Windows Vista® operating system. It provides prescriptive information for independent hardware vendors (IHVs) to develop WCS-enabled print drivers for Universal (Unidrv), PostScript (PScript), monolithic, and XPSDrv print drivers. It also provides guidelines for independent software vendors (ISVs) to control color management functions that are exposed by print drivers.

IHVs that have existing printers or are developing new printers can use this paper to help plan their implementation of WCS support in new and existing print drivers.

This information applies for the following operating systems:  
 Microsoft Windows Server® 2008  
 Windows Vista®  
 Microsoft Windows XP  
 Microsoft Windows Server® 2003

For up-to-date information, see “Color Management for Printers” in the Windows Driver Kit documentation at:

<http://msdn2.microsoft.com/en-us/library/ms802157.aspx>.

The current version of this paper is maintained on the Web at: <http://www/microsoft.com/whdc/xps/color-drv.mspx>

References, papers, and other resources discussed in this paper are listed at the end of this paper.

Contents

[Introduction 3](#_Toc134377000)

[WCS in the Print Driver 3](#_Toc134377001)

[PrintTicket and PrintCapabilties Features 3](#_Toc134377002)

[Printing with Color Management 4](#_Toc134377003)

[Color Components during Configuration 4](#_Toc134377004)

[Color Components during Printing 5](#_Toc134377005)

[Developing a WCS-enabled Print Driver 6](#_Toc134377006)

[Driver Package 6](#_Toc134377007)

[Installation 6](#_Toc134377008)

[WCS in the Configuration Module 7](#_Toc134377009)

[Configuration Support 7](#_Toc134377010)

[Configuration UI 8](#_Toc134377011)

[WCS in the Render Module 9](#_Toc134377012)

[Reading the PrintTicket 9](#_Toc134377013)

[Performing Color Management 10](#_Toc134377014)

[Color Management in Earlier Versions of Windows 11](#_Toc134377015)

[Next Steps 11](#_Toc134377016)

[Resources 12](#_Toc134377017)

Disclaimer

This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© 2006 Microsoft Corporation. All rights reserved.

Microsoft, Win32, Windows, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

#### Revision History

|  |  |
| --- | --- |
| **Date** | **Change** |
| 12/17/2007 | Update WDK documentation references |
| 5/01/2006 | Created |

Introduction

Microsoft® Windows Vista® introduces a new color management system that is called the Microsoft Windows® Color System (WCS). WCS offers applications the opportunity to perform wide-gamut, high dynamic–range color processing of spool file data that exceeds what is possible in the standard Red-Green-Blue (sRGB) color space and is not available in earlier versions of Windows. These features enable advanced image processing and photographic scenarios that are not possible in earlier versions of Windows.

The Image Color Management (ICM) functions have been enhanced in Windows Vista to use WCS; however, in some areas this general assumption does not apply. This paper concentrates on the additional steps that are necessary to add support color management in a print driver so that it will run correctly in Windows Vista and earlier versions of Windows.

The Windows Driver Kit (WDK) and the other resources listed at the end of this paper provide more information about writing a Windows print driver.

## WCS in the Print Driver

The primary components of a graphics device interface (GDI)-based, version-3 print driver and an XPSDrv print driver are the configuration module and the rendering module. WCS features can be applied in both modules.

The configuration module plays several roles in color management. Adding WCS to the configuration module can:

* Expose the color capabilities to applications in the PrintCapabilities document and by using the GetDeviceCaps function.
* Process all application settings input for Universal (Unidrv), PostScript (PScript), and monolithic print drivers.
* Query WCS to extract profile information that is necessary to perform these steps.
* Perform installation-time registration of driver-supplied WCS components.

The rendering module or rendering module plug-in of a Unidrv, PScript, or monolithic print driver performs all print-driver color management or device preprocessing color management. The rendering module of the driver or plug-in:

* Interprets the color profile configuration that is described in PrintTicket.
* Performs the required color transformations that are described by that configuration.

## PrintTicket and PrintCapabilties Features

Windows Vista also introduces a new eXtensible markup language (XML)-based printer capabilities and configuration mechanism that is based on the Print Schema and implemented by the PrintTicket and PrintCapabilities technologies. The Print Schema addresses the issues of opaqueness and ambiguity that have troubled earlier printer settings technologies. These issues have complicated communication between print subsystem components and between the print subsystem and applications. The Print Schema ships with Windows Vista and is planned for later release in Window XP and Microsoft Windows Server™ 2003.

The PrintTicket and PrintCapabilities features enable communication and control of advanced color processing in Windows Vista for WCS-enabled print drivers and general print drivers. This means that print drivers can accurately communicate color information to applications and eliminate cases of multiple color management that occurred in earlier versions of Windows.

The PrintTicket and PrintCapabilities features make it possible for printing applications to fully control color processing in the print path. With these features, applications can access the color functions of print drivers and explicitly coordinate the color management settings between the application and the print driver. These features can be incorporated into new, XPSDrv print drivers and the GDI-based, version-3, print drivers that were written for earlier versions of Windows.

# Printing with Color Management

Windows Vista offers applications the ability to explicitly define color processing control and includes functions that eliminate color mismanagement ambiguity in the print path. Windows Vista components and third-party software and hardware make up the color components in the Windows Vista print subsystem.

## Color Components during Configuration

During the configuration phase of printing, the application queries the printer properties to determine how to configure the printer and the application data for the desired output. An application can query the print driver to determine what color functions it supports, and it can also query WCS to determine what relevant color profiles are available on the computer. Based on the information that the print driver and WCS return, the user and the application can determine how to apply color management for the desired results.



Figure 1. WCS Color Components during Configuration

Figure 1 illustrates the different paths that an application can use to access the print driver components and WCS during configuration. In Windows Vista, both Microsoft Win32® and Windows Presentation Foundation (WPF) applications can use WCS.

Before printing, the application can query the print driver by calling the DeviceCapabilities or the GetPrintCapabilities function to query the color support from the print driver. The print driver, in turn, can query WCS to determine the profiles that are currently associated with the print device so that they can return the correct printer capabilities to an application. The application can then decide, possibly based on user input, whether to perform the color management or to direct the print driver to perform the color management. In either case, the application should communicate to the print driver the steps that the driver is expected to perform.

## Color Components during Printing

After the application has determined how it will perform color management, it can direct the print driver by including configuration information in PrintTicket objects. The PrintTicket objects are included with the document or added to the document when it is spooled for printing.



Figure 2. WCS Color Components during Printing

Figure 2 shows how WCS is accessed after the application spools the document and the print driver processes the document for printing. The application queues the print job to the print spooler. During the spooling process, the application can add PrintTicket objects to the spooled document. These PrintTicket objects can specify how the driver should handle color management and include other printer configuration information.

The print spooler sends the spooled document to the print driver filters where the document is rendered for printing. The rendering module components of the print driver read the PrintTicket objects from the spooled document and query WCS for the appropriate profile information to use when applying the color management that PrintTicket described.

# Developing a WCS-enabled Print Driver

The Windows ICM 2.0 functions are supported in Windows Vista and earlier versions of Windows. Windows Vista also offers new color management functionality that the ICM functions do not handle. In addition, some of the ICM functions behave differently in Windows Vista than they do in earlier versions of Windows. To design and implement a print driver that is compatible in Windows Vista and earlier versions of Windows, the print driver developer must understand these differences.

This section describes some of the most common issues that might arise when adding WCS and color management support to a GDI-based, version‑3 print driver or an XPSDrv print driver. However, because the Windows SDK and WDK describe the color management functions in detail and indicate any differences in behavior from one version of Windows to another, they should be considered the final reference on the subject.

When designing the print driver to use WCS, remember that WCS functions are available only when the print driver runs in Windows Vista. Drivers that can also be installed to run in earlier versions of Windows must include alternate ways to handle the functionality that WCS functions provide, such as by using the existing ICM functions.

## Driver Package

WCS profiles are listed in INF files for Windows Vista in the same way as International Color Consortium (ICC) profiles are listed. Both types of profiles are listed in the same way as the ICC profiles were listed in INF files for Windows XP and Windows Server 2003. The WDK describes how color profiles are listed in the INF file.

Because WCS profiles are supported only by Windows Vista and not by earlier versions of Window, the INF file must list WCS profiles in a section of the INF file that will be run only in Windows Vista.

WCS plug-ins are listed in the INF file and registered in the same way as other printer plug-ins. However, because WCS plug-ins are supported only in Windows Vista, they must be listed in an INF-file section that is installed only on Windows Vista machines.

## Installation

The installation of a WCS-enabled print driver is fundamentally the same as the installation of a print driver that does not use WCS because most of the WCS-related features are exercised at run time. Installation time, however, might be a convenient time to define static parameters or configuration switches for the print driver to save run-time checking of these parameters.

In addition to the required steps for any print driver installation, print drivers must install and associate any color profiles that are unique to the print driver and also install and register any WCS-specific plug-in modules.

When coding installation plug-ins that will run in Windows Vista and earlier versions of Windows, it is important to remember that WCS functions are not available in earlier versions of Windows. Also, the ICM functions behave slightly differently in Windows Vista then they do in earlier versions of Windows.

One critical difference between the ICM functions and the WCS functions in Windows Vista is that ICM functions run in the user's account whereas the WCS functions can run in the System account, assuming that the user has the required access. Developers of custom installation code should consider this difference to ensure the correct installation behavior in each version of Windows that the installation code supports.

For example, if AssociateColorProfileWithDevice, an ICM function, is run in Windows Vista, the profile and device association is installed for only the user who is installing the printer. For the profile to be associated with the device for all users when installed on a Windows Vista computer, the WcsAssociateColorProfileWithDevice function, with the scope set to WCS\_PROFILE\_MANAGEMENT\_SCOPE\_SYSTEM\_WIDE, should be used. Windows Vista Setup considers this difference and uses the WCS function to install WCS profiles with the system-wide scope in Windows Vista.

## WCS in the Configuration Module

The configuration module of a print driver contains the configuration user interface support and the configuration support functions. A print driver that supports WCS might be required to modify both of these components. All modifications must be aware of the version of Windows in which they are running.

### Configuration Support

Configuration support in a print driver provides the printer configuration information to an application. The driver should test the operating system version and return only the features that are supported by that version because the features can be different in different versions of Windows.

Print Drivers that are based on the PScript5 or the Unidrv print drivers provide this information from the plug-in modules. By default, standard print driver configuration module supports the PageColorManagement and PageICMRenderingIntent features of the Print Schema. If the print driver or the target printer cannot fully support the standard color management features, it should use a plug-in to remove those entries from the PrintCapabilities document.

The plug-in can add printer capabilities such as WCS support to the PrintCapabilities document by supporting the IPrintOemPrintTicketProvider interface. This is described in the WDK and the paper titled PrintTicket and PrintCapabilities Support in Windows Print Drivers. The PrintCapabilities entries that the plug-in returns should reflect the capabilities of the print driver. Because not all features are supported by all versions of Windows, the features and options that are returned might need to be modified depending on which Windows version the driver is running at the time. This is important to consider when developing the print driver because the application uses this information to determine how to apply color management for printing.

The Print Schema offers the following additional feature names that pertain to color management:

* JobOptimalDestinationColorProfile
* PageSourceColorProfile
* PageSourceColorProfile
* PageBlendColorSpace
* PageColorManagement
* PageDestinationColorProfile
* PageDeviceColorSpaceUsage
* PageICMRenderingIntent
* PageSourceColorProfile

These Print Schema features are described in the Print Schema documentation.

### Configuration UI

The configuration user interface (UI) of the print driver must test any PrintTickets that it receives from the application before populating and displaying a color profile selection UI. Some applications perform color management within the application. Those applications send a PrintTicket to the print driver when requesting the Printer Properties dialog box. The print driver should test the PageColorManagement feature of the PrintTicket to determine where the color management will be performed. If color management will be performed in the application, the color management options of the printer should be disabled (unavailable) or not appear to the user.

Print drivers that provide custom property sheets for color configuration should enumerate all installed color profiles and present then to the user rather than just display those that the print driver installed. Some advanced users will have created their own color profiles and want to use them to print. The possible profiles that are associated with the device to display include:

* IHV-provided device profiles
* User-defined profiles
* ISV-provided or third-party profiles

When the Printer Configuration dialog box presents a list that is different from what appears in other lists, such as the system color control panel, users cannot be certain if their color profile will be used and the application can apply the profiles incorrectly. Also, if users have associated their own custom profiles with a device, they expect to see them in the Printer Configuration dialog box for that device.

The functions that are used to enumerate these profiles depend on the version of Windows. In Windows Vista, for example, the WCS function WcsEnumColorProfiles is preferred because this API allows specific control over whether the query is system wide or user specific.

BOOL WINAPI WcsEnumColorProfiles(

\_\_in WCS\_PROFILE\_MANAGEMENT\_SCOPE profileManagementScope,

\_\_in PENUMTYPEW pEnumRecord,

\_\_out\_bcount(dwSize) PBYTE pBuffer,

\_\_in DWORD dwSize,

\_\_out\_opt PDWORD pnProfiles

);

When the print driver runs in an earlier version of Windows because it cannot use the WCS function, it should use the corresponding ICM function, EnumColorProfiles.

BOOL WINAPI EnumColorProfiles(

PCTSTR pMachineName,

PENUMTYPE pEnumRecord,

PBYTE pBuffer,

PDWORD pdwSize,

PDWORD pnProfiles

);

Windows Vista supports both user- and system-level profile associations. The profileManagementScope argument of the WcsEnumColorProfile function allows the print driver to select which group to enumerate. The EnumColorProfiles function, when run in Windows Vista, list only the user-level color profiles. When run in earlier versions of Windows, however, it lists the system-level color profiles because only system-wide color profiles and associations are in earlier versions of Windows.

The process of getting the profile name from the profile is also different between ICC profiles and WCS profiles. The profile name of an ICC profile is obtained by calling the GetColorProfileElement to return the profileDescriptionTag (desc) element. For a WCS profile, the profile name is extracted by parsing the XML of the profile and retrieving the value of the ProfileName element.

To determine the profile type, the ICM function GetColorProfileHeader works on both WCS and ICC profiles. It can be used to read the value of the CMM Type signature (phCMMType). Because of the difference in format, however, when reading a WCS profile header, none of the other fields of the WCS profile is correct.

In a WCS profile, the value of the phSignature field in little-endian notation is ‘pmdc’ or, in big-endian notation, ‘cdmp’. The value of the phCMMType field in the header in little-endian notation is ‘1scw’ or, in big-endian notation, ‘wcs1’. If the value of the phSignature field is ‘acsp’ in big-endian notation, then the profile is an ICC profile.

## WCS in the Render Module

Depending on the driver design, color management features can be incorporated with other rendering functions or designed as a separate filter or driver plug-in.

Print drivers that run in Windows Vista and earlier versions of Windows must check the operating system version to ensure that the WCS functions are called only if the driver is running in Windows Vista. The ICM functions that do not have a corresponding WCS function should work properly in Windows Vista and earlier versions of Windows.

The filter module that supports color management in Windows Vista must support the PrintTicket object because that is how the color management information is sent to the print driver. Reading PrintTicket information in a print driver filter is described in the paper titled PrintTicket and PrintCapabilities Support in Windows Print Drivers. If color management must be applied by the filter, it must examine the stream from the InterFilterCommunicator for elements that need color correction or transformation. The XPSDrv print driver sample in the WDK includes a sample of how this can be done.

### Reading the PrintTicket

To determine what type of color management the filter must perform, evaluate the PageColorManagement feature of the PrintTicket. The possible values of the SelectionType property are:

* None. The application has already performed color management, so the driver is not required to perform any additional color management.
* Device. The device performs color management.
* Driver. The driver performs color management.
* System. The operating system performs color management.

If the value of the SelectionType property of this feature is not Driver, then the print driver filter is not required to perform color management and can just pass the document parts through the filter unchanged. The rest of this section describes the situation in which the value is Driver and the print driver filter must perform color management.

### Performing Color Management

When the PrintTicket settings direct the filter to perform color management, Windows Vista offers the following features in the PrintTicket the filter module can query to obtain the required color management configuration.

* JobOptimalDestinationColorProfile
* PageSourceColorProfile
* PageDestinationColorProfile
* PageBlendColorSpace
* PageDeviceColorSpaceUsage
* PageICMRenderingIntent

These PrintTicket features are described in the Print Schema documentation.

The PageBlackGenerationProcessing PrintTicket feature is not directly supported in the version of WCS that is available in Windows Vista. The term, however, can still be used in a PrintTicket and can be processed by the filter.

Based on the information from the PrintTicket, the filter must then perform the operations on the document that are described by the PrintTicket. For example, the PrintTicket might include a PageSourceColorProfile and a PageDestinationColorProfile. In that case, the filter would:

* Call OpenColorProfile or WcsOpenColorProfile to obtain a handle to the profiles.
* Call CreateMultiProfileTransform to create a handle to the transform that must be performed.
* Call TranslateColors, WcsTranslateColors, or TranslateBitmapBits as required by the driver to perform the color transforms on the objects that require color management.

The filter must determine the functions to take based on the operating system version in which it is running and the profile type. This is a simple example that addresses how two of the possible PrintTicket features might be addressed. Similar considerations must be made when implementing support for other PrintTicket features.

## Color Management in Earlier Versions of Windows

Because WCS is supported only by Windows Vista, print drivers that can run in Windows Vista and earlier versions of Windows must ensure that WCS profiles and functions are used only when the driver is running in Windows Vista. Ideally, the driver can call an equivalent function when the driver is not running in Windows Vista to provide similar functionality in earlier versions of Windows. The following code illustrates how this might be implemented:

if (IsPlatformGreaterOrEqualToVista()) {

// running on Windows Vista so

// call Windows Color System function

\*phProf = WcsOpenColorProfile(&profile,

NULL,

NULL,

PROFILE\_READ,

FILE\_SHARE\_READ,

OPEN\_EXISTING,

0);

} else {

// not running on Windows Vista so

// call equivalent Image Color Management function

\*phProf = OpenColorProfile(&profile,

PROFILE\_READ,

FILE\_SHARE\_READ,

OPEN\_EXISTING);

}

From the user's perspective, the driver should provide comparable features as much as possible regardless of the version of Windows. Because WCS is available only in Windows Vista, however, some features might not be available completely when run in earlier versions of Windows. In those cases, the functionality should degrade in such a way as to render the output as well as possible with the available software support.

Sometimes it may be impossible to provide the same options in all versions of Windows. In all cases, the options that appear in the Printer Configuration dialog box and the features that are returned in the PrintCapabilities document should agree and reflect the features that the print driver can perform in the version of Windows on which it is running. The constrained attribute can be used with a feature element in the PrintCapabilities document to indicate that the feature is supported by the driver but is not available for use. Any other programming interfaces that the driver supports should also return the same set of available features and options as the PrintCapabilities document returns and printer configuration UI displays.

# Next Steps

Developers and vendors of printer and document processing products are encouraged to review the advantages and requirements of supporting WCS and the XPS print path. They are encouraged to review how to apply this technology and how it can fit in their immediate and strategic product plans.

Microsoft encourages vendors to take the following steps:

* Read the XML Paper Specification to understand the XPS Document format.
* Review all technical documentation about the XPS print path to better understand printer consumption challenges and opportunities.
* Consider implementation opportunities to take advantage of scalable and direct consumption, which can maximize software development resources across product lines.
* Consider providing updated drivers for the installed base to take advantage of improved printing capabilities that are enabled through XPSDrv drivers.
* Become familiar with areas of extensibility to host value-added functions.
* Design support for the Microsoft XML PrintTicket and XML PrintCapabilities interfaces into GDI-based drivers for Windows Vista and later versions of Windows.

# Resources

If you have questions about XPS or this white paper, send an e-mail to: [xpsinfo@microsoft.com](mailto:xpsinfo@microsoft.com)

XML Paper Specification  
(Information about XPS and related topics as well as links for end users, IT professionals, ISVs, and IHVs)  
<http://www.microsoft.com/whdc/xps>

Windows SDK  
(Print Schema documentation under WinFX Development/General Reference)  
<http://windowssdk.msdn.microsoft.com/library/>

Windows Driver Kit (WDK)  
<http://www.microsoft.com/whdc/driver/WDK/aboutWDK.mspx>

White Papers:

XPS and Color Printing Enhancements in Windows Vista

<http://www.microsoft.com/whdc/xps/vista_print.mspx>

XPSDrv Filter Pipeline

<http://www.microsoft.com/whdc/device/print/XPSDrv_FilterPipe.mspx>

PrintTicket and PrintCapabilities Support in Windows Print Drivers

<http://www.microsoft.com/whdc/device/print/XPSDrv_PrintTicket.mspx>

XPSDrv Configuration Module Implementation

<http://www.microsoft.com/whdc/device/print/XPSDrv_config.mspx>

Printer Installation in Windows Vista

<http://www.microsoft.com/whdc/xps/pkginstall.mspx>