

What's new in Feature Pack 2007 for XP Embedded

By Sean D. Liming and John R. Malin

Copyright © 2006 SJJ Embedded Micro Solutions, LLC., All Rights Reserved

No part of this guide may be copied, duplicated, reprinted, and stored in a retrieval system by any means, mechanical or electronic, without the written permission of the copyright owner.

First Printing: January 2006

Published in the United States by

SJJ Embedded Micro Solutions, LLC.

11921 Tivoli Park Row #5
San Diego, CA 92128 USA

www.sjjmicro.com

Attempts have been made to properly reference all copyrighted, registered, and trademarked material. All copyrighted, registered, and trademarked material remains the property of the respective owners.

The publisher, author, and reviewers make no warranty for the correctness or for the use of this information, and assume no liability for direct or indirect damages of any kind arising from the information contained herewith, technical interpretation or technical explanations, for typographical or printing errors, or for any subsequent changes in this article.

The publisher and author reserve the right to make changes in this publication without notice and without incurring any liability.

Windows, .Net Embedded, and Visual Studio are registered trade mark of Microsoft Corporation.

Table of Contents

1	INTRODUCTION	4
2	WHAT'S NEW	4
2.1	COMMAND LINE TOOL – XPECMD.WSF	4
2.2	USB 2.0 BOOT	6
2.3	FILE BASED WRITE FILTER (FBWF)	6
2.4	REGISTRY FILTER	7
2.5	NEW COMPONENTS AND EEF CATEGORY	7
2.5.1	. New EEFs category node in Target Designer	7
2.5.2	Microsoft SQL Express 2005 Edition	8
2.5.3	Font and CodePage Application Compatibility components	8
2.5.4	Additional Locales and Keyboard Layouts	8
2.5.5	DirectShow Indeo Codecs	9
2.5.6	Previously orphaned file: Cacls.exe	9
2.5.7	Previously orphaned file: Gpupdate.exe	9
2.6	COMPONENT CHANGES	9
2.6.1	EFW configuration UI Setting page Enhancements	9
2.6.2	User Account and Administrator Account	10
2.6.3	System Cloning Tool	11
2.6.4	Standard Start Menu Short Cuts updated	11
2.6.5	Dr. Watson Debugger component	11
2.6.6	Updated Windows Installer Service	12
2.6.7	Component Descriptions for Primitives	12
2.7	FOOTPRINT REDUCTION	12
2.7.1	Internet Explorer Footprint Reduction	13
2.7.2	Footprint reduction for SMS Client scenarios	14
2.8	SUMMARY	14

1 Introduction

Windows XP Embedded (XPe) is one of the most popular embedded operating systems available. XPe is being used in Point-Of-Service, gaming systems, ATM machines, test equipment, control systems, office equipment, thin clients, and multimedia solutions. Feature Pack 2007 builds on the XPe features and adds some new capabilities and updates some old components. Here we will take a high level look at the latest release.

2 What's New

Feedback! In my XPe conference sessions and training courses, I always suggest that everyone provide feedback to the XP Embedded team. Feedback is an important part of any software development process. The more the same feature request comes back, the better the chance of it being in the final release. Feature Pack 2007 (FP2007) is a culmination of much of the feedback that the MVPs and developers have provided over the past 4 years. Many changes to components and some new embedded enabling features (EEF) are introduced. Here we will review the following features that will be part of FP2007:

- Development Tools
- USB 2.0 boot
- File Base Write Filter (FBWF)
- Component Changes and/or Updates
- Footprint improvements: Various components fixed to remove outdated component relationships.
- Several New components such as SQL Server Express, font and code page compatibility and components for orphan files.

2.1 Command Line Tool – XPECMD.WSF

The biggest request over the past couple years is the need for a command line utility that performs the same operations as Target Designer. The need comes from the idea that a build can be scripted directly from the version control system without having to manually open Target Designer and perform the build.

XPECMD.WSF is the new Windows Script File that allows you to perform the following:

- Create a new configuration
- Load a configuration
 - Example: XPECMD> load cfg "c:\myimages\image.slx"
- Save a configuration
- Search for components
 - Example (Search for IE):
XPECMD> new myInst Instance
XPECMD> myInst = 'inst:^Internet Explorer'
- Add components to a configuration
 - Example (Add IE to a configuration):
XPECMD> load cfg "c:\myimages\image.slx"
XPECMD> new myInst Instance
XPECMD> myInst = 'inst:^Internet Explorer'
XPECMD> cfg.Instances.Add myInst
- Modify the properties of an instance of a component
- Perform a dependency check and resolving dependencies
 - Example:
XPECMD> cfg.ResolveDependencies 2

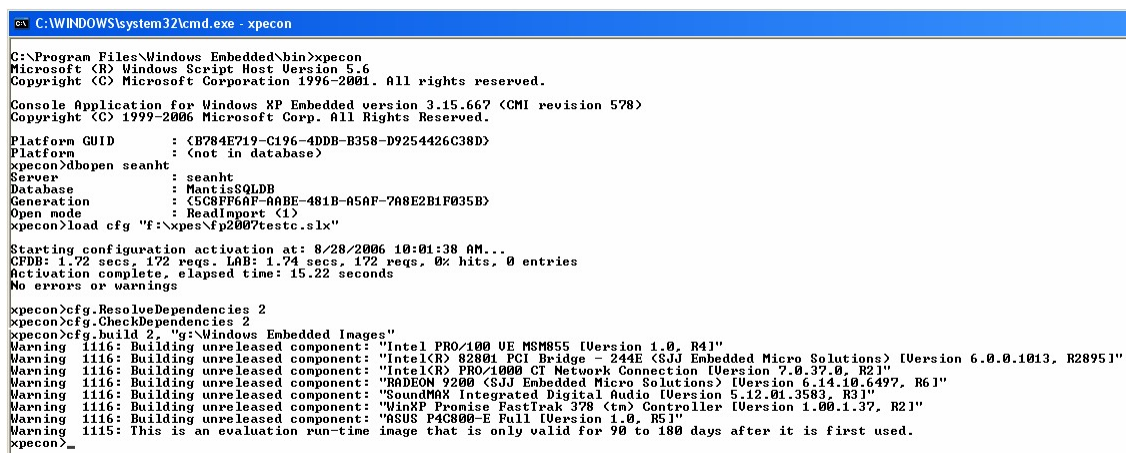
```
XPECMD> cfg.CheckDependencies 2
```

- Build a configuration
 - Example:


```
XPECMD> cfg.Build 0,"c:\images\myimage"
```

With XPECMD, you can create script files that update and build images directly from the command line. Here is an example set of commands that I used for building an existing configuration:

```
C:\Program Files\Windows Embedded\bin\>XPECMD
XPECMD>dbopen seanht
XPECMD>load cfg "f:\xpes\fp2007testc.slx"
XPECMD>cfg.ResolveDependencies 2
XPECMD>cfg.CheckDependencies 2
XPECMD>cfg.Build 2, "g:\Windows Embedded Images"
```



```
C:\WINDOWS\system32\cmd.exe - xpecon
C:\Program Files\Windows Embedded\bin>xpecon
Microsoft (R) Windows Script Host Version 5.6
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.
Console Application for Windows XP Embedded version 3.15.667 (CMI revision 578)
Copyright (C) 1999-2006 Microsoft Corp. All Rights Reserved.
Platform GUID      : <B784E719-C196-4DDB-B358-D9254426C38D>
Platform          : <not in database>
xpecon>dbopen seanht
Server           : seanht
Database         : MantisSQLDB
Generation       : <5C8FF6AF-AA8E-481B-A5AF-7A8E2B1F035B>
Open mode        : ReadImport <1>
xpecon>load cfg "f:\xpes\fp2007testc.slx"
Starting configuration activation at: 8/28/2006 10:01:38 AM...
CFDB: 1.72 secs, 172 reqs. LAB: 1.74 secs, 172 reqs, 0% hits, 0 entries
Activation complete, elapsed time: 15.22 seconds
No errors or warnings
xpecon>cfg.ResolveDependencies 2
xpecon>cfg.CheckDependencies 2
xpecon>cfg.build 2, "g:\Windows Embedded Images"
Warning 1116: Building unreleased component: "Intel PRO/100 UE MSM855 [Version 1.0, R41]"
Warning 1116: Building unreleased component: "Intel(R) 82801 PCI Bridge - 244E (SJJ Embedded Micro Solutions) [Version 6.0.0.1013, R28951]"
Warning 1116: Building unreleased component: "Intel(R) PRO/1000 CT Network Connection [Version 7.0.37.0, R21]"
Warning 1116: Building unreleased component: "RADEON 9200 (SJJ Embedded Micro Solutions) [Version 6.14.10.6497, R61]"
Warning 1116: Building unreleased component: "SoundMAX Integrated Digital Audio [Version 5.12.01.3583, R31]"
Warning 1116: Building unreleased component: "WinXP Promise FastTrak 378 (tm) Controller [Version 1.00.1.37, R21]"
Warning 1116: Building unreleased component: "ASUS P4C800-E Full [Version 1.0, R51]"
Warning 1115: This is an evaluation run-time image that is only valid for 90 to 180 days after it is first used.
xpecon>_
```

Fig 1 - Building an image from the command prompt.

Instead of typing in each command manually, a response file could be created. Each line in the response file is a command line to be performed by XPECMD. Here is an example, `myscript.rsp`, for the above command line entries:

```
dbopen seanht
Load cfg "f:\xpes\fp2007testc.slx"
cfg.ResolveDependencies 2
cfg.CheckDependencies 2
cfg.build 2, "g:\windows Embedded Images"
dbclose
quit
```

With a single response script file you can perform the entire open, dependency check, and build with a single command line:

```
XPECMD @myscript.rsp
```

Here are some important notes

1. There are two modes for running Windows scripts. Command mode must be set to run XPECMD:

```
cscript.exe //H:cscript
```

2. cfg is a variable for the configuration file. The variable name could be anything like config, myslx, mycfg, etc.
3. The build directory must be empty before the build command, or the build will fail.
4. Close the database when finished

2.2 USB 2.0 Boot

Another highly requested feature is the ability to boot from USB 2.0 Flash disk. Booting from USB has not been technically possible. The problem has been that XP/XPe doesn't support booting from a USB device. The UBS drivers load late in the boot cycle, thus the kernel doesn't have access to USB drives when it takes control during the boot process. There have been a few custom solutions to boot XPe from USB flash disks. These solutions require special hardware or special steps to get XPe to boot. FP2007 introduces a new EEF called USB 2.0 boot which officially adds Microsoft's support. Most important USB 2.0 boot will support any USB 2.0 flash disk. There are some system considerations that need to be address:

1. System must support USB 2.0 and USB 2.0 boot
2. In some cases, the BIOS needs to emulate the USB flash drive as a hard drive.
3. USB 2.0 boot brings in dependencies that will conflict with the USB components from the chipset vendor. The chipset vendor's components must be removed from the configuration.

The ability to boot from USB flash disk opens new possibilities for development deployment and field upgrade. For development deployment, a USB 2.0 Flash disk could hold an image that supports TCP/IP file sharing. The developer can boot to the USB flash disk prepare and share the local hard drive on the network and then transfer over a new XPe image. This would save a whole lot of time compared to the alternative WinPE/SDI solution currently offered. Field support staff can boot to a USB flash disk and copy down log files, troubleshoot hardware, or upgrade whole images or a few files.

2.3 File Based Write Filter (FBWF)

The Enhanced Write Filter (EWF) is the most important and popular EEF that comes with XPe, to date. EWF protects whole volumes or partitions from any writes. All writes to a EWF protected partition are sent to an overlay. The overlay can either be Disk or RAM / RAM-REG. Booting from read only media, preserving the operating system, protecting from viruses or file corruption, or extending the life of a flash disk are the main examples for using EWF. The problem with EWF is that it is an all or nothing approach. If you want to save data, you would need a separate partition or disk drive that is unprotected.

The new File Based Write Filter (FBWF) takes a different approach by protecting on a file level instead of the whole volume or partition. The developer can set specific files or directories to be unprotected. Any writes made to protected files are sent to RAM overlay. Any writes to unprotected files are passed through to the disk. Figure 2 shows how FBWF interacts with the protect disk

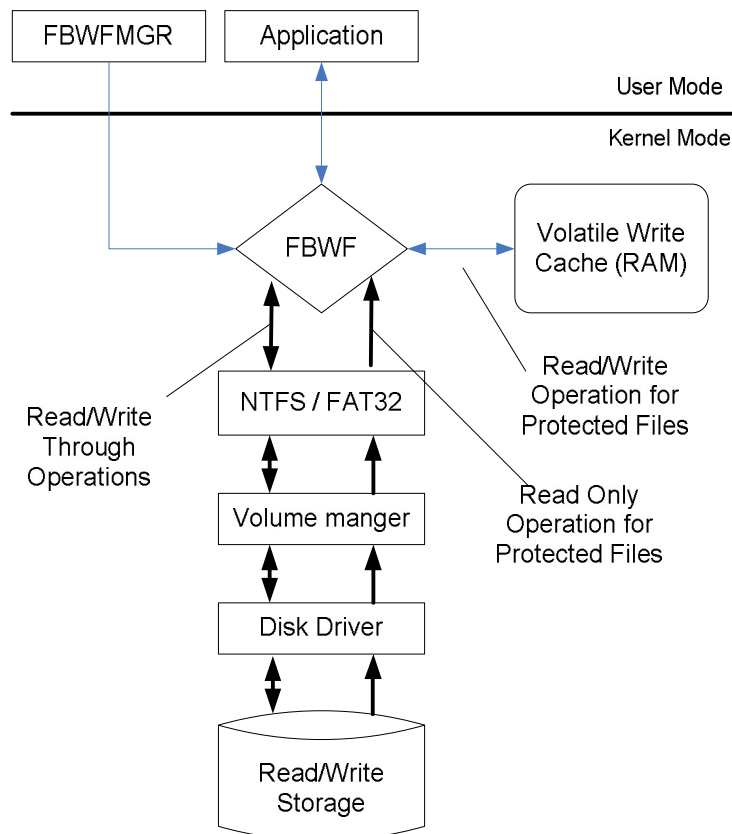


Fig 2 – FBWF Architecture

Now there are two filters to choose from. EWF will protect an entire volume, where almost nothing can get through. FBWF will protect on a file level, where you can select the files you want to pass through. Both filters can be used to protect the OS on a flash disk, but FBWF will allow you to pass through write access so data can be stored. FBWF also offers an API set so you can control FBWF from within your own application. FBWF will be ideal for those using databases and still protect the OS. When it comes to controlling files, FBWF is going to be a popular feature.

2.4 Registry Filter

One of the problems with EWF was Domain secret keys and TSCAL preservation. With EWF in place, the registry keys for these two items could be lost during a reboot, thus a failure to connect to the domain or loss of a TSCAL could occur. The Registry Filter was developed to preserve these two registry changes across reboots without requiring all registry changes in a file to be persisted. The Registry Filter is now available in FP2007.

2.5 New Components and EEF Category

There are several new components and categories that have been added to FP2007. A couple of components add support for orphan files that have been part of the repository, but didn't have a component.

2.5.1 . New EEFs category node in Target Designer

To help quickly locate EEFs, a new category has been added to the database and component browser: Embedded Enabling Features. This new category lists all the unique features (components) that are only found in XP Embedded.

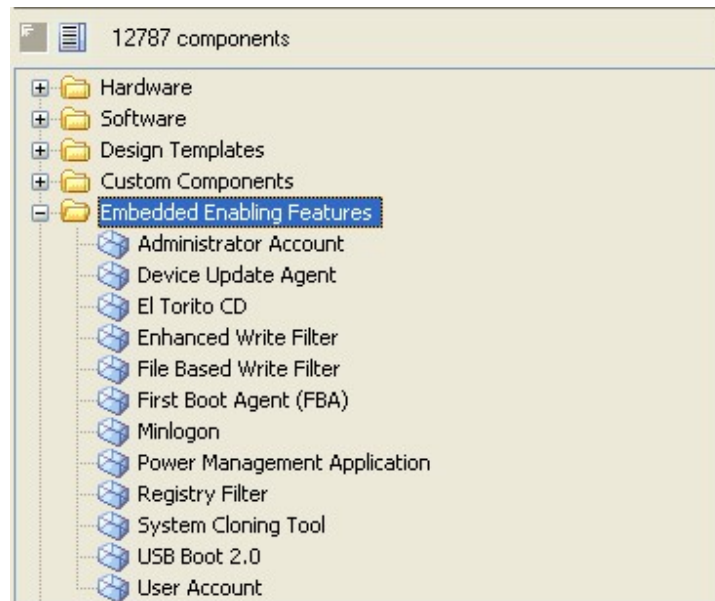


Fig 3 – New EEF Category

2.5.2 Microsoft SQL Express 2005 Edition

There are many little support components required to run SQL Express. This macro component adds only the support to install and run SQL Express or MSDE. You must still download the appropriate database engine from the download center on Microsoft.com separately.

2.5.3 Font and CodePage Application Compatibility components

These two new components have been added to support the look and feel issues that have occurred in various control panels and dialog applications.

- **Fonts Application Compatibility:** Adds all English fonts that are in the default install of the dependency database. However, this does not include font components normally installed from separate XPe MUI packages. It also does not include any non-English font in the default database.
- **Codepage Application Compatibility:** Adds all Codepage components that are in the default install of the dependency database.

2.5.4 Additional Locales and Keyboard Layouts

Additional keyboard layout components have been created to address certain locales. Bug fixes have also been done in some other keyboard layout components. Below is a list of keyboard layout components that have been added or modified:

Keyboard Layout	Description
Keyboard: Bosnian	New component
Keyboard: Finnish with Sami	New component
Keyboard: Swedish with Sami	Corrected display name registry key
Keyboard: Norwegian with Sami	Added as component dependency to Norwegian Language Support component
Keyboard: Sami Extended Norway	Added as component dependency to Norwegian Language Support component

Keyboard: Spanish	Added as component dependency to Spanish Language Support component
Keyboard: Spanish Variation	Added as component dependency to Spanish Language Support component

2.5.5 DirectShow Indeo Codecs

The Indeo Codec components were re-added to the database to maintain parity with XP-Pro. This includes the following new components:

- DirectShow Indeo3
- DirectShow Indeo4
- DirectShow Indeo5
- DirectShow Indeo5Audio

2.5.6 Previously orphaned file: Cacls.exe

Cacls.exe is now owned by "Primitive: Cacls.exe" component. This feature is a command line tool to control the Account Control Lists (ACLs).

2.5.7 Previously orphaned file: Gpupdate.exe

Gpupdate.exe is now owned by the "Group Policy Refresh Utility" component. This is a command line utility to manually update any changes that are made to group policies (some changes can be made immediately).

2.6 Component Changes

Many components offer customization settings within a configuration, but locating these settings have been a little inconsistent. Some components were using the expected Setting page and others had settings buried in the Advanced properties dialog. FP2007 corrects these inconsistencies and adds new features to some components.

2.6.1 EWF configuration UI Setting page Enhancements

The Enhanced Write Filter (EWF) component has had multiple changes made since it was released with the first version of XP Embedded. RAM-REG implementation solution, HORM, defragmentation issues are just a few changes that have been made along the way. The new EWF setting page features some updates to make configuring EWF easier:

- The configurable options have been modified to make it more intuitive and robust
- An option to disable Background Disk Defragmentation has been added. By default defragmentation is being disabled for your device to reduce the disk write activity, saving wear on the flash and saving overlay space.
- RAM (Reg) has been added as an Overlay Type option
- An option for HORM has been added and the need to manually add the resmany.dat file has been removed. You must still enable hibernation under the platform component in order to enable HORM.

Here is the new EWF settings page:

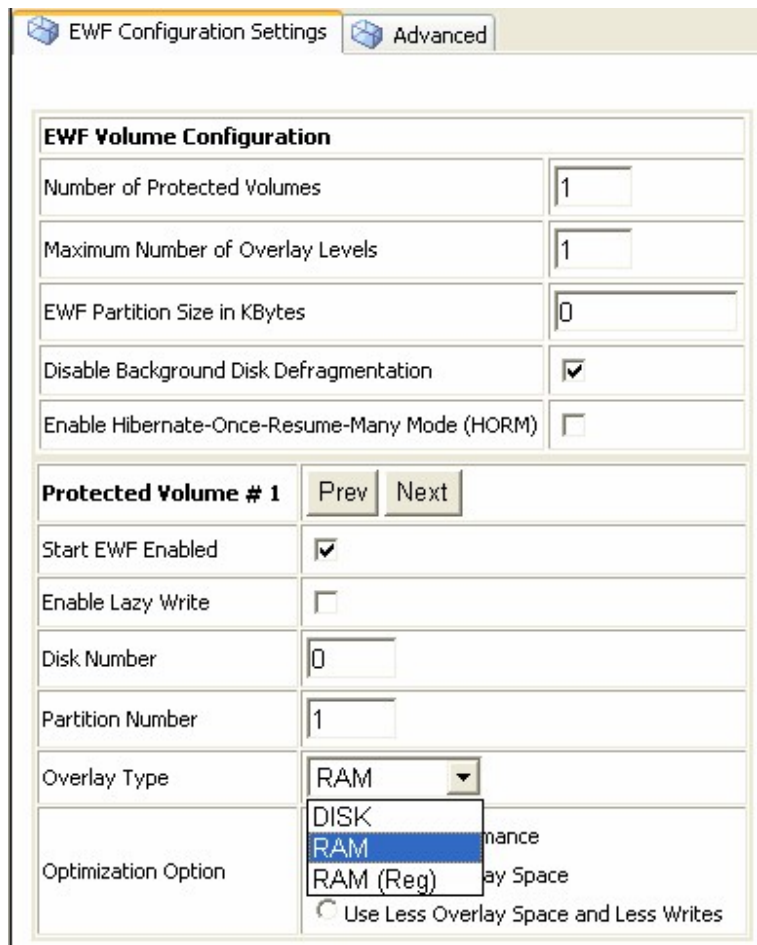


Fig 4 – EWF Settings Page

2.6.2 User Account and Administrator Account

A much needed update to the “User Account” and “Administrator Account” components has been made. Both now have a UI settings page that allows for easy setup of Passwords and other settings. Figure 5 shows the User account settings page

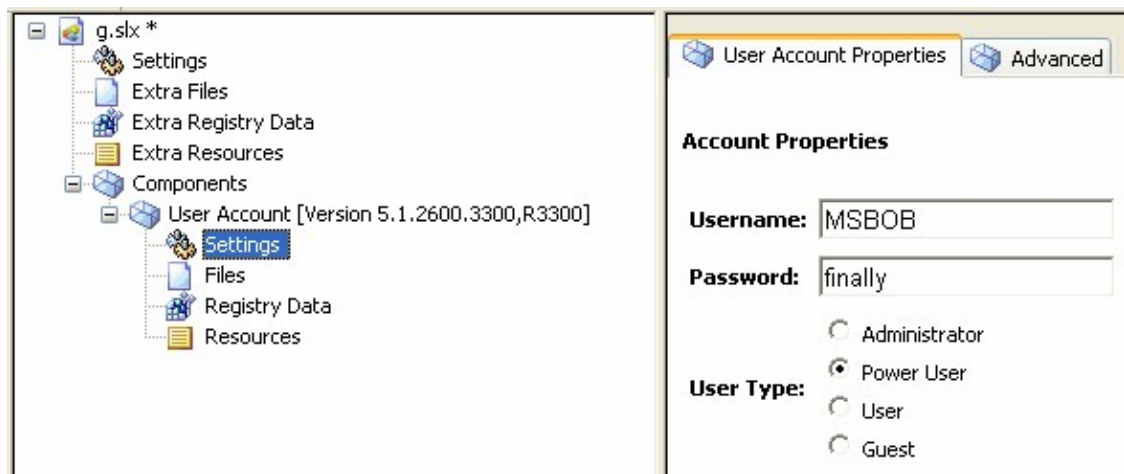


Fig 5 – User Account Settings

2.6.3 System Cloning Tool

The System Cloning Tool now includes a UI settings page that makes it easier set the various reseat options. Also, there is a place for the custom DLL and entry point.

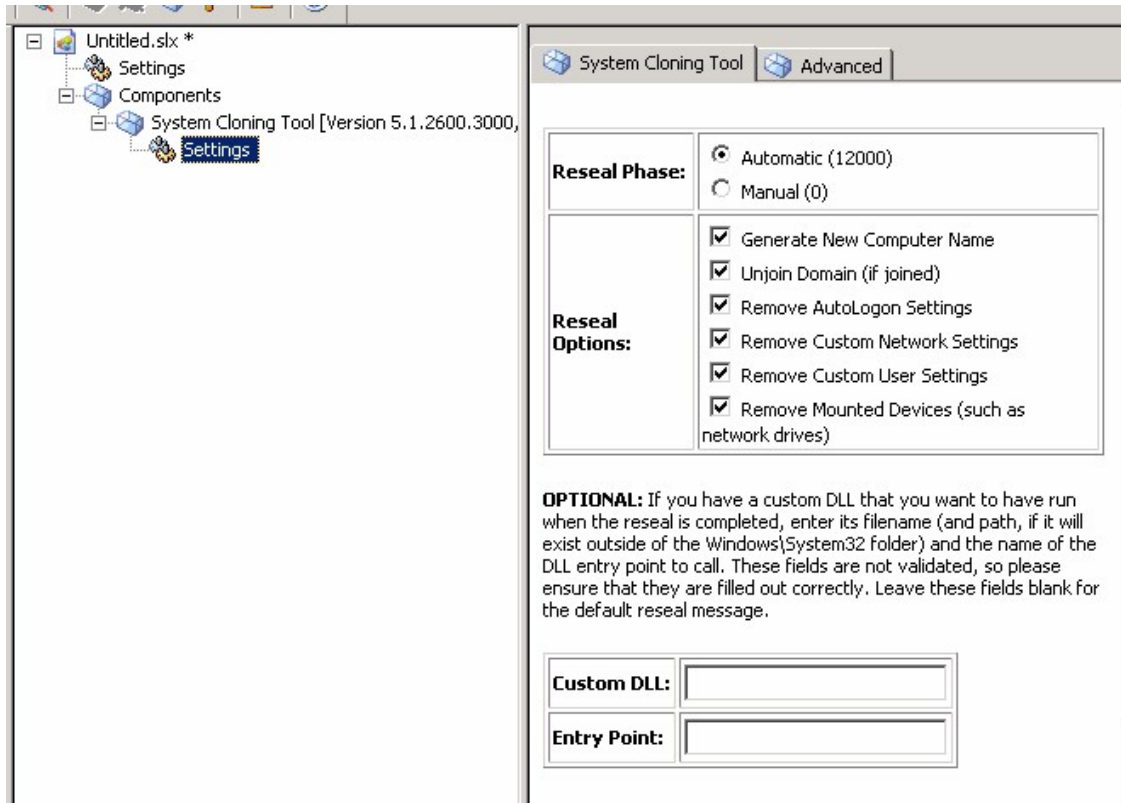


Fig 6 – System Cloning Tool Settings Page

2.6.4 Standard Start Menu Short Cuts updated

The **Standard Start Menu Short Cuts** component was updated to reflect changes made in SP2, adding of new features, and deprecating of features from SP1 to SP2.

2.6.5 Dr. Watson Debugger component

The **Dr. Watson Debugger** component has a new UI settings page to enable the debugger to be configured easier offline in Target Designer. The following illustration shows the new configurable settings of this component.

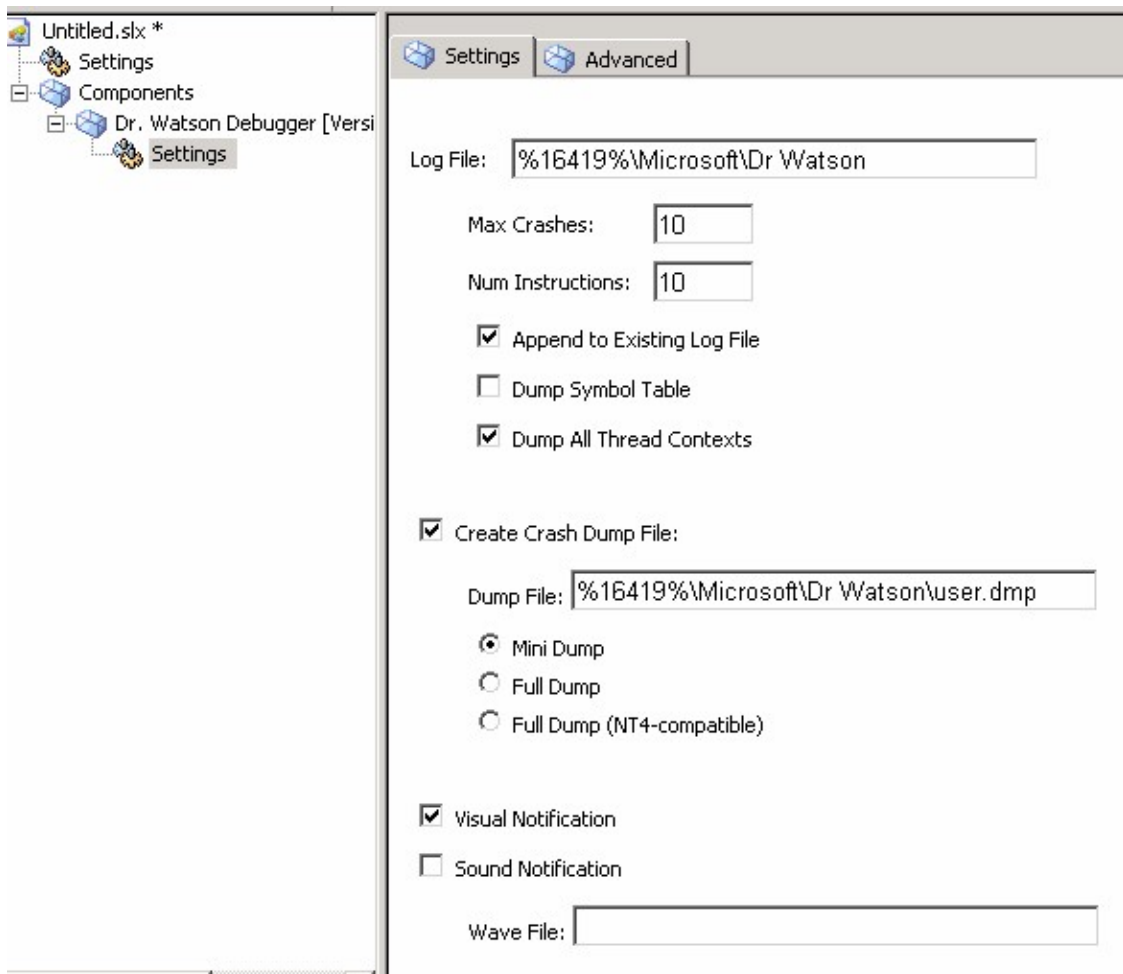


Fig 7 – Dr. Watson Setting's Page

2.6.6 Updated Windows Installer Service

The Windows Installer Service component has been updated to support the new Windows Installer version 3.1.

2.6.7 Component Descriptions for Primitives

All Primitive components now have some description. Previously there were many Primitive components with a blank description field in Target Designer. Adding these descriptions to the components enables the developer to make more intelligent decisions about feature components at design time.

2.7 Footprint Reduction

A major effort was made to fix some legacy component dependencies that brought in additional components that you might not want in a configuration. IE and Outlook Express are two examples. Effort was made to reconfigure dependencies that were left over when going from one service pack to the next. Some dependencies among components existed to satisfy a registry dependency, so the registry keys can be moved to a new shared component that other components can depend on. This removes the file dependency chain that previously was being caused.

Depending upon the features being selected (e.g. Outlook Express), certain platform build scenarios are able to achieve a footprint reduction from 3% to 10%. For example, here is a comparison between XP Embedded SP2 and FP2007 using the following components:

- Control Panel Applets – Custom component that has dependencies on many control panel applets.
- Registry Editor
- Runtime Quick Start Helper Macro
- Task Manager
- User Interface Core
- Virtual PC 2004

	XPe SP2 (KB)	FP2007 (KB)	Total Size Difference (KB) (%)
NTFS Compressed	171017	163134	7883 (4.61%)
Uncompressed	237524	226576	10948 (4.61%)

2.7.1 Internet Explorer Footprint Reduction

Many features in Windows depend on Internet Explorer, including these features:

- .NET Framework 1.1
- Help and Support Services
- HTML Rendering Engine
- IMAPI - CD Burning
- Internet Connection Wizard
- Microsoft Vector Graphics Rendering (VML)
- Outlook Express
- Shell Namespace Extensions
- Soap Client Runtime
- Windows .Net Messenger
- Windows Media Player 10.0
- Windows Media Player 9.0
- Windows Movie Maker

Those features have either a static or delay load dependency on one or more files out of the dozens of files owned by the Internet Explorer 6 component. Those dependent files were removed from the Internet Explorer component and are now 'Primitive' components.

Internet Explorer now depends on these primitives and the other higher level features like .Net Framework now also depends on just these Primitives.

You can still have a fully functional Internet Explorer feature, but you now must explicitly add that feature to your runtime, otherwise you will only pick up one or a couple of IE's individual file resources.

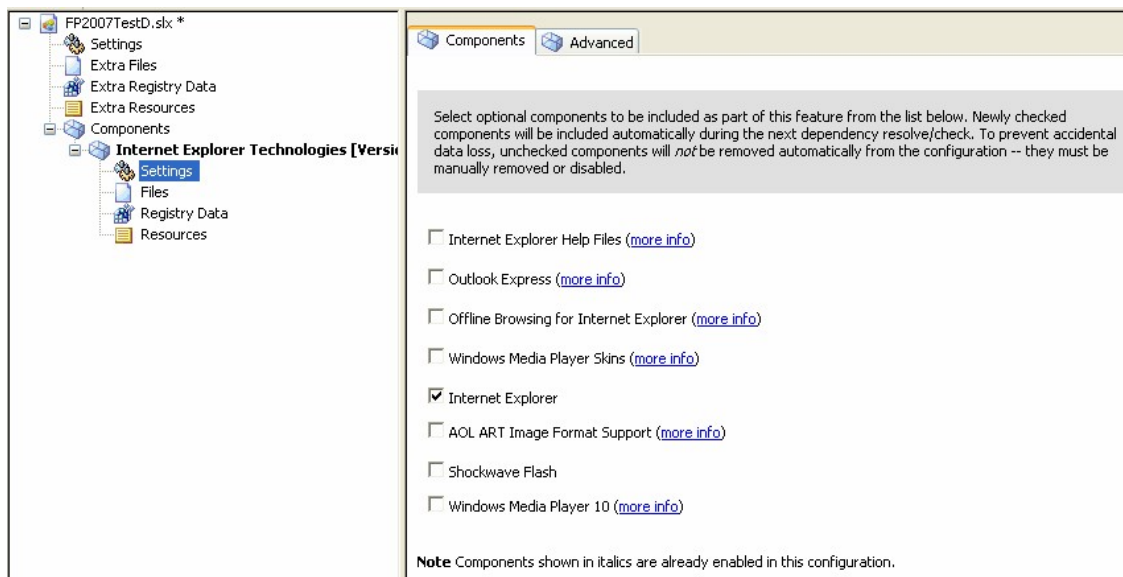


Fig 8 – IE Macro Settings

Depending upon the features being selected (e.g. Outlook Express), certain platform build scenario without a fully functional Internet Explorer feature may achieve a footprint reduction from 3% to 10%.

2.7.2 Footprint reduction for SMS Client scenarios

Remote Assistance Channel dependency on Help and Support was removed as an optional dependency. This can result in a footprint reduction of up to 30MB for some devices with SMS Client.

2.8 Summary

Feedback, Feedback, Feedback! FP2007 adds a number of little updates and great new features that have been requested over the past couple years. FBWF and USB 2.0 boot are the two sure hit features. The XPECMD.WSF adds the long awaited command line build that everyone has been looking for. The Setting page updates and footprint reduction round out the great new features that FP2007 will provide.