



Using a Hive Registry

White Paper 258

Mike Aanenson

Logic Product Development

Published: September 2004

Abstract

This white paper describes how to build a Windows CE® image with a hive-based registry using a Logic Product Development Board Support Package.

This file contains source code, ideas, techniques, and information (the Information) which are Proprietary and Confidential Information of Logic Product Development, Inc. This information may not be used by or disclosed to any third party except under written license, and shall be subject to the limitations prescribed under license.

No warranties of any nature are extended by this document. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed license or agreement to purchase or lease equipments. The only warranties made by Logic Product Development, if any, with respect to the products described in this document are set forth in such license or agreement. Logic Product Development cannot accept any financial or other responsibility that may be the result of your use of the information in this document or software material, including direct, indirect, special or consequential damages.

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

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

© Copyright 2002, Logic Product Development, Inc. All Rights Reserved.

REVISION HISTORY

REV	EDITOR	DESCRIPTION	APPROVAL	DATE
A	Mike Aanenson	Release	HAR	9/24/2004

1 Introduction

This document first explains why one might prefer to use a hive registry instead of a RAM registry, and then describes how to build a WindowsCE® image with a hive registry using a Logic Product Development BSP (Board Support Package).

The following topics will be covered in this white paper:

- Windows CE registry basics
- Hive registry and RAM registry comparison
- Changing a RAM registry to a hive registry in Windows CE Platform Builder

For further information, consult Microsoft's Windows CE documentation available online.

2 Overview

A 'hive' registry can be used to store system settings in non-volatile memory on a Windows CE device. The hive is stored on a file system (like CompactFlash or on-board NOR flash), thus saving your settings in non-volatile memory. In contrast, a RAM-based registry only stores these settings into RAM, so once power is shut off, the settings are lost.

Logic Product Development's Windows CE BSP software can easily be modified to allow for the use of a hive registry. To do so, the hive-based registry catalog component must be added to the Platform Builder workspace, and specific registry settings must be present in the registry areas that control the file system interface that the hive file will be located on.

3 Windows CE Registry

The CE Registry can be of two types: RAM or hive.

The RAM registry is stored in the Windows CE object-store, which is in the system RAM. The registry only persists as long as there is power to RAM. As there is no need to write the data to non-volatile memory, a RAM-based registry is quick to launch, write to, and flushed. It can be used if there is no need to save updated registry settings or if there is battery-backed RAM that can maintain RAM contents during a power loss.

The hive registry is stored on a file system – usually on a non-volatile storage device. This means that the current state of the registry can be saved even when power to the device has been turned off. That's the good news. The bad news is that it can take longer to boot and the registry data must be written to the file system ("flushed") at some point before shutting down, which can slow down the shutdown process.

4 Using Hive over RAM Registry

Why use a hive registry instead of a RAM registry? The hive registry gives you the ability to save the system settings to non-volatile memory. For example, after calibrating the touch screen, the new calibration settings are placed in the registry. With the default RAM registry, these settings will only be good as long as the device has power (or as long as battery backup is maintained – if the necessary circuitry is implemented). If a hive registry is used, these settings are saved into non-volatile memory, and will be available after a power cycle.

5 Registry Settings / Platform Builder Configuration

When a hive registry is used, it is useful to understand that the system will actually be using two (or more) registries: the boot registry and the system/user registry. The boot registry will always be loaded first and cannot be modified. The boot registry is used to initialize enough interfaces so that the system/user portions can be loaded. The system/user registry is actually the portion stored in the designated file system device and can be modified. The hive registry also supports multi-user logins, allowing multiple user versions to be loaded based on the logged-in username.

The registry settings that are used by the device to properly load a hive registry are listed in detail below. Examples are shown after a brief explanation of each. The *.reg files included in a LogicPD BSP contain most of these settings already and may not require any modifications. These files are all viewable and editable. To have a good understanding of using the hive registry values, please scan through these files.

The hive registry will never work unless it is chosen from the list of components in Platform Builder. The RAM registry is the default type. The hive registry catalog component can be found at this location in the catalog: 'Catalog/Core OS/Display based devices/File Systems and Data Store/Registry Storage'. You will need to add this to your project. See Figure 5:1, below.

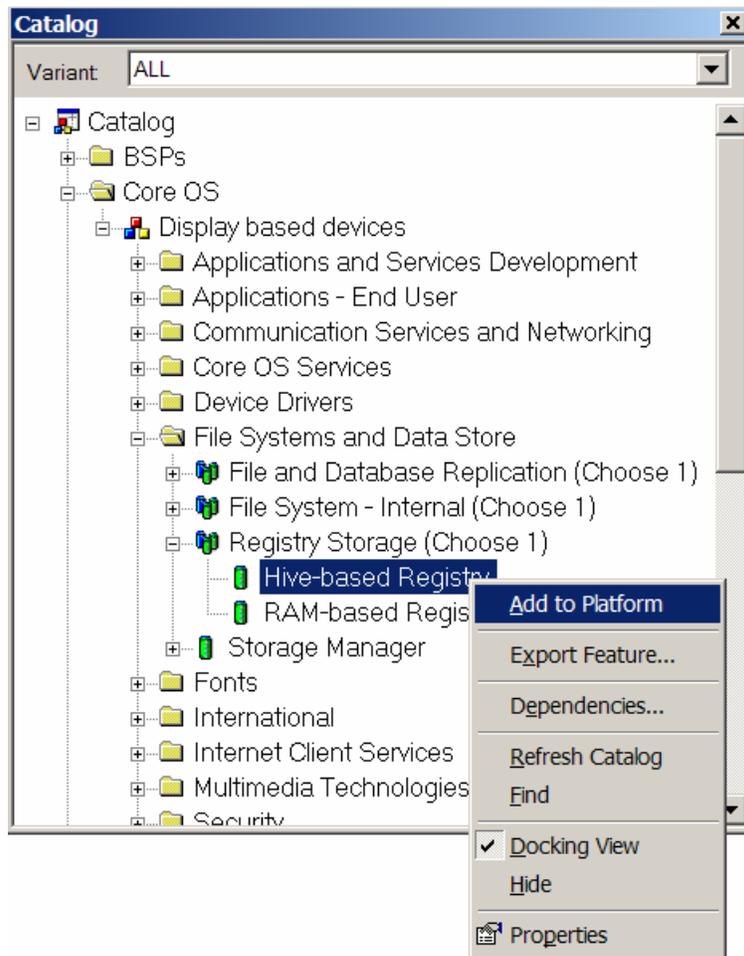


Figure 5.1: Registry Type Selection: “You will need to add this to your project.”

5.1 Registry Entries: Brackets

Registry entries to be loaded as part of the hive boot registry must be bracketed by "; HIVE BOOT SECTION" and "; END HIVE BOOT SECTION." Registry entries found between these hive boot registry brackets will be loaded in the first boot phase, allowing access to the drivers needed to access the system/user portions of the hive registry. You will notice these comments throughout the applicable LogicPD *.reg files. See example below:

5.1.1 Example

```
; HIVE BOOT SECTION  
  
~ REGISTRY ENTRIES NEEDED FOR PERSISTENT REGISTRY STORAGE ~  
  
; END HIVE BOOT SECTION
```

5.2 Registry Entries: Location and Name of the Hive Registry

The following registry entries (typically in platform.reg) are used to determine the location and name of the hive and whether to start the device manager to start drivers to use with the hive registry. (If RAM registry is used, this can be '0') When starting the DevMgr, the necessary drivers are started to access the hive. The default hive to be loaded can also be specified by using the 'DefaultUser' key, which selects the hive based on the user.

The "SystemHive" entry for the hive simply specifies the relative location of the hive registry files on the storage device. When the registry entry ' "MountFlags"=dword:2 ' exists in a device's registry settings, it tells the system that it should use that device to store the hive files on.

5.2.1 Example platform.reg entry:

```
[HKEY_LOCAL_MACHINE\init\BootVars]  
    "SystemHive"="PSM\System.hv"           ; Name and location on file system of the hive  
    "Start DevMgr"=dword:1                 ; 1 = start device manager to access  
                                           ; the file system  
    "DefaultUser"="WinCE_User"            ; determine the default user hive to load
```

5.2.2 Examples from specific driver .reg files

Note: This registry entry should only be set for one device. If it is set for multiple devices, the hive may be located on any of them.

```
; if on memory-mapped compact flash card  
[HKEY_LOCAL_MACHINE\System\StorageManager\Profiles\ATAMEMCF]  
    "MountFlags"=dword:2  
  
; if on pcmcia/compact flash card  
[HKEY_LOCAL_MACHINE\System\StorageManager\Profiles\PCMCIA]  
    "MountFlags"=dword:2  
  
; if on flashfx disk  
[HKEY_LOCAL_MACHINE\System\StorageManager\Profiles\FlashFXDisk1Profile]  
    "MountFlags"=dword:2  
  
; if on flashfmd disk  
[HKEY_LOCAL_MACHINE\System\StorageManager\Profiles\lpd_MSFLASHFMD]  
    "MountFlags"=dword:2
```

5.3 Registry Entries: Flags

The ' "Flags"=dword:1000 ' entry must also be set on any devices that need to be loaded in the first boot phase. This tells the Device Manager that they are to be loaded using the boot registry and are not to be loaded in the second boot-phase with the system registry.

5.3.1 Examples from specific driver .reg files:

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\FlashFX]
    "Flags"=dword:1000
```

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\lpd_ATAMEMCF]
    "Flags"=dword:1000
```

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\PCMCIA]
    "Flags"=dword:1000
```

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\lpd_MSFLASHFMD]
    "Flags"=dword:1000
```

5.4 Hive Registry Flush

One thing that is very important to know about when using a hive registry is that the persistent hive is not updated automatically when a registry change is made. The registry data must be "flushed" to the hive for changes to be saved. On a suspend event or a graceful software shutdown, this will be done automatically (however, when power is lost to a system before a graceful shutdown can occur, this is not the case and registry changes may be lost). One way to make sure certain data is flushed is to use the *RegFlushKey()* function. This will flush the specified registry key. Please note that this should be done only when necessary as the time associated with flushing the registry to non-volatile memory can bog down the performance of the system.

Further documentation: For more information, please look in the Windows CE documentation at <http://www.msdn.microsoft.com/> or in Platform Builder 'help'. Search for "Hive-Based Registry" or "Hive-Based Registry Startup Sequence" to locate graph of the start-up sequence.

6 Example:

The following entries would be required if the hive registry was to be placed on a CompactFlash card. This example uses Logic PD's memory mapped CF card driver.

Please note that the settings as specified below are the default entries for Logic's software and should not need modifications if using the memory mapped CompactFlash device as the hive registry repository.

The 'IF SYSGEN_FSREGHIVE' lines check if the HIVE registry has been added to the project. If not, the code contained by the IF/ENDIF statements will not be added as they are not used.

(platform.reg entry)

```
IF SYSGEN_FSREGHIVE
; HIVE BOOT SECTION
[HKEY_LOCAL_MACHINE\init\BootVars]
  "SystemHive"="PSM\System.hv" ; use whatever name you want
  "Start DevMgr"=dword:1 ; 1-start device manager to access the file system
  "DefaultUser"="WinCE_User"
; "RegistryFlags"=dword:1 ; if this is uncommented, the registry will be flushed every time regclosekey is called
; END HIVE BOOT SECTION

ENDIF
```

(lpd_ATAMEMCF_rel_100.reg entry)

```
IF SYSGEN_FSREGHIVE
; HIVE BOOT SECTION
ENDIF

[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\lpd_ATAMEMCF]
  "Prefix"="DSK"
  "Dll"="lpd_ATAMEMCF_rel_100.dll"
  "Ioctl"=dword:4
  "BaseAddress"=dword: 60201800
  "BusWidth"=dword:10
  "ScratchReg"=dword:0
  "Profile"="lpd_ATAMEMCF"
  "FSD"="FATFSD.DLL"
  "Class"="{A4E7EDDA-E575-4252-9D6B-4195D48BB865}"

IF SYSGEN_FSREGHIVE ; Only define if HIVE registry is used
  "Flags"=dword:1000
ENDIF

; In the subkey here "MountFlags" must be a "2" if you are going to use the
; CF card as the HIVE repository.
[HKEY_LOCAL_MACHINE\System\StorageManager\Profiles\lpd_ATAMEMCF]
  "Name"="lpd_ATAMEMCF/Compact Flash Device"
  "Folder"=LOC_STORE_PCMCIA_FOLDER
IF SYSGEN_FSREGHIVE
  "MountFlags"=dword:2
ENDIF

IF SYSGEN_FSREGHIVE
; END HIVE BOOT SECTION
ENDIF
```