WinCE image and files for CEView v6.1+SP2

List of CEView Files and WindowsCE Image Settings for CEView

Introduction

This document describes the components (files) of CEView runtime software for Microsoft WindowsCE (WinCE) platform, such as the image requirements in order to run CEView on the WinCE device.

The information in this document is useful to configure a third-party WinCE device (for example: iPAQ) in order to run CEView. Also, it provides the necessary information for OEM (hardware manufacturers) to configure the WinCE image of their devices in order to receive the advantage of all tools provided by CEView.

> ✎ **Note:** CEView is the runtime program, which executes the applications developed with InduSoft Web Studio, on the WinCE operating system.

Versions and Platforms

The table below shows the WinCE platforms supported by each version of CEView:

CEView version	WinCE version	Processor Type
v3.0	v2.11 / v2.12	ARM, MIPS, PPC, SH3, SH4, X86
V4.1	v2.11 / v2.12	ARM, MIPS, PPC, SH3, SH4, X86
V4.4	v3.00	ARM, MIPS, MIPSFP, PPC, SH3, SH4, THUMB, X86
V5.1	v3.00	ARM, MIPS, MIPSFP, PPC, SH3, SH4, THUMB, X86, Pocket-ARM, Pocket-MIPS, Pocket-SH3
V5.1+SP2	v3.00	ARM, MIPS, MIPSFP, PPC, SH3, SH4, THUMB, X86, Pocket-ARM, Pocket-MIPS, Pocket-SH3
	V4.00 (.NET)	ARMV4, ARM4I, ARM4T, MIPSII, MIPSII_FP, MIPSIV, MIPSIV_FP, SH3, SH4, X86
V5.1+SP3	v3.00	ARM, MIPS, MIPSFP, PPC, SH3, SH4, THUMB, X86, Pocket-ARM, Pocket-MIPS, Pocket-SH3, Pocket2002-ARM, Pocket2002-X86
	V4.00 (.NET)	ARMV4, ARM4I, ARM4T, MIPSII, MIPSII_FP, MIPSIV, MIPSIV_FP, SH3, SH4, X86
	V4.10 (.NET)	ARMV4, ARMV4I, ARMV4T, MIPS16, MIPSII, MIPSII_FP, MIPSIV, MIPSIV_FP, SH3, SH4, X86
V6.0 and higher	v3.00	ARM, MIPS, MIPSFP, PPC, SH3, SH4, THUMB, X86, Pocket-ARM, Pocket-MIPS, Pocket-SH3, Pocket2002-ARM, Pocket2002-X86
	V4.00 (.NET)	ARMV4, ARMV4I, ARMV4T, MIPSII, MIPSII_FP, MIPSIV, MIPSIV_FP, SH3, SH4, X86, Pocket2003-ARMV4

> ✎ **Note:** The CEView program compiled for WinCE v4.0 (.NET) is supported by WinCE v4.x and WinCE v5.0.

Architecture

InduSoft Web Studio is the workbench used to configure the application (Development Station), under WinNT/2000/XP. CEView is the runtime software that runs the applications (configured with InduSoft Web Studio) under the WinCE operating system.
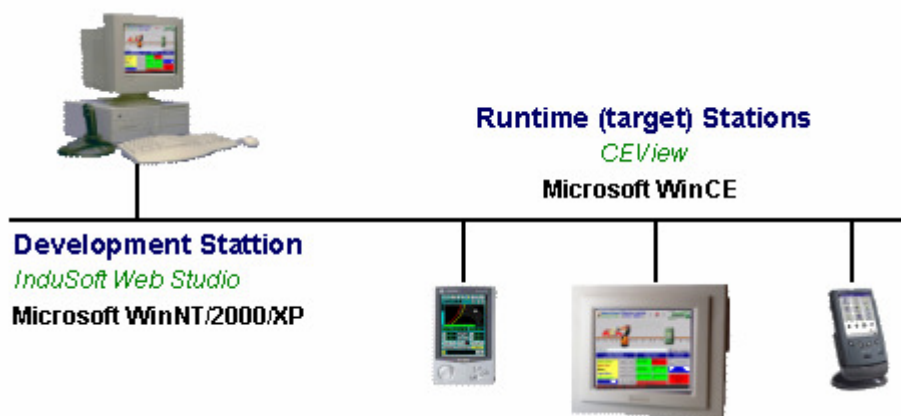
When installing InduSoft Web Studio on the development station (WinNT/2000/XP), the CEView files are stored in the *…\InduSoft Web Studio\Redist\<OS Version>\<Platform>* folder, where:

- **<OS Version>** = WinCE version. The files for WinCE v3.0 are stored in the *\CEView* sub-folder. The files for WinCE v4.0 are stored in the *\Wince 4.0* subfolder and so forth.
- **<Platform>** = Processor type.

For instance, the CEview runtime files for WinCEv3.00-x86 platform are stored in the *…\InduSoft Web Studio\Redist\CEView\x86* folder. The CEView runtime files for the WinCE.NET(v4.0)-sh4 platform are stored in the *…\InduSoft Web Studio\Redist\Wince 4.0\sh4* folder, and so forth.

InduSoft provides tools to easily download the CEView files (such as the application files) to the WinCE device (target station) using either a serial link (RS232) or a TCP/IP (for example: Ethernet) link – refer to InduSoft Web Studio Technical Reference Manual for further information. The files can also be copied manually to the WinCE device – in this case, it is necessary to adjust some settings manually (for example, edit a few INI files) before downloading the files to the WinCE device. The next topics provide further information about the configuration of these files.

> ✎ **Note:** It is recommended that you use the InduSoft tools to download files from the development station to the target (WinCE) devices. However, if you cannot use InduSoft tools to transfer files to the WinCE device (for example: if you are using a USB interface), we suggest you use the Microsoft ActiveSync utility to transfer files to the WinCE devices.

CEView and Application files

Usually, most of the folders on the WinCE device are allocated on the volatile memory (for example, the data stored on these folders is lost when the device is powered off). In order to store data that should not be lost when the device is powered off, the WinCE devices provide at least one folder on the non-volatile memory area. Since each hardware manufacturer defines a different name for this folder, we will reference to it as *<non-volatile>* folder in this document. For practical effects, this reference must be replaced by the real name of the non-volatile folder of the WinCE device used.

In this section, we classified the files in five groups:
- **Program files**: CEView runtime files used to execute the application.
- **Driver files**: CEView communication drivers to exchange data with the plant-floor devices (PLCs, I/Os, PC Based Control, Motion Controllers, etc).
- **Utilities/Auxiliary files**: Configuration files and utilities which execute auxiliary tasks.
- **CEView image files**: Operating system files required by CEView, stored on the image of the WinCE device.
- **Application Files**: Files created by the user. These files are created using InduSoft Web Studio (workbench) on the development station.

Program Files (Path: \<non-volatile>\BIN\ )	
File Name	**Description**
CEView.exe	Main program file. It executes the InduSoft application under the WinCE device.
Viewer.exe	Viewer module program file.
CEServer.exe	Remote Agent. This program should NOT be executed from this path. It must be copied into the *<non-volatile>* folder before being executed. It is stored in this folder for InduSoft Web Studio when updating the CEView files remotely because the CEServer.exe from the *<non-volatile>* folder is in use and cannot be replaced during the update procedure.
OPCSvrReg.exe	Utility program used to set/get the CLS-ID code of OPC Servers on the WinCE device.
Unipad.exe	Text files viewer.
PServer.exe	Printer Server utility, used to print text files from CEView application.
StADOSvrCE.exe	Studio ADO Gateway used to exchange data with SQL Relational databases via ADO Providers.
UniSoft.dll	Main library used by CEView.exe.
IndScriptCompiler.dll	Library used by CEView.exe to compile the scripts configured in VBScript.
IndScriptRun.dll	Library used by CEView.exe to execute the scripts configured in VBScript.
StudioADO.dll	Main library used by StADOSvrCE.exe.
IndVKStd.dll	Virtual Keyboard library. When using the WinCE device as a Web Thin Client, this file must be stored in the same folder where ISSymbolCE.ocx is registered in order to use the Virtual Keyboards on the Web Thin Client.
VKxx.INI	Lay-out for the Virtual Keyboard (VK). The chars and commands available in the key of the Virtual Keyboard(s) can be customized by the VKxx.INI file(s). Using these files, it is possible to customize the VKs for different languages – e.g.: VKEN.INI (English lay-out), VKGE.INI (German lay-out) and so forth.
CELog.dll	Library used to log the application messages and events when debugging the application.
IndRAS.dll	CEView library to interface with the WinCE RAS (Remote Access Service), used for Dial-Up connections.

UniComm.dll	CEView library for communication drivers with Serial or Infra-Red interface.
UniTCPIP.dll	CEView library for communication driver with TCP/IP interface.
Studio Toolkit.dll	CEView Toolkit library for CEView.
IndMail.dll	SMTP mail interface library. This file is necessary to execute the Email functions from CEView.
IndHTTP.dll	Main library used by the Studio Web Gateway (IndWebSvr.dll).
DrvStd.dll	Library for standard configuration of communication drivers. The communication drivers based on this library require this file to run.
IndHST.dll	Library for handling of History Files saved in the proprietary format.
AGLink.dll	Library used by the communication drivers that implement the Siemens protocols (SIPPI, SIEME and SIETH).
ISSymbolCE.ocx	Web Thin Client ActiveX controller.
Tagi.bin	Internal tags list (time, date, second, username, etc.).
Pserver.ini	Pserver settings. This file is not installed with IWS.
CEView.ini	CEView program settings.
Unipad.ini	Unipad program settings. This file is not installed with IWS.
StADOSvrCE.ini	Studio ADO Gateway settings.
IndRAS.ini	Description for the RAS (Remote Access Services) error codes.
Unicomm.msg	Standard error codes and messages generated by the UniComm.dll.
Recipe.xsl	Style sheet file for XML recipe files

✎ **Note:** The following files are necessary to run any CEView application: CEView.exe, Viewer.exe, Unisoft.dll, IndScriptCompiler.dll and IndScriptRun.dll. The other files are used depending on the features used in the application.

Driver Files (Path: \<non-volatile>\DRV\ )	
File Name	**Description**
<DriverName>.dll	Communication Driver file which implements a specific protocol (for example: Modbus, DF1, etc.).
<DriverName>.ini	Default settings for the communication driver.
<DriverName>.msg	Custom error codes and messages generated by the <DriverName>.dll.

✎ **Note:** The communication drivers are extremely modular in the CEView structure. CEView requires three files (mentioned in the previous table) for each driver. When CEView is launched, it links to the proper communication driver(s) dynamically, according to the application settings.

Utilities/Auxiliary files (Path: \<non-volatile>\ )	
File Name	**Description**
CEWebServer.exe	Web Server (HTTP Server). The home directory is the same folder from where this program is executed. It cannot be executed in devices that have their own embedded Web Server.

WebGtw.exe	Installer for the Studio Web Gateway (IndWebSvr.dll).
OPCProxy.dll	Proxy file for OPC out-proc communication (using DCOM).
StOleProxy.dll	Proxy file for Remote LogWin and Database Spy (OLE).
IndWebSvc.dll	Studio Web Gateway (Web Service for Microsoft IIS under WinCE), used to encapsulate the messages from the TCP/IP Server module of CEView under HTTP (or HTTPS) for remote Web Thin Clients.
MyRapi.dll	Library used to exchange data between the development PC and the WinCE device (usually, PocketPC) using Microsoft ActiveSync.
MyRapiGetPlatformType.dll	Library used to link the development PC to the WinCE device (usually, PocketPC) using Microsoft ActiveSync.
OPCComn_ps.dll	OPC Communication library.
PMControl.ocx	ActiveX controls used with projects converted with the Import Wizard for PanelMat®.
CEApp.ini	Path of the current application. When CEView is executed, it reads from this file the path and name of the application which will be executed. When the application is downloaded to the WinCE device, the Remote Agent updates this file automatically. This file is not installed with IWS.

✍ **Note:** The ISSymbolCE.ocx, OPCProxy.dll, StOleProxy.dll and ISSymbolCE.ocx files must be registered on the WinCE device (for example: using the "regsvrce <Path+FileName>" command). Also, the Studio Web Gateway (WebGtw.exe) must be installed in order to use this feature with remote Web Thin Client. After registering any file, it is necessary to save the registry information in order to keep the new settings after rebooting the device.

Application Files (Path: \<non-volatile>\<Application Name>\ )	
File Name	**Description**
<Application>.app	Text file with the application main settings.
\Alarm	Default path for the history alarm files.
\Config	Tasks worksheets (Maths, Scheds, etc.).
\Database	Application tags database files (tags created by the user).
\Hst	Default path for the history trend files.
\Screen	Application screen files (displays).
\Symbol	Application symbol objects (customized by the user).
\Web	Web files used for the Web Thin Client station.

✍ **Note:** The application files are created by the user on the development station, using InduSoft Web Studio software (workbench).

CEView Settings

> ✎ **Note:** In the examples below it is necessary to consider that:
> - The *<Non-volatile Folder>* text must be replaced by the non-volatile folder name, according with each CE BOX.
> - The <Application Folder> text must be replaced by the application folder name.
> - The <Application File> text must be replaced by the application file name (.APP file).

▪ \<**non-volatile**>**PSERVER.INI**

Syntax
[Font] FontName=<Name of the font used to print plain text reports from CEView> Height=-<Size of the font used to print plain text reports from CEView>

Example
[Font] FontName=Arial Height=-14

▪ \<**non-volatile**>**CESERVER.INI**

Syntax
[Setup] Device=<Device connection: 0=Serial Port; 1=TCP/IP> SerialPort=<Seral Port: 0=COM1 ; 1=COM2 ; 2=COM3 ; 3=COM4> RunCEViewOnStartup=<Flag: 0=Do not start automatically ; 1=Start automatically> SleepTime=<Sleep time (milliseconds) executed by CEServer to scan the COM ports after pressing the Setup button> SerialBaudRate=<Baud-rate for serial communication> SerialRTS=<RTS HandShake signal: 0=Disabled ; 1=Enabled> SerialCTS=<CTS HandShake signal: 0=Disabled ; 1=Enabled> [Communication] Delay=<Delay time (in milliseconds) between the TCP/IP packages during the downloading and uploading.> PackSize=<Maximum size (Bytes) of the TCP/IP packages during the downloading and uploading> [Remote] Port=<TCP/IP Port number for TCP/IP connections> [OEM] RegFileName=<Register file name (*.REG) where the registry settings are saved. This parameter is used to save the license settings to the registry> ProductName=<Label displayed in the title bar of the CEServer.exe program> [Target] Count=<Number of License Types supported> [Target<N>] ProductID=<Product ID code>

Name=<Product Type Name>

[Watchdog]
Enable=<Watchdog features: 0=Disabled ; 1=Enabled>
FailRecovery=<Re-start CEView automatically if a Fatal Error occurs: 0=Disabled ; 1=Enabled>
FailRecoveryMsg=<Fatal Error message for Fail Recovery. Default value = Fatal Application Error)
TimeOut=<Time, in minutes, waited for Remote Agent before re-starting CEView automatically when it is not running>

Example

[Setup]
Device=1
SerialPort=0
RunCEViewOnStartup=1
SleepTime=500
SerialBaudRate=19200
SerialRTS=0
SerialCTS=0

[Communication]
Delay=250
PackSize=1024

[Remote]
Port=4322

[OEM]
RegFileName=\HardDisk\WinCE\WinCE.REG
ProductName=Remote Agent

[Target]
Count=3

[Target1]
ProductID=4
Name=CEView Lite

[Target2]
ProductID=10
Name=CEView Standard

[Target3]
ProductID=23
Name=CEView PRO

[Watchdog]
Enable=1
FailRecovery=1
FailRecoveryMsg=Fatal Application Error
TimeOut=15

▪ **\<non-volatile>\CEAPP.INI**

Syntax
[Application] Directory=\<Non-volatile Folder>\<Application Folder> Configuration=\<Non-volatile Folder>\<Application Folder>\<Application File>

Example
[Application] Directory=\Disk\Demo\ Configuration=\Disk\Demo\Demo.app

▪ **\<non-volatile>\BIN\CEVIEW.INI**

Syntax
[Install] UnisoftPath=<CEView program path> Version=<CEView version> Language=<CEView language: EN=English> TestCompatibility=<Check application version before running it: 0=Disabled(default); 1=Enabled> SaveAsUnicode=<Save files in the Unicode format: 0=Disabled ; 1=Enabled(default)> WarningLowMemory=<Launches warning message if free memory lower than 1MB: 0=Disabled(default) ; 1=Enabled> [OEM] CompanyID=<CompanyID Code> BlindDevice=<Warning/Error messages: 0=Display all messages (default) ; 1=Do NOT display any message> SplashWnd=<Name of the bitmap file that must be launched as Splash Window, if any> SplashWndTime=<Time, in milliseconds, while the Splash Window is displayed> [Period] Math=<Minimum time (in milliseconds) between math worksheets executions> Alarm=<Minimum time (in milliseconds) between alarm worksheets executions> Trend=<Minimum time (in milliseconds) between trend worksheets executions> Sched=<Minimum time (in milliseconds) between scheduler worksheets executions> Recipe=<Minimum time (in milliseconds) between recipe worksheets executions> Report=<Minimum time (in milliseconds) between report worksheets executions> AppBuilder=<Minimum time (in milliseconds) between screen editor module executions> Viewer=<Minimum time (in milliseconds) between Viewer module executions> BlinkFast=<period (in Milliseconds) of the BlinkFast parameter from the color dynamics> BlinkSlow=<period (in Milliseconds) of the BlinkFast parameter from the color dynamics> Driver=<Minimum time (in milliseconds) between Driver module executions> OPCClient=<Minimum time (in milliseconds) between Driver module executions> Script=<Minimum time (in milliseconds) between Script module executions> [TimeSlice] Math=<Maximum time of interrupt execution of the Math worksheets> Driver=<Maximum time of interrupt execution of the driver module> Viewer=<Maximum time of interrupt execution of the Viewer module>

OPCClient=<Maximum time of interrupt execution of the OPC Client module>
Script=<Maximum time of interrupt execution of the Script module>

[International]
Order=<Date order format: MDY=Month/Day/Year ; DMY=Day/Month/Year>
Separator=<Date Separator>

Example

[Install]
UnisoftPath=\Hard Disk\CEView
Version=6.1.01
Language=EN
TestCompatibility=0
SaveAsUnicode=1
WarningLowMemory=0

[OEM]
CompanyID=70
BlindDevice=0
SplashWnd=splash.bmp
SplashWndTime=5000

[Period]
Math=100
Alarm=200
Trend=1000
Sched=1000
Recipe=2000
Report=2000
AppBuilder=50
Viewer=50
BlinkFast=200
BlinkSlow=600
Driver=20
OPCClient=20
Script=100

[TimeSlice]
Math=50
Driver=10
Viewer=200
OPCClient=10
Alarm=50
Script=50

[International]
Order=MDY

Separator=/

- **\\<non-volatile>\\BIN\\UNIPAD.INI**

Syntax
[Tab] NumSpaces=<Number of spaces to each TAB> [Font] nSize=<Font size> FaceName=<Font face name> NWeight=<Font Weight: 100; 200; 300; 400; 500; 600; 700; 800; 900) [ScreenPosition] x=<left coordinate (number of pixels) of the UniPad window> y=<top coordinate (number of pixels) of the UniPad window> Width=<Width (number of pixels) of the UniPad window> Height=<Height (number of pixels) of the UniPad window> [Alloc] MaxNumLines=<Max number of lines showed in the Unipad window>
Example
[Tab] NumSpaces=4 [Font] nSize=12 nWeight=500 FaceName=Courier New [ScreenPosition] x=100 y=100 Width=400 Height=250 [Alloc] MaxNumLines=256000

WinCE device image – Requirements and Comments

The modularity of WindowsCE allows each hardware manufacturer to build an image of the operating system with the specific files and services supported by the WinCE device. This section describes the minimum requirements to run CEView on the WinCE device, such as recommended actions to take advantage of all functionalities of CEView.

❑ **Minimum System Requirements**

▪ Store the following windows files in the \Windows folder: ATLCEXXX.DLL, MFCECEXXX.DLL and OLECEXXX.DLL. Otherwise, CEView will not run at all.

✎ **Note:** XXX is the operating system version number – for example: 300 for WinCEv3.x, 400 for WinCE.NETv4.x and WinCE v5.x, and so forth.

▪ Enable the support for VBScript on the MS Windows CE device, so CEView will be able to execute the scripts configured in the VBScript language on the device. When VBScript engine is enabled in the image, the file **VBScript.dll** must be available in the \Windows directory.

✎ **Note:** When enabling **VBScript** in the image, make sure to include support to MsgBox() and InputBox().

▪ Include an application or mechanism to save the Registry settings to the non-volatile registry file. We recommend storing the **regsave.exe** windows program in the *\Windows* folder.

▪ Include an application or mechanism to configure the **CEServer.exe** to be started automatically upon the unit startup.

❑ **Recommended System Requirements**

✎ **Note:** Some of these requirements may be mandatory, depending on the features used in your application.

▪ Enable DCOM (DCOM and Remote DCOM). When this option is enabled, following functions are included in the windows OLE32.DLL file, which must be stored in the *\Windows* folder:
  – **CoGetMalloc()**
  – **CoFileTimeNow()**
 – **CoRegisterClassObject()**
  – **CoRevokeClassObject()**

✎ **Note:** The OPC protocol requires DCOM to connect the OPC Client and the OPC Server (unless for in-proc connections, which requires COM only).

▪ Have a way to register files such as OCX and DLLs. We recommend storing the **regsvrce.exe** windows program in the *\Windows* folder, allowing the user to register files when necessary.

▪ Include an application to edit the register (e.g.: regeditce.exe). It is useful for debugging purposes.

▪ Include and enable the embedded Web Server (HTTP Server from Microsoft) on the WinCE image with the Web Root Directory pointing to a non-volatile folder, or along with an interface for the user to configure the Web Root Directory. This parameter is stored at the Registry file under the key HKEY_LOCAL_MACHINE->Comm->HTTPD->VROOTS->\, on the *Default* String parameter.

- Include a text editor (for example: WordPad for WinCE) on the WinCE image to edit configuration files on the WinCE device, when necessary.

- Include support for Remote Desktop

- In order to use the WinCE device as an InduSoft Web Thin Client, its Web browser needs to support ActiveX controls and scripts. The **Pocket Internet Explorer** does not support Microsoft ActiveX controls. The **Internet Explorer 5.5 for Windows CE** or higher supports Microsoft ActiveX controls. See the following URL for further information on this topic

- Store and register the following files in the *\<non-volatile>* folder: OPCProxy.dll and StOleProxy.dll. These files are available in the *…\InduSoft Web Studio\Redist\CEView\<Processor>* folder after installing InduSoft Web Studio on the development station.

- Store CEServer.exe in the *\<non-volatile>* folder and configure the WinCE device to run this program automatically on start-up. CEServer.exe launches the Remote Agent, which is used by InduSoft Web Studio to connect to the WinCE device remotely. In addition to CEServer.exe, it is recommended that you configure the CESERVER.INI and store it in the *\<non-volatile>* folder in case the default settings do not fit with the WinCE device or application.

- Store the CEView Program Files in the *\<non-volatile>\BIN* folder. Make sure that the CEVIEW.INI file on the *\<non-volatile>\BIN* folder is properly configured.

- Store the CEView Driver Files in the *\<non-volatile>\DRV* folder.

- Store a CEView demo application on the *\<non-volatile>\<App Name>* folder. Make sure that the CEAPP.INI file on the *\<non-volatile>* folder is properly configured.

 Note: The CEView Program Files, CEView Driver Files and CEView demo application can be downloaded to the WinCE device using InduSoft Web Studio. However, they can be pre-loaded on the WinCE device for demonstration purposes.

- When using Platform Builder v4.x to create the image for the WinCE device, it is recommended to select the **Industrial Automation Device** and **Human Machine Interface (HMI) panel** options from the **New Platform Wizard**.
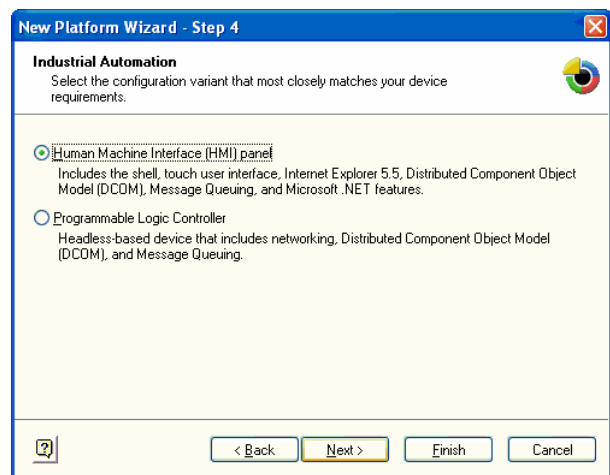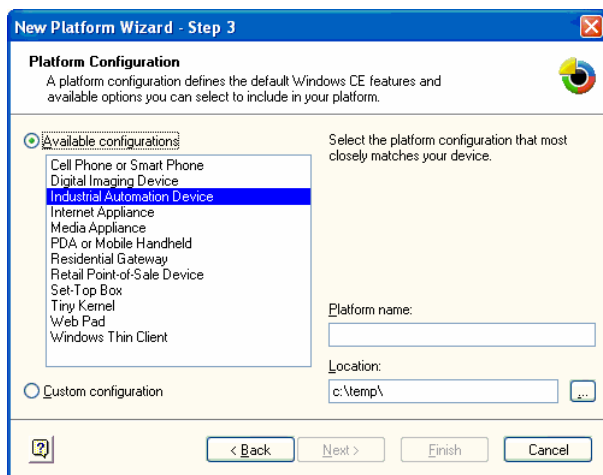
Map of Revision

Revision	Author	Date	Comments
A	Fabio Terezinho	January 6, 2006	Initial revision
B	Andre Bastos	August 9, 2006	General revision
C	Fabio Terezinho	October 11, 2006	Updated information for v6.1+SP2.