AVALANCHE .INI FILE FORMAT

This document provides information about Avalanche .INI files. It is intended primarily to provide developers and Avalanche customers with the information they need to interpret Avalanche INI files. Developers can also use the information to construct or modify INI files. The options listed are available with Enablers 4.06-42 and newer.

INI FILE OVERVIEW

An INI file is an optional component of an Avalanche software package that assists with the installation of the package on mobile units. The INI file contains sections that specify the installation parameters for the software package (for example, registry entries for the software package).

An INI file must follow a certain format or the installation of the software package will fail. The INI file is processed by the Enabler which recognizes the syntax of the INI file.

You create or modify an INI file in the Avalanche Package Builder or any text editor. An INI file has two primary components:

- **Section Headers.** Headers divide the Avalanche INI file into sections that specify various parameters for the Avalanche software package. For example, the [SHORTCUT] header precedes commands to create shortcuts to specific programs.

  Headers indicate a particular feature of the software package that is being specified (for example, shortcuts or registry keys). Each section header must be enclosed in brackets ([]). The brackets are delimiters that indicate the beginning and the end of a section header.

- **Commands.** Indicates a command within a section. Each command follows a specific format.

The following is an example of an Avalanche INI file.

```
[AVALANCHE]

[REGFILE]  // Symbol devices only
"Name" = "\Platform\TelnetCE.reg"

[CPYFILE]
"Name" = "\Platform\TelnetCE.cpy"

[HKEY_LOCAL_MACHINE\Software\Wavelink\TelnetCE]
"InstallDir" = ".\TNCE8140"
"RegFilePath" = "\Platform"
"ConfigDir" = "\Application\Avalanche\TNCE8140"

[SHORTCUT]
".\TelnetCE.lnk" = ".\TNCE8140\TelnetCE.exe"
"\Windows\Start Menu\TelnetCE.lnk" = ".\TNCE8140\TelnetCE.exe"
"\Windows\Start menu\Programs\TelnetCE.lnk" = ".\TNCE8140\TelnetCE.exe"
```

PLACEMENT OF SECTION HEADERS

The [AVALANCHE] header must be the first line in the INI file. The [AVALANCHE] section header informs the INI processor that it is processing a legitimate Avalanche INI file.

If strings are used in the INI file, then the [STRINGS] section should follow the [AVALANCHE] section.

It is also acceptable to use some section headers more than once. For example, you may have several [COPY] sections. However, you should be aware of how each section operates, as certain sections may affect other sections of the INI file (for example, the [REGFILE] and [CPYFILE] sections). The following headers should not be used more than once within the same file: [AVALANCHE], [EXECUTE], [REGFILE], and [CPYFILE].

DELIMITERS

The INI file uses the following delimiters:

`$varname$`	Used to indicate a string. (See the description of the [STRINGS] section for more information.)
//	Used to comment out text. You can use two forward slashes to add comments or notes to an INI file. The INI processor the point between where the delimiter is placed and the end of the line.
" "	Used to separate group parts of a command to make them easier to read or recognize.
****	Used at the end of a line to continue the line.

SECTION HEADERS AND COMMANDS

This section provides information about the commands that can be used within each section header of the Avalanche INI file.

[AVALANCHE]

The Avalanche section header must appear on the first line of each Avalanche INI file and controls how the package is installed. It specifies that the INI file is indeed an Avalanche INI file. The [AVALANCHE] section of the INI file may contain commands that specify whether the package is backed up on the device and whether the application that the package installs is closed before an update.

The following table includes options for this section:

Option	Values	Meaning	Example
`BackupPackage`	`Yes` | `No` Default: `Yes`	Specifies whether the package is backed up on the mobile device during the package installation. Windows Mobile 5 and Windows Mobile 6 have persistent object store.	`BackupPackage = No`
`RestartOnUpdate`	`Yes` | `No` Default: `No`	Indicates whether the application is closed before installation of a new version of that application begins.	`RestartOnUpdate = Yes`
`ServerReconnect`	`Yes` | `No` Default: `No`	Indicates whether the Enabler reconnects to the Avalanche server after it processes the INI file. `No` means the Enabler will not reconnect.	`ServerReconnect = Yes`

[STRINGS]

The [STRINGS] section allows you to specify text strings that can be used throughout the rest of the INI file. If the Avalanche INI file uses strings, then the [STRINGS] section should immediately follow the [AVALANCHE] section in the INI file. Also, any string that is used to create other strings in the [STRINGS] section must be defined before it is used.

The following table includes options for this section:

Option	Value	Meaning	Example
<StringName>	<StringValue>	Sets a string value for later use.	WindowsStartupString = \Windows\Startup

Once you have specified a string in the [STRINGS] section of the Avalanche INI file, you can use the string throughout the rest of the INI file, even in other lines of the [STRINGS] section.

The INI processor uses the dollar symbol ($) to denote the beginning and ending of a specified string. Whenever you use a string value in the rest of the INI file, the string value must be surrounded by this symbol.

For example:

```
[COPY]

$APPS$\MyPackageMyFile.lnk = $WindowsStartupString$\MyFile.lnk
```

The following strings are pre-defined in the INI processor and should only be used in the context that they are defined:

Pre-defined String	Definition
AVA	Replaced with the Avalanche Enabler configuration directory.
$APPS$	Used to indicate the directory where Avalanche-deployed applications are stored on the mobile unit. Typical path: \Program Files\Wavelink\Avalanche\APPS The INI processor also recognizes "." as the APPS directory.
$DESKTOP$	Used to indicate the Desktop folder. Typical path: C:\Documents and Settings\All Users\Start Menu This string is only available on Windows Desktop.
$DESKTOP_DIR$	Used to indicate the directory where the Desktop is stored. Typical path: \Windows\Desktop
$FAVORITES_DIR$	Used to indicate the Favorites directory. Typical path: \Windows\Favorites
$FLASH$	Used to indicate the directory where Avalanche files are stored on a mobile unit's Flash drive. This string may not exist on some devices.
$FONTS_DIR$	Used to indicate the virtual folder that contains all fonts. Typical path: \Windows\Fonts
$MY_DOCUMENTS_DIR$	Used to indicate the folder containing a user's document files. Typical path: \My Documents
$PROGRAMS_DIR$	Used to indicate the folder that contains the program groups. Typical path: \Windows\Start Menu\Programs

$PROGRAM_FILES_DIR$	Used to indicate the folder that contains installed program files. Typical path: \Program Files
$REG.x$	Replaced with the path as specified in the Enabler registry entries where x is AVA, WORK, INSTALLDIR, etc.
$STARTMENU$	Used to indicate the folder that contains the shortcuts and program groups that appear for all users. Typical path: C:\Documents and Settings \All Users\Start Menu This string is only available on Windows Desktop.
$STARTMENU:xxxx.lnk$	Similar to $STARTMENU$, but searches for the folder containing xxx.lnk. Typical path: C:\Documents and Settings\All Users\Start Menu This string is only available on Windows Desktop.
$START_MENU_DIR$	Used to indicate the folder that contains Start menu items. Typical path: Windows\Start Menu
$STARUP_DIR$	Used to indicate the startup folder for items that get started automatically. Typical Path: \Windows\Startup
$SYSTEM_DIR$	Used to indicate the Windows system folder. Typical path: C:\Windows\System32
$TEMP$	Used to indicate the directory where Avalanche temporary files are stored.
$WINDOW_DIR$	Used to indicate the WIndows base folder. Typical path: \Temp

[REGFILE]

The [REGFILE] section of the INI file specifies the name and location of the REG file for the Avalanche software package. Any registry entries created after this entry in the INI file are stored in the REG file that is specified in this section. Registry keys and values are created in the [HKEY_*] sections of the INI file.

The following table includes options for this section:

Option	Values	Meaning	Example
name	<path>	Provides the path and name of the REG file.	name = \Application\TelnetCE.reg

[CPYFILE]

The [CPYFILE] section of the INI file specifies the name and location of the CPY file for the Avalanche software package. For each copy or backup operation (see [COPY] and [BACKUP]), an entry is made in the specified CPY file for processing during cold-boot recovery. Entries made by backup operations are reversed to restore files from the destination during cold-boot recovery. (For Motorola devices only.)

The following table includes options for this section:

Option	Values	Meaning	Example
name	<path>	Provides the path and name of the CPY file.	name = \Application\TelnetCE.cpy

[RUNFILE]

The [RUNFILE] section is used to create shortcuts to the applications and utilities of a software package. It is used for Motorola mobile units that support launching RUN files from a Flash location, as opposed to using LNK files.

The following table includes options for this section:

Option	Values	Meaning	Example
<shortcut path>		The path (including the file name) where the RUN file is created.	\Application\MyRun.run = \Windows\PWord.exe
	<target path>	The path (including the file name) to the executable.	

[HHP_AUTORUN]

The [HHP_AUTORUN] section is used to add an autorun entry into the autorun file of HHP (Hand Held Products) mobile devices.

The following table includes options for this section:

Option	Values	Meaning	Example
<file>		The path to the existing autorun file.	\IPSM\Autorun.ini=Program=\IPSM\Avalanche\AvaInit.exe|Args=|Wait=1|StartOption=2 adds the AvaInit.exe file (with specified Wait and StartOption arguments) to the autorun.ini file in the \IPSM directory.
	<target>	The EXE file to add to the autorun file, along with the support parameters. Support parameters are separated by bars (|)	

[INIFILE]

The [INIFILE] section is used to add entries to a specific INI file or text files.If the file does not exist, ti will be created. An entry in the [INIFILE] section uses the following format:

<INI> = <SECTION>,<ENTRY>,<VALUE>~<MARKER>

The following table includes options for this section:

Option	Values	Meaning	Examples
<INI>		The path to the INI file that will be modified.	\Flash\Autorun.ini = program1,Program,monitor.exe adds Program=monitor.exe to the [program1] section of the Autorun.ini file in the \Flash directory. -Or- yardword.txt = INI,,water~trees adds the word "water" before the word "trees" in the [INI] section of the file.
	<SECTION_NAME>	The section of the INI file to which the data will be added. (This can be an empty value.)	
	<ENTRY_NAME>	Any valid entry name in the INI file. (This can be an empty value.)	
	<VALUE>	A valid INI value.	
	<MARKER>	The line or string of text that the <VALUE> should be inserted before. If the <MARKER> text is not found, then the command will be entered at the beginning of the file or section.	

[SHORTCUT]

The [SHORTCUT] section of the INI file specifies where shortcuts to components or utilities of the Avalanche software package will be created.

The following table includes options for this section:

Option	Values	Meaning	Examples
`<shortcut path>`		The path and name of the shortcut.	`$StartMenu$\TelnetCE.lnk = \Avalanche\APPS\TelnetCE8140 \TelnetCE.exe` creates a shortcut to the TelnetCE.exe application in the **Start** menu.
	`<target path>`	The path and name of the component or utility that the shortcut launches.	

[RENAME]

The [RENAME] section renames or moves a file. The following table includes options for this section:

Option	Values	Meaning	Examples
`<source>`	`<destination>`	The path and filename for the source /destination.	`"\MyFile.dat" = "\MyFile.old"`

[COPY]

The [COPY] section of the INI file specifies the copying of files or directories from one location to another. The options `exclude`, `overwrite`, and `versioned` should precede the copy operation they modify.

The following table includes options for this section:

Option	Values	Meaning	Examples
`exclude`	`<file>`	Excludes the specified file from the following copy operation.	`exclude = AS-Enabler.dat`
`overwrite`	`yes | no`	Specifies whether to overwrite a file during the preceding copy operation.	`overwrite = no`
`versioned`	`yes | no`	Specifies whether all following [COPY] operations will verify version information before copying.	`versioned = yes`
`<source>`	`<destination>`	The path and filename for the source /destination.	`$RAMInstallDir$\*.* = $AvaBackupDir$\*.*`

NOTE: When enabled, the versioned option remains enabled for all following [COPY] operations. Ensure you disable this option after copying the versioned files.

[BACKUP]

The [BACKUP] section specifies where you want to backup single file or all files in a single folder to a specific location. The optional parameter `exclude` must **precede** the command to which it applies. The following table includes options for this section:

Option	Values	Meaning	Examples
`exclude`	`<file>`	Excludes the specified file from the following backup operation.	`exclude = AS-Enabler.dat`
`<source path>`	`<destination path>`	The path and filename for the source /destination.	`"\MyFile.dat" = "\Flash\MyFile.new"`

[DELETE]

The [DELETE} section specifies files, registry keys or values, or directories to be deleted after the Avalanche software package is installed. XXX or is this AS the package is installed?XXX The following table includes options for this section:

Option	Values	Meaning	Examples
`key`	`<key name>`	The key to be deleted. This will also delete all values for that key. The key must not have sub-keys.	`key = HKEY_LOCAL_MACHINE\Software\KeyName2`
`value`	`<key name>\ <value name>`	The key value to be deleted.	`value = HKEY_LOCAL_MACHINE\Software\KeyName2\Value4`
`file`	`<path>`	The path to the file to be deleted.	`file = "\Temp\oldfile.txt"`
`dir`	`<path>`	The path to the empty directory to be deleted.	`dir = "\Temp\Junk"`
`tree`	`<path>`	The path to the directory to be deleted. This will delete the folder and all its contents.	`tree = "\Temp\Junk"`

[ATTRIB]

The [ATTRIB] section of an INI file allows you to assign attributes, such as read-only, to a file in the package. The following table includes options for this section:

Option	Values	Meaning	Examples
`Read`	`<path>`	Designates the file as read-only.	`Read = $CABDIR$\WLEnabler.ARM.CAB`
`Write`	`<path>`	Clears the read-only attribute from the file.	`Write = $CABDIR$\WLEnabler.ARM.CAB`

H	<path>	Designates the file as hidden.	
S	<path>	Designates the file as system.	
R	<path>	Designates the file as read-only.	
A	<path>	Designates the file as archive.	
+	<path>	Sets all the following attributes.	
-	<path>	Clears all the following attributes.	+H-SRA = \MyFile.cab

[EXECUTE]

The [EXECUTE] section of the INI file specifies reboot or CAB options that the mobile device should use once the Enabler or an Avalanche package has been installed. The following table includes options for this section:

Option	Values	Meaning	Examples
Reboot	Yes	The device is rebooted after installing the package. A prompt allows the user to override the reboot.	Reboot = Yes
	Auto	The device is automatically rebooted after installing the package.	Reboot = Auto
RebootType	Warm | Cold | Hard	When the reboot option is set, specifies the type of reboot performed. Hard is the same as a cold reboot.	RebootType = Cold
CabFiles	Yes	Install CAB files contained in the package.	CabFiles = Yes
	LaunchAndRemove	Install and then delete CAB files contained in the package.	CabFiles = LaunchAndRemove
CabCommand	<Executable>	The command used to launch CAB files.	CabCommand = \Windows\WCELoad.exe
CabParams	<Parameters>	Parameters used when launching CAB files.	CabParams = /silent /nodelete "@"
CabMonitorInstall	Yes | No	Enables or disables CAB installation monitoring. Use No for CAB files that reboot the device.	CabMonitorInstall = No

NOTE: Monitoring CAB files that initiate a reboot will cause the Enabler to keep trying to reinstall the CAB file.

[MESSAGE]

The [Message] section displays a message box stopping the execution of the INI file until the OK button has been pushed. The following table includes options for this section:

Option	Value	Meaning	Example
<title>	<message>	The name and body of the message.	"INI Message" = "Press OK to continue."

[HKEY_*]

The [HKEY_*] section creates registry entries. The wildcard symbol (*) is used to denote the rest of the registry key.

For example:

```
[HKEY_LOCAL_MACHINE\Software\Wavelink\Avalanche]
```

is the header of a section that contains commands that specify entries that will be created in the HKEY_LOCAL_MACHINE\Software\Wavelink\Avalanche registry key.The following hives are supported:

```
[HKEY_LOCAL_MACHINE\<value>]
[HKEY_CLASSES_ROOT\<value>]
[HKEY_CURRENT_USER\<value>]
[HKEY_USERS\<value>]
[HKEY_CURRENT_CONFIG\<value>]
[HKEY_PERFORMANCE_DATA\<value>]
[HKEY_DYN_DATA\<value>]
```

You can use the ordinal $#$ to represent a number matching the first number that does not already exist in a registry key.

For example:

```
[HKEY_CURRENT_USER\Software\Symbol\Launcher\Configuration\configtype0\Default\Programs\Prog$#$]
```

Under the header, list the registry key and the type and value of the key. If no type is specified, the value is considered a string. The following table lists indicators for the types of keys:

Data Type	Indicator	Example
String (REF_SZ)		MyString = Hello
DWORD (REG_DWORD)	dword:	MyDWORD = dword:000000005
NOTE: DWORD values are in hexadecimal.		
Binary (REG_BINARY)	hex:	MyBinary = hex:01,02,03
ExpandableString (REG_EXPAND_SZ)	hex(2):	MyExpandedString = hex(2): 25,53,79,73,74,65,6d,6f,7425,00
MultiString (REG_MULTI_SZ)	multi_sz	MyMultiString = multi_sz:"one","two","three"
MultiString (REG_MULTI_SZ)	hex(7):	MyMultiString = hex(7):31,31,31,00,32,32,32,00,33,33,33,00,00

If a REG file is specified in the [REGFILE] section, then the keys and values changed in the [HKEY_*] sections will be added to the REG file. The [REGFILE] section should precede any [HKEY_*] sections.

[CREATEDIR]

The [CREATEDIR] section creates a new directory. The following table includes options for this section:

Option	Values	Meaning	Example
dir	<path>	Provides the path and name of the new directory.	dir = "\Temp\MyNewFolder"

[REGISTER]

The [REGISTER] section registers a COM component. The following table includes options for this section:

Option	Values	Meaning	Example
COM	<path>	Provides the location of the component to be registered.	COM = "\Program Files\MyApp\Special.dll"

[SERVICE]

The [SERVICE] section stops or starts a service. This section is specific to Windows CE. The following table includes options for this section:

Option	Values	Meaning	Example
Start	<name>	Starts the service.	Start = "MyService"
Stop	<name>	Stops the service.	Stop = "MyService"

[UNREGISTER]

The [UNREGISTER] section unregisters a COM component. The following table includes options for this section:

Option	Values	Meaning	Example
COM	<path>	Provides the location of the component that will be unregistered.	COM = "\Program Files\MyApp\Special.dll"

[PROPERTIES]

The [PROPERTIES] section of an INI file allows you to create new properties in the Enabler property file. The property will be removed when the package associated with the properties is deleted.

An entry in the [PROPERTIES] section can contain an optional vendor name so that the property will appear under the vendor heading when viewed from Avalanche. The entry uses the following format:

```
<Vendor>.<PropertyName> = <Value>
```

If no vendor is specified, the property appears under the General heading.

Examples:

```
CustomerName.StoreLocation = Pittsburgh
```

```
StoreLocation = Pittsburgh
```

[PERM_PROPERTIES]

The [PERM_PROPERTIES] section of an INI file allows you to create new properties in the Enabler property file. Properties added under this section are not deleted when you delete the package that is associated with the properties. See the [PROPERTIES] section for format information.

[OMA]

The [OMA] section processes a provisioning XML file for Windows Mobile devices. The following table includes options for this section:

Option	Values	Meaning	Example
`provision`	`<path>`	Provides the location and name of the XML file.	`provision = "$APPS$\MyConfig\MyConfig.xml"`

[RUN]

The [RUN] section allows you to run files. The following table includes options for this section:

Option	Values	Meaning	Example
`run`	`<path>`	Starts the specified file and suspends the INI file until the file exits.	`run = "\Windows\IPI.EXE"`
`start`	`<path>`	Starts up the specified file to run concurrent with the INI file.	`start = "\Windows\IPI.EXE"`
`params`	`<parameters>`	Provides parameters for the next run or start command.	`params = /F`

DOCUMENT VERSION HISTORY

- 08/03/2004. Document created.
- 14/10/2004. Added information about [AVALANCHE] commands.
- 20/03/2007. Added [Properties] section header. Updated [Avalanche] and [Execute] commands.
- 10/11/2008. Added CAB commands to [Execute].
- 08/05/2009. Complete update.
- 28/12/2010. Reformatted; added [RUN] and [DELETE] tree options.

Wavelink Corporation
USA and Canada: 1.888.697.WAVE (9283)
Outside the USA and Canada: + 800 WAVELINK (9283 5465)
CustomerService@wavelink.com