InstallWare

Version 3.51 -- August 1994

By Bob Dolan

Information

Required Files

README

This is a file that will be displayed to the user, who is installing your application. You can name this file anything (readme, readme.txt, readme.1st, etc.) you want to name it.   You will be including the name of this file in the filename = section of the APPSETUP.INF file.

Basics

NOTICE: If you are upgrading from a previous version, please check the Development History section for any changes that may affect your INF file.

Thank you for trying InstallWare. This is a small, easy to use application installation program. It costs nothing to use if you are distributing shareware or freeware. I wrote this system to distribute my own shareware because I couldn't find anything that met my requirements. Those requirements were:

- Small: I didn't want to add too much extra baggage to my distribution or upload packages.

- Cheap: I didn't mind paying a shareware type price but all I found were commercial type prices.

- Simple: It had to be easy for me to configure and easy for my customers to use.

- Attractive: It had to look good since it is the first impression your customers get of your product. In this case I used the CTL3D.DLL provided by Microsoft.

This code is based on some sample code that Microsoft made available in 1991. I have added a lot of functionality and cleaned up the code and dialogs.

As far as how to use this package...

The INSTALL.EXE application reads a configuration file when it starts up. This file is called APPSETUP.INF and must be present with InstallWare. The entire personality of InstallWare is defined within this program. Rather than go into a long discussion on how to do this, I have commented each section of this file with what I feel are rather descriptive hints on how to change things to suit your needs. Hey, I figured this out from the Microsoft stuff with no help and you're probably smarter than me. However, if you are lost, then contact me and I'll see if I can help.

Files that are compressed using Microsoft's COMPRESS utility, will be automatically de-compressed by InstallWare. When COMPRESS processes your uncompressed files it will put an underscore character into the filename. InstallWare requires that the filename have this format. Just be sure that the files are listed in the APPSETUP.INF file without the underscore character. It seems that the only version of this utility that works reliably is version 2.0 (1992).

To see what the system looks like, you can run INSTALL.EXE right from this directory. You will see what a user of yours would see.

Remember to include the CTL3D.DLL file. I will be doing a version that tries to dynamically load this file. This will allow the program to continue if it is not found. When it is available, I will upload it. My shareware is usually found on CIS, GEnie, and SDN.

The method that has been implemented for handling multiple floppies involves using 2 executables. The first one (INSTALL.EXE) is used to copy the actual installer (IW.EXE), the APPSETUP.INF, and CTL3D.DLL to the WINDOWS directory which is assumed to be in the PATH. CTL3D.DLL is only copied if there is not one there already. After these files are copied, IW.EXE is launched using WinExec. IW does all of the installing work and then deletes these files before exiting. IW.EXE knows whether it was copied from the floppy or not and will work properly in either case. The file 2DISK.INF has been included as an example of a multiple disk installation.

If your installation only uses one disk, then you don't need the loader file. Just rename IW.EXE to INSTALL.EXE (or SETUP.EXE, or anything) and go. The only time that IW.EXE must retain that name is in a multi-disk installation. The file APPSETUP.INF must retain it's name and the README file must retain the name you have specified in the APPSETUP.INF file. Otherwise, all files can be renamed to whatever you require.

The main dialogs of the system have had an icon added to them. If you would like to have a version of InstallWare that uses your own icons, send them to me along with $15 and I'll re-compile the system and send it back to you. If anyone has any code that loads an ICO file and places it in a dialog, I can add that functionality to the INF file to make the system that much more configurable.

The code seems robust and has worked with everything that quite a few people have thrown at it. Large apps with multiple disks should work. I'd be interested in hearing about your experiences. However, see the disclaimer.

Good Luck! and have fun!

Distribution

Restrictions:

InstallWare in it's archive format may be freely distributed, subject to the following restrictions:

- InstallWare must be distributed without modification, in its entirety. If you distribute InstallWare to friends, associates, or to a computer bulletin board system (BBS), you may only distribute the compressed file including the copyright and disclaimer.

- InstallWare may not be distributed by any for-profit entity which fails to disclose the following points somewhere in their literature:

- InstallWare is not to be used as an enticement to purchase another product without the explicit consent of the Author.

- InstallWare is not to be packaged for sale with its reference manual, or other supporting documentation pre-printed for the end-user.

- InstallWare is not to be "rented" or leased.

These distribution restrictions apply to anyone who wishes to distribute InstallWare, whether they be commercial vendors, user groups, BBS operators, or individuals. All parties are hereby given permission to capture screen shots of InstallWare for use in catalogs or newsletters listing or reviewing our product. Please identify the image in some way as being from InstallWare.

Software developers that employ the components of InstallWare to assist their users in the setup of the developer's product may distribute any portion of the contents of InstallWare that the developer see fit. However any distributed use of these components requires that the developer register with the author.

CTL3D.DLL

A dynamic link library supplied by Microsoft that gives the program a sculpted appearance. This library **MUST** be included with your distribution floppy or upload file.   InstallWare will crash without it.

Registration Form

To print this registration form, open the **File** menu and select **Print Topic**.
Mail to:

Bob Dolan
P. O. Box 16514
Rochester, New York 14616-0514

Please register me as a satisfied user of InstallWare (ver 3.51):

Amount Enclosed:

$0 [ ] ShareWare Author Registration

$25 [ ] Commercial Registration

Application Name:____

Name:____

Address:____

City:_____________________State:_______Zip:_________________

Phone(optional):____

GEnie/CompuServe/Internet mail address:______________________

Where did you get InstallWare ?_______________________________

Comments:____

____

____

____

____

Registering InstallWare

InstallWare is copyright 1994 by Bob Dolan

If you are distributing Shareware with this system, then you are not obligated to send me money. However, I won't refuse any donations if you feel so inclined. I do require that you register with me and send me a copy of what you are sending out with my program. No money, just a disk. If you are using this in a commercial environment or distributing commercial software (in other words, your customers pay before they try), then you must register by sending $25 along with a copy of what you are distributing. Make checks payable to Bob Dolan and send them to:

Please print the registration form and fill in the requested information.

You may reach me on at the following EMail addresses:
GEnie: DOLAN
Compuserve: 71075,3256

Distribution

If you distribute InstallWare to friends, associates, or to a computer bulletin board system (BBS), you may only distribute the compressed file including the copyright and disclaimer.

Disclaimer

The Copyright Owner hereby disclaims all warranties relating to this software, whether express or implied, including without limitation any implied warranties of merchantability or fitness for a particular purpose. The Copyright Owner will not be liable for any special, incidental, consequential, indirect or similar damages due to loss of data or any other reason, even if The Copyright owner or an agent of The Copyright Owner has been advised of the possibility of such damages. In no event shall The Copyright Owner's liability for any damages ever exceed the price paid for the license to use the software, regardless of the form of the claim. The person using the software bears all risk as to the quality and performance of the software.

Other Products from Bob Dolan Software

Windows Shareware

DeskMenu! -- A low profile, text based Program Manager for Windows

There is probably no other program you can add to your system that will make better use of it's screen and disk space than DeskMenu! This easily configured program manager and resource monitor speeds access to all your applications. DeskMenu! places a menubar on your desktop and allows you to setup the menu headings to reflect how you categorize your applications. An optional InfoBar shows time, date and up to six different system resources. DeskMenu! can be used in addition to the Program Manager or it can replace it as the Windows shell. All of this functionality in a good looking, easy to use package!

Yacht-Z -- A dice strategy game for Windows

Yacht-Z is a version of the classic dice game. You can play single or triple scoring varieties of the game. Up to four players can play at once. This is the original and most often reviewed of all Yacht-Z type dice games. Several years of constant use and valuable input from users has honed this into one of the most enjoyable and challenging games available (unless you prefer blowing up things!).

WinGo! -- The classic Bingo for Windows

Finally, Bingo has arrived for Windows! You match wits with Rosie, an alert and cunning adversary. You can play with as many cards as you can handle at the speed you like. Fully configurable to adapt to your playing style. The registered version includes digitized voice calling for the most realistic Bingo play possible.

The Joker -- Jokes On Demand

The Joker gives you a laugh whenever you need it. Just as your day is getting completely boring, a window will appear with a joke to break the ice. If one joke doesn't do it for you, then press the ANOTHER button for more. The program is easy to use, and once setup to your liking, it will require no maintenance. You can load different joke files, order more from the author, and even create your own.

BattleGrid -- War in a window -- Battleship for Windows 3.x

BattleGrid is a version of the classic board game. Play against the computer or another person. Watch the computer play against itself using different strategies. Several levels of a computer opponent are available, from idiot to genius.

SCAT -- A strategic and daring card game

Don't get stuck with the lowest hand! SCAT is a card game involving four players that can be as simple or challenging as you want. Features include defining strategy levels, setting the game speed, and leaving the cards face up or face down as you play. This game even includes conversational chit chat between the players!   Very addicting and fun!

WinDial -- A   Windows Phone Book that dials

An easy to use phone book that will automatically dial any selected number as long as you have a modem.   When a busy signal is detected, you can tell WinDial to retry until someone answers.   Other busy attempts can be placed into a queue, and all will be retried in order.

Mister Clean -- Clean up your hard drive

One of the most efficient ways to clean up those hidden but unwanted or unused files that take up needed space.   A slick Windows 3.x user interface makes this utility a must have for your toolbox.   Mister Clean finds files the you specify and sorts them by size.   You click on the ones you want deleted and then off they go, one-by-one (to be safe) or all at once.   Not sure what the file is?   Then run it right from Mister Clean and check it out before dumping it!   Truly a unique way of managing precious disk space.

Windows Freeware

InstallWare -- An application installer

InstallWare is a small, easy to use application installation program for Windows applications.   It costs nothing to use if you are distributing shareware or freeware.

16 Tons -- Drop your sound files here!

Simply plays WAV and MIDI files.   Accepts drag & drop operations from the file manager.   Most handy when used it as an icon.

Random Saver -- Variety in savers

Make full use of that collection of screen savers you've got.   Now you can get them out of your WINDOWS directory.   This utility will run them at random from any location when you install it as the default screen saver in the Windows Control Panel.

BBS Sysops & Other Vendors

Please post InstallWare using the following keywords:
WINDOWS INSTALL SETUP

Suggested one-liner:
An attractive and easy to use application installer.

Other Information:
Name: InstallWare
Version: 3.51
Category: Windows Utility
Registration: $0 for shareware authors
$25 for commercial distribution
Requirements: Windows 3.X

Author: Bob Dolan
Company: Bob Dolan Software
Address: P.O. Box 16514
Rochester, N.Y.
14616-0514 U.S.A.

Telephone: 716-865-8248

GEnie DOLAN

APPSETUP.INF

```
;
;
;
; This file, APPSETUP.INF, contains
; information that InstallWare uses
; to install applications. This file
; must be in the same directory as
; the program files IW.EXE(if used)
; and INSTALL.EXE. The two files
; can be on a floppy disk, a hard
;disk, or a network disk drive.
;
; The information in this
; file determines:
;
;    * The names of the disks and
;         directories from which, and
;         to which, InstallWare copies
;         files.
;
;    * The name of the group that
;         InstallWare creates in
;         Program Manager's window,
;         and the names of program
;         items that InstallWare adds
;         to that group.
;
; You can create your own
; InstallWare program by changing
; some of the information in this
; file.
;
; The Parts of APPSETUP.INF
;
; APPSETUP.INF is similar in format
; to a Windows initialization (.INI)
; file.
;
; Sections: APPSETUP.INF is divided
;      into sections. Each section is
;      identified by a name enclosed
;      in square brackets.
;
;      Most section names are
;      "hard-coded". However, some
;      section names are defined
;      within other sections, and are
;      therefore easily customizable.
;
; Comments: A comment begins with
;      a semicolon. You can include a
;      comment on the same line as
;      syntax, as long as it comes after
;      the syntax.
```

```
;
; Spaces:     Spaces are ignored,
;      except when between double
;      quotes.   Blank lines are also
;      ignored.
;
; Sections in This File
;
; The rest of this file contains
; the actual sections and statements
; that make up the working file. It
; also includes comments that
; explain each section and statement.
;
;
[infofile]
  filename = README
  appname = SAMPLE APPL
  ;exefile = "\notepad.exe sub\install.txt"
  exefile   = "#W\notepad.exe #D\sub\readme."
  message   = "Do you want to view the readme file?"
;
;
[dialog]
  caption = "Sample Installation"
;
;
[data]
  defdir = C:\SAMPLE
;
;
[disks]
  1 =., "INSTALL"
;
;
[needed.space]
  minspace = 40000
;
;
[dest.directories]
  #app.main, 0:
  #app.sub, 0:sub
  #app.SETUPSYSDIR, 0:SETUPSYSDIR
  #app.SETUPWINDIR, 0:SETUPWINDIR
;
;
[app.main]
  1:APPSETUP.INF,        "Application Configuration File"
;
;
[app.SETUPSYSDIR]
  1:README,                "Readme File"
;
;
[app.SETUPWINDIR]
```

```
  1:INSTALL.TXT, "Installing Doc file"
;
;
[app.sub]
  1:INSTALL.TXT,         "INSTALL.TXT"
  1:INSTALL.EXE,         "INSTALL.EXE"
;
;
[progman.groups]
  Bob Dolan Shareware, BOBDOLAN.GRP
;
;
[Bob Dolan Shareware]
  "Install", SUB\INSTALL.EXE
;
;
[INIFiles]
            WIN.INI, InstallWare, TestEntry1, TestString1
            WIN.INI, InstallWare, TestEntry2, #D
            SYSTEM.INI, InstallWare, TestEntry1, TestString1
            CONTROL.INI, InstallWare, TestEntry2, #D\SUBDIR\FILENAME.EXT
            BOB.INI,  InstallWare, TestEntry1, "Bob wrote this!"
;
;
[ExtCmds]
            SUB\SFX.EXE, NO
            "NOTEPAD.EXE #V\BOOTLOG.TXT", NO
;
;
[Options]
            DirChg, YES
            OverWrite, NO
            Restart, YES
            Reboot, YES
            RestartString
            DeleteIW, NO
```

DirChg

If this option is set to NO, the Install directory editbox will not be displayed.   If it is set to YES, then it will be shown and the user may specify a path to use for the installation.   This is typically used for installing screen savers or drivers that must reside in the WINDOWS directory.

OverWrite

This option determines what the default setting of the OverWrite radio buttons will be.   The user will still be able to change this, but it allows the installer to suggest the mode.

ExtCmds

This section lists the files that will be executed to complete the installation. The user is not prompted as to whether they would like these files to be run. Common uses are running self-extracting archives or tutorials. The filename that is specified is assumed to be relative to the install path. The DELETE FLAG parameter indicates whether the file is deleted after it has completed executing. All files run in succession, and processing of InstallWare is suspended until each file terminates.
The format is as follows:

```
EXECUTABLE FILENAME, DELETE FLAG (YES/NO)
```

[Options]

This section allows for the customization of several InstallWare functions. The format is as follows:

```
OPTION, FLAG (YES/NO)
```

[INIFiles]

This section defines any INI file modifications that will be performed. The Installation Directory can be referenced by using the #D macro as is shown in the examples below. If you specify an INI file that does not exist, it will be created. If you do not specify a path, the WINDOWS directory is used. Be sure to use quotes around text that contains spaces (see example).

The format is as follows:

INIFILENAME, INI SectionName, Entry Label, Actual Text to be added

filename =

The FILENAME label indicates the name of the text file that is shown to the user when InstallWare starts up.

appname =

The APPNAME label is used to refer to the name of the application being installed.

defdir =

This section defines the default directory. This directory is placed in the InstallWare location editbox when the program starts. If the user does not change it, then this will be the base directory of the installation.

minspace =

This section tells InstallWare how much space is needed on the target disk to do a successful installation.   The user will get an error message if they try to install to a disk that has less space.

```
[app.SETUPSYSDIR]
      1:README,                 "Readme File"
```

[app.SETUPSYSDIR]

This is the file list for the \WINDOWS\SYSTEM directory. The format is as follows:

```
  [Disk Number]:[File Name], "Descriptive text to display"
```

```
[app.SETUPWINDIR]
      1:INSTALL.TXT,         "Install Doc File"
```

[app.SETUPWINDIR]

This is the file list for the \WINDOWS directory. The format is as follows:

```
  [Disk Number]:[File Name], "Descriptive text to display"
```

Development History

3.51 Maintenance release

- In version 3.5, I upgraded the CTL3D.DLL file to the latest version. This caused problems on machines that did not have the same version of the DLL located in it's SYSTEM directory (a warning dialog would appear). This behavior is totally unnecessary and Microsoft really goofed on this. Anyway, I have put the previous version of the DLL back into the release package. The installation process temporarily places the DLL into the WINDOWS directory for the duration of the installation (unless it is there already), and then deletes it (if it put it there). If this doesn't make sense, don't worry. Just use this version (3.51) and your users will thank you!

- A sporatic GPF caused by answering No to the reboot prompt messagebox on some machines has been fixed and verified to be gone.

- An early abort in the installation process left some InstallWare files on the host system. This no longer happens.

3.5 Some bug fixes and functional additions

- Major overhaul of macro processing. Macros can be specified in any string, and the literal #W can be used in the form ##W. Several other useful macros have been added. See the macro section. for details.

- The system can be rebooted or restarted when the process completes. The message that is displayed to the user at this point can be specified in the INF file.

- When the directory change function is disabled in the Options section, the install dialog is reconfigured to show where the installation will occur.

- In multi-disk installations, the deletion of the IW.EXE file can be skipped. This causes SHARE violations in some applications. (we're looking into it!)

- User requested aborts did not work properly, now they do.

3.2 Some bug fixes and functional additions:

- Spaces now work correctly in Program Group specifications. You need to put the text in quotes to enable this. See the INF file for an example of how to do this.

- The static text field indicating the title string has been enlarged to accomodate long app names.

- The Default Directory specification can now reference the user's WINDOWS directory by using the #W macro as part of the path name.

- INI files can now be created and modified. The actual installation directory can now be referenced (in this section only) by using the #D macro as part of a text string. See the INIFiles section in the APPSETUP.INF file.

- Files can now be executed after all files have been copied. This is mainly intended for running self-extracting archives but can be used for demos or tutorials, although this has not been tested. There is a self-extracting archive (SFX.EXE) included to demonstrate this capability. You can see it work by running the included APPSETUP.INF file as part of a test install. See the ExtCmds section for details.

- The behavior of the user setup dialog can now be controlled by setting flags in the

Options section.

3.1 Some bug fixes and functional additions:

- Under certain conditions, the size and location of the ProgMan window would be reset.
- Specifying a non-embedded icon to use in the program group did not work.
- Program icons are now replaced, rather than duplicated in the group.
- Cleaned up all dialogs.   Better button sizes and spacing.
- The INSTALL launcher would not run IW.EXE if the launch point was a hard drive.

3.0 Major Changes including the following:

- Fixed some grammatical errors in the displayed text.
- The handling of Compressed files is discussed.
- Multiple disks are supported.
- The [app.copy.appstuff] section name has been changed to [dest.directories] because that is what it is. I don't know why it was named the former!
- A user specified executable can be run once the installation is complete.
- Added huge amounts of comments to the sample APPSETUP.INF file to help with setting it up.
- The contents of an existing Program Group are no longer erased when InstallWare adds an icon to it. (ouch!)

2.0 Added the capability to place icon executables in directories other than the base directory.

```
[infofile]
      filename = README
      appname   = SAMPLE APPL
      ;exefile   = "\notepad.exe sub\install.txt"
      exefile   = "#W\notepad.exe #D\sub\readme."
      message   = "Do you want to view the readme file?"
```

[infofile]

The [infofile] section describes various miscellaneous aspects of the installation. Within [infofile], the labels available are:

filename
appname
exefile
message

exefile =

The EXEFILE label is used to tell InstallWare that you would like to run a file after the installation is complete.   This is useful for showing a readme file, unpacking some archives, or just starting up the program that was just installed.   In the example below, note that the filename starts with a backslash. This indicates that the program is in the PATH. Otherwise, it is assumed to be located relative to the installation directory.   You can also use macros to indicate directories. The MESSAGE label denotes the text that will be displayed in the message box that prompts the user just before the EXEFILE is run.   If there is no MESSAGE, then the EXEFILE is run without prompting the user.

message =

The MESSAGE label denotes the text that will be displayed in the message box that prompts the user just before the EXEFILE is run.   If there is no MESSAGE, then the EXEFILE is run without prompting the user.

```
[dialog]
    caption = "Sample Installation"
```

[dialog]

The [dialog] section is used to define the CAPTION of the windows that shown in the InstallWare program.

```
[data]
      defdir = C:\SAMPLE
```

[data]

The [data] section defines the default directory. This directory is placed in the InstallWare location editbox when InstallWare starts.   If the user does not change it, then this will be the base directory of the installation.   To reference the WINDOWS directory on the user's system, use the #W macro.

```
[disks]
      1 =., "INSTALL"
```

[disks]

The [disks] section defines the distribution disks that contain the application files. InstallWare uses this information to tell the user to insert the correct disk.

Elsewhere within this .INF file, the distribution disks are normally referred to by a single-character disk ID. This section defines those disk IDs, and includes information about the disk to which each disk ID refers.

The disk ID '0' is reserved; it represents the installation directory, the directory in which the user is installing the application.

The format of each disk definition is:

```
      n = path, title
```

where

n is the disk ID (a single character from 1-9 or A-Z).

path
The path of the source directory from which InstallWare should copy the files to the disk. The path can be relative to the source directory (see examples below).

title
Title is a descriptive name for the disk. The title should match the disk's printed or written label exactly.

The following statements would define two distribution disks.

```
      1 =.,           "Demo Application Disk 1"
      2 =.\files,   "Demo Application Disk 2"
```

The first statement tells InstallWare to refer to Disk 1 as "Demo Application Disk 1". Because the period (.) denotes the current directory, the files on that disk will be copied from the root directory of the distribution disk.

The second statement tells InstallWare to refer to Disk 2 as "Demo Application Disk 2"; the files that Disk 2 contains will be copied from the \FILES directory of the distribution disk.

You can include as many disk-definition statements as necessary. Every distribution disk should have a corresponding disk-definition statement; otherwise, InstallWare cannot tell the user to insert the appropriate disk.

```
[needed.space]
      minspace = 40000
```

[needed.space]

This section tells InstallWare how much space is needed on the target disk to do a successful installation.   The user will get an error message if they try to install to a disk that has less space.

```
[dest.directories]
    #app.main,      0:
    #app.sub,       0:sub
    #app.SETUPSYSDIR,      0:SETUPSYSDIR
    #app.SETUPWINDIR,      0:SETUPWINDIR
```

[dest.directories]

The [dest.directories] section contains section-definition statements. Each statement defines a section that lists application files to be copied as part of installation. The sections are organized by file destination; you should define a separate section for each destination directory.

Each section definition has the following form:

```
    #section_name, 0:dest_pathname
```

where

#section_name
defines the name of the .INF section that lists the files to be copied.

0
0 is the disk ID that represents the installation directory. (0 is a reserved disk ID, and always represents the installation directory, the directory the user specified when asked where to install the application.)

dest_pathname
Is the pathname of the destination directory, relative to the installation directory. For example, "0:FILES" represents the FILES subdirectory of the installation directory.

In the example above, the first two entries define application specific directories. The #app.main directory will be the base directory defined by the user (or the default as described above). The #app.sub directory will cause a directory called sub to be created off of the base directory.

The third and fourth entries refer to the two Windows specific directories: \WINDOWS and \WINDOWS\SYSTEM. InstallWare will find these directories and copy the specified files into them.

```
[app.main]
      1:APPSETUP.INF,        "Application Configuration File"
```

[app.main]

This section is a user-defined section that lists files to be copied to a particular destination directory. The [dest.directories] section defines the name of this section and the destination directory of the files.

In each section like this one, you should list all files that you want copied to the same destination. (For example, all the files in this section, [app.user], will be copied to the installation directory.)

InstallWare copies the files listed in this section in the order in which they are listed.

The syntax of each file listing is

```
      N:FILENAME, "Description"
```

where

N:
Is the disk ID of the disk that contains the file. (Disk IDs are defined in the [disks] section.) If the specified disk is not in the disk drive, InstallWare prompts the user to insert it.

FILENAME
Is the name of the file, including any filename extension.

Description
Is descriptive text that InstallWare displays as it is copying the file or group of files.   If you leave the description blank, InstallWare will continue displaying the descriptive text from the previous file. This lets you use a general name for a group of files.

```
[progman.groups]
      Bob Dolan Shareware, BOBDOLAN.GRP
```

[progman.groups]

The [progman.groups] section (optional) tells InstallWare to create Program Manager groups for your application. (InstallWare then uses DDE to communicate with Program Manager.)

The section lists the names of the groups you want to create. You then define additional sections in this file; those sections list the program items you want in each group.

The syntax for each group name is:

```
      groupname, [groupfile.grp]
```

where

groupname
Is the title you want Program Manager to display under the icon that represents the group. (The groupname will also be the name of the section that defines the contents of the group.)

groupfile.grp
Is the filename of the file in which Program Manager will save information about the group. (You must include the .GRP filename extension.) This parameter is optional; if you omit it, InstallWare uses a default name for the group file.

The above group-definition statement tells InstallWare to create a group named "Bob Dolan Shareware", and store its information in a file named BOBDOLAN.GRP.

```
[Bob Dolan Shareware]
    "Install", SUB\INSTALL.EXE
```

[Bob Dolan Shareware]

This section describes the Program Item Icon to add to the Program Group defined in the section heading.

In each section like this one, you should list all items that you want InstallWare to add to the group.

The syntax for item-definition statements is:

```
    "Description",    APPFILE.EXE, [ICONFILE.EXE[, N]]
```

where

Description
Is the text that will appear below the program icon when displayed in the Program Manager group.

APPFILE.EXE
Is the command line that starts the application.

ICONFILE.EXE
Is the application file that contains the icon you want to represent the application. Typically, this is the executable application file, but it could be a different file. (You can also specify a .ICO file, created using the SDKPaint tool.) This parameter is optional; if you omit it, InstallWare will use the first icon it finds in APPFILE.EXE.

N
Is the offset of the icon you want to use within the file ICONFILE.EXE. This parameter is optional; if you omit it, InstallWare uses the first icon it finds in ICONFILE.EXE. You must include this parameter if the file you specify contains more than one icon, and you want to use an icon other than the first icon. To use the Nth icon, specify the number N-1. For example, to use the third icon, specify 2.

In the above example, the following item-definition statement tells InstallWare to add an item titled "Install" to Program Manager. The application command line is a file named INSTALL.EXE which is located in the SUB subdirectory off of the installation directory.

```
[app.sub]
    1:INSTALL.TXT,        "INSTALL.TXT"
    1:INSTALL.EXE,        "INSTALL.EXE"
```

[app.sub]

This is the file list for the sub directory located off of the base directory.

Credits

I would like to thank the following people for making this product all that it is today. A development tool of this type must handle many unusual and unexpected uses, and these folks put InstallWare in those situations for me, and then worked with me until each issue was resolved. I am very grateful.

In particular:
Dennis Fischer - The original composer of this help file. He took my INSTALL.TXT file and created the basic format you are now looking at. I have simply taken it from him, put some additional information in, and updated it to this version of the code. Thanks Dennis!

David McCarter - Author of Shareware/Freeware Report and VB Tricks & Tips
For finding some unusual bugs and detecting some functional omissions which have been added.

Robert Wooldridge and Jim Koontz for their excellent input that resulted in the improvements contained in Ver 3.5 of the product.

Timothy Campbell
William Mann
Mike Craven

And thanks to all of the developers who have registered with me. I have enjoyed seeing how you have made use of InstallWare. Your adoption of this installation solution should make other potential users of this utility more confident in going with InstallWare.

Restart

If this option is set to YES, the user will be prompted to restart Windows.   If set to NO, it will just happen (The developer should warn about this at the beginning of the install).   If the field is not present, then no restart will occur.

Reboot

If this option is set to YES, the user will be prompted to reboot the system. If set to NO, it will just happen (the developer should warn about this at the beginning of the install). If the field is not present, then no reboot will occur.

DeleteIW

Due to SHARE problems on some systems, you might want to leave IW.EXE on the target system.   To do so, set this option to NO (i.e. don't delete it)   Note that this is not an issue with single disk installations.

RestartString

This field MUST be present if you use either of the reboot/restart fields. It will be displayed in a MessageBox along with Yes/No buttons.

Macros

Any String that you define in this file can include one of the following macros to enable defining directory locations that would be impossible otherwise. Simply place the desired macro in the string where you would
normally use a pathspec.

Supported Macros in this Version

- #W The user's WINDOWS directory
- #S The user's SYSTEM directory
- #D The directory in which the installation actually occurred
- #V The DRIVE on which the installation occurred, such as C: (no backslash)

If you need to specify any of these letter combinations and do not want the macro expansion to take place, just precede the combination with an extra #, as in the following: "The ##Workplace ##Solution"