

InstallShield® **Express**

Version 3

Getting Started Guide Visual FoxPro Limited Edition



InstallShield Express™ – Visual FoxPro Limited Edition Getting Started Guide

© 2000 - 2001 InstallShield Software Corporation. All Rights Reserved.

This document is being furnished by InstallShield Software Corporation for information purposes only to licensed users of the InstallShield DemoShield software product and is furnished on an "AS IS" basis, that is, without any warranties, whatsoever, express or implied. InstallShield is a registered trademark and service mark of InstallShield Corporation. DemoShield is a trademark of InstallShield Software Corporation. Other brand and product names are trademarks or registered trademarks of the respective holders. Microsoft Windows, Windows NT, the Windows logo, the Windows NT logo and FrontPage are registered trademarks of the Microsoft Corporation. Adobe Acrobat is a registered trademark of Adobe Systems Incorporated. Information in this document is subject to change without notice and does not represent any commitment on the part of InstallShield Software Corporation. The software described in this document is furnished under a license agreement. The software may be used only in accordance with the terms of that license agreement. It is against the law to copy or use the software except as specifically allowed in the license. No part of this document may be reproduced or retransmitted in any form or by any means, whether electronically or mechanically, including, but not limited to the way of: photocopying, recording, or information recording and retrieval systems, without the express written permission of InstallShield Software Corporation.

The InstallShield Express – Visual FoxPro Limited Edition Getting Started Guide
Initial Publishing January 2001
Printed in the USA

Visit our Web site at [**http://www.installshield.com**](http://www.installshield.com).

January 2001

Dear Express – Limited Edition User:

InstallShield Express – Visual FoxPro Limited Edition is provided to you courtesy of InstallShield via Microsoft Visual FoxPro. It offers a subset of the capabilities and features available in the full edition of InstallShield Express 3.03.

The Getting Started Guide, Online Reference, and Knowledge Base comprise the comprehensive InstallShield Express documentation set. Please note that some information and examples might not apply to this limited edition, as some features are available only in the full edition of InstallShield Express.

For additional InstallShield Express information, please visit our Web site at **<http://www.installshield.com>**. The latest InstallShield news, hot installation topics, technical white papers, and downloadable files are now available to you 24 hours a day, seven days a week.

Thank you for using InstallShield Express – Visual FoxPro Limited Edition.

Sincerely,

Viresh Bhatia
CEO InstallShield Software Corporation

Chris Knoll
Technical Publications Manager

Lori Erickson
Technical Writer

Table of Contents

Welcome to InstallShield Express – Visual FoxPro Limited Edition	xi
Features and Benefits	xiii
Easy-To-Use Interface	xiii
Easy Upgrade Path to InstallShield Professional—Windows Installer Edition	xiii
Embedded Help	xiv
Support for Third-Party Technology	xiv
Microsoft Windows Installer Service	xiv
System Requirements	xiv
About InstallShield Software Corporation	xv
InstallShield's Products	xv
Administrator Solutions	xv
Installation Development	xvi
Multimedia Authoring	xviii
Internet Deployment	xviii
Technical Support Resources	xix
Online Help	xix
Help Library	xx
Setup Map	xx
Context-Sensitive Help	xx
Web-Based Help	xx
InstallShield Knowledge Base	xxi
InstallShield Newsgroups	xxi
InstallShield Training Courses	xxi
InstallShield Consulting Services	xxi
InstallShield Subscription Plan	xxii
Document Conventions	xxii
 Chapter 1 Setup Fundamentals	 1-1
What is a Setup?	1-3
Typical Elements of a Setup	1-3
Perform File Transfer	1-3
Display User Interface	1-3
Create Shortcuts	1-4
Register File Associations	1-4
Register COM, COM+, and DCOM Files	1-4

Register Product for Uninstallation	1-5
What is Windows Installer?	1-5
Benefits of Windows Installer	1-5
Self-Repair	1-5
Feature Advertisement	1-6
Componentized Installation	1-6
Rollback	1-6
Single File	1-6
Merge Modules	1-6
User-Accessible Information	1-6
Windows 2000 Logo Compliance	1-7
Clean Uninstallation	1-7

Chapter 2 For Continuing Express Users 2-1

Migrating Your Existing Express Projects	2-3
Migrating Multi-Language Setups	2-3
Steps to Migrate	2-3
Files, File Groups, and Components	2-3
Setup Types	2-3
InstallShield Objects	2-4
Express Extensions	2-4
Registry Entries	2-4
Shortcuts	2-4

Chapter 3 Using InstallShield Express – Visual FoxPro Limited Edition 3-1

The Setup Project File	3-3
Tour of the IDE	3-3
Toolbar	3-3
Viewbar	3-4
Checklist	3-4
Views	3-4
Step 1: Organize Your Setup	3-5
General Information	3-5
Features	3-5
Setup Types	3-6

Upgrade Paths	3-7
Step 2: Specify Application Data	3-7
Files	3-8
Files and Features	3-8
Objects/Merge Modules	3-9
Dependencies	3-10
Step 3: Configure the Target System	3-11
Shortcuts/Folders	3-11
Registry	3-11
ODBC Resources	3-12
INI File Changes	3-13
File Extensions	3-14
Environment Variables	3-15
Step 4: Customize the Setup Appearance	3-15
Dialogs	3-16
Billboards	3-16
Text and Messages	3-17
Step 5: Define Setup Requirements and Actions	3-17
Requirements	3-17
Custom Actions	3-18
Support Files	3-18
Step 6: Prepare for Release	3-19
Build Your Release	3-19
Test Your Release	3-19
Distribute Your Release	3-20

Chapter 4 Basic Setup Tutorial	4-1
Overview	4-3
Step 1: Organize Your Setup	4-3
Step 2: Specify Application Data	4-5
Step 3: Configure the Target System	4-6
Step 4: Customize the Setup Appearance	4-7
Step 5: Define Setup Requirements and Actions	4-8


Step 6: Prepare for Release	4-8
Summary	4-11

Chapter 5 Advanced Features **5-1**

ODBC	5-3
Merge Modules	5-3
Custom Actions	5-5
EXE Custom Actions	5-5
DLL Custom Actions	5-6

Chapter 6 Frequently Asked Questions **6-1**

How Do I Specify the Destination Folder for My Application?	6-3
Using INSTALLDIR	6-3
Specifying File Destination Folders	6-3
Hard-Coding a Destination Directory	6-3
How Can I Be Sure the Target Machine Has the Windows Installer Service?	6-3
How Do I Get Changes I Make in the IDE to Take Effect When I Run My Setup?	6-4
How Do I Display a License Agreement?	6-4
How Do I Create Shortcuts?	6-4
How Do I Associate File Types with My Program?	6-5
How Do I Launch an External Application from My Setup?	6-5
How Do I Conditionally Install My Setup Based On the Operating System of the Target Machine?	6-5
How Do I Upgrade My Applications or Send Out Updates?	6-6
How Does Windows Installer Determine Which Files Should Be Overwritten?	6-6
How Do I Create a Self-Extracting EXE for My Installation?	6-6
Why Doesn't My Setup Fit on a Floppy Disk?	6-7
Disk Space	6-7
Disk Spanning	6-7
How Can I Create My Setup So it Installs Files Based on the Target Operating System? ..	6-7
What is the Difference Between Extracting COM Information and Self-Registration?	6-8
Where Can I Access Updated Merge Modules?	6-8



Appendix A	CD Browsers	A-1
	What is a CD Browser?	A-3
	Creating a CD Browser with DemoShield	A-3
	Creating a CD Browser: The Basic Process	A-4
Appendix B	Internet Distribution	B-1
	Preparing Your Setup for Internet Distribution	B-3
	Compressed Size	B-3
	Self-Extracting	B-3
	Digitally Signed	B-3
	Easy to Use	B-3
	Packaging Your Setup with PackageForTheWeb	B-4
	Glossary	G-1
	Index	I-1





Welcome to InstallShield Express – Visual FoxPro Limited Edition

Welcome to InstallShield Express – Visual FoxPro Limited Edition! This chapter describes some of the many features and benefits of Express, and provides information on how to get help when you need it.

Table of Contents


Features and Benefitsxiii
Easy-To-Use Interfacexiii
Easy Upgrade Path to InstallShield Professional—Windows Installer Editionxiii
Embedded Helpxiv
Support for Third-Party Technologyxiv
Microsoft Windows Installer Servicexiv
System Requirementsxiv
About InstallShield Software Corporationxv
InstallShield's Productsxv
Administrator Solutionsxv
Installation Developmentxvi
Multimedia Authoringxviii
Internet Deploymentxviii
Technical Support Resourcesxix
Online Helpxix
Help Libraryxx
Setup Mapxx
Context-Sensitive Helpxx
Web-Based Helpxx
InstallShield Knowledge Basexxi
InstallShield Newsgroupsxxi
InstallShield Training Coursesxxi
InstallShield Consulting Servicesxxi
InstallShield Subscription Planxxii
Document Conventionsxxii



Features and Benefits

InstallShield Express – Visual FoxPro Limited Edition provides you with an easy way to create fully functional professional installations in minimal time. With built-in support for many third-party technologies and an easy-to-use development environment, InstallShield Express – Visual FoxPro Limited Edition offers you an installation solution that is both effective and efficient. Some of the many Express 3 features are summarized on the following pages.

Easy-To-Use Interface

Express makes creating setups easy with its installation development environment (IDE). All of the work you need to do in Express centers around the setup checklist, which details and keeps track of every step of setup creation—from naming your project to distributing the final setup files to your target media. It does all this in a familiar Windows environment,  fully enabled with drag-and-drop functionality.

You don't need to do any scripting in Express. With the Express IDE and the tracking checklist, you'll be completing setups quickly and easily.

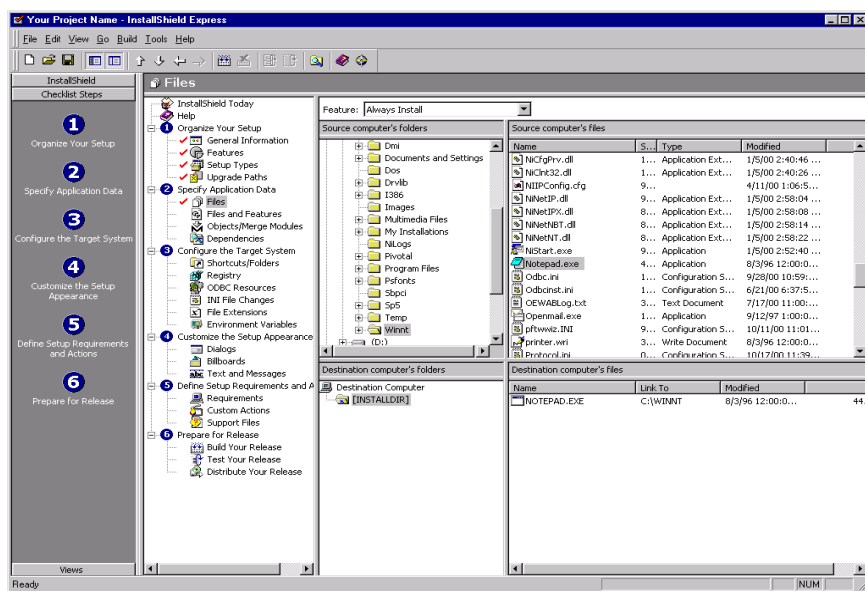


Figure I.
The InstallShield Express User Interface.

Easy Upgrade Path to InstallShield Professional—Windows Installer Edition

If you find that your setup outgrows the functionality available in Express 3, InstallShield provides a simple and direct upgrade path to InstallShield Professional – Windows Installer Edition. InstallShield Professional – Windows Installer Edition allows you to customize your setup to a greater extent by allowing you to edit dialog layout and content or create new dialogs, add additional installation conditions, cre-




ate patch installations, and support multiple languages simultaneously in a single setup file. InstallShield Professional – Windows Installer Edition also offers source code control, scripting functionality, and the ability to debug your setup in minutes.

Embedded Help

Express has an extensive compilation of help topics displayed directly in the IDE. These help topics contain information relevant to the current view. When applicable, these topics link to the Help Library to display more information.

Support for Third-Party Technology

You can quickly and easily add commonly used pieces of functionality to your setup by adding  merge modules. Each merge module has the files and logic to properly install the functionality they contain. You can add new merge modules just by placing them in the correct folder. Continuing users of Express will notice that merge modules assume the functionality that Objects provided in the past.

Microsoft Windows Installer Service

InstallShield Express uses the new Microsoft Windows Installer service, which provides a standard format for all installations. The Microsoft Windows Installer service provides several new and helpful features, including the ability to repair broken installations automatically and the ability to advertise features—that is, install features of your application only when your customer requests them.

System Requirements

For successful installation of InstallShield Express – Visual FoxPro Limited Edition, your computer must meet the minimum system requirements.

- Windows 95, Windows 98, Windows Me, Windows NT 4.0 (Intel) operating system with Service Pack 4 installed or Windows 2000 (Administrative privileges required)
- Microsoft Internet Explorer 4.01 (SP2) or higher if running on NT 4.0 or Win 95
- Pentium-class PC (300 MHz or higher recommended)
- 64 megabytes (MB) of RAM (128 MB recommended)
- 180 MB of free disk space
- Super VGA monitor (with the screen resolution set to at least 800 x 600, small fonts setting)



About InstallShield Software Corporation

InstallShield Software Corporation is a leading provider of professional Windows® software development tools. We focus on bringing innovative software distribution technologies to independent software vendors, corporate build and release engineers, and enterprise administrators worldwide by providing solutions for delivering applications from development to installation via all possible paths and mediums—especially the Internet.

In 1987, Viresh Bhatia and Rick Harold founded our company, adopting the mission of producing applications and software development tools for emerging Microsoft® Windows platforms. Today, with a vast network of international partners and a staff of more than 240 employees at our corporate headquarters in Schaumburg, IL, we provide industry-standard software distribution products worldwide.

Our dedication to providing the latest technology to our customers and their continued use of our software distribution solutions have led Media Metrix Inc. to rank InstallShield's products as the sixth-most-frequently owned and the eighth-most-frequently used software. Moreover, InstallShield recently received its third inclusion in the annual Inc.® 500, a ranking of the fastest growing private companies.

Whether you want to create bulletproof installations, solve enterprise distribution challenges, or deploy applications over the Internet, we offer the latest software and service solutions for your installation and distribution needs. Let InstallShield provide the software delivery solution you need today. Find out more about us and our products on our Web site www.installshield.com.

InstallShield's Products

In 1990, Microsoft introduced Windows 3.0, which quickly became the standard for graphical operating systems. InstallShield Software Corporation (then named the Stirling Group) was among the first in the marketplace to respond, offering the Shield series of tools for Windows application development. From this series, InstallShield and DemoShield® emerged as market leaders, filling the need for Windows installation and demonstration development.

InstallShield Software Corporation has continued to lead all competitors through the years, providing products and services that meet and exceed industry demands for software distribution.

Administrator Solutions

InstallShield Software Corporation advances its status as the industry's leader in software distribution by providing IT managers and system administrators with an easy-to-use, yet robust solution for software distribution and workstation management within the corporate network.

InstallShield Tuner

Released in September 2000, InstallShield Tuner marks the Network Administration field's first solution for graphically creating MST transform files. These MST files are used to alter or customize setups created for the Windows



Installer service. Native to Windows 2000 and Windows Me, and extendable to other Windows platforms, the Windows Installer service was created by Microsoft to reduce total cost of ownership for organizations by standardizing installations via several relational databases.

The file format (MSI) that targets the Windows Installer service cannot be altered before deploying across a network or an enterprise. Network Administrators, however, can use InstallShield Tuner to create MST transforms that allow file addition or subtraction, feature manipulation, and other customizations when installation occurs. The original MSI is not altered, and Administrators have an easy-to-use solution designed to meet their specific needs—only with InstallShield Tuner.

Installation Development

With a successful reputation honed over years of dedicated use to its bullet-proof systems, InstallShield Software Corporation is the undisputed industry standard for installation development.

InstallShield Professional—Standard Edition

InstallShield 1.0, released in 1990, was instantly recognized as the premier installation development tool for Windows applications. As developers created more intricate applications, InstallShield responded to their needs with more robust versions of its flagship product. InstallShield 3 was developed with setup guidelines formed by a relationship with Microsoft for the Windows 95 operating system. In March 1997, InstallShield 5 Professional provided users an integrated development environment for faster and easier creation of intricate setup applications.

With the release of InstallShield Professional 6 in September 1999, InstallShield has met developer needs with enhanced third-party technology support, new event-based scripting, and many other enhancements and features to lower installation development cycles and offer more powerful control over every aspect of the installation.

The January 2000 release of InstallShield Professional 6.1 adds an object development kit, support for Windows' new Application Manager, and more to the functionality of version 6.

InstallShield Professional—Windows Installer Edition

First released in July 1999, InstallShield for Windows Installer allows developers to create setups that support the new Microsoft® Windows Installer service, enabling an application to meet the installation requirements for the Windows 2000 logo. Additionally, InstallShield for Windows Installer reduces developers' total cost of ownership with support for Windows Installer features such as component level install and uninstall, application advertising, auto-repair of corrupt application components, and automatic rollback after a failed installation.

The January 2000 release of InstallShield for Windows Installer 1.1 included support for InstallScript™, InstallShield's familiar scripting language, as well as



support for Merge Module authoring and reuse. The April 2000 release of version 1.5 includes software patching for easy Internet update creation among other new features, making it the complete solution for Windows 2000 deployment from the leaders in software distribution.

InstallShield Express

In 1995, InstallShield Express met Visual Basic developers' need for a quick and easy-to-use installation development solution. This completely visual installation tool offered setup creation in minutes and quickly earned praise from developers for its ease of use.

In May 1997, the second release of the award-winning InstallShield Express was well received by rapid application developers. InstallShield Express 2 provided more custom support for Visual Basic and other rapid development environments than any other tool.

The May 2000 release of InstallShield Express 3 gives developers a rapid, productive solution for authoring Windows Installer setups. Easy migration to InstallShield Professional – Windows Installer Edition allows developers to choose whether they need speed or customization power to deliver their applications. A new interface means quick deployment has never been easier.

InstallShield Multi-Platform Editions

InstallShield entered the cross-platform market in September 1997 with the release of InstallShield Java Edition. The first available tool for producing true cross-platform installations, InstallShield Java Edition gave Java developers a single solution for their software deployment.

With features focused on extensibility and Web-distribution, InstallShield Java Edition 2, released in January 1998, further utilized the power of Java for installation building. InstallShield Java Edition 2.5 improved with localization features and flexible source and target directory support.

The January 2000 release of InstallShield Java Edition 3.0 gave greater customization power with reusable beans that developers may take from an included gallery or author on their own. Additionally, version 3.0 included a new, powerful IDE allowing greater flexibility and control throughout the installation authoring process.

The June 2000 release of InstallShield Java Edition 3.5 further defines the product as the standard for cross-platform development. New support for Linux as well as existing support for Windows and Solaris Platform Packs allows version 3.5 to let developers create more robust cross-platform installations than ever before.

InstallShield continues to evolve its offering, enhancing functionality to meet emerging needs of true multi-platform developers. In November of 2000, InstallShield co-developed in partnership with IBM, an entirely new family of Multi-Platform installation solutions, thus replacing InstallShield Java Edition. InstallShield introduced:

- InstallShield Express - Multi-Platform Edition



- InstallShield Professional - Multi-Platform Edition
- InstallShield Enterprise - Multi-Platform Edition

This new family of products further extends InstallShield de facto installation standard to the multi-platform world, allowing developers to create one powerful and consistent application installation that meets the needs of multiple platforms, including Solaris, Linux, OS/2, AIX, OS/400, and Windows. Offering three distinct products, InstallShield provides developers with the freedom to choose the features and functionality they need most. With InstallShield Multi-Platform Edition products, developers are freed from redundant coding and platform-specific errors, allowing them to create reliable, sophisticated installations that combine many different capabilities into one smart install.

Multimedia Authoring

InstallShield Software Corporation pioneered dynamic, software-specific demonstration technology. Multimedia Authoring, Technical Publications, Development, Sales, and Marketing departments can now target their audiences and show—not just tell—what their software can do.

InstallShield DemoShield

DemoShield's first two releases promoted the ability to create demos specifically for Windows platforms. DemoShield 3 became the first visual demonstration software, and DemoShield 4 allowed creation of a wider variety of demos. In July 1996, DemoShield 5's automated processes made it easier to show and sell Windows applications, solidifying DemoShield's position as the leading Windows software demonstration product.

The March 2000 release of DemoShield 6.5 gave users the flexibility and features to create a variety of demos that grab audience attention. DemoShield 6.5 offers easy-to-use yet powerful tools to create stunning visual software demonstrations, including pre-sales demos, CD browsers, tutorials, and quick tours.

Internet Deployment

InstallShield's Internet distribution products successfully deliver your bits and any installations you build with InstallShield's installation development systems, allowing you to enhance software downloads and maximize online sales.

PackageForTheWeb

Introduced in January 1997, PackageForTheWeb helps to ensure secure delivery of files by offering password-protection and digital signing with Microsoft Authenticode technology. Developers using PackageForTheWeb can easily create single, self-extracting packages containing ActiveX controls, applications, or any other types of files.

PackageForTheWeb 3.0, released in March 2000, is available to all InstallShield customers at no additional cost.



WebUpdate

Released in October 1999, WebUpdate 1.0 is a solution for software vendors and enterprise development groups looking for an easier method to ensure that end users have the most recent product version available. WebUpdate allows a user to easily find and download the latest release. As a result, support personnel aren't taxed with questions resolved in the most recent version.

WebUpdate can also enable a machine to automatically download any free updates or present users with the opportunity to purchase upgrades. The release of WebUpdate marks InstallShield's continued commitment to providing the software community with powerful solutions for easily distributing and deploying applications over the Internet.

Technical Support Resources

NOTE: *The resources described in this section are designed to support the full edition of InstallShield Express. Therefore, some of the topics contained in the help resources might not apply to the limited edition.*

Online Help

When you have questions about InstallShield, first consult the online InstallShield Help Library. Help is available both from the installation development environment's (IDE) Help menu and directly from particular IDE elements.

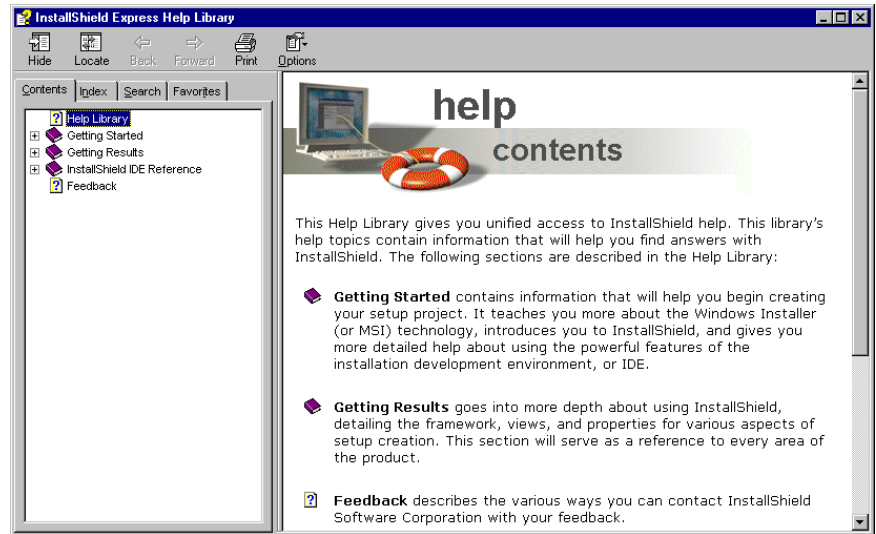
The help resources listed below are available from within InstallShield Express.



Help Library

The Help Library is your complete user's guide for InstallShield Express. This library contains reference materials that cover every aspect of your setup development.

Figure II.
The Help Library.



Setup Map

The Setup Map provides an interactive online tour of InstallShield Express. This resource is useful when you want to learn how to use Express.

Context-Sensitive Help

Online help is available for every dialog box, wizard panel, and view in the InstallShield IDE. You can press the F1 key from within the IDE to view context-sensitive help for a particular IDE element. In a wizard dialog, you can press F1 or click the Help button to view the related help topic.

Web-Based Help

Consult the InstallShield Web site (<http://www.installshield.com>) for immediate access to an extensive selection of useful information. The site is a comprehensive resource of free support materials including the following:

- Technical information and product news
- Sample projects
- Frequently asked questions
- White papers
- Documentation updates
- Maintenance releases and updates for registered software
- Information about consulting, support, and professional services

This information is available to you 24 hours a day, 7 days a week.



InstallShield Knowledge Base

The InstallShield Knowledge Base is located in the Support section of our Web site. It contains answers to many commonly asked questions and includes new information about InstallShield that may not appear in the documentation. You can use the Knowledge Base search engine to search for articles by phrase, article number, platform, and product version.

InstallShield Newsgroups

The newsgroups are another excellent resource when you have questions about InstallShield Express. In these newsgroups, developers like you share tips and ideas and help each other get the most out of InstallShield Express. Visit the InstallShield newsgroups at news.installshield.com.



InstallShield Training Courses

An InstallShield Training Course can show you everything you need to know to create setups that install perfectly the first time and every time. In an InstallShield Training Course, you learn how to take advantage of the Windows Installer service with the latest InstallShield tools, allowing you to streamline installation development and tackle the complexities of software deployment.

InstallShield Training Courses are offered in five United States locations and several more around the globe or can be conducted at your place of work. For more information:

- Call (888) 560-6933.
- Visit <http://www.installshield.com/training>.
- Email training@installshield.com.

InstallShield Consulting Services

If you have very complex requirements for your setup, or are just pressed for time, you might consider hiring InstallShield Consulting Services. InstallShield Consulting Services provides access to a group of experts who are trained in every aspect of Windows software deployment and who specialize in helping developers create highly sophisticated and customized installations for Windows applications. The services that InstallShield Consulting Services can provide include the following:

- Analysis and troubleshooting of existing setups
- Specification, design, development, integration, and testing of new setups
- Guidance in meeting Microsoft logo requirements
- Customized training and knowledge transfer

For information on InstallShield Consulting Services:

- Call (888) 999-1334.
- Visit <http://www.installshield.com/consult>.
- Email consulting@installshield.com.



InstallShield Subscription Plan

Purchase orders are a hassle. New software approval takes time and is sometimes not worth the effort. Then, when a new version appears six months later with hot new features you want to utilize, you have to start the process all over again. Then again, maybe you don't...

What if you could purchase industry-standard InstallShield Professional products once, invest in future releases of the same product so you can stay current without the hassles of reqs and purchase orders, and save significant money while you do it? With InstallShield's Subscription Plan, you can!

When you upgrade to the latest release of an InstallShield Professional product, you can join the InstallShield Subscription Plan. You receive a year's worth of new upgrades, you stay current on InstallShield technology, and best of all, you need to get purchase approval only once! All this for only \$499!

Take advantage of:

- Convenient Purchasing: Submit only one order per year.
- Cost Savings.
- Latest Technology: Use hot new features.
- Free Shipping: Price includes shipping and handling fees.

Interested? See how it works and how much you can save at <http://www.installshield.com>. Select a Professional product and click on Subscription Plan for more information.

Document Conventions

To enhance readability, certain text elements are specially identified as follows:

NOTE: *Helpful information, tips, and warnings you should keep in mind as you use this product are set off by horizontal bars, and can appear in either the main text flow or in the margin.*

Italic text denotes points of emphasis.

Bold text denotes commands, menu items, selectable options, Internet and email addresses.

Monospace text denotes command line input, file listings, scripts, and path specifications. It can appear in a shaded area, depending on its length.



Setup Fundamentals

This chapter explains the nature of setups and how they work. It also describes how your installation benefits from using Microsoft's Windows Installer service.

Table of Contents

What is a Setup?	1-3
Typical Elements of a Setup.	1-3
Perform File Transfer	1-3
Display User Interface.	1-3
Create Shortcuts.	1-4
Register File Associations	1-4
Register COM, COM+, and DCOM Files	1-4
Register Product for Uninstallation	1-5
What is Windows Installer?	1-5
Benefits of Windows Installer	1-5
Self-Repair.	1-5
Feature Advertisement	1-6
Componentized Installation	1-6
Rollback	1-6
Single File	1-6
Merge Modules	1-6
User-Accessible Information	1-6
Windows 2000 Logo Compliance	1-7
Clean Uninstallation	1-7





What is a Setup?

Even if you don't know what a setup is, chances are you've used one before. If you've ever installed software onto your computer, you've seen a setup from the end user's perspective. A setup program installs an application onto an end user's computer.

Typical Elements of a Setup

The typical setup performs a number of different functions. A setup transfers files from the source medium to your local drive. It also makes the required registry entries, creates shortcuts, and registers COM servers. In addition, a setup commonly gathers information about the target machine and the end user. The elements of a setup are described in detail below.

Perform File Transfer

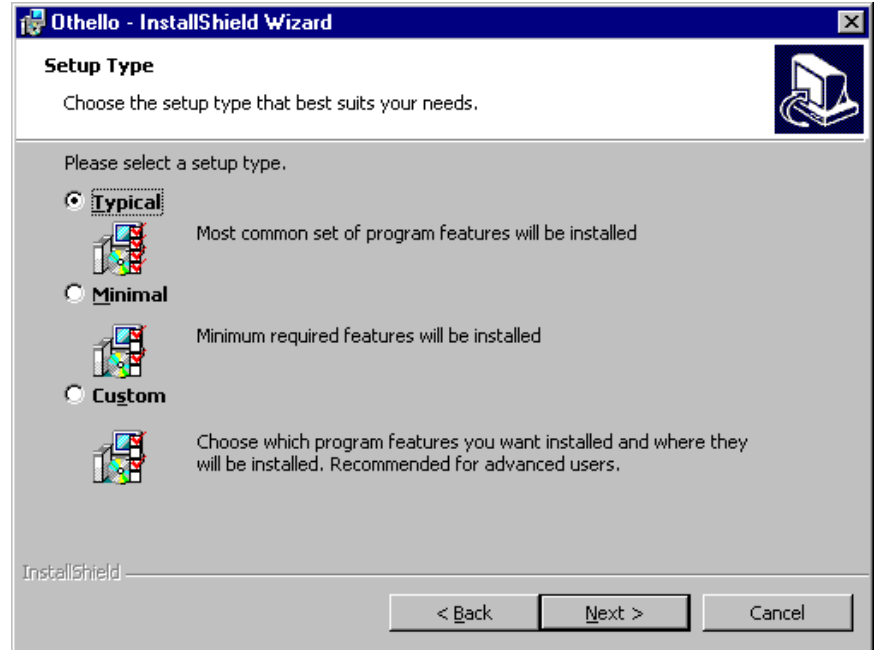
File transfer involves copying files from the source medium, such as a CD or a floppy disk, to a local drive on the end user's machine. Depending on the configuration the end user chooses, all or only some of the files may be transferred to the local drive. During file transfer, a setup can display billboards that provide product information such as new features or usability tips. The setup may also display a status bar to show the progress of the file transfer process.

Display User Interface

The user interface of a setup provides information and setup configuration options to the end user. Through the user interface, an end user can choose to install only part of a product, choose to leave some files on the source medium, view a license agreement, or provide information to the installer that may be necessary to ensure proper installation configuration. The user interface can be



Figure 1-1.
An example of a run-time dialog.



Create Shortcuts

Shortcuts are links to files and applications that the setup creates on the end user's machine. Shortcuts are usually placed on the desktop or the Start menu of the target machine to provide quick and easy access to a program or its files.

Register File Associations

If your product uses a distinct file type, you must register the file type for your application on the end user's system. For example, InstallShield creates a file with an .ism extension. In order for a file type to be recognized by your end user's system, it must be registered in the system registry. For example, if your application creates and uses files with the .foo extension, you need to register this file type during your application's setup.

Register COM, COM+, and DCOM Files

COM servers (such as ActiveX, COM, and COM+ files) require special registration so that applications can access the files' interfaces. Traditionally, these EXEs, DLLs, and OCXs contain self-registration functions that are invoked to register the files during installation. However, relying on self-registration can cause problems: the end user can't always be certain of what information is being registered or that the registry entries are removed completely when the file is uninstalled.

The Windows Installer service solves this problem by writing the necessary registry entries during setup and then removing them when the COM component is unin-



stalled. This method ensures that all the COM servers are registered appropriately whether the file is immediately installed or advertised.

Register Product for Uninstallation

In order to uninstall a product, the operating system must know that the product is present. To accommodate this, a setup registers an application with the operating system so that it can be easily uninstalled. This registration is required for Windows 2000 logo compliance.

Much of the information registered in this process is available to the end user through the Add/Remove Programs dialog box (on Windows 2000 systems). For example, technical support contact information, product update information, product version, and product publisher information are all registered in this process.

What is Windows Installer?

The Windows Installer service incorporates application installation and management into the basic Windows infrastructure. The operating system is then able to track installed applications and manage components that are shared by various applications. Thus, Windows Installer adds a layer of intelligence to the installation process, providing users with a more comprehensive view of application management. Windows Installer brings a number of new features and time-saving procedures that traditional setups lack.

This service gathers installation information from .msi files. An .msi file is actually a relational database. Each table in the database contains instructions and setup information. Windows Installer creates a script of actions from this database. These actions are then executed in the order in which the installer encounters them. An .msi file is the compiled output of InstallShield Express project (.ism) files.

Windows Installer is a native service of Microsoft Windows 2000 and Windows Me. This service can also run on Windows 95, Windows 98, Windows NT 4.0. If the Windows Installer service is not present on the end user's system, your Express setup can install it automatically.

Benefits of Windows Installer

In addition to providing a standardized method of handling installation and uninstallation, Windows Installer has many other benefits, such as advertisement and rollback. The following section describes a few of the more exciting features that Windows Installer offers.

Self-Repair

By adding a few lines of code to your application, you can protect against missing and corrupted files. If—after installation—a file becomes corrupted or is accidentally deleted, the Windows Installer service automatically reinstalls the file.



Feature Advertisement

Windows Installer allows you to advertise features. Advertised features can appear on an end user's system in the form of shortcuts and registered file types, but are not actually installed until needed. For example, you can advertise your help system so that no hard disk space is taken up until the user first accesses the help. When the user clicks a shortcut to the help for the first time or you program-matically access the help, the installer installs the necessary files and launches the help system.

Componentized Installation

Your product can be installed on a feature-by-feature basis. For example, if you have split your setup into two features (Program_Files and Help_Files), your end users can choose to install only Program_Files and not Help_Files. Your users can also uninstall features as necessary without removing your entire application. This aspect of Windows Installer is especially convenient for end users who have limited hard drive space.

Rollback

When a traditional setup fails, little is done to protect the end user's system from harm. In traditional setups, necessary registry entries or files can be deleted without maintaining a backup of the original configuration. The roll-back capabilities in Windows Installer ensure that the end user's system is returned to its pre-installation state if the setup fails for any reason. During a setup, Windows Installer identifies the files that have changed and makes a backup copy of those files. If the setup fails, those files are returned to their previous state, leaving the system unchanged.

Single File

All of your setup files can be streamed into a single .msi file for easy distribution. Because you don't need to ship your files separately from the installer package, this makes deployment much less complicated. Your setup application can also be shipped in the traditional Setup.exe format, which checks the target system for the Windows Installer service, installs it if necessary, and then launches your compiled .msi file.

Merge Modules

Windows Installer supports the use of merge modules, which allow you to add distinct pieces of functionality to your setup without having to recreate that aspect of the setup every time you want to use it. For example, if your application requires a standard Visual Basic library in order to function properly, you can import the merge module that contains those Visual Basic components rather than adding all the necessary files to your setup each time.

User-Accessible Information

The Add/Remove Programs applet in Windows 2000 provides information about installed products. This information includes links to Web sites that



offer support for the product and sites that host product updates. The software publisher and product version number are also displayed here. InstallShield Express provides full support for these general information tables, making it easier for you to pass information on to your end users.

Windows 2000 Logo Compliance

Use of Windows Installer is required in order to achieve the Windows 2000 logo. Although this is only one factor in achieving compliance, it is an important consideration. InstallShield Express takes you one step closer by using the Windows Installer service and following the Windows Installer best practices guidelines.

Clean Uninstallation

Since the Windows Installer service maintains a database of all the files that are installed and all the changes that are made by the service, uninstalling programs is much more reliable.





For Continuing Express Users

Table of Contents

Migrating Your Existing Express Projects	2-3
Migrating Multi-Language Setups	2-3
Steps to Migrate	2-3
Files, File Groups, and Components	2-3
Setup Types	2-3
InstallShield Objects	2-4
Express Extensions	2-4
Registry Entries	2-4
Shortcuts	2-4

This chapter gives an overview of how to migrate your older InstallShield Express Installations (2.13 and earlier) into Express 3. The contents of this section contain general information on the conversion process. If you need more specific information, see the online help included in Express or the Knowledge Base, available from the Support section of InstallShield's Web site.





Migrating Your Existing Express Projects

If you have a setup project that you created using a previous version of InstallShield Express, you can import that project into Express 3 to take advantage of the new features available. Because of the technological differences between older versions of Express and the current version, not all of the information in your older projects may be converted in your new Express 3 project.

You can migrate any previous Express setup projects to version 3. Version 2.x projects retain more information than setup projects created with version 1.x, due to the lesser distance between the underlying technologies that the versions use.

Migrating Multi-Language Setups

If you migrate a multi-language setup that you created using Express 2.x, the strings are in English in Express 3.

Steps to Migrate

To open a setup project that you created with an earlier version of Express, do the following:

1. Select Open from the File menu or click the Open button on the toolbar. The Open InstallShield Project dialog box appears.
2. In this dialog box, select “InstallShield Express Projects (*.ism; *.iwz)” from the “Files of type” drop-down list and browse to the Express project you want to open.
3. Click the Open button to begin migrating the project.
4. Verify that everything has transferred properly.

As noted previously, not everything in your earlier setup projects is converted to your new Express 3 project. Express 3 creates setups that use an entirely different framework than previous versions of Express did. The following sections explain how elements of your older projects are migrated to Express 3.

Files, File Groups, and Components

In older versions of Express, files were added to file groups, and file groups were added to components. In Express 3, files are added directly to features. No further levels exist. Therefore, when you migrate your setup project, files are no longer associated with file groups. Instead, all files that were part of your older Express project belong to a feature in your converted Express 3 project. For every component in your old project, a feature is created in your new project. Your file groups map to destination folders.

Setup Types

Setup types in Express 3 provide functionality similar to that of earlier versions. The major difference between the two is the fact that setup types used to



be based on components. In the new version of Express, components do not exist and setup types are instead based on features. The setup types you specified in your old Express project are recreated in the new Express project, with features taking the place of components.

InstallShield Objects

InstallShield objects have been replaced with merge modules. If a merge module exists that can provide the functionality provided by the object you used in your project, that module is added to your setup. Otherwise, you need to find the merge module (either within Express or elsewhere) that contains the functionality that you need to include, and add it to your project.

Express Extensions

Express extensions have been replaced by custom actions. With custom actions, you can call a function from a DLL or launch an executable, just as in previous versions of Express. If you had an extension that performed one of these two tasks, that extension is converted into a custom action in your new project.

Registry Entries

All of the registry data in your previous project is added to your new project.

Shortcuts

Shortcuts in Express 3 are much more robust than those found in previous versions. Any shortcuts you had in your old setup are migrated into your new setup project. Due to their more robust nature, you might want to review and update the shortcut properties in your new project.

For more information on migration, see the online Help Library.



Using InstallShield Express – Visual FoxPro Limited Edition

InstallShield Express – Visual FoxPro Limited Edition provides a complete, easy-to-use installation development environment (IDE) for creating your setups. The IDE provides a checklist of items that you need to consider as you author your setup. This chapter takes you through these steps and gives you the information you need in order to begin creating setups of your own.

Table of Contents

The Setup Project File	3-3
Tour of the IDE	3-3
Toolbar	3-3
Viewbar	3-4
Checklist	3-4
Views	3-4
Step 1: Organize Your Setup	3-5
General Information	3-5
Features	3-5
Setup Types	3-6
Upgrade Paths	3-7
Step 2: Specify Application Data	3-7
Files	3-8
Files and Features	3-8
Objects/Merge Modules	3-9
Dependencies	3-10
Step 3: Configure the Target System	3-11
Shortcuts/Folders	3-11
Registry	3-11
ODBC Resources	3-12
INI File Changes	3-13
File Extensions	3-14
Environment Variables	3-15
Step 4: Customize the Setup Appearance	3-15
Dialogs	3-16
Billboards	3-16
Text and Messages	3-17
Step 5: Define Setup Requirements and Actions	3-17



Requirements	3-17
Custom Actions	3-18
Support Files	3-18
Step 6: Prepare for Release	3-19
Build Your Release	3-19
Test Your Release	3-19
Distribute Your Release	3-20



The Setup Project File

The setup project file is your central storage file for all of the information required to build your setup. The information that you enter into the various checklist steps is saved to this file, including your setup's dialog boxes, application system requirements, and files. The setup project file provides a convenient interface between Express and compiled setups.










Tour of the IDE

This section discusses the IDE components and how you use them to create setups. The terms and concepts presented here appear throughout the remainder of this guide.

NOTE: *Some features described are available only in the full edition of InstallShield Express. Views that are not supported in this limited edition of InstallShield Express are indicated with a note.*

Toolbar

Like many other Windows programs, InstallShield Express has a toolbar. The toolbar contains shortcuts to commonly used functions. The following functions can be accessed by clicking on the button shown.

-  The New button creates a new project.
-  The Open button opens an existing project. The Open InstallShield Express Project dialog prompts you to browse to the desired file.
-  The Save button saves the current project to the location and with the file name that you specified when you created the project.
-  The Toggle Viewbar button hides or displays the viewbar on the left side of the IDE. The viewbar is described later in this chapter.
-  The Toggle Checklist button hides or displays the checklist on the left side of the IDE. The checklist is described later in this chapter.
-  The Previous View button takes you to the view directly above the current view as shown in the checklist.
-  The Next View button takes you to the view directly below the current view as shown in the checklist.
-  The Back button takes you to the view that you last visited in the history of your view selections. This can be done multiple times, as long as there are multiple entries in your view history.
-  The Forward button takes you to the next view in the history of your view selections. This can be done until you reach the view you were at when you first clicked the Back button.



The Build button builds your setup project into a functional setup.



The Stop Build button halts the current build and cleans up any files that the build created.



The Run button runs your setup on your computer, going through all dialogs, installing files, and making any specified system changes as if you had run the setup by double-clicking the setup executable.



The Test button runs the user interface portion of your setup, but does not install any files or make any system changes unless those changes are part of a custom action.



The Windows Explorer button launches the Windows Explorer, in which you can manipulate files and folders.



The Help View button displays the Help view, in which you can find the answers to many of your setup questions.



The Setup Map button launches the Setup Map, an interactive overview of InstallShield Express.

Viewbar

The viewbar allows you to navigate through various views in InstallShield Express. When you first open InstallShield Express, the viewbar displays only two items: InstallShield Today and Help. Above these items, a header bar reads, “InstallShield.” This indicates that you are in the InstallShield viewbar.

When you open an existing project or create a new project, two new viewbars appear: Checklist Steps and Views. Initially, the Checklist Steps viewbar is displayed and the Views header bar appears at the bottom of the viewbar. The InstallShield header bar appears above the Checklist Steps viewbar. To toggle between the three viewbars, click the desired header bar and the viewbar displays the views for the selected header bar.

Checklist

The checklist is your guide to creating a setup. The checklist lists every view necessary for creating and testing your setup. The first time you visit one of these views, Express puts a checkmark to the left of the name of that view. This allows you to see at a glance which views you have visited.

Views

Views are where you do most of the work in creating your setup. The views are organized into six steps, where each step is a logical group of information that is required to build a functional setup. To display a view, you can either click on the name of the view in the Views viewbar or in the Checklist. Below are the steps to consider when you create a project, and the corresponding views where you can complete these steps.



Step 1: Organize Your Setup

The first step in creating your setup is to provide basic application information and lay the foundation for your setup's structure. In these views, you provide information about your company and your application, define the logical groupings that will eventually hold all of your application data, specify the options that end users have when installing your application, and define upgrade specifications.

General Information

This view allows you to specify contact information, product information, and general setup features by completing the fields in the property grid. While the bulk of the information you enter here appears in various end-user dialog boxes, some are solely for your own use (such as the Comments property). You can also indicate whether or not end users can repair, remove, or modify your product from the Add/Remove panel. This view contains a number of important fields, so you should review the properties here carefully.

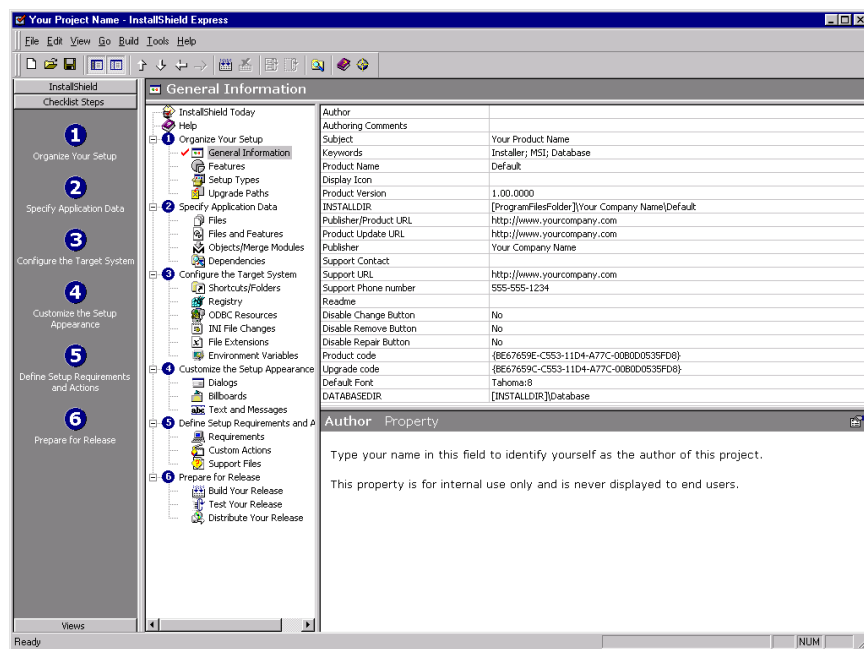


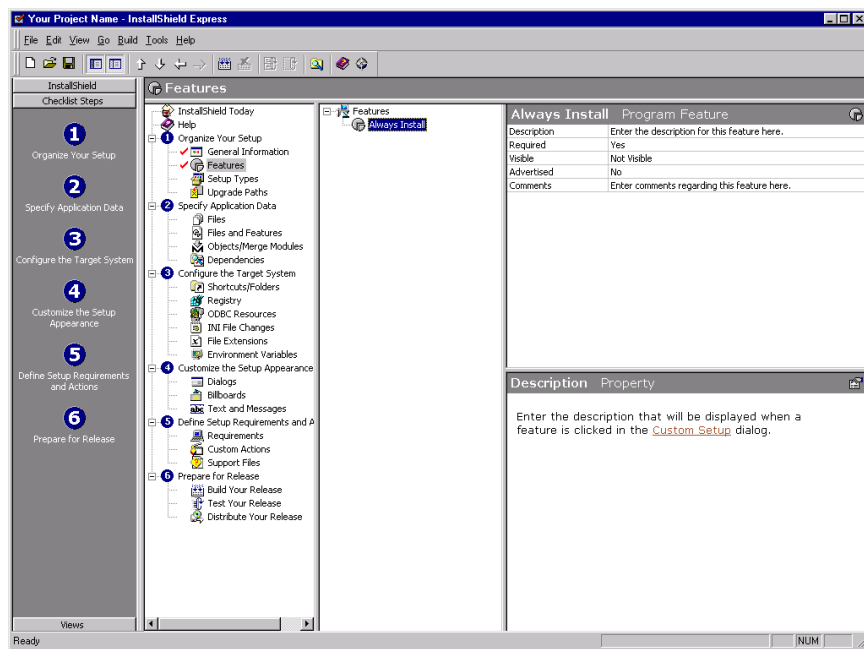
Figure 3-1.
The General Information view.

Features

The Features view lets you add and define the features that your installation contains. Features are logical groupings of functionality within your application and are used to organize all of your application's data into different units that end users can select at run time. For example, your application could be divided into an Always Installed feature (that contains your required application data), a Help feature, and a Tutorial feature.



Figure 3-2.
The Features view.



Setup Types

The Setup Types view allows you to organize the features you created in the previous step into setup types. Setup types are the various configurations from which end users can select to install your application. By default, the three setup types are Typical, Minimal, and Custom. You can rename these setup types, deselect them so they are not included in your setup, and change the features they install, but you cannot add new setup types. In the Custom setup type, end users can select the features they want to install from the features



listed—as long as the features’ Visible properties are set to visible in the Features view.

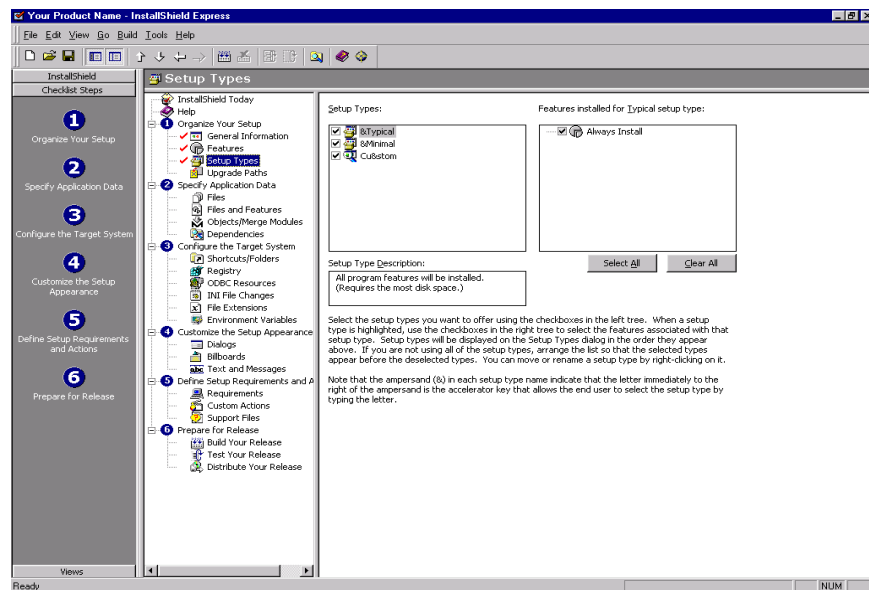


Figure 3-3.
The Setup Types view.

Upgrade Paths

NOTE: This feature is available only in the full version of InstallShield Express.

The Upgrade Paths view allows you to indicate upgrade information if you have previously distributed different versions of your application. In order for an upgrade to be successful, the product code and version of your new application must be different from your previously released application. In this view, you can provide the upgrade code, specify versions and languages, and select criteria for upgrading an end user’s system if he or she has a previous version installed.

Step 2: Specify Application Data

Now that you have outlined your application’s basic information and your setup’s structure, you need provide specific application data. InstallShield Express has several views dedicated to this purpose. You can add your files to predefined destination folders, subfolders of predefined destinations, or to hard-coded destination directories, and then change the features with which they are associated.

To save time, you can scan your application executable for dependent files, scan a Visual Basic project, or add merge modules (prepackaged groups of com-



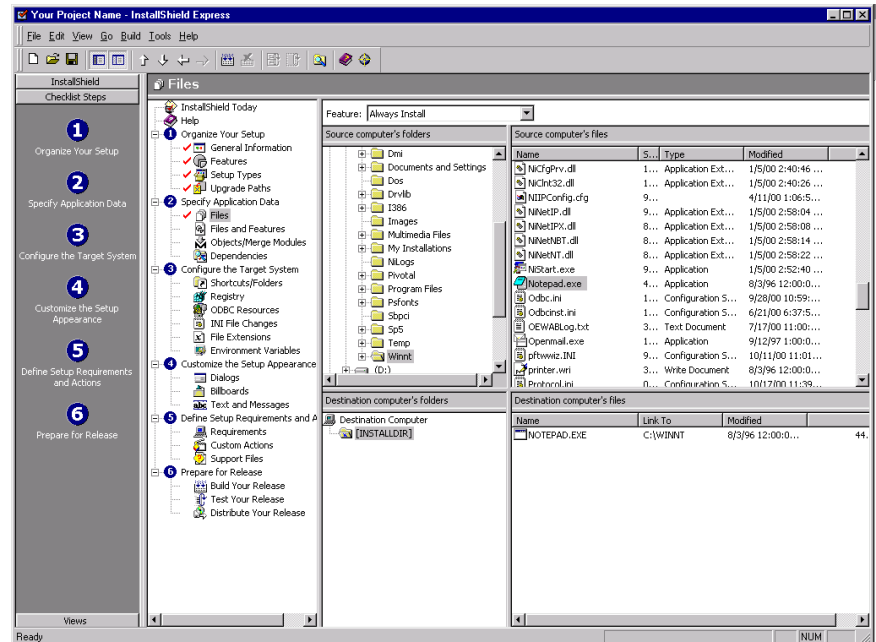
monly used application files). You can combine any or all of these processes to create your final application structure.

Files

In the Files view, you can select your application files from their location on your system and place them in the destination computer location. The interface is split into two sections: the top is a display of your system, the bottom is a display of the destination system. In the destination portion of the display, you can select predefined destination folders from a context menu or create your own hard-coded destination directories. These folders represent destinations on the end user's system.

The Files view operates like Windows Explorer, so you can drag files from the top section to the bottom section. The files that you move are assigned to the feature displayed in the drop-down Feature list above the display. You can change the feature to which you are adding files at any time by selecting a different feature from the drop-down list.

Figure 3-4.
The Files view.



Files and Features

The Files and Features view allows you to modify your file and feature associations. For example, perhaps you moved all of your application files into destination folders (in the Files view) without changing the feature in the Feature drop-down list. In the Files and Features view, you can drag files from one feature to another without affecting the files' destination folders. You can also copy files from one feature and paste them in another, if you need a particular



file installed with multiple features. The copied file retains all of its properties within InstallShield Express.

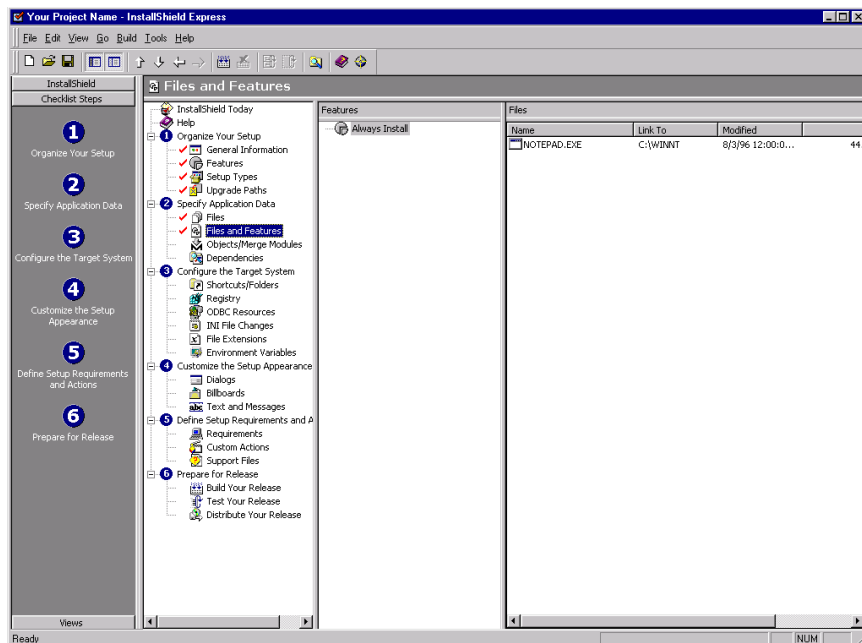


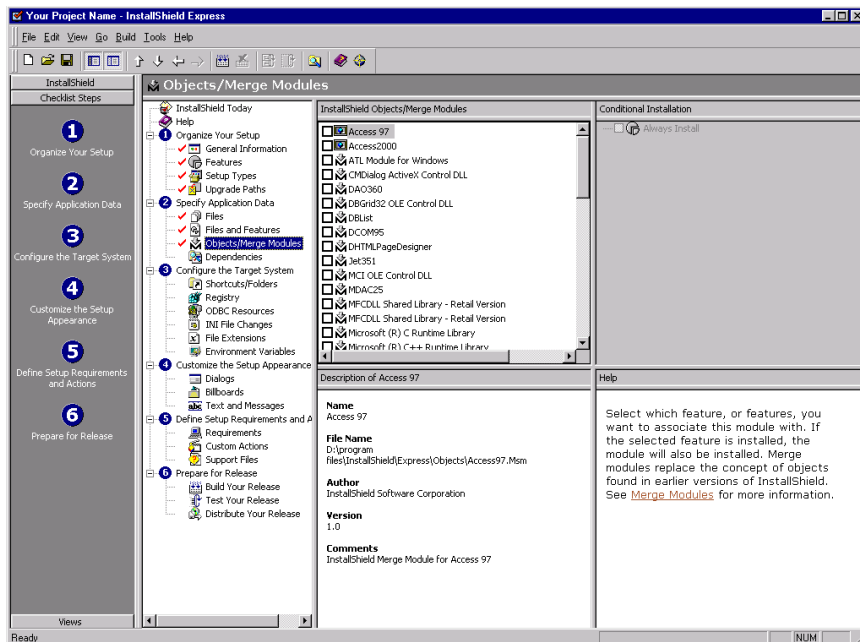
Figure 3-5.
The Files and Features view.

Objects/Merge Modules

In the Objects/Merge Modules view, you can add groups of functionality—known as merge modules—to your setup. This is helpful if your application requires commonly used file libraries, such as an MFC library. This way, you don't need to collect and manually add all of these files every time you package



Figure 3-6.
The Objects/Merge Modules view.



Dependencies

NOTE: *This feature is available only in the full edition of InstallShield Express.*

The Dependencies view lets you perform several types of scans on your application to determine which files you need to distribute with it. Express then includes those files automatically. You can perform a static scan on all the applicable files included in your setup, a dynamic scan on your application while it is running, or a scan on a Visual Basic file to see which files your application needs in order to run correctly. All three are great time-saving methods of gathering your application files.



Step 3: Configure the Target System

Applications often have shortcuts, registry entries, INI files, and file extensions associated with them. In the views included with this step, you can configure these less tangible application components simply and quickly.

Shortcuts/Folders

The Shortcuts/Folders view gives you the means to place shortcuts to your application's files on the target system. There are already several default shortcut folders listed here for your convenience, including the Desktop, the Program Menu, and the Send To menu. You can add subfolders and shortcuts to these defaults.

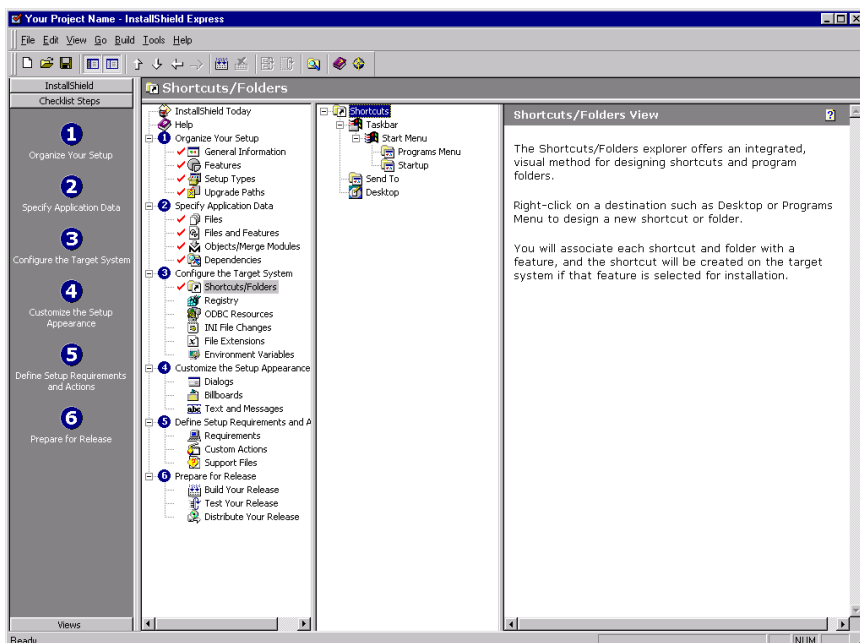


Figure 3-7.
The Shortcuts/Folders view.

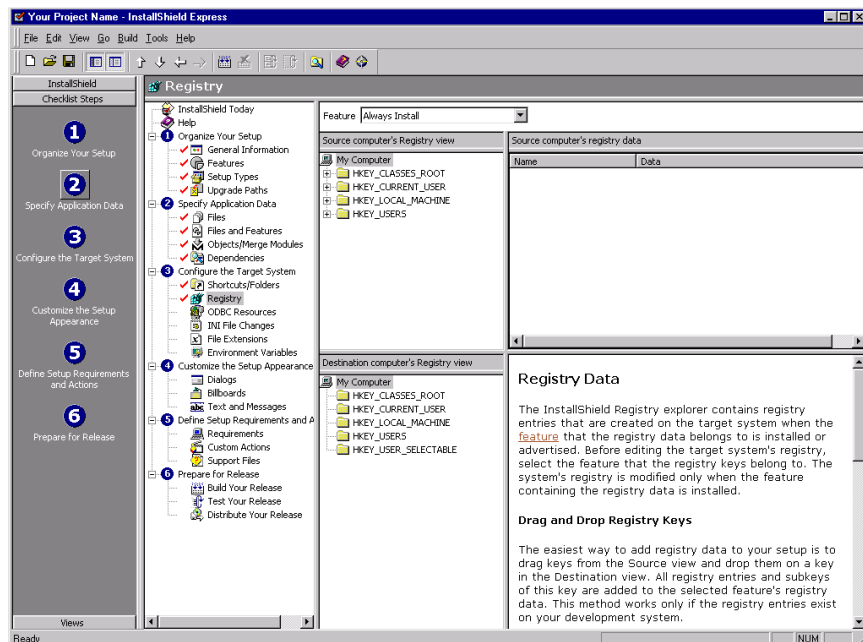
Registry

The Registry view lets you arrange the changes you want to make to the target system's registry. If you have all the registry entries organized on your machine as you would like them to be organized on the target machine, you can drag them in the same way you did with files in the Files view. If you prefer to organize your registry entries as REG files, you can import those files in this view and all registry entries in the REG file appear as registry entries in the



Figure 3-8.
The Registry view.

target registry. You can add your entries manually, as well as edit any entries you have added.



ODBC Resources

The ODBC Resources view allows you to include in your setup any ODBC driver, data source name (DSN), or translator that is installed on your system. To add an ODBC resource, select the resource you want to include from the ODBC Resources list (in the upper-left corner of the view) and select the feature with which you would like this ODBC file to be installed from the Features list (in the upper-right corner of the view). Below the list of ODBC resources are the properties for the selected ODBC resource, as configured on



your system. You can edit the values for all ODBC resources, and you can add new properties to this list for drivers and DSNs.

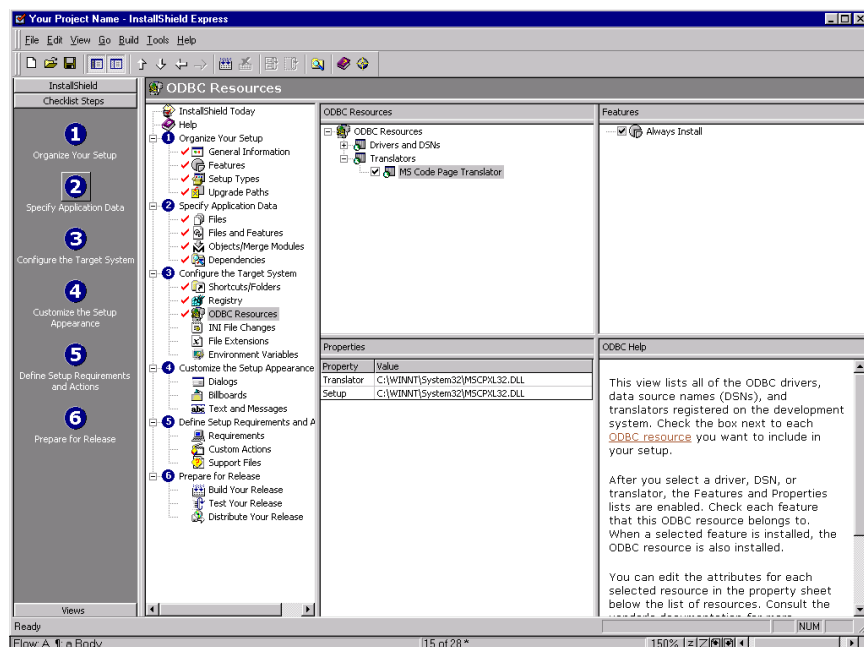


Figure 3-9.
The ODBC Resources view.

INI File Changes

In the INI File Changes view, you can specify any changes that you need to make to existing INI files on the target system. To make a change, add the file you want to change, the section name, and the keyword. To the keyword, add the specific value that you want to associate with this keyword and the method by which to add this value.

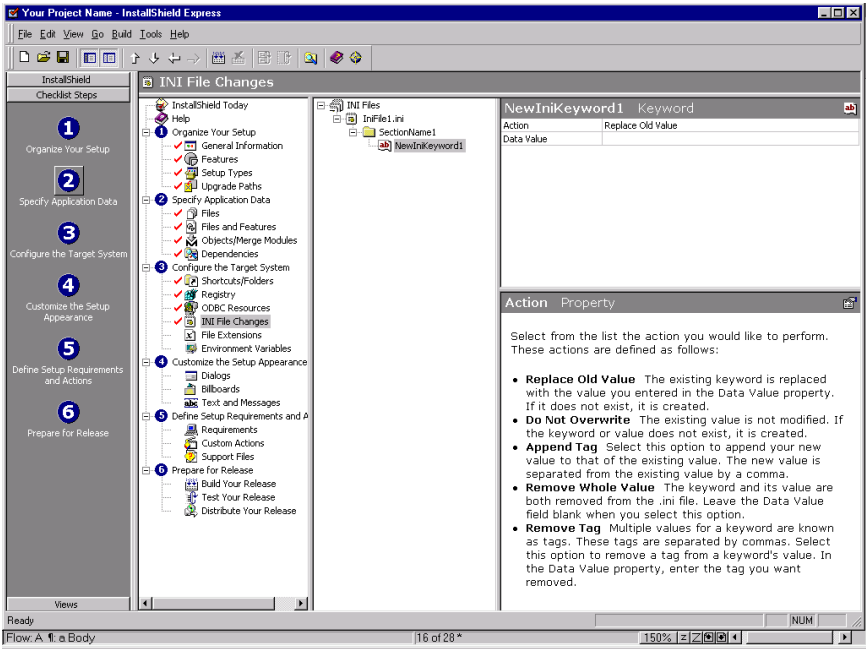
For example, if you want to change the dBASE driver listed in the Odbc.ini file, do the following:

1. Add the “Odbc.ini” file.
2. Add the “dBase Files” section.
3. Add the “Driver32” keyword.



4. Add the path to the new driver as the value.

Figure 3-10.
The INI File Changes view.



File Extensions

The File Extensions view allows you to associate a specific file type with an executable file that is included in your setup. This view eliminates a lot of the work involved in creating a new file association. Usually, you have to create a registry entry for each file extension. In Express, it is much simpler: You can add an extension, and rename it to the file extension you want to associate with an executable file. Next, select the executable file from your setup, select



an icon file for the file type, and provide the command and arguments to run it.

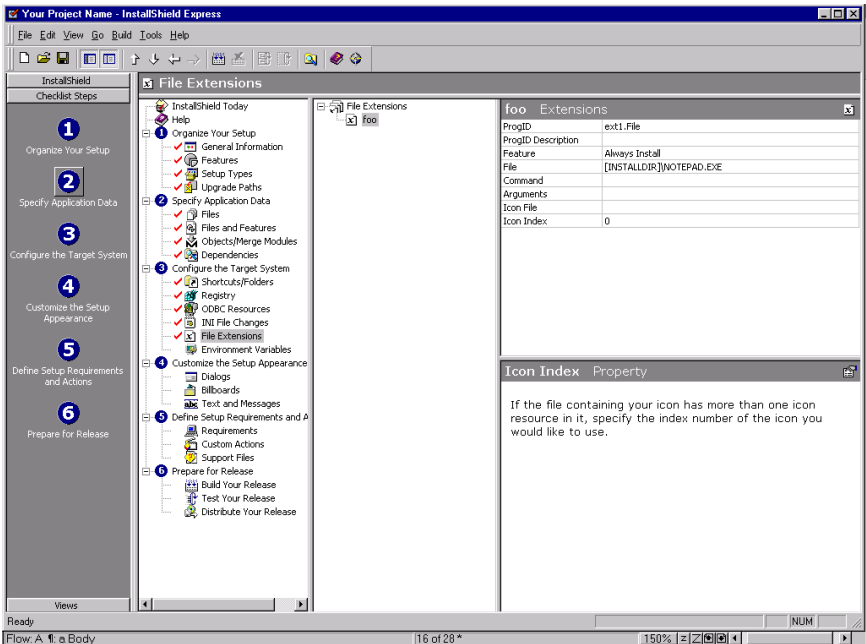


Figure 3-11.
The File Extensions view.

Environment Variables

NOTE: This feature is available only in the full edition of InstallShield Express.

In the Environment Variables view, you can create, modify, and remove environment variables on the target system via your installation program. To affect environment variables on the end user's system, click Environment Variables in the checklist or the Views viewbar, select Add Environment Variable from the context menu, and complete the properties in the grid to the right.

Step 4: Customize the Setup Appearance

After you've organized all of your application's data and determined the changes it makes to a target system, you can focus on the delivery interface for your setup. What your customers see during an installation has a big impact on how they regard your product and your company. This view in this step gives you the opportunity to customize all of the visual aspects of your setup in order to create an appealing installation experience for end users.



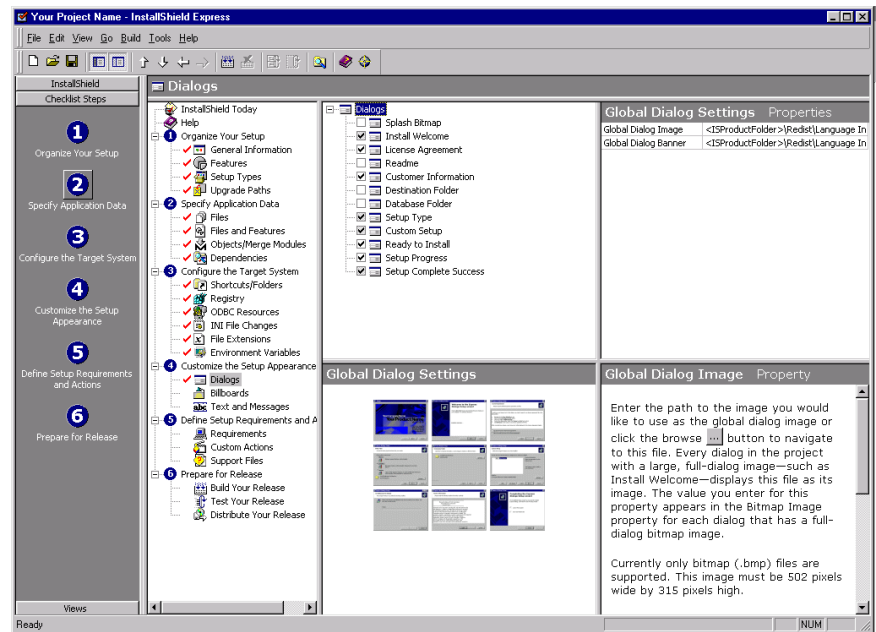
Dialogs

The Dialogs view allows you to determine which of the InstallShield Express standard end-user dialogs your customers see during an installation. Some of these dialog boxes are required for all setups and cannot be removed. Required dialogs have a checkbox with a grayed-out background to the left of them.

You can select or deselect all other dialogs in this view. For each dialog box you include in your setup, you can customize its attributes using the property grid. The Global Dialog Settings feature allows you to specify the bitmap (.bmp) images that are to be used as banner and whole-dialog images for all dialogs. This eliminates the need to indicate a value for each dialog image property separately. If you want particular dialog to display a different image, you can modify the image property value for that dialog.

Click a dialog in the list to view a sample image in the lower-left corner of the view. Note that the sample image is static and does not reflect the values you provide for that dialog's properties.

Figure 3-12.
The Dialogs view.



Billboards

NOTE: *This feature is available only in the full edition of InstallShield Express.*

In the Billboards view, you can add any number of bitmap (.bmp) images for display during file transfer. This process is optional, but billboards give you the opportunity to advertise, inform, or entertain end users while your application installs. Billboards are highly customizable—you can set the position,



duration, background color, and transition effects to make the file transfer a more interesting experience.

Text and Messages

NOTE: *This feature is available only in the full edition of InstallShield Express.*

The Text and Messages view gives you access to every string in the installation, including dialogs and error messages that might be displayed. Each string has a default value associated with it, but you can modify any of these values. For each dialog or message box, Express displays a sample image, so you can see where each string appears on the user interface.

If you want your setup to run in an unsupported language, you can globalize your setup by exporting the strings for translation and importing the translated strings back to your setup.

Step 5: Define Setup Requirements and Actions

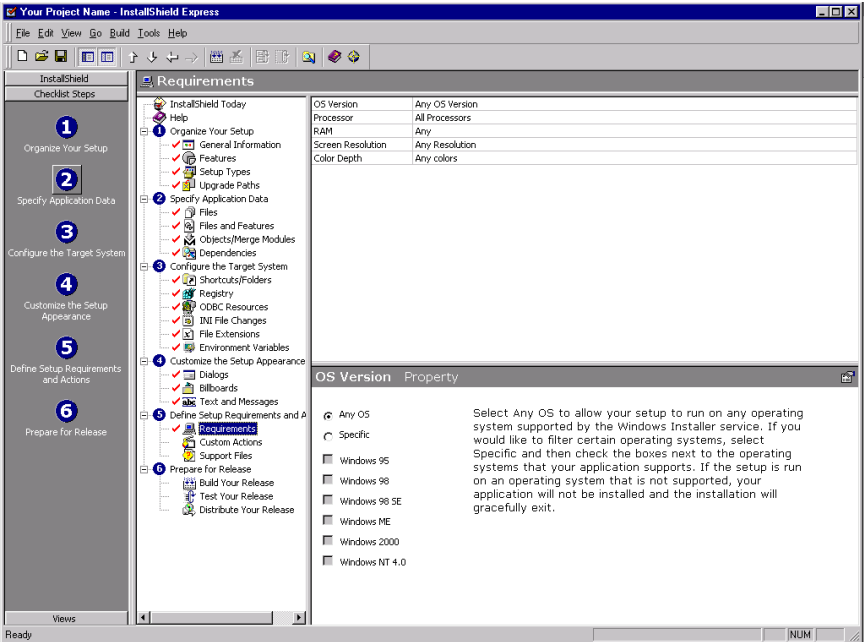
Your application and its setup may have special requirements. The views in this step let you set target system requirements, add additional functionality through custom actions, and indicate support files that are available to your setup only during installation.

Requirements

The target systems need to be able to provide the processing power, video capabilities, and memory to ensure that your application runs smoothly and properly. In the Requirements view, you can specify what capabilities the target system must have in order for your product to be installed successfully. If



Figure 3-13.
The Requirements view.



Custom Actions

NOTE: *This feature is available only in the full edition of InstallShield Express.*

In the Custom Actions view, you can add DLL and EXE files that provide extra functionality if your setup requires it. You can add any number of custom actions to predefined points in your installation, but you must either code these executables yourself or have the distribution rights necessary to include them in your setup. You should also provide a custom action for uninstallation to undo the effects of your install-time custom action. Custom actions are optional—you need not use any to have your setup function properly.

Support Files

NOTE: *This feature is available only in the full edition of InstallShield Express.*

In the Support Files view, you can add files that you want to be available on the target system only during the installation process. These files are added to a temporary support directory and are removed from the target system after installation is complete.



To add support files to your setup project, right-click anywhere in the upper pane of the Support Files view, browse to the required file, and click Select.

Step 6: Prepare for Release

The final step in creating an effective setup is preparing your setup for release. You need to compile your setup into a working installation file, test the installation, and distribute your setup to your target media. The views in this step let you complete all three of these tasks without leaving the Express IDE.

Build Your Release

The Build Your Release view, as the name implies, allows you to compile the information you've entered into a functional setup. You can select one of a number of standard media formats (such as CD-ROM or various DVD-ROM types), compile your setup into a single disk image, or create a custom size for your disks. Once you have selected and configured your media type, you can build your setup. After you build your setup, Express provides feedback on your build, including build logs and reports. These are unique to each build and are overwritten by subsequent builds of that media type.

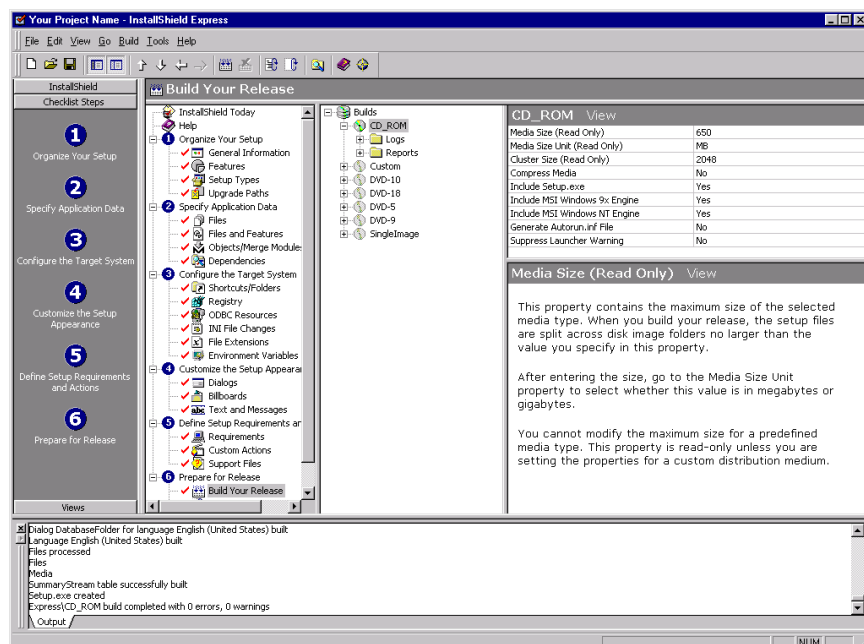


Figure 3-14.
The Build Your Release view.

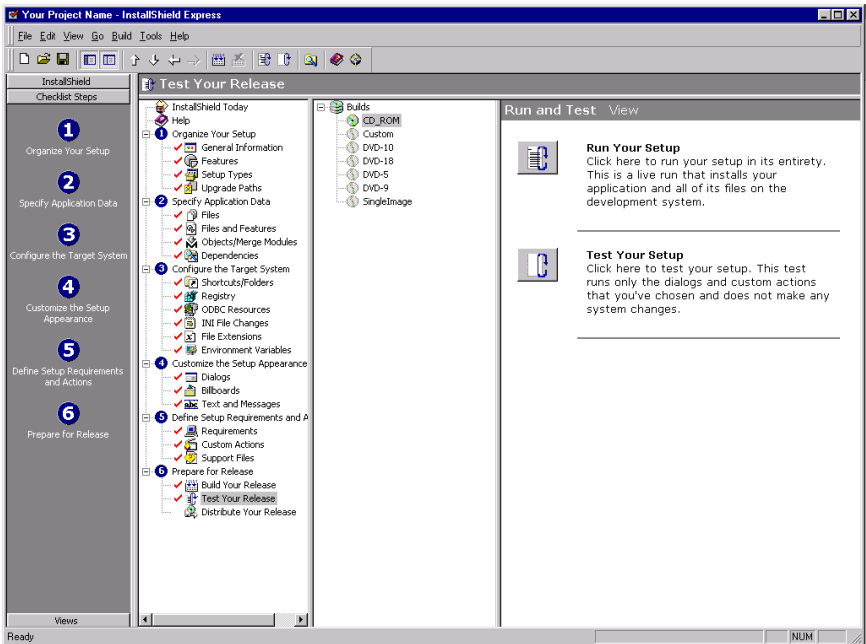
Test Your Release

In the Test Your Release view, you can test your newly built setup to make sure everything goes as planned. You have two options: Run Your Setup or Test Your Setup. If you run your setup, it performs exactly as if you had dou-



ble-clicked the setup executable. All files are installed, all dialogs appear, and all custom actions execute. If you test your setup, no changes are made to the target system; only the user interface and any custom actions you have included execute. This second option is useful if you want to test how your setup looks without taking up disk space on your system.

Figure 3-15.
The Test Your Release view.



Distribute Your Release

Once you've built your setup project and tested it to ensure that it works as you planned, you can move your release to the target media in the Distribute Your Release view. You have the option of copying it directly to the specified



media type or a staging directory, or uploading it to an FTP site, so it can be accessed from a remote location.

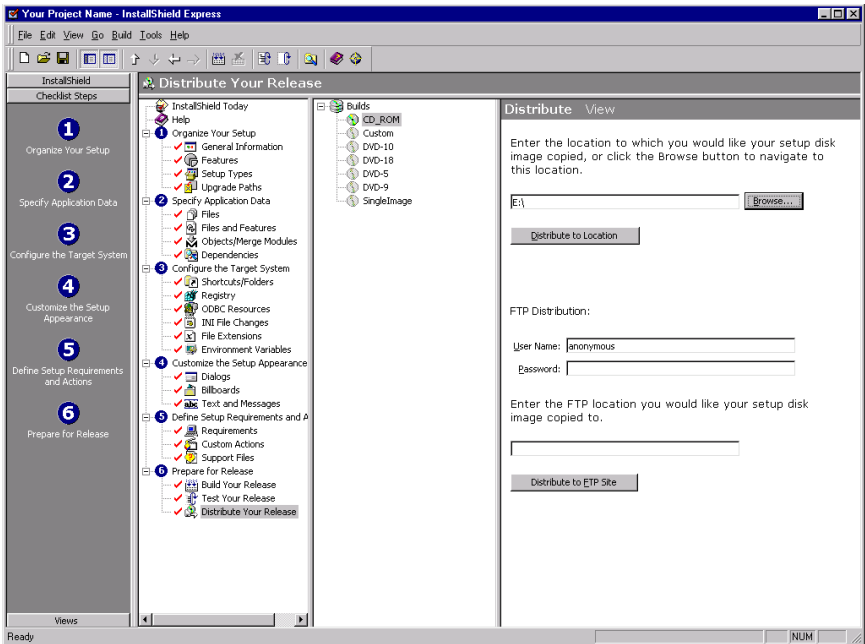


Figure 3-16.
The Distribute Your Release view.





Basic Setup Tutorial

This chapter is designed as a tutorial that you can work along with at your computer. It takes you step by step through creating a basic project using the checklist.

Table of Contents

Overview	4-3
Step 1: Organize Your Setup	4-3
Step 2: Specify Application Data	4-5
Step 3: Configure the Target System	4-6
Step 4: Customize the Setup Appearance	4-7
Step 5: Define Setup Requirements and Actions	4-8
Step 6: Prepare for Release	4-8
Summary	4-11





Overview

In this tutorial, you'll build a basic setup, using the checklist to guide you through the process. All of the files that you need are located in the \Samples\Othello\Data Files subfolder of your main Express program file folder.

To begin, launch InstallShield Express – Visual FoxPro Limited Edition and create a new project, by clicking the New button or selecting New from the File menu. The New Project dialog box appears.

1. Browse to or type the name of the project location, and name the project file “Othello.”
2. Click OK.

Step 1: Organize Your Setup

In the checklist, select General Information. This view allows you to enter information about your application, from your company's name to the Web site your customers can access in order to learn more about your application. Type “Othello” in the Subject property and “[ProgramFilesFolder]\Othello” for the INSTALLDIR property. Complete the rest of the General Information properties as shown below.

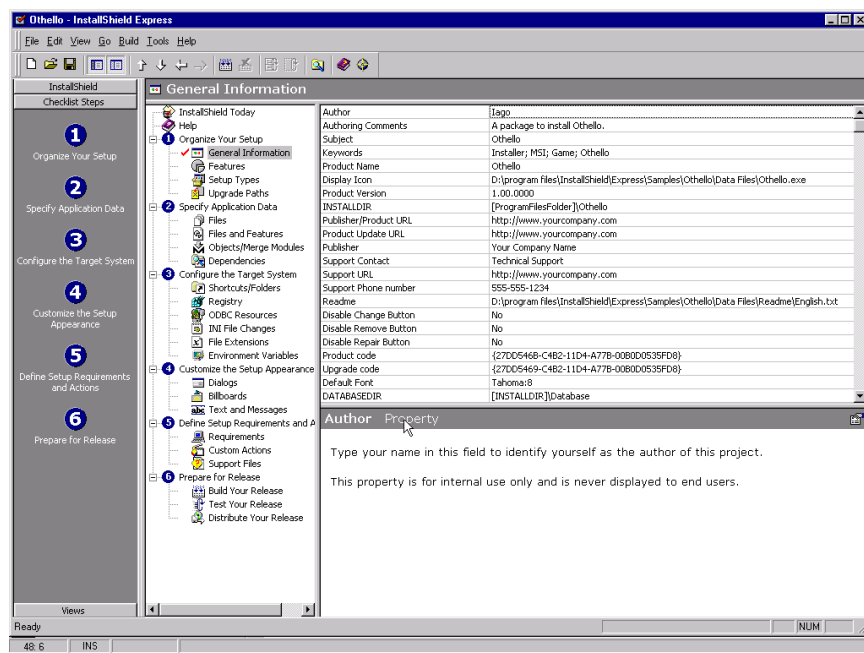


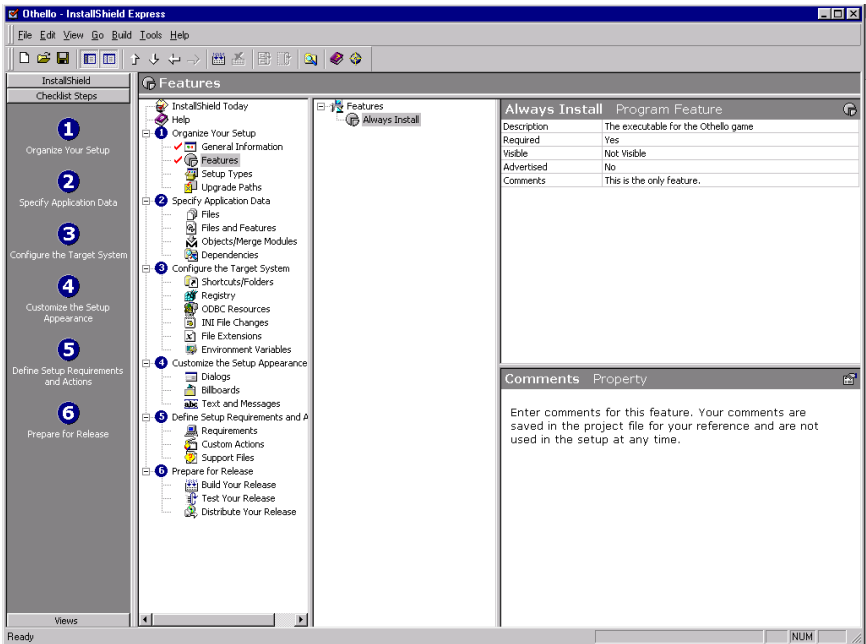
Figure 4-1.
Set the General Information view properties.

Now select Features from the checklist. This view lets you add new features to your setup, which are then used to install files and data at run time. Express automatically creates one feature for you: Always Installed. This is the only



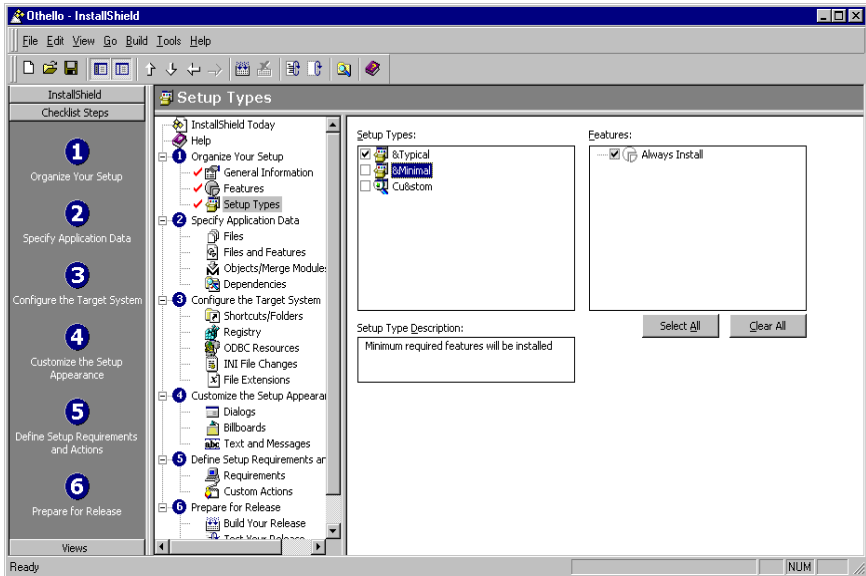
feature you need for this project. Type a description and make sure the properties are set as shown below.

Figure 4-2.
Set the Features view properties.



Finally, select Setup Types. In this view, you can configure the setup types that the end user can choose from during installation. We need only one setup type, so deselect Custom and Minimal by clicking the check boxes beside them.

Figure 4-3.
Select the Setup Types.





Step 2: Specify Application Data

Select Files from the checklist. In this view, you add the application's data files. In the top portion of this view, navigate to the \Samples\Othello\Data Files subfolder of your main Express program files folder and click that folder to open it. The files appear in the “Source computer's files” pane.

In the “Destination computer's folders” pane, right-click Destination Computer, select Show Predefined Folder to display the folder list, and select [INSTALLDIR] from the list. Drag Othello.exe and the three .gif files from the “Source computer's files” pane to the [INSTALLDIR] folder in the “Destination computer's folders” pane. Your Files view should look like this:

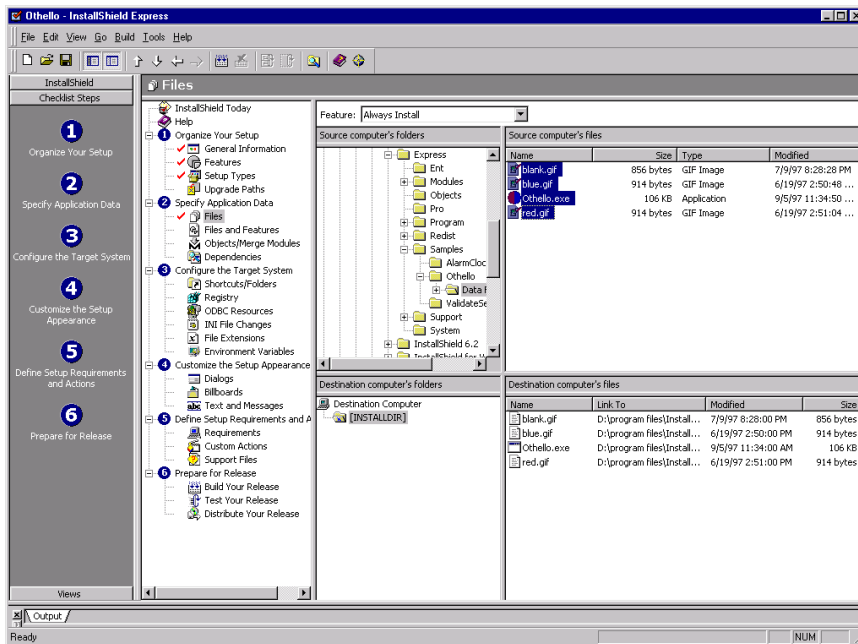


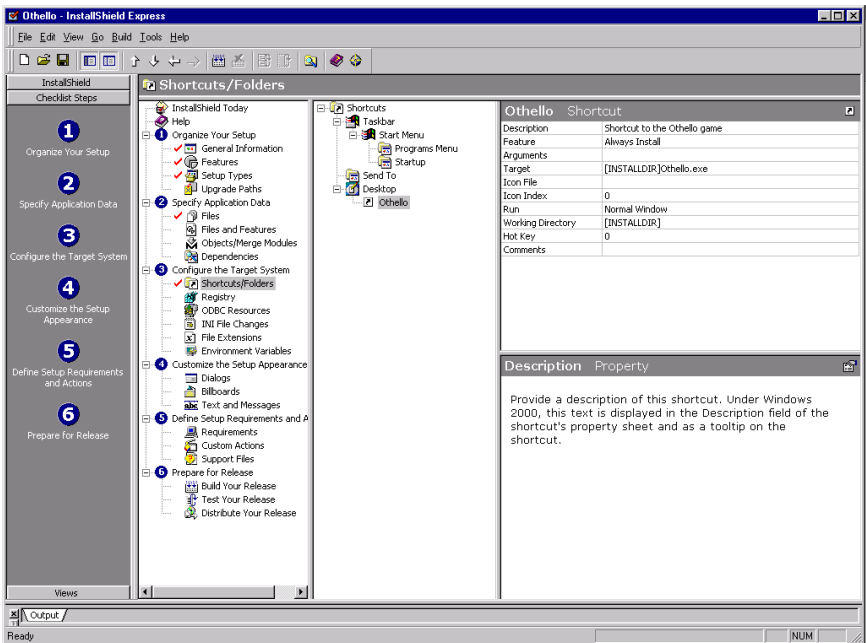
Figure 4-4.
Move files into INSTALLDIR.



Step 3: Configure the Target System

Select Shortcuts/Folders from the checklist. This view lets you create application shortcuts on the target system. Right-click on the Desktop destination in this view and select New Shortcut from the pop-up menu. Name your shortcut “Othello.” Set the properties for this shortcut so that it points to your executable file as shown below.

Figure 4-5.
Create a shortcut for your application.





Step 4: Customize the Setup Appearance

Select Dialogs from the checklist. In this view, you can select and configure the end-user dialogs that comprise the user interface for your setup. You can leave the default values for the Global Dialog Settings properties. Deselect the License Agreement dialog and select Destination Folder. For the Destination Folder dialog, set the Show Change Destination property to Yes.

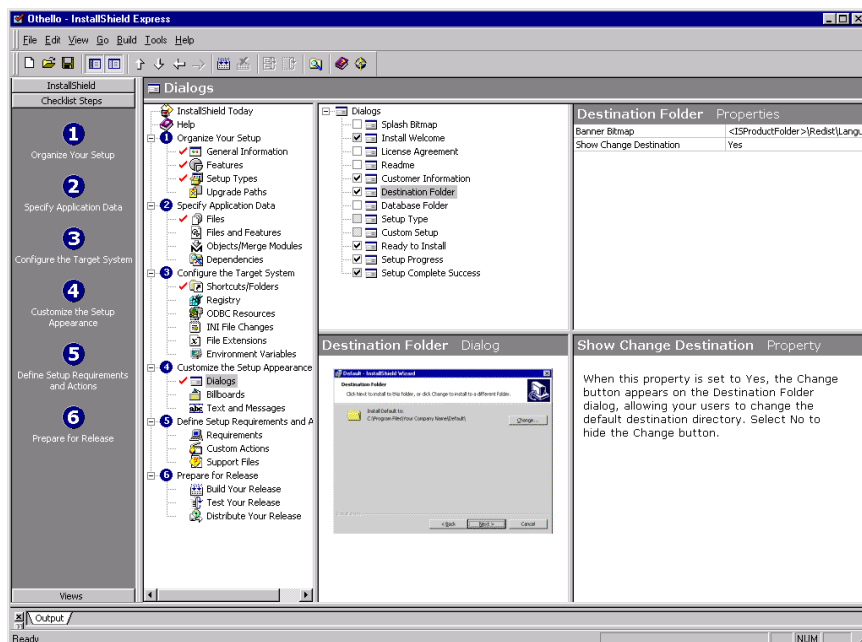


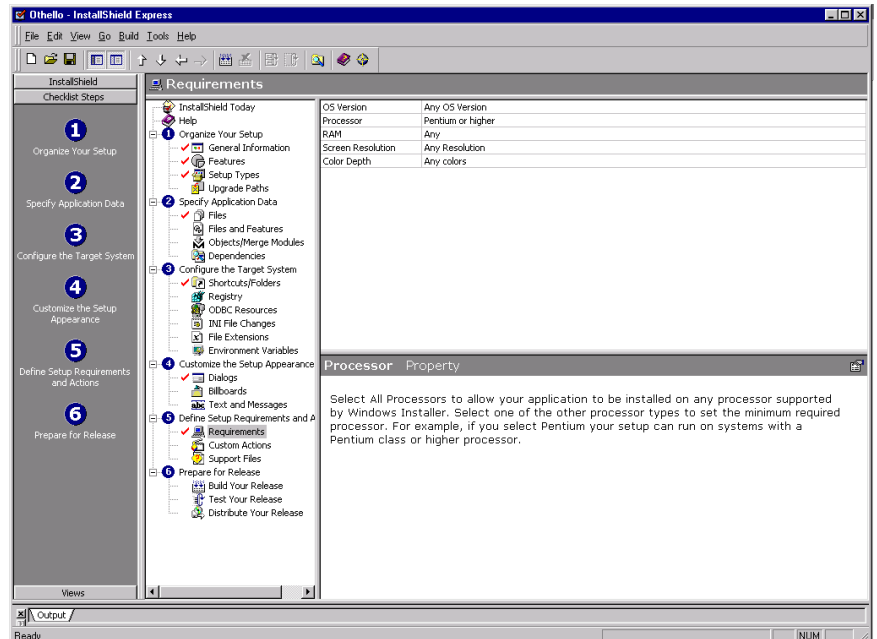
Figure 4-6.
Select and configure the end-user dialogs.



Step 5: Define Setup Requirements and Actions

Select Requirements from the checklist. In this view, you can define the target system requirements. If these requirements are not met, your application cannot be installed on the end user's system. The Othello application is fairly simple, but it does require a Pentium processor. For the Processor property, select “Pentium or higher” from the drop-down list.

Figure 4-7.
Define target system requirements.



Step 6: Prepare for Release

Select Build Your Release from the checklist. Here you compile your setup information into an .msi executable file. In this view, right-click CD_ROM in the Builds list and select Build from the context menu. As the build begins, the Output window appears at the bottom of the IDE to display the progress of



the build and any error messages. When the build process is finished, your setup files are located in a subfolder of the folder where your Othello project is located, called “Othello.”

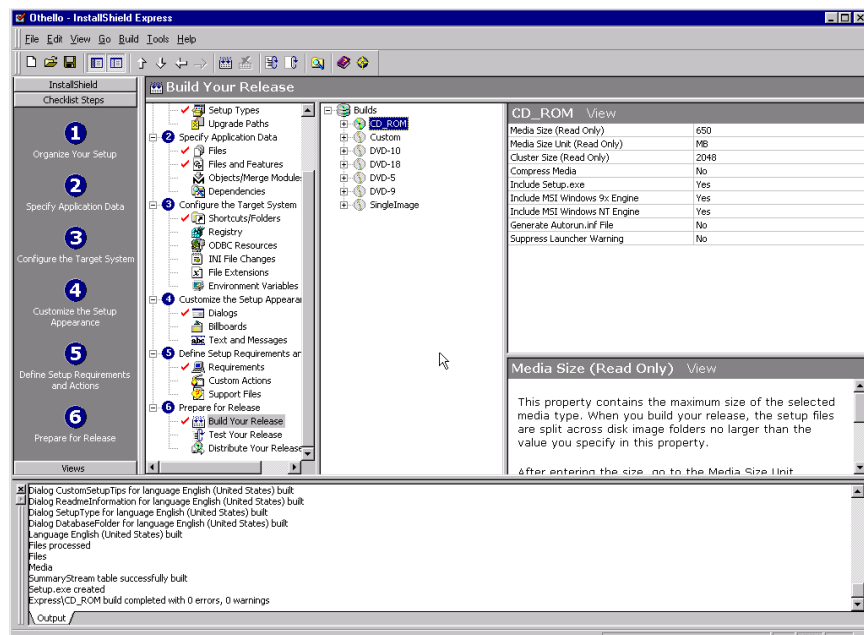


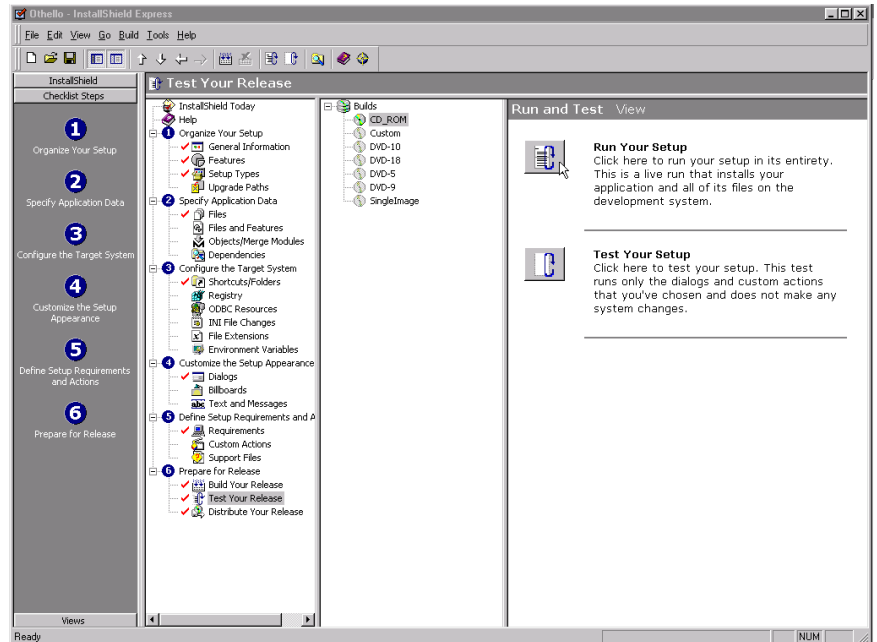
Figure 4-8.
Build your release.

Select Test Your Release from the checklist. This view lets you test your Othello setup program to ensure that it runs smoothly. The CD_ROM item has a colored icon to indicate that this is a functional build. Click the CD_ROM item to select it. Click the Run Your Setup button in the right panel of this view. Your setup launches to install the Othello program. Follow the instruc-



Figure 4-9.

Run your setup to test it.



When the setup is complete, verify that your files and data were installed correctly. Navigate to C:\Program Files\Othello (or the folder to which you chose to install the program) and verify that Othello.exe is there as it should be. Next check to see if there is a shortcut called “Othello” on your desktop. Double-click this shortcut to verify that it was configured correctly.

The final step in creating this setup is to distribute it to your target media. Since this is a tutorial and not a setup for an application that you want to distribute to your customers, this step in the tutorial is optional.

Select Distribute Your Release from the checklist. Make sure you have a CD-ROM drive with an empty CR-ROM in the drive. Select CD_ROM from the list of media. Under the text field for the location to distribute this release to, click the Browse button. Browse to your CD-ROM drive and click OK. Finally, click the Distribute to Location button and Express copies all of the



necessary setup files to this location, as long as your CD-ROM drive supports direct copying to CD-ROMs.

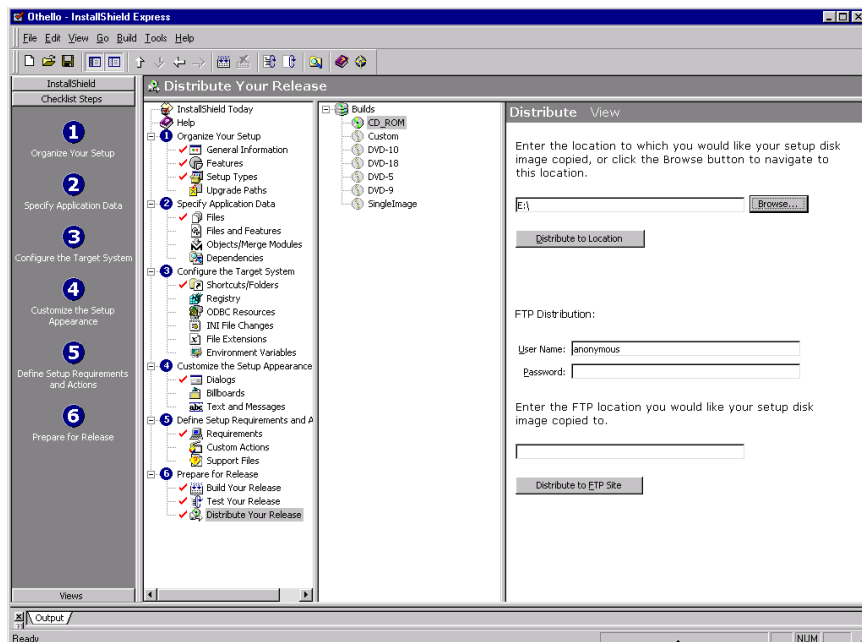


Figure 4-10.
Distribute your release.

Summary

Now that you've finished the tutorial, you can apply what you've learned here to your own setup projects. Most setup projects require the steps and concepts described in this chapter.





Advanced Features

Table of Contents

ODBC.	5-3
Merge Modules	5-3
Custom Actions	5-5
EXE Custom Actions	5-5
DLL Custom Actions	5-6

InstallShield Express – Visual FoxPro Limited Edition provides you with a number of advanced features that help you create more complex setups easily. These features greatly expand the choices you have in creating a setup by allowing you some extra abilities for installation.

The advanced features covered in this chapter include the ODBC view, the Merge Modules view, and the Custom Actions view (available only in the full edition of InstallShield Express).





ODBC

The Open Database Connectivity (ODBC) standard API lets you as a developer create applications that interface with a database using a common standard and the Structured Query Language (SQL). Many companies have developed their own ODBC drivers that you can use, such as Microsoft, Inprise, or Oracle.

You can repackage existing ODBC drivers with your InstallShield Express setup through the ODBC Resources view. This view displays all of the ODBC drivers and translators that Express has located on your system. These drivers, when installed through a setup, are automatically installed and configured as they exist on your machine.

If you want to add a driver or translator to your setup, select the checkbox beside the name of the ODBC file you want to include. All files must be associated with a feature, so once you have selected the ODBC files, select the checkbox beside one of the features that is already a part of your installation. Your ODBC files are installed and configured on the target system when that feature is installed.

If you need to change the configuration of the ODBC file you have selected, you can do so in the lower-left pane in this view. You can modify existing settings for the selected file or add new ones at the bottom of the list. You can add as many key name and value pairs as you need.

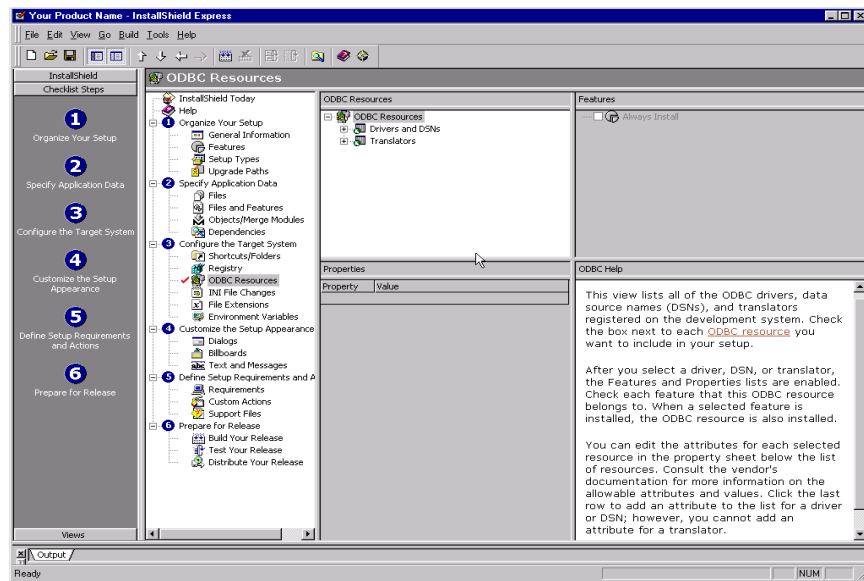


Figure 5-1.
The ODBC Resources view

Merge Modules

Merge modules (.msm) are prepackaged archives that contain all of the files and installation logic to install a distinct piece of functionality. Merge modules provide the same functionality as objects did in previous versions of InstallShield Express. If



NOTE:

Microsoft recommends that all merge modules be created by the owner of the technology. For example, Visual Basic libraries should be created and distributed by Microsoft.

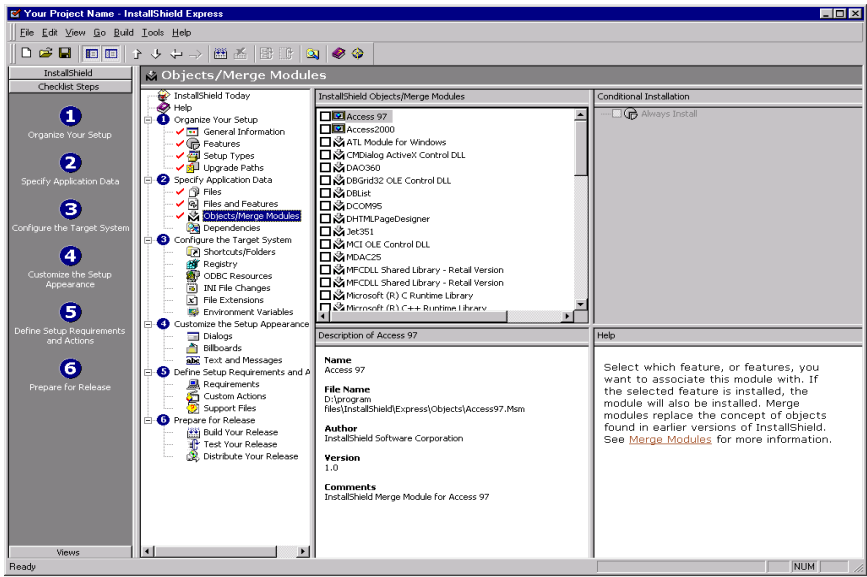
your application requires external files to operate, such as a Microsoft C++ library or ActiveX DLL, you can add that functionality to your setup quickly with a merge module.

You can add merge modules through the Objects/Merge Modules view. In this view, you can add a merge module to your setup by selecting the checkbox beside its name in the right-hand side of this view. All merge modules must be associated with a feature, so select the checkbox beside the name of the feature with which you want to install this merge module. You need not worry about the directory for the merge module; the .msm file contains all the information necessary to fully install the files contained within the merge module.

If you want to add merge modules to the list contained in this gallery, you must copy the .msm file to the [InstallShield Express Directory]\Modules\i386 folder. When you restart the InstallShield Express IDE, your new merge module appears in the Merge Module gallery. You cannot modify existing merge module files with InstallShield Express, however. It is recommended that you do not attempt to modify modules using another application, as you may break the functionality the module's author intended.

You can remove a merge module from the Merge Module gallery just as easily. Navigate to the [InstallShield Express Directory]\Modules\i386 folder. Remove the .msm file for the merge module you no longer want to include as part of the Merge Module gallery. When you restart the InstallShield Express IDE, the merge module you have removed is no longer listed in the Merge Modules view.

Figure 5-2.
The Objects/Merge Modules view





Custom Actions

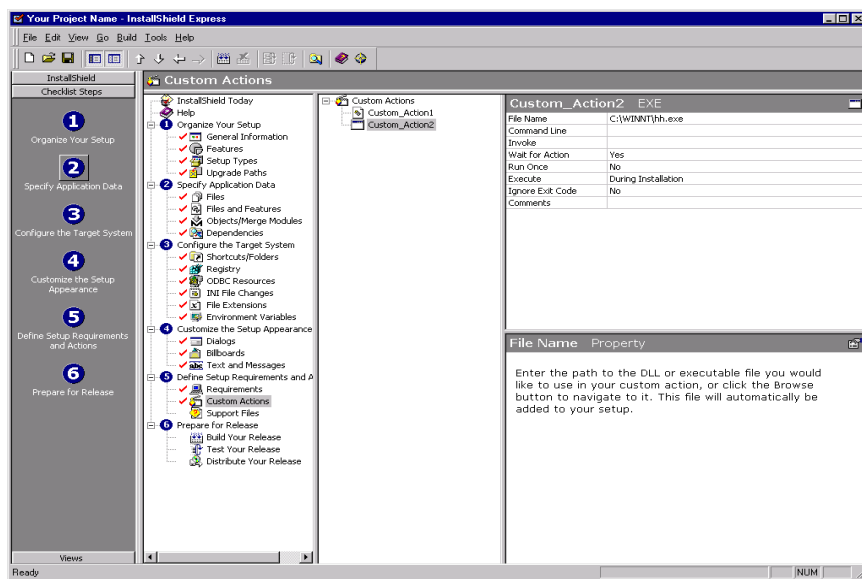
NOTE: *This feature is available only in the full edition of InstallShield Express.*

Custom actions are how InstallShield Express lets you add more functionality to your setup. These actions can occur at predefined points in the setup and can be either an EXE or a DLL. Either way, the action runs according to the value you specify in the Invoke property.

To add a custom action to your setup, you first need to have the compiled EXE or DLL file. You can add the file to your setup using the Custom Actions view. Right-click on the Custom Actions icon within the view and select either New EXE or New DLL from the context menu. Rename the custom action so its name relates to the name of the file you are using as a custom action. You then need to configure how your action should execute.

In this view, you can indicate when in the setup you want your custom action to run, whether you want it to execute during installation or uninstallation, and if you want it to run only once. You can also indicate whether or not you want the setup to ignore the custom action's return value (exit code) and continue with the installation regardless of the value.

Figure 5-3.
The Custom Actions view.



EXE Custom Actions

EXE custom actions run just like a regular application, the same as if you had double-clicked it yourself. You can add command-line arguments that you would like to pass to the executable, but they aren't necessary. You should decide whether you want your setup to wait until your executable finishes its commands and if you want this executable to run only once during a setup.



This sort of action is useful for reading documents—reading text files, playing media files, and so on, as well as launching applications packaged with your product, including your main product application.

DLL Custom Actions

DLL custom actions call a specific function in a DLL file in the same manner as you would if you wanted to call that function from an application that you were developing. You need to specify the function name but not its parameters (see below). You should decide whether you want your setup to wait until your DLL finishes its commands and if you want this function to run only once during a setup. This sort of action is useful for creating custom dialogs or performing more “behind-the-scenes” type actions.

Your DLL function must follow the prototype shown below:

```
CHAR WINAPI Foo(HWND, LPSTR, LPSTR, LPSTR, LPSTR);
```

Express uses this prototype to pass certain information to the function.

1. Parameter 1 passes the InstallShield main window handle. This parameter always returns NULL.
2. Parameter 2 passes the source directory [SRCDIR].
3. Parameter 3 passes the support directory [SUPPORTDIR].
4. Parameter 4 passes the main target directory [INSTALLDIR].
5. Parameter 5 passes the database directory [DATABASEDIR].

If you are prototyping a custom extension to validate the serial number entered in the Customer Information run-time dialog box, then Parameter 4 is the serial number.



Frequently Asked Questions

This chapter deals with some of the questions that Express users most frequently ask. If you have a question about Express that is not answered here, you should check the InstallShield Knowledge Base, available on the Support page of InstallShield's Web site. Your question may already be answered there.

Table of Contents

How Do I Specify the Destination Folder for My Application?	6-3
Using INSTALLDIR	6-3
Specifying File Destination Folders.	6-3
Hard-Coding a Destination Directory.	6-3
How Can I Be Sure the Target Machine Has the Windows Installer Service?	6-3
How Do I Get Changes I Make in the IDE to Take Effect When I Run My Setup?	6-4
How Do I Display a License Agreement?	6-4
How Do I Create Shortcuts?	6-4
How Do I Associate File Types with My Program?	6-5
How Do I Launch an External Application from My Setup?	6-5
How Do I Conditionally Install My Setup Based On the Operating System of the Target Machine?	6-5
How Do I Upgrade My Applications or Send Out Updates?	6-6
How Does Windows Installer Determine Which Files Should Be Overwritten?	6-6
How Do I Create a Self-Extracting EXE for My Installation?	6-6
Why Doesn't My Setup Fit on a Floppy Disk?	6-7
Disk Space	6-7
Disk Spanning	6-7



How Can I Create My Setup So it Installs Files
Based on the Target Operating System?. 6-7

What is the Difference Between Extracting
COM Information and Self-Registration? 6-8

Where Can I Access Updated Merge
Modules? 6-8



How Do I Specify the Destination Folder for My Application?

You can use the General Information property, `INSTALLDIR`, to specify a destination folder and the Files view to specify destination for your files on an individual basis. In addition, you can specify a hard-coded destination directory in the Files view.

Using `INSTALLDIR`

You specify the destination folder for your application through the `INSTALLDIR` property in the General Information view. This property defaults to `[ProgramFilesFolder]\Your Company Name\Default`.

Specifying File Destination Folders

You specify the destination folder for all of your files individually when you first add them to your project. To specify the destination folder of a new file, perform the following:

1. Click Files in the checklist.
2. From the drop-down list at the top, select the feature with which you would like your file to be installed.
3. Browse to your file on your system using the top half of the view.
4. Drag your file to a target folder in the lower part of this view.

Hard-Coding a Destination Directory

If you want to indicate a hard-coded destination directory for your project files, you can do this in the Files view.

1. Click on Destination Computer in the lower-left pane
2. Select Add from the context menu. A new folder appears.
3. In the New Folder name field, type the drive letter followed by a colon (for example, C:).
4. Press Enter.

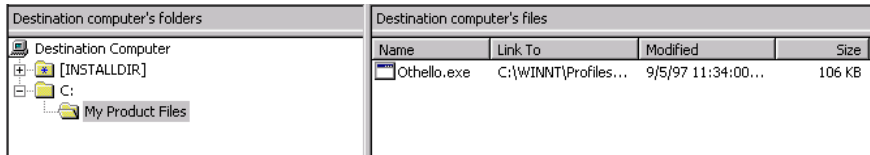


Figure 6-1.
Create a hard-coded destination directory.

How Can I Be Sure the Target Machine Has the Windows Installer Service?

Although the Windows Installer service is built into Windows 2000 and Windows Me, other Windows platforms require that the service be installed before



any Windows Installer setups can run on those systems. InstallShield Express gives you the option of including the Windows Installer service with your application setup. If the Windows Installer service is not installed on the target machine, or if an older version exists, your setup installs Windows Installer. When finished, your setup continues with your application's installation, as you have configured it.

The Build Your Release view gives you the option of including the Windows Installer service for Windows 95/98 and Windows NT 4. If you aren't sure what platforms your end users run, you might want to include both versions of the Windows Installer service with your setup package.

How Do I Get Changes I Make in the IDE to Take Effect When I Run My Setup?

For changes that you've made in the IDE to take effect in your setup package, you need to rebuild it. To build your setup at any time, click the Build button in the toolbar. This creates your setup as specified by the properties for the last media type that you have selected. If you haven't configured any properties in the Build Your Release view, then Express builds your setup using the default settings.

How Do I Display a License Agreement?

A license agreement dialog is included as part of the default user interface. All you need to do is associate a rich text file (.rtf) with this dialog. To add your text to the license agreement:

1. Click Dialogs in the checklist.
2. Select License Agreement in the Dialogs list (upper-left pane).
3. In the License File property, type or use the browse button to navigate to the .rtf file you want to use.

How Do I Create Shortcuts?

Shortcuts allow your end users to have quick access to your product. To create a shortcut with Express, perform the following steps:

1. Click Shortcuts/Folders in the checklist.
2. In the list of locations under the Shortcuts folder icon, right-click on the location in which you would like to place the shortcut.
3. Select New Shortcut from the context menu. If you need to create your shortcut in a subfolder of the selected location, select New Folder and repeat steps 2 and 3 until you create the final location for your shortcut.
4. Rename your shortcut. The name of the shortcut is highlighted initially, so you can type your shortcut's name when you create the shortcut.



5. Configure the shortcut's properties in the property grid. These properties include the icon file and index, the target executable for the shortcut (which must already be included in your setup), and the hot key combination.

How Do I Associate File Types with My Program?

If your program creates or relies on custom file types, you need to associate those file types with your program on the target machine. File types can be configured in the File Extensions view. To create a file extension, do the following:

1. Click File Extensions in the checklist.
2. Right-click on the File Extensions item.
3. Select New Extension from the context menu.
4. Specify the extension to match the file type you want to configure. For example, if you wanted to associate your program with text files, you would specify the .txt extension.
5. Configure your file association's properties. These properties include the program with which you would like to associate this extension, the icon for all files with this extension, and the feature that registers this file type.

How Do I Launch an External Application from My Setup?

With the use of custom actions, you can launch executable files or call functions from DLLs. You must first create a custom action, then insert that action into your dialog sequence. Custom actions can be added in the Custom Actions view. For more information on creating and using custom actions, see the Custom Actions section in Chapter 7.

How Do I Conditionally Install My Setup Based On the Operating System of the Target Machine?

If you want your setup to run only on certain operating systems—Windows 2000, for example—you can set this as an installation requirement. If the target system does not meet the installation requirements you specify, your application cannot be installed. To set installation requirements:

1. Click Requirements in the checklist.
2. Set the target system requirements for your application. Configure the requirements for operating system, processor speed, RAM memory amount, screen resolution, and color depth. You do not need to indicate require-

NOTE:

If your setup requires more complex conditions, either for your application as a whole or feature to feature, you may want to consider upgrading to InstallShield Professional—Windows Installer Edition.



ments for all of these properties—only for those that are pertinent to your application.

How Do I Upgrade My Applications or Send Out Updates?

NOTE: *This feature is available only in the full edition of InstallShield Express.*

If you want end users to be able to upgrade to your current application without having to uninstall any previous versions they might already have installed on their systems, you can use the Upgrade Paths view to do this. In the Upgrade Paths view, you can indicate information about the previously released versions that you want your current version to update. See the Upgrade Paths section in Chapter 3 for more information.

How Does Windows Installer Determine Which Files Should Be Overwritten?

When files are transferred during a setup, and an existing version of a file already exists on the target machine, the Windows Installer service compares the version, date, and language of the files to determine which version should remain. The following guidelines apply:

- **Versioned files** - In all cases, the file with the highest version is maintained, even if the file with the higher version is one that is already on the target machine. A file of any version is maintained over unversioned files.
- **File language** - All other things being equal, the file that is the same language as the setup is maintained over different language versions of the file. The only exception to this rule applies to multiple language files. Files with multiple languages are maintained over single-language versions of a file.
- **Date** - If the modified date of a file already present on the target machine is later than the creation date of that file, the file is not overwritten. This rule protects user preference files from being deleted during an upgrade or reinstallation.

How Do I Create a Self-Extracting EXE for My Installation?

To create a self-extracting .exe file for your installation, do the following:

1. Click Build Your Release in the checklist.
2. Click on the Single Image option in the Builds list.



3. Select Yes for the Compress Media property.

Why Doesn't My Setup Fit on a Floppy Disk?

There are two primary reasons why your setup might not fit on a floppy disk—disk space and disk spanning.

Disk Space

Although Microsoft ships Windows 2000 and Windows Millennium Edition with the Windows Installer service, all other operating systems to date do not ship with this service. Therefore, the Windows Installer service must be installed before your setup can run. There are two versions of this setup—one for Windows NT 4.0 and one for Windows 9x. Each of these setups requires more than 1.4 MB of disk space. Because of these file size requirements, it is difficult to distribute your setup program on a floppy disk if you need to distribute the Windows Installer service setups as well.

Disk Spanning

In addition the .msi file created by InstallShield that interacts with the Windows Installer service cannot be spanned across multiple disks and it must reside on the first disk of your setup. Therefore, if you want to include all of your files in one compressed .msi file, it might not fit on one floppy disk. However, if your files can remain uncompressed, they can be included on successive disks of the setup.

How Can I Create My Setup So it Installs Files Based on the Target Operating System?

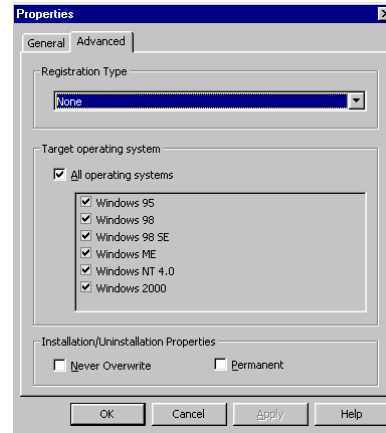
In the Files view, you can set properties for each file using the File Properties dialog. To access the dialog:

1. Click Files in the checklist.
2. Right-click on a file in the “Destination computer’s files” pane.
3. Select Properties from the context menu. The Properties dialog appears.
4. In the Properties dialog, click the Advanced tab.



5. In the Target operating system section, indicate the operating system(s) for which this file is intended. To select a specific operating system, you need to deselect “All operating systems” (the default option).

Figure 6-2.
The File Properties dialog.



What is the Difference Between Extracting COM Information and Self-Registration?

In the File Properties dialog (shown above), the Registration Type property allows you to indicate how you want your file to be registered. Depending on the file type, you can select from two or three of the following options. The Self-Registration option is not available for .exe files.

- **None** Select this option if you don't want the selected file to be registered on the target machine. This is the default setting for all files.
- **Extract COM Information** Select this option if you want InstallShield to extract all COM registration data from your file and register it on the system during setup. This is the recommended way to register COM objects.
- **Self-Registration** If your file supports self-registration you can choose this option. Note that self-registration is not as reliable as having Windows Installer register and unregister the file with extracted COM information.

Where Can I Access Updated Merge Modules?

You can download updated merge modules from the InstallShield Merge Module Gallery. Visit the gallery at <http://support.installshield.com/download/modules.asp>.

CD Browsers

Table of Contents

What is a CD Browser?	A-3
Creating a CD Browser with DemoShield.	A-3
Creating a CD Browser: The Basic Process.	A-4

This appendix explains how to create a CD browser. CD browsers can launch automatically when the CD is inserted into the CD-ROM drive in order to begin the installation process in a simple and efficient way. This appendix includes a brief tutorial on using DemoShield to create a CD browser.





What is a CD Browser?

A CD browser is a demo that provides a graphical interface for your CD-ROM files and your product setup. There are several benefits to supplying a CD browser with your application.

- Present a visually appealing interface for your setup. A self-launching CD browser helps you make the setup process more enjoyable by providing an interactive, graphical interface from which the setup can be launched. Impress your customers by using graphics, animation, video, and sound that reflect the overall image of your product.
- Showcase the other applications you develop. You may have developed several applications that would be of interest to your current customers. By providing product information within the CD browser, you can market these applications directly to people who are familiar with your company.
- Provide contact information for your company. A CD browser allows you to include a handy list of names, numbers, addresses, and Web site URLs that your customers can reference quickly. Placing your general company, sales, marketing, and technical support contact information on the CD-ROM ensures that your customers can reach you whether or not they have access to printed documentation or a connection to the Internet.

You used a CD Browser when installing InstallShield Express. To take another look at the InstallShield Express CD Browser, insert the InstallShield Express CD into your CD-ROM drive. To see other examples of CD browsers, visit the DemoShield Gallery at <http://www.installshield.com/demoshield/gallery/default.asp>.

Creating a CD Browser with DemoShield

DemoShield is an authoring tool that is particularly well suited to helping you quickly create interactive CD browsers.

No coding or scripting is required. DemoShield is completely point-and-click, providing several objects that you can place within a scene. Fill these objects with your media files, and set their properties as you like. Assign actions to interactive objects such as Hot Spot objects, Button objects, Edit Field objects, and Listbox objects by following a few simple steps in DemoShield's Build Action wizard.

DemoShield helps you with the basic layout. DemoShield's New Demo wizard provides preformatted CD browser templates. These templates contain placeholders for your graphics, text, and other objects. Fill and reformat these objects to customize your browser.

Launch applications from within the browser. Build a "Launch Application" action that allows your customer to launch your setup from within the browser. This feature is also useful when you are providing several evaluation copies of your products—just place a button in your browser, change the caption to indicate the product being installed, and build the action in a few quick steps.



You can point your customer's Web browser to any URL you wish by using the "Associate URL" feature available with many objects. Or you can give your customers control over where their Web browsers go by creating an Edit Field object that launches the URL they enter into it.

A free evaluation copy of the latest version of DemoShield is accessible from the InstallShield Express CD Browser; or you can download an evaluation copy from the InstallShield Web site, <http://www.installshield.com/demoshield>. You can also request a copy by calling InstallShield at 847-619-1550 or 800-250-2191.

Creating a CD Browser: The Basic Process

This list, adapted from the DemoShield Knowledge Base, outlines the steps involved in creating a CD browser.

1. *Determine the audience for the CD browser.*

The market for your CD-ROM affects how the information should be presented in the CD browser. If the CD-ROM is for the banking industry, marble textures and subdued colors are appropriate. If the CD-ROM is given to graphic artists, attention-grabbing multimedia elements and vibrant colors help keep the viewer interested.

2. *Determine the content included in your browser.*

The content for your browser determines the interface appropriate for delivering it. Content includes marketing copy, graphics, sounds, video, animation, and so on. Consider what you require to fulfill the purpose of the browser. Bells and whistles, such as sound effects, can always be added later if you have extra time.

3. *Design an interface that is appropriate for the content and topics.*

An interface should be suited to the content and topics that the CD browser will contain. A simple browser might have only one scene with a few buttons providing users with access to other features within the browser. If you wish to include several scenes, be sure to provide a consistent method of navigation so your customers can find their way to the information that interests them most. Remember: the interface should fit the content-not the other way around.

4. *Create a prototype using placeholders.*

A prototype of a browser takes your ideas off paper and onto the screen. A prototype will save you a lot of grief by giving others a chance to review your ideas before you go through the effort of creating the entire browser. At this point in the project, the interface and layout should match the final version. Any content not yet ready should be indicated with a placeholder. A placeholder is a temporary filler for content that is not yet ready, for example, using graphics from a clip art library or scanning in your storyboards and displaying the scanned images in place of the actual graphics and text.

5. *Put it all together.*

Once all the required tasks are complete you can put the pieces together. Replace placeholders with actual content and finish the project.

6. *Perform quality assurance testing.*



Those testing the browser should try it on a variety of configurations and systems that match as closely as possible your audience's configurations.



Internet Distribution

This appendix describes how to distribute your software using the Internet. InstallShield provides a complimentary downloadable utility, PackageForTheWeb, for this purpose.

Table of Contents

Preparing Your Setup for Internet Distribution .	B-3
Compressed Size	B-3
Self-Extracting	B-3
Digitally Signed	B-3
Easy to Use	B-3
Packaging Your Setup with PackageForTheWeb.	B-4





Preparing Your Setup for Internet Distribution

The way in which consumers receive software is quickly changing. Before the advances in Internet technology and high-speed connections, all software was shipped on some sort of removable media, such as floppy disk or CD-ROM. Today, many people are getting their software directly from the Internet. In order to take advantage of this time and money-saving software distribution process, you must package your setup in an easily downloadable and installable manner. There are several criteria that your Web-ready setup may need to meet.

Compressed Size

Although many people are now connecting to the Internet via high-speed cable modems or DSL lines, many others are still using slower 28.8 modems. Package size becomes very important to these people due to the amount of online time required to download an application.

Self-Extracting

Many file compression utilities require a special client-side application in order for the file to be uncompressed. This need for another utility makes the already confusing download and installation process that much more difficult. The compression utility you use should be self-extracting. This way, no other application is required and the installation process is simplified.

Digitally Signed

Viruses and other harmful items are easily transferred on the Internet. To make your customers feel more at ease about downloading and installing your software, you should provide a digital signature. This signature lets your customers know that your software is from a trusted, virus-free source. Visit <http://www.verisign.com> to learn more about digital signatures.

Easy to Use

Perhaps the most important aspect of packaging your setup for Internet distribution is to make it easy to use. Your customers don't want to have to specify a location where the setup files should be saved, then dig through their computer to try and find those files. Instead, the setup should be seamlessly integrated into the compression package, requiring only one step to begin the installation.



Packaging Your Setup with PackageForTheWeb

PackageForTheWeb, InstallShield's complimentary application, is ideally suited to bundle and compress your setup files for distribution across the Internet. When you've packaged a setup using PackageForTheWeb, the result is a self-extracting executable file that works on any 32-bit Windows operating system. Your customers won't need a special client-side application to uncompress your files.

In addition, PackageForTheWeb has built-in support for digital signing, helping you to ensure your customers that your software is safe to install. Most importantly, PackageForTheWeb is easy to use. By double-clicking on the compressed executable file, your setup is extracted to a temporary location and launched automatically. The temporary files are deleted when the installation is complete.

You can download your free copy of PackageForTheWeb at <http://www.installshield.com/pftw>.

Glossary

Action

The Windows Installer service works through a series of specific executions called actions. There are two types of actions: built-in actions (such as file transfer or checking for disk space) and custom actions (such as launching an EXE or calling a function in a DLL).

Advertisement

Advertisement is a type of just-in-time installation in which features are not installed immediately during setup, but only when the installer requests them. Advertisement can save space on a user's hard drive.

App Paths

The registry key that Windows uses to find your application and its DLLs if their locations are not already in the system's path. App Paths can be set through the Advanced Settings folder.

Context Menu

A context menu, also known as a right-click menu or a popup menu, appears when a user right-clicks an item on the desktop, in Windows Explorer, or in an application.

End User

The term end user, in Express documentation, refers to the customer who installs your product onto his or her computer.

Dynamic Link Library (DLL)


A shared code-base file containing functions that can be called from other applications.

Feature

Parts of a setup from an end user's perspective, such as a help file, a clip art package, or program files. These are logical groupings of files and functionality that the end user can install individually during a custom setup.

Globally Unique Identifier (GUID)

A globally unique identifier is a long string of numbers created by InstallShield for Windows Installer to uniquely identify your product from others. Enter



string GUIDs throughout the IDE in the format {XXXXXXXX -XXXX-XXXX-XXXX-XXXXXXXXXXXX}.

.ism File

The .ism file is the working file that InstallShield Express uses to store your project information. When you build a release, InstallShield Express uses the .ism file to create an .msi file for distribution.

Merge Module

A merge module is a package containing all of the logic and files needed to install distinct pieces of functionality such as run-time DLLs and virtual machines. Merge modules are built once and can be added to any setup project. For detailed information on merge modules, see the online help under Merge Modules and Merge Module Authoring.

.msi File

The .msi file, created by InstallShield for Windows Installer, is the setup package in its finished state. It includes setup resources files and can have compressed within it all of the application's data files. The .msi file is the one that is distributed to the end users and the one that interacts with the Windows Installer service to install your application.

Publishing

Publishing is a type of advertising (just-in-time installation) in which no user-interface elements are created for the component during installation, but the component can still be installed through the Add/Remove Programs applet of the Control Panel or when an installed component requests the published component from the installer.

Product

Refers to the actual application or collection of files that is to be installed.

Project

Refers to the entire collection of source files and the .ism file that make up your setup while it is under construction.



Registry or REG file

A text file of a predefined format that contains keys and values that can be merged into a registry.

Reinstallation

When a product has already been installed on a machine and its setup is run again, the setup reinstalls the product by overwriting its existing files, shortcuts, and registry entries.

Rollback

The Windows Installer service tracks all changes that are made during the installation process so that, if an error occurs and the installation is aborted, the changes are rolled back and the target machine is restored to its original state.

Run Time

The time during which the .msi file interacts with the Windows Installer service to install your application on the target machine.

String Table

A database that maintains the string IDs, values, and comments for a specific language.

Uninstallation

The process where the setup undoes the changes made during the installation. Uninstallation is the installation maintenance option that allows the end user to remove the product files and reverse any changes to the machine made during the original installation.



Index

Symbols

.bmp 3-16
.msm files 5-3

A

advertise features 1-6
Always Installed feature 4-3
associate file types 3-14

B

billboards 3-16
Billboards view 3-16
bitmap images 3-16
Build Your Release view 3-19, 4-8

C

CD browsers A-3
checklist 3-4
Conditionally Install 6-5
conventions xxii
custom actions 5-5, 6-5
Custom Actions view 3-18

D

DemoShield A-3
Dependencies view 3-10
destination folder 6-3
dialogs 3-16
Dialogs view 3-16, 4-7
disk image 3-19
Distribute Your Release view 3-20, 4-10
DLL actions 3-18, 5-6
 prototyping 5-6

E

Environment Variables view 3-15
EXE actions 3-18, 5-5

extensions
 migrating 2-4
External Application 6-5
Extract COM Information 6-8

F

features 3-5
Features view 3-5, 4-3
File Extensions view 3-14, 6-5
File Properties Dialog 6-7
file types 6-5
 associate 3-14
Files and Features view 3-8
Files view 3-8, 4-5
Floppy Disk 6-7
FTP 3-21

G

General Information view 3-5, 4-3
globalization 3-17

H

help resources -xix

I

INI File Changes view 3-13
INI files 3-13
INSTALLDIR 6-3
Internet Distribution B-3

K


Knowledge Base xxi

L

license agreement 6-4

M

merge 6-8



Merge Module gallery 5-4
Merge Modules 6-8
merge modules 3-9, 5-3
 add 5-4
migrate 2-3
msm files 5-3

N

newsgroups xxi

O

objects
 migrating 2-4
Objects/Merge Modules view 3-9, 5-4
ODBC 3-12, 5-3
ODBC Resources view 3-12, 5-3
online help -xix

P

PackageForTheWeb B-4
project file 3-3
prototype DLLs 5-6

R

REG files 3-11
Registration Type 6-8
registry entries 3-11
Registry view 3-11
Requirements view 3-17, 4-8, 6-5
rollback 1-6
run setups 3-19

S

scanning 3-10
 active 3-10
 passive 3-10
 Visual Basic projects 3-10
Self-Extracting .exe 6-6

Self-Registration 6-8
 setup 1-3
 setup types 3-6
Setup Types view 3-6, 4-4
shortcuts 3-11, 6-4
Shortcuts/Folders view 3-11, 4-6
strings 3-17
Support Files view 3-18
system requirements -xiv

T

test setups 3-19
Test Your Release view 3-19, 4-9
Text and Messages view 3-17
toolbar 3-3

U

Upgrade Paths view 3-7, 6-6
uploading an FTP site 3-21

V

viewbar 3-4
Views 3-4

W

Windows Installer 1-5
 including 6-3



