

User's Guide

VoiceAssistTM 2.1

Mindmaker, Inc.

Mindmaker, Inc.
End-User Software License Agreement

THIS IS A LEGAL AGREEMENT BETWEEN YOU AND MINDMAKER, INC. PLEASE READ THIS DOCUMENT CAREFULLY BEFORE USING THE SOFTWARE. BY USING THE SOFTWARE, YOU AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT INSTALL OR USE THE SOFTWARE AND PROMPTLY RETURN, WITHIN 15 DAYS, THE SOFTWARE AND ALL ACCOMPANYING MATERIAL TO THE PLACE OF ACQUISITION FOR A FULL REFUND.

SOFTWARE PRODUCT LICENSE

1. **Ownership.** The enclosed software programs ("Software") and the accompanying material are owned by Mindmaker, Inc. and its suppliers. The Software and the accompanying material are protected by United States and international laws and treaties regarding copyrights and other intellectual property rights. The Software is licensed, not sold, to you for use only under the terms of this Agreement. As between you and Mindmaker, Inc., Mindmaker, Inc. retains all title to and ownership of the Software and any accompanying material, and reserves all rights not expressly granted to you in this Agreement.
2. **Use on a Single Computer.** The Software may be used only on a single computer. You may transfer the machine-readable portion of the Software from one computer to another computer, provided that (a) the Software (including any portion or copy thereof) is erased from the first computer and (b) there is no possibility that the Software will be used on more than one computer at a time.
3. **Archival Copy.** You may make only one (1) archival copy of the machine-readable portion of the Software for backup purposes only, provided that you reproduce on the copy all copyright and other proprietary rights notices included on the original Software.
4. **Permanent Transfer of License.** You may transfer your license of the Software, provided that (a) you transfer all portions of the Software, (b) you do not retain any portion of the Software or any copy thereof, and (c) the transferee reads and agrees to be bound by the terms and conditions of this Agreement.
5. **Prohibitions on Modifying and Renting of the Software.** Except to the extent expressly permitted by the laws of the jurisdiction where you acquired the Software, you may not modify, create derivative works of, adapt, translate, rent, lease, loan or sublicense the Software or its accompanying material.
6. **Decompiling, Disassembling or Reverse Engineering.** You acknowledge that the Software contains trade secrets and other proprietary information of Mindmaker, Inc. and its suppliers. Except to the extent expressly permitted by the laws of the jurisdiction where you are located, you may not decompile, disassemble or otherwise reverse engineer the Software, or engage in any other activities to obtain underlying information that is not visible to the user in connection with normal use of the Software. In any event, you will notify Mindmaker, Inc. of any information derived from decompiling, disassembling, reverse engineering or such other activities and the results thereof will constitute the confidential information of Mindmaker, Inc. You shall have no right to make available to any third party such confidential information or use such confidential information except in connection with the normal use of the Software.
7. **Termination.** The license granted to you is effective until terminated. You may terminate it at any time by returning the Software (including any portions or copies thereof) to Mindmaker, Inc. The license will also terminate automatically without any notice from Mindmaker, Inc. if you fail to comply with any term or condition of this Agreement. You agree upon such termination to return the Software (including any portions or copies thereof) to Mindmaker, Inc. Upon termination, Mindmaker, Inc. may also enforce any rights provided by law.
8. **Export Control.** You will not, without prior authorization of the U.S. Department of Commerce or any other governmental entity which has jurisdiction, export or transmit the Software to any country to which such export or transmission is restricted by any applicable U.S. regulation or law.
9. **U.S. Government Restricted Rights.** The Software and accompanying material are provided with restricted rights. Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software-Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Mindmaker, Inc., 224 Airport Parkway, Suite 550, San Jose, California 95110.

MISCELLANEOUS

This Agreement shall be governed and interpreted in accordance with the laws of the State of California, U.S., as applied to contracts entered into and performed entirely in California. This Agreement shall be construed in such fashion as to make each provision enforceable to the maximum extent possible under governing law. This Agreement contains the entire agreement between the parties with respect to the subject matter hereof and supersedes all prior agreements or understandings (oral or written).

Failure or delay in enforcing any right or provision of this Agreement shall not be deemed a waiver of such provision or right with respect to any subsequent breach. If any provision of this Agreement shall be held by a court of competent jurisdiction to be contrary to law, that provision shall be enforced to the maximum extent permissible, and the remaining provisions of this Agreement will remain in full force and effect.

MICROSOFT DCOM95

All title, including but not limited to copyrights, in and to Microsoft's DCOM95 and any copies thereof are owned by Microsoft or its suppliers.

LIMITED WARRANTY

Mindmaker, Inc. warrants that the Software will perform substantially in accordance with its accompanying documentation for a period of ninety (90) days from the date of purchase.

EXCEPT AS SET FORTH IN THE PRECEDING SENTENCE, MINDMAKER, INC. AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. NEITHER MINDMAKER, INC. NOR ITS SUPPLIERS WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET YOUR REQUIREMENTS OR THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE. YOU AGREE TO ASSUME FULL RESPONSIBILITY FOR THE SELECTION OF THE SOFTWARE TO ACHIEVE YOUR INTENDED RESULTS, AND FOR THE INSTALLATION, USE AND RESULTS OBTAINED FROM THE SOFTWARE. YOU ALSO ASSUME THE ENTIRE RISK OF ANY USE OF THE SOFTWARE. NO DISTRIBUTOR, DEALER OR ANY OTHER ENTITY OR PERSON IS AUTHORIZED TO EXPAND OR ALTER THIS WARRANTY.

THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM COUNTRY/STATE TO COUNTRY/STATE. SOME COUNTRIES/STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED CONDITIONS AND WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. MINDMAKER, INC. DISCLAIMS ALL CONDITIONS AND WARRANTIES OF ANY KIND IF THE SOFTWARE HAS BEEN CUSTOMIZED, REPACKAGED OR ALTERED IN ANY WAY BY YOU OR ANY OTHER PARTY.

EXCLUSIVE REMEDY AND LIMITATION OF LIABILITY

THE SOLE AND EXCLUSIVE REMEDY FOR ANY BREACH OF THE LIMITED WARRANTY SET FORTH ABOVE WILL BE, AT MINDMAKER, INC.'S OPTION, (a) RETURN OF THE PURCHASE PRICE OR (b) REPLACEMENT OF THE DEFECTIVE SOFTWARE. IN NO EVENT WILL MINDMAKER, INC. OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES OR FOR ANY LOST PROFITS, LOST SAVINGS, LOST REVENUES OR LOST DATA ARISING FROM OR RELATING TO THE SOFTWARE, ITS USE OR INABILITY OF USE. IN NO EVENT WILL MINDMAKER, INC.'S OR ANY OF ITS SUPPLIERS' LIABILITY OR DAMAGES TO YOU OR ANY OTHER PARTY EVER EXCEED THE AMOUNT PAID BY YOU TO USE THE SOFTWARE, REGARDLESS OF THE FORM OF THE CLAIM (WHETHER IN CONTRACT, TORT OR OTHERWISE). SOME COUNTRIES/STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU.

ALL USE OF THE SOFTWARE IS SUBJECT TO THE TERMS OF THE END-USER SOFTWARE LICENSE AGREEMENT SET FORTH IN THE INSTALLATION SETUP, AND ON THE PREVIOUS PAGE OF THIS USER'S GUIDE.

Copyright 1999 Mindmaker, Inc.
All rights reserved.

Information in this document is subject to change without notice and does not represent a commitment on the part of Mindmaker, Inc.

All Mindmaker, Inc. product names are trademarks or registered trademarks of Mindmaker, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders.

Mindmaker, Inc.
224 Airport Parkway, Suite 550
San Jose, CA 95110
United States
www.mindmaker.com

Mindmaker Ltd.
Konkoly-Thege M. út 29-33, 18. ép.
H-1121 Budapest
Hungary
www.mindmaker.hu

Contents

Preface.....	ix
What's in This Manual?	ix
Document Conventions	ix
Related Documentation	x
Register and Get Free Updates	x
Technical Support	x
 Chapter 1: Introducing VoiceAssist.....	1
What is VoiceAssist?	1
What Can I Do with VoiceAssist?	2
What Are VoiceAssist's Key Features?	2
 Chapter 2: Getting Started with VoiceAssist	3
System Requirements	3
Before You Install	3
Installing and Uninstalling	4
Starting and Quitting	4
Checking Your Audio Setup	6
 Chapter 3: Quick Tour.....	9
Basic Features	9
Advanced Features	12
Getting the Most out of VoiceAssist	14
 Chapter 4: Controlling Applications	15
Understanding the Toolbar	15
Types of Commands	16
Command Methods	16
Availability of Commands	18
Feedback for Commands	20
 Chapter 5: Training Command Inputs	21
Overview of the Training Process	21
The Training Dialog Box	23
Basic Training Procedures	25
Training for Voice Input	25
Testing for Voice Input	31
Training for Keyboard Input	32
Testing for Keyboard Input	34
Confirmation Before Executing	34
Creating New Commands	35
 Chapter 6: Training Actions	37

Training an Action	37
The Action Dialog Box	38
Advanced Training with MiniScript	41
Chapter 7: User Settings.....	49
Understanding User and Application Templates	49
The Users Dialog Box	49
Managing User Templates	50
Managing Application Templates	51
Appendix A: Troubleshooting	55
Appendix B: Glossary	61
Index	65

Preface

Welcome to VoiceAssist[®], the powerful command-and-control agent from Mindmaker. VoiceAssist enables you to interact with your computer in an exciting new way. You can use voice input, keyboard input, or double-clicking commands to carry out a variety of useful tasks, including starting programs, executing program commands, and performing complex macros.

What's in This Manual?

This manual is designed to provide a thorough explanation of every major feature of VoiceAssist. For a quick introduction to the basic features and functions of the program, read “Chapter 3: Quick Tour.”

This manual contains:

- “Chapter 1: Introducing VoiceAssist”: An overview of the program's features.
- “Chapter 2: Getting Started with VoiceAssist”: Instructions and information on installation, and system requirements.
- “Chapter 3: Quick Tour”: Step-by-step instructions for performing common tasks.
- “Chapter 4: Controlling Applications”: An overview of the command methods supported by VoiceAssist.
- “Chapter 5: Training Command Inputs”: How to train voice and keyboard input for commands.
- “Chapter 6: Training Actions”: How to train actions associated with the commands.
- “Chapter 7: User Settings”: How to create and manage files containing user-specific training data.
- “Appendix A: Troubleshooting”: Tips and suggestions for solving and answering some common issues and questions.
- “Appendix B: Glossary”: A collection of definitions and explanations of some of the terms used in this guide.

Document Conventions

The following typographical conventions are used in this guide:

Bold text

Indicates the name of a dialog box, command button, script command, or file name.

EXAMPLE: Click on the **Close** button in the **Users** dialog box.

Italicized text

Indicates new terminology or a document title.

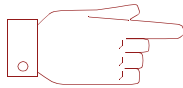
EXAMPLE: Your training information is stored in your *user template*.

“text inside quotation”

Indicates an input command name used.

EXAMPLE: Issue the command “Start Notepad”.

The pointing hand



Introduces important information.

EXAMPLE: The application will not function unless a user template is selected.

Related Documentation

Besides this *User's Guide*, you can find additional information about VoiceAssist in the following documents:

- *Mini Manual*
- *Online Help*

Register and Get Free Updates

The setup program will ask if you want to register your product online, or you may send in your registration card via postal mail. It is recommended that you register, because it is free, and you will get access to free updates and qualify for special pricing on related products.

If you did not register upon setup, you can do so at any time either by clicking on the **Registration...** button in the **About VoiceAssist** dialog box, or by visiting our web site:

<http://www.mindmaker.com>

Technical Support

If you need help with VoiceAssist, first consult this *User's Guide* and the online Help system. If you continue to have problems then you can contact our technical support staff. As a registered user, our technical support staff is available to you via the Internet on our web site:

<http://www.mindmaker.com>

Chapter 1: Introducing VoiceAssist

This chapter provides a quick overview of the purposes and features of the software. It explains what VoiceAssist is, why it is useful, and introduces its key features.

What is VoiceAssist?

VoiceAssist is a command-and-control agent that allows you to navigate the Windows® environment and run applications by using voice, keyboard commands, or by double-clicking commands in the **Commands List** window. For example, you could say “e-mail” to launch your e-mail software, type “L” to insert a letterhead into a Microsoft Word document, or double-click the “Minimize” command in the **Commands List** window to minimize the active window on your desktop. Since VoiceAssist automatically trains itself to recognize your commands, you can perform an almost limitless number of tasks with minimal effort.

VoiceAssist acts as a type of intelligent assistant that runs quietly in the background, monitoring your input. Whenever you open a program or window, VoiceAssist automatically extracts the available menu commands. When you open your Web browser, for example, VoiceAssist automatically loads all the commands available for that program. You can then execute a command by simply saying its name, e.g., “home”, into the microphone, typing its name into the **Keyboard Input** box, or double-clicking the command in the **Commands List** window, and VoiceAssist will execute the appropriate action.

We have taken special care to make VoiceAssist as easy to use as possible. Consequently, you do not have to train it to recognize your commands. If you wish to retrain the program to recognize a different input, it is easy to do so. You can also quickly train VoiceAssist to execute personally customized actions.

While VoiceAssist can be used as an intelligent command agent, it can also be integrated by software developers into their applications. This manual describes VoiceAssist for end users. For information about the VoiceAssist Software Development Kit please contact Mindmaker. (see the LIMITED WARRANTY).

What Can I Do with VoiceAssist?

Using VoiceAssist, you can:

- Use voice commands when you want hands-free control of your applications; use keyboard commands or double-click commands in the **Commands List** window when you do not want to speak out loud or when speaking out loud might be disturbing to those around you.
- Navigate the Windows environment and the Web.
- Open and close any program such as Calculator or Notepad.
- Execute program menu commands as well as complex, fully-customized actions.

What Are VoiceAssist's Key Features?

Intelligent Assistant Functionality

Make VoiceAssist your assistant. You can train your assistant to navigate the Web for you, start a letter, and open your most frequently used programs.

Multiple Input Options

You have three ways of issuing commands—via voice, keyboard input, or double-clicking commands in the **Commands List** window.

Multiple Users

Any number of people may share the same computer and each use VoiceAssist in his or her own way. With VoiceAssist's user template feature, you can store, import, and export files that contain your own training data for specific applications.

Pretrained Actions and Commands

Use VoiceAssist for a new application instantly. VoiceAssist automatically extracts menu commands, buttons and other items from a dialog box from any Windows-based program and immediately knows how to perform the associated actions. Just launch the application, and the commands are available.

Speech Recognition

Communicate with VoiceAssist through state-of-the-art speech technology without additional training.

Scripting Capability with MiniScript

Describe any sequence of actions, commands, and macros in an easy scripting language.

Chapter 2: Getting Started with VoiceAssist

This chapter guides you through the process of installing VoiceAssist on the Windows platform. It contains information on the minimum hardware and software requirements, as well as instructions for checking your audio setup before you install. Finally, it gives instructions for starting the VoiceAssist application.

System Requirements

- Microsoft® Windows 95, Windows 98, or Windows NT 4.0.
- PC with a 90 MHz Pentium®-class processor for Windows 95/98, or a PC with a 120 MHz Pentium-class processor for Windows NT 4.0.
- At least 16 MB of RAM for Windows 95/98 and 32 MB of RAM for Windows NT 4.0.
- 10 MB of free hard drive space.
- Windows95/98/NT 4.0-compatible, full-duplex multimedia audio hardware. Creative audio hardware is recommended.
- A unidirectional mid- to high-end microphone.

Requirements for Advanced Features

- **Microsoft Internet Explorer 4.0** (or higher) required to allow web navigation by voice.

Before You Install

Determining What Software Package You Have

If you purchased VoiceAssist as part of the Prody Parrot 2.0™ or Zymo the Alien 2.0™ software package, it must be installed as part of that package to operate these Assistants. In fact, when you install one of these Assistant 2.0 packages, VoiceAssist is automatically installed as one of its components. Please refer to the *Prody Parrot 2.0 User's Guide* or *Zymo the Alien 2.0 User's Guide* for installation and uninstallation instructions.

If you purchased VoiceAssist as a stand-alone product, you may install it directly, following the instructions in this chapter.

Installing and Uninstalling

First you need to determine what software package you have. See “Determining What Software Package You Have” on page 3.

Installing VoiceAssist Stand-alone

To install VoiceAssist:

1. Have the product serial number ready, which is provided on the Registration Card.
2. Shutdown all other applications before proceeding with the installation.
3. Insert the CD into your CD-ROM drive. If the setup program does not launch within 30 seconds, click **Start** on the Windows taskbar, then **Run**, and enter “**d:\install\setup.exe**” (substitute **d:** with the drive letter of your CD-ROM drive). Click **OK**.
4. Follow the instructions of the setup program.

Removal Instructions

To uninstall VoiceAssist from the Start Menu:

1. Click **Start** on the Windows taskbar, then select **Programs**, then **VoiceAssist**.
2. Now click **VoiceAssist Uninstall**.



If you have Prody Parrot 2.0 or Zymo the Alien 2.0 installed, you will be able to uninstall VoiceAssist, however the 2.0 Assistants will no longer work because they both require VoiceAssist 2.1.

Registering

At the end of the setup process, you will be prompted to register. See “Register and Get Free Updates” on page x.

Starting and Quitting

Starting VoiceAssist

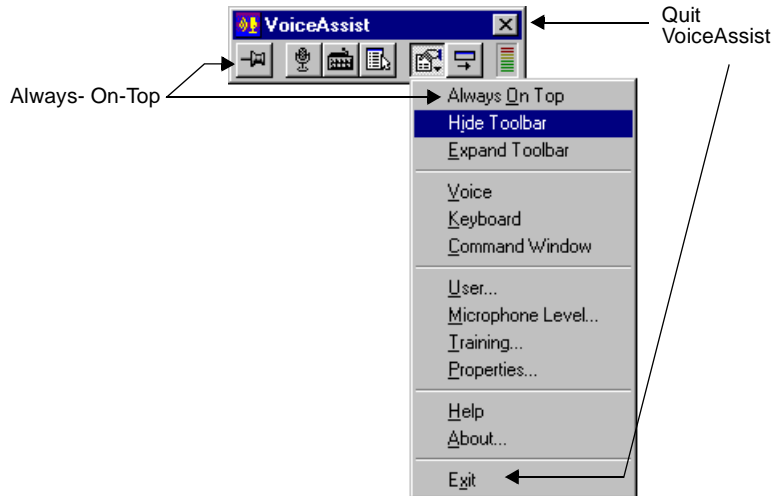
To start VoiceAssist:

1. Click **Start** on the Windows taskbar and select **Programs**.
2. From the **Programs** menu, select **Assistant**, then **VoiceAssist**, and then click **VoiceAssist**.

When you start VoiceAssist, a splash screen will be briefly displayed, and then the VoiceAssist toolbar will appear.

Displaying and Hiding the Toolbar

The VoiceAssist toolbar lets you access all the features and functions of VoiceAssist. You will learn more on how to set and adjust these features and functions in later chapters. For now, note the ways in which you can keep the toolbar on top of all other windows on your desktop, and quit the program.





If you deactivate the **Always-On-Top** option, the VoiceAssist toolbar may be covered by other windows on your desktop. If the toolbar is covered by another window, and you would like to bring it into view, you can do so by either clicking the **VoiceAssist** taskbar button on the Windows taskbar or by double-clicking the **VoiceAssist** taskbar icon in the Windows tasktray.



Quitting VoiceAssist

To quit VoiceAssist from the toolbar:

- Click the  in the upper right corner of the toolbar.
- or
- Click the **Main Menu** button  on the toolbar and then **Exit**.

To quit VoiceAssist from the tasktray:

- Right-click the **VoiceAssist** tasktray icon on the Windows tasktray and then left-click **Exit**.

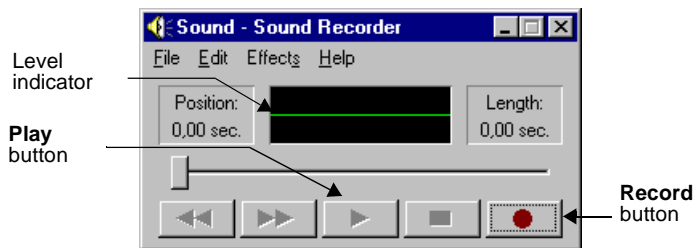
Checking Your Audio Setup

In order for VoiceAssist to work properly on your computer, you must have the correct audio setup. If you are not certain that your audio setup is correct, then complete the following steps. This should take only a couple of minutes.

To check audio recording and playback:

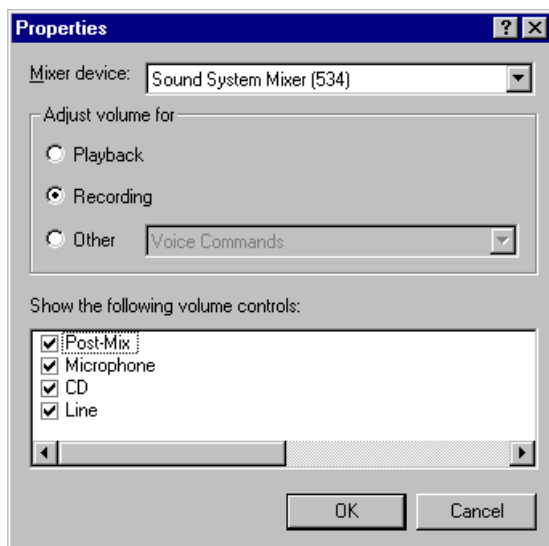
To check your audio setup, make sure that you can successfully record and playback your own voice using the Windows **Sound Recorder**. For example, in Windows 95 you would do the following steps:

1. Click **Start** on the Windows taskbar, then **Programs**, **Accessories**, **Multimedia**, and **Sound Recorder**.

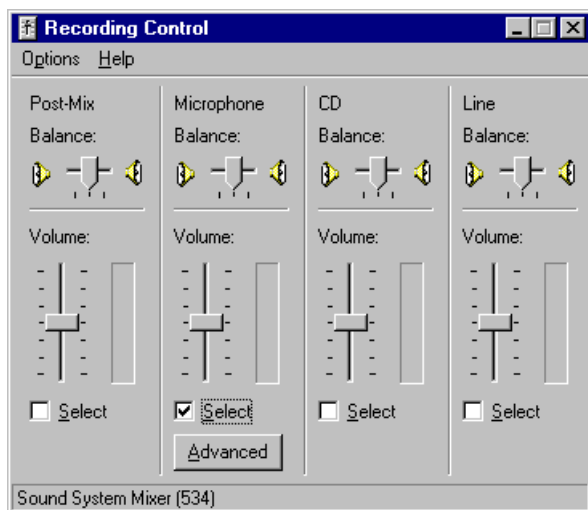


In the **Sound Recorder** dialog box, click the **Record** button and then say a few words into your microphone. You should see the green line in the level indicator box fluctuate as you speak. If it does not, make sure that your microphone is plugged into the correct jack and that your mouth is close enough to the microphone. If the green line still does not fluctuate when you speak, go on to step 2. If the green line does fluctuate when you speak, then your microphone is ready to communicate with VoiceAssist.

2. If you had trouble getting your microphone input to display anything on the level indicator in the previous step, then you will need to verify that the **Mixer** properties for your microphone are correct. Click the Windows **Start** menu button, then **Programs**, **Accessories**, **Multimedia**, and **Volume Control**. In the **Options** menu, click **Properties**. The **Properties** dialog box will appear.



3. In the **Adjust volume for** group, select **Recording**. Then, in the **Show the following volume controls** list, make sure that the **Microphone** check box is selected. Click **OK**. The **Recording Control** dialog box will appear.



4. Adjust the **Volume** slider for the **Microphone** and check the **Select** box below it. Clear the **Select** check boxes under all of the other items. When you have finished, click the **X** in the upper right corner of the dialog box to close it. Check your microphone input again (as

described in step 1). If the green line fluctuates in the level indicator box, then you are ready to use VoiceAssist.



In Windows 98 you will follow a similar procedure, but consult your Windows 98 user's guide for details.

For more information on either the **Volume Control** or the **Sound Recorder** application, click the **Help** button in the relevant dialog box.

Chapter 3: Quick Tour

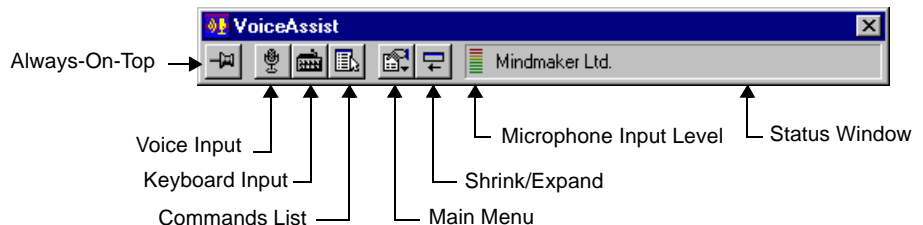
This chapter helps you learn how to use VoiceAssist in a short period of time. As you take the tour, you can learn about the basic and advanced features of the software.

In the “Basic Features” section, you can learn how to use the toolbar and how to issue commands by voice and keyboard input, as well as by double-clicking commands. In the “Advanced Features” section, you can learn how to use VoiceAssist to execute personally customized complex actions.

Basic Features

Understanding the Toolbar


Before beginning, please familiarize yourself with the VoiceAssist toolbar and its buttons:

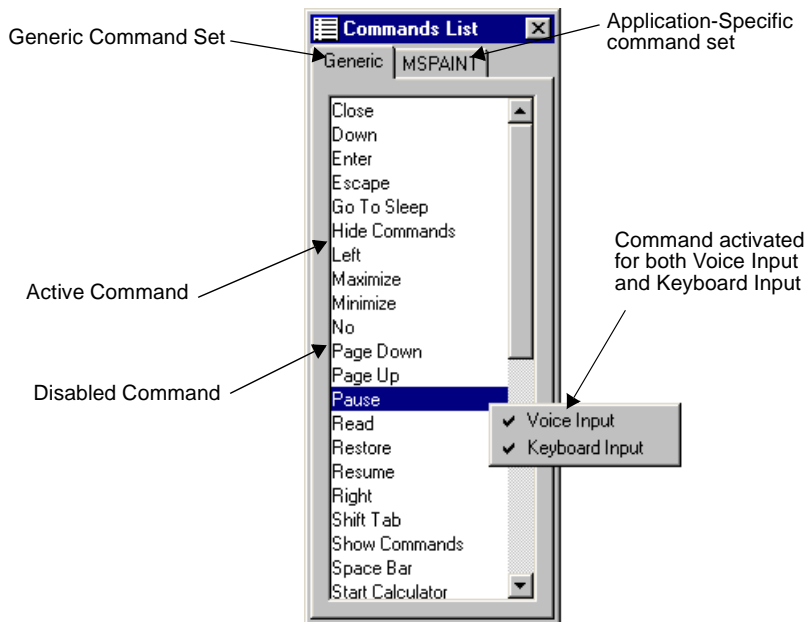


You can communicate with VoiceAssist through voice input, keyboard input, and by double-clicking commands in the **Commands List** window. Just click the appropriate button on the toolbar to make sure VoiceAssist is prepared for your type of input.

Displaying the Available Commands

VoiceAssist comes with a set of predefined commands, which are displayed in the **Commands List** window. Although you don’t have to display the **Commands List** window to issue commands, you may find it useful.

To display the **Commands List** window, click the **Commands List** button  on the VoiceAssist toolbar. The **Commands List** window is now displayed. You will see the **Generic** tab page displayed in the window:



The *generic commands* allow you to perform common Windows actions such as opening and closing a program, scrolling up and down a document, etc.

If you have an application open, you will also see a set of *application-specific commands*. These are commands that VoiceAssist automatically extracts from the active application and makes them available to you. You can also add your own commands for specific applications (see “Chapter 5: Training Command Inputs”).





Some of the commands in the **Commands List** window may not appear in the normal black font. This signifies that the inputs for these commands have been disabled by the user. You can see which inputs are active for a particular command by right-clicking on the command name. See “Chapter 5: Training Command Inputs” to re-enable these inputs.

Issuing Commands


Even without training, you can start issuing commands to VoiceAssist. Conveniently, commands are automatically self-trained, meaning you merely have to say, type, or double-click the command names in order to execute them.

To issue commands by voice:

1. If it is not already depressed, click the **Voice Input** button  on the VoiceAssist toolbar to activate it.
2. If you have not already done so, connect your microphone to the computer and test the audio input (see “Chapter 2: Getting Started with VoiceAssist” if you need help).
3. For example, say “Start Notepad” into the microphone. Once VoiceAssist recognizes your command, it will open Notepad. Say “Close” to quit the program.

Say any of the generic commands listed in the **Commands List** window. For example, say “*Start Internet Explorer*” or “*Start Media Player*”. In order to display this window, click on the **Commands List** button .


To issue commands by keyboard:

1. If it is not already depressed, click the **Keyboard Input** button  on the toolbar to activate it.
2. Press SHIFT+ESC to display the **Keyboard Input** dialog box.
3. For example, type “*Start Windows Explorer*”, and then click **OK**. VoiceAssist will open Windows Explorer. Repeat the process for any other command you wish.

Notice that VoiceAssist completes the command before you finish typing it. This is designed to save you time. Simply press ENTER when you see your command appear in the **Keyboard Input** dialog box.

To issue commands by double-clicking commands:

Double-clicking in the **Commands List** window is a quick way to execute commands.

1. If it is not already depressed, click the **Commands List** button  on the toolbar to display the window.
2. For example, double-click “Start Calculator” in the **Generic** tab list. The Calculator application will then be launched. You can also close Calculator by double-clicking on the “Close” command in the same **Generic** tab list.

Advanced Features

This section shows you even more ways you can use VoiceAssist as your intelligent assistant. You will learn how to train VoiceAssist to recognize new commands, and how to create new actions for these.

Creating New Commands

The first step when creating a new command is to define the command name. Follow the steps below to define a new command name. In the next section, you will learn how to train the actions for the new command.

To create a new command:

1. Open the **Training** dialog box by selecting the **Training...** item in the **Main Menu**.
2. If you want this command to be used in a particular application, then open that application.
3. In the **Training** dialog box, click anywhere in the generic command set (or click in the application-specific command set for a command to be used for a particular application).
4. Click the **Add...** button, and the **Command Name** dialog box will appear.
5. Enter a name for the command you want to add in the **Enter new name** text box. Click **OK**.

You've now created a command, but you're not quite done yet since VoiceAssist does not yet know what to *do* with these commands. We will now train VoiceAssist to perform the appropriate actions.

Command Inputs

For your convenience, VoiceAssist trains itself as soon as you create a new command. It extracts the command name that you defined when creating the command, and uses this for voice, keyboard, and double-clicking input. Simply say, type, or double-click the new command name.

Of course, you can always retrain these input types to recognize different inputs. See "Chapter 5: Training Command Inputs" for more information.

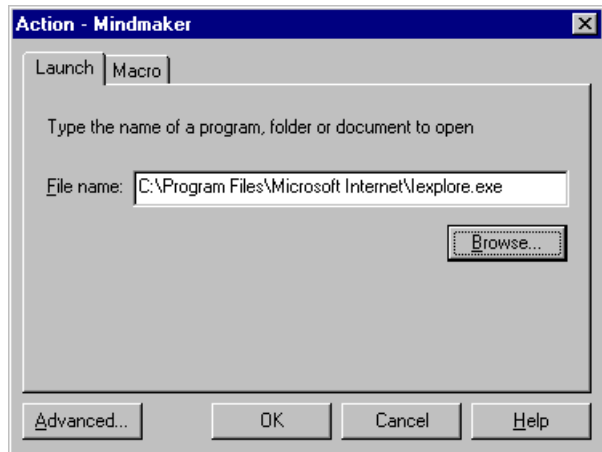
Creating a New Action

If you would like VoiceAssist to execute an action, you have to train it with the steps you'd like it to execute.

To create an action for checking the date:

1. Open the **Training** dialog box, if it is not already open.

2. Select the command for which you want to train the action. You can do this by clicking on the name of the command in the commands list in the **Training** dialog box. Remember that the corresponding application must be open to select a command from an application-specific command list.
3. Click the **Action...** button. The **Action** dialog box will now appear.



4. Here you can specify what action your command will execute. You can choose between two types of actions: **Launch** and **Macro**. Click the **Launch** tab if you want to train your action to start an application. Click the **Macro** tab if you want to train a sequence of mouse and/or keyboard actions.
5. Click the **Macro** tab, and let us record a mouse movement. Make sure the **Mouse** check box and **Screen** radio button are selected.
6. Choose something productive to do with your mouse, for instance double-clicking on the clock in your taskbar to check the date. When you are ready, click the **Record** button, then double-click on the clock in your taskbar.
7. When you have finished double-clicking on the clock, press the PAUSE key on the upper right-hand corner of your keyboard. You have just trained an action macro.
8. Now click **OK**.

For more information on creating customized actions, see "Chapter 6: Training Actions."

Your newly designed action will now be performed when the associated command is issued.

Getting the Most out of VoiceAssist

As you've seen in this Quick Tour, VoiceAssist's basic features can be used in many ways to help you as you work on your computer. Without any additional training, you can use VoiceAssist to control your Windows applications by voice, keyboard, or double-clicking. With just a little imagination and some simple training, you can also extend VoiceAssist's basic skills and create your very own customized personal assistant.

The following are some ideas of ways in which you can take advantage of VoiceAssist's features. Some of them may strike you as very useful, and others may not apply to you. Still others may help spark new ideas of your own for using VoiceAssist to its full potential. Instructions for implementing the ideas can be found in "Chapter 6: Training Actions".

- Create a new command called "Search the Web" and train VoiceAssist to launch your favorite search site when you issue the command.
- Look over the list of generic commands for starting programs, for example "Start Calculator", "Start CD Player", etc., and identify any that open programs you seldom use. Then, rename and retrain those commands to open programs that you frequently use. For example, if you never use the Solitaire program but you very frequently use Photoshop, rename the "Start Solitaire" command "Start Photoshop" and retrain it to open Photoshop.
- Train VoiceAssist to open frequently used files that are deeply embedded in your file directory. For example, if you often use a document that is located in C:\MyDocuments\Letters\Business\Orders\Acknowledgments\Research.doc, you can create a new command called "Open Research" and train VoiceAssist to open the file when you issue that simple command.
- Create a new command called "Show me the news" and train VoiceAssist to launch your favorite news site when you issue the command.
- Create a new command called "Show me the weather" and train VoiceAssist to launch your favorite weather site when you issue the command.

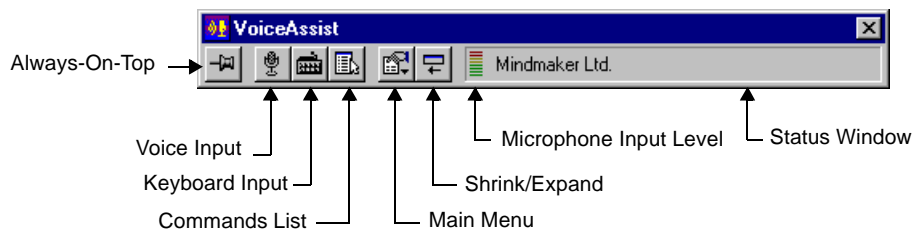
You've now completed the Quick Tour of VoiceAssist, which briefly covered the majority of the skills you need to know in order to turn VoiceAssist into your personalized command-and-control assistant. For more detailed information about the topics covered here, as well as information about features not mentioned here, please refer to the appropriate sections in this *User's Guide*.

Chapter 4: Controlling Applications

This chapter describes the various ways in which you can use VoiceAssist to control applications in your Windows environment. The different types of commands, the different options for issuing these commands and receiving feedback, as well as the availability of certain command sets are all explained in this chapter.

Understanding the Toolbar


The diagram below shows all available buttons and features of the VoiceAssist toolbar:



Adjusting the Toolbar

There are different modes which you can view and access the features on toolbar, viewing or hiding the status window, and always keeping it on top of your desktop.


Shrink or expand


You can either shrink or expand the VoiceAssist toolbar depending on whether or not you want to view the **Status Window**. The last button on the right of the toolbar is the **Shrink/Expand** button .

If you wish to hide the **Status Window**, then click on the **Shrink/Expand** button .

Hide or show


If you don't want to see the VoiceAssist toolbar, yet still use the program, you can choose to hide the toolbar.

1. Click the **Main Menu** button  on the toolbar.
2. Select **Hide Toolbar** from this menu, and the VoiceAssist toolbar will disappear.

You can view your hidden toolbar by simply double-clicking on the VoiceAssist tasktray icon .

Always on top

You may choose to have the VoiceAssist toolbar to be always showing on your desktop, even when there is another window currently active. To enable this feature, either depress the button on the far left side of the

toolbar, or click the **Main Menu** button  and select **Always On Top** so a check mark appears.

Types of Commands

VoiceAssist organizes commands into three different categories: generic, application-specific, and HTML.

Generic Command Set

The *generic command set* is a list of the most frequently used Windows commands. It is present at all times and can be used in any Windows application. Generic commands include common commands such as “Close”, “Minimize”, etc.

Some of the generic commands open commonly used applications such as Calculator and WordPad. Although these applications have been predefined, you are free to change them. For example, you can rename and retrain the “Start Solitaire” command to open Microsoft Word.

Application-Specific Command Set

The *application-specific command set* is, as its name suggests, a collection of commands intended for a specific application. It consists of all the active menu commands in the corresponding application. For example, the application-specific command set for Microsoft Excel contains commands that work within Excel but may have no use in another program. The command “chart” is an example of a command which exists in Excel, but not necessarily in other applications. Application-specific command sets can also include additional commands that you create yourself.

As a diligent assistant, VoiceAssist automatically loads the application-specific command set for the application that is currently active on your Windows desktop, or will create it on the fly as needed. Consequently, you do not need to load the application-specific command set for the program you are using—VoiceAssist automatically does it for you.

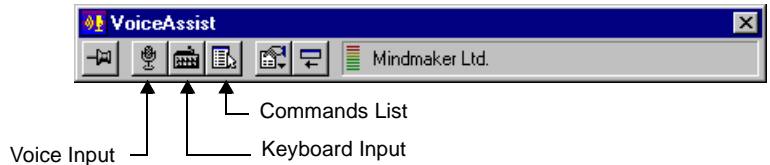
HTML Command Set

The *HTML command set* is a collection of commands which allow you to access HTML links. VoiceAssist automatically extracts the HTML links from the active window. This means that you can navigate the Web by simply issuing the name of the HTML link.

Command Methods

VoiceAssist responds to input from three sources: voice, keyboard, and double-clicking commands.


Each input source corresponds to a button on the toolbar.




In order for VoiceAssist to respond to input from any of the three sources, the corresponding button must be depressed on the VoiceAssist toolbar.

Commanding by Voice

To issue a command by voice:

1. Make sure your microphone is properly connected to the computer.
2. Make sure that the **Voice Input** button  is depressed on the VoiceAssist toolbar.
3. Say the name of any command into the microphone. The **Status Window** on the toolbar displays the name of the recognized command.



If you want to see a list of the currently available commands, then click on the **Commands List** button .



If VoiceAssist does not recognize your command or mistakes it for another command, try saying it again as clearly as possible. If VoiceAssist still does not recognize it, see “Appendix A: Troubleshooting.”

Commanding by Keyboard


To issue a command by keyboard:

1. Make sure that the **Keyboard Input** button  on the VoiceAssist toolbar is depressed.
2. Press SHIFT+ESC. The **Keyboard Input** dialog box will display.
3. Type an enabled command. (If you want to see a list of the currently available commands, then click on the **Commands List** button .) The words will appear in the text box even before you finish typing them.
4. Click **OK** or press ENTER when you are finished typing, or when you see your desired command in the **Keyboard Input** window. The command will be executed, and the **Status Window** on the

VoiceAssist toolbar will display the name of the command that was recognized.

Commanding by Double-Clicking

To issue a command by double-clicking:

1. To display the **Commands List** window, press the **Commands List** button  on the VoiceAssist toolbar.
2. Double-click on any command in the **Commands List** window, and VoiceAssist will perform the action associated with it.

Availability of Commands

Sleep Mode

In order to use your computer's system resources as efficiently as possible, VoiceAssist can be told to go to sleep when you know you will not be issuing any commands for a while, but you don't want to quit the program. Simply issue the command "Go To Sleep" to activate sleep mode. To bring VoiceAssist out of sleep mode, issue the command "Wake Up".




VoiceAssist will not respond to input of any kind if it is in sleep mode, with the exception of the command "Wake Up". To bring the program out of sleep mode and reactivate input for normal commands, issue the command "Wake Up".

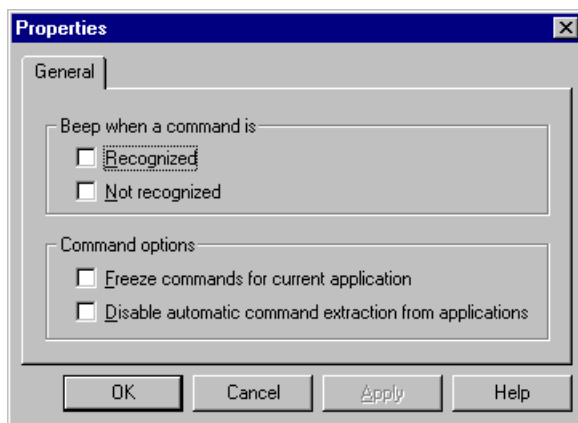
Extracting and Loading Application-Specific Commands

As explained earlier, the generic command set is always available while VoiceAssist is running, unless it is in sleep mode. Furthermore, VoiceAssist automatically loads the application-specific command set for the application that is currently active on your Windows desktop, or creates it on the fly as needed. In most situations, you will probably find this automatic extraction or loading of application-specific commands very helpful and convenient.

However, you may want to keep the application-specific commands for one particular application loaded, regardless of what other applications are active. For example, you may be working on a document in Microsoft Word, but want to control your CD Player without having to leave Microsoft Word. In this case, you can "freeze" the application command set of the CD player, so the CD player commands are available even when you are working on your document. This way you can tell your CD Player to skip to the next track while you are composing the next Nobel Prize winning piece of literature, or writing a love letter.

To freeze commands for the current application:

1. Make sure the application which contains the commands you want to freeze is the active one on your desktop.
2. Click the **Main Menu** button  on the VoiceAssist toolbar, and select **Properties....**



3. In the **Properties** dialog box, select the **Freeze commands for current application** check box, then click **Apply** and **OK**.

Disabling command extraction

There may also be a time when you would like to temporarily, but completely disable the automatic extraction or loading of application-specific commands. Notice that you can do this by selecting the **Disable automatic command extraction from applications** check box in the **Properties** dialog box.



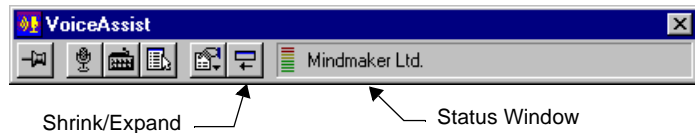
When you launch an application while **Disable automatic command extraction from applications** is active, VoiceAssist saves the commands it extracts. So next time you load the same application, VoiceAssist does not have to waste time extracting the commands. This means that the commands for a previously extracted application will be available every time you launch the application, even when **Disable automatic command extraction from applications** option is disabled.

Feedback for Commands

VoiceAssist provides both visual and audible responses to your commands. On the most basic level, it will tell you the name of the command it recognized, or tell you that your command was not recognized. This is particularly useful when you want to monitor how VoiceAssist recognizes your commands. This can be observed in the **Status Window**, where both the name of the active application and the recognized command are automatically displayed.

Visual Feedback

To view this feedback, expand the VoiceAssist toolbar, if it is not already expanded, by clicking the **Shrink/Expand** button .




If you are running any applications, the name of the currently active application will appear in the **Status Window**. When you issue a command, the recognized command name, or a “Not recognized” message, will appear in the **Status Window**.

Audible Feedback

VoiceAssist can be set to make a beep sound when it recognizes or does not recognize a command.

To select audible feedback options:

1. Click the **Main Menu** button  on the VoiceAssist toolbar, and select **Properties....** The **Properties** dialog box will display.
2. In the **Beep when a command is** group box, you can specify when you would like VoiceAssist to tell you how your command was processed. Select the **Recognized** check box if you would like to hear a beep each time VoiceAssist recognizes a command, and select the **Not recognized** check box if you would like to hear a beep each time VoiceAssist does not recognize a command. Leave both check boxes clear if you do not want audible feedback at all.

Chapter 5: Training Command Inputs

This chapter introduces you to the input training process and explains the training dialog box. Furthermore, it provides detailed instructions for training commands for voice or keyboard input, and also describes the process of creating a new command.

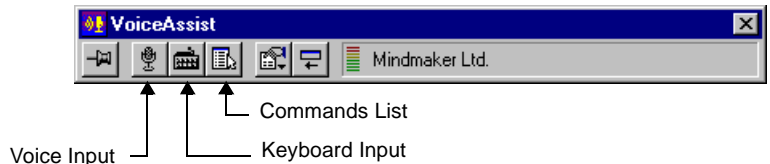
Overview of the Training Process

Before you start the training process you should familiarize yourself with the following elements of training:

- Input Options
- Command Sets
- Command Components
- Automatic Training

Input Options

VoiceAssist responds to input from three sources: voice, keyboard, and double-clicking commands. Each input source corresponds to a button on the VoiceAssist toolbar:



The corresponding button must be depressed in order to issue a command of the type of input.

Command Sets

VoiceAssist organizes commands into three different categories: generic, specific, and HTML.

Generic command set

The generic command set is a list of the most frequently used Windows commands. It is present at all times and can be used in any Windows application. Generic commands include common commands such as "Close", "Page Up", "Down", etc.

Some of the generic commands open commonly used applications such as Calculator and WordPad. Although these applications have been predefined, you are free to change them. For example, you can rename and retrain the "Start Solitaire" command to open Microsoft Word.

Application-specific command set

The application-specific *command set* is, as its name suggests, a collection of commands intended for a specific application. It consists of all the active menu commands, buttons and other items on a dialog box from the corresponding application. For example, the application-specific command set for Microsoft Excel contains commands that work within Excel but may have no use in another program. The command “chart” is an example of a command which exists in Excel, but not necessarily in other applications. Application-specific command sets can also include additional commands that you create yourself.

As a diligent assistant, VoiceAssist automatically loads the application-specific command set for the application that is currently active on your Windows desktop, or will create it on the fly as needed. Consequently, you do not need to load the application-specific command set for the program you are using—VoiceAssist automatically does it for you.

HTML command set

The *HTML command set* is a collection of commands which allow you to access HTML links. VoiceAssist automatically extracts the HTML links from the active window. This means that you can navigate the Web by simply issuing the name of the HTML link.

In order to train VoiceAssist correctly, you should be familiar with the three parts that make up each command:

- The command name
- The input
- The action

The command name

The *command name* is the title given to a command in order to describe what it does, and to distinguish it from other commands. You typically issue commands using the command name, but of course you can always rename a command or use an input that differs from the command name. For example, you can replace the voice input for the “minimize” command to “hide.”

The input

The *input* is what you enter when you train or issue a command. There are three types of input: voice, keyboard, and double-clicking commands. You can even have multiple inputs for the same input type. For example, you can train VoiceAssist to close a window when you say either “close” or “exit.”

The action


The *action* is what VoiceAssist does when it recognizes a command you issue. It can start and quit applications, open and close files, or execute more complex actions like inserting the signature at the end of an e-mail message.

Automatic Training

To make things easy, VoiceAssist automatically defines command names, actions, and even inputs whenever possible. For example, VoiceAssist automatically assigns a name for all commands and pre-trains the voice and keyboard input. That means you can use VoiceAssist without doing any training whatsoever.

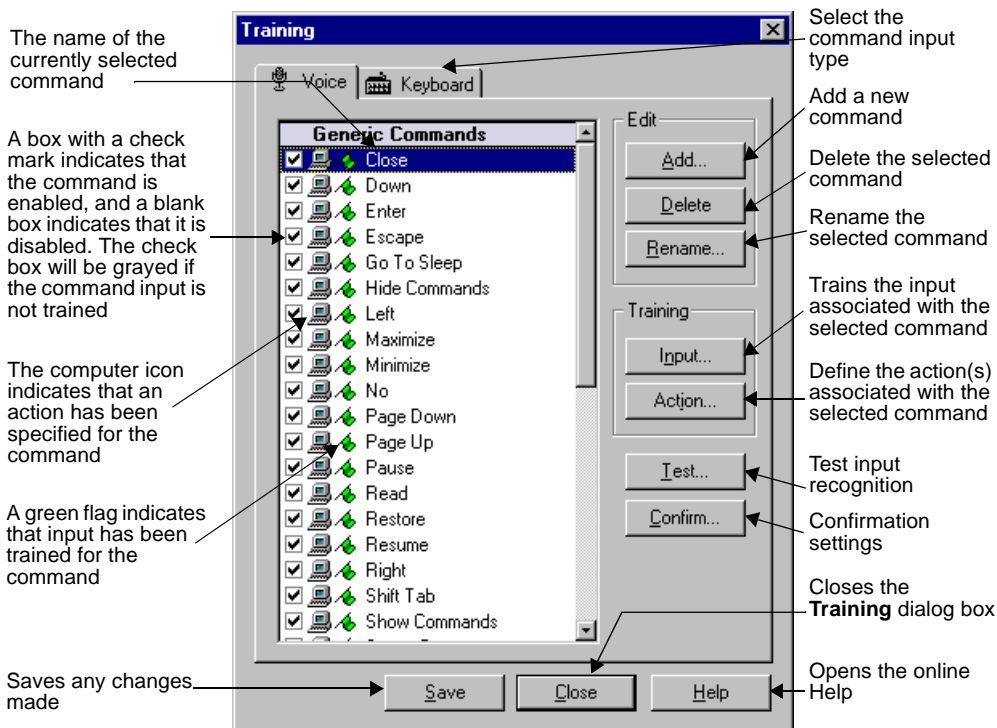
The Training Dialog Box

You use the **Training** dialog box to train and customize VoiceAssist. Before beginning, familiarize yourself with the training environment.

To open the Training dialog box, click the Main Menu button  on the VoiceAssist toolbar and click Training.... The Training dialog box will appear.

Understanding the Training Dialog Box

The **Training** dialog box allows you to select the commands you want to train, as well as create and edit new commands and actions. The **Training** dialog box diagram follows:



Disabling Commands

You may want to disable some of the commands you never use, especially if VoiceAssist mistakes another command for a command you never use.

To disable a command:

1. In the **Training** dialog box, click the input tab for which you want to disable the command.
2. Now clear the check box on the left-side of the command name.

You can re-enable commands by simply re-checking the check box on the left-side of the command name.

Basic Training Procedures

VoiceAssist is designed to make training as easy as possible. The basic training procedure is done in two steps: training the input, and training the action. This chapter covers the first step. The basic procedure to train the command input is the same for each input type. For an explanation on the second step, training the action, please read “Chapter 6: Training Actions.”

Retraining Existing Command Inputs

To retrain command inputs:

1. In the **Training** dialog box, click the tab corresponding to the appropriate input: **Voice** or **Keyboard**.
2. Select the command you want to retrain. For example, if you want to retrain the input for the “Minimize” command, then you would select “Minimize” in the list of commands by clicking on the command name. Now click **Input....**
3. Train the desired input command and click **OK** when you are satisfied.

For detailed information on how to train the different inputs, see the individual sections for training voice and keyboard.

Before you start training, familiarize yourself with the **Apply to all similar commands** option which appears in all input trainings.

Apply to all similar commands

By checking this box, you are allowing this command to be trained for all applicable programs. If you only want this command to operate in one specific application, do not check this option.

Training for Voice Input

Since voice input is always automatically trained for commands, whatever “training” you do for voice input is actually retraining. Such retraining is required only where there is some difficulty with or limitation in the automatically trained voice command.

Conveniently, the input for every voice command is automatically trained to be identical with its name. If you want to use a different voice input for a command, you will need to retrain VoiceAssist to recognize your desired input.



If the reason you want to retrain a command is poor recognition accuracy, first consult “Appendix A: Troubleshooting.” Then, if you still experience poor recognition accuracy, you should try renaming the command to a different word or phrase before retraining it. Renaming a voice command solves most speech recognition difficulties in the best and easiest way.

For training voice input, you have two options. You can either train the voice input by actually saying the command into the microphone, or you can type the text you want VoiceAssist to recognize.

Before you start training a command by voice, please make sure you follow these guidelines:

- Train in a quiet environment.
- Speak in a natural, conversational, and relaxed tone.
- Avoid prolonged pauses between the words of a command phrase.
- Try to use the same vocal tone and speed during training that you will use when you execute the command.
- If using a hand-held microphone, position the microphone between 5 and 11 inches from your mouth.
- Use a unidirectional or high-quality hands-free headset microphone for optimum recognition results.
- Refer to the section “Checking Your Audio Setup” in “Chapter 2: Getting Started with VoiceAssist” for instructions on how to check your audio setup.

Now that you know the recipe for easy and successful voice training, you can move on to retraining commands for voice input.


Retraining a Command for Voice Input

The **User Training** tab page of the **Voice Training** dialog box allows you to train your input by using your personal voice. Before you start training your input by voice, familiarize yourself with the **Add/New** radio button options.

Add/New option buttons

The **Add/New** radio buttons, which appear in the **Mode** group box, let you add additional, similar input methods, or completely replace the original command input. If **Add** is selected, then you will be adding a new input command to the existing list. If you select **New**, then you will be replacing the old list with a brand new one.

Training the voice input command by speaking:

1. Click the **Main Menu** button  on the VoiceAssist toolbar, and then click **Training....** The **Training** dialog box will appear.
2. In the **Training** dialog box, click the **Voice** tab page.
3. Select the command you want to train, for example “Close”.
4. Click on the **Input...** button, and the **Voice Training** dialog box will display:



5. Select the **User Training** tab, where you can train the input by speaking the command three times. The counter in the **Trainings** group box indicates how many times you need to repeat the command to complete the input training.
6. When you are ready to start recording your new input command, click **Start**.
7. Say the name of the command displayed in the **Command name** window, which in this case is “Close”. (Of course, you are not obligated to say the name of the command name. You may pick a new input word or phrase for this command. We simply suggest the command name.) As you say the name, you should see green bars indicating the level of your voice in the **Recording** box above the **Start** button. If VoiceAssist does not understand your command, then the number in the **Trainings** box will not change. If your voice input is longer than 7 seconds, the program will indicate that you should choose a shorter input. If VoiceAssist accepts the command input, the number will change to 2. Say the command name twice more.

8. Once you have finished training the command, you will automatically be returned to the **Training** dialog box.



A check mark beside a command name in the **Training** dialog box indicates that the command is enabled. A green flag indicates that the command has been trained.

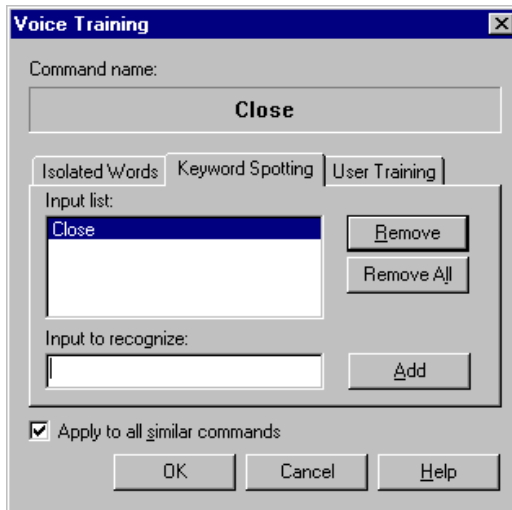
9. Click **Save** to save your training data.

Keyword Spotting

For added flexibility in the way you issue voice commands, VoiceAssist supports *keyword spotting*. This means that VoiceAssist can listen for the voice command as part of a larger phrase. For example, if you activate keyword spotting for the “Close” command, you can issue the command via voice by saying “Close”, “Please Close,” “Close this program now”, etc. The command will be recognized essentially anytime “Close” is said clearly.

To activate keyword spotting for a command:

1. In the **Training** dialog box, select the **Voice** tab page. Now select the command for which you would like to activate keyword spotting.
2. Click **Input...**, and the **Voice Training** dialog box will appear. Select the **Keyword Spotting** tab page.




3. In the **Input to recognize** text window, you can insert the command input you would like VoiceAssist to listen for while you speak. After typing in the input, click **Add** then **OK**.



The more commands that are enabled for keyword spotting, the more complex voice recognition becomes for VoiceAssist. If you enable keyword spotting for too many commands, VoiceAssist's recognition speed and accuracy may be adversely affected.


To deactivate keyword spotting for a command:

1. Open the **Training** dialog box by clicking on the **Main Menu** button  on the VoiceAssist toolbar, then selecting **Training...**
2. In the **Training** dialog box, click on the **Voice** tab, and then select the command for which you would like to deactivate for keyword spotting. Remember, to select an application-specific command, the corresponding application must be open.
3. Click **Input...** and the **Voice Training** dialog box will display. Click on the **Keyword Spotting** tab page. Here you can remove the inputs that were originally specified for keyword spotting.
4. Select an input by clicking on the name of the input in the command list window in the **Training** dialog box. Now click the **Remove** button to remove the selected items. Notice that there is a **Remove All** button as well. You can click on this if you want to remove all voice inputs from keyword spotting.

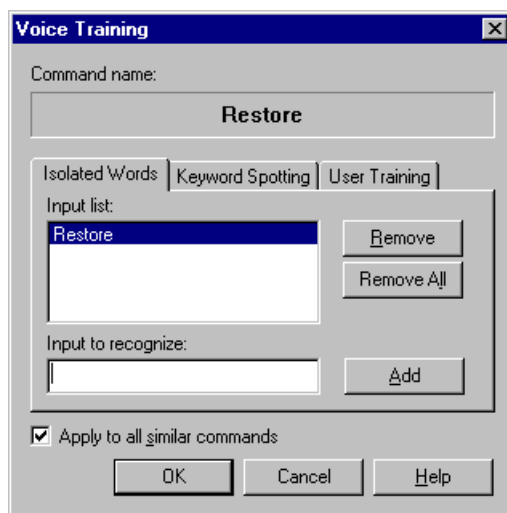
Training Voice Input by Keyboard

Not only can you train a voice input using your personal voice, but you can also train the voice input by keyboard.

To train voice input by keyboard:

1. Click the **Main Menu** button  on the VoiceAssist toolbar, and then click **Training...** The **Training** dialog box will appear.
2. In the **Training** dialog box, click the **Voice** tab page.
3. Click the command you want to train, for example "Close". The selected command is highlighted.

4. Click on the **Input...** button, and the **Voice Training** dialog box will display:



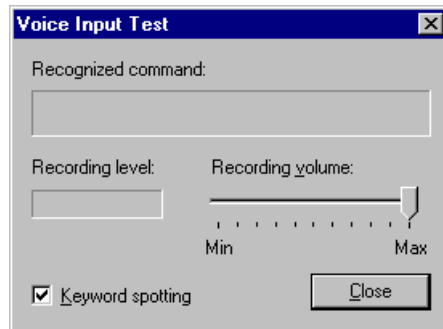
5. Click the **Isolated Words** tab page.
6. In the **Input to recognize** text window, you can insert the command input you would like VoiceAssist to listen for while you say that word or phrase. After typing in the input, click **Add** then **OK**.
7. You can also delete inputs. To do so, click on the input, then click the **Remove** button to remove the selected item. Notice that there is a **Remove All** button as well. You can click on this if you want to remove all the inputs.

Testing for Voice Input

You can test VoiceAssist's recognition of your voice input through the **Voice Input Test** dialog box. If you have a particularly soft or loud speaking voice, you can also adjust the recording volume to optimize voice input recognition.

To test voice input recognition:

1. In the **Training** dialog box select the **Voice** tab, then click **Test...** to display the **Voice Input Test** dialog box:



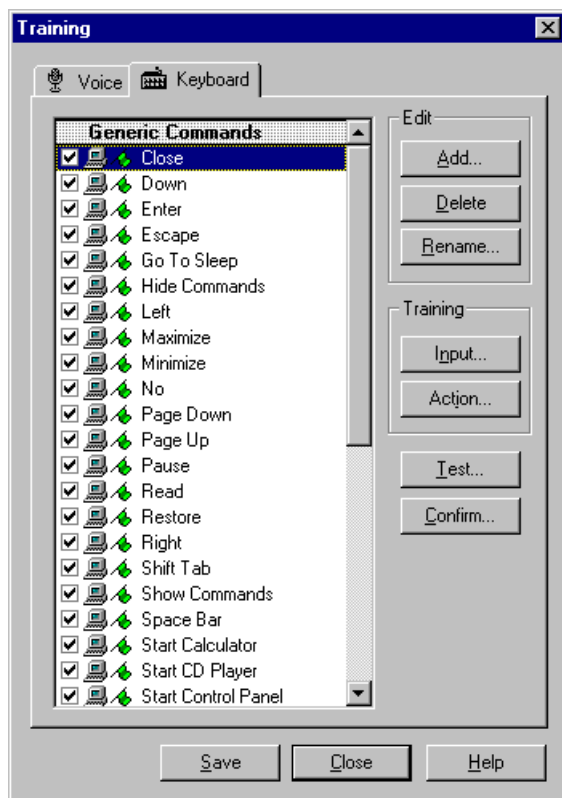
2. Say the name of any enabled command into your microphone. If VoiceAssist recognizes the command, the command name will be displayed in the **Recognized command** box. If it fails to recognize your command, it will display the **Not recognized** message. Speak several commands such as "Start Word Pad" and "Enter". Your voice should register on the **Recording level** meter, and you should see about 3 to 7 green bars fluctuate as you speak. You should be able to tell if you are speaking too loud or soft according to this meter. If it looks like VoiceAssist thinks you are speaking too loud or soft, adjust the **Recording volume** slider left or right, until you achieve the desired results.
3. Click **Close** when you have finished testing.

Training for Keyboard Input

Conveniently, the input for every keyboard command is automatically trained to be identical with its name. If you want to use a different keyboard input for a command, you will need to train VoiceAssist to recognize the new input.

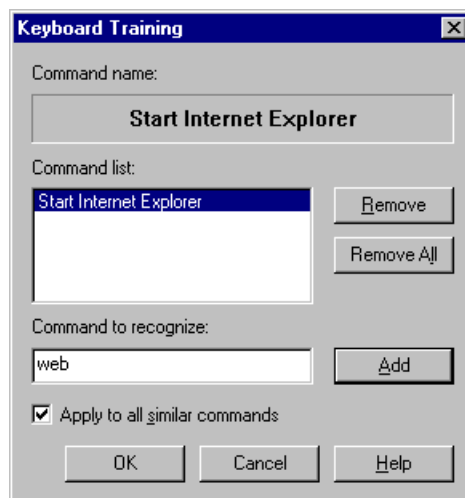
To train commands for keyboard:

1. Open the **Training** dialog box and click the **Keyboard** tab page.



2. Select the command you want to train for keyboard input by clicking on the command name.

3. Click **Input...** to display the **Keyboard Training** dialog box:




The command name appears in the **Command name** and **Input to recognize** boxes.

4. In the **Input to recognize** box, type the text you want VoiceAssist to recognize for this command, and then click on the **Add** button.
5. Click **OK** to save your changes.


Deleting Keyboard Inputs

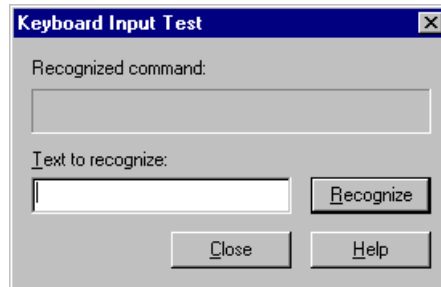
To remove keyboard inputs:

1. Open the **Training** dialog box by clicking on the **Main Menu** button  on the VoiceAssist toolbar, then selecting **Training...**
2. In the **Training** dialog box, click on the **Keyboard** tab, and then select the command for which you would like to delete a keyboard input. Remember, to select an application-specific command, the corresponding application must be open.
3. Click **Input...** and the **Keyboard Training** dialog box will display.
4. Select an input by clicking on the name of the input in the command list window in the **Keyboard Training** dialog box. Now click the **Remove** button to remove the selected item. Notice that there is a **Remove All** button as well.

Testing for Keyboard Input

You can test your keyboard input to make sure that VoiceAssist recognizes your commands.

1. Open the **Training** dialog box by clicking on the **Main Menu** button  on the VoiceAssist toolbar, then selecting **Training....**
2. Click **Test...** in the **Training** dialog box to display the **Keyboard Input Test** dialog box:



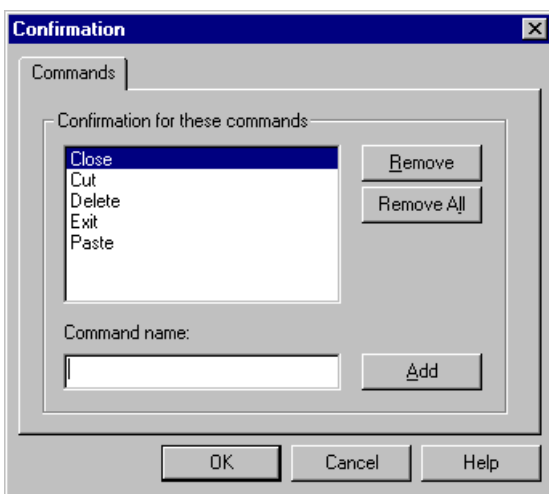
3. Type the name of any enabled command in the **Text to recognize** box.
4. Click **Recognize**. If VoiceAssist recognizes the command, the command name will be displayed. If the system fails to recognize your input, **Not recognized** will be displayed.
5. Repeat this procedure for each command you wish to test, then click **Close** when you are done.

Confirmation Before Executing

You can have VoiceAssist ask for confirmation before executing a command. This option can be activated in the **Training** dialog box. Confirmation will come in the form of a dialog box.

Click the **Confirm...** button in the **Training** dialog box. The **Confirmation** window will appear. Here you can specify which commands require confirmation before executing. For example, if

“Close” is part of the list, then every time VoiceAssist begins to execute the “Close” command, it will ask for your confirmation first.




1. To add a new command to the confirmation list, type in the command name and then press the **Add** button.
2. To delete a command for the confirmation list, select a command by clicking on the command name. Now click the **Remove** button to remove all selected items. Notice that there is a **Remove All** button as well. You can click on this if you want to remove all commands in the confirmation list.
3. When you are done editing the confirmation list, click the **OK** button. Or you can click the **Cancel** button to exit without saving your changes.

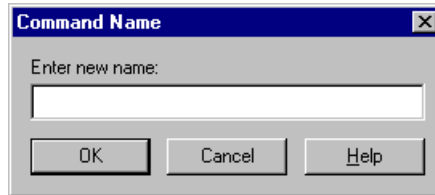
Creating New Commands

You can customize VoiceAssist to perform actions that suit your individual needs. In order to do this you must first create a new command name.

To create a new command name:

1. Open the **Training** dialog box by clicking the **Main Menu** button  on the VoiceAssist toolbar and then selecting **Training...**
2. If you want to create an application-specific command, open the application to which you want to add a new command. For example, open Internet Explorer.
3. In the **Training** dialog box, click in the appropriate command list — generic or application-specific.

4. Click **Add...**. The **Command Name** dialog box will appear.



5. Type the name for your new command in the **Enter new name** box, and then click **OK**. Your new command will appear in the appropriate command set. To learn how to train an action to associate with a new command, see "Chapter 6: Training Actions."

Chapter 6: Training Actions

This chapter explains how to train the action or actions associated with a command. It provides detailed instructions for training simple actions such as opening files and launching applications, and executing macros. Furthermore, instructions to create fully customized actions using MiniScript are described. After reviewing this chapter, you will be ready to train VoiceAssist to do virtually anything in the Windows environment.

Training an Action

After creating a new command, you have to teach VoiceAssist the action or series of actions to be executed when you issue that command.

All actions, including those that VoiceAssist already knows and those you have taught him, are described in a simple scripting language called *MiniScript*. This chapter explains the three methods of creating and modifying MiniScript.

1. Manually type the script directly in the **ScriptEditor** dialog box.
2. Use simple drag-and-drop to create script in the **ScriptEditor** dialog box.
3. Describe the action in the simpler **Training** dialog box, and let VoiceAssist automatically generate the script. With this method you will never see the underlying script in the **Training** dialog box.

The third method, in which you specify the actions in the **Training** dialog box is much easier to do, but it doesn't allow for as many possibilities. Editing the MiniScript in the **ScriptEditor** dialog box gives you the most control and options when specifying the actions associated with the given command.

Selecting the command

When selecting a command in the **Training** dialog box, it is possible to switch between the various input tabs. However, you will notice that the same commands appear under every input tab. Notice that when you modify an action, it applies to the command regardless of the input type.

The Action Dialog Box


The **Action** dialog box is where you can easily specify the action you would like your command to execute. Whatever you specify in the **Action** dialog box, it will automatically be generated into MiniScript. MiniScript and the **ScriptEditor** dialog box are further explained in the "Advanced Training with MiniScript" on page 41 of this chapter.

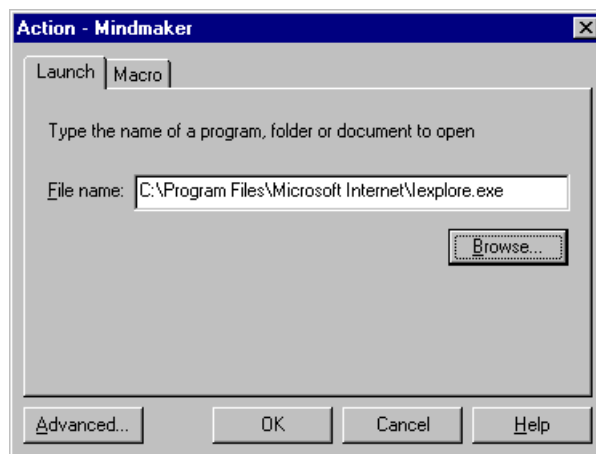
The **Action** dialog box allows you to specify the following types of actions.

- **Launch**—start a program or open a file
- **Macro**—perform mouse and/or keyboard actions

The tabs in the **Action** dialog box allow you to specify these types of actions. You can also execute a combination of these actions for one command, by training the actions under both tabs. As soon as you describe your action(s) in the **Action** dialog box, it will automatically be generated into MiniScript. To view the MiniScript open the **ScriptEditor** dialog box. See "Advanced Training with MiniScript" on page 41 of this chapter.

To access the Action dialog box:

1. Open the **Training** dialog box by clicking on the **Main Menu** button  on the VoiceAssist toolbar, then select **Training...** Now select the command for which you want to train the action. *Remember, to select an application-specific command, the corresponding application must be open.*
2. Click on the **Action...** button in the **Training** dialog box.
3. The **Action** dialog box will appear.




Select the appropriate tab(s) and train the action(s). In order to train more complicated actions or view the MiniScript, click on the **Advanced...** button to open the **ScriptEditor** dialog box.

The Launch Tab Page

To train the action to open a file or launch an application, select the **Launch** tab page.

To open a file or launch an application:


1. Open the **Training** dialog box by clicking on the **Main Menu** button  on the VoiceAssist toolbar, then selecting **Training....** Select the command for which you want to train the action. *Remember, to select an application-specific command, the corresponding application must be open.*
2. Now click on the **Action...** button in the **Training** dialog box.
3. The **Action** dialog box will appear. Select the **Launch** tab page. In the **File name** edit field, type in the name of the file you want the command to open. (You will have to specify the path of the file if it is not in the *Windows system path*.) Or click on the **Browse...** button to find the file manually.
4. Click **OK**.

This action is now trained to open the file or launch the application you specified.

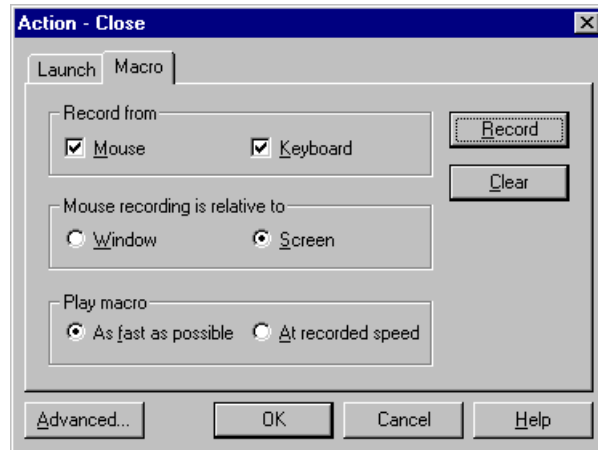
The Macro Tab Page

If you want to train keyboard and/or mouse actions for your command, you can do so on the **Macro** tab page.

To record a macro:

1. Open the **Training** dialog box by clicking on the **Main Menu** button  on the VoiceAssist toolbar, then select **Training....** Select the command for which you want to train the action. *Remember, to select an application-specific command, the corresponding application must be open.*
2. Click on the **Action...** button in the **Training** dialog box.

3. The **Action** dialog box will appear. Select the **Macro** tab page.



4. There are several options available to customize the macro. In the **Record from** group box, check the appropriate box according to the inputs you will give your macro. Check **Keyboard** to use the keyboard, and **Mouse** to use the mouse during recording. Of course you can check both **Keyboard** and **Mouse** to use both types of actions to define the macro.
5. Now choose if you want the macro to be recorded relative to the active window or to the whole screen. For example, if you are recording an action involving something that may not always be at the same place on the screen, like the play button on your CD player, then select **Window**. Conversely, if you are recording an action on something that is always in the same place on the screen, like the **Start** menu button on the Windows taskbar, then select **Screen**.
6. Finally, in the **Play macro** group box choose if you want the macro to be executed as fast as possible or at the speed at which it was recorded.
7. Now you are ready to start recording the macro. Press the **Record** button.
8. Perform the sequence of actions that you want to associate with the command. For example, click the **Address** window of Internet Explorer and then type "www.mindmaker.com", and press ENTER to go to the Mindmaker Web site.

In the case of this macro, you would select **Window** in the **Record from** group box since the action will be executed in the Internet Explorer window. You would also need to select both **Mouse** and **Keyboard**, since this macro will involve both mouse and keyboard actions.

9. To stop macro recording, press the PAUSE key, which is located in the upper-right area of the keyboard.
10. Click **OK** to save the recorded macro. You will be returned to the **Training** dialog box.



Macros will not work properly if a window which is involved in the macro action is minimized or not open. However, the same window can be inactive and behind other windows on your desktop, and the macro will execute as trained. Of course you can always train the action to first open the necessary window and then execute the original action.

To stop macro recording and playback

As described in the previous section, you can stop macro recording by pressing the PAUSE key. The macro recording will also stop if you press the key combination of CTRL+ALT+DELETE. Both of these methods can be used to stop a macro playback. In the case that you want to interrupt a macro action, press either the PAUSE key or the key combination of CTRL+ALT+DELETE.

Advanced Training with MiniScript


What is MiniScript?

MiniScript is a simple scripting language developed by Mindmaker, which allows you to describe complex actions to execute for each command. Although using MiniScript to describe an action requires a bit more attention than using the **Launch** and **Macro** tab pages of **Action** dialog box, it has significant advantages. First, it is easier to specify exactly what action you want VoiceAssist to perform. With MiniScript, you can modify and edit a variety of different components of the action without completely retraining it, and it offers a much larger variety of action types. All in all, MiniScript simply gives you more control over the actions you associate with the command. MiniScript is edited in the **ScriptEditor** dialog box.

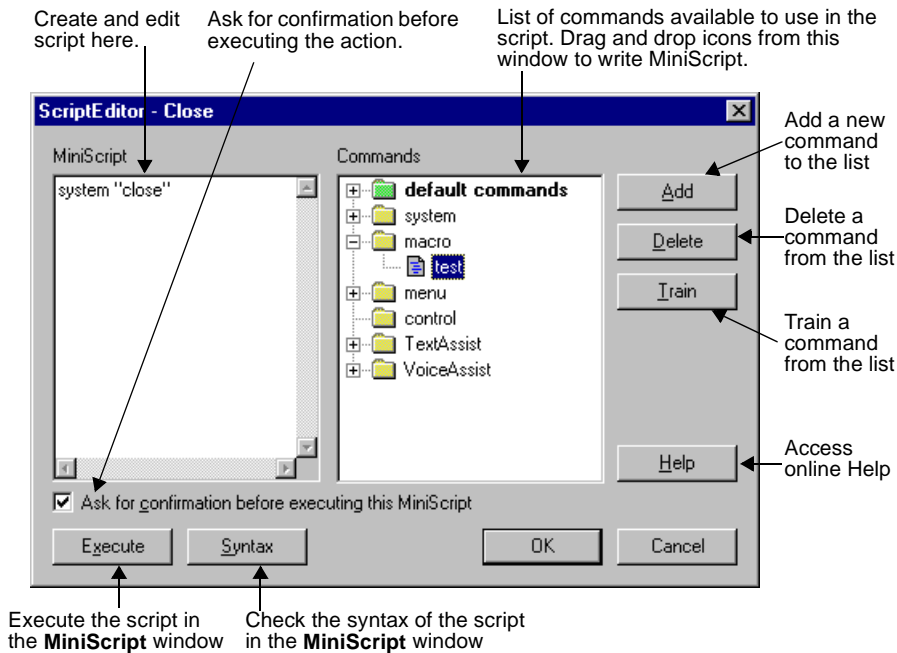
The ScriptEditor

If you want to train your command for more sophisticated actions or view the existing MiniScript, you can do so in the **ScriptEditor** dialog box.

To access the ScriptEditor dialog box:

1. Click the **Main Menu** button  on the **VoiceAssist** toolbar, and then select **Training....** This will display the **Training** dialog box.

2. In the **Training** dialog box select the command for which you would like to train the action. For example, select the “Close” command. *Remember, to select an application-specific command, the corresponding application must be open.*
3. Click the **Action...** button to display the **Action** dialog box, and then click the **Advanced...** button. The **ScriptEditor** dialog box will appear.



The following section explains how to edit script in the **ScriptEditor** and what other options are available in this dialog box.

4. When you are done editing and testing the script, click **OK** to return to the **Training** dialog box.

Editing MiniScript

In the **ScriptEditor** dialog box, you may create, modify, and test the script associated with the selected command.

The **Commands** window contains a list of folders with a variety of actions you can choose to be executed within MiniScript. Descriptions of each script command type are found later in this chapter.

Edit the script by either typing directly into the **MiniScript** window, or drag-and-drop the script commands from the list in the **Commands** window. If the script command requires parameters, a dialog box will be displayed where you can specify them. Note that the script engine can only execute commands that appear in the **Commands** window.

When you are done editing the script, click on the **Syntax** button to verify the syntactical correctness of the script. If the script syntax is correct, a message highlighted in green will appear at the bottom of the **MiniScript** window, informing you that the script is okay to use. If there is a syntactical error in the script, then the first line of script which contains an error will be highlighted in red.

Click **Execute** if you want to test the script by executing it. VoiceAssist will execute the actions specified in the **MiniScript** window.

Adding, deleting, and training script commands

The **Add**, **Delete**, and **Train** buttons in the **ScriptEditor** dialog box apply to the script commands that appear in the **Commands** window. These buttons may or may not be available depending on the type of command currently selected in the list. For example, you may select the **macro** folder, and click **Add** to create a new macro.

Confirmation setting

Select or clear the **Ask for confirmation before executing command script** check box, depending on whether you want VoiceAssist to ask you for confirmation before executing the script associated with the command.



*Once you have created MiniScript for a command, and you click **Action...** on the **Training** dialog box to modify this command, the **ScriptEditor** dialog box will appear directly. You will no longer see the **Action** dialog box appear for that command.*

Understanding MiniScript Syntax

The commands used in MiniScript are typically followed by one or more parameters to specify the details of what the script command does. You do not have to remember the number and type of parameters for each command, because the editor shows you the syntax for each one as you are creating the script.

See the "The ScriptEditor" on page 41 and "Script Command Types" on page 44 for detailed explanations of the ScriptEditor and the various script command types.

Script command typography

Each script command is followed by a space, and one or more parameters. If more than one parameter is used, then the parameters must be separated by a comma. To make the script easier to read, you may separate the parameters using a comma followed by a space.

The following is a specific example of the **system** script command, which may have two parameters:

```
system "enter", 2
```

Some script commands require a string parameter. A string parameter is always surrounded by two quotation marks. The string between the quotation marks may contain any character, including spaces. The following is another example that shows the proper syntax for writing a string parameter:

```
say text, "Hello!"
```



*With the exception of the string parameters for the **say**, **keyboard**, and **launch** script commands, all command parameters are case sensitive.*

Script Command Types

MiniScript supports a variety of script command types. Each require different parameters, which are described in the following sections.

Default commands

There are three default commands that are always available to be included in the script.

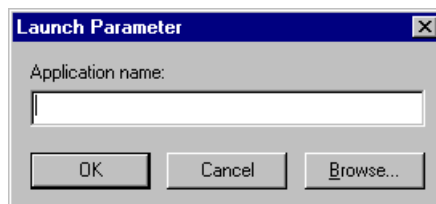
Launch

The **launch** script command allows the action to open or launch an application or file.

Example:

```
launch "notepad.exe"
```

When you drag-and-drop this command from the **Commands** window, the **Launch Parameter** dialog box appears where can type in the file name you want to launch. Optionally, you can manually navigate to find the file by pressing the **Browse...** button.

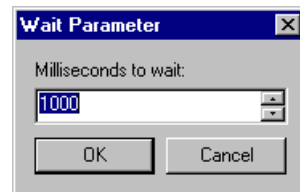




Here is a trick to easily create a line of MiniScript to open a file. Drag the icon of the file you would like to open to the **MiniScript** window. A script line should appear, which will now open the file that you selected.

Wait

The **wait** script command allows your action to pause for a specified amount of time. This is particularly useful when your MiniScript action is waiting for another action to process. When you drag the **wait** script command into the **MiniScript** window, the **Wait Parameter** dialog box will appear.



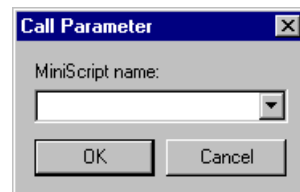
Specify how many milliseconds you would like the system to wait. Note, 1000 milliseconds equals 1 second.

Example:

```
Wait 1000
```

Call

The **call** script command can execute another, existing MiniScript action. When you drag the **call** script command from the **Commands** window to the **MiniScript** window the **Call Parameter** dialog box will display.



Here you can type in the name of the MiniScript action you would like to execute, or click on the arrow to select the action from the drop-down combo box. If you create an endless loop with the **call** command, then the action will be terminated after 10 iterations.

Example:

```
call "test"
```

System commands

These functions are available to run for all commands. However, you cannot add to, delete, or train the system commands. These commands appear in the **Generic Commands** list of the **Training** dialog box and **Commands List** window. They will execute the specified action in the active application.

The second parameter is an integer number (1-65000), and it specifies the number of times the line will be executed. This parameter is optional, and if not used then the system command is executed only once.

Example:

```
system "page down", 4
```

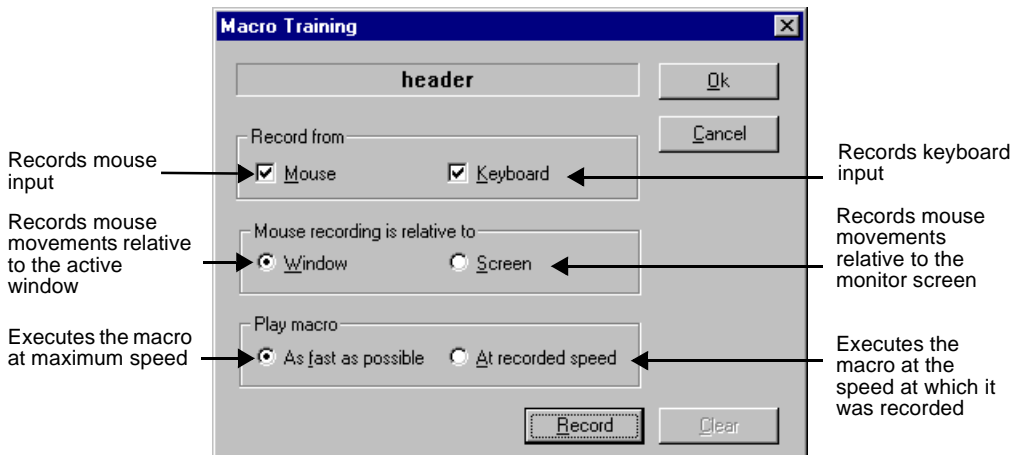
Macro commands

The **macro** script command allows the action to perform a previously trained macro.

You may add, delete or train macros. Macros that you have created and trained are available to all other commands, therefore their names must be unique.

Example:

```
macro "header"
```



You train macros in exactly the same manner as training macros in the **Action** dialog box. See the section, "The Macro Tab Page" on page 39 of this chapter for more details.



Once a macro has been trained, it will be available for you to use whenever you are creating MiniScript. Just look under the **macro** folder in the **Commands** window of the **ScriptEditor** dialog box.

Menu Commands

The menu items of the active window are available to use, but not to train. When using these commands from the list, you should make sure that the application you wish to control via the script is open. Only the active application menu commands are available.

Example:

```
menu "Print "
```

Control commands

If a dialog box is opened in the active application, the controls (buttons, lists, combo boxes) of that dialog box become available in the **control** script command folder. Dragging them into the MiniScript has the effect of clicking them.

Example:

```
control "OK "
```

TextAssist commands

TextAssist commands can be used to automatically start TextAssist, and have it read out loud from the open document. If you want to include these commands within a MiniScript, make sure that you have opened the right document before launching TextAssist. If not, TextAssist will start to read the text from the currently active application.

There are four TextAssist commands that are available in the ScriptEditor:

- TextAssist "start": TextAssist is started and it begins to read the currently opened document.
- TextAssist "pause": TextAssist is paused. To restart it, you have to use the TextAssist resume command later in the MiniScript.
- TextAssist "resume": TextAssist continues reading from where it paused.
- TextAssist "stop": TextAssist stops reading.

VoiceAssist commands

There are four VoiceAssist commands that are available in the ScriptEditor:

- VoiceAssist "show commands": VoiceAssist will display the **Commands List** window.
- VoiceAssist "hide commands": VoiceAssist will hide the **Commands List** window.
- VoiceAssist "wake up": VoiceAssist will "wake up" from its "sleep".
- VoiceAssist "go to sleep": VoiceAssist will "go to sleep", and not respond to any commands other than "wake up".

Sample MiniScript

Here is an example of a working MiniScript command. A line-by-line explanation of the script follows.

Let's presume that you have previously recorded two macros:

1. "Hello": types the word "Hello"
2. "Test": types the word "Test"

The sample script:

```
Launch "notepad.exe"
wait 1000
macro "Hello"
menu "File|Save As..."
macro "Test"
control "Save"
system "Close"
VoiceAssist "show commands"
```

The line-by-line explanation of the code above:

Starts the Notepad application.

Waits one second.

Types the word "Hello".

Opens the **File** menu and the **Save As** dialog box

Save with the file name "Test"

Click on the **Save** button

Close Notepad

Displays the **Commands List** window



It is important to remember that a script command must fit on a single line, since the script engine can only interpret commands on a single line.

Chapter 7: User Settings

This chapter teaches you how to manage VoiceAssist training data by manipulating user and application templates. *User templates* are files that contain a user's complete training data. For example, "JDoe" could be a user's template that would contain all of John Doe's changes and additions to training. *Application templates* are part of a user template that contain the training data for each individual application. For instance, an application template called "CD Player", would contain all the training changes and additions for the CD Player application. User and application templates can be copied, renamed, deleted, imported, exported, and removed.

The ability to control user and application templates in these various ways can save you a lot of time. If another user has added and trained several commands to their VoiceAssist user template, you can simply import their training data to your user template and take advantage of their additional commands. And, provided that the original user did not retrain any of the voice input, you can use the imported commands immediately, without having to do your own training.

Understanding User and Application Templates

You manage user and application templates in the **Users** dialog box. User templates hold the commands trained by each user. Whenever a new user file is created, commands in the generic set are automatically copied into the user file and saved with an ***.srt** extension in the VoiceAssist directory. Application templates, which hold the trained application-specific commands, e.g., the commands for an application like Microsoft Word, are saved with an ***.apt** extension, but first must be exported to a file to be accessible.

The Users Dialog Box

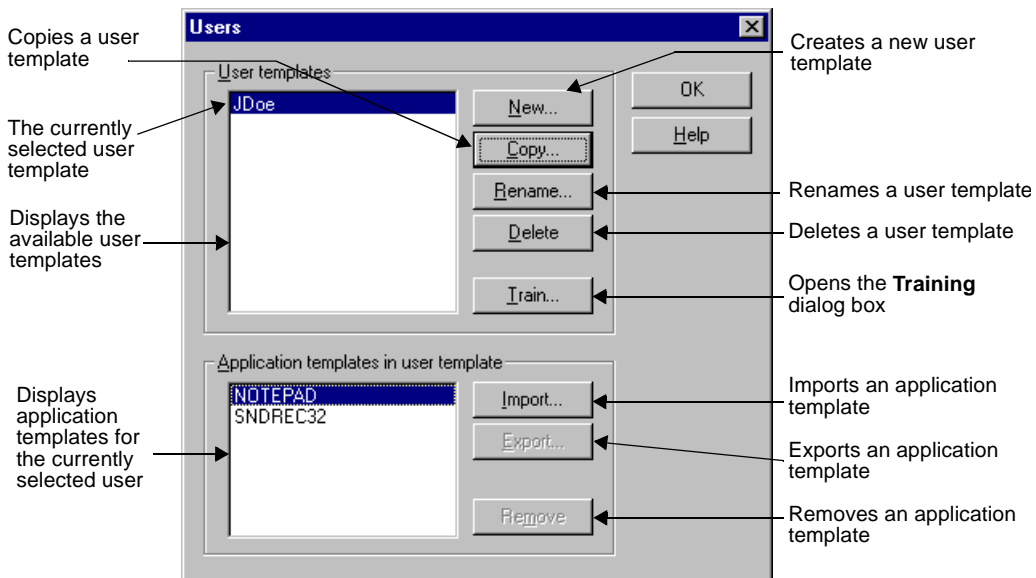
Manipulation of user and application templates takes place in the **Users** dialog box.

Opening the Users Dialog Box

To open the Users dialog box:

1. Click the **Main Menu** button  on the VoiceAssist toolbar.
2. Click **User....**

You can create, copy, rename, delete, import, export, and remove user and application templates in the **Users** dialog box.



Managing User Templates

When you open VoiceAssist for the first time, the program automatically creates a user template with the same name as your Windows login name. For example, if you enter "JDoe" to log into Windows, then your default user template name will be "JDoe". If you do not use a login name, your default user template name will be "Unknown".

By managing your user templates, you can effectively reduce the amount of training that is necessary both for yourself and for other users. VoiceAssist allows you to:

- Create new user templates
- Copy user templates
- Rename user templates
- Delete user templates

Creating User Templates

To create a new user template:

1. Click **New...** in the **Users** dialog box to access the **New User Template** dialog box.
2. Enter a name in the **Enter name of new user** text field. Then click **OK**.

3. The name of the file will now appear in the **User templates** window of the **Users** dialog box.

Copying User Templates

You can use the trained commands from other users by copying their templates. Copying this information into a new template can dramatically reduce training time for new users.

To copy a user template:

1. In the **Users** dialog box, select the name of the user template that you would like to copy.
2. Click **Copy...** to open the **Copy User Template** dialog box.
3. Type in the name you want to give to the new user template in the **Enter name of user to copy template to** text field and click **OK**. The name of the new user template will appear in the **User templates** window.

Renaming User Templates

You may wish to rename a user template—for example, if you don't like the default name given to your user template.

To rename a user template:

1. Select the file to rename from the **User templates** list.
2. Click **Rename...** to open the **Rename User Template** dialog box.
3. Delete the old user name and type a new user name in the **Enter new name** text field. Click **OK** to confirm. The new user template appears in the **User templates** window.

Deleting User Templates

VoiceAssist allows you to delete unwanted user templates.

To delete a user template:

In the **Users** dialog box, select the user template from the **User templates** list. Click **Delete**, then click **Yes** to confirm.

Managing Application Templates

Application templates, which use the extension ***.apt**, hold trained commands for specific applications. Whenever you train one or more application-specific commands in VoiceAssist, an application template is automatically created to store the training data. A separate application template is formed for each application in which you perform training. So, if you train at least one command in both Internet Explorer and Netscape Navigator, two application templates will be created—one for each application.

Your application templates form your user template. If you copy your user template, all of your application templates are copied as well. You can also “copy” individual application templates by importing them from another user or exporting them to another user.



If you, or anyone with whom you would like to share application templates, are an Assistant (Prody Parrot or Zymo the Alien) user, please note that Assistant applications templates are not totally interchangeable with VoiceAssist application template files. VoiceAssist templates may be successfully imported and used by the Assistant; however, Assistant application templates that are imported into VoiceAssist may not work properly.

Creating an Application Template

Application templates are automatically created when you train a new command for a specific application. If you want to create an application template, you must train or edit a command for that application. (see the “Basic Training Procedures” section in “Chapter 5: Training Command Inputs” of this User’s Guide.)

Removing Application Templates

When you no longer want previously changed training for a specific application, you can delete the corresponding application templates from your user template.

To remove an application template from a user template:

1. Click the relevant user template in the **User templates** window of the **Users** dialog box.
2. Select the application template to be deleted from the **Application templates in user template** window.
3. Click **Remove** and then **Yes** to confirm.

Exporting Application Templates

To make an application template available to another user, you need to *export* it, which means creating it as a separate file. You can then copy this file into whichever user template you choose (see “Importing Application Templates” below).

To export an application template:

1. In the **Users** dialog box, select the user template which contains the application template you want to export.
2. Select the application template to be exported from the **Application templates in user template** box.

3. Click **Export...** to open the **Export Application Template** dialog box.
4. Select the directory in which you want to save the file, or keep the default directory.
5. Enter a new name in the **File name** box or keep the default name. Make sure to use the ***.apt** extension.
6. Click **Save**. The application template file will now appear in the directory you selected.

Importing Application Templates

To include an application template in a user template, you need to *import* it from the directory in which it is stored. You can copy this file into whichever user template you choose.

To import an application template:


1. In the **Users** dialog box, select the user template into which you want to copy an application template.
2. Click **Import...** to open the **Import Application Template** dialog box.
3. Select the VoiceAssist application template file (*.apt) you wish to import.
4. Click **Open**, and the application template file will appear in the **Application templates in user template** window.

Appendix A: Troubleshooting

This appendix contains suggestions on how to solve common problems that may arise with VoiceAssist.

VoiceAssist does not recognize any voice commands.

If VoiceAssist is recognizing none, or few, of your voice commands, try the following steps:

1. Make sure your audio set-up is correct.
2. Check the recording level in the **Voice Input Test** dialog box (see “Basic Training Procedures” in “Chapter 5: Training Command Inputs” for instructions).
3. Make sure that the **Voice Input** button  is depressed on the VoiceAssist toolbar.
4. Speak in a natural, conversational, and relaxed tone.
5. Avoid prolonged pauses between the words of a command.
6. If you trained some of the voice commands, then try to use the same vocal tone and speed that you used during training.
7. If using a hand-held microphone, position the microphone between 5 and 11 inches from your mouth.
8. Try re-naming misrecognized commands to different words or phrases.
9. If the problem still persists, do additional training on the misrecognized words.
10. Try deactivating some of the voice commands that you never use. (See “Chapter 5: Training Command Inputs”.)
11. Use a unidirectional or high-quality hands-free headset microphone for optimum recognition results. Both “headset-style” and “stick-style” microphones are acceptable for use with VoiceAssist. Which style is best for you depends on the environment in which you will be using the software. In general, however, most users find that the headset style provides the most consistent and reliable voice recognition performance.

VoiceAssist sometimes confuses one command with another.

If VoiceAssist recognizes most of your voice commands, but confuses a few of them, then try the following steps:

1. Rename the most commonly misrecognized commands to different words or phrases.
2. Don't use command names that sound similar to other commands—they may be misrecognized.
3. Avoid very short command names. For example, using "Bold Font" instead of just "Bold" yields better results.
4. If keyword spotting is enabled for numerous commands, try disabling it for some of them.
5. If the problem still persists, do additional training on the misrecognized words.
6. If this problem happens often, you might want to consider trying a different microphone. Some users notice a significant difference in speech recognition accuracy when using one microphone versus another.

See *also* "VoiceAssist misrecognizes a lot of commands for one particular application." below.

VoiceAssist misrecognizes a lot of commands for one particular application.

If VoiceAssist is having problems recognizing commands of one particular application, this is because this application has many active commands. To improve the recognition of VoiceAssist for this application, disable or delete commands that you will not frequently use from the application-specific command set. (See the section in "Chapter 5: Training Command Inputs".) This will reduce the number of commands that VoiceAssist has to recognize for that application, and thereby improve the recognition accuracy.

See *also* "VoiceAssist sometimes confuses one command with another."

VoiceAssist does not respond to any commands issued by any input method.

If VoiceAssist is not responding to your commands, first make sure that the appropriate input button(s) are depressed. If there is still no response, then it is possible that VoiceAssist is in sleep mode. While "sleeping", VoiceAssist will not respond to input of any kind, with the exception of the command "Wake Up". To bring the program out of sleep mode and reactivate input for all commands, issue the command "Wake Up".

VoiceAssist does not allow voice input training.

If VoiceAssist does not let you train by voice and indicates that the voice input was too long, this could be due to background noise. In this case, try the following steps:

1. Make sure your audio setup is correct. (See “Getting Started with VoiceAssist” on page 3.)
2. Check the recording level in the **Voice Input Test** dialog box.

I need a microphone but I don't know what kind to get.

There are many factors to consider when selecting a microphone to use with your voice recognition product:

- For use with your sound card, you will need a microphone with an 1/8" plug (as opposed to the 1/4" plug size that is commonly used with stereo equipment).
- There are two basic types of microphone elements: dynamic and condensor. As shown in the picture on the next page, they have slightly different plugs. Condensor microphones usually give better performance than dynamic ones; however, unlike dynamic microphones, condensor microphones need power to operate. This power may be supplied by batteries in the microphone, or by the sound card itself.



Dynamic microphone plug



Condensor microphone plug

- If you have, or are planning to buy, a condensor microphone, keep in mind that it needs a power source. Some microphones have a battery to provide this power. Otherwise, the sound card must provide power to the microphone. Make sure your sound card can do this.
- Be aware that the configuration of the microphone jack on some sound cards and the plug on some microphones is not standard. An incompatible connection between microphone and sound card can result in damage to the microphone, the sound card, or both. If your microphone was not included with your sound card, check the microphone packaging, or with the sound card manufacturer, to make sure the microphone is compatible.

- Microphones are available with different pick up schemes. You should select a *unidirectional* microphone, which picks up sound input from just one direction and suppresses sound input from all other directions. It is the preferred type of microphone for use with speech recognition products because, when positioned properly, it picks up the user's speech and ignores extraneous background noise.
- Both "headset-style" and "stick-style" microphones are acceptable for use with VoiceAssist. Which style is best for you depends on the environment in which you will be using the software. In general, however, most users find that the headset style provides the most consistent and reliable voice recognition performance. Also, many users appreciate the convenience of a headset that has both a microphone for speaking and earphones for listening.
- Do not select a cheap microphone. Although there may be exceptions, most microphones under \$10.00 will not provide the desired level of voice recognition accuracy.

I've retrained a command action, but the action is not executing as I specified.

Every action, both previously known to VoiceAssist and those trained by you, are described in MiniScript. This even applies to the actions which you defined in the **Action** dialog box. When retraining a command in the **Action** dialog box, the existing action still exists, preceding the action you are training. To delete the previous action, remove the MiniScript which describes it. Look in the **ScriptEditor** dialog box to do this.

I can't move my mouse or type.

Possibly, you've trained a long macro that is executing while you are trying to use your mouse or keyboard. While a macro is executing, you do not have control over these two devices. Instead, try retraining your macro to execute as fast as possible, or you could also try specifying your action in MiniScript in the **ScriptEditor** dialog box.

Sometimes VoiceAssist does not execute the command or commands I issue.

When a command of a certain type (generic or application-specific) is recognized, and the associated action is executing, a second command of the same type cannot be executed. Before issuing the next command of the same type, the current action must first end. This situation may also occur if you define the same hot spot for more than one command. Keep in mind that only the first command of the same type (generic or application-specific) will execute.

The software freezes when launched.

Perhaps you have launched the software and it brings up the splash screen, but does nothing afterwards. If you encounter this problem, install the latest version of Internet Explorer, and launch the software again.

Appendix B: Glossary

action	<p>An <i>action</i> is the procedure that VoiceAssist carries out when you issue a command.</p> <p><i>See also</i> command</p>
agent	<p>An intelligent program that is able to carry out your commands.</p> <p><i>See also</i> intelligent assistant</p>
application template	<p>A file that contains application-specific command training. VoiceAssist application template file names have an *.apt extension.</p> <p><i>See also</i> user template</p>
application-specific command	<p>A command that is associated with a specific application. Whenever you open a program, VoiceAssist extracts the menu commands and makes them available for your use.</p> <p><i>See also</i> command, command set</p>
automatic training	<p>VoiceAssist automatically self-trains the inputs and actions associated with the menu commands of any application that you open.</p>
command	<p>A command has three components: the command name, the input, and the action. There are three types of commands: generic commands, application-specific commands, and HTML commands.</p> <p><i>See also</i> input, action</p>
command set	<p>A group of commands. VoiceAssist uses three command sets: generic commands, application-specific commands, and HTML commands.</p> <p><i>See also</i> command</p>
Commands List window	<p>The window that lists all currently available commands.</p> <p><i>See also</i> command</p>
generic command set	<p>Commands that are always available when VoiceAssist is running, except when it is in sleep mode.</p> <p><i>See also</i> application-specific command, command</p>

HTML command set

A group of commands that is associated with HTML links. Whenever you open a program with HTML links, VoiceAssist extracts the link(s) and makes them available for your use.

See also command, command set

input

A voice or keyboard command, or double-clicking of a command in the **Commands** window. For example, you can say the “Close” command to close a window, type “Start Calculator” to start the Calculator program, or double-click the “Start CD Player” command to start the CD Player.

See also command

intelligent assistant

A program that performs a variety of useful tasks on the user’s behalf.

See also agent

Keyboard Input box

The dialog box that enables you to issue keyboard commands to VoiceAssist. You open it by pressing SHIFT+ESC.

keyword spotting

A feature that enables VoiceAssist to recognize voice commands as part of larger phrases. For example, if you activate keyword spotting for the “Close” command, you can issue the command via voice by saying “Close,” “Please Close,” “Close this program now,” etc.

macro

A sequence of actions executed by a single command. For example, you can execute a macro that will open a program, enter text, then save the document whenever you say “letter”.

MiniScript

The language you use to describe actions performed in the Windows environment.

ScriptEditor

The advanced dialog box where you can train the actions of commands. The description of these actions are written in MiniScript.

See also MiniScript

script command

A command that appears in the **Commands** window of the **ScriptEditor** dialog box. Use script commands to specify the action associated with an input command.

See also ScriptEditor, Training dialog box, command

speech recognition

The ability of a program to understand spoken words or sound. The two types of speech recognition are dictation, which translates words into text, and command-and-control, which uses speech to execute commands.

sound card

A piece of hardware that enables a computer to generate and record sound.

training

The process by which you train VoiceAssist to recognize a command. You can create a new command name, teach VoiceAssist to recognize your input, and then teach it to perform an action.

See also command

Training dialog box

The dialog box in which you create, edit, and train commands.

See also command, training

unidirectional microphone

A unidirectional microphone picks up sound input from just one direction and suppresses sound input from all other directions. It is the preferred type of microphone for use with speech recognition products because, when positioned properly, it picks up the user's speech and ignores extraneous background noise.

user template

A file that contains your training data. Whenever you train VoiceAssist to recognize a command, the information is saved in your user template. VoiceAssist user template file names have an *.srt extension.

See also application template

VoiceAssist toolbar

The main VoiceAssist user interface. The toolbar allows you to determine the program's input and access the **Main Menu**.

Index

A

- actions
 - training new 37
- application templates
 - creating 52
 - exporting 52
 - importing 53
 - managing 51–53
 - removing 52
- automatic training 23

B

- Before 3

C

- command components
 - action 23
 - command name 22
 - input 22
- command inputs 12
- command methods
 - by double-clicking 18
 - by keyboard 17
 - by voice 17
- command sets 21
- command types
 - application-specific 16, 22
 - generic 16, 21
- Commands window, displaying 9
- condensor microphone 57
- confirmation before executing 34
- conventions, typographical ix
- Creating 12
- creating new commands 12, 35

D

- displaying available commands 9

- dynamic microphone 57

E

- exiting VoiceAssist 5

G

- Go To Sleep command 18

H

- headset-style microphone 58
- HTML command set 16

I

- input options, activating 21
- installation
 - launching setup 4
 - system requirements 3
- issuing commands
 - by double-clicking 11, 18
 - by keyboard 11, 17
 - by voice 11, 17

K

- key features 2
- keyboard input
 - testing 34
 - training 32
- keyword spotting 28

M

- microphone, selecting the best type of 57

N

- new commands, creating 35

P

Prody Parrot package version 3

Q

quitting VoiceAssist 5

R

recording level, adjusting 31
removal 4

S

sleep mode 18
stand-alone version 3
starting VoiceAssist 4
stick-style microphone 58
system requirements
 installation 3

T

testing
 keyboard input 34
 voice input 31
The pointing hand x
toolbar
 button functions 9
 displaying and hiding 5
Training 37
training input
 keyboard input training 32–33
 voice input training 25–28
training, overview 21–23
types of microphones 57

U

undoing voice input retraining 28
uninstalling VoiceAssist *See* removal
user templates
 copying 51

creating 50
deleting 51
managing 50–51
renaming 51

V

version, stand-alone vs. Prody Parrot
 package 3
voice input
 keyword spotting 28
 testing 31
 undoing retraining 28
VoiceAssist Software Development Kit 1

W

Wake Up command 18