

RIGHT SEAT SOFTWARE, INC.



**3D Talking
Animated Characters
for PowerPoint®!!!**

User Manual

Right Seat Software, Inc.
1110 12th Street, Unit A
Golden, CO 80401
303/278-2244
support@voxproxy.com
©2001-2004, Right Seat Software, Inc.

Table Of Contents

Introduction.....	1
Welcome to Vox Proxy!	1
What's New in Vox Proxy Version 3	1
The Vox Proxy Player	4
Getting Started.....	6
PowerPoint menu	7
Script Basics	8
Speech Basics	9
Character Animation.....	10
Guide to your First Script.....	12
Sample Scripts	16
Simple script with two characters:	16
Script with PowerPoint commands:	16
Program Updates and Support.....	17
Character Gallery	19
Character Gallery Form.....	19
Close All	19
Downloading Characters	20
Inserting a Show Command	20
Placing Characters	20
Previewing Animations.....	21
Properties	21
Removing a Character	23
Selecting characters.....	23
Character Images	23
Macros	25
Introduction	25
The Macro Menu	25
Running macros.....	27
Reference.....	29
Character commands	29
Balloons.....	29
Disappear	29
Do	30
GetstureAt	30
Hide.....	31
IdleAnimations	31
Interrupt.....	32
Language.....	32
Listen	33
ListenFor	34
MoveTo	35
NoBalloons.....	35
PointAt	36
Say	36
SayWav	37
ScaleCharacters.....	38
Show.....	39
Size	40
SoundEffects	40
Speak.....	41
Think.....	41
TTSEngine	42
Wait.....	43
WaitCommands	43
General	44

File References in Vox Proxy.....	44
General Rules for Scripts	45
Speech Tags.....	46
Wait Options for Script Commands.....	47
Miscellaneous commands.....	48
Background.....	48
Bitmap	48
CloseBitmap.....	49
CloseExcel	50
CloseFlash	50
CloseMedia.....	51
CloseWord	51
Delay	51
DisplayClock.....	52
DualMonitor	54
EndMacro.....	55
GetMacro	55
Input.....	56
ListBox	57
MacroMenu	58
MediaVolume.....	59
MessageBox	59
onExcelChange	60
PlayFlash	61
PlayMedia	62
PlayWav	64
RandomLineFrom.....	65
RandomMacro	65
ReadExcel	66
ReadFile	66
Run.....	67
RunMacro	68
Set.....	69
SetPrefs.....	69
ShowExcel	70
ShowWord	72
StartTimer	73
WaitFor	74
WriteExcel	75
WriteFile.....	76
PowerPoint commands	77
Click.....	77
EndShow	78
FirstSlide	78
GotoSlide.....	78
LastSlide.....	79
LastSlideViewed.....	79
Notes about navigating among slides in PowerPoint.....	79
NextSlide.....	80
PreviousSlide.....	80
RunSlideShow	80
WaitforClick.....	81
WriteShapeText.....	81
Programming commands	82
BookMark	82
GoTo.....	82
:If ... :Else ... :End	83
:Select... :Case... :Else... :End	84
Variables	85

Script Wizard	91
Script Wizard.....	91
Character Commands	91
Miscellaneous Commands	92
PowerPoint Commands	93
Programming Commands.....	94
Script Writer	95
Script Writer Form	95
Script Edit Window.....	95
Power Scripting Menu.....	96
Slide Tab	96
Macro Tab.....	97
Menus	97
Tools.....	97
File location preferences	97
Miscellaneous preferences.....	98
Preferences.....	98
Pronunciation	99
Variables Maintenance.....	100
Wait preferences.....	101
Edit menu	102
File menu.....	103
Help menu	103
Insert menu	104
Tools menu	105
View Menu	107
Toolbars	109
Speech Toolbar.....	109
Standard Toolbar	110
Tools Toolbar	110
Wizard.....	111
Tips and Techniques	113
Interacting with PowerPoint.....	113
Navigating in PowerPoint	114
Scripting Speech.....	114
Using Multiple Characters.....	116
Troubleshooting	119
Character Behavior	119
Miscellaneous	120
PowerPoint Behavior	121
CD Prep	123
Introduction	123
What's new in Version 3	123
Presentation Guidelines	124
The PowerPoint 2003 Viewer	124
Minimum Requirements for Playback	125
Recommended:	125
How the CD will Run.....	125
Administrative Rights and other XP Issues	127
The CD Prep Wizard	127
Introduction	127
CD Layout.....	128
QuickPrep	128
CD Image Folder.....	129
Single Presentation or Menu	129
Menu Builder.....	129
CD Files.....	130
Characters	131
Languages	132

Vox Proxy Documentation

Info File	132
Author File	133
CD Startup Message and Copyright	133
Finish	134
Burning the CD	135
CD Diagnostics	136
Copyrights and trademarks	139
Glossary.....	141
Index.....	143

Introduction



3-D Talking Animated Characters for PowerPoint®

Welcome to Vox Proxy!

Vox Proxy is a user-oriented authoring and playback system for scripting 3-D talking animated characters and other advanced multi-media features in Microsoft PowerPoint® slide shows. For authoring, it works with PowerPoint 2000 or newer. For playback, it works with any version of PowerPoint as well as the free PowerPoint 2003 Viewer.

Though a PowerPoint add-in, the Vox Proxy menu appears in PowerPoint and allows you to create and edit your scripts side-by-side with your slides. Scripting is click-simple with Vox Proxy's comprehensive Script Writer and Script Wizard.

Vox Proxy works through scripts that are saved in each slide of the presentation file. We give you some basics about scripts, and an introduction to character speech and animations. An extensive reference section provides details about each Vox Proxy script command.

Previous users of Vox Proxy should check out What's new in Version 3.

To get started quickly, look at the step-by-step Guide to your first script.

For information on playing your Vox Proxy presentations on other computers, read about the Vox Proxy Player.

Vox Proxy's optional CD Prep feature lets you create self-starting, self-contained CDs that you can distribute royalty-free. The CD's will play on most Windows computers from 98SE on. Each CD will contain the Microsoft PowerPoint 2003 Viewer, so PowerPoint is not required to play it.

Additional, highly animated characters are available on our web site at www.voxproxy.com. Click on the character gallery and look for the extra-cost characters.



Vox Proxy uses Microsoft Agent technology and the advanced application development tools of APL2000, Inc.

Copyrights and trademarks

[What's New in Vox Proxy Version 3](#)

With Version 3, Vox Proxy adds support for two important new Microsoft products: the PowerPoint 2003 Viewer and Windows Media Player 9.

- Support for the new PowerPoint 2003 Viewer means that you can distribute presentations to people who do not have PowerPoint and, as long as they've installed the free Vox Proxy version 3 Player, they can still play your slide shows complete with all your Vox Proxy scripts.

- The Media Player 9 support is completely transparent – any media files played with the PlayMedia command will detect and support the version of Media Player available. Previously, if the user had Media Player 9, PlayMedia would not work.

Version 3 comes with an entirely new online help system, complete with context-sensitive help topics. The full users' manual is available for printing as both a Microsoft Word document and a PDF file, both located in the vox proxy\help folder.

The CD Prep add-on for Vox Proxy has been considerably enhanced to make it much easier to use, and to automatically embed the new PowerPoint 2003 Viewer files on the CD.

New commands

DisplayClock

Displays a digital clock with optional timer. Styles include clock time, countdown timer, or countup timer. You can specify an alarm sound file. Size and colors can be selected. Options are included to hold the script for a specified length of time or until the countdown or countup time is reached. The clock should be especially useful for breaks during live presentations.

Input

Provides a simple single-field input form. Allows you to ask the user for text data and assign it to a Vox Proxy variable. For example, you could ask for the user's name and then use his name in speech.

ScaleCharacters

This new command allows you to specify an "authored" screen resolution (width). When the file plays, it will scale all characters up or down to match the size at the authored resolution.

PointAt

Causes a character to point in the direction of another character or a visible window. For characters having "Pointing" animations, it will point in 45-degree increments. For others, it will use gestures (left, right, up, and down). Characters can point at other application windows as well as other characters.

CDPrep Enhancements

Quick Prep

If you have a single presentation file (rather than a menu of multiple presentations), and do not need info/author files or changes to the standard copyright notices, you can use a new feature called Quick Prep, which will prepare everything you need for the CD in a single step.

Automatic File Scanning

CD Prep will now scan through the scripts in your presentation file looking for files that you call, including media, macros, other presentations, etc. It will automatically include those files on the CD and alter the scripts if necessary to change all paths to the appropriate file locations. It also locates and includes any characters used in your scripts.

Optional Info and Author buttons

The "info" and "author" buttons displayed on the dialog box shown when a CD starts are now optional.

PowerPoint 2003 Viewer

You can specify that the CD prepared with CD Prep should always play on the PPT 2003 viewer, even if the user has PowerPoint. This provides a consistent presentation that does not depend on the version of PowerPoint that the user may have.

Other Enhancements

Folder Preferences

Preferences has a new Folder Preferences tab, allowing you to change the default folders for presentations, macros, media, and characters. One thing this allows you to do is keep files (including characters) on a network drive.

Internet-resident Files

You can now use the RunMacro or the RunSlideShow commands to run a script macro or presentation residing on the internet. For example:

```
RunMacro http://www.voxproxy.com/samples/job losers
```

The **Macro Menu** has been enhanced to allow changing the macro folder displayed. It shows the current macro folder.

:noloop option

Character animations now allow the option ":noloop". This is needed if the character has a non-looping animation whose name ends in "ing". Vox Proxy does not know whether any given character animation is looping or not, so it assumes that any animation whose name ends with "ing" is a looping animation. In order to prevent the looping animation from going on forever (hanging your script), it starts a timer that will stop character animations after a specified time. If the animation is not actually looping, it can stop your script unexpectedly. The :noloop option prevents the timer from starting. For example, many characters have animations called "StartListening" and "StopListening". These are intended for use with speech recognition and are not looping animations. To use them, after inserting the animation name into your script, you should add the :noloop option:

```
StopListening:noloop
```

There is now an **online registration** for new users. For existing users upgrading to version 3, you can access the online registration on the Script Writer's Help menu. Help.

PlayMedia

Now supports Microsoft Media Player 9. The support is completely transparent. Existing scripts will work on any version of Media Player.

PlayMedia now supports the ":nowait" option that works exactly like a character :nowait. It will start the media player before the previous character has finished its speech or animation. Media Player can take a few seconds to load in memory the first time it's called. The purpose of the :nowait option is to let the Media Player load while the previous character is doing something, thus allowing the media to start playing promptly.

The **Click** command also now supports the ":nowait" option, allowing you to start the slide animation while a character is speaking or performing an animation.

PowerPoint 2003 Viewer

Vox Proxy presentations can now be played with the new Microsoft PowerPoint 2003 Viewer. The viewer files are included in the Vox Proxy installation. CD Prep disks can be configured to bundle the viewer files on the CD.

Run Mode now turns on automatically. If you open a PowerPoint presentation file that contains Vox Proxy scripts and the Vox Proxy run mode is not already on, it will be turned on automatically.

Prevent Double Execution

If you try to execute a PPV or VPM file while PowerPoint and Vox Proxy are open, you will now get an error message asking you to close PowerPoint before re-trying. Currently, if you run a PPV, VPM, or CDPrep disk while you are in PowerPoint, it opens 2 copies of Vox Proxy simultaneously, which causes speech to play twice and also causes a shut-down problem that can lead to other errors.

The **Script Writer** has a new "Tile/Untile" button. Tiling works much like the help window in PowerPoint 2002/2003. It will tile itself on the left side of PowerPoint. Character animation pull-down lists now include a Print menu item so you can print a list of animations for that character.

The **search path** has been enhanced for media and macro files so that the presentation file location will always be in the search path. When specifying files in commands like PlayMedia and RunMacro, you can now leave the path off altogether if the files are located in EITHER their default folder (like vox proxy\media) OR wherever the presentation file is located. For example:

```
PlayMedia "entertainr.mid" ....  
RunMacro "Merlin Pearl of Wisdom"
```

Waitfor media now uses the actual media position instead of the computer clock so that if the user pauses and restarts, the script stays in synch.

[The Vox Proxy Player](#)

Purpose

The Vox Proxy Player is a set of software which allows other users to play Vox Proxy-enhanced PowerPoint presentations without having to purchase Vox Proxy itself. The Player software can be installed on any computers running on Windows-95 or newer. Remember, however, that the Player is designed for people who do not have Vox Proxy. Do not attempt to install it on a computer that already has Vox Proxy installed. If the user does not have PowerPoint, the Vox Proxy Player will run on the Microsoft PowerPoint 2003 Viewer, which is included with the VP Player.

When the Player runs, it will attempt to start Microsoft PowerPoint. If PowerPoint does not exist, it will attempt to start the Microsoft PowerPoint 2003 Viewer. The PowerPoint viewer is bundled with the Player software, but it requires Windows 98-SE or newer.

Limitations of the Player

The Player software has two limitations compared with the full Vox Proxy software:

1. **Characters.** The CD version of the Player contains all characters delivered with Vox Proxy. The downloadable version contains only the following six characters: Paul, Chuck, Merlin, Peedy, Genie, and Robby. If you use any characters other than these, you must deliver them to the end user separately. The character files are composed of the character's name with the file extension "ACS" and are located in the folder:
C:\windows\msagent\chars (for Windows 95/98/ME/XP)
Or
C:\WINNT\msagent\chars (for Windows NT and 2000)

Any needed character files (other than those delivered in the VP Player) should be saved in the corresponding folder on the end-user's computer.

2. **TTS Engines.** The Player comes only with the US English "Tru-voice" speech engine. If you intend to use any other language or TTS engine, you must install it on the end-user's computer. The appropriate files are located in your computer in the folder:
C:\program files\vox proxy\speech

Note, however, that the optional AT&T Natural Voices TTS engine cannot be distributed to others, and is not included in the Vox Proxy Player. Therefore, you should not use Natural Voices within presentations that are to be played on the Vox Proxy Player.

Installing the Player

The end user must install the VP Player on his/her computer before playing presentations enhanced with Vox Proxy. There are two sources for the VP Player software, and they are installed differently:

- Downloadable from www.voxproxy.com/downloads/vpplayer.exe
Download the file and save it anywhere on the end-user's computer; Then run the file to begin the installation
- The VP Player CD delivered with Vox Proxy.
You may duplicate this CD as needed. Insert the CD in the end-user's computer to start the installation.

Using the Player

The player may be used in any of several ways.

By running a VP Player presentation file

A VP Player presentation file is a copy of the PowerPoint file with the file extension "**PPV**" rather than "PPT" or "PPS". Windows will recognize this as a Vox Proxy-enhanced presentation and run the Player automatically. If the user does not have PowerPoint, it will launch and play in the PowerPoint Viewer. The user may run a PPV file in any of the normal Windows methods:

- Double-click on the file name in Windows Explorer
- Right-click on the file name and select "Open" or "Slide Show"
- From an attachment to an e-mail, select "Open" rather than "Save" (or save the file and then run it from Windows Explorer)
- For a PPV file saved on a web site, single-click on a link to the file. The file will download and then automatically start running.
- Double-click on a shortcut to the file.

You can create the PPV file in two ways:

1. From the Script Writer File menu, click "Save Presentation for VP Player". A duplicate copy of your presentation file will be created with the file extension "PPV"
2. In Windows Explorer, make a copy of the PPT file and rename it with a file extension of "PPV"

By opening a presentation in PowerPoint and manually starting the slide show

In this case, the user must click on the Vox Proxy menu item "Run Vox Proxy with Slide Shows" before running the slide show. This is the same as the normal procedure in the Vox Proxy software itself.

If the user has Windows 95/98/ME and is running PowerPoint 2002 (from Office XP – not Windows XP), then the Vox Proxy menu is located on Windows Start/program files. Otherwise, it is on the PowerPoint main menu.

Getting Started

When PowerPoint starts, it automatically installs the Vox Proxy menu:



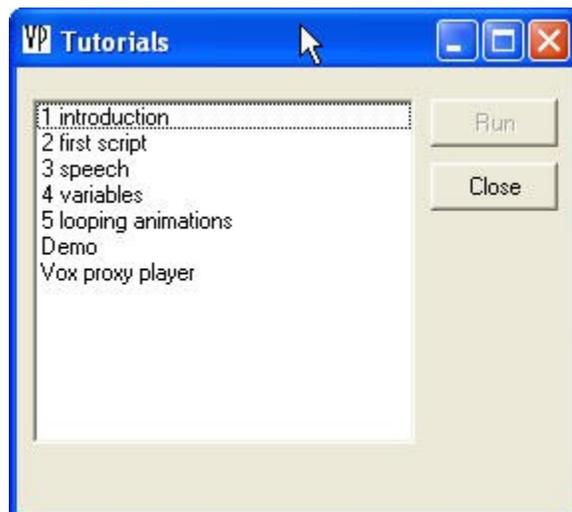
If you are running from PowerPoint-2002 or 2003 and your computer is operating with Windows-95, 98, or ME, PowerPoint requires that you start Vox Proxy from Windows Start/Programs/Vox Proxy Menu. After doing so, you will see a small "floating" menu which will appear on top of PowerPoint. This menu is exactly the same as the Vox Proxy menu above, and all of the features of Vox Proxy work in the same way.

Run the Tutorials

The simplest way to start getting familiar with Vox Proxy is to run one or more of the tutorials:

- Click on Vox Proxy
- Click on Tutorials

You will see a list of available tutorials and other presentations similar to the following:



We suggest that you run the tutorials in the following order:

- Introduction
- First Script
- Speech
- Variables
- Looping Animation

The Vox Proxy Player tutorial explains the player version of VP.

These are actually simple macros that call PowerPoint presentations located in Vox Proxy's Presentations folder.

- Select a tutorial (such as "Introduction to Vox Proxy") and click "Run" (or simply double-click your selected tutorial). This will start the presentation as a full-screen slide show.
- When the tutorial is finished, just press escape to end the slide show.
- You may interrupt a tutorial while it is running by pressing escape to end the slide show.
- You may run any tutorial directly from PowerPoint:
 - Open the tutorial in PowerPoint by browsing the folder: Program Files\Vox Proxy\Presentations
 - Turn Vox Proxy's Run Mode ON by clicking Vox Proxy|Run Vox Proxy with slide shows
 - Start the slide show

Open the Script Writer

Click on Vox Proxy|Script Writer from the PowerPoint menu. The Script Writer contains nearly all of the authoring features of Vox Proxy.

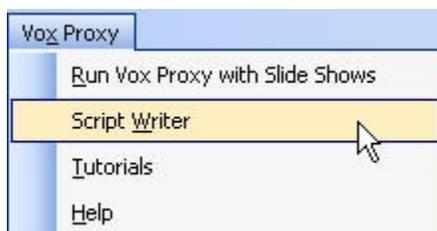
To start your first script, click on the Wizard button, then click on "Show a new character". Once you're familiar with using the Wizard, you can quickly reach all of its features, plus context-sensitive help, characters, animations, and others from the Power Scripting Menu by right-clicking anywhere in the Script Writer's edit window.

Follow the Guide to your First Script

It provides a step-by-step illustration for creating your first script in Vox Proxy.

PowerPoint menu

Vox Proxy appears on PowerPoint's main menu as follows:



- **Run Vox Proxy with Slide Shows**

This turns on Vox Proxy's "Run Mode", which will play Vox Proxy scripts when the slide show is viewed. With Run Mode off, the slide show will play without Vox Proxy scripts.

- **Script Writer**

This is the primary authoring tool for Vox Proxy scripts. It contains all of the script writing wizards, tools, and access to Script Macros as well as PowerPoint slide scripts.

- **Tutorials**

Each tutorial is simply a PowerPoint presentation wherein the characters explain the basics of how to use Vox Proxy. The tutorial files reside in the folder:

```
{program files}\vox proxy\presentations
```

As a useful exercise, you may open each tutorial and use the Script Writer to examine its scripts.

- **Help**

The online help file for Vox Proxy.

Script Basics

Scripts are simply a list of text commands describing the actions and speech that you want performed by a character. All of the commands in a script can be entered using drop-down menus from the Vox Proxy Script Wizard. Only the speech text itself must be entered from the keyboard.

Here is a simple Vox Proxy Script:

```
Show Peedy at center  
Greet  
Say I hope you enjoy the show!  
Hide
```

Scripts can be saved in three ways:

- With PowerPoint slides
- In text files called Script Macros
- Embedded in either PowerPoint presentation files or in Script Macros

PowerPoint Slide Scripts

PowerPoint scripts are kept in the presentation file itself, in what may be thought of as a "hidden" text box in each slide. The script text box can be edited only using the Script Writer. Each slide in the presentation has its own script. Characters shown from the script in a particular slide will remain on the screen until explicitly hidden, which may be in the same slide or any subsequent slide.

Script Macros

"Macros" are just Vox Proxy scripts that have been saved into text files with a file extension of **VPM** and are normally located in Vox Proxy's "macros" folder. Macro scripts are edited from the Script Writer just like PowerPoint Slide Scripts.

Embedded Scripts

An Embedded Script is actually a special kind of Script Macro that has been saved within either the PowerPoint presentation file (but NOT associated with any particular slide) or within a Script Macro file. Embedding a script macro allows you to call that macro from any slide script without the necessity of having to have access to a Script Macro file. Therefore, when you distribute a presentation to another user, you do not have to distribute the Script Macros separately. They can be included within the presentation file itself.

Speech Basics

Vox Proxy Characters speak in one of two ways:

1. **Using a SAPI4-compliant Text-To-Speech engine.**

TTS engines convert text into audible speech. Vox Proxy includes the TruVoice TTS engine distributed by Microsoft, but any SAPI4 TTS engine can be used. AT&T Natural Voices are available as an option for Vox Proxy.

The following script commands are used to produce speech from a TTS engine:

```
Say
Speak
```

A sample speech command would be:

```
Say Good morning!
```

The wizard form used to enter, test, and modify speech using TTS engines is the "Say Something" wizard, which is accessed by the "say something" button on the Script Wizard.

Speech text can also be displayed in optional Speech Balloons. Speech balloons provide a visual cue to the speech in cartoon-like bubbles displayed over the character's head. You can change the font, size, and other characteristics of speech balloons using the Balloons wizard.

Speech can be modified in several ways using Speech Tags. Speech Tags are special character strings inserted into the body of the speech text and are used to modify the pronunciation, pitch, tempo, or interpretation rules (called "Text Normalization") for the speech. Specific TTS engines may or may not support speech tags. The most commonly-used speech tag is `\emp\`, which is used to emphasize a specific word. For example:

```
Say Vox Proxy is \emp\very cool!
```

To modify the pronunciation of words or phrases, use the Pronunciation Dictionary, available on the Tools menu of the Script Writer.

Rules for pronunciation of various kinds of numbers, addresses, and abbreviations are defined in a "text normalization" document for the TTS engine. The text normalization documentation for the TruVoice TTS engine is available in the document `{program files}\vox proxy\speech\vttn.doc`.

2. **Using pre-recorded WAV files**

Vox Proxy supports a special command that allows characters to speak using your own pre-recorded voice files:

SayWav

The only file format supported is WAV. You can record your own wav files using the Windows sound recorder, found in Start/programs/accessories/entertainment. Since the character's lips synch with the sound, however, it is important that you record with a very quiet background or the character's lips will react to any background noise.

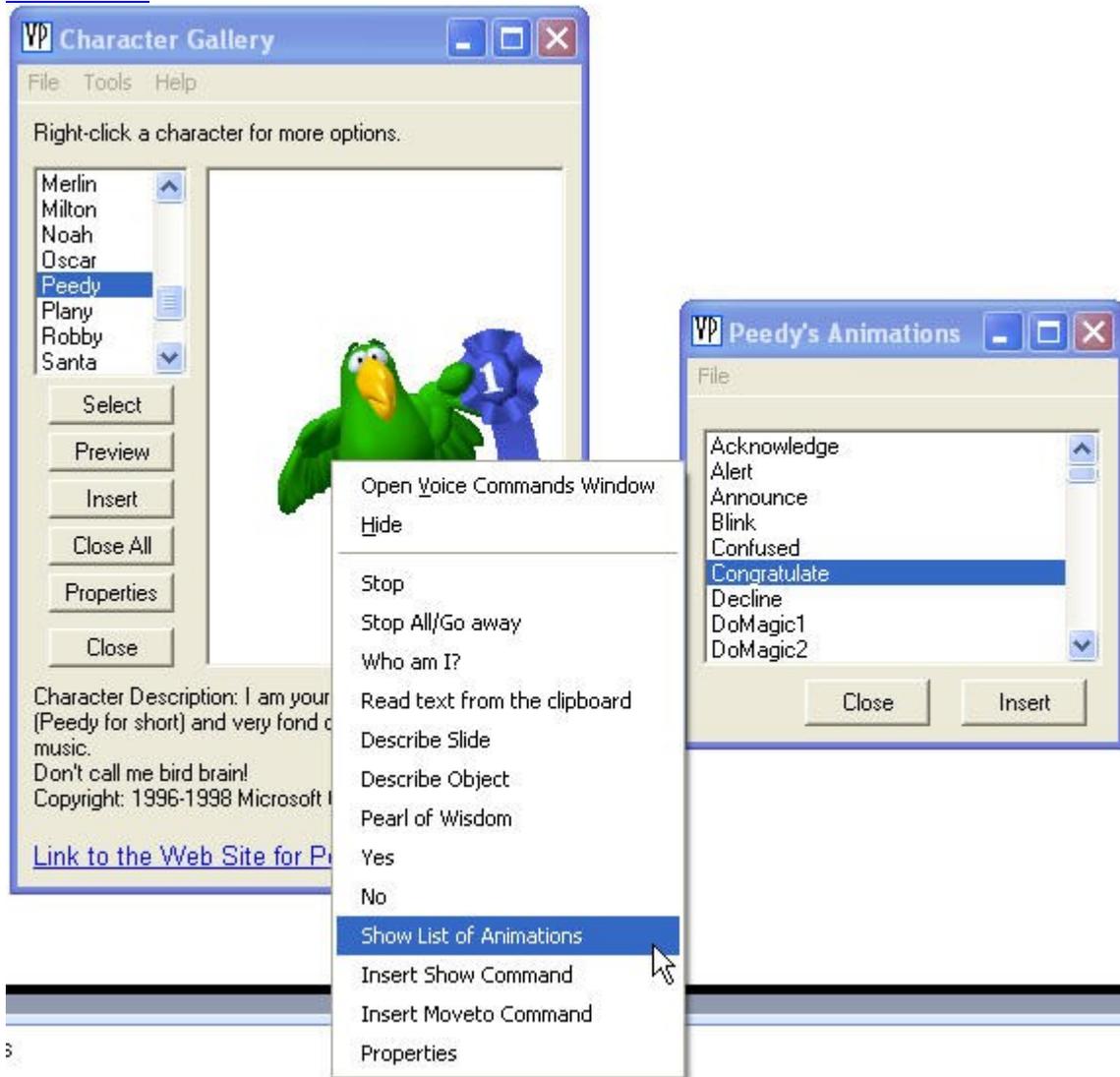
Users of AT&T Natural Voices can generate wav files directly from the AT&T TTS engine in two ways:

1. The SayWav wizard
The wizard will allow you to specify a file name and generate the file from the speech text entered.
2. Convert Speech to Wav Files feature on the Tools menu of the Script Writer. This feature will convert all speech of a specified character throughout your presentation

and will modify each Say command in your script to use the newly-created wav files. This feature is especially important for distributing disks created with CDPrep since the AT&T license does not permit distribution of the TTS engine itself.

Character Animation

Characters animations are performed from a list of pre-rendered movements. Each character has its own list of available animations, many of which are common. The animations for any character may be previewed from the Character Gallery. In addition, any time a character is visible on the screen, you may right-click on the character and select "[Show list of animations](#)" from the menu.



With many animations, a character either gestures or looks in a certain direction and then "holds" that state for a period of seconds or until another animation is performed. Most of the time, characters can speak after it has moved to its animated state. For example, with most characters, you can say:

```
GestureLeft
Say Look at this picture!
```

The character will speak while gesturing.

Characters cannot speak while they are in the process of moving during an animation. However, if you have two or more visible characters, one character can speak while the other is performing an animation. See the discussion of the `nowait` option. You can exploit this technique to speed up the pace of speech and animations in a script. [Example](#)

```
Show Paul at left center
Show Peedy at right center
Paul Say:nowait I'd like to introduce you to my friend Peedy.
```

Because of the `:nowait` option after `Say`, Paul will speak while Peedy is in the process of appearing.

Show Animations

When you **Show** a character, you are actually running a script command to show the character. When the character appears, it automatically plays its built-in "show" animation. There is, therefore, a difference between the `Show` command and the "show" animation. In the midst of a script, if you want an already-visible character to perform its show animation, you must use the "Do" command:

```
Show Peedy at center
Say Let me fly back in for you.
Do show
```

Hiding Characters

Like `Show`, there is both a `Hide` command as well as a `Hide` animation. By default, when you use the `Hide` command in a script, the character automatically plays its `Hide` animation. You can avoid the `Hide` animation by using the `Disappear` command rather than `Hide`.

Idle Animations

When characters are on the screen without speaking or moving for more than a few seconds, they default to randomly playing their "idle animations", animations whose names begin with "idle". You can change the default to turn off idle animations in preferences, and you can turn them on and off in your script using the `IdleAnimations` command.

Sound Effects

Some animations are accompanied by a sound, called "sound effects". Sound effects can also be turned off in preferences as well as in your script using the `SoundEffects` command.

Wait Options

Normally, animations will not play until the previous character's speech or animation is complete. The `:nowait` option allows the command to be acted on without waiting for the previous character. For example, to allow Peedy and Merlin to hide at the same time:

```
Merlin Hide
Peedy Hide:nowait
```

Also see wait options in the Reference/General section.

Looping Animations

Some animations are designed to run continuously in a loop until explicitly stopped. These are called "looping animations". Their name normally ends in "ing", like "searching". Such animations must be handled with care because, if not stopped properly, they can run forever and effectively "hang" your script. Vox Proxy includes a built-in timer that is invoked automatically when you play any animation whose name ends in "ing". When the timer times out, it stops whatever the character is doing, allowing the script to continue. Since Vox Proxy assumes that any animation ending with "ing" is looping, you must take special care for these special cases:

1. Animations whose names end in "ing" but are not looping. In your script, follow the animation name with the option :noloop. For example:

```
Show Peedy at center
Listening:noloop
```

2. Looping animations whose names do not end in "ing". In your script, follow the animation name with the option :loop. For example:

```
Show Chuck at center
Hearing_1:loop
```

Random Animations

You may specify a list of animations, one of which will be randomly selected to play. This can be useful on a menu slide, for example, that is returned to often in a presentation, so that the gestures are not always the same. Separate each animation in the list with the "style" character, or vertical bar ("|"). For example:

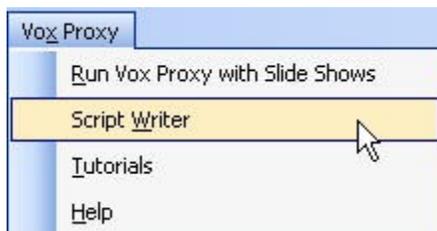
```
Show Paul at center
explain | explain | explain2 | explain3 | crossarms | handsonhips
```

[Guide to your First Script](#)

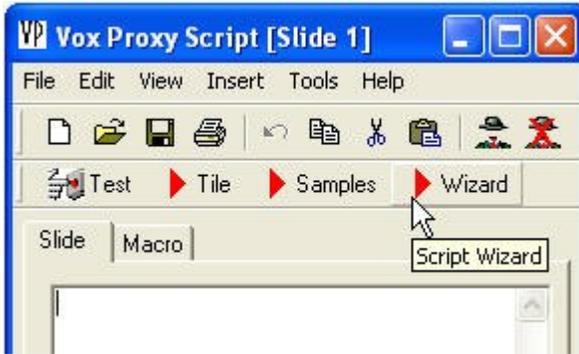
Creating A Slide Script

The Script Writer is the place where you create and edit slide scripts. Follow these simple steps to create your first script:

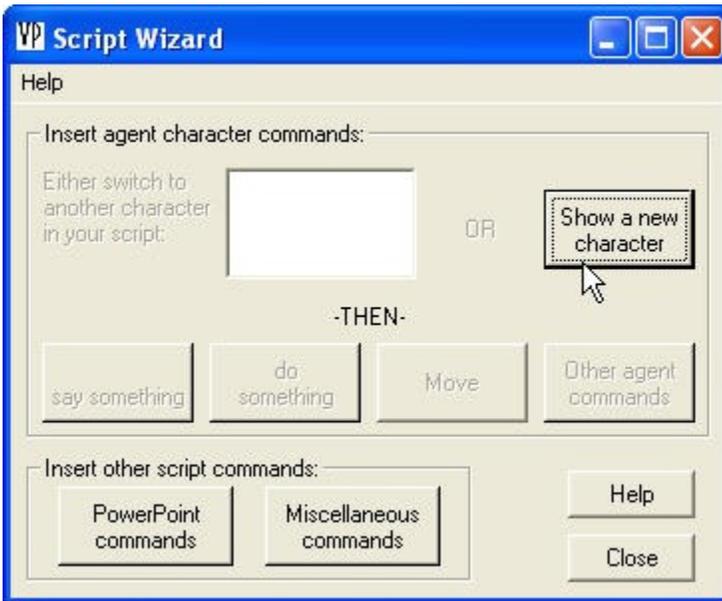
1. Open a new presentation in PowerPoint or open an existing presentation file and navigate to the slide you wish to create the script on. This is important. If you start creating a script without a presentation file open, you will lose that script when you open the file.
2. Open the Vox Proxy Script Writer from the PowerPoint menu:



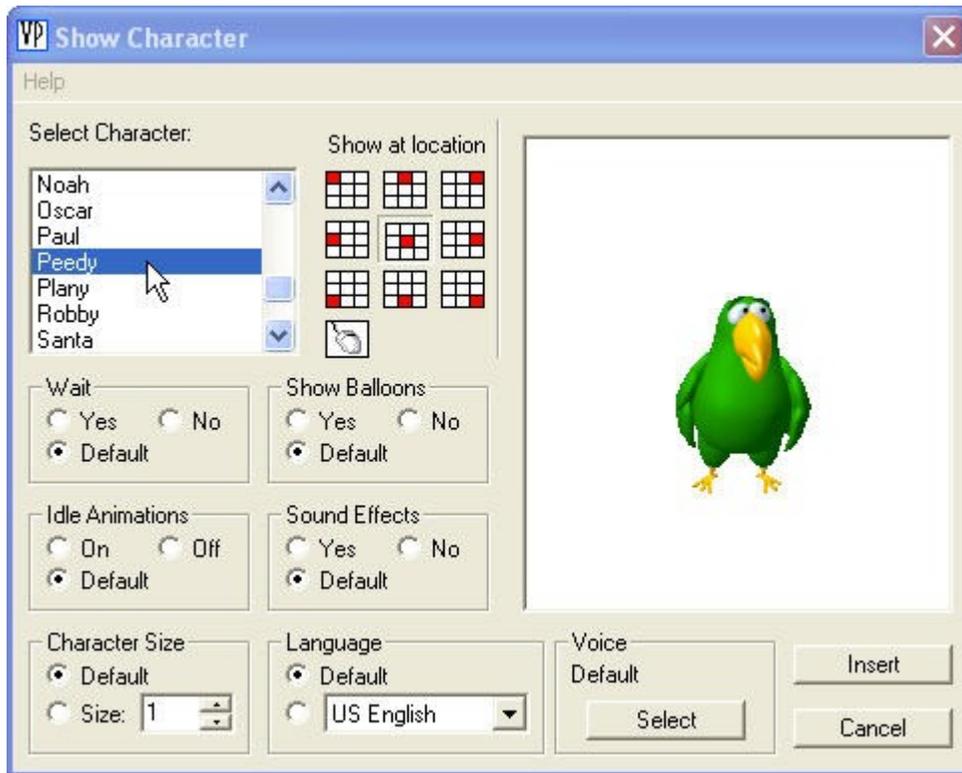
3. On the Script Wizard form, click on the Wizard button:



4. On the Wizard form, click on Show a New Character:



5. On the Show Character form, click on Peedy:



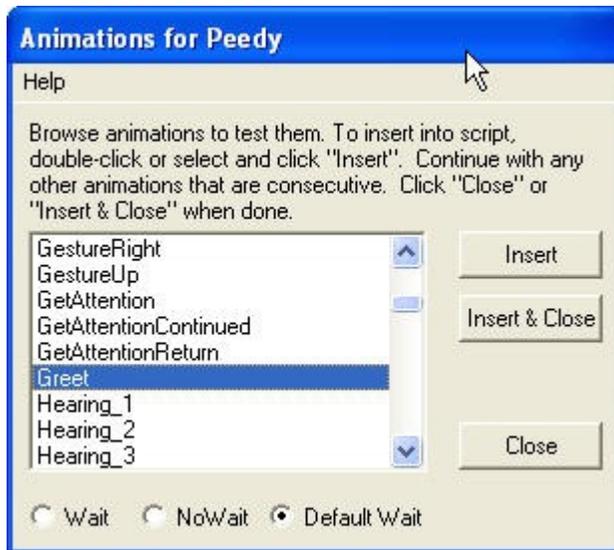
6. Click the **Insert** button.
For this illustration, we are leaving all of the defaults alone. For example, if you don't want to display speech balloons when the character talks, just check the No option under Speech Balloons on the Wizard form above before clicking Insert.

To show the character at a different location, either click one of the grid location buttons or the mouse button to drag and drop the character on the slide.

You will see the following line of script inserted into your Script Writer's edit window:



7. Now let's have Peedy take a bow. Click Wizard, then click "Peedy Do Something". Peedy will appear along with a drop-down list showing all of Peedy's animations. Notice that as you scroll through this list, Peedy will demonstrate each animation for you. Scroll to Greet, and click the Insert and Close button:



8. You will see the word Greet inserted in your script. Once you learn some of these simple animation names, you can just type them into the script rather than using the Script Wizard.



9. OK, now let's have Peedy speak. For now, we're just going to type the word Say along with the text of what you want Peedy to say. Later, you can explore the Say Something button on the Script Wizard to test modifications to the speech, such as speed, pitch, or emphasis.



10. Just to be sure the script is saved in the presentation, click on the save button (the disk icon) on the toolbar or click File, Save.
11. We're ready for a test. Click on the Test button. You will see the slide appear full-screen. Peedy will fly into the center of the screen and speak.
12. There's one more save required – since the script is embedded into the PPT file, you must also save the PowerPoint presentation.

You've done it! Your first script has been written, tested, and saved. When you're ready to run this script in your slide show, you must turn on Vox Proxy's Run Mode. Do this by clicking Vox Proxy, Run Vox Proxy With Slide Shows. If you forget this step, your slide show will run without the character scripts.

As a time-saving tip for entering and editing scripts, you can right-click in the Script Writer's edit window to open the Power Scripting Menu. This can substitute for the Script Wizard. All script commands are available from a cascading menu, and selecting a command will display the corresponding wizard.

From the Power Scripting Menu, you can also insert character Show commands, select animations for the current character, insert speech tags, display help for the current script command, and test a line of speech. This menu is context sensitive, so if you are editing a script, be sure to click at the location you're editing in the script.

Sample Scripts

Simple script with two characters:

```
show Merlin at Left Center balloons=on
Greet
say Peedy, where are you?
show peedy at Center
say Hi, Merlin. Got any good crackers?
moveto Bottom Right
say None here.
moveto Top Left
say Here we are!
Idle2_2
say Not bad. See you, Merlin!
Hide
```

Script with PowerPoint commands:

```
Show Paul at Left Center balloons=off size=1.5
Greet
Say This &tod we'll be talking about our three goals for the next quarter.
Point45
Click
RestPose
Say First, blah
Gesture Left
Click
Explain
Say Second, blah, blah
Point135
Click
RestPose
Say and, third, blah, blah, blah
NextSlide
```

[Program Updates and Support](#)

Support is available via E-mail at support@voxproxy.com

Program updates, tutorials, script macros, FAQs, and other files are available in the support pages of our web site:

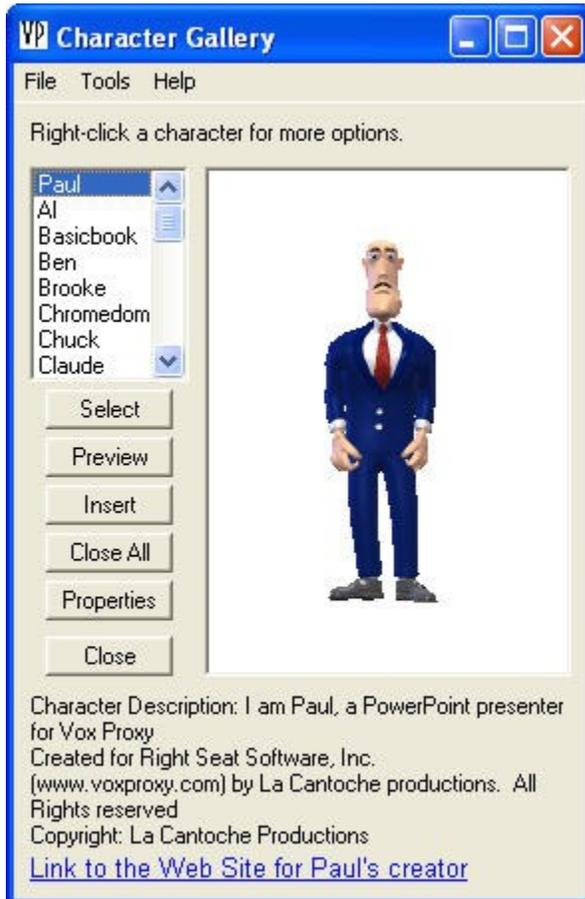
www.voxproxy.com/support.htm

Vox Proxy's author, Tom Atkins, can be reached at:
Right Seat Software, Inc.
1110 12th Street
Golden, CO 80401-1118
303-278-2244

Character Gallery

Character Gallery Form

The character gallery form appears similar to the following. Click on hot areas of the form for more information.



Functions that you can perform on the Character Gallery include:

Selecting a Character

Preview Character Animations

Inserting "Show Character" Command

Removing a Character

Placing a Character on a Slide

Downloading Characters from the WEB

Close All

Press the "Close All" button to close all visible characters. Any time characters are visible, there are two other ways to close them:

- Press the "Close Characters" button on the Script Writer.
- Right-click on a character and select "Stop Playing". If a script is playing, it will stop, and all characters will be closed.

[Downloading Characters](#)

Many Microsoft Agent characters are available on the WEB. Once you download and install the character's file, it will be displayed in the Character Gallery. For agent characters included with Vox Proxy, the Character Gallery contains links to the characters' creators. Our web site contains a list of available download sites.

If you would like referrals for creating your own agent characters, contact us by e-mail: support@voxproxy.com

[Inserting a Show Command](#)

If you are creating or editing a script and the Script Writer is visible, you may insert a command into your script at this time by pressing the "Insert" button on the gallery form. The Insert button will display the Show character wizard form. A Show [character] command will be inserted into your script at the current cursor location.

[Placing Characters](#)

If you are building a script and have a Vox Proxy Script Writer Edit Window open for either a macro or a PowerPoint slide, the "Character Gallery" form can be a valuable way to locate the character on the screen and insert "Show" or "Moveto" commands at the appropriate locations in your script. Follow these steps:

- First, be sure that the cursor in your script text field is at the position where you want the Show or Moveto command inserted.
- Select the Character Gallery, either from the Script Writer toolbar or View menu or from the Vox Proxy menu in PowerPoint.
- Display one or more characters.
- While displaying the slide you're working on, start a full-screen slide show in PowerPoint (press PowerPoint's Slide Show view button).
- If you want to see the character in a particular animation (pointing or gesturing, for example), so that, when you position him on the slide, he will be pointing precisely where you want, click on the "Preview" button. You may also right click on any visible character, and choose "Animations" from the character's menu.
- Drag the character wherever you want.
- Right click on the character and select "Insert Moveto" or "Insert Show". An appropriate command will be inserted into your script text.
- Repeat the last two steps as needed, each time relocating the cursor to the appropriate place in your script.
-

Previewing Animations

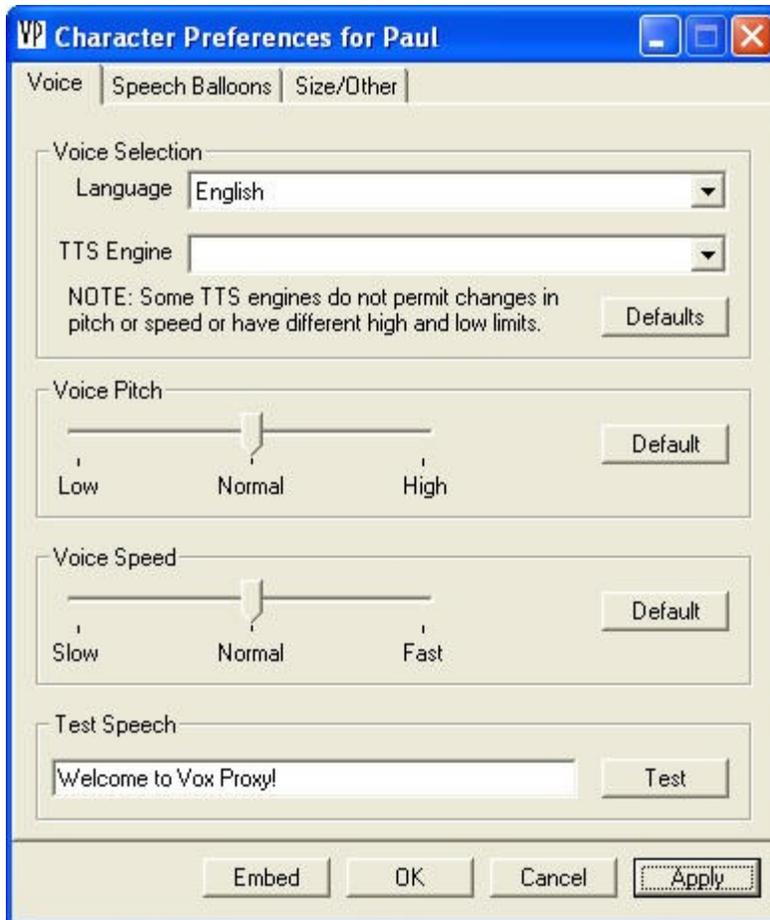
While a character is displayed in the gallery window, press the "Preview" button. A list of animations available for that character will appear, similar to the following:



As you scroll through each animation in the list, the character will demonstrate it. If you are creating or editing a script and the Script Writer is visible, you may insert any selected animation commands at this time by pressing the "Insert" button.

Properties

Press the "Properties" button to display the character property form. Properties that are set in this form become defaults for the specified character.



Character defaults include:

- Voice Defaults
- Language
- TTS Engine
- Pitch (NOTE: AT&T Natural Voices do not support pitch changes.
- Voice Speed
- Speech Balloons
- Show balloons (on or off)
- Balloon size
- Font
- Font size
- Other
- Character size
- Idle animations on or off
- Sound effects on or off

You can embed a character's properties into your presentation (PPT) file by clicking on the embed button. In this way, if you copy the presentation file and run it on another computer, the character properties will be preserved.

These properties are saved in a special file along with other system preferences. The file is: C:\program files\vox proxy\voxproxy.dat

If you copy this file to another computer that has Vox Proxy or the Vox Proxy Player installed on it, it will overlay all preferences and character properties on the other computer.

Removing a Character

You may remove a character from the screen in any of several ways:

1. If the Script Writer is visible, you can remove all characters from the screen by clicking on the "Close Agent Characters button" located on the standard toolbar.
2. "Recapture" the character either by clicking on the character's name or double-clicking on the character itself. This character will disappear automatically when you either close the form or select another character.
3. Right click on the character itself and select "Hide". The character will remain active but will disappear from the screen. A symbol will appear in your Windows System Tray. To re-display the character, double click on the character's symbol in the system tray.
4. If your computer is equipped with a microphone and you have installed Vox Proxy's voice recognition software, press the Scroll Lock key, wait until you see the message "...is listening", then say the word "Hide"
- 5.

Selecting characters

All available characters are listed by name. To show one of those characters, click on the character's name. The character will be shown in the display panel on the right. In order to keep this character on the screen, press the "Select" button. The character will move off the form and will remain on the screen, even after the Gallery form is closed. In order to "recapture" the character into the Gallery form, either click on the character's name or double-click on the character itself.

Character Images

Vox Proxy Documentation



* Characters available at extra cost

Macros

Introduction

Script Macros are Vox Proxy scripts that are not associated with a PowerPoint slide. A Macro script can be saved in one of three places:

1. In a Macro File
Macro Files are text files with the file extension "VPM" (Vox Proxy Macro). They can be run directly from Windows, outside of PowerPoint, by opening the vpm file from Windows Explorer or from a shortcut to the file.
2. Embedded in a PowerPoint presentation file
Macros that are embedded in a PowerPoint presentation can be run anywhere within the presentation's slide scripts using the RunMacro command. In order to run an embedded macro from a slide script, do not specify a path in the RunMacro command.
3. Embedded in a Macro File
Macro Files themselves can contain other embedded macros. In this way, the script in a Macro File can run other embedded macros using the RunMacro command.

Example

```
Show Merlin at center  
RandomLine "pearls"
```

The macro "pearls" can be embedded in this Macro File, making it self-contained. This technique can be used for jokes, sayings, or other expressions from which you want to play a single, randomly-selected line.

Pearls might contain:

```
say I've gradually risen from lower-class background to lower-class  
foreground. Marvin Cohen  
say Man can climb to the highest summits, but he cannot dwell there long.  
George Bernard Shaw  
say Man is the only creature that strives to surpass himself, and yearns  
for the impossible. Eric Hoffer  
say Be nice to people on your way up, because you'll meet them on your  
way down. Wilson Mizner  
...etc
```

In order to investigate Script Macros, open the Macro Menu by clicking on the Macro tab of the Script Writer and look at the sample Macro Files included with Vox Proxy. Test them from the Macro Menu by clicking on the Run button. Test them from Windows by exiting from PowerPoint, browsing to the folder:

{program files}\vox proxy\macros

and opening one of the sample Macro Files such as:
Merlin pearl of wisdom

The Macro Menu

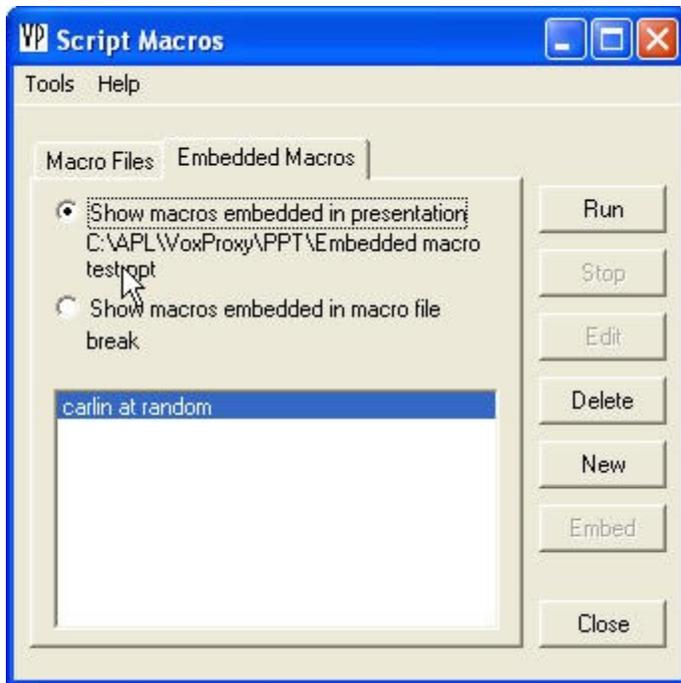
The Macro Menu is comprised of two tabs, or pages. Page 1 shows a list of Script Macro Files in a selected folder:



Buttons on the Macro Menu allow you to:

- Run the selected macro
- Stop the macro once it is running
- Delete the selected macro file
- Create a new macro
- Embed the macro script into the active presentation file or into a macro file
- Select alternate macro folders

Page 2 of the Macro Menu allows you to work with **embedded macros**:



On this page, you can display macro scripts that are:

- Embedded in the active presentation file
- Embedded in a macro file

[Running macros](#)

Running a Macro File

Macro Files may be run from the macro menu in any of three ways:

1. Single-click on the macro name, then click on the "Run" button
2. Double-click on the selected macro name
3. Click the "Test" button on the Script Writer

From Windows, with PowerPoint (and Vox Proxy) close, you can run a macro file by opening the file.

Macros can be run from any Vox Proxy script using the RunMacro command. Example

Interrupting a Running Macro

A running macro can be interrupted in several ways, depending upon what the macro has displayed:

- If the macro is running on your desktop (on top of your desktop or any applications that may be open), either click on the small "X" close button in the upper-right corner of the screen or right-click the "Script Macros" button on the task bar and select "close". The macro will be interrupted and the macro menu will be automatically restored.

- If the macro has started a PowerPoint presentation, end the presentation in the normal way (by pressing escape, for example, or by right-clicking on the slide and selecting "end show").
- If the macro is running on a "background" (other than a PowerPoint presentation), click anywhere on the background. The macro menu will be automatically restored.

Returning to the Macro Menu

When a macro is finished, you may exit the macro and return to the macro menu in any of the same ways that you would interrupt a macro (see Interrupting a macro).

Reference

Character commands

Balloons

Turn on or off the speech balloons that are displayed when a character speaks or thinks. Also allows selection of the balloon font and font size, and whether the balloons should be a fixed length or automatically sized according to the spoken text.

Syntax:

Balloons [on|off] [lines=#] [size=#] [font="name"]

Optional parameters:

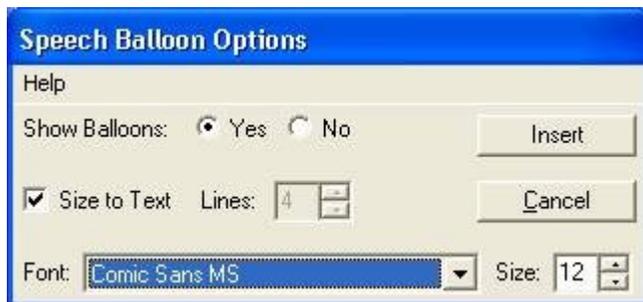
- on or off [default=on]
- lines=# [default=auto-sized]
- specify a fixed number of lines for the balloon size. If this is not specified, the balloons will be auto-sized according to the text displayed.
- size=#
Specify the text size in points
- font="name"
Specify the name of the font

Examples:

```
Balloons on size=14 font="arial" lines=5  
Balloons off
```

Wizard:

Other Agent Commands



Disappear

Make the currently active character disappear. This command closes the character and does not play the character's "Hide" animation. This may be useful if the character's "Hide" animation is lengthy or plays unwanted sounds.

Do

The active character performs an animation. This command is not normally necessary because the default command is to perform a character animation by that name. The optional :wait or :nowait will over-ride the default behavior.

The following commands are equivalent:

```
do Greet
Merlin Greet
Greet
```

The "style" character, or vertical bar ("|") is used to provide a list of alternative animations to be played. The program will randomly choose to play one of the animations. In the list This may be useful in a slide or macro which is repeated often to provide some variety in the character's actions.

Syntax:

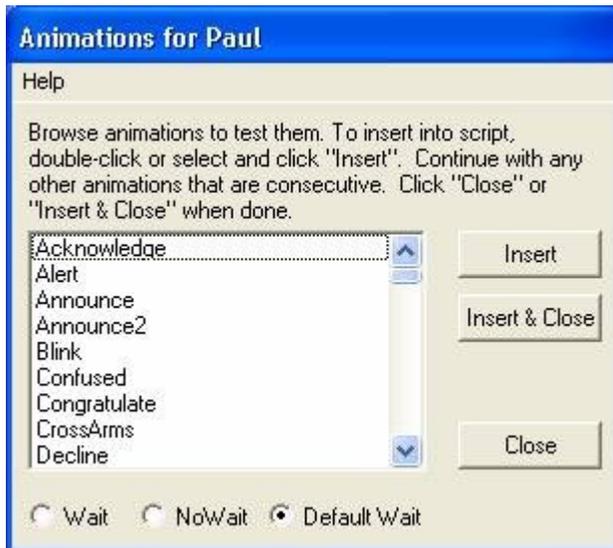
Do animation

Examples:

```
Do Greet
Do Explain
Greet | Explain | Announce
```

Wizard:

Do Something



GetstureAt

Causes the active character to gesture toward another visible character on the screen. The active character does not move, but merely gestures either up, down, left, or right, depending upon the selected character's location on the screen.

Syntax:

GestureAt name

Wizard:

Other Agent Commands



Hide

Hides the character. If the character name is not specified, the character currently active is hidden. This is a command that plays the character's default hide animation. If you want the character to disappear without playing the hide animation, use the disappear command.

Syntax:

Hide

Examples:

```
Show Peedy at Center
Say Nice to see you!
Hide
```

Wizard:

Other Agent Commands

IdleAnimations

Temporarily turn on or off a character's idle animations. If the word "all" appears after "on" or "off", it will apply to all characters which are visible at the time the command is encountered in the script. Otherwise, it applies only to the currently active character. This over-rides the default, which is a preference item. See Tools/Preferences on the Script Writer and also see help for the "SetPrefs" command.

This command is especially useful for animations such as "SitLeft" and "SitRight" to ensure that the character remains seated. If idle animations are on, after a few seconds, the character will stand and begin its normal idle animations. The command will remain in effect only until the character is closed.

Syntax:

IdleAnimations on|off

Examples:

```
Show Paul at Center
Announce
Say Good morning...
...
IdleAnimations off
SitLeft
say How nice it is to...
...
```

Wizard:

Other Agent Commands

[Interrupt](#)

The Interrupt command signals another character which is in the process of playing a looping animation to stop playing. Note that this command must be issued by another character (not the one that is looping). See also the "timeout" preference item.

Syntax:

Interrupt (character name)

Examples:

```
Show Genie at center
Processing
Show Peedy at right center :nowait
Say:nowait What are you doing, Genie?
Interrupt Genie
Genie say ..whatever
```

Wizard:

Other Agent Commands

[Language](#)

Specify the language to be spoken by the currently active character.

'name' may be any of the following:

1. A valid language name with or without quotes
2. The numeric language code (for example, US English=1033)
3. The hexadecimal language code in format: "&&Hxxxx" (for example, US English is "&&H0409")

A TTS engine for the specified language must be installed on your system. If an appropriate TTS engine is not found, it will use the character's default speech engine.

Syntax:

Language (name) [decimal code] [hex code]

Examples:

Language French

Language 1036

Language "&&H040C"

Wizard:

Other Agent Commands

A drop-down list of languages is displayed. Select the desired language and press Insert. Note that the drop-down list contains all possible languages, not just those that are installed on your computer.



[Listen](#)

Listen for a voice command or "yes" or "no". Requires that a voice-recognition program has been installed on your system. The listening state persists only for a fixed period of time (about 10 seconds). The listening state may also be invoked on a visible or hidden character by pressing the Scroll Lock key.

To listen for specific words or phrases, see the ListenFor command

Syntax:

Listen [:wait] [:nowait]

Wizard:

Other Agent Commands



ListenFor

This command causes the current character to listen for a "voice command" - a list of words or phrases. To listen for a simple "yes" or "no", see Listen. The user speaks one of the possible commands into a microphone. The result is placed into the &Response variable, where it may be evaluated using IF or Evaluate. Items in the list are specified in quotes such as: ListenFor "First thing" "second thing" ...etc

Listenfor is used where you want to take different actions depending upon the user's selection. It is an alternative to the ListBox command. For example, you may start different slide shows or branch to different slides within the current slide show.

Each item may consist of either just the text to appear in the list, or as two items, separated by the "style" character, or vertical-bar (|):

1. The text to appear in the list
2. The text to be returned in the &Response variable

Syntax:

Balloons [on|off] [lines=#] [size=#] [fint="name"]

Examples:

```
Listenfor "First thing|one" "Second thing|two"
```

In this example, the user will see "First thing" and "Second thing". If he selects First thing, the &Response variable will be set to "one".

Wizard:

Other Agent Commands



MoveTo

Move the character to a different position on the screen.

Location may be specified as either an X,Y value pair (horizontal, vertical), as a percent of screen (0-100), or as any logical combination of the words: Upper, Center, Lower, Left, and Right. For example, "at Upper Left". The location (50 50) is the same as "Center"; (25 25) is the same as "Upper Left"; and so on.

The speed parameter S is a number equal to zero or larger. Higher numbers are slower. The default value is 1000. A speed of zero causes the character to be relocated instantaneously without playing its move animation. The fastest animated move is a value of 1. The word "atspeed" may be elided.

Syntax:

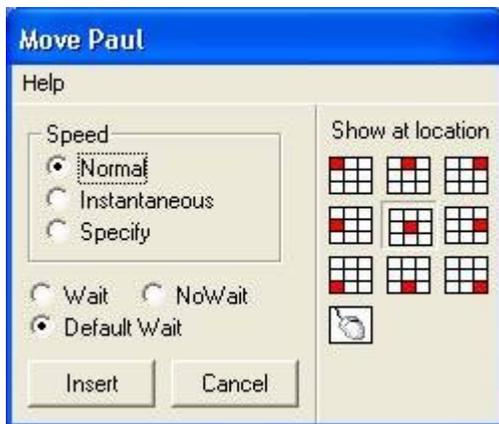
Moveto location [atspeed S]

Examples:

```
Show Peedy at Upper Left
Moveto Lower Right
Moveto 75 75
Moveto Lower Right atspeed 1
```

Wizard:

Other Agent Commands



NoBalloons

Do not display text balloons when the current character speaks.

Equivalent commands:

Balloons off

Balloons=off

Show character at location balloons=off

Wizard:

Other Agent Commands

[PointAt](#)

Point in the direction of another character. Unlike Gestures, pointing animations are made in 45-degree increments, but are supported only by characters that are known to support pointing animations. At the time of this writing, the characters are Paul, Chuck, Brooke, Ben, Susan, Simon, Doc, and Tom. If you use PointAt with a character that does not support pointing animations, then it will default to Gestreat.

PointAt can also be used to point in the direction of a visible window. Substitute for the name any unique portion of the window's caption text. The text is case-sensitive.

Syntax:

PointAt name

Examples:

1. Point at another character:

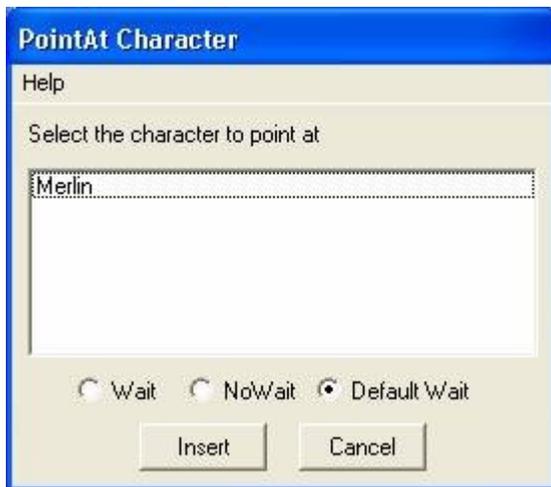
```
Show Merlin at Bottom Right
Show Paul at Center
PointAt Merlin
Paul points to his lower left (135 degrees)
```

2. Point at an Excel window:

```
Show Paul at Center
ShowExcel file=C:\apl\VPtest.xls at 75 25 size=50 50 cell=A3
PointAt Excel
```

Wizard:

Miscellaneous Commands



[Say](#)

Speak the specified text. Text may include various "speech tags" (also see Speech). All text is "filtered" by the Pronunciation Dictionary. Variable names in the text are replaced by the variable's value. The optional :wait or :nowait will over-ride the default behavior. If speech balloons are on, the text will also appear in the balloon while it is spoken. The character's name may be placed before the Say command to clarify who is speaking. If the character's name is not specified, the current active character will speak.

The "style" character, or vertical bar ("|") is used to separate alternative phrases to be spoken. The TTS will randomly choose to speak one of the phrases. This may be useful in a macro which is repeated often in scripts.

It is common to use variables within lines of speech. A common example would be the pre-defined variable "&tod" (Time Of Day). You may also use user-defined variables such as "&name".

Syntax:

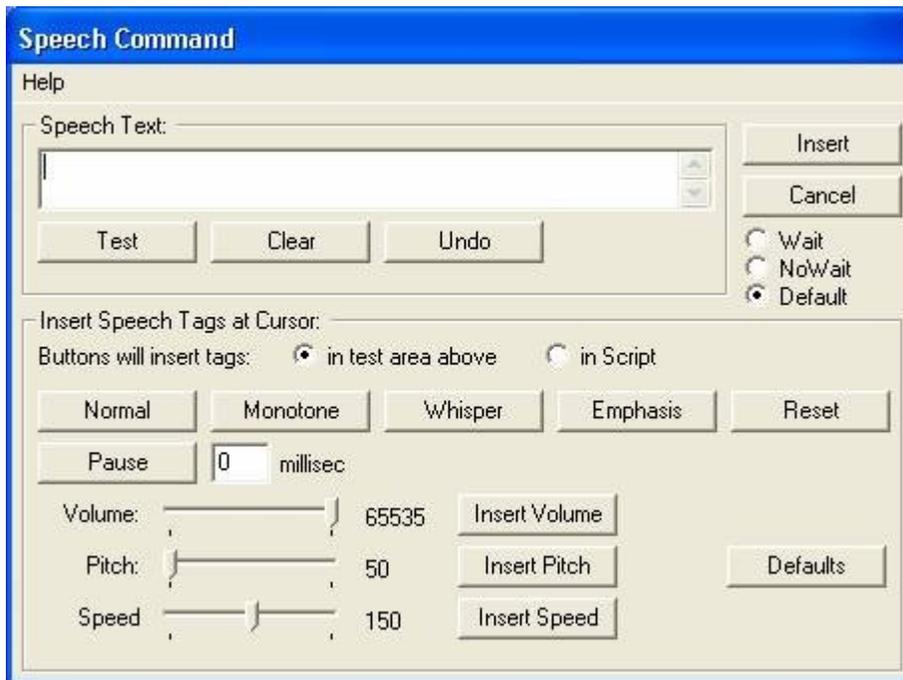
[character name] Say [:wait] [:nowait] text

Examples:

```
Show Merlin
Say Good &tod! Let me show you my cyber-magical powers!
Merlin Say Welcome to the show!
Say You can change slides now | Go ahead to the next slide | It's OK to
change slides now | Next slide | I'll wait for you to change slides
```

Wizard:

Say Something



SayWav

Speak using an audio (.wav) file. The character essentially "lip-synchs" the file. No Text-To-Speech engine is used. The "balloon text" will be displayed in the word balloon (if visible).

If speech balloons are not displayed, you may leave out the balloon text, but the quote marks must remain.

For specifying the file path, see file paths.

If you have AT&T Natural Voices installed, you can convert typed speech text into a wav file. Click in the check box; select the voice to use; enter or browse for the file name to save; press the "Create wav file" button to create the file. The Test button will play the file before you insert the command into your script.

Syntax:

WayWav "balloon text" "file"

Examples:

```
Show Merlin  
SayWav "Good Morning" "c:\my files\gm.wav"  
SayWav " " "gm.wav"
```

Wizard:

Other Agent Commands



[ScaleCharacters](#)

Vox Proxy characters are defined with a fixed pixel size. Consequently, if the slide show is played at a screen resolution different than that at which it was authored, the characters will appear smaller or larger than on the authoring screen.

Use this command to automatically scale characters' size to appear the same regardless of the screen resolution the slide show is played at.

Syntax:

ScaleCharacters H

H represents the screen height of the authoring system in pixels.

For example, if you are authoring with a screen resolution of 800x600 pixels, then you should set H to 600. If the slide show is played at a screen resolution of 1024x768, the character's size would be scaled to 768/600, or 1.28 times its default size. The character will appear to be the same height on the slide as it was during authoring.

Examples:

```
ScaleCharacters 600
```

Wizard:

Miscellaneous Commands

Show

Display a character at a specified location.

Syntax:

```
Show character_name at location [size=N] [idle=on|off] [balloons=on|off] [language=L]
[voice=V] [:wait|:nowait]
```

Options:

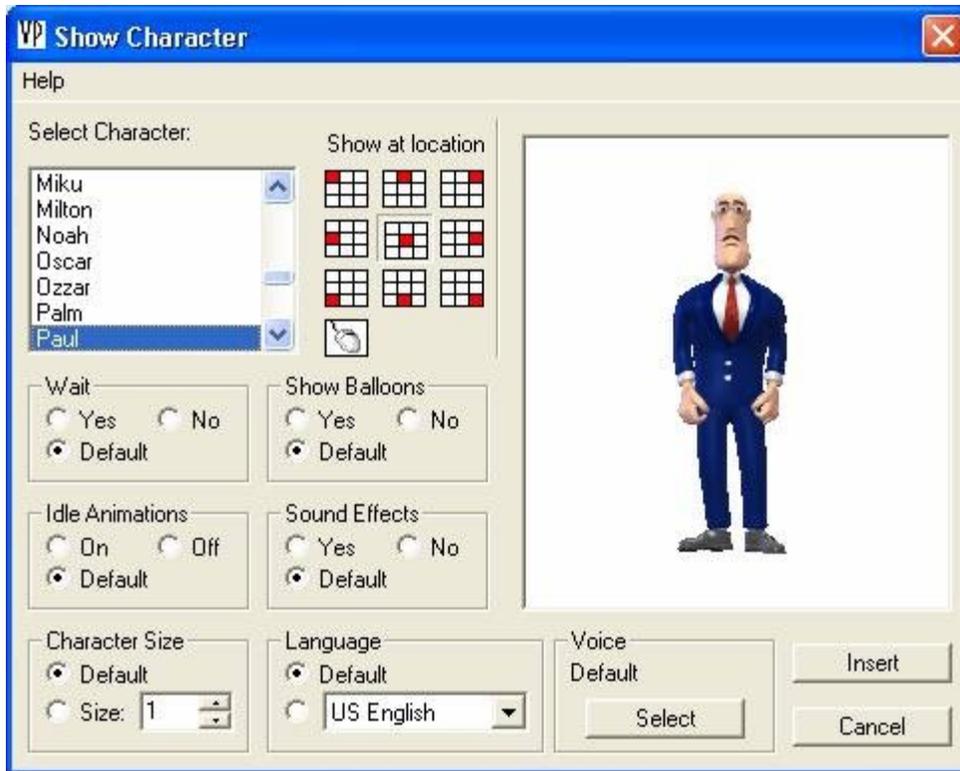
- **at location** - Location may be any combination of the words: center, upper, lower, left, or right; or two numbers for X and Y, representing percentages of the screen size. Numbers greater than 100 or less than zero are off-screen locations. Note: an X-value between 100 and 200 can be used to display on the second monitor of a dual-monitor system.
- **:wait or :nowait** Over-rides the system preference
- **size=N** size of character as a multiple of its original size. For example, size=2 means twice normal size.
- **idle=on or off** - temporarily turn on or off the character's idle animations.
- **sounds=on or off** - temporarily turn on or off the sound effects (if any) associated with the character's animations.
- **balloons=on or off** - display or don't display word balloons for speech.
- **language=L** - change the character's language to that specified. The appropriate TTS engine must be installed.
- **Voice=V** - change the character's voice. This is a simpler method to accomplish the same result as using the TTSEngine command. For a list of available voices, open the Script Wizard, Show a New Character, and click the Select button in the Voice frame.

Examples:

```
Show Merlin
Show Merlin at 50 50
Show Merlin at Lower Left
Show Paul at Center size=2 idle=on sounds=off balloons=off
Show Paul at Center Voice=Mike
Show Paul at Right Center Language=Spanish
```

Wizard:

Show a New Character



Size

Specifies the relative size of the active character. N should be a number greater than zero. A size of one (1) specifies the character's default size. NOTE: This command is functional but obsolete with Version 2. Use the size=N option in the Show command.

Syntax:

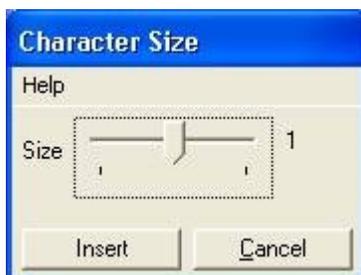
Size N

Examples:

```
Show Merlin at 50 50
Size 2
Greet
```

Wizard:

Other Agent Commands



SoundEffects

Temporarily turn on or off the sound effects which are played by some of a character's animations. If the word 'all' appears after 'on' or 'off', it will apply to all characters which are visible at the time the command is encountered in the script. Otherwise, it applies only to the currently active character. This over-rides the default, which is a preference item. See Tools/Preferences on the Script Writer and also see help for the "SetPrefs" command. Sounds may also be turned on or off as part of the 'Show' command (See the Show command).

This command is useful in some situations to avoid suppressing the speech for a subsequent character. The command will remain in effect only until the character is closed.

Syntax:

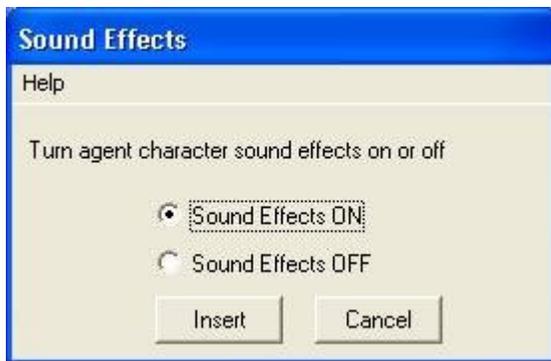
SoundEffects on|off

Examples:

```
Show Paul at Left Center
Show:nowait Merlin at right center
SoundEffects off
Congratulate
Paul say:nowait Congratulations to ....
```

Wizard:

Other Agent Commands



[Speak](#)

Same as Say except that the text is not filtered by the Pronunciation Dictionary. The optional :wait or :nowait parameter will over-ride the default behavior.

Syntax:

Speak [:wait] [:nowait] text

Examples:

```
Speak Now is the time for all good men to come to the aid of their country.
```

Wizard:

N/A

[Think](#)

Display the specified text in a "thinking" word balloon, but do not speak it. Rules for use are the same as "Say". The optional :wait or :nowait parameter will over-ride the default behavior.

Syntax:

Think [:wait] [:nowait] text

Examples:

Think Now is the time for all good men to come to the aid of their country.

Wizard:

Other Agent Commands



TTSEngine

Select a specific speech engine for the selected character. The {clsid} is a formatted string defined by the speech vendor that uniquely identifies the engine's mode. A list of available speech engines detected on your computer is shown in the wizard form.

A TTS engine for the specified language must be installed on your system. If an appropriate TTS engine is not found, it will use the character's default speech engine.

Syntax:

TTSEngine "{clsid}"

Examples:

TTSEngine "{33f7d612-a2ea-4d6e-8585-112AF0C4491A}"

Wizard:

Other Agent Commands



Wait

Causes the script to wait at this point until the active character has completed whatever it is doing. This can be useful to signal when another character should begin its actions.

Syntax:

Wait

Examples:

```
Paul say Good morning, etc, etc...
Wait
Say I'd like to introduce my friend Merlin.
Show:nowait Merlin at Center
```

In this example, Merlin will begin to appear just as Paul starts to introduce him (and not before). This helps to speed up the activity by simultaneously showing Merlin while Paul is introducing him, but not while Paul is making his previous remarks.

Wizard:

N/A

WaitCommands

Changes the list of commands for which a character will wait before performing. Available commands are: Hide, Moveto, Do or Play, Say, Speak, Show, Think, GestureAt, and Listen. If you leave off 'exclude' or 'include', the list should be inclusive. This changes the system Preferences in the file and will persist until they are changed back.

Syntax:

WaitCommands [exclude] list

Examples:

```
WaitCommands Say Do Think Listen
WaitCommands exclude Moveto Show
WaitCommands include Moveto Show
```

Wizard:

Other Agent Commands



General

[File References in Vox Proxy](#)

Several Vox Proxy commands require referencing a file. The syntax of each command is important, so please refer to the specific command for the correct syntax. Directory paths, however, are consistent for all commands. Paths can be **implied** or **expressed**.

Implied paths

If a file name is specified without a path, then the path is implied. Depending upon the file type, Vox Proxy will automatically look through a series of folders called "search paths" to find the file before it gives up. See below for the search path of each file type recognized by Vox Proxy.

Expressed paths

Expressed file paths may be specified either explicitly or with a variable.

An example of a file with an explicit path is:

```
C:\program files\vox proxy\media\drumroll.wav
```

The same file expressed with a variable could be:

```
&mediadir\drumroll.wav
```

See below for a list of variables containing paths that are always available in Vox Proxy.

Search Paths

For each file type recognized, the search paths are as follows.

PowerPoint Presentation Files:

1. The Vox Proxy presentation folder (c:\program files\vox proxy\presentations)

2. The Vox Proxy application folder (c:\program files\vox proxy)
3. The folder from which Vox Proxy was started

Media Files (audio, video)

1. The Vox Proxy media folder (c:\program files\vox proxy\media)
2. The Vox Proxy presentation folder (c:\program files\vox proxy\presentations)
3. The Vox Proxy application folder (c:\program files\vox proxy)
4. The folder from which Vox Proxy was started

Script Macro Files

1. The Vox Proxy macro folder (c:\program files\vox proxy\macros)
2. The Vox Proxy presentation folder (c:\program files\vox proxy\presentations)
3. The Vox Proxy application folder (c:\program files\vox proxy)
4. The folder from which Vox Proxy was started

Variables

Variables containing search paths include:

&appdir	c:\program files\vox proxy
&chardir	c:\windows\msagent\chars
&DiskDrive	[current default disk drive]
¯odir	c:\program files\vox proxy\macros
&mediadir	c:\program files\vox proxy\media
&presdir	c:\program files\vox proxy\presentations
&tempdir	[windows temp folder]
&windir	[windows folder]

General Rules for Scripts

- All commands and character names are NOT case-sensitive. Say, SAY, and say are all equivalent.
- Before a character can speak or play an animation, it must first be visible. Hidden characters will move, but not speak.
- Before a character takes any action, it automatically waits for any previous character to complete its action. You may over-ride this default using the ":nowait" parameter.
- A typical script to show a character, perform an animation, and say something, would look like this:

```
Show Peedy
Greet
Say Hi. My name is Peedy.
```

- Screen location is specified in one of two ways:
 1. Two numbers from 0 to100 (X and Y values), representing the relative horizontal and vertical positions of the character on the screen.
 2. Any logical combination of the words: Upper, Center, Lower, Left, and Right. For example, "at Upper Left".
The location 50 50 is the same as "Center"; 25 25 is the same as "Upper Left"; and so on.
To move the character after it is shown, use "Moveto" For example:

```
Moveto Lower Right
```

- Comments – notes to yourself which are ignored by the script - may be included in scripts by preceding the comment with a semicolon. Any text following a semicolon is ignored except where a semicolon may appear in lines of speech. Comments may not appear in lines of speech (Say, Speak, or Think commands).
- With two exceptions, script commands must always be the first word on a line of script. The first exception is where the command follows a character name. For example:

```
Merlin say Hello
Merlin Moveto 50 50
Merlin Greet
```

The second exception is for animations. Multiple animations may appear on the same line. For example:

```
Greet Pleased GestureLeft
```

- Multiple animations may appear on the same line. Commas separating animation names are optional, but at least one space is required. For example:

```
Greet, Pleased, Announce
Or: Greet Pleased Announce
```

- Use the ":nowait" parameter to force simultaneous animations among multiple characters. Also see Wait Options. This parameter may be used on the following commands: Do, GestureAt, Hide, Listen, MoveTo, Say, Show, Speak, and Think. Enter the parameter immediately after the command. For example:

```
Hide:nowait
Moveto:nowait Lower Right
Say:nowait What are you doing?
```

Speech Tags

Speech Tags are special character strings inserted into the body of the speech text and are used to modify the pronunciation, pitch, tempo, or interpretation rules (called "Text Normalization") for the speech. Specific TTS engines may or may not support speech tags. The TruVoice TTS engine included with Vox Proxy supports the following speech tags:

TAG	Syntax	Var	Description	Example
Emphasis	\emp\	&e	Emphasize a specific word	Say Vox Proxy is \emp\cool!
Monotone	\Chr="Monotone"\	&m	Monotone	Say \Chr="Monotone" I am a robot.
Whisper	\Chr="Whisper"\	&w	Whisper	Say \Chr="whisper" I'll tell you a secret!
Normal	\Chr="Normal"\	&n	Normal	Say \Chr="whisper" I can whisper\Chr="Normal" and speak normally in the same sentence!
Reset	\Rst\	&r	Reset to defaults	Say \Chr="whisper" This is a secret!\Rst\ But you didn't hear it from me!
Pause	\Pau=N\		Pause for N millisec	Say Now \pau=50\ listen\pau=50\ to\pau=50\ me!
Volume	\Vol=N\ (default=65535)		Lower speech volume	Say This is normal \Vol=32000\but I can also speak softly.

Pitch	\Pit=N\		Set relative pitch	Say I can speak normally \Pit=150\or in a squeaky voice.
Address	\Ctx="Address"\	&ca	Special context rules for text normalization	say \Ctx="Address"\I am from Den, CO 80401.
Email	\Ctx="E-mail"\	&ce	Special context rules for text normalization	Say \Ctx="E-mail"\Visit us at www.voxproxy.com.

Special speech tags are also used to specify different text for display in the speech balloons from that which is spoken. These tags are used by the Pronunciation Dictionary.

The address and email context tags are the same when used with the TruVoice TTS engine in English, but are different with other languages. For documentation on text normalization rules for each language included with Vox Proxy, see the MS Word files in the folder: {program files}\vox proxy\speech

[Wait Options for Script Commands](#)

:nowait

Normally, script commands will not be acted upon until the previous character's speech or action is complete. The :nowait option allows the command to be acted on without waiting for the previous character. This option is normally not needed when you are working with only a single character. Its principal use is to speed up the pace of the presentation by allowing a character to perform an animation simultaneous with another character's animation or speech. For example, when you're showing two characters, the second one doesn't begin to show until the first one is finished showing. To show them both at the same time:

```
Show Paul at left center
Show:nowait Peedy at right center
```

:wait

This option applies to commands in the same way that :nowait does, but it forces a wait for the previous character. Since this is the normal default, the only reason you should ever need to use :wait is if you have changed the default setting from wait to nowait (see wait preferences)

Wait command

While :nowait solves the problem of simultaneous actions, if the previous character performs several consecutive animations or speech, the :nowait can cause the new character to act before you want it to. For this reason, you can use the Wait command by itself to hold the script at that point.

In this example, we'll have Paul do his Explain animation, say something, and return to RestPose. Then we'll have Merlin gesture right and speak. We want Merlin to gesture while Paul is running RestPose, NOT while Paul is speaking:

```
Show Merlin at right center
Show:nowait Paul at left center
Explain
Say Welcome to Vox Proxy!
Wait
RestPose
```

```
Merlin GestureRight:nowait  
Say Thank you, Paul!
```

Without the Wait (prior to RestPose), Merlin would GestureRight while Paul does Explain.

Miscellaneous commands

Background

Display either a solid color or a bitmap file as a background in a script macro. Background is normally used only with Script Macros, not PowerPoint slides. Script Macros can play on the desktop, making a background desirable under some circumstances.

Syntax:

Background color fit

Options are:

1. Color – may be either a color number or a Bitmap file name in quotes. Use the Script Wizard to select colors.
2. Fit – if a bitmap file is displayed, you may specify one of the following fit options:
 - tile
 - scale
 - unscaled

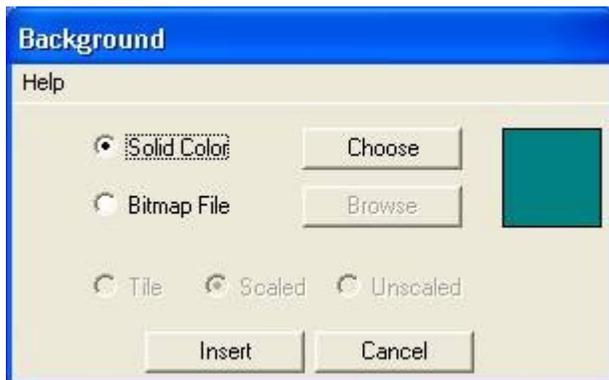
The Wizard form offers a color palette and bitmap file browsing.

Examples:

```
Background "c:\windows\clouds.bmp" scale  
Background 8421376
```

Wizard:

Miscellaneous Commands



Bitmap

Display a bitmap file on the screen. Use the CloseBitmap command to remove the bitmap.

Syntax:

Bitmap [options]

Options are:

- 1) File name and path, enclosed in quotes
- 2) AT X Y -- optional location, where X and Y are percent of screen (see Show).
- 3) Display size. One of the following: (default=1/4)
 - size=1X: as authored
 - size=2X: twice authored size
 - size=1/16: 1/16 of screen area
 - size=1/4: 1/4 of screen area
 - size=1/2: 1/2 of screen area
 - size=FS: full-screen
- 4) One of the following scaling options:
 - tile scale unscald

Example:

```
Bitmap "C:\my files\picture.bmp" at 25 25 size=1/16 scale
Bitmap "c:\program files\vox proxy\media\Clouds.bmp" at 50 50 scale size=1/4
Bitmap "Clouds.bmp" at 50 50 scale size=1/4
```

Wizard:

Miscellaneous Commands



[CloseBitmap](#)

Close the display of a bitmap file.

Syntax:

CloseBitmap "file"

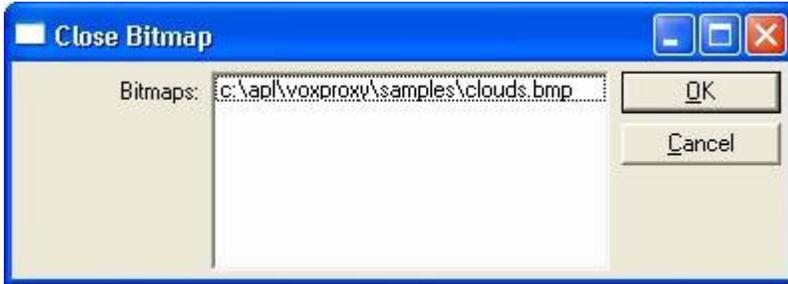
Specify the name of the bitmap file to close in quotes.

Examples:

```
CloseBitmap "c:\program files\vox proxy\media\Clouds.bmp"  
CloseBitmap "Clouds.bmp"
```

Wizard:

Miscellaneous Commands



[CloseExcel](#)

Closes the Microsoft Excel window that was opened by the ShowExcel command. Takes effect after the active character is finished whatever it is doing.

Syntax:

CloseExcel

Wizard:

Miscellaneous Commands

[CloseFlash](#)

Closes the flash movie "file" that was displayed by the PlayFlash command.

Syntax:

CloseFlash "file"

"file" should include the full path and file name. Takes effect after the active character is finished whatever it is doing.

Examples:

```
CloseFlash "c:\myfiles\flash\vp home.swf"
```

Wizard:

Miscellaneous Commands



[CloseMedia](#)

Closes a media file that was started using the PlayMedia command. Takes effect after the active character is finished whatever it is doing.

Syntax:

CloseMedia

Wizard:

Miscellaneous Commands

[CloseWord](#)

Closes the Microsoft Word window that was opened by the ShowWord command. Takes effect after the active character is finished whatever it is doing.

Syntax:

CloseWord

Wizard:

Miscellaneous Commands

[Delay](#)

Syntax:

Delay S

Where S is the number of seconds to delay

Causes the script to pause for the specified number of seconds before proceeding. This command does NOT wait for any previous character actions before running. In many cases, especially if you want the delay to start after the previous character actions, it is preferable to set a custom timer and wait for it.

Wizard:

Miscellaneous Commands

[DisplayClock](#)

Displays a digital clock showing one of the following: Time of day, a countup timer, or a countdown timer. While the clock is visible during a slide show, it may be deleted at any time by clicking anywhere on it.

Syntax:

```
DisplayClock [at X,Y] [alarm=file] [style=clock|countup|countdown] [size=N] [timer=T]
[bgcolor=C] [fgcolor=C] [hold=yes|no] action=release|delete]
```

Options:

1. **at location**, where location is an X,Y screen location for the center of the display in percent. Default is center.
2. **alarm="file"**, where file is the path and name of an audio file to be played as an alarm for either the timer or countdown options. Default is no alarm.
3. **style=clock**, countdown, or countup. If not specified, clock is assumed. For clock, the following display styles are available:
 4. clock hh:mm:ss show seconds
 5. clock hh:mm do not show seconds
 6. clock hh:mm am show am/pm
 7. clock HH:mm:ss show hours in military (24-hour) display
8. **size=N** Relative size of the display, where N is an integer number greater than or equal to 1. The default is 10.
9. **timer=T** If the style is either a countdown or countup timer, this specifies the time limit in either seconds or hr:min:sec. For clock display, the timer events will fire after this incremental time elapses from the start of the display.
10. **bgcolor=C** Specifies the background color for the display. Available colors are: red blue green black yellow cyan magenta white gray
11. **fgcolor=C** Specifies the foreground (text) color. Same color options as background. Default colors are the windows defaults (normally black on white).
12. **hold=yes or no**. Specifies whether the script should hold and wait while the clock is visible. The hold will be released either when the clock display is closed or after the "Timer" time, if action=release is specified.
13. **action=release** or delete. Specifies the action to be taken when the timer time elapses. If release is specified, you must also include hold=yes. If delete is specified, the clock display will disappear when the timer fires.

Examples:

```
DisplayClock
DisplayClock at 50 25 style=countdown size=12 timer=15:00 alarm=c:\my
music\alarm.wav bgcolor=black fgcolor=cyan hold=no
DisplayClock at 50 25 style=clock hh:mm:ss am size=12 timer=15:00
alarm=c:\my music\alarm.wav bgcolor=black fgcolor=cyan hold=yes
action=release
```

Some of the clock appearance options are illustrated below:



8:59:04



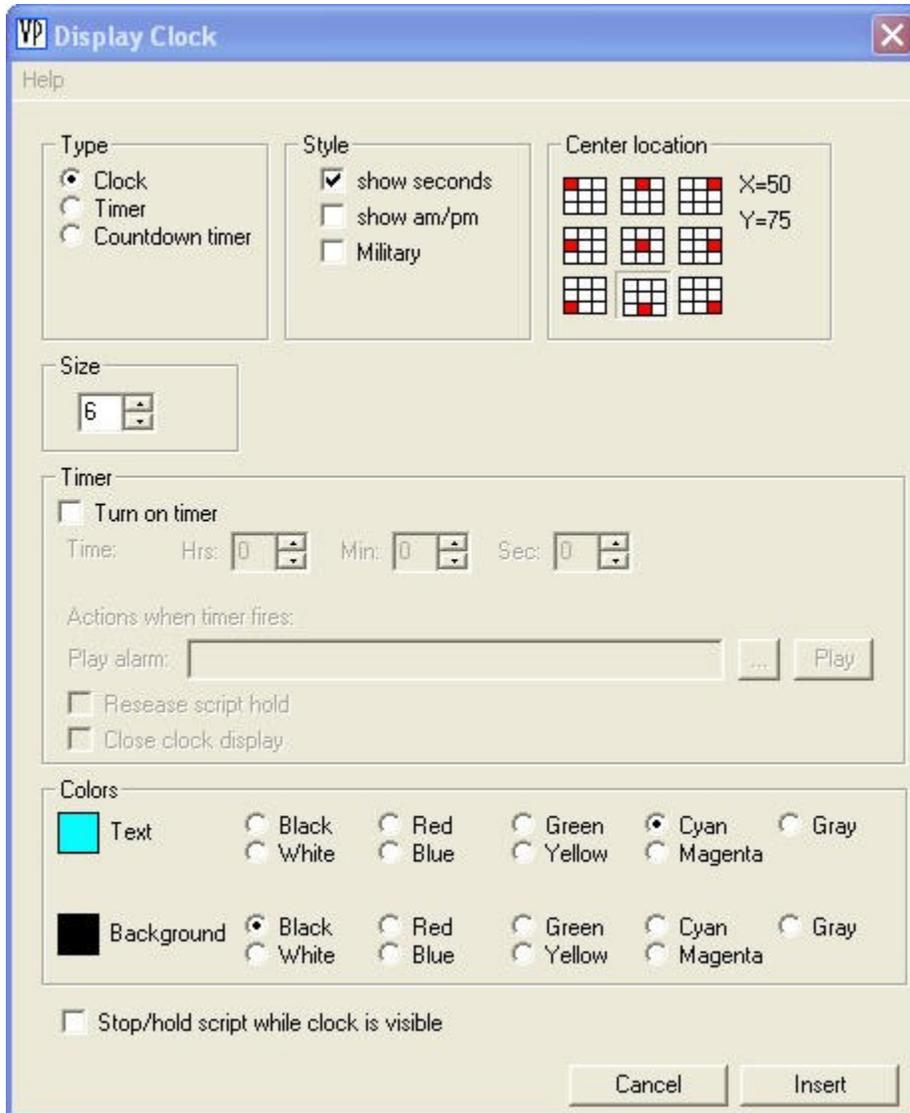
8:59



9:02am

Wizard:

Miscellaneous Commands



DualMonitor

Turns on or off dual monitor support without changing the permanent dual monitor preference item. It may be turned on and off within a script.

This command automatically offsets the X location of characters and forms by +100%. It requires that the resolution of each monitor be set the same.

Syntax:

DualMonitor on|off

Examples:

DualMonitor on

Wizard:

Miscellaneous Commands



EndMacro

End any currently-running macro. Close any characters which may be visible. If the macro was started from a menu, re-display the menu.

Do not use EndMacro in a macro which will be called from another script (either a slide script or another macro). All executing scripts will be closed by this command.

NOTE: this command is not normally necessary, since Vox Proxy automatically closes any macro which is started from a user-selected menu.

Syntax:

EndMacro

Examples:

```
Balloons on size=14 font="arial" lines=5
Balloons off
```

Wizard:

Miscellaneous Commands

GetMacro

Retrieves the script for the macro "name" and inserts it into the current script at this point. This is essentially a "cut and paste" which takes place just before the current script begins to run.

Note that this is functionally different than a "RunMacro" command, which actually runs the specified macro "on the fly" when it encounters the command during execution of the current script.

The specified macro name is not case sensitive.

Syntax:

GetMacro "name"

Examples:

```
GetMacro "Merlin pearl of wisdom"
```

Wizard:

Input

Displays an input form containing a single data field and captures input from the user. The captured data is placed into the variable specified.

Syntax:

Input [where=X,Y] [prompt=text] [instructions=text] [caption=text] [variable=&varname]

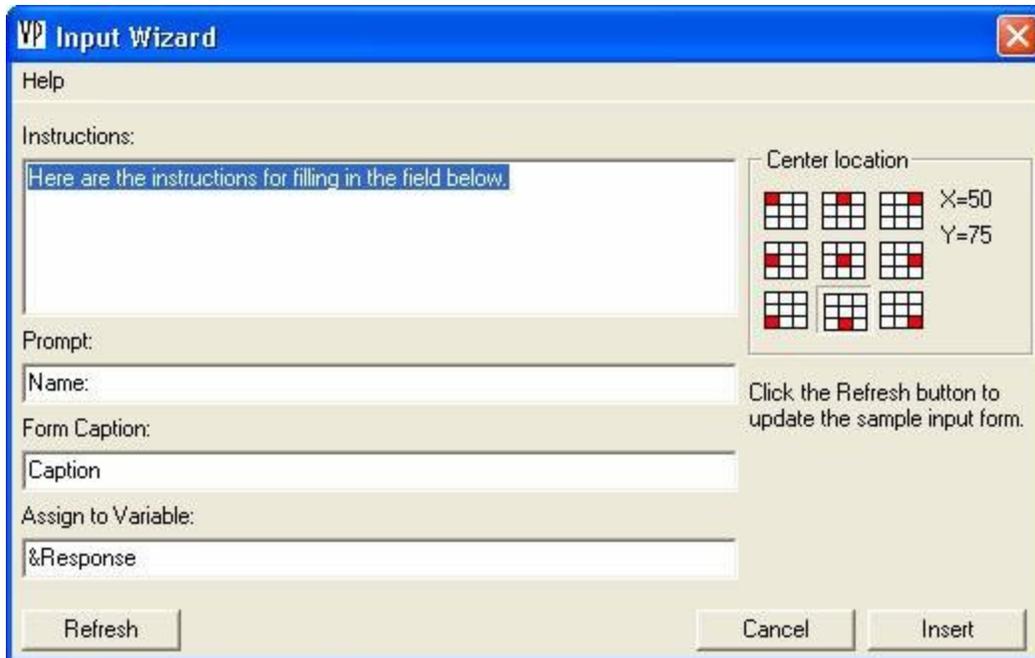
Options:

1. **where=X Y**
Location, specified as an X,Y screen location for the center of the display in percent.
Default is center.
EXAMPLE: where=50 75
2. **instructions=text**
Text to be displayed above the input field as instructions. Default is none. Instructions may be multi-line. Line breaks are entered as the "stile" character (vertical bar): |
3. **prompt=text**
Text to be displayed to the left of the input field as a prompt.
4. **caption=text**
Text to be displayed as the caption of the input form.
5. **variable=&whatever**
Name of variable to which result will be assigned. Variable name must begin with and ampersand.
Default: &Response

Examples:

```
Show Paul at center
Say Please enter your first name in the box below and click OK.
Input prompt=First Name: caption=Enter name instructions=Enter your first
name in the field below. variable=&FirstName
Say Good morning, &FirstName!
```

Wizard:



Listbox

This command displays a list box containing a list of words or phrases. The user selects a single item by clicking on it. The result is placed into the &Response variable, where it may be evaluated using IF or Evaluate. Items in the list are specified in quotes such as:

Listbox "First thing" "second thing" ...etc

Listbox is used where you want to take different actions depending upon the user's selection. It is an alternative to the voice command ListenFor. For example, you may start different slide shows or branch to different slides within the current slide show.

Syntax:

Listbox "item text [|return]" ["item text [|return]"] ...

Each item may consist of either just the text to appear in the list, or as two items, separated by the "style" character, or vertical-bar (|):

1. The text to appear in the list
2. The text to be returned in the &Response variable

Examples:

```
Listbox "First thing" "Second thing"
```

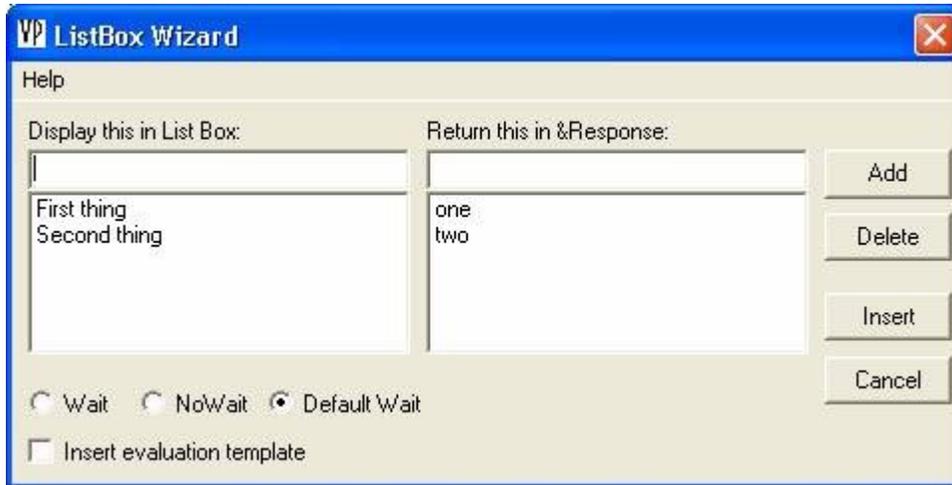
In this example, the user will see "First thing" and "Second thing" in the list box. If he selects First thing, the &Response variable will be set to "First thing".

```
Listbox "First thing|one" "Second thing|two"
```

In this example, the user will also see "First thing" and "Second thing". If he selects First thing, the &Response variable will be set to "one".

Wizard:

Miscellaneous Commands



MacroMenu

Displays a menu from which the user selects a macro to run. Wildcard is a specification for matching macro names. Note that case is significant in macro names.

Syntax:

MacroMenu [name]

Name must be enclosed in quotes. Name can include "wild cards" to select all macro names containing certain text.

Examples:

MacroMenu " "

Displays a menu containing all macros in the macro folder.

MacroMenu "Tutorial*"

Displays a menu of all macros whose name starts with "Tutorial"

MacroMenu "Tutorial**"

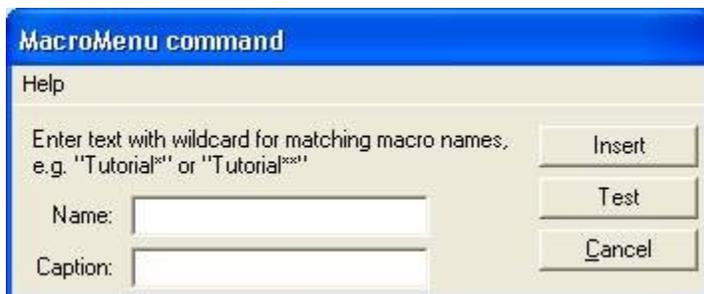
Same as above, but the macro names displayed contain only the text following the word "Tutorial"

MacroMenu "Chapter ? Tutorial"

Display matching macro names containing any character in the place of "?"

Wizard:

Miscellaneous Commands



MediaVolume

Changes the audio volume for a media file that has been started using the "PlayMedia" command. This command can be useful to lower the volume of music or other audio when a character is speaking.

Syntax:

MediaVolume level [time]

The level can vary from 0 to 100. The default volume is 100, the maximum. Volume levels below 50 are nearly inaudible.

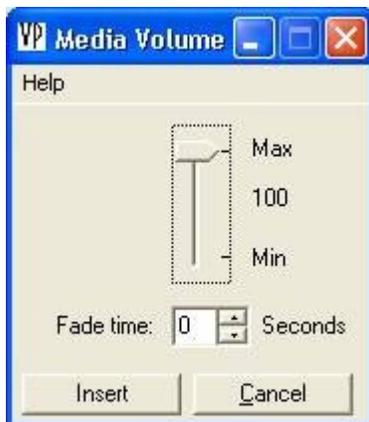
If the optional time is specified, the volume will "fade" up or down in the specified number of seconds.

Examples:

```
MediaVolume 60 2
```

Wizard:

Miscellaneous Commands



MessageBox

Display a message dialog box that requires the user to press a button.

Syntax:

MessageBox text title buttons symbol

Options:

1. **Text** to be displayed in the Message Box
2. Message Box **Title**
3. **Buttons**. Options are:
 - "OK"
 - "OC" (OK, Cancel)

- "ARI" (Abort, Retry, Ignore)
- "RC" (Retry, Cancel)
- "YN" (Yes, No)
- "YNC" (Yes, No, cancel)

4. **Symbol**

Symbols may be:

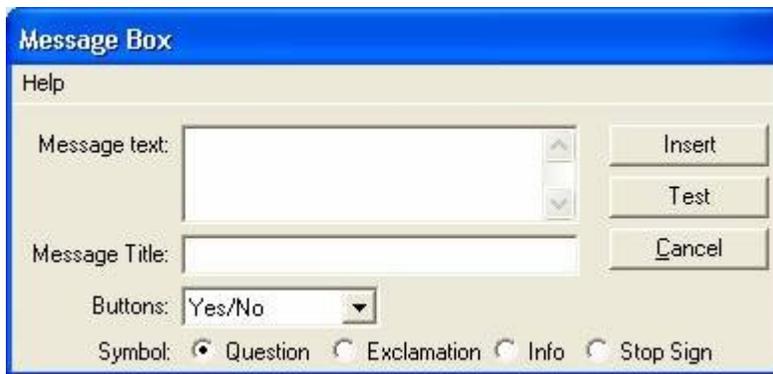
- "?" Question
- "!" Exclamation
- "i" Information
- "S" Stop Sign

Examples:

```
MessageBox "Your voice input was not understood" "Listening" "OK" "S"
```

Wizard:

Miscellaneous Commands



[onExcelChange](#)

onExcelChange allows you to specify a script macro to execute when the system detects that the user has changed the value of a cell in a visible Excel window. It provides you with a facility to react interactively to user changes in the Excel worksheet. The only available action is to run a Script Macro, which can either be a Macro File or an embedded Macro.

When a change is detected, Vox Proxy assigns two variables:

1. **&excelChangeCell** is assigned the cell reference of the cell changed.
2. **&excelChangeValue** is assigned the new value in the cell.

You may also use ReadExcel to assign the cell value into a user-defined variable. There is a difference between the data assigned to the &excelChangeValue and the data assigned by the ReadExcel command: ReadExcel always reads the text of the cell, whereas &excelChangeValue may receive a numeric value, depending on the cell properties. If you want the character to say the value, it is recommended that you use ReadExcel rather than &excelChangeValue.

Syntax:

```
onExcelChange RunMacro "macro"
```

where "macro" is the name and optional path of a Script Macro.

Example:

In the slide script:

```
Show Paul at Bottom Left
ShowExcel file=&pres\mortgage.xls at 70 70 sheet=Sheet1 cell=A1 size=40 40
onExcelChange RunMacro "xlchange"
GestureLeft
Say Go ahead and change something in the worksheet.
```

In the macro "xlchange":

```
Set &newvalue=ReadExcel cell=&excelChangeCell
Say You changed the value of cell &excelChangeCell to &newvalue.
```

PlayFlash

Play a Macromedia Shockwave Flash movie in a window. Note: this uses (and requires) the Macromedia Flash Player.

Use the CloseFlash command to close the flash window.

Syntax:

PlayFlash [file=] [at X% Y%] [size=] [align=] [scale=] [loop] [close]

Options are:

1. **file**=file name.swf
2. **at location** as screen %. Default: 50 50. Location is at the CENTER of the form.
3. **size**=width,height (as screen %). Default: 50 50
4. **align**=LTRB Aligns the movie with the Left, Top, Right, and/or Bottom of the form. Any combination is valid. Default is to display the movie in the center of the form.
5. **scale**=X Options for X are:
 - **ShowAll** - (default) Preserves the aspect ratio of the movie, but ensures that it will fully display in the form window.
 - **NoBorder** - Forces the form window to be fully filled. If the window is not the same aspect ration as the move, one edge of the movie will not be visible (off the form).
 - **ExactFit** - Scales the movie to the form. If the aspect ratio of the movie is significantly different than the form, some distortion will be apparant.
6. **loop** - causes the movie to run continuously in a loop. Note that this require you to explicitly close the movie or it will go on forever.
7. **close** - causes the form to close (and disappear) when the movie stops playing. Note that, if the movie has a stop or pause button, pressing it will close the flash form.

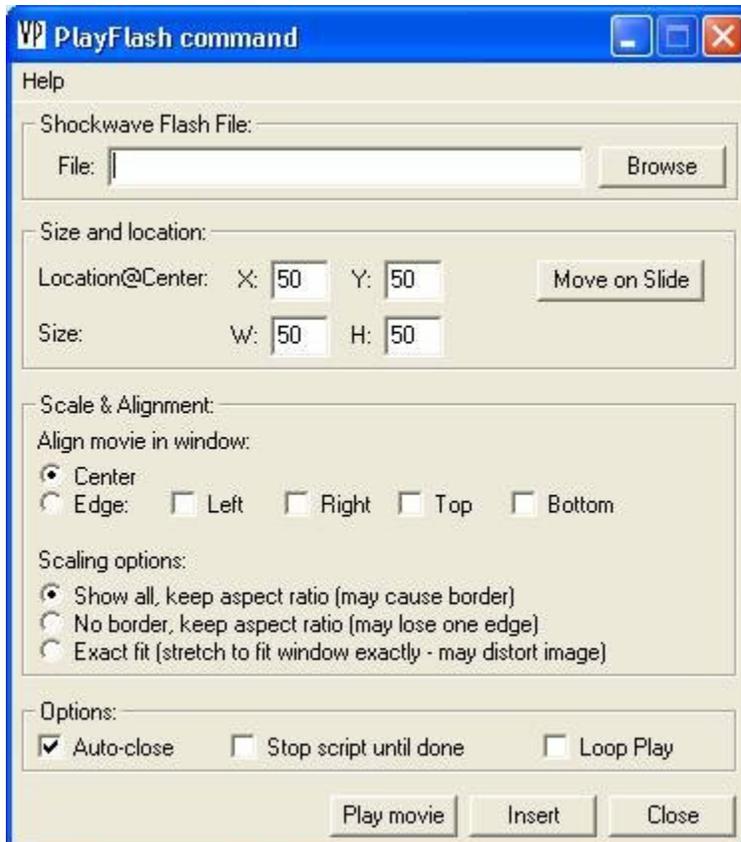
The PlayFlash command assigns a value of one to the variable &Response if the media file successfully starts or a zero if it does not.

Examples:

```
PlayFlash file=c:\program files\vox proxy\sample.swf at 25 25 size=50 50  
close  
PlayFlash file=c:\program files\vox proxy\sample.swf at 25 25 size=50 50  
scale=ExactFit  
PlayFlash file=c:\program files\vox proxy\sample.swf at 25 25 size=50 50  
close Align=TL
```

Wizard:

Miscellaneous Commands



[PlayMedia](#)

Plays a media file (audio or video) using Microsoft Media Player. Media Player has the advantage of playing more media types than the default MCI player used by PowerPoint.

Use the CloseMedia command to close the media window and/or audio.

Syntax:

PlayMedia[:nowait] "file" [at X Y] [hold] [size=S] [controls=C] [volume=V] [loop=N] [ontop]

Options:

1. **File.** Be sure to include quote marks around the file name. If no path is specified, the standard search path will be used.

2. **Screen position** (AT X Y). [default=50 50]
3. **Hold**. If you include the word hold, the script will not continue until the video file is complete.
4. **Display size**. Select one of the following:
 - size=1X Authored size
 - size=2X Twice authored size
 - size=1/16 One-sixteenth of the screen area
 - size=1/4 one-fourth of the screen area [default]
 - size=1/2 one-half of the screen area
 - size=FS full-screen
 - size=NONE No display (useful for audio files)
5. **Controls**. Select one of the following:
 - controls=none
 - controls=some [default]
 - controls=all
6. **Volume**. Specify a number from 0 to 100. The default volume is the maximum (100). Zero is not actually mute, but a very low volume.
7. **Loop**. Specify the number of times the file should play.
8. **Ontop**. This optional parameter is sometimes needed to ensure that the media is visible on top of other images.
9. **:nowait**. This works like all other :nowait options: if it is specified, the Microsoft Media Player will begin to load before the previous character is finished its speech or actions. May be used to "hide" the load time for the Media Player.

The PlayMedia command assigns a value of one to the variable &Response if the media file successfully starts or a zero if it does not.

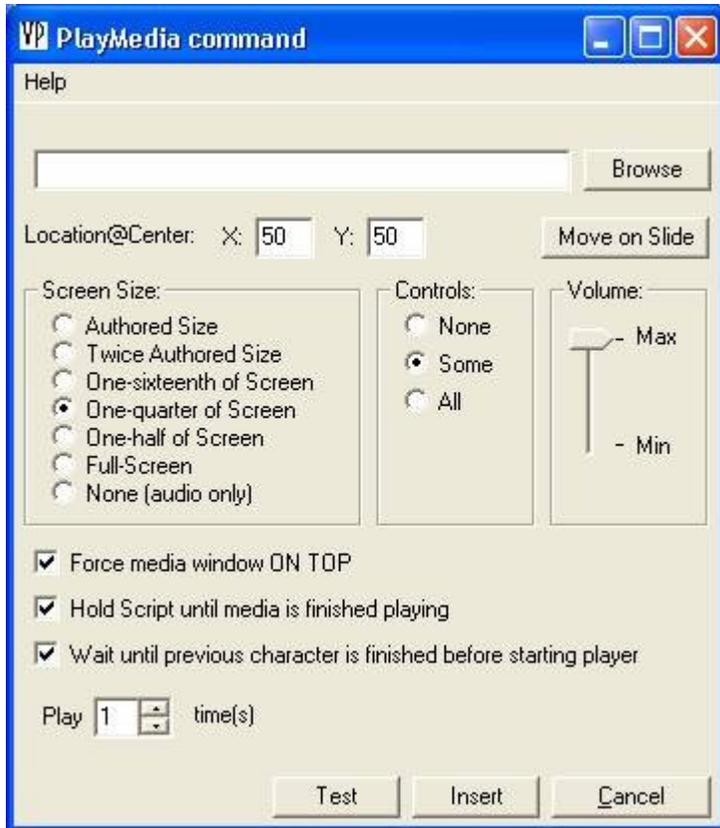
Examples:

```
PlayMedia "&media\STSlnd.mpeg" at 50 50 wait size=FS
PlayMedia "&media\guitar.wma" at 50 50 size=NONE controls=none
PlayMedia "c:\my files\music.mp3" at 50 50 size=1/4 controls=none
```

Note in these examples the use of the variable "&media". This variable is created automatically by Vox Proxy and points by default to the directory:
 \program files\vox proxy\media

Wizard:

Miscellaneous Commands



[PlayWav](#)

Play an audio (.wav) file using the standard Windows MCI media player (not Microsoft Media Player).

Syntax:

PlayWav [wait] "file"

Be sure to include quote marks around the file name. If no path is specified, the standard search path will be used.

If you include the word "wait", the script will be held until the audio is complete before proceeding. Without the word "wait", the script will continue to run while the audio plays.

Examples:

```
PlayWav wait "c:\windows\media\chimes.wav"
```

Wizard:

Miscellaneous Commands



RandomLineFrom

Inserts into the current script a single line, selected at random, from the specified Macro Script. Note that the line retrieved should have any needed script command in it. If the line is for some spoken text, for example, it should start with the "Say" command.

This command can be used with a macro of data to produce a random joke, saying, etc.

Syntax:

RandomLineFrom "macro"

Examples:

RandomLineFrom "Pearls"

Wizard:

Miscellaneous Commands



RandomMacro

Randomly select a script macro whose name begins with "name". Get and insert its script into the currently running script. When using this command, enter the name in quotes.

Syntax:

RandomMacro "name"

Examples:

RandomMacro "joke-"

Wizard:

Miscellaneous Commands



[ReadExcel](#)

This command reads a value from an Excel worksheet. It is normally used with the Set command to assign the value to a variable.

The value is accessed by specifying the file name, worksheet name, and cell range.

Syntax:

Set &variable=ReadExcel file=F sheet=S cell=C

File is an Excel workbook (XLS)

S is a valid **worksheet name** in the workbook

C is a valid **cell**

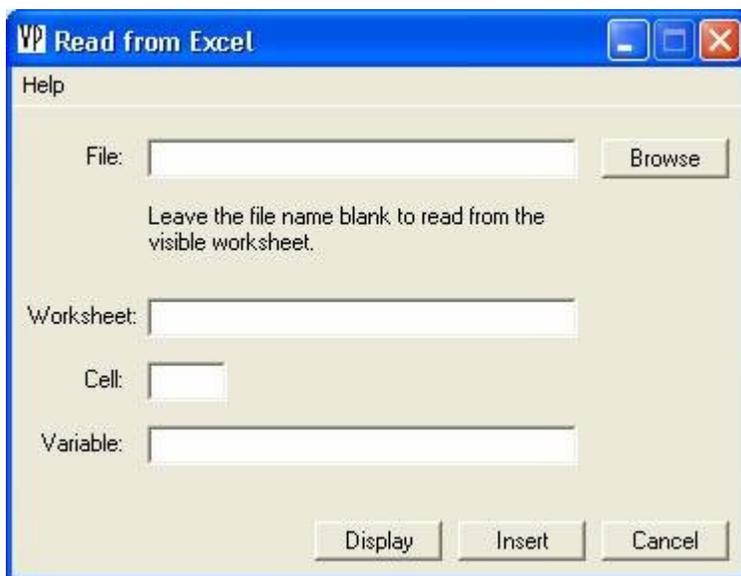
Examples:

```
Set &Q=ReadExcel file=c:\tests\questions.xls sheet=Sheet1 cell=B9
```

Assigns the contents of cell "B9" of worksheet "Sheet1" of the file "c:\tests\questions.xls" to the variable &Q

Wizard:

Miscellaneous Commands



[ReadFile](#)

This command reads a value from an "ini-style" text file. It is normally used with the Set command to assign the value to a variable.

The value is accessed by specifying the SECTION name and KEY for the value.

Syntax:

Set &variable=ReadFile file=F section=S key=K

file is the full path and name of an ini-style text file

S is a valid **section** in the file

K is a valid **key**.

Examples:

The file "c:\program files\vox proxy\test1.txt" contains the following:

```
[history]
```

```
q1=In what year did President Lincoln deliver the Gettysburg Address?
```

```
q2=...
```

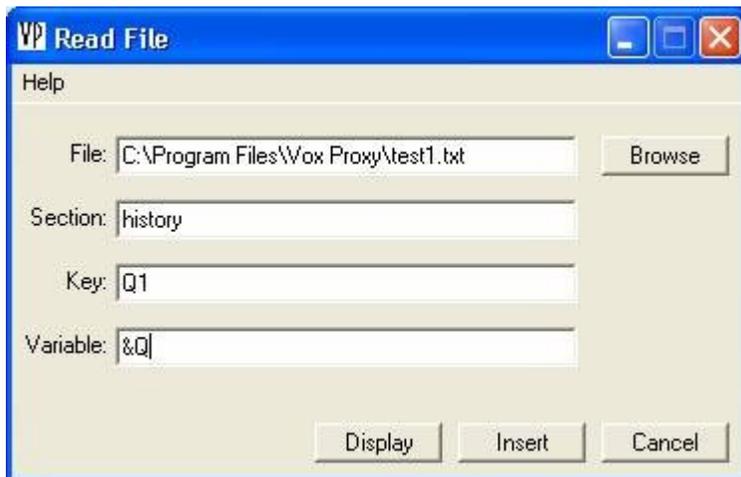
```
q3=...
```

The script command to assign question 1 to the variable &Q would be:

```
Set &Q=ReadFile file=c:\tests\questions.txt section=history key=q1
```

Wizard:

Miscellaneous Commands



Run

Execute a Windows program.

Syntax:

Run program [options] [default folder]

Allows two parameters:

1. Program options or parameters
2. Default directory

Examples:

Open a text file with Notepad:

```
Run notepad "readme.txt"
```

Special cases allow you to invoke Windows Explorer, Internet Explorer, or your E-mail client.

Start the default mail client:

```
Run "mailto:tomatkins@voxproxy.com"
```

Start the default browser on a web page:

```
Run "http://www.voxproxy.com"
```

Wizard:

Miscellaneous Commands



RunMacro

During the execution of the current script, when it encounters a RunMacro command, it loads and runs that macro. When that macro is complete, the remainder of the current script continues to run.

Note that this is functionally different than a "GetMacro" command, which is essentially a "cut and paste" operation that takes place just before the current script is executed. One of the key differences is that RunMacro allows "local" copies of Variables and Pronunciation Dictionary entries which may be different from those in the current script.

Syntax:

```
RunMacro "file"
```

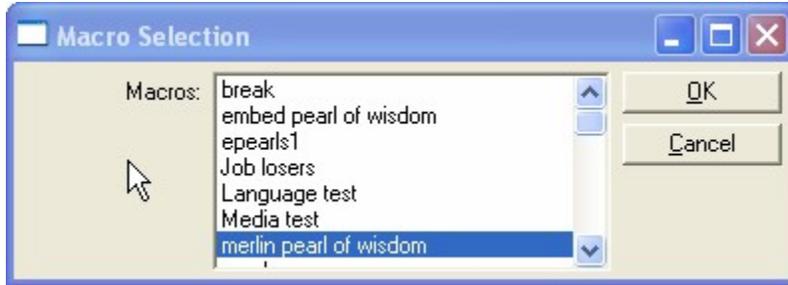
The specified macro file name is not case sensitive.

Examples:

```
Show Peedy at Right Center  
Say Merlin, give us a pearl of wisdom!  
RunMacro 'Merlin Pearl of Wisdom'  
Peedy say Thank you
```

Wizard:

Miscellaneous Commands



Set

Defines a Variable. Variable names must contain only alphanumeric characters (case-sensitive), no spaces, and no symbols except underline ("_"). These variables will persist as long as the current PowerPoint session remains open and will disappear when closed.

Syntax:

Set &varname=value

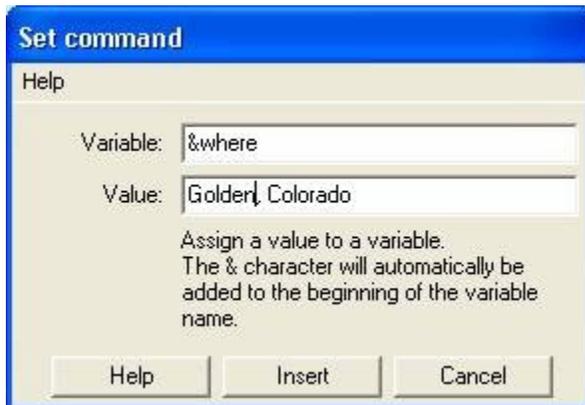
&varname is any valid variable name. Value is an alphanumeric or numeric string. For "value", you may also use ReadExcel or ReadFile to assign values from an Excel worksheet or an ini-style text file.

Examples:

```
Set &name=Tom
Set &place=Orlando
Set &greet=Good &tod, &name!
```

Wizard:

Miscellaneous Commands



SetPrefs

Set one or more preference items. This sets the preferences that apply to all characters as the default when the character is shown. It has the same effect as changing the corresponding item from Tools/Preferences in the Script Writer.

The command takes effect when it is encountered in the script. Also see help for the temporary commands: IdleAnimations and SoundEffects.

Note that this change remains in effect for all future sessions of Vox Proxy. If your need for these changes is temporary, you should set the preferences back at the end of the script.

Syntax:

SetPrefs [idle=on|off] [sound=on|off] [timeout=T]

Options:

idle=on or off - turns idle animations on or off

sound=on or off - turns sound effects (associated with various animations) on or off

timeout=T - sets the time in seconds after which known looping animations will be stopped.

Examples:

```
SetPrefs idle=off sounds=off timeout=15
```

Wizard:

Miscellaneous Commands



[ShowExcel](#)

Display an Excel worksheet in a window. Use the Wizard form to resize and drag-and-drop the Excel window on the slide.

The file name can refer to a file on the web. If you leave off the file= option, it will display a new, blank worksheet.

A timer will start automatically when the Excel window opens. You may access the timer using ExcelTime with the WaitFor command.

Use the CloseExcel command to close the Excel window.

Syntax:

ShowExcel [file=] [sheet=] [cell=] [at X% Y%] [size=] [readonly] [displayoptions]

Options:

The option names are not case-sensitive.

1. **file**=file name.xls. Default: an empty sheet
2. **sheet**=sheet name. Default: Sheet1

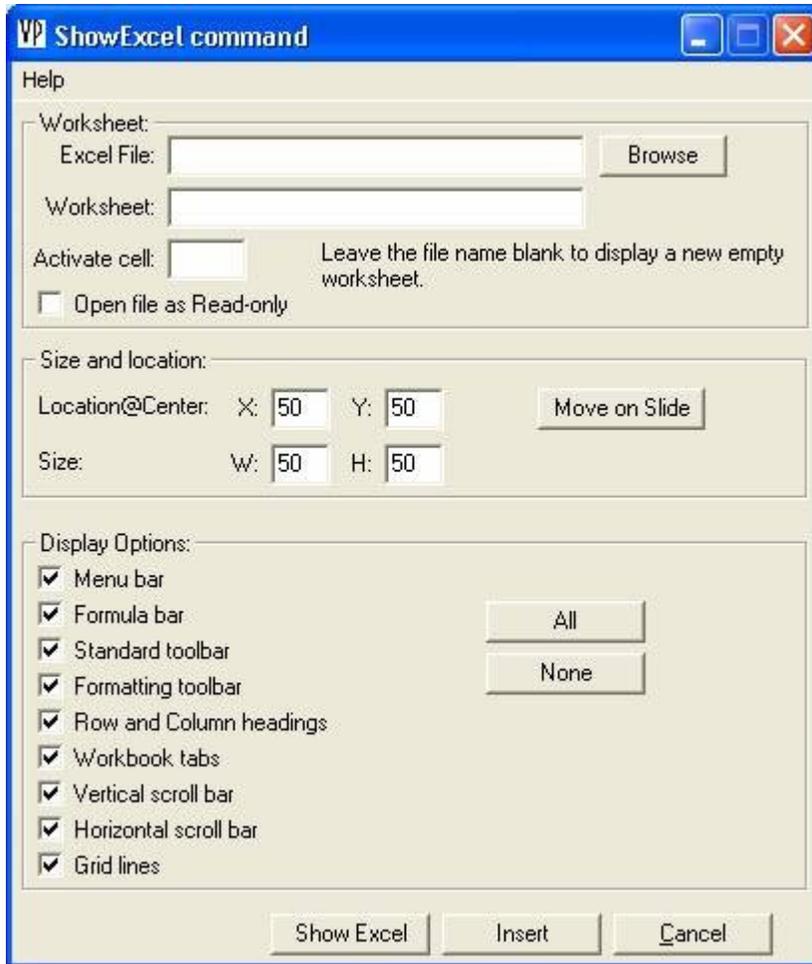
3. **cell**=cell reference to activate. Default: A1
4. **at location** as screen %. Default: 50 50. Location is at the CENTER of the Excel form.
5. **size**=X,Y (as screen %). Default: 50 50
6. **readonly** If not specified: read/write unless commandbars are off, then readonly.
The following options specify window display options for the Excel window. The options for each are on or off.
7. **FormulaBar**=on|off
8. **MenuBar**=on|off
9. **StandardToolBar**=on|off
10. **FormattingToolBar**=on|off
11. **Headings**=on|off (Row and Column headings)
12. **WorkBookTabs**=on|off
13. **VScrollBar**=on|off (Vertical scroll bar)
14. **HScrollBar**=on|off (Horizontal scroll bar)
15. **GridLines**=on|off

Examples:

```
ShowExcel file=c:\program files\vox proxy\sample.xls at 25 25 size=50 50
sheet=january FormulaBar=off MenuBar=off StandardToolBar=off
FormattingToolBar=off WorkBookTabs=off
ShowExcel file=http://www.voxproxy.com/downloads/vptest.xls at 25 25 size=50
50 sheet=january FormulaBar=off MenuBar=off StandardToolBar=off
FormattingToolBar=off WorkBookTabs=off
```

Wizard:

Miscellaneous Commands



ShowWord

Display a Microsoft Word document in a window.

A timer will start automatically when the Word window opens. You may access the timer using WordTime with the WaitFor command.

Use the CloseWord command to close the Word window.

Syntax:

ShowWord [file=] [at X% Y%] [size=] [readonly] [commandbars=]

Options:

1. **file**=file name.doc. Default: an empty document
2. **at location** as screen %. Default: 50 50. Location is at the CENTER of the Word form.
3. **size**=X,Y (as screen %). Default: 50 50
4. **readonly** If not specified: read/write unless commandbars are off, then readonly.

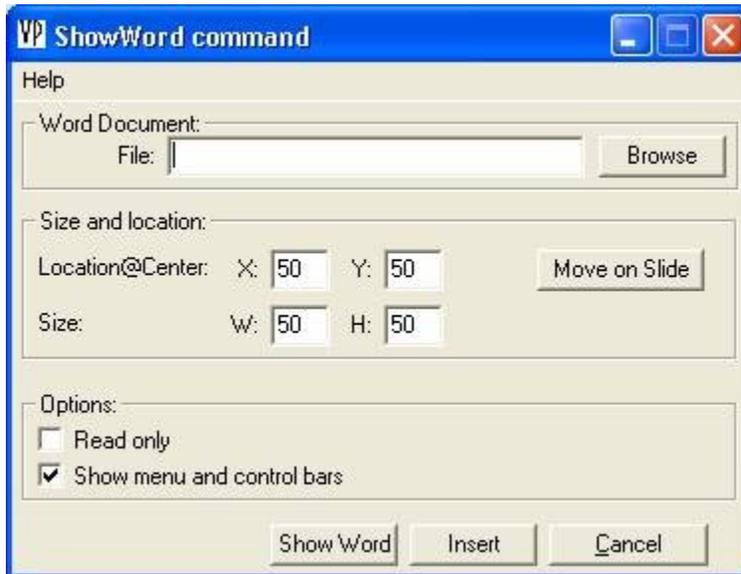
5. **commandbars**=on or off. Default: on. This includes the menu, standard, and formatting bars. If commandbars are off, the document will open readonly.

Examples:

ShowWord file=c:\program files\vox proxy\sample.doc at 25 25 size=50 50 commandbars=off

Wizard:

Miscellaneous Commands



StartTimer

Sets the start time for a new timer with the name 'name'.

Once started, the WaitFor command can be used to wait until this timer reaches a specific number of seconds.

Syntax:

StartTimer name

Examples:

```
StartTimer MyTimer
...
Waitfor MyTimer 0:32
```

Wizard:

Miscellaneous Commands



WaitFor

This command will hold the script and wait for one of several events to occur:

- A specified character to complete its speech or actions
- The end of a media file or flash file
- A specified length of time from:
 1. The start of the current slide
 2. The start of the slide show
 3. The start of a media file (audio, video, or Flash)
 4. The start of a custom timer
 5. The completion of the previous character's speech or action

Syntax:

WaitFor Item [Time]

Time can be in seconds or min:sec

Item can be one of the following:

- Character (followed by character name)
- ShowTime, SlideTime, or MediaTime (followed by time)
- MediaEnd
- FlashEnd (followed by full path and name of flash file)
- CustomTimerName (followed by time)

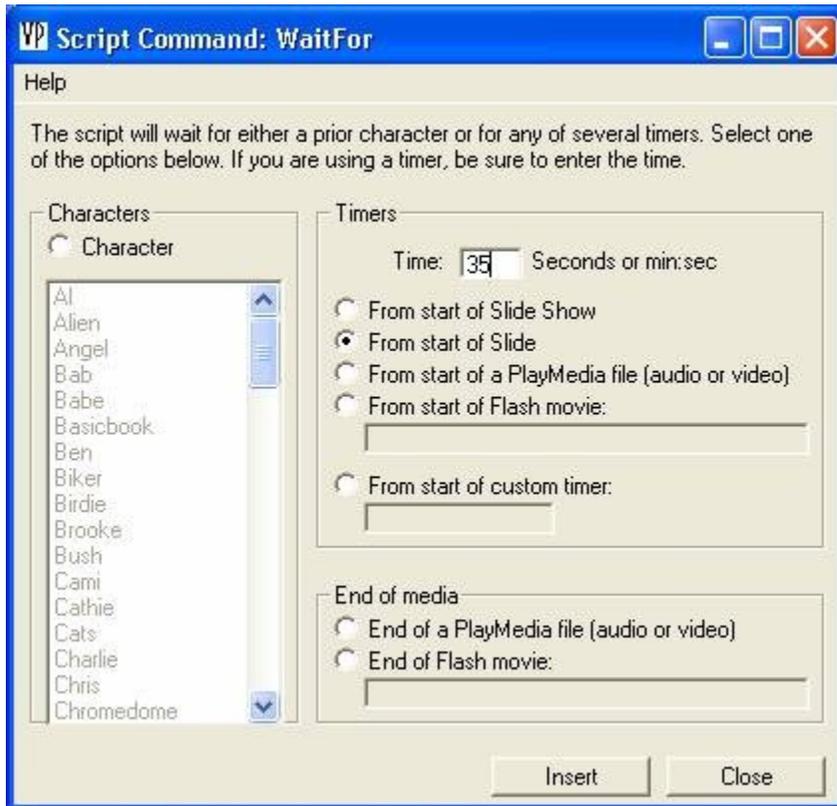
Item can be omitted, in which case the script will be held at that point for the elapsed time specified.

Examples:

```
WaitFor character Merlin
Waitfor SlideTime 1:23
WaitFor MediaTime 4:15
WaitFor MyTimer 5
Waitfor FlashTime "C:\Program Files\Vox Proxy\Media\VPweb.swf" 30
WaitFor MediaEnd
WaitFor 5
```

Wizard:

Other Agent Commands



WriteExcel

This command writes a value to a cell in an Excel worksheet. It is normally used to write the contents of a variable. The value is written by specifying the name of the file, the worksheet, and the cell range.

Syntax:

WriteExcel file=F sheet=S cell=C value=&Count

File is an Excel workbook (XLS)

S is a valid **worksheet name** in the workbook

C is a valid **cell**

value is either a constant or the name of an existing variable

Examples:

```
WriteExcel file=c:\tests\questions.xls sheet=Sheet1 cell=B9 value=&Count
```

Wizard:

Miscellaneous Commands



WriteFile

This command writes a value to an "ini-style" text file. It is normally used to write the contents of a variable.

The value is written by specifying the SECTION name, KEY, and VALUE. If the specified section and key names are new, they will be created in the file. If the file does not exist, it will be created.

Syntax:

WriteFile file=F section=S key=K value=&var

file is the full path and name of an ini-style text file

S is a valid **section** in the file

K is a valid **key**

value is either a constant or the name of an existing variable

Examples:

You are keeping count of the number of incorrect answers to a question in the variable &Count and wish to write the value into the "history" section of the file "c:\tests\questions.txt":

```
WriteFile file=c:\tests\questions.txt section=history key=wrong value=&Count
```

Wizard:

Miscellaneous Commands



PowerPoint commands

[Click](#)

Send a mouse click to PowerPoint to start the next custom animation on the slide. If the slide does not contain another custom animation, it will cause PowerPoint to change slides.

This command is used in self-narrated slides or slide shows to control the custom animations on the slide. It is important the custom animation be set in PowerPoint to start "on click" and that timings are not used in PowerPoint to run the slide show. For self-narrated slides, follow these rules:

- In PowerPoint Custom Animation:
- Start "on click" (not "with previous" or "after previous")
- in the timing option, set to start "on click" and not after a specified time.
- In PowerPoint Slide Show/Set up show , check the box to advance slides manually. Do not use timings.
- Make sure you have the same number of Click or WaitForClick commands as you have PowerPoint custom animations on slide objects. If you are unsure or if the slide does not play properly, use the Diagnose button on the Script writer.

Click supports the :nowait option, allowing you to start the slide animation while a character is speaking or performing an animation.

Synonym:

NextAnimation

Syntax:

Click or NextAnimation

Examples:

```
Show Paul at left center balloons=off
Point45
Click
Say Good morning. The first topic Id like to talk about is..
```

Wizard:

PowerPoint Commands

[EndShow](#)

End the current slide show. Close any characters which may be visible.

Syntax:

EndShow

Wizard:

PowerPoint Commands

[FirstSlide](#)

Go to the first slide in the presentation.

Syntax:

FirstSlide

Wizard:

PowerPoint Commands

[GotoSlide](#)

Go to the specified slide number. The slide number may be specified either as a relative slide number (starting with 1) or a Slide ID. Slide ID's are shown in the Wizard form for the GotoSlide command. If you use relative slide numbers, they must be changed manually if you add or delete slides. Slide IDs remain constant with moving or deleting slides and so do not need to be kept updated. However, the PowerPoint Viewer does not support slide IDs, so if you anticipate showing your presentation with the viewer (including from CD Prep), you should use slide numbers rather than IDs.

Slide animations do not reset themselves to the unplayed state when navigating with GotoSlide. See the discussion on playing self-narrated slides containing custom animations.

Syntax:

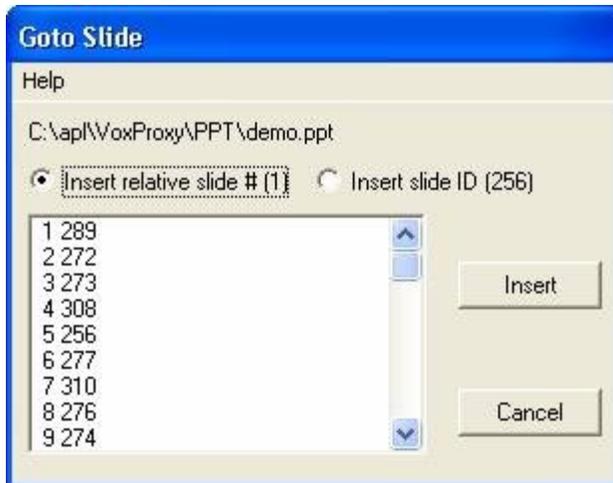
GotoSlide N

Examples:

```
Balloons on size=14 font="arial" lines=5  
Balloons off
```

Wizard:

PowerPoint Commands



LastSlide

Go to the last slide in the presentation.

Syntax:

LastSlide

Wizard:

PowerPoint Commands

LastSlideViewed

Go to the last slide which was actually viewed in the slide show. Note that this may not be the same as the "PreviousSlide".

Important note: This command will not work when playing the slide show with the PowerPoint Viewer.

Syntax:

LastSlideViewed

Wizard:

PowerPoint Commands

Notes about navigating among slides in PowerPoint.

If you change the relative slide number of a slide by inserting or deleting slides, the script will not change automatically. You must manually update any "GotoSlide" command with the correct slide number.

If you re-visit a slide that has custom animations on it, PowerPoint will not start with the animations un-played. In other words, the second time the slide is shown, the animations will already be done and the first click command will just change slides. There is one important exception that allows you to re-set custom animations: when you do a "NextSlide" command. NextSlide re-sets all animations so that they will play the same way they played the first time

the slide was shown. If, therefore, you need to navigate directly to slides, it is best to insert a duplicate slide (with custom animations removed) in front of the target slide and navigate to the blank slide. The blank slide should contain a script with only the one command: NextSlide.

NextSlide

Go to the next slide in the presentation. If there are no more animations on the current slide, PowerPoint will move to the next slide.

Syntax:

NextSlide

Wizard:

PowerPoint Commands

PreviousSlide

Go to the previous slide in the presentation. The previous slide is the slide numerically preceding the current slide and may not be the last slide viewed by the user.

Syntax:

PreviousSlide

Wizard:

PowerPoint Commands

RunSlideShow

Run the specified file in another slide show window. The script continues on in its current slide show after the new slide show is ended. You may optionally specify the starting and ending slides. User-defined variables may be used in the file specification.

Syntax:

RunSlideShow "file" [startslide endslide]

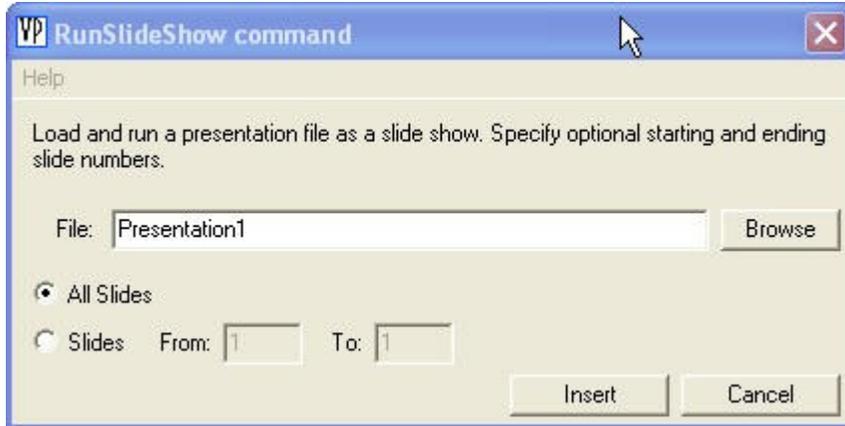
File must be a PowerPoint file name, with or without a path specification (see specifying files). File can also refer to a file on the internet, using the file specification as http://etc/file.ppt.

Examples:

```
RunSlideShow "c:\tutorials\Excel Tutorial.ppt"  
RunSlideShow "&pres\Excel Tutorial.ppt" 6 12  
RunSlideShow "http://www.voxproxy.com/downloads/Tutorial - Variables.ppt"
```

Wizard:

PowerPoint Commands



WaitforClick

Wait until the user presses a mouse click to start the next animation on the current slide before proceeding with the script. See the discussion on using Click and WaitforClick in self-running presentations.

Synonym:

WaitforNextAnimation

Syntax:

WaitforClick or WaitforNextAnimation

Wizard:

PowerPoint Commands

WriteShapeText

Displays text in a specified PowerPoint slide shape on the current slide at a specified position. This is one way to display text "on the fly" during a slide show. It can be used with a user-defined variable to display text that has been read from a file or worksheet.

Important note: This command will not work when playing a slide show with the PowerPoint Viewer.

Syntax:

WriteShapeText [shape=S] [position=P] [text=T]

Options are:

S is the name of the shape into which the text will be displayed.

P is the position at which to insert the text. Position 1 will insert at the beginning of any existing text.

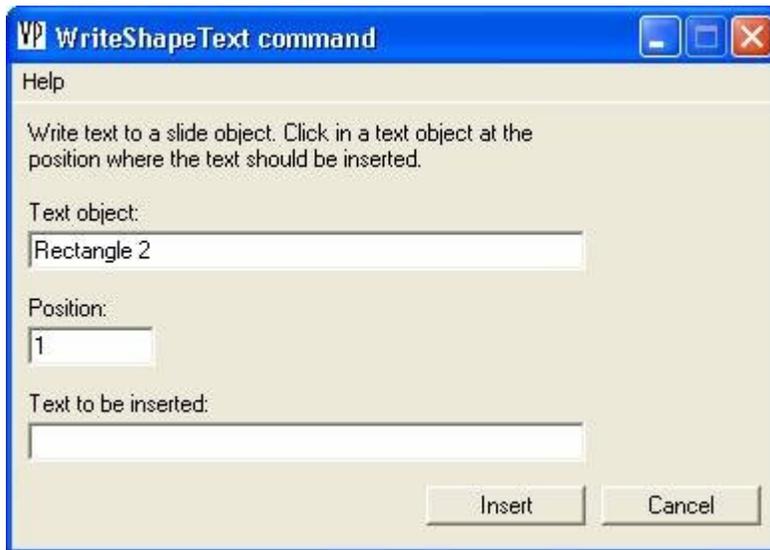
T is the text to insert

Examples:

```
WriteShapeText shape=Rectangle 1 position=1 text=Tom Atkins
WriteShapeText shape=Rectangle 1 position=1 text=&name
```

Wizard:

PowerPoint Commands



Programming commands

[BookMark](#)

A Bookmark is a single word which creates a place holder in the script at this point. The name must consist only of alphanumeric characters, no spaces, no symbols. The only purpose for a bookmark is to branch to it using the GOTO command.

Syntax:

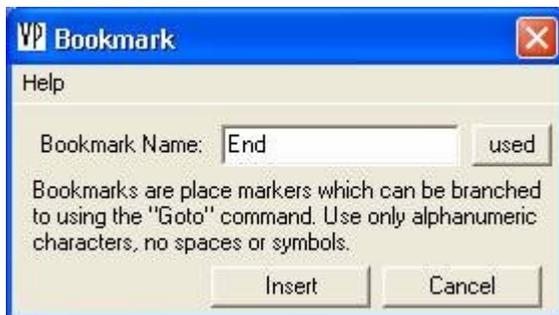
Bookmark name

Examples:

See Script Programming Examples

Wizard:

Miscellaneous Commands



[GoTo](#)

Goto is used to branch to a bookmark in the script. It is most often used in evaluating the user's response to a ListBox or ListenFor command.

Syntax:

Goto bookmark

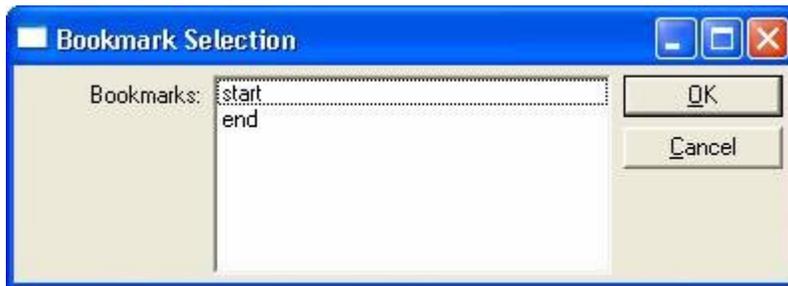
Example:

```
Bookmark Start
ListenFor "yes" "no"
:SELECT &Response
:CASE "yes"
say You said yes
:CASE "no"
say You said no
:ELSE
say Please try again.
say I am only listening for yes or no.
:GOTO Start
:END
```

Wizard:

Miscellaneous Commands

The GOTO wizard will prompt you with a list of bookmarks in your script.

**:If ... :Else ... :End**

:IF evaluates a test statement for whether it is true or false. If the statement is true, the script immediately following it is run. If false, it is skipped. Script which follows the optional :ELSE command, if any, is just the reverse: it runs only if the statement is false. :ELSE can also be written as OR:ELSE. An :END command must follow each :IF command in the script.

The statement evaluated is most often a test of the &Response variable after the user has responded either to a Listen, ListenFor, or ListBox command.

Synonyms:

:ORELSE = :ELSE

Syntax:

:IF test statement

...

[:ELSE]

...

:END

The test statement normally uses one or more comparison functions to evaluate whether the statement is true or false.

Comparison functions available for use in an :IF statement are:

- **IS** (:IF &Response IS "yes") Compares a variable to text or another variable. If they are the same, the script continues. The comparison is not case-sensitive.
- **ISNOT** (:IF &Response ISNOT "yes") Compares a variable to text or another variable. If they are not the same, the script continues. The comparison is not case-sensitive.
- **LT** (less than) (:IF &count LT 10) Compares a numeric variable to a number or another numeric variable. If the first is less than the second, the script continues.
- **LE** (less than or equal to) (:IF &count LE 10) Compares a numeric variable to a number or another numeric variable. If the first is less than or equal to the second, the script continues.
- **GT** (greater than) (:IF &count GT 10) Compares a numeric variable to a number or another numeric variable. If the first is greater than the second, the script continues.
- **GE** (greater than or equal to) (:IF &count GE 10) Compares a numeric variable to a number or another numeric variable. If the first is greater than or equal to the second, the script continues.
- **EQ** (equals) (:IF &count EQ 10) Compares a variable to a value or another variable. If they are equal, the script continues. If the values are text, EQ is the same as IS.
- **NE** (not equal) (:IF &count NE 10) Compares a variable to a value or another variable. If they are not equal, the script continues. If the values are text, NE is the same as ISNOT.
- **NOT** (not) (:IF NOT &count EQ 2) Not reverses the logic test that follows.
- **AND** (and) (:IF (&Response EQ "yes") AND (&count LT 3)) And combines logical tests, each of which must be enclosed in parentheses. Only if both tests are true does the script continue to execute.
- **OR** (or) (:IF (&Response EQ "yes") OR (&count LT 3)) OR combines logical tests, each of which must be enclosed in parentheses. Only if either test is true, the script continues to execute.

Examples:

```
Show Peedy at Center
...
Say Would you like more detail?
Say Please say yes or no.
Listen
:IF &Response IS "yes"
Say OK, let's take a more detailed look.
:ELSE
Say OK, we'll skip the detail.
GotoSlide 26
:END
```

Wizard:

Miscellaneous Commands

[:Select... :Case... :Else... :End](#)

The :Select command (which can also be written :Evaluate) begins a section of script which evaluates multiple possible responses from the user, called Cases, and runs a piece of script for each case.

It can be interpreted in English as follows: "Select the user's response. In case it's ONE, run the following script; In case it's TWO, run the following script; In case it's THREE, run the following script; OR ELSE, if it's none of the above, run the following script...Now I'm done.

:Select is most often used with either the ListenFor or the ListBox command.

Synonyms:

:Select = Evaluate
:InCase = :Case
:ORELSE = :ELSE

Syntax:

```
:SELECT variable
:CASE value1
...script to be run if variable equals value1
:CASE value2
...script to be run if variable equals value2
...etc (as many additional :CASEs as needed)
:ELSE
...script to be run if variable does not equal any of the cases
:END
```

Examples:

```
Show Peedy at Center
Say Welcome to Peedy's Pizza.
Say Would you like Pepperoni, Sausage, Hawaiian, or Veggie?
ListenFor "Pepperoni" "Sausage" "Hawaiian" "Veggie"
:Select &Response
Case "Pepperoni"
...place your "Pepperoni" script here
Case "Sausage"
...place your "Sausage" script here
Case "Hawaiian"
...place your "Hawaiian" script here
Case "Veggie"
...place your "Veggie" script here
:ELSE
...place script here for the case where Peedy didn't recognize what the user
said.
:END
```

Wizard:

Miscellaneous Commands

[Variables](#)

Variables are special words that are used to represent text or numbers within a script. Names of variables always begin with an ampersand (&) and consist of a single case-sensitive word without spaces. An example of a variable is **&date**, which contains the current date in the form "January 1, 2005". Another example is **&name**, which might contain the text "Alice".

Variable can be used within speech text or within other script commands.

There are two kinds of variables: pre-defined and user-defined.

Pre-defined Variables

Pre-defined variables are variables that are built into Vox Proxy. You can use them wherever variables can be used, but you cannot change them.

Pre-defined variables include:

- [Date and time-related variables](#)

- **&date** Date in form January 1, 2005
- **&time** Time in form 2:00pm
- **&day** Day of week in form: Monday
- **&month** Month in form: January
- **&year** Year in form: 2005
- **&tod** Time of day in form: morning, afternoon, or evening

- [Directory path variables](#)

Several pre-defined variables are provided for directory paths. These variables do NOT end with a back-slash, so when you use them, you should supply the back slash as appropriate.

- **&tempdir** Contains the path to the Windows temp folder on the user's computer. For use in CD Prep scripts to locate files temporarily copied to the user's hard drive.
- **&windir** Contains the path to the user's Windows folder
- **&appdir** c:\program files\vox proxy
- **&chardir** c:\windows\msagent\chars
- **&DiskDrive** [current default disk drive]
- **¯odir** c:\program files\vox proxy\macros
- **&mediadir** c:\program files\vox proxy\media
- **&presdir** c:\program files\vox proxy\presentations
- **&tempdir** [windows temp folder]
- **&windir** [windows folder]
- **&DiskDrive** Contains the drive letter of the default drive (and a colon). It is used in presentations to be played on CD Prep disks to determine the drive on which the CD is playing.

- [Presentation-related variables](#)

- **&pptpath** Contains the path to the currently-active PowerPoint presentation. Does NOT end with a back-slash.
- **&SlideNumber** Contains the number of the currently-displayed slide.

- [The special Response variable](#)

&Response is a special pre-defined variable that contains the result of specific user inputs or actions. For example, it can contain the text of the user's response to a Listen, ListenFor, or ListBox command. If the response was not recognized, it is empty.

- [Other special variables](#)
 - &excelChangeCell - the cell reference for a cell changed by the user
 - &excelChangeValue - the new value of a cell changed by the user

See onExcelChange

Note that the names of all these variables are case-sensitive.

User-defined Variables

User-defined variables may be defined in the Variables section of the Script Writer's Tools menu or from within a script using the Set command. Names should consist of a single word without spaces. When used in the script, names must always begin with the character "&".

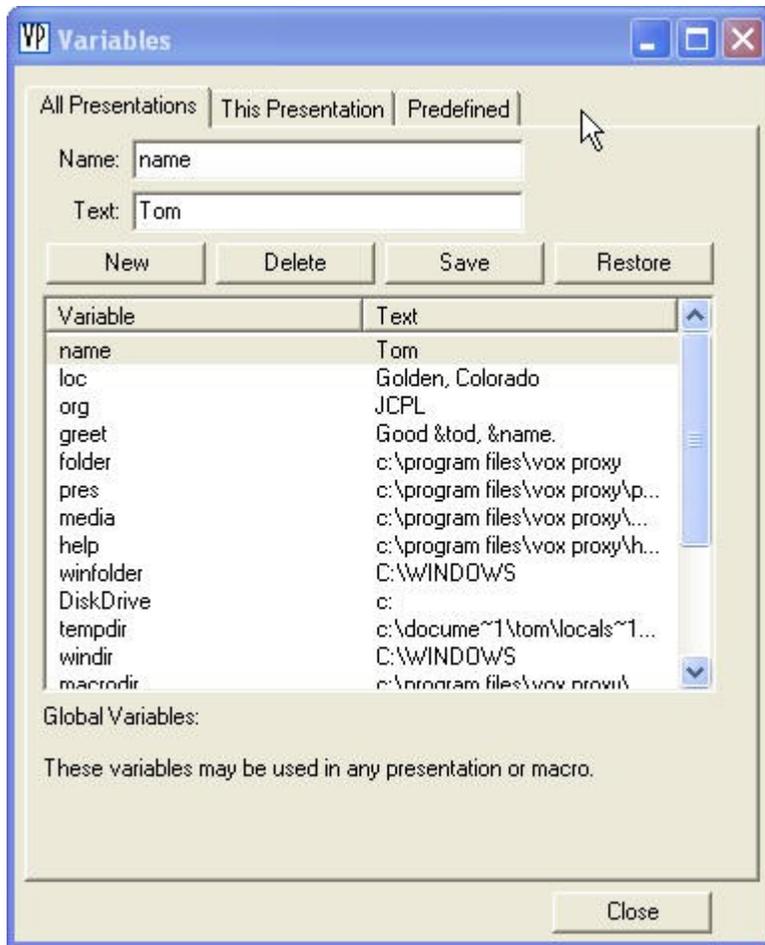
Creating Variables

Variables can be created in two ways:

1. With the [Variables dialog box](#) on the Script Writer tools menu.

There are three pages, or tabs, on the Variables form:

- **All Presentations**
These variables apply to all presentations and macros.
- **This Presentation**
These variables apply only to the active presentation or macro. Any variables defined here will override those defined on the "All Presentations" page.
- **Pre-defined**
These variables are available in any presentation or macro. They include time-oriented variables: &day, %date, &month, &year, &time, and &tod (time of day), and several speech tags: &w (whisper), &m (monotone), &n (normal), &r (reset), and &e (emphasize).



2. Within a script using the Set command.

Using Variables

In Speech

Examples may be the simplest way to understand the use of user-defined variables in speech.

Variable name: &speaker

Variable text: "Tom"

Sample line of script: `Say Good morning, &speaker. My name is Peedy.`

Variable names must only consist of a single word, but the text in variables may be longer:

Variable name: &org

Variable text: "the Jefferson County Public Library"

Sample line of script: `Say Welcome to &org.`

A variable may contain other variables:

Variable name: &greet

Variable text: "Say Good &tod, &speaker."

Sample line of script: `Say &greet. Welcome to &org.`

In Directory Paths

Anywhere you refer to media files, Macros, or other files, it is often a good idea to use a variable for the path rather than to hard-code the path to a location on your hard drive. Several pre-defined variables are provided for this purpose (see above). [Example](#)

```
PlayWav wait &Media\APPLAUSE.WAV
```

To Capture Input

Variables are the only way to capture input from the user. Several commands in Vox Proxy automatically assign user input to the variable **&Response**:

- ListBox
- Listen
- ListenFor
- MessageBox

The Input command will assign data to any user-defined variable, defaulting to &Response.

After using any of the above commands, you can re-assign the data to a user-defined variable of your choice. [Example](#)

```
Set &Name=%Response
```

In Calculations

Numeric variables can be used in a variety of calculations. For example, if you have a quiz and you would like to keep score, you can use a variable such as &Score to accumulate the total.

At the beginning of the quiz, you would set the value to zero:

```
Set &Score=0
```

On a slide where the user made a correct response, you could increment &Score:

```
Set &Score=&Score+1
```

For Slide Navigation

Variables provide a useful way to navigate in slides. On any given slide, you can capture the current slide number into a variable, say &S:

```
Set &S=&SlideNumber
```

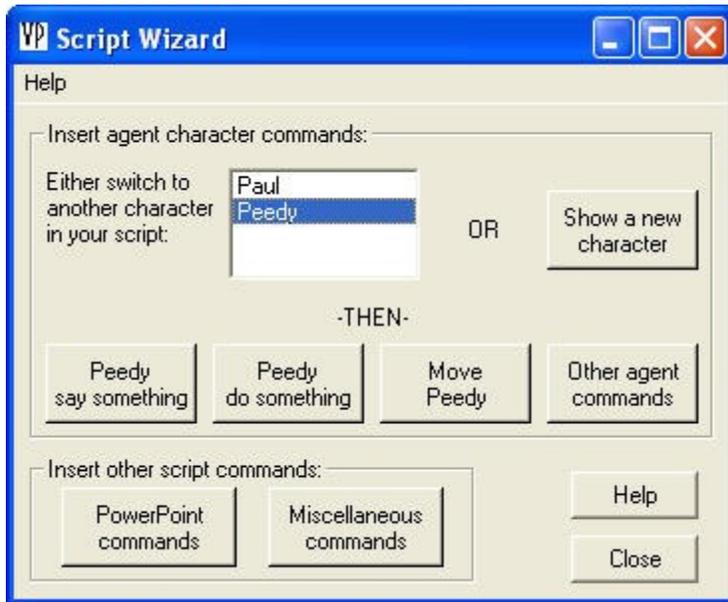
Then, if you have action settings or hyperlinks to other slides, you can always return to the starting slide using the GotoSlide command:

```
GotoSlide &S
```


Script Wizard

[Script Wizard](#)

Click on buttons below for more information:



Scripting character speech and actions using the Script Wizard requires only two simple steps:

1. Select a character or click to show a new character
2. Press a button to:
 - Enter and test speech text
 - Insert animations for the selected character
 - Specify a new location for the character by dragging it on the screen
 - Select and insert PowerPoint commands such as NextSlide or Click
 - Select and insert other character commands
 - Select and insert other miscellaneous commands

Note that all of the features available from this wizard form are also available through the Power Scripting Menu.

[Character Commands](#)

- Balloons
- Disappear
- Do
- GestureAt
- Hide

- IdleAnimations
- Interrupt
- Language
- Listen
- ListenFor
- MoveTo
- NoBalloons
- PointAt
- Say
- SayWav
- ScaleCharacters
- SetPrefs
- Show
- Size
- SoundEffects
- Speak
- Think
- TTSEngine
- Wait
- WaitCommands

Miscellaneous Commands

- Background
- Bitmap
- CloseBitmap
- CloseExcel
- CloseFlash
- CloseMedia
- CloseWord
- Delay
- DisplayClock
- DualMonitor
- EndMacro
- GetMacro

- Input
- ListBox
- MacroMenu
- MediaVolume
- MessageBox
- PlayFlash
- PlayMedia
- PlayWav
- RandomLine
- RandomMacro
- ReadExcel
- ReadFile
- Run
- RunMacro
- Set
- SetPrefs
- ShowExcel
- ShowWord
- StartTimer
- WaitFor
- WriteExcel
- WriteFile

PowerPoint Commands

- Click
- EndShow
- FirstSlide
- GotoSlide
- LastSlide
- LastSlideViewed
- NextSlide
- PreviousSlide
- RunSlideShow
- WaitForClick

- WriteShapeText

Programming Commands

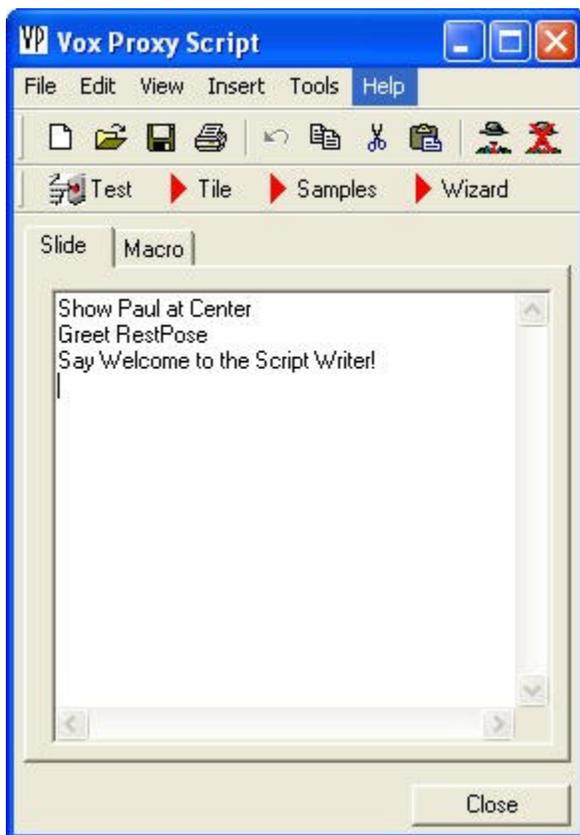
- BookMark
- GoTo
- If..Else..End
- Select..Case..Else..End

Script Writer

Script Writer Form

The Script Writer is the place where all of your script authoring takes place. It gives you access to a host of script-writing tools, including the extensive click-simple Script Wizard. As you become proficient at writing Vox Proxy scripts, you can simply type the commands directly into the script edit window. Right-clicking in the script edit window brings up the cascading menus of the Power Scripting Menu, where you will find handy shortcuts to most script-writing features.

Click on any "hot" region below for more details. The cursor will change to a hand where additional details are available.



Script Edit Window

The Script Edit window is a plain text field used for writing or editing scripts. The window resides on one of two selectable pages with tabs:

1. The SLIDE tab
If a presentation is open, this page contains the script for the active slide.
2. The MACRO tab

This page displays the text of the active macro. Macros can be opened either using the OPEN toolbar button, the File|Open menu item, or by selecting a macro from the Macro Menu. The

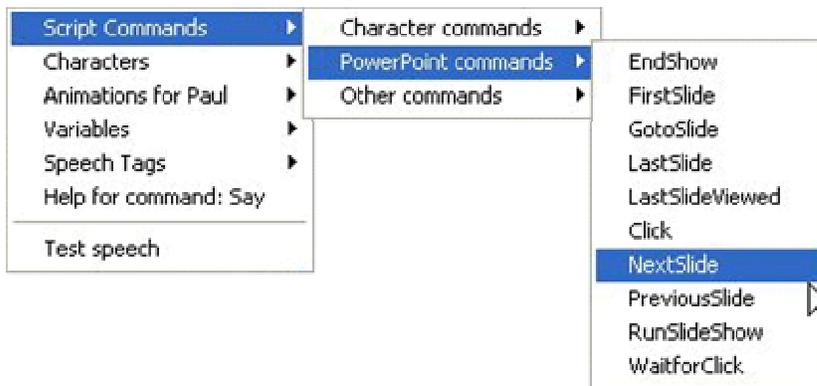
Macro Menu is opened automatically when the Macro page is selected. See Running Script Macros.

The easiest way to get start writing scripts is to play the tutorial "Your First Script". It is often helpful to examine existing scripts. Once the tutorial is finished, for example, just open the Script Writer and look at the scripts on each slide.

When you are ready to write you own, start by opening the Script Wizard. Just click on the Wizard button on the toolbar.

Power Scripting Menu

As a powerful shortcut to inserting commands into your script, you may right-click at any point in the script edit window. A context-sensitive, cascading menu called the Power Scripting Menu will appear:



The Power Scripting Menu allows you to insert into your script any of the following.

- Script Commands. This item cascades into Character commands, PowerPoint commands, and Other commands. Each script command selected will display the appropriate wizard.
- Characters. A list of available character names is displayed. This is a context-sensitive feature. When you select a character, if that character already appears in this script, it will just insert the character name. Otherwise, it will bring up the "Show Character" form.
- Variables. This item displays a list of all variables that are currently defined, along with their current values.
- Animations. This context-sensitive item displays a list of all animations for the character active at the cursor position.
- Speech tags. Displays a list of all available speech tags, such as emphasize, whisper, etc.

In addition, the Power Scripting Menu offers two additional scripting features:

- HELP for the script command on the line where the cursor appears
- TEST the current line of speech (if it is a "Say" command).

Note that if you select text in the script before right-clicking, you will see the standard "undo/ cut/ copy/ paste/ delete" menu rather than the Power Scripting Menu.

Slide Tab

The **Slide** tab displays in the script edit window the script text associated with the active PowerPoint slide. When you select the Script tab, you will see the active slide number displayed in the caption of the Script Writer dialog box.

The script text is actually saved as an invisible text box within the PowerPoint file. When you change slides in PowerPoint, the script text is automatically saved and the script for the new slide is displayed. If you move or copy slides, the script will be moved or copied with the slide.

While on the Slide tab, the save button will save the script within the presentation file. The presentation file must be saved within PowerPoint to retain the script before you close the file or the PowerPoint window.

Macro Tab

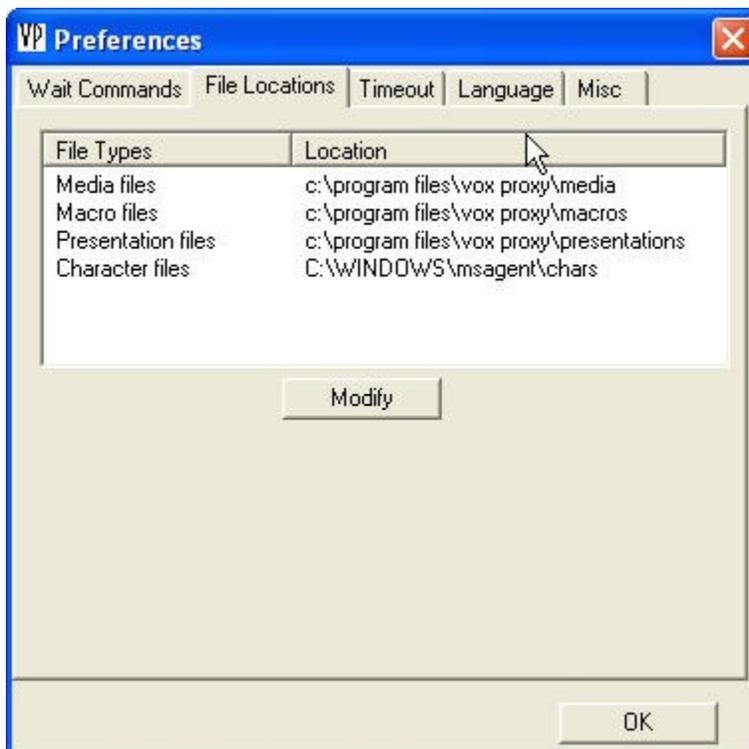
The **Macro** tab displays in the script edit window the script text associated with the selected Script Macro. When you select the Macro tab, you will see the macro name displayed in the caption of the Script Writer dialog box.

Pressing the Macro Tab also opens the Macro Menu. On this menu, script macro files may be open, edited, run, or embedded into a presentation file or other macro file.

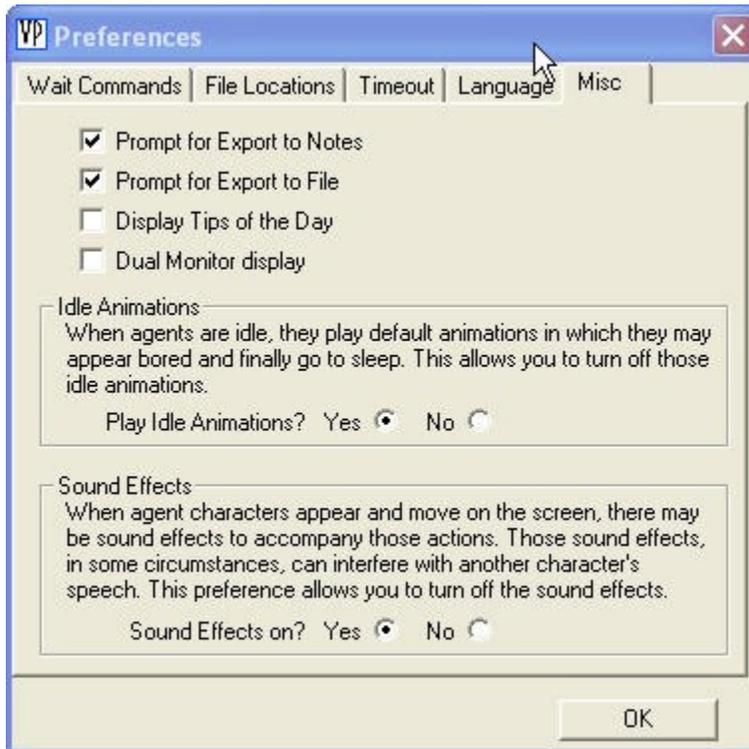
Menus

Tools

File location preferences



[Miscellaneous preferences](#)



- Prompt for export to notes
- Prompt for export to file
- Display tips of the day
- Dual-monitor display
- Idle Animations
When characters are idle on the screen, they randomly play a series of animations called "idle animations". As time goes on, they appear bored and finally go to sleep. This option allows you to turn these idle animations on or off.
- Sound Effects
Many animations (including Show and Hide) are accompanied by sounds. Those sounds can sometimes interfere with the speech of another character, if the animation is being played at the same time the other character is speaking. This option allows you to turn those sound effects on or off.
- Timeout
This Preference allows you to set the maximum amount of time that any animation will play. Its primary use is to prevent "looping animations" from running forever. Looping animations will run until stopped in one of three ways: a Stop command; an Interrupt command; or by timing out. Note that if you set this timeout too short, it may affect other non-looping animations as well.

[Preferences](#)

- **Wait Commands**

- File Locations

- **Timeout**

This Preference allows you to set the maximum amount of time that any animation will play. Its primary use is to prevent looping animations from running forever. Looping animations will run until stopped in one of three ways: a Stop command; an Interrupt command; or by timing out. Note that if you set this timeout too short, it may affect other non-looping animations as well.

- **Language**

All characters use the language specified here as their default language. You must have a Text-To-Speech engine for the default language installed on your computer. Vox Proxy does not check this. If you do not, the default will be English (Tru-Voice). You may override this default when you Show individual characters using the Language command.

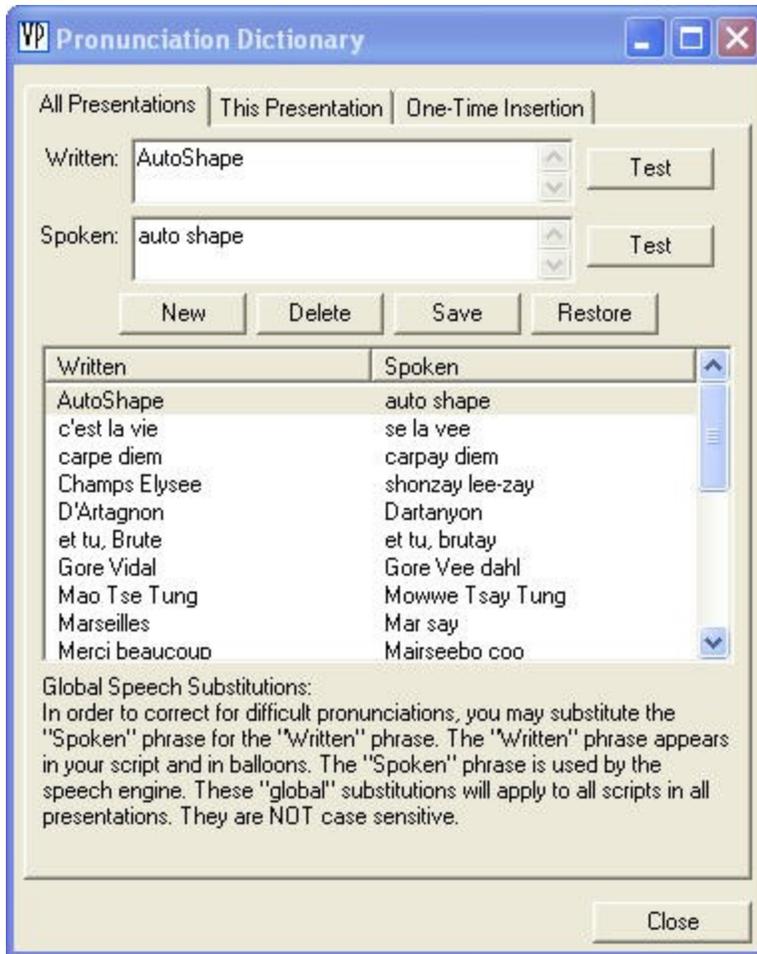
- **Miscellaneous**

Pronunciation

The Pronunciation Dictionary provides a way for you to adjust the pronunciation of special words or phrases. Whenever a character's speech encounters one of these items, the item will appear in the speech balloon as it is written in the script (the "written" words), but the character will say it as it appears in the "spoken" words. This is useful not only for foreign words or phrases, but for technical terms which may not be defined within the speech engine.

There are three pages in the Pronunciation Dictionary's editor:

- **All Presentations** These items apply to all presentations and macros.
- **This Presentation** These items apply only to the active presentation or macro. Any words or phrases defined here will override those defined on the "All Presentations" page.
- **One-Time Insertion** These words or phrases are inserted directly into the script (at the current cursor position) as one-time "map" commands. Actually, all Pronunciation Dictionary entries end up as map commands when they are actually spoken by the speech engine, but you only see those entered as one-time insertions.

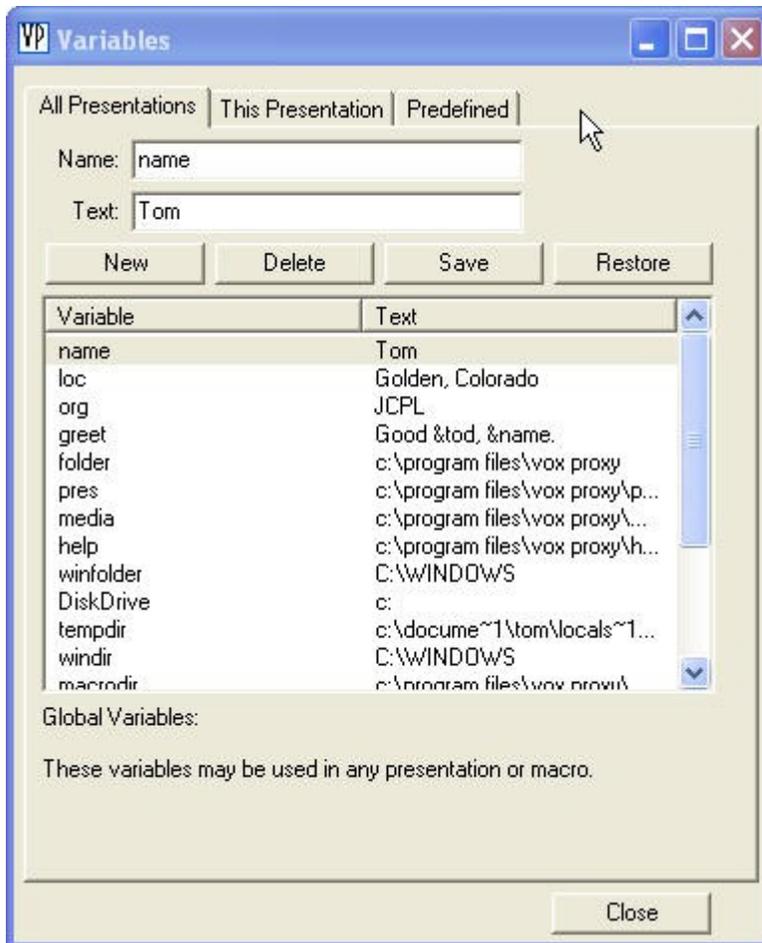


Variables Maintenance

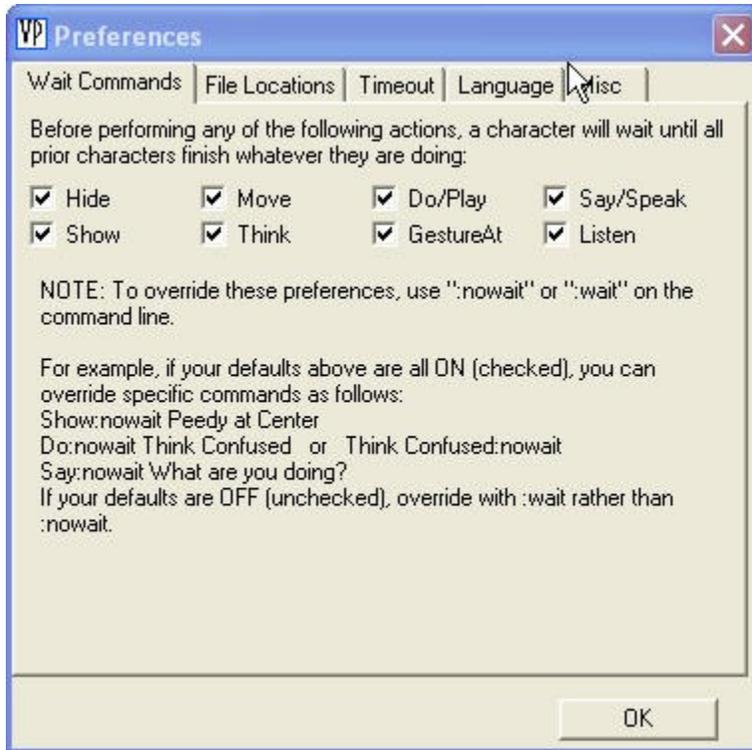
Variables are items which are used to represent text within a phrase to be spoken by a character. Some variables, such as &date, are pre-defined. They are listed below. Others may be user-defined. They are defined and edited in this menu item or by using the "Set" command in a script. Names should consist of a single word without spaces. When used in the script, names must always begin with the character "&". For examples, see the Variables section under Script Commands.

There are three pages, or tabs, on the Variables form:

- **All Presentations** These variables apply to all presentations and macros.
- **This Presentation** These variables apply only to the active presentation or macro. Any variables defined here will override those defined on the "All Presentations" page.
- **Pre-defined** These variables are available in any presentation or macro. They include time-oriented variables: &day, %date, &month, &year, &time, and &tod (time of day), and several speech tags: &w (whisper), &m (monotone), &n (normal), &r (reset), and &e (emphasize).



Wait preferences

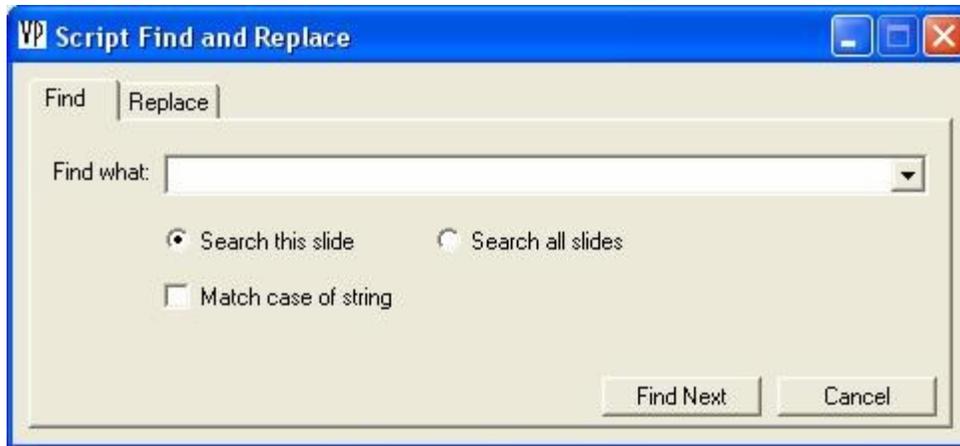


When a character is about to perform a command listed here, if that command is checked, the character will wait for the previous character, if any, to complete its speech or actions before acting on the command. If the command is not checked, it will not wait. Commands listed here include: Hide, Do, GestureAt, Listen, Moveto, Say, Show, and Think. Note that the Show command often contains an implied move (e.g. "Show Paul at Center"). Therefore, if you do not want to wait on a Show command, you should uncheck both the Show and the Moveto commands.

It is often better to use the ":nowait" parameter on individual commands than to change the system preferences here.

Edit menu

- **Undo** Undo the last insertion into the script.
- **Cut** Cut the script text currently selected. Saves into the Windows clipboard.
- **Copy** Copy the currently selected script text into the Windows clipboard.
- **Paste** Paste the text contents of the Windows clipboard into the script at the current cursor position.
- **Find and Replace**
 - The find and replace feature is also available by pressing control-F while in the Script Writer. Find and replace will work on the current slide (or macro) or on all slides in a presentation. The find and replace form appears similar to the following:



File menu

- **New** Begins a new macro script. This item is enabled only when the Macro Page is open and is disabled while viewing slide scripts. Same as pressing the "New" button on the standard toolbar.
- **Open** Open a new macro file. This item is enabled only when the Macro Page is open and is disabled while viewing slide scripts. Same as pressing the "Open" button on the standard toolbar.
- **Save** Save the current script. Same as pressing the "Save" button on the standard toolbar. Windows shortcut key is Control-S.
- **Save As** Same as Save, but allows you to specify the file name, and applies only to macros. This item is enabled only when the Macro Page is open and is disabled while viewing slide scripts.
- **Save As Presentation for VP Player** Saves a copy of the active presentation file with the file extension "PPV" (rather than "PPT"). That file may then be played from the Vox Proxy Player software by simply opening the file. It will automatically start PowerPoint and play the slide show.
- **Restore** Restore the current script from its file copy. This overlays the script currently displayed in the Script Edit window with the last saved version.
- **Page Setup** Displays a dialog box allowing changes to margins, headings, and footings used in Print.
- **Print** Print scripts. Prints either macro or slide scripts, depending on the Script Writer's page selector. When printing slides, allows selection of current slide, range of slides, or slides. Fonts can be selected, but are limited to available fixed-pitch fonts. Windows shortcut key is Control-P.
- **Exit** Closes the Script Writer.

Help menu

- **Vox Proxy Help** Displays this documentation file.

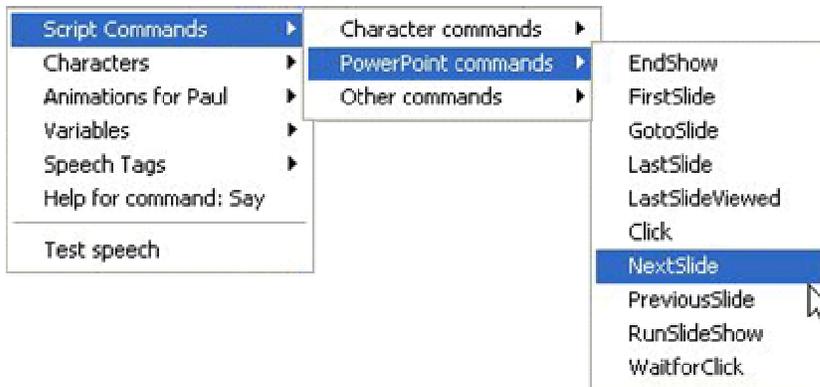
- [Tutorials](#) Displays a menu of macros which run tutorial presentation slide shows.
- [WEB Support](#) Displays several links which navigate to specified web sites such as Microsoft's PowerPoint home page.
- [Register](#) Displays a registration form showing the software owner and serial number. The Trialware version of Vox Proxy may be registered by entering the purchased software serial number here.
- [Download PowerPoint 2003 Viewer](#) Links to the Microsoft download site for the free PowerPoint 2003 viewer.
- [Readme](#) Displays the software readme file.
- [License](#) Displays a copy of the software click-license for Vox Proxy.
- [Speech](#)
 - **Install TTS Languages**

Displays a menu of TTS engines in available languages. Select a language to install the TTS engine and appropriate documentation. The documentation for Speech Normalization for the selected language is saved as a Microsoft Word® document in the speech folder (\program files\vox proxy\speech). This documentation explains pronunciation issues such as numbers, abbreviations, and so on according to the selected speech context. (see Address Context or E-mail Context).
 - **TTS Help**

Displays a Help File containing Speech Normalization information for English. See above for documentation in other languages. This documentation explains pronunciation issues such as numbers, abbreviations, and so on according to the selected speech context.
- [Tips of the Day](#) Displays the Tips form with a random tip, which also allows you to turn on or off the tips feature.
- [About Vox Proxy](#) Contains version, copyright, and registration information about the Vox Proxy software.

[Insert menu](#)

Insert displays the Power Scripting Menu, also available by right-clicking in the edit window of the Script Writer. In either case, the menu is context-sensitive to the position of the cursor in the edit window. The menu appears similar to the following:

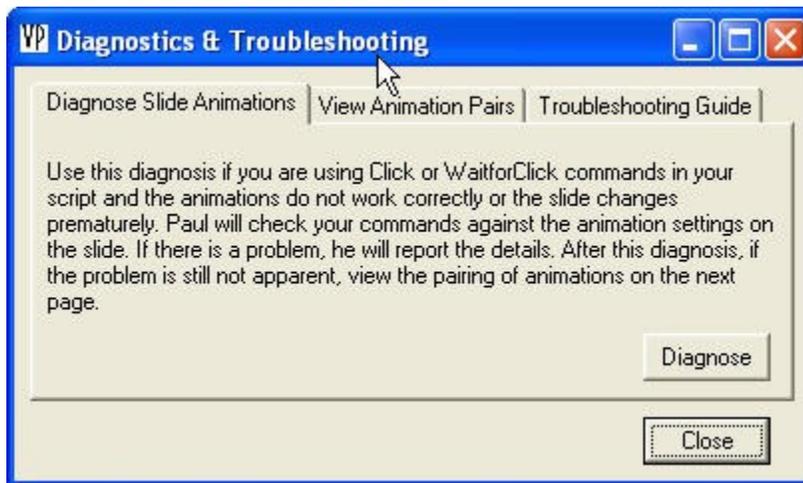


- **Script Commands** Displays a cascading list of all script commands, in three categories: Character commands, PowerPoint commands, and Other commands.
- **Characters** Displays a list of all characters. If you select a character already used in your script, it will just insert the character's name. For characters not yet in your script, it will display the "New Character" wizard.
- **Animations for [character]** Displays a list of all animations for the character that is active at the cursor position.
- **Variables** Displays a list of all variables currently defined, including the current value for each variable. Note that these values may be different when the script is run.
- **Speech Tags** Displays all available speech tags, such as emphasize.
- **Help for command** Displays help for the command at the current cursor location in the script.
- **Test Speech** Allows you to test the sound of a line of speech without playing the whole slide script.

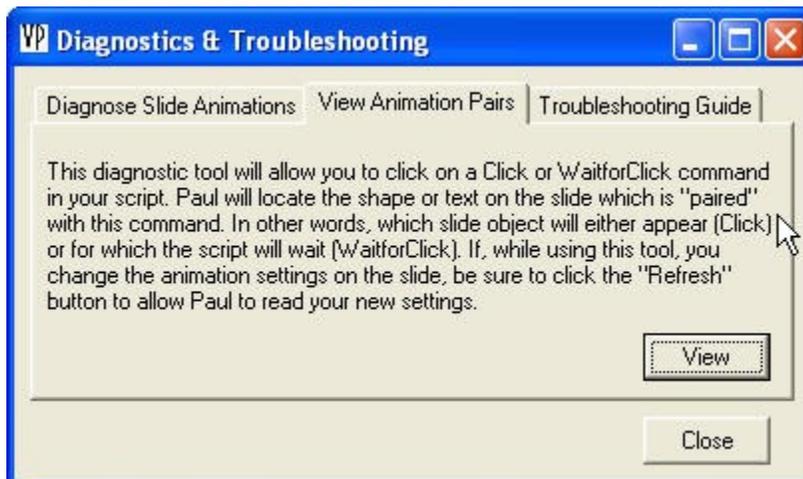
Tools menu

- **Wizard** Displays a dialog box with access to all available script commands. For a complete description and screen shot, see Script Wizard.
- **Macros** Activates the Macro Menu and switches the Script Edit Window to Macros. See Macro Menu.
- Preferences
- Variables
- Pronunciation Dictionary
- **CD Prep Wizard** If you have the optional CD Prep feature of Vox Proxy, you will see the menu item for CD Prep here. This program allows you to create self-starting CDs containing your Vox Proxy PowerPoint presentations. For detailed documentation about CD Prep, see the separate CD Prep help file.
- **Diagnose**

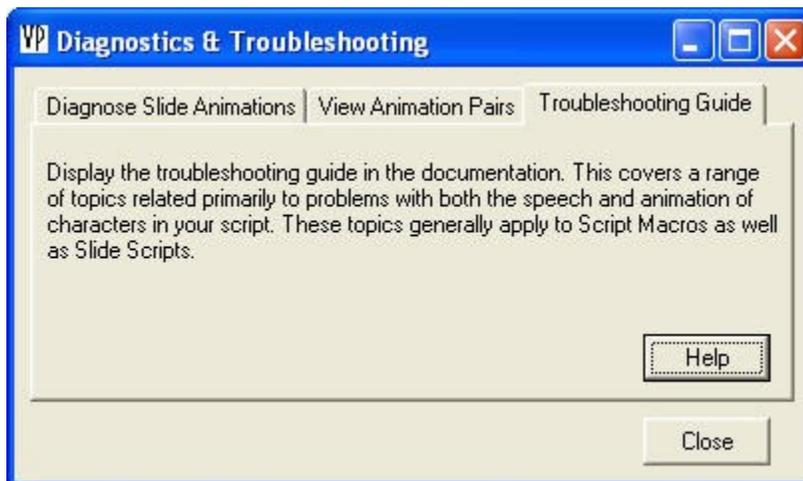
- [Diagnose Slide Animations](#)



- [View Animation Pairs](#)



- [Troubleshooting Guide](#)



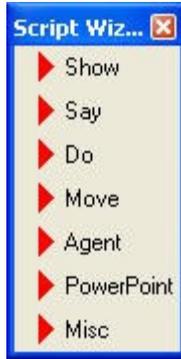
- **[Export to Notes](#)** This option will copy all slide scripts into PowerPoint's "Notes" for each slide. The purpose for this option is to allow you to print both slides and scripts together. Be aware, however, that in PowerPoint, the print program will truncate any notes that will not fit on the same page as the slide. For longer scripts, we suggest that you use PowerPoint's "Send-to" feature to send slides and notes to Microsoft Word. Word will print both slides and notes regardless of the length of the notes.
- **[Export to File](#)** This export option will copy all slide scripts into a file with the same name and in the same location as the presentation file, but with a file extension of ".VPS" (Vox Proxy Script). The VPS file is used to play slide shows under PowerPoint 97 or the PowerPoint Viewer.
- **[PowerPoint Animation Diagnostics](#)**

The diagnostic dialog box provides two diagnostic tools and a link to the troubleshooting section of this documentation:

- **Diagnose Slide Animations**
Use this option if you are having trouble using "Click" or "WaitForClick" commands. Paul will analyze your script and compare it with the Custom Animation settings on the slide, reporting any problems he may detect.
 - **View Animation Pairs**
This diagnostic tool will allow you to click on a Click or WaitForClick command in your script. Paul will locate the shape or text on the slide which is "paired" with this command. In other words, which slide object will either appear (Click) or for which the script will wait (WaitForClick).
- Convert Speech to WAV Files

[View Menu](#)

- **[Tile with PowerPoint](#)** Tiles or un-tiles the Script Writer form with the PowerPoint window. Tiling occurs on the left side of PowerPoint.
- **[Slide Script](#)** Switches the Script Edit window's active page to the slide page.
- **[Macro Script](#)** Switches the Script Edit window's active page to the macro page and activates the Macro Menu.
- Tool Bars
 - Standard
 - Tools
 - Wizard
This toolbar contains all of the features of the Script Wizard in a compact floating toolbar. For detailed documentation of Script Wizard features, see Script Wizard. The Wizard toolbar appears similar to the following:



- **Speech**
This toolbar contains many of the features of the Speech panel from the Script Wizard. When each button is pressed, it inserts the appropriate "speech tag" at the cursor position in the script. The "Test" button is convenient for testing that line of speech in the script. The toolbar appears similar to the following:



- **Emphasize**
Insert the speech tag: `\emp\` at the cursor position in the script. Emphasize the next word only.
- **Monotone**
Insert the speech tag: `\Chr="Monotone"\` at the cursor position in the script. Monotone begins at this point and ends at the end of the current line or when either the "Normal" or "Reset to Defaults" tag is encountered.
- **Whisper**
Insert the speech tag: `\Chr="Whisper"\` at the cursor position in the script. See Monotone.
- **Normal**
Insert the speech tag: `\Chr="Normal"\` at the cursor position in the script. See Monotone.
- **Reset to Defaults**
Insert the speech tag: `\Rst\` at the cursor position in the script.
- **Address Context**
Insert the speech tag: `\Ctx="Address"\` at the cursor position in the script. Any speech on the line following this tag will be pronounced using "Address context".

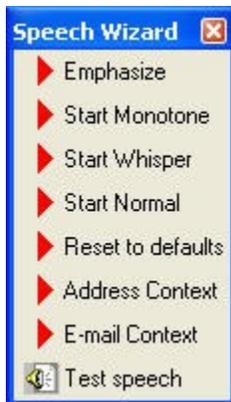
See the Speech Normalization documentation in the speech folder or on the menu Help/Speech for the appropriate language.

- **E-mail Context**
Insert the speech tag: \Ctx="E-mail\" at the cursor position in the script. Any speech on the line following this tag will be pronounced using "E-mail context". See the Speech Normalization documentation in the speech folder or on the menu Help/Speech for the appropriate language.
- **Test Speech**
Test the line of speech where the cursor is in the script. Use the character which is current at that point in the script. Prompt if the character is not visible.
- **Character Gallery**
Display a dialog box showing all available characters. Animations can be demonstrated for each character shown. See Character Gallery.
- **Close Agent Characters**
Remove any characters currently displayed on the screen.

Toolbars

Speech Toolbar

The Speech toolbar is visible as a floating toolbar. It contains many of the features of the Speech panel from the Script Wizard. When each button is pressed, it inserts the appropriate "speech tag" at the cursor position in the script. The "Test" button is convenient for testing that line of speech in the script. The toolbar appears similar to the following:



- [Emphasize](#) Insert the speech tag: \emp\ at the cursor position in the script. Emphasize the next word only.
- [Monotone](#) Insert the speech tag: \Chr="Monotone\" at the cursor position in the script. Monotone begins at this point and ends at the end of the current line or when either the "Normal" or "Reset to Defaults" tag is encountered.
- [Whisper](#) Insert the speech tag: \Chr="Whisper\" at the cursor position in the script. See Monotone.
- [Normal](#) Insert the speech tag: \Chr="Normal\" at the cursor position in the script. See Monotone.

- [Reset to Defaults](#) Insert the speech tag: \Rst\ at the cursor position in the script.
- [Address Context](#) Insert the speech tag: \Ctx="Address"\ at the cursor position in the script. Any speech on the line following this tag will be pronounced using "Address context". See the Speech Normalization documentation in the speech folder or on the menu Help/Speech for the appropriate language.
- [E-mail Context](#) Insert the speech tag: \Ctx="E-mail"\ at the cursor position in the script. Any speech on the line following this tag will be pronounced using "E-mail context". See the Speech Normalization documentation in the speech folder or on the menu Help/Speech for the appropriate language.
- [Test Speech](#) Test the line of speech where the cursor is in the script. Use the character which is current at that point in the script. Prompt if the character is not visible.

[Standard Toolbar](#)

The Standard Toolbar contains buttons for:

- New
- Open
- Save
- Print
- Undo
- Copy
- Cut
- Paste

[Tools Toolbar](#)

The Tools toolbar is visible on the Script Writer by default. It can be moved to other locations on the Script Writer form.

Individual items on the Tools toolbar are:

- [Test](#) For slide scripts, this button allows you to test the script on the current slide. The active slide will be displayed as a full-screen slide show, and the script will run. When finished (or at any time during the test), click on the slide or press <escape> to end the test and return to the normal display. Another way to test the script is to make sure that Vox Proxy is in "Run mode" (you have clicked on "Run Vox Proxy with Slide Shows"). Then simply click on PowerPoint's Slide Show View button.

- [Tile](#) Causes the Script Writer to tile or un-tile with the main PowerPoint window. The default is not tiled.
- [Samples](#) Starts a PowerPoint presentation named "samples.ppt" containing sample scripts that may be copied and pasted.
- [Wizard](#) The Wizard button opens the main Script Wizard form. All Vox Proxy script commands are available from the Script Wizard.

[Wizard](#)

The Wizard toolbar contains many of the features of the Script Wizard, but allows you to keep it visible in a floating toolbar.



- Show
- Say
- Do
- Move
- Agent
- PowerPoint
- Misc

Tips and Techniques

Interacting with PowerPoint

Vox Proxy slide scripts are automatically started as soon as the slide is displayed in a slide show. There are several techniques you can use with PowerPoint commands to cause an interaction between the characters, PowerPoint animations, and the presenter. Let's look at some examples.

Example 1

You want the active character to trigger a click or "NextAnimation" (sometimes called "next build") on the slide. A typical use for this is that the character is acting like the presenter and will bring up some "bullets" on the slide and talk about each.

Use the "Click" command in your script. A Click command acts just like clicking the mouse during your presentation. Be aware that you must set PowerPoint's "Custom Animation" for this object so that it is animated by a mouse click, **not automatic by time**.

For example:

```
Show Paul at Left Center
Point45
Click
Say Our first topic will be...
```

Example 2

You want the script to wait until you (the presenter) click to bring up the next animation. Then the script should continue to run. Use the "WaitForClick" command in your script. Each WaitForClick command must have a companion slide object with a custom animation set to appear on mouse click. Usually, this object will be off of the visible area of the screen because its purpose is not visual, but simply to hold the script until you click the mouse.

For example:

```
WaitForClick
Show Paul at Left Center
Say Wait just a minute, Tom! I have a question.
```

Example 3

You want to have a dialog with an on-screen character. Use the same technique as above to interact with characters. Again, you normally don't want visible PowerPoint animations for this purpose, so for each WaitForClick command, create an invisible companion slide object set to appear off-screen on mouse click. For this example, create two simple autoshapes (stars, let's say) and drag them off the visible slide. For each of the autoshapes, set their custom animation to appear on mouse-click. Here's your script:

```
WaitForClick
Show Peedy at Center
say Wait just a second, Tom. I have a question!
WaitForClick
Say There's something about that last point I don't understand.
Say etc...
```

Here's how it runs:

- You bring up the slide and talk your talk.
- When you're ready for Peedy to appear, you click the mouse. An off-screen slide object appears, releasing the script.
- Peedy flies in on the center of the slide and says "Wait just a second, Tom. I have a question!"
- You say something like "Hi, Peedy. What's the problem?"
- You click the mouse again. The next off-screen slide object appears, releasing the script.
- Peedy says "There's something about that last point I don't understand."
- Peedy waits
- You proceed to answer his question
- You can continue this back-and-forth if needed, using the WaitForClick commands with their companion slide objects.

[Navigating in PowerPoint](#)

If you would like the user to be able to re-visit certain slides in your presentation, PowerPoint presents us with some challenges. If those slides contain objects with custom animations, including simple bulleted text, when the user views that slide and then later tries to return to it, the custom animations will be shown as completed. In other words, the actions will not replay as they did the first time the slide was visited. This will cause problems with self-narrated slide shows if you use the Click or WaitForClick commands in your slide script.

The only way to replay a slide exactly as it originally played is to navigate to it from the slide immediately before it, either using the NextSlide command or in PowerPoint by automatically changing slides from the previous slide. Consequently, one way around the problem is to precede each slide to which the user might re-visit with a "dummy" slide that just immediately goes to the next slide. On each dummy slide, you can just set the slide transition in PowerPoint to advance after zero seconds. Alternatively, you could use a VP script containing just the NextSlide command.

Under some circumstances, you may wish to use variables with this navigation. For example, on each dummy slide, you could assign the current slide number to a variable:

```
Set &S=&SlideNumber
```

Then, to return to that slide from a subsequent slide, you could use:

```
GotoSlide &S
```

[Scripting Speech](#)

Synthetic speech technology using free TTS engines like those included with Vox Proxy will not fool people into thinking that a live person is talking. On the other hand, they can be amazingly effective and, with a little extra effort, you can at least achieve correct pronunciation and even convey reasonable variations in emphasis. This creates a far more effective presentation.

Be sure to play the tutorial on speech. It provides some good illustrations using the tools in Vox Proxy.

The most convenient tool for scripting speech in Vox Proxy is the Speech Toolbar. Find it in the View/Toolbars menu. Using this toolbar, you can place the cursor on any line of speech in your script and press the "Test Speech" button to see how it sounds.

Punctuation in speech affects both timing and emphasis. Commas, semicolons, and periods cause a pause in the speech and slightly affect the pronunciation of the immediately preceding word. A question mark also affects the emphasis of the preceding word. An exclamation point can affect the pronunciation of an entire sentence. Using the "Test Speech" button, experiment with the sound of each sentence.

Methods for changing pronunciation and emphasis in speech will depend on whether you are displaying the speech in a speech balloon. If you are not, then you can be free to use incorrect punctuation or even spelling in order to adjust the sound of the speech. With speech balloons, you can still use variations in punctuation and spelling, but you will need Vox Proxy's Pronunciation Dictionary in order to achieve variations in pronunciation while preserving correct spelling and punctuation.

Use the Pronunciation Dictionary in the following ways:

- For foreign-language words or commonly used "made-up" words that are consistently mispronounced by the TTS engine, enter them on the "All Presentations" page. We have included a number of such words such as "autoshape" and "c'est la vie" as samples. Any time Vox Proxy encounters one of these words or phrases in a line of speech, it will automatically use your substitute pronunciation. Look over and test these examples to get ideas for your own pronunciation adjustments.
- For words that are needed only in the current presentation, but throughout it, enter them on the "This Presentation" page. Be aware that such entries will cause a special slide to be created to contain that information. The special slide will be marked "hidden" so that it won't be displayed in the slide show.
- For one-time pronunciation adjustments, use the "One-time Insertion". This will insert a "map" tag into the line of speech showing both what is to appear in the speech balloon and how it is to be pronounced. You can be creative with using this feature. For example, you can insert pauses into the speech by using commas that don't show up in the speech balloon.
- You will occasionally have to adjust the pronunciation of a word whose emphasis is wrong for the current context. The TTS engine, for example, does an amazingly good job of determining when the word "object" should be pronounced with the emphasis on the first syllable and when it should be on the second. But it isn't perfect. ("perfect" is another one of those words!) When you test these words in the Pronunciation Dictionary, you may have to enter several words or the whole sentence in order to determine how the TTS engine will pronounce it.

Many other tools are available to work with pronunciation. See individual descriptions for the Power Scripting Menu and the Speech Toolbar. Using speech tags, you can emphasize a specific word or change the "character" of speech (whisper and monotone). Also on the Speech Toolbar are "Context" buttons called "address" and "E-mail". For U.S. English, these buttons are the same. Context primarily affects the way the engine speaks numbers, punctuation, and abbreviations. Web addresses, for example, require use of the context tag. In order to review these pronunciation rules in detail, see Help/Speech/TTS Help on the Script Writer menu.

Additional tools are available in the Script Wizard's "Say Something" button. There, you can vary the speed of speaking as well as its pitch.

A couple of other tips that may be somewhat obscure:

- The "Speak" command works exactly like "Say" except that it bypasses the Pronunciation Dictionary. If you want to illustrate a mispronounced phrase, for example, you might use the same line of text with both commands:

```
Say C'est la vie, senior Peedy. We have no rellenos.  
Speak C'est la vie, senior Peedy. We have no rellenos.
```

- The vertical bar character, also called "style" is used to separate alternative phrases to be spoken. The TTS engine will randomly choose to speak only one of the phrases. Suppose you have a simple macro, for example, which you might call on multiple slides when the character is finished and you can proceed to the next slide. A typical slide script might be:

```
...script  
...  
RunMacro "Next Slide"
```

The macro script for "Next Slide" might be:

```
Say You can change slides now | Go ahead to the next slide | It's OK to  
change slides now | Next slide | I'll wait for you to change slides
```

Another use would be in working with speech commands. If the user's input is not recognized, you might want the character to respond with:

```
Say Sorry, I didn't understand that | What did you say? | What was that? |  
Huh? | Please stick to words that I understand
```

A little care in testing and adjusting your characters' speech will go a long way in improving the effectiveness of your presentation.

Using Multiple Characters

Interaction between characters is one of the most effective tools for creating interest and avoiding a boring presentation, but it also presents many challenges. The greatest challenges, of course, are in the script's dialog. But the more technical issues of making two or more characters interact effectively is also important. Of these, timing is one of the more important. To understand the timing of characters, you must understand that, in order to simplify the scripts, the default in Vox Proxy is that each character must wait until the previous character finishes its speech and actions before proceeding with its own. This ensures that nothing unexpected will happen when the script runs, but it can result in interactions that may appear slow and stilted.

Vox Proxy has several tools to help you override these defaults and to create more effective and interesting interactions. One of these is the parameter `:nowait`, which can be used to override the wait default on individual lines of speech or actions.

1. Make multiple characters appear and hide simultaneously. This is easily accomplished using the `:nowait` parameter. For example:

```
Show Paul at left center  
Show:nowait Peedy at Center  
Show:nowait Merlin at right center
```

Hiding characters simultaneously is the same except for one thing. It is a good policy to place a "wait" in the script just before the first character hides. Otherwise, the `:nowait` could cause the second and any additional characters to hide too soon.

```
...previous script lines..  
Paul say Bye-bye!  
Wait  
Paul hide  
Peedy hide:nowait  
Merlin hide:nowait
```

Without the Wait, Peedy and Merlin would hide while Paul is still speaking.

Another trick to watch for is that some characters will take longer than others to hide. If your slide contains a "NextSlide" command immediately after several characters hide, you should make the character with the longest hide animation hide last. Otherwise, the slide change will occur before the characters are fully hidden and they can re-appear on the following slide.

2. Let a character start speaking while another is moving. There is only one tricky thing to be aware of when using ":nowait" for speech. Many animations are accompanied by sound effects. These sounds can "step on" the speech of another character. The general rule is that the speech engine will not allow two characters to have simultaneous sounds or speech. And usually, the first sound to be encountered will suppress any other character's sound that tries to play after it starts.

There are two ways around this when using speech in Vox Proxy. One is to shut off animation sound effects. This can be accomplished either on the "Preference" menu item or by using the "SoundEffects off" command in your script. The other way is just to make sure that the speech starts BEFORE the animation sound effect. Consider these two versions of the same script:

```
Show Merlin at right center
Show:nowait Paul at Center
Merlin search
Paul say:nowait Hey, Merlin! What are you looking for?
Show Paul at Center
Show:nowait Merlin at right center
Paul say Hey, Merlin! What are you looking for?
Merlin search:nowait
```

Merlin's "search" animation is accompanied by the sound of bells. In the first version, since Merlin begins his animation before Paul starts to speak, the bells suppress Paul's speech. In the second version, Paul's speech starts first and the bells are suppressed.

Overlapping sounds are not a problem with silent animations. ":nowait" is often helpful in these cases to speed up processing:

```
Paul GestureRight
Say Our first item of business is ...blah blah
Wait
RestPose
Merlin GestureLeft:nowait
Say And second, we want to talk about blah blah..
```

It's a good idea to make Paul return to his rest pose, but we don't want Merlin to wait until Paul is resting before he starts his own gesture, so we use the ":nowait" parameter. The reason for the Wait command above is that it tells Merlin not to begin gesturing until Paul has finished speaking.

Troubleshooting

Character Behavior

Speech

I have no speech at all: characters' lips move, but they do not speak.

The most likely problem is that your sound system is turned down or off. Test this by seeing if you can play a "wav" file. From Windows Explorer, go to:

\program files\Vox Proxy\media

Double-click on the file: CLAP.WAV

Do you hear sound? If not, then your sound system is not playing sounds. Check the "Mute" button on the Windows volume control; check that the volume is not all the way down; check that your speakers are properly connected.

If you do hear sound, re-check the speech. If you still do not hear speech, try manually installing the speech (TTS) engine: In Windows Explorer, go to:

\program files\Vox Proxy\speech

Double-click on the file: TV_ENUA.EXE

This will re-install the default US English speech engine.

If the speech still does not work, make sure that the problem is not an isolated, single line of speech. Test the character with several consecutive lines of speech.

A specific line of speech fails, but remaining lines speak correctly.

This is caused by another sound "stepping on" the line of speech. The speech (TTS) engine will remain silent if another sound begins just prior to the speech and is not finished by the time the speech begins. It is most often caused by another character performing an animation which has sound effects just prior to the failing line of speech. If you're using the ":nowait" parameter, remove it and re-test. If not, try reversing the order of the actions.

Animations

A character's animation begins too soon.

This is almost always caused by a ":nowait" parameter. For example:

```
Show Peedy ...
Show Paul ...
Peedy say Hello. My name is Peedy.
Greet
Paul Greet:nowait
Paul will bow before Peedy does...in fact, he will bow just when Peedy begins
to speak. In order to have Paul bow at the same time as Peedy, place a
"wait" command just above Peedy's Greet:
Peedy say Hello. My name is Peedy.
wait
Greet
Paul Greet:nowait
```

Now Paul will wait for Peedy to stop speaking before he bows.

A specific animation does not work – the remainder of the script works fine.

Check to see that you are using a valid animation command for the character. Not all characters have the same list of animations. Also, some characters' list of animations includes some which do not work: they are simply ignored. Test the animation using the "Preview" button on the Character Gallery. If it works there, it will work in your script. Also check that your spelling is correct. Capitalization is not important, but spelling is.

Performance

When I run my slide show, the animations and/or speech are choppy and intermittent.

Either the computer is too slow, has too little RAM, or, most likely, you are running something else that is CPU-intensive. Both character animations and speech take considerable system resources, so things like animated-GIFs running on a slide can cause character animations and speech to be slow.

During my slide show, the character nearly stops, then sporadically makes some movement or sound.

In PowerPoint 2002, you must be sure that you have the feature called "Show all scripts" turned ON. You can tell if it's on by the small amber icon (called a "script anchor") that appears in the middle of any slides with scripts. If you do NOT see that icon, you must turn "Show all scripts" on. Open PowerPoint help and, in the answer wizard, search for script anchor. Click on "Show or hide anchors on a web page" and follow the instructions to add the "Show all scripts" item to your menu. Then be sure it is turned on.

During my slide show, the character nearly stops, then sporadically makes some movement or sound.

In PowerPoint 2002 or newer, you must be sure that you have the feature called "Show all scripts" turned ON. You can tell if it's on by the small amber icon (called a "script anchor") that appears in the middle of any slides with scripts. If you do NOT see that icon, you must turn "Show all scripts" on. The problem is that this feature does not appear in the default PowerPoint menus. Here's how you add it:

- In PowerPoint, click Tools/Customize
- Click on the Commands page (it is probably the default)
- In the "Categories" column, select "Tools"
- You will drag-and-drop the new menu item onto your Tools/Macro menu. In the "Commands" column, click on "Show All Scripts" and HOLD DOWN THE MOUSE BUTTON while you drag the command up onto PowerPoint's "Tools" menu. The tools menu will drop down. Continue to hold the mouse button and move down to the "macro" menu. Move the cursor over the macro menu and release the button when you are pointing below the "Security" item. The Show All Scripts menu item will now appear on your Tools/Macro menu.

After adding the menu item, make sure that Show All Scripts is turned ON. It will show a check-mark next to it when it is on. This will solve the halting slide show performance problems.

Miscellaneous

Whenever I open the Help files in Vox Proxy, I get either a Dialer or "Work Offline" dialog box.

Vox Proxy Help is in HTML and uses Microsoft Internet Explorer for its display. If you have a web url address set as your "home" page in Internet Explorer and you are not connected to the Internet, then Internet Explorer will prompt to connect. You can avoid the connection dialog box by setting the home page to "Blank" in Internet Explorer Tools/Internet Options/General/Home page.

PowerPoint Behavior

During a slide show, either a slide changes too soon or the wrong slide animation appears.

Your script contains more "Click" commands than there are animations on the slide. Remember that the Custom Animation must be set to appear "on mouse click" and not automatically by time. Use the "Diagnose" button on the Script Writer to check your script animation commands for consistency with your slide animations.

When I go to a specific slide during a slide show, a character appears in the upper-left corner of the screen rather than where it is supposed to.

Each slide script must stand alone if it is to be restartable. At the beginning of each slide script, be sure to put a "Show" command to place each character on the screen. Otherwise, the slide show will work correctly if run continuously, but may not if you skip to a specific slide.

One or more characters that "Hide" at the end of a slide reappear as the next slide begins.

This is usually caused by hiding several characters at the same time at the end of a slide. The "NextSlide" command waits for the LAST character to finish hiding before changing slides, but if an earlier character has a slower Hide animation, it may not be completed when the slide changes, so he may reappear. The solution is to rearrange the order in which the characters hide, making the slowest one hide last.

If the character reappears in the middle of a script, then you have inadvertently used a "hide" animation rather than a Hide command. Vox Proxy always interprets "Hide" as a command unless it appears on the same line as another animation. To avoid this problem, always use "Hide" on a separate line by itself.

I see a small amber icon in the middle of my slide. How do I get rid of it?

This icon is a new feature in PowerPoint 2002 and 2003 called a "Script anchor". Normally used for web scripts, It appears on any slide containing a Vox Proxy script. This anchor must be visible for your scripts to work, but you can simply drag it off the visible area of the slide to "hide" it.

CD Prep

Introduction

The purpose of the CD Prep program is to create a CD-R that will auto-start on an end-user's computer and run slide shows with Vox Proxy scripts without requiring the user to have first installed the Vox Proxy Player. The CD Prep program creates a "CD image" folder on your hard drive that contains all of the files and folders to be placed on the CD. You will use your CD-burner software to copy those files and folders to the CD.

The Vox Proxy CD Prep program supports running PowerPoint presentations and/or Vox Proxy Script Macros. CDs created with CD Prep can be configured to run automatically from the PowerPoint 2003 Viewer, bundled on the CD, so that it will run the same whether the user has PowerPoint or not.

Please be aware that there are many issues involving the end-user's computer and configuration which could cause the presentations on the CD not to work properly. Some of those are discussed below.

You should also be aware that the CD you will distribute is not a zero-footprint application. Although your presentation files and the Vox Proxy application software are not installed on the user's computer, there are certain speech components and Microsoft Agent files that must be. Because of this, the user must have administrator rights to play a CD Prep disk. See the installation details below for more information about what is installed on the end-user's computer.

What's new in Version 3

Version 3 contains many enhancements to CD Prep, all designed to make it easier for you to generate self-starting CDs.

Quick Prep

If you have a single presentation file (rather than a menu of multiple presentations), and do not need info/author files or changes to the standard copyright notices, you can use a new feature called Quick Prep, which will prepare everything you need for the CD in a single step.

Automatic File Scanning

CD Prep will now scan through the scripts in your presentation file looking for files that you call, including media, macros, other presentations, etc. It will automatically include those files on the CD and alter the scripts if necessary to change all paths to the appropriate file locations. It also locates and includes any characters used in your scripts. The automatic scanning for files works whether or not you use Quick Prep.

Optional Info and Author buttons

The "info" and "author" buttons displayed on the dialog box shown when a CD starts are now optional.

PowerPoint 2003 Viewer

You can specify that the CD prepared with CD Prep should always play on the PPT 2003 viewer, even if the user has PowerPoint. This provides a consistent presentation that does not depend on the version of PowerPoint that the user may have.

PowerPoint Package for CD

If you use the new "Package for CD" feature in PowerPoint 2003 and specify that folder as your CD Image folder, CD Prep will automatically re-configure the files for use with Vox Proxy.

Presentation Guidelines

When preparing a presentation and Vox Proxy scripts for delivery on a CD, it is very important that any files used in the presentation (such as media files) will be locatable on the CD. This generally means that you cannot "hard-code" the location (the directory path) of any files, because you do not know what the end-user's CD drive letter will be.

CD Prep will automatically locate any hard-coded paths in your script to the appropriate variables. However, in the event that you want to locate files in CD folders other than those assumed in CD Prep, Vox Proxy provides a number of variables for this purpose.

- **&DiskDrive** – The drive where the CD is playing. Example: "D:"
- **&folder** – The folder where your Vox Proxy program files are located.
 - This corresponds to "\\program files\vox proxy" on your system.
- **&mediadir** – Vox Proxy's media folder. This corresponds to: {program files}\vox proxy\media on your system.
- **&tempdir** – The windows temp folder on the end-user's computer (where the CD is playing). Any files listed in the "temp" folder on the CD will be copied into this folder when the CD starts and deleted when the CD ends.
- **&presdir** – Vox Proxy's presentations folder. This corresponds to: {program files}\vox proxy\presentations on your system.
- **&windir** – The windows folder on the computer where the CD is playing.
- **&pptpath** – The folder where the PowerPoint presentation resides on the computer where the CD is playing. This would normally be: &DiskDrive\presentations

The PowerPoint 2003 Viewer

Presentations run on the CD can support PowerPoint 97 or the PowerPoint 2003 Viewer. Vox Proxy scripts, however, cannot be accessed from within the presentation file by these programs. Instead, it uses a companion script text file with a file extension of "VPS". If you have not manually run the "Export to file" feature on the Tools menu of the Script Writer to create this companion file, CD Prep will run it and prompt you.

On the "finish" page of the CD Prep wizard, you may opt to require that the CD play on the PowerPoint 2003 Viewer, which will be bundled on the CD. For consistency of playback, we strongly recommend that you use this feature. If you do not, when the end user starts the CD, the software will detect whether the user's computer has access to PowerPoint. If so, it will use whatever version the user has. If not, it will automatically use the PowerPoint 2003 Viewer.

The author should be aware that each version of PowerPoint has many features which are not supported in prior versions. You must design your presentations accordingly if you want them to work properly with all versions. The new PowerPoint 2003 Viewer offers a consistent play-

back, and therefore alleviates this problem, so we strongly recommend that you require the CD always play on the viewer.

One important feature of Vox Proxy will not work in PowerPoint 97: the WaitforClick command. As an alternative to WaitforClick, split your slide into duplicate slides at the points where you want to wait for a click. WaitforClick WILL work in the PowerPoint 2003 Viewer.

The PowerPoint Viewer does not support slide ID numbers. In order to work in the Viewer, be sure to use slide numbers rather than slide IDs in any "GotoSlide" commands.

Minimum Requirements for Playback

The computer on which the CD will be played must have Windows 98SE, Me, NT 4.0, 2000 Professional, or XP. It will NOT play correctly under Windows 3.1. The PowerPoint 2003 Viewer requires Windows 98 SE or newer, so if you anticipate supporting 95 or early versions of 98, do not force the CD to run on the 2003 Viewer.

Other minimum requirements:

Disk space: In addition to your PowerPoint files, at least 8MB plus the size of any characters you include plus any media or other files you use.

Recommended:

RAM: 128MB

CPU: 500MHz P-III or faster

How the CD will Run

When the end-user inserts the CD in a drive, and assuming that the user has not turned off the auto-start feature of their drive in Windows, the CD will autostart and perform the following operations.

First, a window will be displayed giving the user an opportunity to continue or abort and quit the CD.



At this point, no installation information has been checked. The user may display your optional "Author" and "Info" files, where you can explain what's on the CD and provide contact information, instructions, or other materials.

If the user clicks No, the program will terminate.

If the user clicks Yes, Vox Proxy will start and run from the CD. It will first check the user's computer for the following items, install if necessary, and copy certain files. Neither the Vox Proxy software nor your presentation files will be installed to the end-user's computer, but will actually run from the CD. Certain Microsoft Agent and speech features, however, must be installed on the user's computer. These include the Agent control (which is pre-installed on almost all versions of Windows), any Text-To-Speech engines needed, and the character files. The space needed on the user's computer is minimal, a little more than the size of the character files you are using.

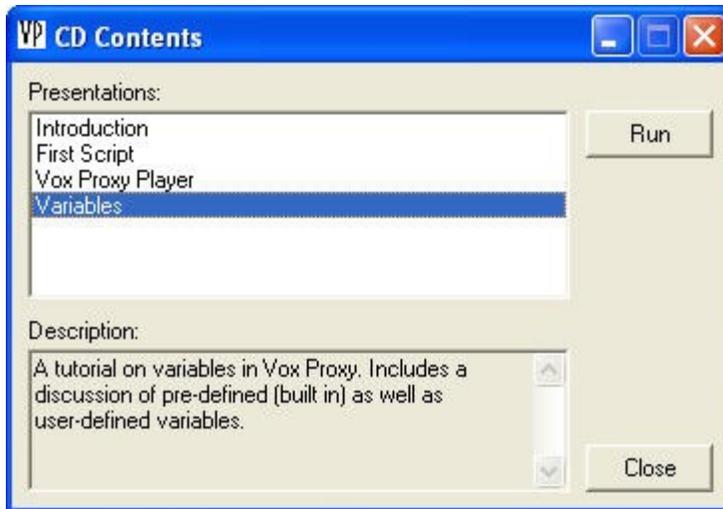
During the installation process, the user will see the following:



This indicates that the software is checking for necessary speech and Agent components and will install whatever it needs, as follows:

1. Microsoft Speech API version 4. If not found, it will install from the file "spchapi.exe". The speech API is normally installed on any Windows system containing a sound card, but on Windows XP, the speech API must be installed.
2. Microsoft Agent control. If not found, it will install from the file "msagent.exe". This should almost never happen, because the agent control has been a standard feature in Windows since at least Win-98.
3. Languages (TTS engines). Checks for the TTS engines which you have specified in the CD Prep program. Installs if necessary from the files included in your setup. This will usually be necessary, because the "TruVoice" TTS used by all of Vox Proxy's standard characters does not come pre-installed in Windows. The setup file for TruVoice is "tv_enua.exe". For other languages, the setup files begin with "lstts" and "agtx".
4. Speech Recognition Engine. Installs only if you need it, according to your setup. If installed, the setup file is "actnc.exe".
5. Character files: copies onto the user's hard drive any character files which you have selected in your setup. These files are identified by the file extension "ACS" and are copied into the folder:
 \windows\msagent\chars (for Windows 95, 98, and ME)
 or
 \WINNT\msagent\chars (for Windows NT, 2000, and XP)

Vox Proxy will start PowerPoint and display your menu of presentations or, if you have configured it for only a single presentation, open your presentation file from the CD, and start the slide show. A sample menu is illustrated here:



When the presentation is complete, Vox Proxy will return to your presentation menu. PowerPoint will remain open on the end-user's computer. If you have specified only a single presentation, there will be no menu and the program will terminate when the slide show is ended.

If the same user either reruns the CD or runs other CDs from you in the future, they will start much faster because the speech files will have already been installed.

Administrative Rights and other XP Issues

On systems where it is necessary to install any speech components (as described in "How the CD will run"), the user must have administrator rights to permit those installations. On Windows 95, 98, and ME, those rights are always available, but on NT, 2000, and XP, they may or may not be. In labeling your CD, and in any user documentation included with your CD, you should make it clear that the user should have administrator rights in order to run the CD.

Note that these installations are required only the first time the CD is played. Any subsequent playing can be done without administrator rights. Subsequent CDs which you send out will also not require administrative rights unless you include a TTS engine not included on a previous CD.

Windows XP contains a version of the Microsoft speech API (SAPI-5) which is not backward compatible, and is not compatible with the speech supported by Microsoft Agent (SAPI-4). Vox Proxy's CD Prep program will automatically install SAPI-4. Windows will support both versions concurrently without conflict. If users report that your presentations run and the characters appear but do not speak, you should advise them to reboot and reinsert the CD.

The CD Prep Wizard

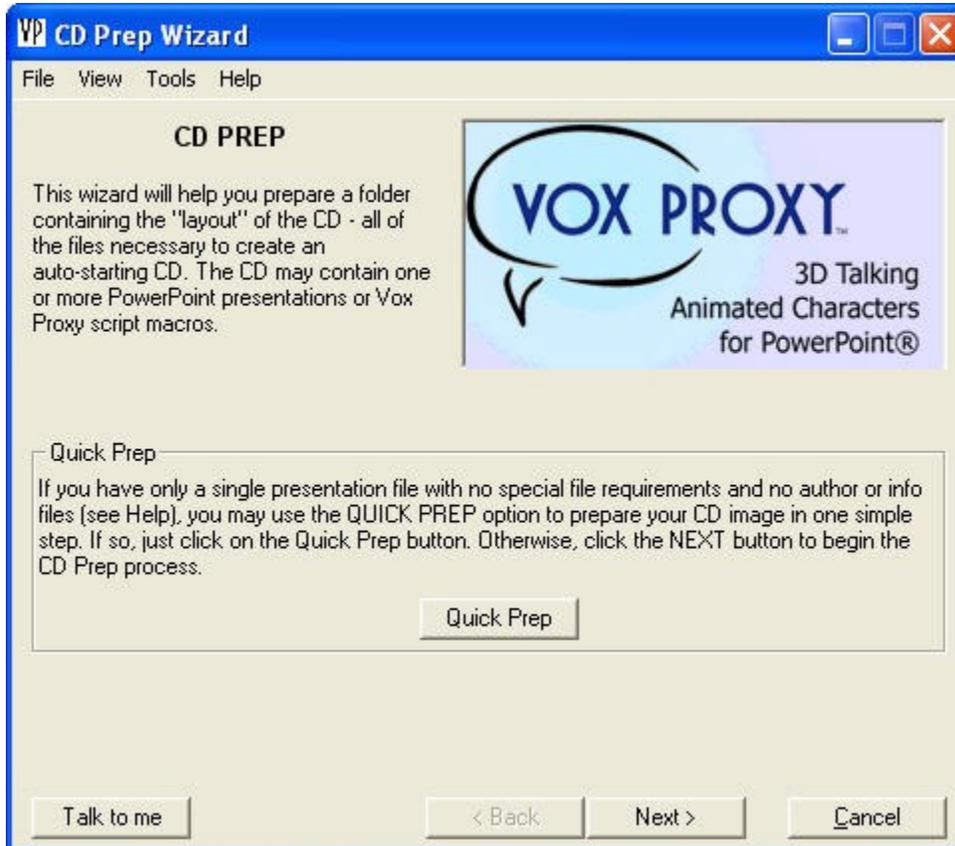
Introduction

The CD Prep Wizard will guide you through each step of the process needed to create a "CD Image Folder" containing all of the files necessary for an auto-starting CD. You may click "Back" or "Next" at any time to review previous information entered. You can also jump to a specific page by clicking on the View menu item.

On any page of the CD Prep Wizard, you may have Paul give you a brief description of the information required on that page by pressing the "Talk to me" button.

CD Layout

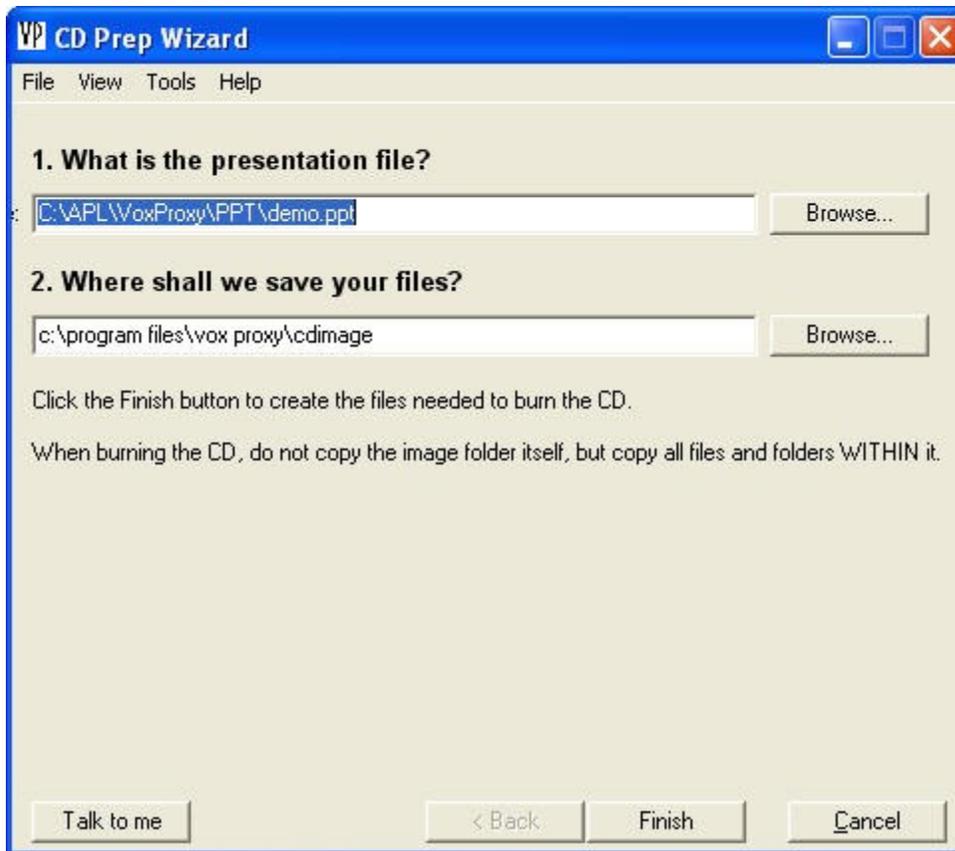
When you run the CD Prep program, you are determining the "layout" of the CD, including files, menu information, and documentation. At the finish of this process, you will be asked for a "layout name". All of the information about the CD layout will be saved under this name. In the future, if you want to modify the layout and re-create it, you can open the layout by clicking File/Open.



Here you may opt to run Quick Prep, which will create the CD image folder in one easy step. You should NOT run Quick Prep if you need to set up "info" and "author" buttons or edit the copyright notices.

QuickPrep

If you select the Quick Prep option, you will go directly to a special finish screen that appears similar to the following:



Just confirm that the current presentation file is the one you want to use, tell it the folder name in which to save all files, and click Finish. That's it!

CD Image Folder

When you finish setting up a CD layout, the actual files and data are saved in a folder called the "CD Image folder" on your hard drive. You will then use your CD-R burner software to copy the contents of this image folder onto the CD.

If you specify an existing folder name, the program will not delete the contents of this folder when it copies the CD layout files into it, so you should be sure that it does not contain extraneous files or sub-folders which you don't want on the CD.

CD Prep will check the CD image folder you specify to see if it appears to have been created by PowerPoint's "Package for CD" feature and incorporate those files appropriately.

Single Presentation or Menu

Specify here whether the CD will contain only a single presentation or multiple presentations. For a single presentation, select the presentation file here. When the CD starts, the slide show will automatically start and run. When it is finished, the program will terminate.

For multiple presentations, you will create a menu which will be displayed to the user. After each presentation is played, the menu will reappear.

Menu Builder

If the CD will contain multiple presentations, this is where you build the menu which the user will see and which will control the files to be played.



For each presentation on the menu, you must specify three things:

1. **The Menu text** - what the user will see as the menu item
2. **The presentation file** - this can be either a PowerPoint file (.ppt or .pps) or a Vox Proxy script macro file (VPs)
3. **Description** - this description will be shown to the user when the presentation is selected on the user menu

CD Files

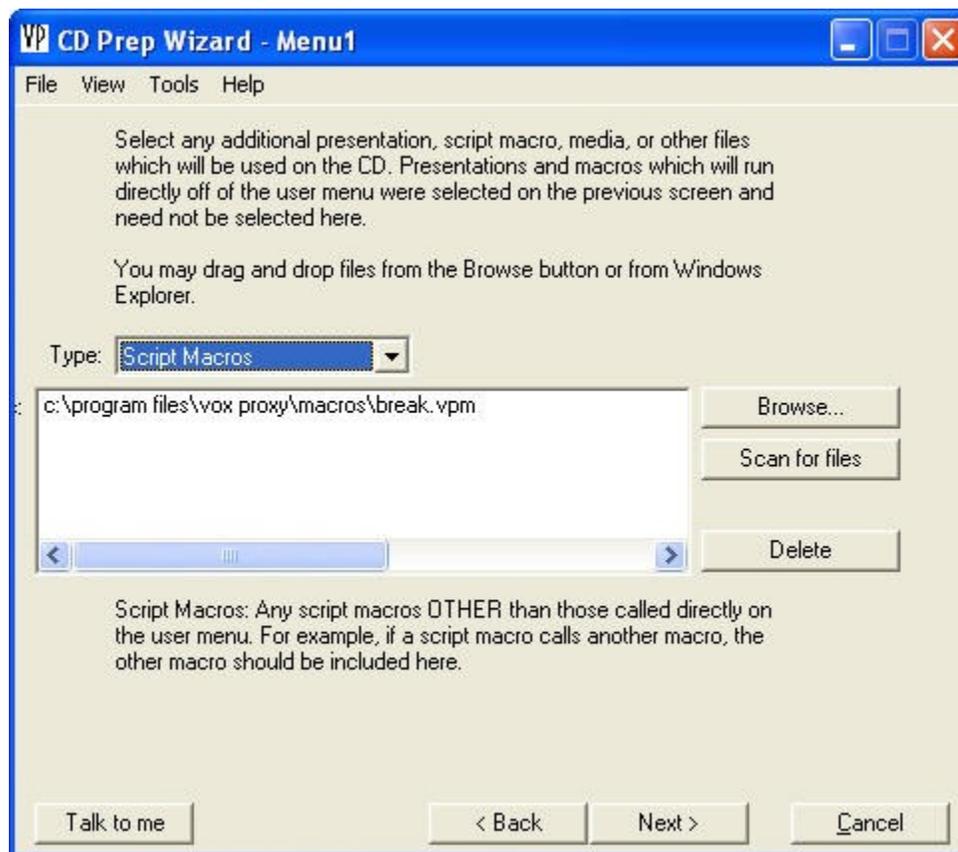
There may be a number of files needed on the CD other than the presentation files themselves. CDPrep will automatically scan your presentation file(s) for file references and show them here.

Five kinds of files are grouped by their location on the CD:

1. **PowerPoint presentations** This may include ".pps" or ".ppt" files. Do NOT include files already specified on the Menu Builder. You may, however, include PowerPoint slide shows which are run from a script macro. All of these files will be placed in the "presentations" folder on the CD.

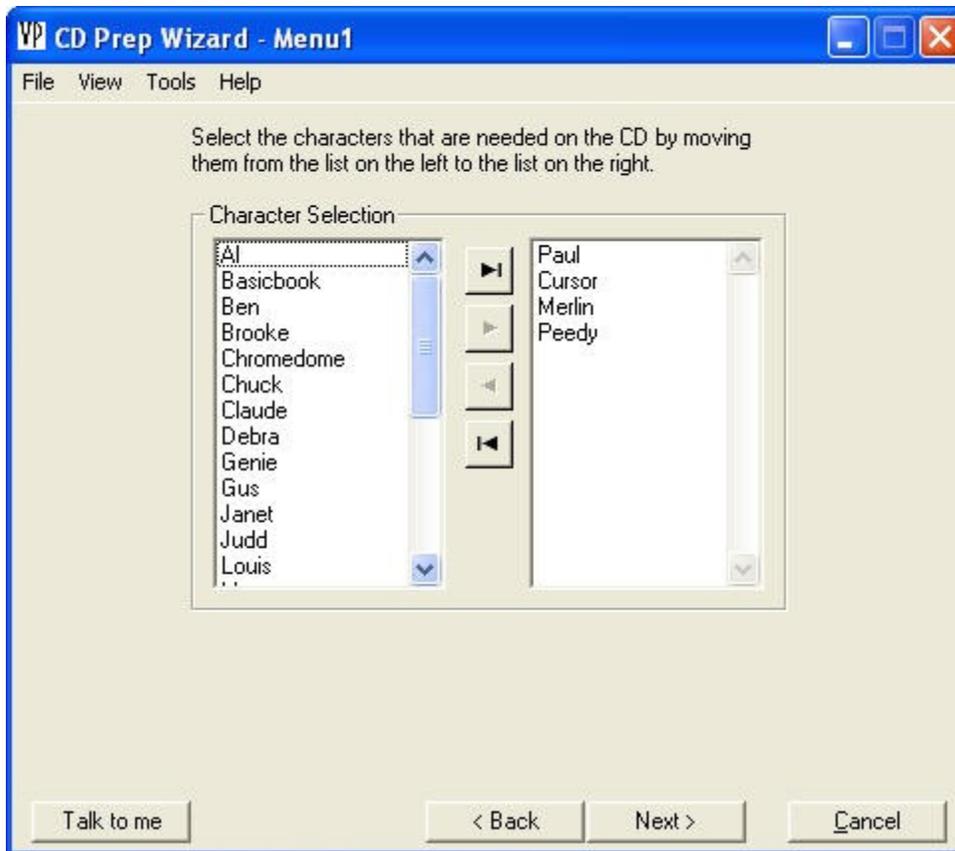
2. **Script Macros.** Include any Vox Proxy script macros here. These files must have the extension ".vpm". They will be placed in the "macros" folder on the CD.
3. **Media files.** Include any media files which are called in your script macros and are not embedded in a PowerPoint presentation. Any files compatible with Microsoft Media Player can be included. These files will be placed in the "media" folder on the CD.
4. **Temp files.** Any files listed here will be copied to the Windows temp folder on the end-user's hard drive. When the presentation is complete, these files will be deleted from the user's hard drive. This feature allows media or files to be rapidly accessed during the presentation. Files accessed from the CD drive can be slow to load after the drive stops spinning. If you use media files copied here, be sure to use the variable %tempdir rather than %media in your script.
5. **Other files.** Any ancillary files may be included here. They will be placed in the "other" folder on the CD.

When selecting files, you can drag-and-drop files displayed by the "Browse" button or from Windows Explorer.



Characters

Select the characters you will need on the CD. When CD Prep scans your presentation file(s), it will automatically select all the characters it finds in your scripts. You may select additional characters as desired. Character files will be copied onto the user's hard drive in the Windows Temp folder in order to speed up access during playback.



Languages

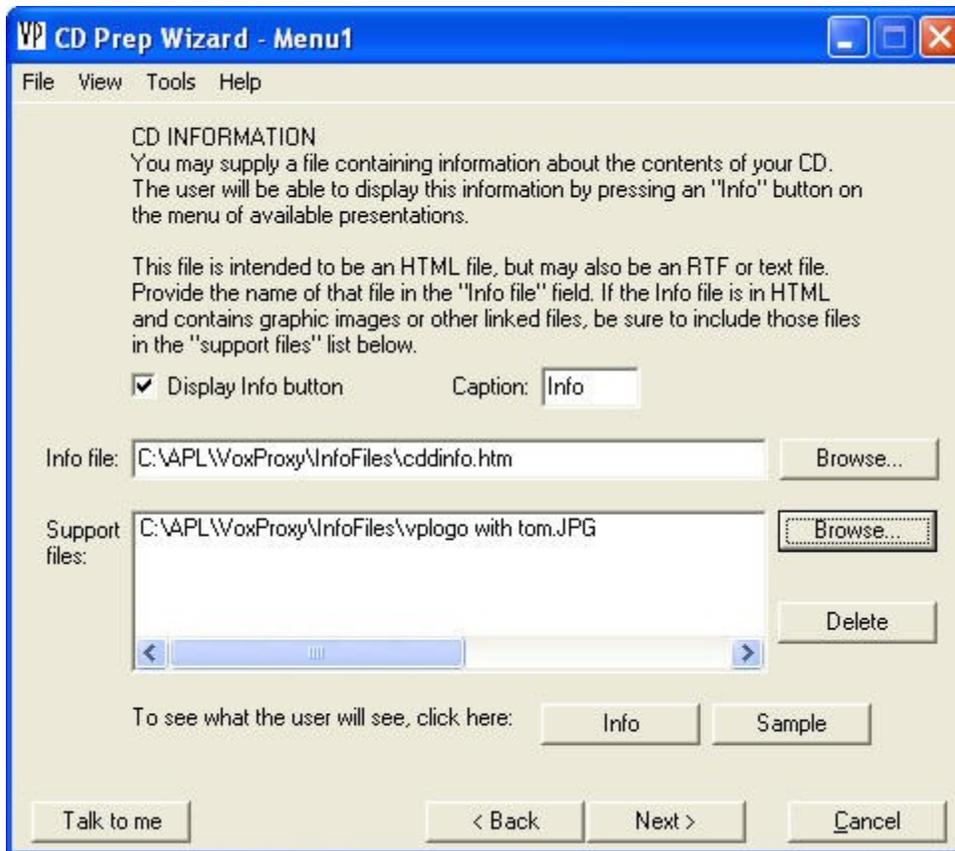
The language page appears similar to the characters page shown above, but lists all of the languages included with Vox Proxy. Select the languages which are used in your scripts. Languages determines the Text-To-Speech engines which must be installed on the user's computer. Most characters require the U.S. English language by default. Do not select any languages which you do not actually need, because their installation slows down the CD startup.

Info File

The "Info file" is an optional HTML, RTF, or text file which will be displayed when the user presses the "CD Info" button on the CD menu. You must create this file in a text or html editor. HTML or RTF offers nicer display formatting than plain text files. For RTF or text, you can create the file using Wordpad. For HTML, you may use any html editor. If you use an HTML file, you can include picture files or other hyperlinked HTML files. Any such "support files" must either be in the same directory as the html file or in a sub-directory.

You may un-check the "Display Info button" field if you do not want to include an info button and you may change the caption that will appear on the button.

After specifying the files, you can see exactly what the user will see by clicking on the "Info" button.

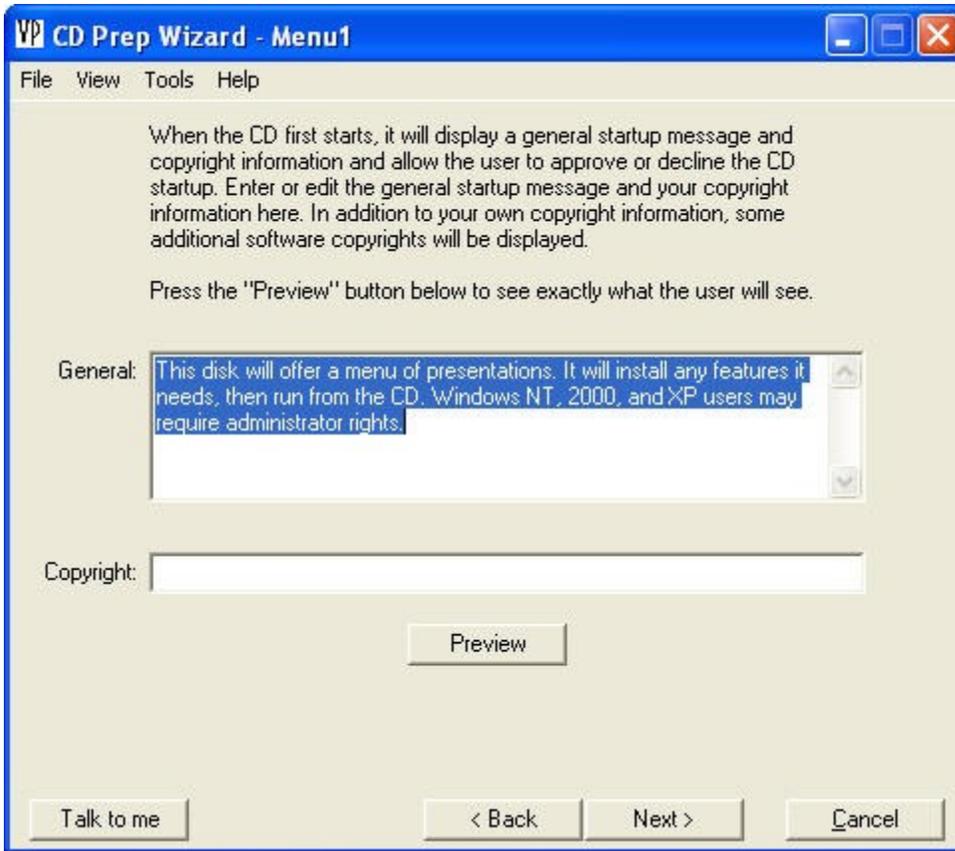


Author File

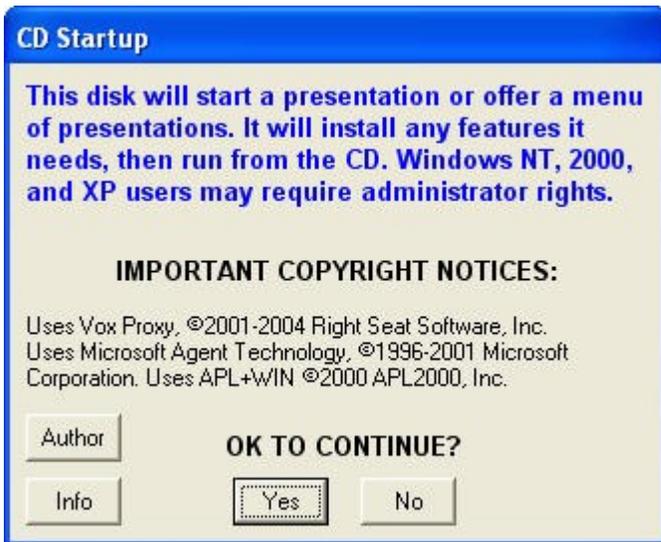
You may designate a file similar to the "Info" file described above for the purpose of identifying the author(s) of your presentations. Like the Info file, this can be an HTML, RTF, or text file.

CD Startup Message and Copyright

When the CD auto-starts, it displays a dialog box to the user allowing him/her to continue with the autostart or not. Copyright messages are displayed on this screen, which appears similar to the following:



The Preview button will allow you to see exactly what the user will see when the CD starts:



Finish

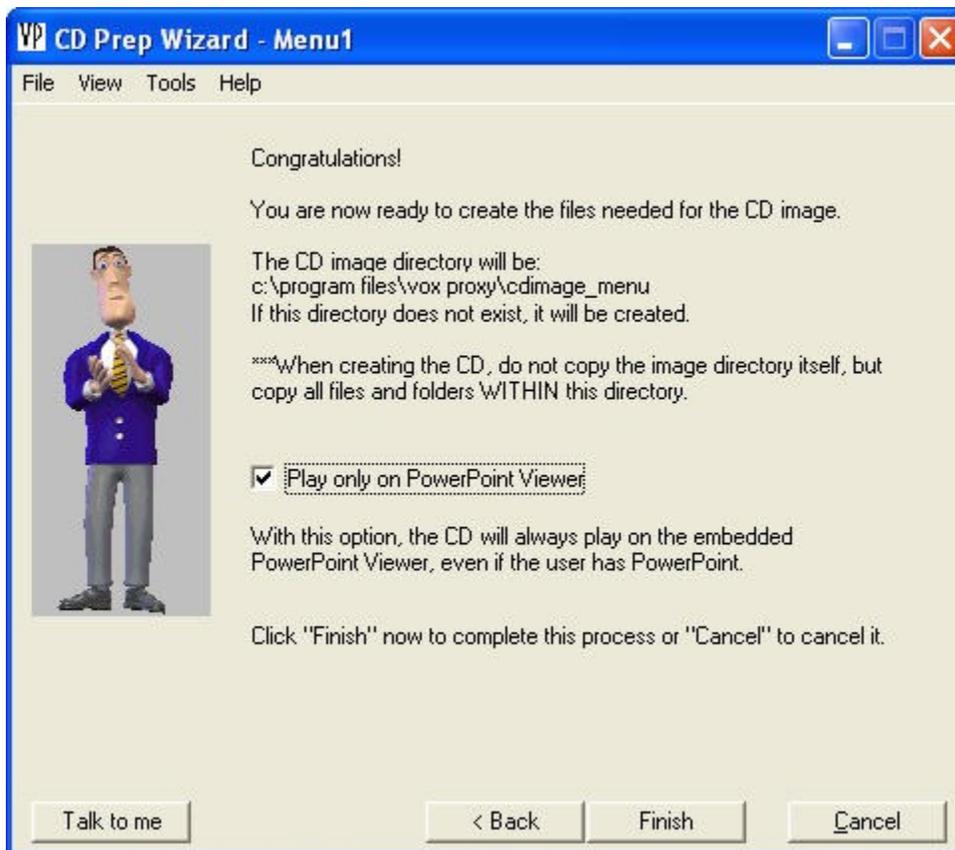
Before clicking Finish, you must decide whether to require that the CD always play on the PowerPoint 2003 Viewer. The viewer files will automatically be included on the CD. The advantage to always playing on the viewer is that the playback will be consistent. If you allow

the presentation to play on whatever version of PowerPoint that the user may have, some of your PowerPoint features may not work.

Why would you NOT require that it always play on the viewer? Only in rare instances. The viewer does not support running VB macros, programs, or opening linked or embedded objects. Also, a few Vox Proxy features are not supported under the viewer: LastSlideViewed; WriteShapeText.

When you click "Finish" on the final page of the CD Prep Wizard, you will be ready to create the actual "CD Image" files. First, if you have not yet named the CD layout, you will be asked to provide a name. The program will then proceed to create the CD Image and copy all of your relevant files to the appropriate folders.

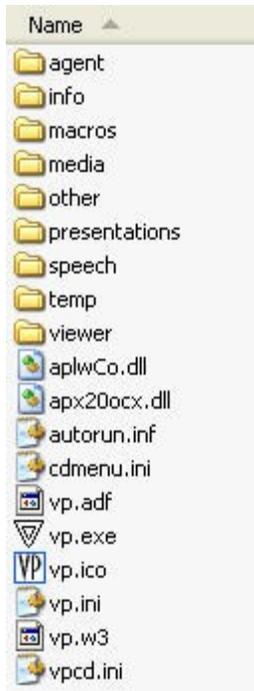
Remember where your CD Image folder is! It is shown on the finish page to remind you. When you burn the CD, remember to burn the **contents** of the CD Image folder, not the folder itself.



Burning the CD

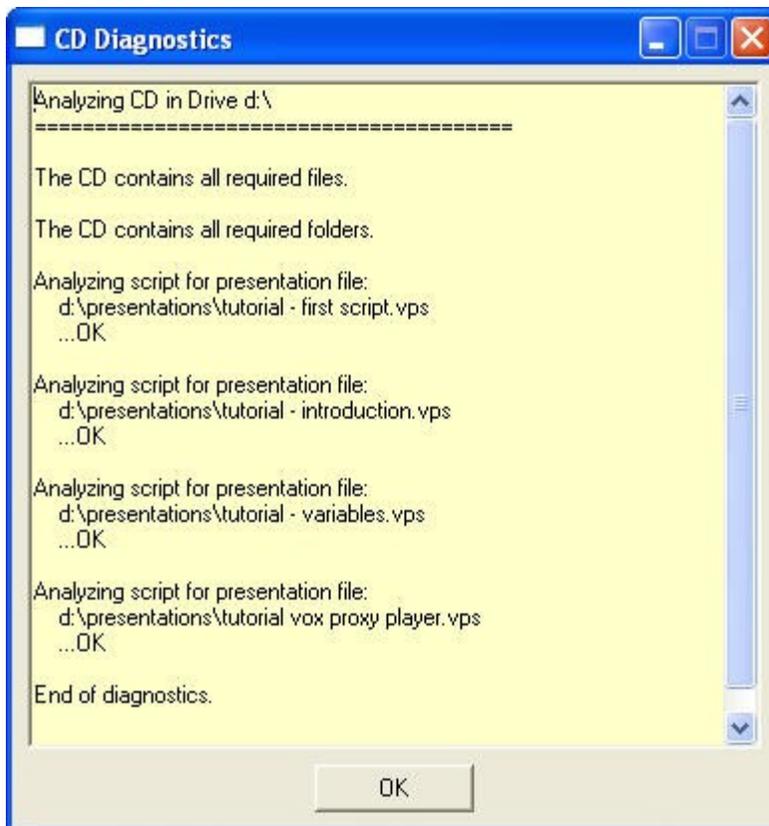
After finishing the CD Prep process, you will have a directory, called the "CD Image Folder" containing all of the files and folders needed to create the auto-starting CD.

Use your CD-writer software to copy the entire contents of this folder, including its sub-folders, to the CD-R. When done, check the contents of the CD using Windows Explorer. In the root directory of the CD, you should see at least the following folders and files:



[CD Diagnostics](#)

After creating the CD, you can run a diagnostics check on it from the tools menu of CD Prep. The results of the diagnostic check will look something like the following:



Copyrights and trademarks

©2001-2004 Right Seat Software, Inc.

Vox Proxy® is a registered trademark of Right Seat Software, Inc.

PowerPoint is a registered trademark of Microsoft Corporation. Microsoft, Microsoft Agent, and the Microsoft Agent logo are registered trademarks of Microsoft Corporation in the United States and/or other countries. The characters Genie, Merlin, Peedy and Robby copyright 1996-1998 Microsoft Corporation, all rights reserved.

Natural Voices is a trademark of AT&T Corporation.

Vox Proxy uses APL+Win, Copyright 1995-2003 APL2000, Inc.

Glossary

A

Address context: A set of TTS Engine pronunciation rules defined in "Speech Normalization". These rules include how to pronounce abbreviations in addresses, phone numbers, web addresses, etc.

Animation: Either a character animation or a PowerPoint slide animation, also called a custom animation.

B

Balloon: A cartoon-like balloon or bubble that appears over a character's head and displays the text being spoken by that character.

C

Character animation: One of a list of discreet animations that a character can perform. Each character has its own unique list of animations.

Click: A Vox Proxy script command that sends a "next build" command to the active PowerPoint slide show. The result is essentially the same as the user pressing the left mouse button. The Click command is used to invoke the next "custom animation" on the slide.

Context: A set of TTS Engine pronunciation rules defined in "Speech Normalization". These rules include how to pronounce abbreviations in addresses, phone numbers, web addresses, etc. See E-mail Context and Address Context.

Custom animation: Animation rules for a PowerPoint slide object. Custom animations can be invoked from a script using the Click command.

E

E-mail Context: A set of TTS Engine pronunciation rules defined in "Speech Normalization". These rules include how to pronounce abbreviations in addresses, phone numbers, web addresses, etc.

I

Idle animation: One of a special set of character animations beginning with the word "idle". These animations are played randomly after several seconds of inaction by the character.

L

List box: A dialog box containing a list of items from which the end user may make a selection. The dialog box is displayed during a slide show and the selection result can be acted upon in the script. See the "ListBox" command.

Looping animation: A character animation that loops until it is explicitly stopped. Animation names usually end with "ing". Special care must be taken using looping animations.

M

Macro: A Vox Proxy script that is saved in a file rather than a PowerPoint slide. Macro files can be executed directly from Windows by opening the macro file. They can be embedded into a PowerPoint presentation file or another macro file.

Message box: A dialog box that can be displayed to the user during a slide show. The message box displays a text message and can have several optional response buttons. The user's response can be captured and acted upon in the script.

Microsoft Agent: Microsoft Agent is a character technology used by Vox Proxy. It allows characters to appear on top of any application, perform pre-selected animations, and to speak using either a TTS engine or pre-recorded sound files.

S

Shape: A PowerPoint slide object. Shapes with custom animations can be animated from a Vox Proxy script using the Click command and can be written to using the WriteShapeText command.

Speech normalization: A set of pronunciation rules built into a Text-To-Speech Engine. These rules include how to interpret and pronounce abbreviations, addresses, various kinds of numbers, and so on.

Speech tag: Special character strings inserted into the body of speech text used to modify the pronunciation, pitch, tempo, or interpretation rules (called "Text Normalization") for the speech.

T

Timer: A timer allows you to place a hold in your script waiting for a specified elapsed time. Timers can be based on the slide show, slide, media, or custom timers defined in your script. See the StartTimer and Waitfor commands.

TTS Engine: A Text-To-Speech application that converts written text into speech played through the computer's audio device. TTS engines are available in multiple languages and from multiple sources. Vox Proxy comes with the TruVoice TTS and can be ordered with the AT&T Natural Voices.

V

Variable: A variable is a special word beginning with an ampersand that contains changeable text or numbers. When used in a script, the text or data is substituted for the variable before it is played.

Index

&		
&Chardir	85	
&DiskDrive	85	
&Macrodir.....	85	
&Mediadir.....	85	
&Presdir	85	
&Response	57	
&Tempdir	85	
&Windir.....	85	
A		
Active slide	95, 96, 107	
Address	107	
Address Context	107	
Administrative Rights	127	
Animation		
Character		
Idle	10	
Looping	10	
Miscellaneous 10, 29, 30, 40, 41, 77, 81, 104, 105, 116, 119		
Random	10, 30	
PowerPoint.....	77, 81, 121	
AT&T Natural Voices	9, 37	
AT&T Natural Voices Text-To-Speech	105	
Author file	125, 128, 133	
B		
Background	48	
Balloons .. 9, 29, 35, 36, 37, 39, 41, 46, 91		
Bitmap	48, 49	
Bookmark.....	82	
C		
Caption	56	
CD		
Burning	135	
Files 124, 129, 130, 132, 133, 135		
Image folder 123, 129, 134, 135		
Layout	128, 134	
Menu	125, 129	
self-starting	123	
Variables	124	
Viewer	123, 124, 125, 134	
CD Prep9, 85, 105, 123, 124, 125, 127, 128, 129, 130, 133, 134, 135, 136, 139		
Cell.....	60, 66, 70, 75	
Center	31	
Character		
Animation	10	
Behavior	119	
Commands	91	
Gallery	19, 20, 21, 23, 107	
Sources	20	
Click	77, 81	
Clock	52	
Close		
Agent Characters	23, 107	
Bitmap.....	49	
Excel	50	
Flash	50	
Media.....	51	
Word	51	
CloseBitmap	49	
Clsid	42	
Color	48	
Column headings	70	
Command bars	70, 72	
Context.....	46	
Controls.....	62	
Ctx.....	46	
Current Slide	110	
Custom Animation	121	
D		
Data file.....	66, 69, 72, 75, 76	
Delay.....	51, 73	
Diagnose	105	
Disappear	10	
Display	38, 54	
Document	72	
Dual monitor	54	
E		
E-mail Context.....	9, 46, 107	
Embedded macro	25	
Emphasize.....	9, 46, 107	
Excel	50, 60, 66, 69, 70, 75	
F		
Fade	59	
FAQs	17	
File path.....	37	
Files	44, 66, 76	
Find	102	
Flash	61, 74	
Font	29	
FormattingToolBar.....	70	
Formula bar.....	70	
G		
Gesture	1, 30	
GestureAt	91	
Grid lines	70	
H		
Headings.....	70	
Hide	10, 29, 31, 91	
HScrollBar.....	70	
I		
Idle animation	10, 39, 69, 91	
Info file	125, 128, 132	
Ini file.....	66, 76	
Input	56	
Insert	104	
K		
Key	66, 76	
L		
Language	32, 39, 42, 46, 91, 107	

List	57	Reset	46, 107
ListBox	57	Run	67, 68
Listen	33, 34, 91	Run mode	1, 7
ListenFor	91	S	
location	30, 35, 39, 70, 72	SAPI-4	9, 127
Loop	10, 61, 62	Say	9, 41, 46, 91, 105
Looping animations	10	SayWay	91, 105
M		Scale	38, 61
Macros	8, 25, 27, 48, 55, 58, 60, 65, 68, 95, 100, 107, 130	ScaleCharacters	91
Mail	67	Screen Resolution	38, 54
Media		Script	
Audio	51, 59, 62, 64	Basics	8
Bitmaps	48, 49	Editor	95, 96, 97
Flash on CDs	124, 130	General Rules for	45
Timer	74	Guide to first Samples	12
Video	51, 62	Tips and Techniques	113, 114, 116
Volume	59	Wizard	91, 92, 93, 94
Menu	58	Writer	95, 96, 97
Menu bar	70	Scroll bar	70
Message box	59	Section	66, 76
Microsoft Word	72	Self-starting CD	105
Monitor	54	Set	46, 69, 73
Monotone	9, 46	SetPrefs	40, 91
Move	35	Shape	81, 105
MoveTo	35, 91	Show	10, 39, 40, 91
N		ShowExcel	70
NextAnimation	77	ShowWord	72
NoBalloons	91	Size	40, 91
Noloop	10	Slide	
Normal	9, 46	Diagnose	110
Normalization	9, 46, 107	Interaction	77, 81
Nowait	10, 116	Navigation	78, 79, 80
P		O&A	121
Package for CD	123, 129	Script	95, 96
Paths	85	Show	80, 121
Pau	46	Tab	96
Pit	46	Test	110
Point	1, 30	Sound effects	10, 39, 40, 69, 91
PointAt	91	Speak	9, 41, 46, 91
PowerPoint		Speech	
Animation	77, 81	Balloons	29, 35, 39
Behavior	113, 121	Commands for	36, 37, 41
Command	77, 78, 79, 80, 81	Engine	9, 42
Navigation	78, 79, 80, 114	Normalization	9, 46, 107
Slide	77, 78, 79, 80, 96	Prerecorded	9, 37
Preferences	40, 43, 69	Pronunciation	99
Program	67	Tags	36, 46, 104, 107
Program Updates	17	TTS	9, 42
Programming commands	82	StartTimer	73
Prompt	56	Stop Playing	19
Pronunciation	9, 41, 46	Style	30, 34, 36
Pronunciation Dictionary	41	Support	17
R		T	
Random	10, 65	TAG	46
Read only	70, 72	Test	9, 104, 110
Remove Agent Characters button	107	Test speech	9, 107
Replace	102	Text	59
		Text-To-Speech	125

- Think29, 41, 91
[Tile](#)107, 110
[Timeout](#)69
Timer..... 51, 52, 70, 72, 73, 74
[Tips](#) 113, 114, 116
[Title](#).....59
Tool bar 70, 107, 110
Top.....62
[Troubleshooting](#) 110
TruVoice TTS 9
TTS 9, 42, 46
TTEngine 9, 42, 91
U
[Using Multiple Characters](#) 116
V
Variables
 About.....85
 Creating 69, 85, 100
 In Excel 66, 75
 In input 56, 57
 In paths85
 In speech 36, 85
 Pre-defined 85, 100, 104
 User-defined 69, 85, 100
 With Files 66, 76
[Viewer](#) 123, 124, 125, 134
Voice
 [Input](#) 33, 34
 Output9, 36, 37, 39, 41, 42, 46
 Prerecorded 37
Vol 46
Volume 46, 59, 62
[Vox Proxy Player](#) 123
W
Wait
 Characters..... 43, 91
 Click81
 Commands 43
 Media..... 74
 Timer..... 51, 52, 74
[WaitCommands](#) 91
[WaitFor](#) 72, 73, 74, 91
WaitforClick..... 77, 81, 105
WaitforNextAnimation 81
Wav Files9, 37, 64
Whisper 9, 46
Wildcard 58
[Wizard](#)..... 107
[Wizard toolbar](#) 107
[Word](#) 72
[WordTime](#)..... 72
Workbook..... 66, 70, 75
Worksheet..... 66, 70, 75
Write 75, 76