Autorun Maestro

Program Version: 1.0

Available Topics

Files

Instructions

Commands

Information

More Help?

Try our Online FAQ and trouble-shooter! Just CLICK HERE!
(requires an internet connection)

Package Contents

When you downloaded this package it should have contained the files listed below. If it did not contain these files, or you suspect the package may have become corrupted, please check our web site for the full software package.

THE FINAL PACKAGE CONTENTS HAVE NOT YET BEEN FINALISED

Maestro.exe The Main Autorun program. This program displays your menus for use.
MaestrEd.exe the Menu Editor. With this program you can create and edit your menus.
Maestro.hlp This Help file.
Maestro.mnu The default menu definition file, use MaestrEd.exe to create your own version.
Gen_CD.ico This is an Icon that will represent your CD on user’s machines, replace if you wish with one of your liking
Back.bmp Background logo image, create your own and replace this file with it.
Btn*.bmp Images for Buttons, Can be any size, create your own.
readme.txt Basic instructions, including information on the location of this file.
autorun.inf Autorun setup file, copy this to the ROOT directory of your CD.
change.wav This sound will be played when the program changes INIs (goes to a sub-menu).
click.wav A sound to play when a button is clicked.
exit.wav A sound to play when a program exits.
select.wav A sound to play when a new button is selected as the active button.
start.wav This sound will play when the program starts, optionally replace.

Optional Files

You can create and add any or all the following files as your needs dictate. If the files are not present then the features they represent will simply not be there (i.e. you are not required to have the files there, and no error will occur if they are missing). Therefore to take advantage of these features just place the files in the same directory as Autorun Maestro Graphic. Most of these files are not included with the downloaded package and are up to you to create as you will.

[button]d.bmp	Images for buttons in their Down Position.
[button]a.bmp	Images for buttons when they are the active button.
start.wav	A sound to play when the program starts.
exit.wav	A sound to play when a program exits.
click.wav	A sound to play when a button is clicked.
select.wav	A sound to play when a new button is selected as the active button.
change.wav	A sound to play when the menu is changed (with a new INI).

System Files

You may also wish to download Sysfiles.zip, this file contains certain files which can help ensure your CD will autorun correctly on most all systems. You can find a link to these files, which are shared by all of our applications, at the end of each program's homepage. Unzip the files contained in the ZIP file to the same directory as the autorun program.

We strongly recommend including these system files if you plan to distribute your CD. While the software runs without these files on many systems, on some computers the software will not function unless these are present.

Introduction

The value of your CD, and the talent and capabilities of you or your company rely heavily on just what the user sees when he is invited into your product. Usually people have little choice about how their CD is presented. Maestro gives you the power to present your CD in any way you choose. That’s because Maestro is flexible and powerful software, which essentially allows you to create your own programs. These programs take the form of menus, from which you can open document, start programs, go to Internet sites, open directories to browse, and even functions like minimizing, exiting and going to another menu (a sub-menu). Maestro offers you the power and flexibility to completely create the look and behavior of your menu. How your interface looks is entirely up to you, it can be any shape, and size, with up to 99 buttons on any one menu. These buttons can be in any location, and can also function as exit and minimise buttons!

Autorun Maestro is easy to set up and comes with a dedicated Menu Editor which allows you to quickly and easily create or edit menus. You can use a variety of software to create your images, from Adobe Photoshop to Windows Paint, even programs like PowerPoint can be used to make impressive graphics. For our sample we used StyleSkin, dedicated skinning software for application just such as this. StyleSkin allows you complete flexibility over your skin design, while making it very easy to create impressive graphics with an impact only professional skinning graphics can offer. You can read more about StyleSkin at the StyleSkin website, (http://www.pollensoftware.com/styleskin/)

Menu Editor Overview

The Maestro Menu Editor allows you to not only edit your menus, but to easily create new menus. You can then link these menus together by simply entering the menu name as the command for a button, then when the user clicks the button they will be taken to the menu specified. In this way you can have as many sub-menus as you wish, all linked together in any way you see fit.

The Maestro Menu Editor has powerful features built in which make it easy to create or edit your menus quickly and with little fuss. Basic features of the Menu Editor are detailed below.

Menu Properties

To change the properties of the menu you can right-click on the background image (outside any buttons or text). This will bring up the Menu Properties dialog. You can also click on the Settings menu and click “Menu Properties…” to bring this dialog up. You can read about the settings in this dialog in Menu Properties.

Button Properties

To change the properties of a button you can simply right-click on the button to bring up the Button Properties dialog. You can also get to this dialog by clicking (left-click) on the button and either clicking the Settings menu and selecting “Button Properties…” or clicking on the Button Properties button in the upper toolbar. You can read about the properties available in this dialog in Button Properties.

Accessory Text

The Accessory Text is an area on your menu where you can display textual information about the menu and each button. This area can be resized by clicking on the text (left-click) and dragging the small handles. You can edit the default text directly by simply right-clicking on the text. The text associated with each button can be edited by right-clicking on each button and going to the Button Properties dialog. The Accessory Text can be formatted by clicking on the formatting buttons on the lower toolbar. For information on what each tool does, simply move you mouse over it and look at the status bar of the program (at the bottom).

Autoseek Button Placement

This feature is a powerful function, which automatically aligns the button according to the background image. You can use this feature by simply selecting a button (left click) and clicking the Autoseek Button Placement button in the upper toolbar. Most often when you create your skin you will create the button areas right on the background image. This features looks for that image on the background, and if it finds it moves the button to that location. You can try this simply by moving one of the buttons in the sample menu a small distance from where it should be and clicking the button. The Menu Editor will then look for the correct location, find it, and move the button to that location. The function only looks for the start of the button image 50 pixels around its current location, so you may need to place the button close to where it should go.

Flash Button

This toggles the current button indication between flashing a reversing field around the image (a selection box) and flashing the button image itself. When on the button flashes back and forth between the chosen button image and the image of the background area where it currently is. This can be useful if you are trying to align the button to a feature or point on the background image.

Alignment Guides

This toggles the presence of alignment guides. If these guides are present, when you drag elements on the menu close to or over the guides they will snap to the them. You can align elements by snapping the left, right or middle of the object to the guide.

Menu Properties Dialog

In the Menu Properties dialog you can set the following:

Menu Caption

The Menu Caption will identify your menu to the user through the taskbar. The text you enter in this field will be passed to MS Windows as the description of the program. In this way if the user wants to return to your menu after doing work in another program, they can click on the taskbar (or press ALT+TAB) and select your menu. It is best to keep this short but descriptive, something like “Pete’s Parts – CD Catalog 1”.

Bitmap to Use as the Background

This defines the image that will be used as the background for your menu. The size of the menu is also dependent on the bitmap. Just select the image you wish to use from the drop-down list, or if you know the name of the image you can type the name of the image in directly. All images used in the menu must be located in the same directory as the program, so do not enter any path information into the field. If you add files to the directory after this dialog is loaded, just click the “Refresh” button to update the drop-down list.

Transparent Color

Any areas of your menu that are the same as the color shown in the box will be removed from your menu when it is displayed to the user. This can allow you to (as we have done on our sample menu) make you program an odd shape, like rounding its corners. You could even, if you wanted a unique effect, create a menu with a hole in the middle, of very oddly shaped edges. Just color the parts of the image you do not want to show one of the listed colors and then set that color as the transparent color, its that easy.

Default Text

The default text is the text shown to the user in the Accessory Text area when the menu loads, or the user moves their mouse off a button. This text would usually be a welcoming message, or perhaps basic instructions or even copyright information.

Menu Location

This sets where you menu will appear on screen when it loads up. You may want to create a long thing menu and line it to the top or side like a toolbar, or you may simply want your menu centered on the screen. Just click the location you want your menu to appear in.

Button Properties

The Button Properties dialog allows you to set the many properties of each button. These properties are basic information on the behavior, appearance and description of the button. A description of each property can be found below.

Button Images – Up Image

This is the image used to represent the button on the menu. This image is shown to the user whenever the button is neither selected nor clicked down. It is also shown to the user during these other events if images are not specified for these events.

Button Images – Active Image

The appearance of the button will change to this image when the user activates the button as the current button. The button can becomes the active button by the user either moving their move over the image, or by pressing Tab through the buttons. This image is usually the same image as the Up image, just highlighted so the user knows which button is active. If the image is not provided, then the Up image will continue to be displayed.

Button Images – Down Image

The appearance of the button will change to this image when the user clicks down on the button, or briefly when the user presses Enter (when the button is the active button). It is usual to provide an image that appears depressed, or one that appears with strong highlight to indicate to the user that they have clicked down on the button. If this image is not provided then the Up image is shown instead.

The Autofill Button

The Autofill button looks for matching Active and Down images based on the name of the Up image. This allows you to use naming conventions when creating your graphics. Just name the Up image ending with a “U”, such as Btn1u.bmp. Then name your active image ending with an “A” (e.g. Btn1a.bmp), and your down image with a “D” (e.g. Btn1d.bmp). Simply specify your Up image in the Button Properties dialog, and then press the Autofill button, the program will then look for the matching Active and Down filenames following the convention above. If it finds them it will enter them in the spaces.

Button Images – Accessory Image

This image will appear in the location of the Accessory Image whenever the user activates the button by moving their mouse over it, or by tabbing to the button. The location of this image is the same for all buttons, and can be set by clicking OK on the Button Properties dialog after setting the Accessory Image, and then moving your mouse over the button so that the Accessory Image appears. You can then click and drag the image to any location you wish.

Execution – Button Command

This is the command the button will take when the user clicks it. This can include a number of different actions, such as opening a document or starting a program. These are defined in the Button Command Summary.

Execution – Commandline Argument

This feature is best left to advanced users, and should normally be left blank. Commandline arguments allow you to pass a string of information to a starting program to instruct the program. Most usually this is used to either instruct a program to immediately start a particular action, or to immediately open a document. You DO NOT need to use this feature to open a document in the user’s default viewer, simply enter the path to the document as the command and leave the commandline field blank.

Execution – Minimise Setting

By clicking the checkbox labeled “When clicked the menu should minimise” the menu will minimise whenever the user clicks it. This could be used to simply implement a minimise feature (as we have done on our sample menu), or it could be used in conjunction with a command. The user can simply click on the menu in the taskbar to restore it to its original size and location.

Execution – Exit Setting

Checking this box causes the menu to exit whenever the user clicks the button. This could be used to create an “Exit Menu” functionality, or could be used in conjunction with a command. You may need to exit the menu when executing some commands, for instance if the button is linked to a setup program that requires the computer to be rebooted before continuing. You should at least have one button which functions solely as an exit button (i.e. no command entered). This way the user can exit the menu if the do not wish to implement any of the options at that time.

Execution – Wizard >>

The Button Wizard will take you through the simple process of creating your button command. It will prompt you for any necessary information (depending on what you want the button to do) and will automatically fill out the button command for you, including minimise and exit settings. This wizard provides an easy way to take advantage of the wide range of button commands and features that Maestro has to offer. Even for advanced users, it also makes sure you spell paths correctly, and avoids the need to remember and type out lengthy paths.

Default Button Setting

This number simply indicates the number of the button in the Tab order. When the user presses the Tab button, the focus will go from button to button in order of the Tab order. You cannot change the Tab order through this dialog, the number is for information only. You can however change the Tab order by clicking on the Tab Order button on the lower toolbar.

Default Button Setting

Clicking this button will cause the button to be the active button when the menu loads.

Menu Position – Left Edge

This field specifies the location of the button. It is given as the number of pixels the left edge of the image is in from the left edge of the menu. You can adjust this value by changing the number. This can be useful for alignment purposes.

Menu Position – Left Edge

This field specifies the location of the button. It is given as the number of pixels the top edge of the image is in from the top edge of the menu. You can adjust this value by changing the number. This can be useful for alignment purposes.

Accessory Text

The text entered into field will be shown in the Accessory Text area whenever the user moves the mouse over the button, or selects the button by pressing Tab. Normally you would enter descriptive information on the action the button will take if the user clicks it. For example, you might have descriptive text which states “Will install the Adobe Acrobat document viewer, version 3.0” attached to a button which is set to launch the Adobe Acrobat Viewer setup.

Step-by-Step Instructions

Create a directory named “Maestro” and unzip all the files into it. The program is ready to run. The program does NOT require ANY installation*. To set up a menu for your CD just follow the following simple steps:

- Create an "Maestro" directory (see NOTE)
- Unzip the files into this directory
- Optionally unzip SysFiles.ZIP (downloadable separately from our site, click here to download) into the same directory. This makes sure your program should work on all systems. We strongly suggest that if you plan to distribute your CD that you include these files.
- Create your own bitmap for the background image (and copy it to the directory). The size of the program screen will be the size of this image. If you would like any area of the form transparent (like the corners) then color these Blue, Green, Cyan, Red, Magenta or Yellow. Just make sure the color is pure and that no other areas on your image are the color you select.
- Create your button images. Each button in your menu will need an image for the Up (default) position. Each button should also have an image for the Down (clicked) and active (selected) positions.
- Run the Menu Editor (MaestrEd.exe) to create your menu file/s. You must at least create one menu and name it “Maestro.mnu”, this menu will be the default menu (first loaded). You can then have this menu linked to as many submenu’s as you like.
- Replace the included wave sounds with your own wave sounds if you like. You may wish to include or remove sounds as well. You can find what sound events can be used in Optional Files
- Write your CD, making sure you moved the autorun.inf file to the root directory of the CD (the rest must be in a subdirectory).

NOTE that the supplied autorun.inf assumes you copied the program into a directory named Maestro, and that you will use the “Gen_CD.ico” icon provided in the package to represent your CD. Open the autorun.inf file to change these details. THE FORMAT OF THE INF FILE MUST NOT BE CHANGED. So if you do choose to change the file, follow the original format exactly.

NOTE the software MUST be in a subdirectory, it has not been programmed to work from the root directory.

* If you have problems running Maestro on your system, please download SysFiles.ZIP (downloadable separately from our site, click here to download) and unzip the file into the same directory as Autorun Maestro.

Design Tips

Maestro gives you complete control over the interface. This can be great for those who have experience in design, and a flair for presentation, but it can be murder to the end user if the person who designed the interface did so poorly. Below you will find our general tips on using Autorun Maestro, creating interfaces and authoring CDs.

Get the Graphics Right

There are many ways to create the graphics for your interface. If you are an advanced user you will have already developed your own preferences and experience creating such graphics. However, if you are new to computer graphics you may not realize that you probably have all the software you need to create the interface you are after. There are many sources on most computers for great interfaces. Programs like MS PowerPoint can create great interface graphics, or you might have a digital camera that you can use to create your base graphics. Assuming you have at least a product like a modern word processor, and an image you like you can easily create an interface by simply inserting the image onto the document, placing your text where you want on the image, and taking a screenshot. You can then paste the screenshot in MS Paint so that you can cut it up and save it. Just change the color of your text for Active and Down images.

Alternatively you can purchase a program like (StyleSkin). Pollen Software's StyleSkin allows you to create an infinite range of stunning professional interface skins with great ease. It can eve create a skin out of a photo!

Keep the Interface Clean

Many users when they get a new computer use every font they have on a single document. It looks awful. The same goes for your interface, so keep the interface clean.

Keep the interface simple. If the interface is overly complex, with complex and contrasting graphics, then people can find it uninviting to look at. They may also have difficulties identifying the relevant parts of the interface.

Do not have too many buttons on a single menu (most would argue more than 5 on the root menu, or 8 on a sub-menu is too many). If you wish to offer many options to the user, try grouping them in logical sub-menus.

Keep you buttons recognizable. Do not have buttons of widely varying shape and sizes on a single menu. Make the button areas stand out from the background enough so that they are easily distinguishable.

Try to align and group your buttons with each other. People can easily miss a button that is on its lonesome far way from the other buttons.

Use descriptive buttons and Accessory Text. People are terrified of pressing buttons when they do not adequately know what the button will do. So give them clear, concise, representative text on the button, and concise descriptive detail in the Accessory Text.

Allow the user to exit. The user may decide they do not wish to do anything at this point in time, so give them a button to simply exit with (no command, exit setting checked). If you are using submenus, remember to allow the user to navigate back to the top (simply type the name of the menu as the command), as well as to exit. This is only an issue if you have set your submenus to not load in a separate window.

Add a "Browse CD" Button

Many users love to just have a look through the file available on the CD. So consider adding a "Browse CD" button. This is easily achieved by simply entering a singe backslash as the Command (\). Then the user clicks on the backslash then Explorer will be opened in the default "Browse" style of large tiled icons. You may wish to check the exit setting, so that the menu exits when the user clicks the button.

Creating Multi-level Menus

With Autorun Maestro you can create a multi-level menu, allowing you to group buttons on separate screens. You can have Maestro load an individual image for each group of buttons, have different button graphics & associated commands, have different numbers of buttons etc.

To build multi-level menus you simply build separate menus and for the button command just refer to the menu (.MNU) file you wish to load (do not enter any path information, just the filename). If you want your current menu to close when the user goes to the submenu, just check the checkbox which says “When clicked this menu should close (exit)”. If you wish the menu to open in a separate menu, leaving your current menu displaying, leave this checkbox unchecked.

You must place all the menu files in the Maestro directory.

You can link the menus in any way you like, including linking back to the default Autorun Maestro menu. You would usually do this if you wish to provide a “<< Back” button of some sort.

If your sub-menus are similar then it is often helpful to open an existing menu, make changes to it, and then save it under a new filename.

HANDY HINT

If you wish to start Autorun Maestro with a particular sub-menu loaded (say through another program), you can pass the submenu as a commandline argument to Autorun Maestro. Autorun Maestro will then start and display that particular submenu.

You can also open a sub-menu by dragging its INI file into the Autorun Maestro program file (Maestro.exe). This can be handy when you are editing, if you want to look at a sub-menu you have not linked to yet, or do not wish to navigate through the menus to look at the particular sub-menu.

Commandline Arguments

First, please note that commandline arguments are not normally required, and only more advanced users should attempt to implement them. Normally the commandline argument field should be left blank, and the Command field would contain the link to the document you wish to open.

Commandline arguments are strings, which you pass to programs to instruct them in their operation. In its most basic form, a program will often open a file that is passed as a commandline argument. An Example of this might be the following string passed to MS Windows NotePad:

```
C:\Windows\Readme.txt
```

which is simply instructing NotePad to open the Windows Readme file. Commandline arguments are often referred to as "slash codes", as the string of text passed to the program often starts with a slash. An example would be passing “/C” to a screensaver requests the screensaver to show its configuration screen. One real use in CD Authoring is when you wish to include a viewer for documents you are distributing on the CD. In this way you can open a document on the CD in a program on the CD.

You can now pass command line switches ("slash codes") to programs with Autorun Maestro *Graphic*. To do this just add your commandline argument in the configurations box titled "Any Commandline Arguments (programs only)”

In many instances you will want to refer to the CD-Drive when passing Commandline arguments to your program. Some programs (very few) will allow a relative path as a commandline argument, but most require you put the full path to the file you wish to open. If an absolute path is given, problems will occur when the user places the CD in a drive with a different Drive Letter (Your CD Drive might be E: drive, but theirs may be D:, or F: or...). To overcome this problem we have included a variable (%CD%) which replaces the drive letter in your commandline argument. Then our software replaces this variable with the drive letter the CD is currently being run from. In this way your Commandline argument, and your CD, will work in any drive, regardless of the CD drive letter.

Therefore the above string, as a Commandline argument to pass to another program, would be:

```
%CD%:\Windows\Readme.txt
```

Please note the retention of the colon [:] in the above statement. PLEASE NOTE that this is NOT the format to simply open a document or start a program, IT IS FOR COMMANDLINE ARGUMENTS ONLY. If you want to simply open a document or start a program please see Button Command Summary

Creating Viewer Installation Menus

In Autorun Maestro you can now create menus that will automatically show if the user does not have a viewer for the type of document you have tried to open. This viewer installation menu can then offer installable viewers, information and options to the user. An example of how this works is given below:

- The user clicks on your button, which you have set to open a document on your CD (say a HTML page: index.html)

- However the user does not have a viewer installed, in this case s/he does not have a Web Browser. Autorun Maestro then looks for the viewer installation menu, which is based on the document extension of the file you tried to open, in this case it will look for HTML.MNU. Having found the HTML.mnu you created (where you would add buttons with links to the browser packages available on the CD), the software loads up the viewer installation menu. The user can then click on the "Install Internet Explorer" or "Install Netscape Navigator" (or whatever) button so that they can view your document.

- If the program does not find the viewer installation menu, then the user will simply receive the message that they do not have a viewer capable of viewing the document.

This allows you to easily set up viewers for any of the documents on your CD, and the user will automatically be prompted to install a viewer only if they do not have a viewer installed. You can do this for as many document types as you like. You may wish to have INI files for such documents as Adobe Acrobat files (PDF.mnu), ZIP Files (ZIP.mnu), you name it. You can also link back to the default menu by including a button on your viewer installation menu that points to the other menu (i.e. "maestro.mnu") or provide an exit button, in case the user does not wish to install software at this time. You may wish to also create a background and/ or default text that reflects the reason the user has been taken to this screen, "This CD requires that you have a web browser installed on your computer. Please install a Web Browser from the choices below to view this CD". All this provides you with a very powerful, yet simple method of offering viewers on your CD, for any software you choose.

Button Command Summary

Paths entered into Maestro should not include drive information. If no drive information is entered, Autorun Maestro will look for the file on the drive it is currently being run from. This means that it will look for the file on the C: drive when you are running it from there (during set up and testing), or from the CD once you have created it. Remember to enter the path as it **will be** on the CD. This means that if you are going to move the files out of a subdirectory when you create the CD, then you should not have that subdirectory in the path.

If the document will be on a specific drive, then you can designate the drive be entering the full path (e.g. "C:\Windows\readme.txt")

Please remember that all the following commands can be created automatically using the Wizard available in the Button Properties dialog in the Menu Editor. You can get to the Button Properties dialog at any time by right-clicking on a button.

The following actions can be performed by a button:

Open File:

"\DirectoryOnCD\MyFile.doc" (on the CD), or "C:\Windows\readme.txt" to open a file on a specific drive, like C: (not the CD).

Start Program:

"\DirectoryOnCD\MyProgram.exe" (on the CD), or "C:\Windows\Notepad.exe" to start a program on a specific drive, like C: (not the CD).

Open Directory:

"\DirectoryOnCD\DirectoryOnCD\" (on the CD), or "C:\Windows\", even "C:\Windows"

Go to Internet Site:

"http://www.pollensoftware.com/register.html". Remember to enter the full path (including page) for best results.

Exit Buttons:

Check the "When clicked the menu should close (exit)" checkbox in the button's properties. Leave the command blank unless you wish to execute a command before exiting.

Minimise Buttons:

Check the "When clicked the menu should minimise" checkbox in the button's properties. Leave the command blank unless you wish to execute a command before minimizing.

Sub-Menus:

The submenu (.MNU file) must be in the applications directory, DO NOT enter any path information in the command, e.g. "SubMenu1.mnu"

Message Box:

You can create a button that simply passes a message (with the information icon) to the user by entering your message, preceded by a dollar sign ($). Such that "$This feature not yet implemented" pops up a message to the user that the feature is not included on the CD when they click the button.

No Action:

You can create a button that does no action, except playing a different sound, by entering a single asterisk (*) as the command. The program will try to play the sound defined in “noaction.wav” (if present in the Maestro directory) when the button is clicked. This can be useful for features that you wish to “dim out” but still present to the User (e.g. for instances where the feature is not included on the CD).

Do not include the inverted commas (") when entering commands.

The absence of the drive (eg C:) means that the link is to the CD.

Slash directions above should be followed. These are Operating System "backslashes" not Internet style "forward slashes".

Trailing slashes on directory links do not matter.

The % symbol indicates that the file or program is in the same directory as the program. We have introduced this to improve the function of the program when downloaded. In this way the sample menu can be fully functional when downloaded. You should not use this variable when creating your commands.

The image displayed must be in Bitmap form, and placed in the same directory as Autorun Maestro.

Other Information

This program is shareware and may be distributed and freely used for testing purposes in its shareware state. For full information on what usage you are authorized for please see our site.

This program, makes no modifications to your system. It does not add files, it does not make registry entries. To completely remove the program, simply delete the directory which you unzipped it into.

PLEASE NOTE: the program is a 32-bit application, and as such will not run on anything previous to Win95.

This program was written by Sean C Flanagan, and is copyrighted 2000, by Pollen Software. If you wish a modified copy for commercial application, or have any suggestions for improvements to the program then contact me.

To register this program (for personal use or for distribution for commercial or non-commercial applications), please see our site:

You can find us on the Internet at:

Autorun Software Central (http://www.pollensoftware.com/)

Registration Details can be found on our Online Registration Page

Program Information and Updates can be found at the Homepage for Autorun Maestro *Graphic*

You can contact us at admin@pollensoftware.com

Our mailing Address is:
Pollen Software Pty Ltd
PO Box 314
Morningside, QLD 4170
AUSTRALIA