Ibm Essay, Research Paper

\* AppleScript Welcome to Macintosh OS Scripting This document is your introduction to the scripting environment on your Mac OS computer. Use it to begin working scripts and scriptable applications. You ll need to know basic Macintosh operations and vocabulary. What is scripting? When you use your Mac OS-based computer, you use the mouse and keyboard to open documents and application programs, move documents and file folders, enter information, print, and perform any other actions you want. Scripting gives you a whole new way to work: You can give the computer a list of things you want it to do a script and let the computer do everything on your list. You have a choice of scripting languages that you can use. The AppleScript system extension in your Extensions folder contains the AppleScript language, along with support for other languages that you can purchase or download from the Internet. These other languages include: UserTalk, part of UserLand Frontier ( http://www.scripting.com ) MacTcl/Tk (Tool Command Language/Toolkit) from Sun Microsystems ( http://sunscript.sun.com/ ) With the Mac OS, your scripts can do some things themselves (like manipulating numbers and text), but they rely on other applications to do the real work for them. You ll need scriptable applications in order to create really powerful scripts one of the most useful applications, the Mac OS Finder, is a scriptable application. You can find out about other scriptable applications at the AppleScript web site, at http://applescript.apple.com. About this guide This guide tells you how to use the Script Editor to record scripts and how to change a script to do something different. The Script Editor is an application program you use to open and run scripts, make new scripts by recording or writing them, and save scripts. You don t need to know how to write computer programs to do anything described in this manual. The Script Editor works with any scripting language that is built for the Open Scripting Architecture ( OSA ). Balloon Help You can use Balloon Help with the Script Editor. Choose Show Balloons from the Help menu.When you point to any feature of the Script Editor, a balloon appears explaining the feature. Where to go from here For more information about using the Script Editor to work with scripts, open the file called Using the Script Editor to see Chapter 1 ( Using the Script Editor ) and Chapter 2 ( Script Editor Command Reference ). If you wish to create more complex scripts, your local bookseller has several tutorial and reference books about scripting; you can get a list of them on the AppleScript web site on the Internet at http://applescript.apple.com . About AppleScript version 1.1.2 Mac OS 8 includes an updated version of the AppleScript scripting system. This new version corrects some problems with the previous versions, and has enhancements specific to Mac OS 8. Problems Corrected in This Version of AppleScript Multiple-character text item delimiters: Setting AppleScript s text item delimiters to a multiple-character value, then trying to get the text items of a string, would result in the system freezing. This has been corrected. Concatenation of record with empty list: Using the concatenation operator with a record on the left and an empty list on the right (for example, set x to {a:1, b:2} & {}) would cause a system crash. This has been corrected. Cancelling a Quit in the Script Editor: If the Results window of the Script Editor is open when you choose Quit, and when offered an opportunity to save an open script you cancel the Quit, the Script Editor would crash the next time the Results window was used. This has been corrected in version 1.1.2 of the Script Editor. Problems loading a large saved script: When loading a large script, AppleScript used an inaccurate value to determine whether a certain part of the script would fit in memory. This resulted in out of memory errors loading scripts that had just been saved. This has been corrected; it should now be possible to load the scripts that earlier versions of AppleScript would not. Recorded script won t run: If you record a script and save it immediately as a script application, it would not do anything. This has been corrected for scripts recorded with AppleScript 1.1.2. For scripts recorded with previous versions of AppleScript, simply open them with the Script Editor, recompile them, and save them again. Problems running on large machines or large virtual memory: Machines with more than 768 Megabytes of real or virtual memory could not compile any AppleScript script, giving an error A numeric value was too large. This has been corrected. Setting month property to a number gives the wrong error: The month property of a date is a symbol for the month name (e.g. January, May). If you assigned a number to the month property, AppleScript would report this as a Stack Overflow Error. It now says more correctly that it can t make the number into the correct type.

Problems with number formatting on certain numbers: AppleScript 1.1 displayed decimal numbers to 13 places to the right of the decimal point. With certain numbers, this is more accuracy than the underlying mathematics of the Mac OS can provide; therefore some numbers would display inaccurately when compiled. (For example, the number 1310.4 would be compiled and displayed as 1310.4000000000001). In AppleScript 1.1.2, only 12 digits to the right of the decimal point are displayed, so compilation shows expected results. NOTE: In general, the floating-point numbers used by AppleScript to represent quantities like 1310.4 cannot represent tenths exactly, and multiplication, division, and equality comparison will not necessarily give you what you expect. To test two floating-point numbers for equality, use a comparison like ((a – b) -1.0E+12). Inability to compile capitalized posessives: AppleScript uses ’s as a synonym for of , as in get AppleScript’s version. Previous versions would give a syntax error for get APPLESCRIPT’S VERSION , even though AppleScript is generally not case-sensitive. In version 1.1.2, AppleScript allows a capital S in posessives, as well as a capital T in DOESN’T . New Capabilities of This Version Larger symbol table size: Previous versions of AppleScript were limited to about 1,018 user-defined symbols (variables, properties, globals, and subroutine names), regardless of the amount of memory available. This limit has been expanded to over 4,000 user defined symbols. Support for new Control Panels: Several of the Mac OS 8 control panels are now driver applications of type APPD, and are scriptable. AppleScript can launch these control panels automatically when needed, just like other scriptable applications. AppleScript can now also launch the Finder and other all-Power PC Native applications, as well. Left margin, circulating log file in Script Editor: In the Script Editor, printing a script or dictionary will now reserve an additional 1/2″ on the left margin for three-hole punching. The Script Editor’s Event Log no longer generates a -20013 error after logging 20,000 characters; when too much text has been logged, the Script Editor will discard events and replies off the top of the log to make room for more. Mac OS 8 Appearance: The dialog boxes in AppleScript and the Display Dialog scripting addition now support the new Appearances feature. Mac OS 8 System Folder Realignment: In Mac OS 8, many things inside the System Folder have been rearranged. Most notably to AppleScript, the Scripting Additions folder has been moved from the Extensions Folder to the System Folder proper. AppleScript looks in this Scripting Additions folder for its dialects and scripting additions first. When you drop a scripting addition onto the closed System Folder, it will offer to place it in this Scripting Additions folder for you. To support scripts, applications, and installers that depend on the old location of the Scripting Additions folder, AppleScript will also recognize items placed in a Scripting Additions folder in the Extensions folder. But it will not recognize dialects in a Dialects folder there if dialects are already present in the System Folder-level Scripting Additions folder; and if duplicate scripting additions are found in the second folder, only the ones in the first folder will be functional. New selectors for Path To command in File Commands Scripting Addition: To help you find folders in the System Folder, the Path To command now recognizes the following folder names: A Note about Scripting the Finder Because of significant changes in the way the Finder works, the scripting implementation has been changed. For example, in the Finder in Mac OS 8, each window has its own Views properties for labels, calculating folder sizes, etc.; these were properties of the Views control panel in earlier versions of the System. Scripts that rely on behaviors of the Finder that have changed will need to be changed and recompiled. In some cases, when you open an old script, the term obsolete will be added after a Finder term (such as file type). This means that the way the Finder implements this feature has been changed, and the old way will not be supported in future versions of the Finder. In all cases the same Finder terms can be compiled to talk to the new Finder once the term obsolete is removed; these recompiled scripts, however, will not function with Finders prior to version 8.0. If you want to write Finder scripts that work with version 8.0 of the Finder as well as earlier versions, use the obsolete term where it s present in the Finder s dictionary. Be aware, though, that such scripts will not necessarily work on future versions of the Finder. Copyright 1997 Apple Computer, Inc. All rights reserved.