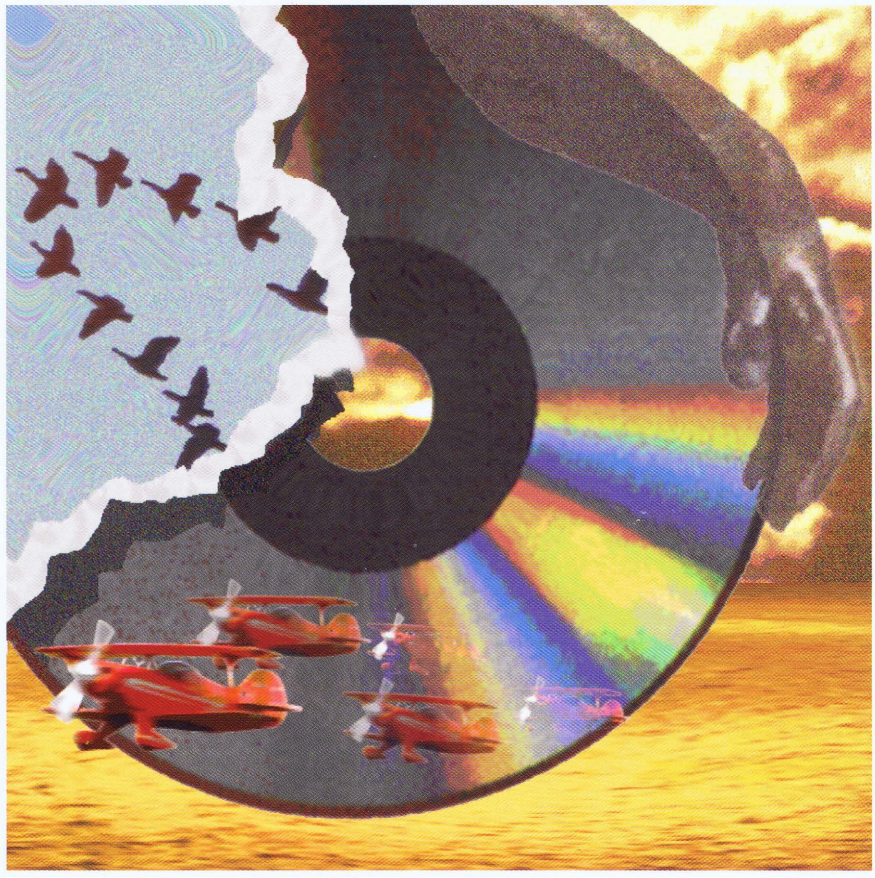
E.T.O.: Essentials–Tools–Objects



030-1991-A

About
E.T.O.: Essentials • Tools • Objects
(#22—DECEMBER, 1996)

CONTENTS

ABOUT E.T.O....

E.T.O.: Essentials•Tools•Objects is Apple's primary core development tools product. It is sold by subscription and is sent to subscribers automatically every four months (releases are scheduled for April, August, and December each year). E.T.O. provides professional Macintosh software developers with the latest development tools for developing 680x0- and PowerPC-targeted applications. Also included are important Pre-release versions of many tools.

The major software components included on the E.T.O. CD-ROMs are as follows:

• Macintosh Programmer's Workshop	• SourceBug debugger (68K)
• C and C++ for 68K Mac development	• Macintosh Debugger (68K and PowerPC)
• C and C++ for Power Mac development	• MacsBug debugger (68K and PowerPC)
• Assembler for 68K Mac development	• ResEdit resource editor
• Assembler for Power Mac development	• Virtual User testing tool
• MacApp application framework	• Macintosh Programmer's Assistant
• OpenDoc Development Framework	

E.T.O. is now provided on two CD-ROM volumes—one called "E.T.O. Frameworks" (disc 2) which contains the two frameworks products (MacApp and OpenDoc Development Framework) and the OpenDoc runtime software, and the other called "E.T.O." (disc 1) which contains everything else. In order to make items easier to locate, we've created top-level folders for all the major tool categories, including MPW, Debuggers, Interfaces and Libraries, and Documentation. In addition, we've placed both Pre-release and archival material in the ":Past&Future:" folder on the "E.T.O." volume. Finally, we've included an installer for the major components on the "E.T.O." volume. MacApp and ODF, on the "E.T.O. Frameworks" volume, can easily be installed by dragging folders and files to your hard disk as described below.

Installation: The E.T.O. installer, located in the "E.T.O. Installer" folder can be used to install many of the tools on E.T.O., although Pre-release components and other tools not included in the installer must be installed using the manual instructions at the end of this document. You can also use the manual instructions if you prefer not to use the installer.

NEW SOFTWARE RELEASES

The "Highlights" folder at the root level of the E.T.O. CD-ROM contains aliases to the folders of many of the new components that have been added since the previous release. Double-click an alias to move directly to the actual folder.

Here are summaries of the featured new releases in this edition:

MPW

(Path: ":MPW:")

There is a new version of ILink (v.2.1.1) in this E.T.O. release with several bug fixes and minor enhancements.

MPW PRE-RELEASE

(Path: ":Past&Future:Pre-release:MPW 3.4.2b3:")

Included in this release are new versions of MPW Shell, SourceServer and ToolServer. Besides bug fixes, we have included many enhancements such as grayscale appearance of windows, improved Projector performance and a resizable Projector window.

In addition, there are new versions of many of the MPW tools. These include Canon, Commando, ConvertExportList, DumpPEF, DumpXCOFF, FileDiv, Gestalt, Get, GetFileName, GetListItem, MergeFragment, Search, Sort, Unmangle and VersionList. DumpPEF now supports the MrCpp-style C++ exception-data sections and updated C++ name-unmangling. PPCLink also receives a new version with enhancements and bug fixes. See the MPW Release Notes for more information.

INTERFACES AND LIBRARIES

(Path: ":Interfaces&Libraries:")

Several interfaces and libraries have been revised to fix miscellaneous bugs. These include removing duplicate symbols in the ColorPickerLib and InterfaceLib. A new stub library for MathLib is included. For the first time, Game Sprockets Stub Libraries are also included.

INTERFACES AND LIBRARIES PRE-RELEASE

(Path: ":Past&Future:Pre-release:Interfaces&Libraries:")

On the libraries side, support for the double-double run time (ddrt) API and Math64 API have been added to all runtime environments. PPCCRuntime library and PPCStdCLib

have been enhanced. Runtime Type Information (RTTI) support has been added to several C++ libraries.
On the interfaces side, some enhanced header files are now included in the Language Library Interfaces. Please see the MPW Release Notes for more information.

MrC/MrCpp 3.0 Pre-release

(Path: ":Past&Future:Pre-release:MPW 3.4.2b3:Tools:")

This pre-release (3.0d1c8) has a new C++ v-table layout and an ABI that is incompatible with previous versions of this compiler, so be sure to rebuild all your libraries. It supports 64-bit integers (long long pre-defined type) and C++ RTTI. MacSOM pragmas and other new pragmas are also included.

MrC/MrCpp — Plugin Version for CodeWarrior

(Path: ":Past&Future:Pre-release:MrC/MrCpp CodeWarrior Plugins:CodeWarrior Plugins:")

This is a new Pre-release version (3.0d1) of the MrC/MrCpp compiler for use with the Metrowerks CodeWarrior environment. Note that it can be used only with CodeWarrior 9 or later. This compiler is undergoing rapid changes, so we recommend visitng our web site at "www.devtools.apple.com" frequently to ensure you're always aware of the most up-to-date version .

Direct-to-SOMobjects Software

(Path: ":Past&Future:Pre-release:Direct SOMobjects™ for Mac OS:")

This folder contains information on how to use the "Direct-to-SOM" capability of the new Pre-release version of the MrCpp compiler. You can use this capability to write SOM-based classes directly using C++, that is, without using the IDL language or the IDL compiler. This folder also contains new SOMobjects header files which must be used instead of the standard set that are installed with SOMobjects for Mac OS 2.0.8.

Power Mac Debugger 2.1a1

(Path: ":Past&Future:Pre-release:Macintosh Debugger:Power Macintosh Debugger:")

This alpha version of the Power Mac Debugger is feature complete and incorporates many bug fixes since E.T.O. #21.

OpenDoc Development Framework Release 2

(Path: ":OpenDoc Development Framework:") — on the "E.T.O. Frameworks" volume (disc 2)

OpenDoc Development Framework is a robust object-oriented framework designed to simplify the creation of OpenDoc parts. This is the second release of ODF with bug fixes. The OpenDoc runtime software and documentation is also included.

MACAPP 3.3.3

(Path: ":MacApp:MacApp 3.3.3:") — on the "E.T.O. Frameworks" volume (disc 2)

This is the current reference release of the MacApp framework. It has been certified with the lastest CodeWarrior 10 and MrC compilers.

MACAPP RELEASE 12

(Path: ":MacApp:MacApp Release 12:") — on the "E.T.O. Frameworks" volume (disc 2)

The release includes lots of bug fixes. It also contains a new version of Ad Lib, the MacApp view editor, with significant user interface enhancements. A debugging aid, TidyHeap, which detects memory leaks is also included. Be sure to read the Release Notes that come with Release 12.

RESEDIT 3.0D3

(Path: ":Past&Future:Pre-release:ResEdit 3.0d3")

ResEdit 3.0 is a completely new implementation of ResEdit, designed to provide a powerful, extensible, and maintainable editing solution for resource files, PEF files, and other file types. It will eventually replace the functionality of the current final release of ResEdit. This release of ResEdit (v3.0d3) provides the general application framework, and includes editors for many resource types.

MRPLUS 1.0D4

(Path: ":Past&Future:Pre-release:MrPlus 1.0d4:")

This is a new Pre-release version of MrPlus, a new performance tuning and analysis tool for accelerating native Power Macintosh applications. Included in this release are bug fixes and an instrument imports and exports feature. MrProf has a revamped user interface with bug fixes.

INSTRUMENTATION SDK 1.0 PRE-RELEASE

(Path: ":Past&Future:Pre-release:Instrumentation SDK 1.0:")

This is the first release of the Mac OS 7.5 Instrumentation SDK, a set of applications and libraries that allow you to instrument your 68K and PowerPC code with traces and statistics under System 7.5.

MAC OS RUNTIME FOR JAVA

(Path: "Other Development Tools:Mac OS Runtime for Java 1.0b1")

This is the first inclusion of Mac OS Runtime for Java (MRJ) on ETO. It includes the MRJ Runtime, a supporting cast of sample applications, sample applets, and documentation concerning several MRJ APIs - JRI, JAM, Debug.

MACINTOSH PROGRAMMER'S ASSISTANT

(Path: "Documentation:Online Reference")

We've included a new version (v2.3f4) of the QuickView application used to read the Programmer's Assistant databases.

MPBUTTONS

(Path: "Past&Future:Pre-release:MPW 3.4.2b3:MPButtons:")

MPButtons is a development utility providing a set of extensions to the MPW environment. It provides searching for string with a single click, multiple smart clipboards which perform copy, paste, cut, and swap, commenting and uncommenting source code, source overview, searching and pasting items from Toolbox Assistant or THINK Reference 2.0, and API for adding custom functionality.

OBJECTSET MAIL SDK

(Path: ":MacApp:MacApp Release 12:ThirdParties:SmartCode Software, Inc.:Internet Mail SDK:") — on the "E.T.O. Frameworks" volume (disc 2)

Included with this release of MacApp is the ObjectSet Demo from SmartCode Software, Inc. ObjectSet is a line of reusable software components and class libraries that programmers can easily include in their own applications; they integrate particularly well with MacApp-based applications. The first toolkit in the line is the ObjectSet MAIL SDK, which includes complete APIs for SMTP, POP3 and MIME protocols. The demo allows you to experiment with these protocols via a fully-functional e-mail application. Check it out!

DOCUMENTATION AND RELEASE NOTES

The ":Documentation:" folder contains electronic versions of the reference books for the major software components of E.T.O. Most of these documents are provided in Adobe Acrobat format for convenient viewing, searching, and printing.

Two versions of the Adobe Acrobat application are provided in the ":Utilities:Adobe Acrobat:" folder—Acrobat Exchange LE and Acrobat Reader. If you are a single-user licensee, we recommend you use Acrobat Exchange LE since it provides better searching capabilities than Acrobat Reader. If you are a site licensee, please read the document called "Exchange LE License Exclusion" in the Adobe Acrobat folder.

To install Acrobat Reader, simply double-click the file called "AcroRead.mac" in the Adobe Acrobat Reader folder. To install Acrobat Exchange LE, double-click the Installer file inside the Acrobat Exchange LE folder.

Most command reference documentation and release notes are now provided in the QuickView format and are located in the Online Reference folder. Aliases to these documents are located in the folders containing the software components they describe. As a convenience, the E.T.O. Release Notes also provide pointers to release notes which are not in the QuickView format. The QuickView format lends itself to rapid lookup and browsing of information. The QuickView application is located in the ":Documentation:Online Reference:" folder and can be installed by moving it to any convenient location on your hard disk. (See the section below called "Macintosh Programmer's Assistant Installation Instructions" for information on how to install QuickView along with the major reference databases we provide with it.)

FINAL AND PRE-RELEASE VERSIONS

Most of the software included on the E.T.O. CD-ROM is of "final" quality (formerly designated as "Latest") and is fully tested. We also provide Pre-release versions of many tools and these can be found in the ":Past&Future:Pre-release:" folder. The software in this folder contains non-final versions that are being used internally at Apple and are generally of sufficient quality for most developers to build stable interim versions of their work.

SITE LICENSING

E.T.O. is a single user product. Like any single user product, you may not make copies for anyone else. Each time you receive a new update release you may not "pass along" the old release to another user. See the software license included with the CD-ROM (and stored in electronic form in the Licensing Info folder). Note that this license allows you to distribute certain software included on the E.T.O. CD-ROM with your applications.

E.T.O. is available for site licensing—terms are available that make it attractive for sites with as few as ten users. For information on the terms of this site license, contact:

Software Licensing Department
Apple Computer, Inc.

2420 Ridgepoint Drive, MS: 198-SWL
Austin, TX 78754
Telephone: 512-919-2645
Fax: 512-919-2120
AppleLink: SW.LICENSE
Internet: SW.LICENSE@applelink.apple.com

Note that special pricing exists for site licenses for accredited educational institutions.

How to Report Bugs

If you detect technical problems with any of the components on the E.T.O. CD-ROM, we want to know about it. Use the Apple Bug Reporter on the E.T.O. CD-ROM to simplify the creation of a formal bug report. Be sure to move it to your hard disk before opening it.

If you are an Apple Partner, please send a detailed bug report to us via Internet mail at devsupport@applelink.apple.com (or via AppleLink at DEVSUPPORT)—you will receive a confirmation of receipt right away and, if appropriate, a reply. We will also log the report into our internal bug database, but we cannot promise immediate resolution of every problem.

Other developers may send reports to us via Internet mail at apple.bugs@applelink.apple.com (or via AppleLink at APPLE.BUGS). This is primarily a one-way address and you should not expect a personal reply to your bug report. We will, however, log the report and consider it when the component is next revised.

If you do not have access to e-mail, you can send us a written report at:

Developer Tools Group
Apple Computer, Inc.
1 Infinite Loop, MS: 303-3D
Cupertino, CA 95014

Please mark the outside of the envelope with the words "Bug Report".

Apple Developer Catalog

The Apple Developer Catalog is Apple's source for development tools and related programming products and resources. The catalog describes hundreds of Apple and third-party products. Contact the Apple Developer Catalog team at the following address:

Apple Developer Catalog
Apple Computer, Inc.
P.O. Box 319
Buffalo, NY 14207-0319
800-282-2732 (U.S.A.)
800-637-0029 (Canada)
716-871-6555 (International)
716-871-6511 (Fax)

The Apple Developer Catalog can also be viewed on the world wide web at http://devcatalog.apple.com/. You can also place orders via the web.

How to Contact Apple

We welcome your general comments on our products. You can contact us at the following address:

Developer Tools Product Marketing
Apple Computer, Inc.
1 Infinite Loop, MS: 303-3X
Cupertino, CA 95014
Fax: 408-974-9456
Internet: dev.tools@applelink.apple.com
AppleLink: DEV.TOOLS

Next Release of E.T.O.

The next release (#23) of E.T.O. is scheduled for April, 1997. You will receive the next release of E.T.O. automatically if your subscription has not expired. (The packing list included with the #22 mailing indicates the number of issues remaining on your subscription.)

For up-to-the-minute news and technical information about E.T.O., we recommend you visit our world wide web site at **http://www.devtools.apple.com/**. We also recommend you subscribe to the mpw-dev Internet mailing list by sending a message to **listproc@solutions.apple.com** and putting the following command in the body of the message: **SUBSCRIBE mpw-dev yourRealName**.

GENERAL INSTALLATION INSTRUCTIONS

E.T.O. now includes an automated installer to simplify the installation of most of the major development tools and related files. It is located at the root level of the CD-ROM—simply double-click it to run it.

This section, and the sections which follow it, cover general installation issues and explain how to install individual components without using the installer application. They also explain how to install Pre-release components. Most users, however, just need to use the installer and do not have to read this material. More advanced users can read on for useful supplemental information and usage tips.

The first component to install from the E.T.O. CD-ROM is MPW:

Macintosh Programmer's Workshop (MPW). This is the development environment included on the E.T.O. CD-ROM. Installing it automatically installs SC and SCpp (68K-code-generating C and C++ compilers), MrC and MrCpp (PowerPC-code-generating C and C++ compilers), the MPW Assemblers (68K and PowerPC), and the MPW tools.

The following major components may then be installed in any order:

	**Macintosh Programmer's Assistant**. This is a set of on-line databases which provides rapid up-to-date access to information on many aspects of the Macintosh programming. The *Toolbox Assistant* provides information about APIs, toolbox managers, data structures, routines, constants, and resources. The *MPW Command Assistant* provides usage information for MPW tools and scripts. The *Standard C Library Reference* explains the routines included in the ANSI standard language library. The *MacApp Assistant* provides complete reference information about Classes, Methods, Data Types, Global Variables, and Macros, for MacApp version 3.3. And last but not least, the *E.T.O. #22 Release Notes* provide last-minute information about the contents of E.T.O. #22. Macintosh Programmer's Assistant includes MPW scripts and tools that let you access it directly from the MPW environment.
	**MacApp**. This is Apple's advanced object-oriented application framework. It can be used for development of 68K- and PowerPC-based applications.
	**OpenDoc Development Framework (ODF)**. This is Apple's framework for simplifying the creation of OpenDoc parts.
	**Power Macintosh Debugger**. This is Apple's debugger for debugging native Power Macintosh code.

Installation instructions for most of the above components are included later in this document.

Installation of most other components usually involves dragging folders containing the components from the CD-ROM to any convenient location on your hard disk. Use this technique for the following applications:

	**ResEdit**. This is a general-purpose editor for creating Macintosh resources.
	**SourceBug**. This is a 680x0 source-level debugger.

Virtual User. This is an automated testing tool. (The companion AgentVU tool must be dropped in the system folder of the target machines being tested.)

Special installation instructions apply to the MacsBug debugger:

MacsBug. This is a 680x0 and PowerPC assembly-level debugger that can be installed by dragging the MacsBug file to the System folder on your startup volume.

MACINTOSH PROGRAMMER'S WORKSHOP (MPW) INSTALLATION INSTRUCTIONS

The ":MPW:" folder contains the complete MPW development system, including release notes, documentation, and required system additions.

Required System Extensions

MPW includes versions of the MPW Shell and tools which run in native mode on a Power Macintosh. They are "fat" binaries, so they will also run on 680x0-based Macintosh systems. However, on a Power Macintosh system, these tools and MPW Shell require an updated StdCLib library for system software prior to version 7.6. The updated StdCLib is included in the ":System Additions for MPW:" folder inside the MPW folder. This file goes into the Extensions folder inside the System folder of your Power Macintosh. To install it, drop StdCLibInit on top of the System folder icon (not into an open System folder) of your Power Macintosh and restart your system so that the extension can execute prior to launching the new MPW Shell. (If the updated StdCLib is not installed properly, the system will notify you that StdCLib cannot be found when you attempt to launch the MPW Shell.)

The ":System Additions for MPW:" folder also includes versions of Clipping Extension and Macintosh Drag and Drop that are required on pre-System 7.5 systems to use the new drag & drop features of the MPW Shell.

The MPW system occupies approximately 30 Mb on disk. Be sure you have that much free space available before trying to install MPW for the first time.

Some of the files included in previous releases of MPW cannot be used with the newer MPW environments. These files are now obsolete due to changes in runtime models or file organization. For example, the old "C" and "CFront" tools cannot be used in the current MPW environment because they are incompatible with the new 68K libraries and do not support the new CFM-68K runtime model. For these kinds of obsolete files we have substituted "dummy" files in the MPW folder which have a modification date of Jan 01, 1998 and generate warnings that cause an abort if they are executed (for tools or scripts) or if they are included (for headers or libraries). The contents of these files also explain the changes you need to make to your make files. We recommend that you leave these dummy files in place until you have finished bringing your sources and build files up to date with the new environments. These files may help flush out places in your source code that need to be changed. Once you are satisfied that your source code is up to date, you can delete these "dummy" files with the following commands:

```
Backup -r -l -do only,'Delete -y -i' -since '01/01/98' -from "{MPW}Scripts"
Backup -r -l -do only,'Delete -y -i' -since '01/01/98' -from "{MPW}Tools"
Backup -r -l -do only,'Delete -y -i' -since '01/01/98' -from "{CIncludes}"
Backup -r -l -do only,'Delete -y -i' -since '01/01/98' -from "{AIncludes}"
Backup -r -l -do only,'Delete -y -i' -since '01/01/98' -from "{Libraries}:"
```

Review, select, and execute the resulting Delete commands.

LATEST MPW

The ":MPW:" folder contains a complete suite of preconfigured "final quality" MPW software components, including the assemblers and C and C++ compilers for development of 68K and Power Macintosh software. We recommend you use this version of the MPW system for your day-to-day development work.

First-Time Installation

If you have never installed MPW on your hard disk before, install it by dragging the ":MPW:" and ":Interfaces&Libraries:" folders to your hard disk. Also, move the files inside the ":MPW:System Additions for MPW:" folder to the System folder on your hard disk (and reboot) as explained in the Required System Extensions note above. You run the MPW environment by launching the application called "MPW Shell" inside the MPW folder on your hard disk.

Updating an Earlier Version

If you have previously installed an earlier version of the MPW system, launch the MPW Shell from your hard disk and install Latest MPW by executing the following commands from the Worksheet window:

```
if not `exists -d "{MPW}"busyFolder`
     newfolder "{MPW}"busyFolder
end
directory "E.T.O. #22:MPW:"
```

```
duplicate -y ":System Additions for MPW:" "{SystemFolder}"Extensions
Backup -r -a -c -from : -to "{MPW}" -busy "{MPW}"busyFolder
Backup -r -a -c -from "::Interfaces&Libraries:" -to "{CIncludes}::"
```

(To execute, remember to type Enter, not Return, after selecting these command lines in the Worksheet.)

These commands generate a list of Duplicate commands for those files in the ":MPW:" folder on the CD-ROM that are either not present in the MPW folder on your hard disk or have been modified more recently. Execute all the Duplicate commands by selecting them, then typing the Enter key. This moves all the new files to the proper folders on your hard disk without disturbing any files you may have added to the MPW hierarchy.

After updating, quit the MPW Shell that's running, then run the new one that's replaced it in the MPW folder. (The old MPW Shell is located in the "busyFolder" inside the MPW folder and can be deleted at your convenience.)

PRE-RELEASE MPW

The ":Past&Future:Pre-release:" folder contains Pre-release versions of many of the software components that make up the MPW development environment. These components are not of final quality, but we recommend you consider them for day-to-day use. In some cases, these components may be more appealing to use because they fix known bugs in MPW components or have been enhanced; they have not, however, been tested against a broad range of system configurations and operating conditions.


Subset only!

Unlike the main ":MPW:" folder, the MPW folder inside the ":Past&Future:Pre-release:" folder does *not* contain a complete MPW system. You must first install MPW as described above, then install the Pre-release MPW components as described below.

The easiest way to install Pre-release components is to first run the MPW Shell you installed on your hard disk as described in the previous section. Next, execute the following commands from the MPW Worksheet window:

```
if not `exists -d "{MPW}"busyFolder`
     newfolder "{MPW}"busyFolder
end
directory "E.T.O. #22:Past&Future:Pre-release:MPW 3.4.2b3:"
Backup -r -a -c -from : -to "{MPW}" -busy "{MPW}"busyFolder
Backup -r -a -c -from "::Interfaces&Libraries:" -to "{CIncludes}::"
```

(To execute, remember to type Enter, not Return, after selecting these command lines in the Worksheet.)

These commands generate a list of Duplicate commands for those files on the CD-ROM that are either not present in the MPW folder on your hard disk or have been modified more recently. Execute all the Duplicate commands by selecting them, then typing the

Enter key. This moves all the new files to the proper folders on your hard disk without disturbing any files you may have added to the MPW hierarchy.

After updating, quit the MPW Shell that's running, then run the new one that's replaced it in the MPW folder. (The old MPW Shell is located in the "busyFolder" inside the MPW folder and can be deleted at your convenience.)

MPW ARCHIVES

The MPW Archive folder, located at ":Past&Future:Archive:MPW Archive:", contains obsolete MPW components which are no longer maintained or supported, but which you may need when working with older software projects. Included here are MPW C 3.3.3 (superseded by SC), MPW Pascal (now available under the name LS Object Pascal from Fortner Research), and MPW PPCC (superseded by MrC/MrCpp).

MACAPP INSTALLATION INSTRUCTIONS

The ":MacApp:" folder on the E.T.O. Frameworks CD-ROM (disc 2) contains the following releases of MacApp:

MacApp 3.3.3	The current reference release of MacApp.
MacApp Release 12	The release includes lots of bug fixes. It contains a new version of Ad Lib, the MacApp view editor, with significant user interface enhancements. A debugging aid, TidyHeap" which detects memory leaks is also included. Be sure to read the Release Notes that come with Release 12.

You should install MacApp only after installing Macintosh Programmer's Workshop, as described earlier in this document, or Metrowerks CodeWarrior or Symantec C++ for Macintosh.

The MacApp 3.3.3 system, as provided on the E.T.O. CD-ROM, occupies approximately 24 Mb on disk. Be sure you have that much free space available before trying to install MacApp.

We recommend you use a Macintosh system with at least 24 Mb of memory for developing MacApp applications. More memory may be required depending on the size of the application you are creating. In addition, you should change the MPW Shell minimum partition size to at least 5000K by changing the value in the *Minimum size* field in the Finder Get Info box for MPW Shell. A larger partition size may be required depending on the size of your project.

MACAPP 3.3.3

This folder contains MacApp 3.3.3. Listed below are the steps you need to follow to install MacApp 3.3.3 on your hard disk. We assume that you have already installed Macintosh Programmer's Workshop (or Metrowerks CodeWarrior or Symantec C++ for Macintosh).

- Drag the ":MacApp:MacApp 3.3.3:" folder to any convenient location on your hard disk. If you are using Symantec C++ for Power Macintosh or Metrowerks CodeWarrior, read the MacApp 3.3.3 Release Notes for information on how to install the necessary support files in these environments.

- Drag the file called "UserStartup•MacApp" from the "MacApp 3.3.3 - HD Ready" folder inside the MacApp 3.3.3 folder on your hard disk to the folder on your hard disk where the MPW Shell is located.

- Look inside the folder where the MPW Shell is located. If there is a file there called "MacApp_Folder", drag it to the trash. (This file contains the pathname of a previously-installed MacApp folder. You will probably want to delete this old MacApp folder as well unless you need to continue to use it for some reason.)

- Run the MPW Shell. MPW will ask you *Where is your "MacApp" folder?*. Move into the "MacApp 3.3.3 - HD Ready" folder on the hard disk using the standard folder navigation dialog, then click the *Select Current Directory:* button.

- (Optional.) Copy the contents of the "insert to mpw.help" file (inside the "MacApp 3.3.3 - HD Ready" folder) and paste them at the end of the "MPW.Help" file in the folder in which the MPW Shell is located. This will make the help for MacApp accessible from the MPW command line. For example, when you type "Help MABuild", the help text for the MABuild command should appear.

MACAPP RELEASE 12

To install MacApp Release 12, follow the same general procedure described above for MacApp 3.3.3, but drag the folder called ":MacApp:MacApp Release 12:" to your hard disk instead.

OpenDoc Development Framework Installation Instructions

This note explains how to install the OpenDoc Development Framework Release 2 when you are using it with the MPW development environment. For instructions on how to install it for use with Metrowerks CodeWarrior or Symantec C++ for Power Macintosh, see the notes in the Getting Started folder inside the OpenDoc Development Framework folder on the E.T.O. Frameworks CD-ROM.

OpenDoc Runtime Software Installation

Before installing ODF, install the OpenDoc runtime software by running the Installer application located inside the ":OpenDoc Support for ODF:Installing OpenDoc™:" folder on the E.T.O. Frameworks CD-ROM. You must install the OpenDoc runtime in order to be able to run the parts you create with ODF. Note that the OpenDoc installer also installs the required SOMobjects™ for Mac OS software.

ODF Release 2

Before installing ODF, you must make sure that your MPW environment is configured for compiling ODF. To do that, you must do the following:

- Install MPW.
- Install the MPW Additions.
- Install the OpenDoc headers, libraries, and build support for MPW.

1) Install MPW

Install the MPW environment using the Installer on disc 1 of the E.T.O. #22 CD-ROM set.

IMPORTANT: You must use the MrC and SCpp compilers found on E.T.O. #22 (or later) to compile ODF, because ODF uses the template instantiation mechanisms of these compilers for its template-based classes.

2) Install the MPW Additions

First navigate into the ":OpenDoc Development Framework:Getting Started:MPW Development:" folder on the E.T.O. Frameworks CD-ROM. The folder here called "MPW Additions" contains additions to your MPW environment that are necessary to build ODF. They include startup files as well as files needed to use SOMobjects™ for Mac OS. To install:

- Open the MPW Additions folder and move the files UserStartup•OpenDoc and UserStartup•ODF into the Startup Items folder inside the MPW folder on your hard disk (i.e., the folder containing the MPW Shell). Move the file UserStartupTS•ODF into the TS Startup Items folder.

- Open the SOMobjects™ for Mac OS folder, then the MPW Additions folder. Move the UserStartup•somc file into the Startup Items folder inside the MPW folder on your hard disk. Move the contents of Scripts, Tools, and Examples folder into the corresponding folders inside the MPW folder on your hard disk. Move the contents of the subfolders of the Interfaces folder and the Libraries folder to the corresponding subfolders inside the Interfaces&Libraries folder on your hard disk (this latter folder is located at the same folder level as your MPW folder) -- if there is not a corresponding subfolder, create one first.

  An easier way to install the SOMobjects MPW Additions is to execute the following lines from the worksheet of the MPW Shell:

```
Directory "E.T.O. Frameworks #22:OpenDoc Development Framework:Getting Started:"
Directory ":MPW Development:MPW Additions:SOMobjects™ for Mac OS:MPW Additions:"
Duplicate -y "UserStartup•somc" "{MPW}Startup Items"
Backup -r -a -c -from ":Examples:" -to "{MPW}Examples"
Backup -r -a -c -from ":Scripts:" -to "{MPW}Scripts"
Backup -r -a -c -from ":Tools:" -to "{MPW}Tools"
Backup -r -a -c -from ":Interfaces:" -to "{CIncludes}:"
Backup -r -a -c -from ":Libraries:" -to "{SharedLibraries}:"
```

  These command generate a set of Duplicate commands which, when selected and executed, will move all the files into place.

3) **Install the OpenDoc headers, libraries, and build support for MPW.**
   In the "OpenDoc Development" folder (inside the "OpenDoc Support for ODF" folder at the root level of the CD-ROM), you will find two folders called "OpenDoc" and "Build Support". Copy them to the MPW folder on your hard disk.

Installing ODF

For the most part, installing ODF is simply a matter of copying the ODF source and tools to your local hard drive. However, since ODF supports several different compilers, there are some files that you don't need to copy if you are doing all of your development with MPW.

The basic steps you need to follow when installing ODF for MPW are as shown in the following list. These steps are explained in more detail below.

- Copy the ODFDev folder to your local hard drive.
- Copy the stationery files for the ODF sample parts to your Stationery folder. (optional)
- Copy any desired reference materials to your local hard drive. (optional)
- Make an alias to the ODF library (and sample parts).

- Relaunch the MPW Shell.

Note: These steps describe how to install the Debug version of the ODFLibrary and examples. If you simply want to play with the Release version of the ODF parts, copy the appropriate files from the "ODF Sample Parts" folder to your system. For details on doing this, see the information about this folder in the Contents of This CD section of the Important Information document inside the OpenDoc Development Framework folder.

1) Copy the ODFDev folder to your local hard drive.
Copy the ODFDev folder (in the OpenDoc Development Framework folder) in its entirety to your local hard drive. This folder contains the ODF source code and the source code for ODF's example parts. The :ODFDev:ODF: subfolder also contains configurations and project setups for building the ODF libraries.

Once you've copied ODFDev, you can throw away folders for other environments. Inside the ODF folder, you can throw away the RB (Symantec C++ for Power Macintosh) and CW (CodeWarrior) folders. You only need to keep the MC (MrC/MrCpp for MPW) and/or SC (SC/SCpp for MPW) and SL (Shared Library) folders. The same is true for all the example projects.

The following table indicates for each folder which compiler is used and the resulting binary. Notice that the 68K ODF Shared Library is currently built with the MPW 68K C++ compiler (SCpp) and the PPC ODF Shared Library is built with CodeWarrior.

Folder	Compiler Used	ODF Folder	Sample Folders
CWPPCDebug	CodeWarrior	PPC Debug ODF Static Libraries	PPC Debug Part Editor
CWPPCRelease	CodeWarrior	PPC Release ODF Static Libraries	PPC Release Part Editor
MCPPCDebug	MrC	PPC Debug ODF Static Libraries	PPC Debug Part Editor
MCPPCRelease	MrC	PPC Release ODF Static Libraries	PPC Release Part Editor
RBPPCDebug	Symantec C++	PPC Debug ODF Static Libraries	PPC Debug Part Editor
RBPPCRelease	Symantec C++	PPC Release ODF Static Libraries	PPC Release Part Editor
CW68KDebug	CodeWarrior	68K Debug ODF Static Libraries	68K Debug Part Editor
CW68KRelease	CodeWarrior	68K Release ODF Static Libraries	68K Release Part Editor
SC68KDebug	SCpp	68K Debug ODF Static Libraries	68K Debug Part Editor
SC68KRelease	SCpp	68K Release ODF Static Libraries	68K Release Part Editor

SLPPCDebug	CodeWarrior	PPC Debug ODF Shared Library	n/a
SLPPCRelease	CodeWarrior	PPC Release ODF Shared Library	n/a
SL68KDebug	SCpp	68K Debug ODF Shared Library	n/a
SL68KRelease	SCpp	68K Release ODF Shared Library	n/a

The following table indicates which folders are necessary for each supported compiler.

Folder	CW/PPC	CW/68K	SCpp	MrC	Symantec C++
CWPPCDebug	Y	N	N	N	N
CWPPCRelease	Y	N	N	N	N
MCPPCDebug	N	N	N	Y	N
MCPPCRelease	N	N	N	Y	N
RBPPCDebug	N	N	N	N	Y
RBPPCRelease	N	N	N	N	Y
CW68KDebug	N	Y	N	N	N
CW68KRelease	N	Y	N	N	N
SC68KDebug	N	N	Y	N	N
SC68KRelease	N	N	Y	N	N
SL68KDebug	N	Y	Y	N	N
SL68KRelease	N	Y	Y	N	N
SLPPCDebug	Y	N	N	Y	Y
SLPPCRelease	Y	N	N	Y	Y

2) Copy the stationery files for the ODF example parts to your Stationery folder. (optional)
If you want to run the ODF sample parts, you will need to copy the stationery files for these parts into your OpenDoc Stationery folder (located, by default, at the root level of your startup volume). The stationery files are located in the ":ODF Sample Parts:Stationery:" folder inside the OpenDoc Development Framework folder on the CD-ROM.

3) Copy any desired reference materials to your local hard drive. (optional)
ODF comes with several documents and tools, such as the ODF Class Reference and MacBrowse, that can make it easier to navigate through the ODF code. You can copy any of these to your local hard drive.

The "Documentation" folder contains the ODF Class Reference, which is a QuickView database that allows you to quickly search for the documentation for a particular class or

method. Make sure that the QuickView application and the ODF Assistant document are in the same folder when you launch the application.

The ":Tools & Goodies:MacBrowse:" folder contains the MacBrowse application that enables you to browse the ODF source code directly. It also contains a subfolder with the MacParse tool, which you can use to create parsed files for use with MacBrowse.

The ":Tools & Goodies:Object Master™:" folder contains an empty ObjectMaster project, which you can use to create a browseable set of the ODF source code.

4) Make an alias to the ODF library (and sample parts).
To make debugging with ODF easier, you need to make an alias to the debug version of the ODF shared library and put the alias into your Editors folder (in the system folder of your startup volume). If your ODF environment is on a different volume than your system folder, you can create an Editors folder at the root level of your development hard drive and put the alias there. (OpenDoc does not allow you to put aliases to files on different volumes in the Editors folder.)

Note: The ODFLibrary (or an alias to it) must be located in your Editors folder. Part editors that were built with ODF require the existence of this library and will not run without it.

The debug versions of the ODF shared library (ODFLibrary) are already built. The ODFLibrary for PowerPC is located in the ":ODFDev:ODF:SLPPCDebug:Bin:" folder. The ODFLibrary for 68K is located in the ":ODFDev:ODF:SL68KDebug:Bin:" folder.

If you want to debug the ODF sample parts, you can also add an alias to each of these part editors to your Editors folder. The debug version of an ODF part editor is located within either the ":MCPPCDebug:Bin:" folder for PowerPC or the ":SC68KDebug:Bin:" folder for 68K of the particular sample part. For example, the debug, PowerPC part editor for ODFDraw is located in the ":ODFDev:Draw:MCPPCDebug:Bin:" folder.

For convenience, you can also make an alias to the symbol files for the ODFLibrary (ODFLibrary.xSYM) and put it on your desktop. You can then use this alias to find the ODFLibrary debugging information quickly when you launch your debugger.

IMPORTANT: Make sure that neither the release version of the ODFLibrary nor any of the ODF sample parts is installed on your machine while you are attempting to trace through either the ODFLibrary or sample parts. The chances are good that you will not get the right library. OpenDoc searches for the first part editor and library that matches its search parameters. The release version of the ODF libraries and samples are simply optimized versions that do not have internal debugging information, such as MacsBug symbols or assertions.

5) Relaunch the MPW Shell.

If the MPW Shell application is running, quit it, then launch it again. It will ask you to specify the location of your ODF folder. The folder you want to specify is ":ODFDev:ODF:" on your hard disk. MPW may also ask you to specify the location of your OpenDoc (or OpenDoc SDK) files (and perhaps more than once -- sorry about that!). If the OpenDoc files are installed inside your MPW folder as described above, simply select the OpenDoc folder inside your MPW folder (this folder contains OpenDoc interfaces and libraries). Finally, you may be asked to locate the Build Support folder -- this folder is also located inside the MPW folder.

POWER MACINTOSH DEBUGGER INSTALLATION INSTRUCTIONS

This note explains how to install the Power Macintosh Debugger v2.0 and also the Pre-release v2.1d14.

POWER MACINTOSH DEBUGGER 2.0

Power Macintosh Debugger 2.0 is designed for debugging native Power Macintosh code and can act as a one- or two-machine debugger. The debugger host communicates with the target application using Power Mac DebugServices, a "high-level" nub that uses either a local connection via direct callbacks, or a remote connection via AppleTalk. In remote (two-machine) mode, the host can also communicate with the target via PPC Debugger Nub, a "low-level" nub that uses a serial connection. The high-level nub should be used, where possible, because it does not stop the whole target machine, only the process being debugged. Conversely, the low-level nub takes control of the whole target system, but is necessary to debug low-level code such as code that runs at interrupt time. The serial cable required is a Mini DIN-8 to Mini DIN-8 serial cable (Apple Part #M0197).

Perform the following steps to install the debugging system:

- If you will be using the low-level nub (PPC Debugger Nub), connect the serial cable to the modem ports of the host and target systems. (You can also use the printer port on one or both systems by setting preferences in the Power Mac Debugger host application and in the Debugger Nub Controls control panel on the target system.)

- Insert the E.T.O. CD-ROM in the CD drive of the host Macintosh and navigate into the ":Debuggers:Macintosh Debugger:" folder.

- Drag the "Power Macintosh Debugger" folder to your hard drive. This folder contains the host debugger application (Power Mac Debugger 2.0) and some ancillary files.

- Open the "PowerPC System Folder Additions" folder inside the "Power Macintosh Debugger" folder.

- If the host is a Power Macintosh system, and it is running a version of system software prior to 7.5, select the file called PPCTraceEnabler (inside the "Pre-System 7.5

Only" folder) and drop it on top of the System folder icon (not into an open System folder). This copies the PPCTraceEnabler file into the Extensions folder.

- If you will be doing remote (two-machine) debugging, eject the E.T.O. CD-ROM and place it in the CD drive of the target Power Macintosh and navigate into the ":Debuggers:Macintosh Debugger:Power Macintosh Debugger:" folder. Then:

- Drag the Power Mac DebugServices nub to any convenient location on the target hard drive (it doesn't matter where, although you can put it in the Startup Items folder in the System Folder so that it is automatically launched at boot time).

- Open the "PowerPC System Folder Additions" folder.

- If you wish to use the low-level nub, select the three files in this folder called Debugger Nub Controls, PPC Debugger Nub, and PPCDebuggerNubINIT, and drop them on top of the System folder icon on the target Power Macintosh (not into an open System folder). This copies the Debugger Nub Controls file into the Control Panels folder, the PPC Debugger Nub file into the Extensions folder, and the PPCDebuggerNubINIT file into the Extensions folder. ***Note that if you install these files, but later want to use the high-level Power Mac DebugServices nub, you must deactivate the low-level nub using the Debugger Nub Controls control panel.***

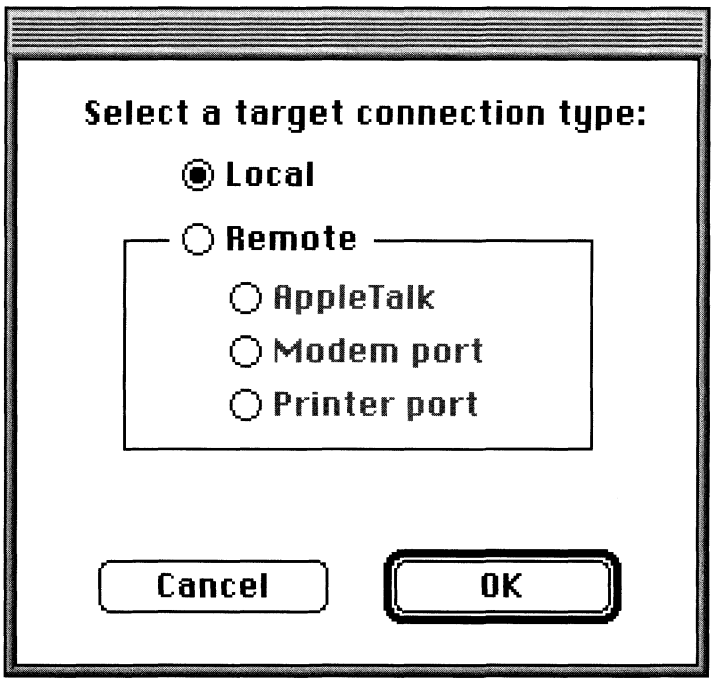
- If the target Power Macintosh is running a version of system software prior to 7.5, select the file called PPCTraceEnabler (inside the "Pre-System 7.5 Only" folder) and drop it on top of the System folder icon (not into an open System folder). This copies the PPCTraceEnabler file into the Extensions folder.

- Restart the target Macintosh system.

- Restart the host Macintosh system. To use the high-level nub, launch the Power Mac DebugServices nub application on the target system (this will be the host system if you are debugging in single-machine mode). If you are debugging in two-system mode, and you have installed the low-level nub on the target system (see previous step), you must deactivate the low-level nub using the Debugger Nub Controls control panel before launching Power Mac DebugServices.

If you use the high-level nub for two-machine debugging, you must use the Sharing Setup control panel on the target system to turn on Program Linking. In addition, you must use the Users & Groups control panel on the target system to specifically grant program linking privileges for the user of the host system. Do this by first creating a New User icon in the Users & Groups control panel and giving it a name which is the same as the Owner Name of the host Macintosh. (This name is set using the Sharing Setup control panel on the host Macintosh.) Then, double-click the user icon just created and select the *Allow user to link to programs on this Macintosh* check box. Alternatively, you can double-click the <Guest> user icon and select the *Allow user to link to programs on this Macintosh* check box to grant program linking privileges to guests.

• Run the debugger host by launching the Power Mac Debugger 2.0 application. When it starts up you will see the following dialog:



Click the *Local* radio button if you want to debug an application running on the same system as the debugger host using the high-level Power Mac DebugServices nub.

Click the *Remote* radio button if you want to debug an application running on a Power Macintosh to which the host system is connected via an AppleTalk connection or a serial connection. If you click the *AppleTalk* radio button, the high-level Power Mac DebugServices nub is used; if you click *Modem port* or *Printer port*, the low-level nub is used.

If you choose to debug using the Power Mac DebugServices nub (by choosing Local or Remote/AppleTalk in the dialog box above), you must deactivate the low-level nub using the Debugger Nub Controls control panel.

POWER MACINTOSH DEBUGGER 2.1A1

This Pre-release version of the Power Macintosh Debugger host is located in the ":Past&Future:Pre-release:Macintosh Debugger:Power Macintosh Debugger:" folder. To install it, first install the v2.0 debugger (see above), then replace the Power Mac Debugger 2.0 application with the v2.1a1 Pre-release application.

Note that in one-system debugging mode, the v2.1a1 host automatically launches the Power Mac DebugServices nub, so there is no need to launch it separately first. In addition, if the low-level nub is already active, Power Mac DebugServices asks you to

deactivate it using the Debugger Nub Controls panel, so there is no risk of having both nubs active at the same time. Similarly, if you activate the low-level nub, the control panel automatically quits the Power Mac DebugServices application first.

68K MACINTOSH DEBUGGER INSTALLATION INSTRUCTIONS

This note explains how to install the Pre-release version of the 68K Macintosh Debugger. It is located in the ":Past&Future:Pre-release:Macintosh Debugger:68K Macintosh Debugger:" folder.

68K MACINTOSH DEBUGGER 2.0B1

The 68K Macintosh Debugger is designed for debugging 680x0-based Macintosh applications and can act as a one-machine or a two-machine debugger. It can be used to debug both classic 68K applications and CFM-68K applications and shared libraries. The debugger host communicates with the target application through 68K Mac DebugServices, a "high-level" nub that uses either a local connection via direct callbacks, or a remote connection via AppleTalk.

Perform the following steps to install the debugging system:

- Insert the E.T.O. CD-ROM in the CD drive of the host Macintosh and navigate into the ":Past&Future:Pre-release:Macintosh Debugger:" folder.

- Drag the "68K Macintosh Debugger" folder to your hard drive. This folder contains the host debugger application (68K Mac Debugger 2.0b1) and some ancillary files.

- Open the "68K System Folder Additions" folder inside the "68K Macintosh Debugger" folder.

- Select the 68K DebugServicesINIT file in this folder and drop it on top of the System folder icon on the host Macintosh (not into an open System folder). This copies the 68K DebugServicesINIT file into the Extensions folder.

- If the host system is running a version of system software prior to 7.5, select the file called ProcessMgrINIT (inside the "Pre-System 7.5 Only" folder) and drop it on top of the System folder icon (not into an open System folder). This copies the ProcessMgrINIT file into the Extensions folder.

- If you will be doing remote (two-system) debugging, eject the E.T.O. CD-ROM and place it in the CD drive of the target Macintosh and navigate into the ":Past&Future:Pre-release:Macintosh Debugger:68K Macintosh Debugger:" folder. Then:

- Drag the 68K Mac DebugServices nub application to any convenient location on the target hard drive (it doesn't matter where although you can put it in the Startup Items folder in the System Folder so that it is automatically launched at boot time).

- Open the "68K System Folder Additions" folder.

- Select the 68K DebugServicesINIT file in this folder and drop it on top of the System folder icon on the target Macintosh (not into an open System folder). This copies the 68K DebugServicesINIT files into the Extensions folder.

- If the target system is running a version of system software prior to 7.5, select the file called ProcessMgrINIT (inside the "Pre-System 7.5 Only" folder) and drop it on top of the System folder icon (not into an open System folder). This copies the ProcessMgrINIT file into the Extensions folder.

- Restart the target Macintosh system.

- Restart the host Macintosh system, then launch the 68K Mac DebugServices application on the target system (this will be the host system if you are debugging in single-machine mode).

For two-machine debugging, you must use the Sharing Setup control panel on the target system to turn on Program Linking. In addition, you must use the Users & Groups control panel on the target system to specifically grant program linking privileges for the user of the host system. Do this by first creating a New User icon in the Users & Groups control panel and giving it a name which is the same as the Owner Name of the host Macintosh. (This name is set using the Sharing Setup control panel on the host Macintosh.) Then, double-click the user icon just created and select the *Allow user to link to programs on this Macintosh* check box. Alternatively, you can double-click the <Guest> user icon and select the *Allow user to link to programs on this Macintosh* check box to grant program linking privileges to guests.

- Run the debugger host by launching the 68K Mac Debugger 2.0b1 application. When it starts up you will see the following dialog:

Click the *Local* radio button if you want to debug a 68K application running on the same system as the debugger host.

Click the *AppleTalk* radio button if you want to debug a 68K application running on a remote Macintosh.

RUNTIME LIBRARIES INSTALLATION INSTRUCTIONS

This note explains how to install the runtime libraries included in the ":Interfaces&Libraries:RuntimeLibraries:" and in the ":Past&Future:Pre-release:Interfaces&Libraries:RuntimeLibraries:"folders on the E.T.O. CD-ROM. The runtime libraries are divided into three groups:

Required for MPW	The software in this folder must be installed in order for the MPW Shell and its tools to operate.
CFM-68K Additions	This folder includes the CFM-68K runtime and various CFM-68K shared libraries.
Power Macintosh Additions	Shared libraries that may be needed to run some Power Macintosh applications.

REQUIRED FOR MPW

This release of E.T.O. includes an update to the Standard C Library ("StdCLib") for Power Macintosh. This update is provided as a system "INIT" (called StdCLibInit) that at boot time overrides portions of StdCLib that ships in ROM on current Power Macintosh systems. The update is required for the MPW Shell and for native MPW tools. However, this update will not be required when running System software 7.6 or later.

To install the update, drag the file called StdCLibInit from the ":Required for MPW:" folder to the System folder icon (not into an open System folder) on your Power Macintosh system. This copies the file into the Extensions folder. After installation, reboot your system.

See the MPW Release Notes for more information about StdCLibInit and whether your own applications will require it or whether they will work fine with the version of StdCLib in ROM.

CFM-68K ADDITIONS

CFM-68K is a new runtime architecture for 68K-based Macintosh applications and shared libraries and is modeled after the Power Macintosh runtime architecture. CFM-68K runtime applications and shared libraries can coexist and run simultaneously with classic 68K runtime applications.

CFM-68K Runtime Software Installation

The CFM-68K runtime software requires System 7.1 or later, 32-bit addressing to be turned on, and a 68020, 68030, or 68040 Macintosh system.

To install the CFM-68K runtime software and related shared libraries, drag the file called CFM-68K Runtime Enabler, as well as all the shared library files in the folder, from the ":CFM-68K Additions:" folder to the System folder icon (not into an open System folder) on your 68K-based Macintosh system. This copies the files into the Extensions folder. After installation, reboot your system.

Documentation on how to develop software that takes advantage of CFM-68K is described in *Building and Managing Programs in MPW, second edition*, in the ":Documentation:MPW:Bldg & Mng Progs in MPW 2nd ed.:" folder.

POWER MACINTOSH ADDITIONS

This folder contains shared libraries and other extensions that may need to be installed to run applications that use certain parts of the Macintosh toolbox. To install, drag the files to the System folder icon (not into an open System folder). This copies them into the Extensions folder.

SOMobjects for Mac OS Installation Instructions

This note explains how to install the SOMobjects runtime software and the tools required to develop software for it. SOMobjects is an object-oriented programming technology for building, packaging, and manipulating binary class libraries. It is an enabling technology for OpenDoc.

Runtime Software Installation

Before installing, check the Extensions folder on your boot volume—if it contains a file called "SOM", delete it. This file is obsolete and incompatible with the new SOMobjects™ for Mac OS file.

To install the SOMobjects runtime, drag the file called "SOMobjects™ for Mac OS" from the ":Other Development Tools:SOMobjects™ for Mac OS:System Additions:" folder to the Extensions folder inside your System folder. After installation, reboot your system. This is a fat shared library which will work on both a 68K Macintosh and a Power Macintosh.

If you want to use this software on a 68K-based Macintosh system, you must also install the CFM-68K runtime software as described in the Runtime Libraries Installation Instructions.

Development Tools Installation

Several MPW-based tools, scripts, libraries, interfaces, and examples are provided to enable you to develop applications that use SOMobjects. Note that with this release you do not need to install the SOMobjects runtime software to use the SOMobjects development tools—of course, the runtime software must be installed to run and test applications that you create using these tools.

Listed below are the steps you need to follow to install these files on your hard disk. We assume that you have already installed the MPW environment itself.

• Run the MPW Shell from your hard disk.

• Go to the Worksheet window, type in the following commands, select them, then type the Enter key:

```
Directory "E.T.O. #22:Other Development Tools:SOMobjects™ for Mac OS:MPW Additions:"
```

```
Duplicate -y "UserStartup•somc" "{MPW}"
Backup -r -a -c -from ":Examples:" -to "{MPW}Examples"
Backup -r -a -c -from ":Scripts:" -to "{MPW}Scripts"
Backup -r -a -c -from ":Tools:" -to "{MPW}Tools"
Backup -r -a -c -from ":Interfaces:" -to "{CIncludes}:"
Backup -r -a -c -from ":Libraries:" -to "{SharedLibraries}:"
```

• These commands generate a list of Duplicate commands for those SOMobjects-related files that are either missing or out of date in your current MPW system. Execute all the Duplicate commands by selecting them, then typing the Enter key. This moves all the new files to the proper folders on your hard disk.

Documentation on how to develop software that takes advantage of the System Object Model is in the ":Other Development Tools:SOMobjects™ for Mac OS:Documentation:" folder.

SOMobjects is a trademark of IBM Corporation.

MACINTOSH PROGRAMMER'S ASSISTANT INSTALLATION INSTRUCTIONS

Macintosh Programmer's Assistant is an on-line reference tool which provides rapid up-to-date access to information on all aspects of the Macintosh API, including toolbox managers, data structures, routines, constants, resources, and MPW commands. It includes MPW scripts and tools that let you access it directly from the MPW environment.

The Macintosh Programmer's Assistant software and databases are located in the ":Documentation:Online Reference:" folder.

This folder includes the following material:

- The QuickView application for displaying and browsing information databases.

- The Programmer's Assistant databases. Each database generally has a filename suffix of ".qv" and contains reference materials from one or more Inside Macintosh or MPW books. For example, the Macintosh Toolbox database contains the reference material from Inside Macintosh: Macintosh Toolbox Essentials and Inside Macintosh: More Macintosh Toolbox.

  The " !Mac Programmer's Asst.qv" database contains a menu of all the major database categories included on the CD-ROM.

- A "Goodies" folder containing information and tools that make it easy to access Programmer's Assistant pages directly from the MPW Shell editing environment. It also contains information on how to create your own Programmer's Assistant databases.

INSTALLING QUICKVIEW AND PROGRAMMER'S ASSISTANT DATABASES

To install the QuickView application and the Programmer's Assistant databases, simply copy to the same folder on your local hard drive the "QuickView™" application file, the "!Mac Programmer's Asst.qv" database file, and any of the other databases that you want to use. If you do not have enough space for the other databases, copy only the ones you use most frequently and create aliases to the remaining ones. These aliases must be placed in the same folder as the QuickView application on your hard drive.

Launching Programmer's Assistant

Launch the Programmer's Assistant by double-clicking either the "QuickView™" application file or any .qv file (e.g., " !Mac Programmer's Asst.qv") on your local hard drive.

Using Programmer's Assistant

Programmer's Assistant gives you rapid access to up-to-date information about the Macintosh API, standard C library routines, and MPW commands. This includes information about all the managers documented in Inside Macintosh, including their data structures, routines, constants, and resources.

QuickView gives you extensive hypertext links for rapid, easy navigation and provides useful features such as copy text, fast full-text search, user annotations, and resizable windows.

If you have questions on how to use QuickView and the Programmer's Assistant databases, use the Programmer's Assistant Help file. It is available through the Help menu when QuickView is active.

When looking up an MPW command that has the same name as a toolbox routine, you must first go to the master page of MPW commands. You can do this by typing "MPW Commands" in the Keyword box or clicking "MPW Commands Assistant" on the Macintosh Programmer's Assistant menu page or clicking "MPW Commands" at the bottom of the Managers page. If you don't take any of these actions first, you will look up the toolbox routine instead.

ACCESSING PROGRAMMER'S ASSISTANT DIRECTLY FROM MPW

The scripts in the ":Goodies:Toolbox Asst. Tools for MPW:" folder inside the "Online Reference" folder allow you to access Macintosh Programmer's Assistant information from the MPW Shell. Using these scripts, you can look up any term in the Macintosh Programmer's Assistant databases or retrieve the template for a toolbox routine.

Installing MPW Access for Macintosh Programmer's Assistant

The "Programmer's Asst. Tools for MPW" folder contains the following files:

- A "UserStartup•MPTA" MPW script
- A "MPTA_MPWAccess" MPW script

To install these files, take the following steps:

- Quit the MPW Shell if is currently running.
- Drag the "UserStartup•MPTA" file into the "{MPW}Startup Items" folder.
- Drag the "MPTA_MPWAccess" file into the "{MPW}Scripts" folder.

- Start up the MPW Shell. During the startup process, you will see a Standard File dialog box asking you for the location of the QuickView browser. (QuickView is the search and display engine for Macintosh Programmer's Assistant.) Locate QuickView and click the "Here It Is" button. The location will be stored in a file called "MPTA_FullPathName" for use during subsequent MPW startups.

- When the startup script finishes, it will have installed a new menu called "Info". That menu contains three commands, "Look up Routine" (for looking up a toolbox or library routine), "Look up MPW Command" (for looking up an MPW command), and "Get Template" (for returning the template of a toolbox or library routine).

- As a test, type "NewWindow" on the MPW worksheet, highlight just the word "NewWindow", and select the "Look up Routine" item in the Info Menu. MPW will send an Apple Event to the QuickView application, which will open and bring the page for the NewWindow toolbox routine to the front.

- As another test, type "AddMenu" on the MPW worksheet, highlight just the word "AddMenu", and select the "Look up MPW Command" item in the Info Menu. MPW will send an Apple Event to the QuickView application, which will open and bring the page for the AddMenu MPW command to the front.

- To test the template retrieval, type another toolbox call into the worksheet (for example "NewCWindow"), highlight the call name, and select the "Get Template" item on the Info Menu. The template for the function call will be retrieved and will replace the current selection.

- That's it! Subsequent startups of MPW will automatically use the QuickView application that you selected, and the "Look up Routine", "Look up MPW Command", and "Get Template" functions will automatically be available.

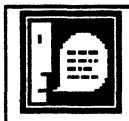
There is one more thing you need to be aware of:

- If you move the QuickView application from its original location, MPW will be unable to find it and you will again be prompted to locate QuickView. After you locate it, everything will work once again.

Developer on-line resources

Developer Tools Web site
(http://www/devtools.apple.com)

A repository of product information on developer tools from Apple and other tools vendors. It provides in-depth technical information.



E.T.O. Web page
(http://www.devtools.apple.com/eto/index.html)

This web site contains essential information on E.T.O. which includes news, updates, surveys, bug fixes and other topics of interest.

MPW-dev mailing list

We've now established an Internet mailing list for MPW users. To subscribe, send a message to **listproc@solutions.apple.com** and put the following command in the body of the message: **SUBSCRIBE mpw-dev yourRealName**. The mpw-dev list is an excellent way to meet other MPW users who can provide advice on using MPW, resolve problems, etc.