Technical Note TN1066

E.T.O: The Right Tools for the Right Job

CONTENTS

The motivation behind this Technote was the frequent inquiries from developers about what Apple development tools are available for their specific programming tasks. This Technote showcases the major components of Apple's E.T.O. (Essentials - Tools - Objects), which now comes on two CD-ROMs.

This Technote serves as a reference for any Macintosh developer who wants to know what Apple development tools and utilities they have at their disposal. It does not, however, cover the multitude of third-party development tools that are also available.

Note:
This Technote was written shortly after E.T.O. 22 shipped. All references are accurate as of E.T.O. 22.

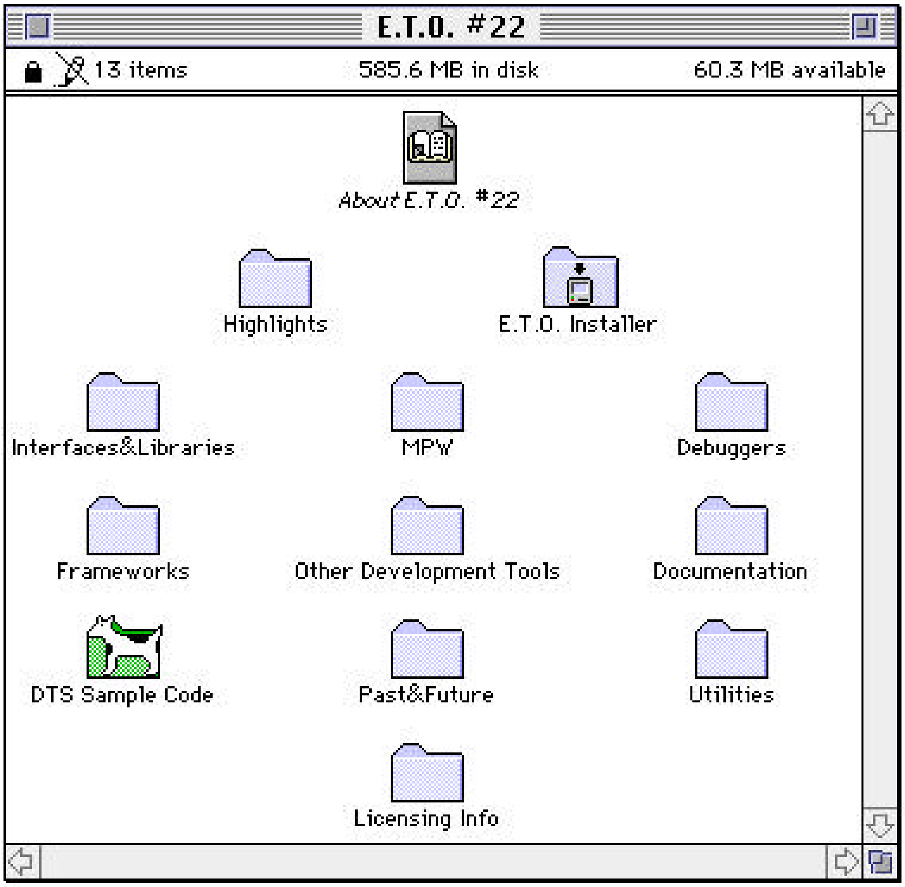
Updated: [Jan 01 1997]

About E.T.O.

E.T.O. is Apple's primary product for delivering core development tools to developers who are creating, debugging, and testing Macintosh applications written in C, C++, or assembly language. It is designed for programmers who are working on applications designed for both 680x0- and PowerPC-based Macintosh systems.

E.T.O., which includes the MPW development environment, the MacApp application framework, and the OpenDoc Development Framework, is sold only by subscription and updates are automatically sent to subscribers every four months during the subscription year. New subscribers receive the current issue followed by the next two scheduled updates.

While E.T.O. provides you with a complete development system, it also includes many components that are not available with either Metrowerks CodeWarrior or Symantec C++, making it a valuable companion to these other popular environments. Users of other development environments will find E.T.O. the convenient source for the Macintosh Programmerpis Assistant, performance analysis tools, Pascal to C conversion tools, and the Virtual User testing tool. CodeWarrior users also get the benefit of a plug-in MrC/MrCpp compiler and the Power Mac Debugger.

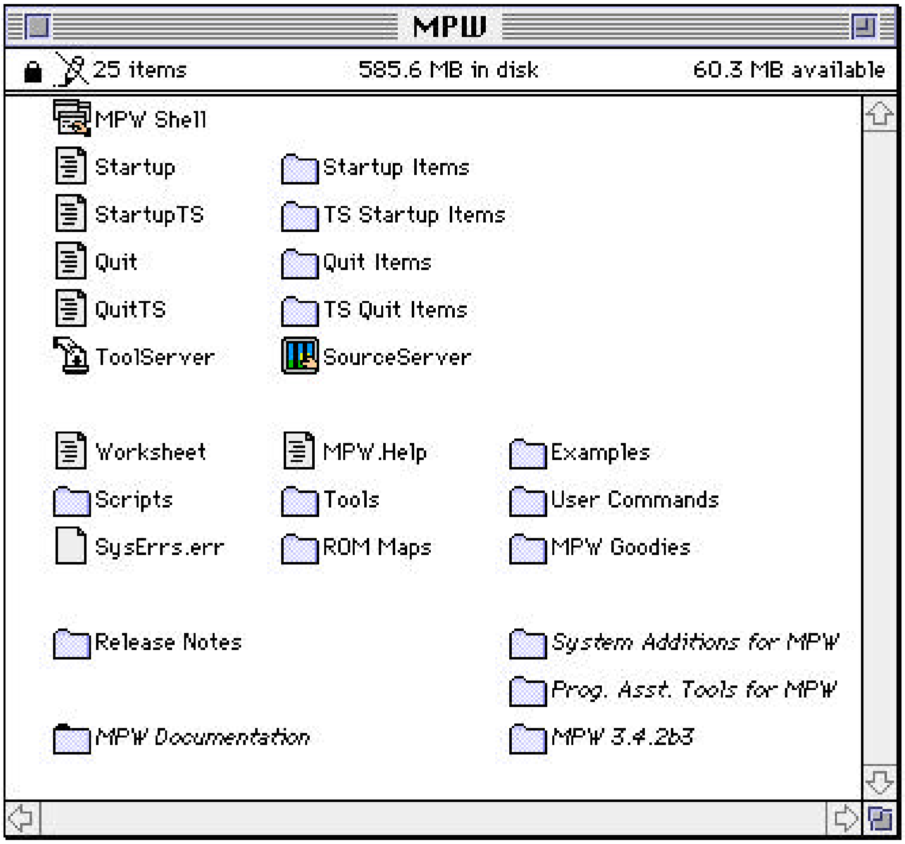


Back to top

Major Components of E.T.O.

MPW Shell

The Shell is the centerpiece of the Macintosh Programmer's Workshop (MPW) development environment. It is an integrated multiwindow text editor and command executer which supports numerous built-in commands and a powerful scripting language that allows you to customize the environment to suit your needs. The Shell also includes Projector, an integrated source code control system that simplifies the management of large, complex software projects being worked on by a team of programmers.



SC/SCpp -- C and C++ Compilers (68K)

SC and SCpp are new ANSI-compliant 680x0-code-generating C and C++ compilers that are faster than previous compilers included with MPW, run in native mode on a Power Macintosh, and support the new CFM-68K runtime model.

MrC/MrCpp -- C and C++ Compilers (PowerPC)

MrC and MrCpp are new MPW compilers which offer greatly improved turnaround time and also generate efficient and highly-optimized PowerPC code for fastest binary execution on Power Macintosh systems. MrC adheres to the ANSI C standard (ANSI X3.159-1991) and MrCpp adheres to the ANSI C++ working paper (ANSI X3J16) except for templates and exception handling.

Assemblers (68K and PowerPC)

Two assemblers are included, one which supports the full 68xxx instruction set and the other which supports the PowerPC instruction set. The 68xxx assembler supports all the instructions and addressing modes for the MC68000, MC68010, MC68020, and MC68030 microprocessors, the MC68851 Paged Memory Management Unit (PMMU), and the MC68881 and MC68882 Floating-Point Coprocessors, in all usable combinations.

MPW Tool and Script Suite

Dozens of tools and scripts are included to assist you in the development process. These include linkers, a Make tool, resource manipulation and analysis tools, file comparison utilities, and many more.

MacApp (68K and PowerPC)

MacApp is an advanced object-oriented application framework and class library which can be used to streamline the development of Power Macintosh and 680x0 applications through the benefits of object-oriented programming. For more detailed information, please refer to Apple's MacApp Home Page.

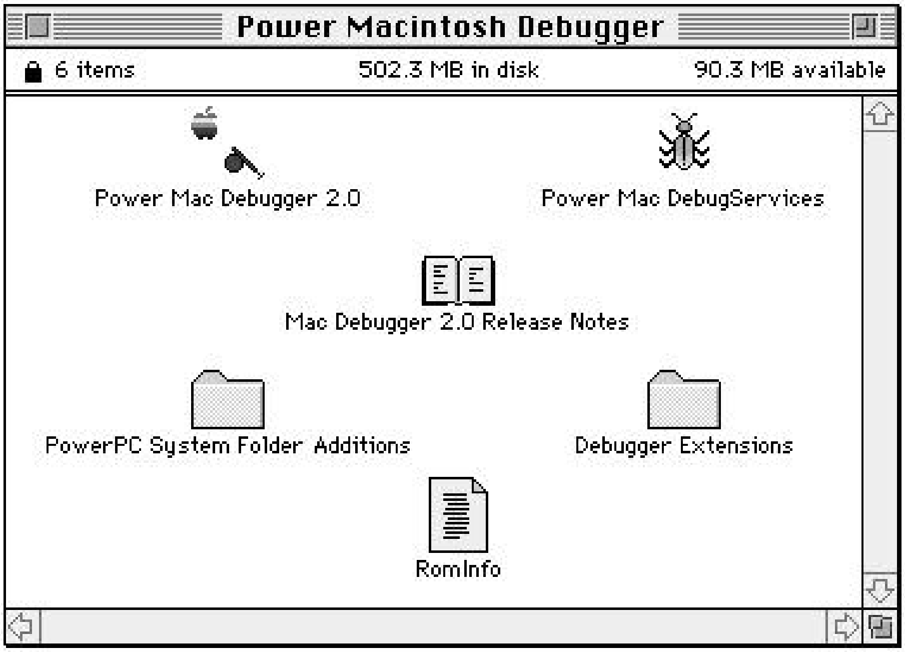
68K Macintosh Debugger

The 68K Macintosh Debugger runs on 68k and PowerPC machines and 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.

Power Macintosh Debugger

Power Macintosh Debugger 2.0 is designed for debugging native Power Macintosh applications 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 Power Mac Debugger also includes ASP (Adaptive Sampling Profiler). ASP is a sampling utility that allows you to measure how often sections of code execute. The ASP is an integral part of the debugger; you enable the ASP by a selection from the debugger's Performance menu. Unlike other samplers, the ASP does not require you to modify your source code in order to take measurements. This means that you do not have to recompile your source to measure its performance. In addition, you can obtain a more reliable picture of your application's performance because you do not have to worry about paging behavior or memory use that is due to inserted sampling code.



Power Macintosh Debugger

6 items 502.3 MB in disk 90.3 MB available

Power Mac Debugger 2.0

Power Mac DebugServices

Mac Debugger 2.0 Release Notes

PowerPC System Folder Additions

Debugger Extensions

RomInfo

MacsBug (68K and PowerPC)

MacsBug is a low-level debugger for debugging software at the assembly-language level (680x0 or PowerPC). It loads at system startup time and idles until it is invoked either manually (by pressing the interrupt switch or a key that you define) or programmatically (by calling special traps from within your program). And since MacsBug needs so little of the system to operate, it can be used even in the case of fatal system errors.

ResEdit

ResEdit is a graphical resource editor for creating and editing elements of the Macintosh user interface, such as menus, windows, icons, dialog boxes, and other standard data structures. Programmers and advanced users can quickly create portions of their application's user interface as well as modify an existing interface.

Macintosh Programmer's Assistant

This is a useful new 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. MPW scripts and tools are also provided that let you access it directly from the MPW environment using "hot keys".

Virtual User

Virtual User is a test automation system that enables a host Macintosh to control the actions of target Macintosh computers remotely. The host executes tests on one or more target machines by reading the commands it is to issue from a file called a script. The script is a text file written in the Virtual User scripting language. This interpreted procedural language provides Virtual User with the ability to recognize and manipulate various elements of an application's human interface, compare the actual result of a command with an expected result, and branch logically according to the result of the comparison.

MPW p2c

p2c is a source-to-source translation tool which translates between Object Pascal and C++. (p2c has an option to generate standard C code instead of C++.)

Back to top

On the Horizon: Pre-Release Tools

ResEdit 3.0

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 is being developed to replace previous versions of ResEdit. Here are some of the features of ResEdit 3.0:

- Standard Macintosh application look-and-feel
- Extensible architecture with plug-ins and enhanced templates
- WorldScript support
- Macintosh Drag and Drop support

MrPlus/MrProf

MrPlus is an MPW performance tool which operates on Apple PEF files. It is a multi-function tool which can do the following:

- Produce static information about the executable file, such as counts of important program structures, frequency of appearance of the various instructions, and frequency of appearance of references to the various registers.
- Produce a PEF file which is an enhanced version of the original file. It contains instrumentation code which, at the time of execution, outputs dynamic information about the program. Corresponding to the static information already listed will be dynamic data on the instruction mix. The distinction is that the dynamic information states the frequency of execution instead of the frequency of appearance. The dynamic information also includes a routine profile: the relative number of instructions executed in each routine.
- Produce a PEF file containing instrumentation that traps attempts to store into low memory (0-32K).
- Optimize the program by reordering portions of code in the PEF file to improve the utilization of the instruction cache and to reduce the number of page faults in a system with virtual memory enabled.

MrProf is an application that provides tabular and graphic presentations of data collected when executing programs that were instrumented by MrPlus. The input data to MrProf are a .pmap file, optionally a .xcoff file, and one or more .pcnt files.

PPCProff and Proff

PPCProff (for PowerPC programs) and Proff (for classic 68K programs) provide profiling and performance monitoring for programs compiled from C and C++. For every routine called during program execution, the Proff tools record the identity of the called routine, the identity of the calling routine, and the time spent in each routine.

PPCProff prints an analysis of the Profiler.pgh data file generated when you run a PowerPC application or shared library that was linked using PPCLink's -profile option.

Direct SOMobjects for Mac OS

A folder which contains information on how to use the "Direct-to-SOM" capability of the new prerelease 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.

MrC/MrCpp - Plugin version for CodeWarrior

This is a new prerelease version of the MrC/MrCpp compiler for use with the Metrowerks CodeWarrior environment. Note that it can be used only with CodeWarrior 9 or later.

Instrumentation SDK 1.0

This is the first release of the Instrumentation Viewer and 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.

The instrumentation system is designed to help you record information about running software. This allows you to expose its behavior and drive performance investigations.

Mac OS Runtime for Java 1.0b1

Mac OS Runtime for Java (MRJ) is Apple's Java execution environment based on technology from Sun Microsystems, Inc. The MRJ is designed to provide a robust, stable, reliable implementation of the Sun JDK 1.0.2 specification.

This beta preview will allow developers to build Macintosh applications which incorporate Java. Using MRJ, developers can build standalone Java applications as well as run Java within existing applications.

For more information about Java at Apple, and to download periodic updates, see Apple's MRJ Home Page.

Back to top

Summary

E.T.O. brings together a deluxe collection of Macintosh core development tools, designed to support C, C++, and assembly-language programmers. It contains a robust development environment (Macintosh Programmer's Workshop), compilers, debuggers, application frameworks, and testing tools that you can use to streamline application development for 68K and Power Macintosh systems.

Back to top

References

If you have questions or comments regarding E.T.O. or any of its components, you can use the Feedback page. More detailed information is also available on Apple's Developer Tools Home Page.

There is also an E.T.O. discussion mailing list. For information on how to subscribe to it, please refer to Apple's MPW-Dev Mailing List Information Web Page.

For general Apple developer information, there is also a new Apple Developer Web Page.

Back to top

Change History

01-September-1996	Originally written
01-January-1997	Updated for E.T.O. 22 Added new pre-release items: Instrumentation SDK, Mac OS Runtime for Java

Back to top

Downloadables

Acrobat version of this Note (K). Download

Back to top

Technical Notes by API | Date | Number | Technology | Title
Developer Documentation | Technical Q&As | Development Kits | Sample Code