WildPackets

EtherPeek™
EtherPeek NX™

User Manual

Information in this manual is subject to change without notice. EtherPeek™ and EtherPeek NX are furnished under a software license agreement. The software may only be used or copied in accordance with the terms of the agreement. It is against the law to copy the software, except as specifically allowed in the license. No part of this manual may be reproduced or transmitted in any form, or by any means, electronic or mechanical, including photocopying, for any purpose, without the express written permission of WildPackets, Inc.

Copyright ©1989-2003, WildPackets, Inc. All rights reserved.

EtherPeek 5.1 for Windows

EtherPeek NX 2.1 for Windows

WildPackets, Inc.
1340 Treat Blvd., Suite 500
Walnut Creek, CA 94597
USA

(925) 937-3200

www.wildpackets.com

AiroPeek, AiroPeek NX, EtherPeek, EtherPeek NX, GigaPeek NX, iNetTools, NAX, NetDoppler, NetSense, Network Calculator, PacketGrabber, PacketScrubber, ProConvert, ProtoSpecs, RFGrabber, RMONGrabber, WildPackets and WildPackets Academy are trademarks of WildPackets, Inc. All other trademarks are the property of their respective holders.

EtherPeek Contents

Introduction

Welcome to EtherPeek and EtherPeek NX, the award-winning network traffic and protocol analyzers from WildPackets.

EtherPeek helps network administrators meet the most demanding network troubleshooting and monitoring challenges. Designed for IT professionals at all levels of experience, EtherPeek's easy-to-use interface lets even novice users get up to speed quickly and efficiently. From troubleshooting a local network to maintaining distributed networks in the enterprise environment, EtherPeek is an indispensable tool.

EtherPeek NX brings the power of Expert Analysis to the full array of network and packet statistics available in EtherPeek standard. The Expert Analysis features track over 90 separate aspects of network performance, monitoring delay and throughput, showing volume and protocol information for each pair of nodes, and checking for anomalies in all layers of your network.

Sophisticated filters are easy to create and use, allowing you to narrow the search to suspect traffic quickly and identify the source of network trouble.

Analysis modules provide more detailed data about network traffic, letting you quickly compare snapshots to reveal changes in the details of traffic patterns and performance over time.

WildPackets' ProtoSpecs™ technology can accurately decode packets carrying data from any one of thousands of protocols and network applications, including all of the most common protocols.

In this Chapter:

EtherPeek, standard and NX

This manual serves for two WildPackets products, EtherPeek 5.1 (“EtherPeek standard”) and EtherPeek NX 2.1 (“EtherPeek NX”). The two share many features. When the text refers to “EtherPeek” without further qualification, it applies equally to either product. The sections below describe the primary differences between the two products.

EtherPeek standard

EtherPeek standard offers all the features of a great network analysis tool at an affordable price. In addition, the ***Conversations*** view in Capture windows and Packet File windows is unique to EtherPeek standard. For more about the ***Conversations*** view, please see “Conversations” on page 168.

EtherPeek standard does not offer the Expert analysis functions, and does not show the ***Expert*** or ***Peer Map*** views found in EtherPeek NX.

EtherPeek NX

EtherPeek NX has all the features of EtherPeek standard plus an advanced set of expert troubleshooting and diagnostic capabilities, expert problem detection heuristics, and a graphical view of communicating peer nodes. The following features are unique to EtherPeek NX:

- The ***Expert*** view in Capture windows and Packet File windows provides Expert Analysis of 91 aspects of network performance in real time.
- You can fine tune the parameters for any Expert diagnostic item and get instant help with event *Description*, *Possible Causes*, and *Possible Remedies* in the **Expert EventFinder Settings** window.
- You can save and reload customized Expert diagnostic settings from the **Expert EventFinder Settings** window.
- The ***Peer Map*** view of Capture windows and Packet File windows creates a continuously updated graphical view of traffic between pairs of network nodes, showing volume, protocol, node address, node type and more. Full customization lets you identify problems and anomalies quickly and intuitively.

For more about the Expert analysis functions, see Chapter 5, “Expert View and Expert EventFinder” on page 101. For more about the Peer Map, see Chapter 6, “Peer Map” on page 119.

The ***Conversations*** view, found in EtherPeek standard, is not present in EtherPeek NX. The ***Expert*** view provides all of the same functionality plus the expert features described above.

Differences in user interface

Where their features are different, EtherPeek standard and EtherPeek NX will show different items in their user interface. In particular, Capture windows (Figure 1.1) and Packet File windows will show different views and view tabs depending on the version of the program.

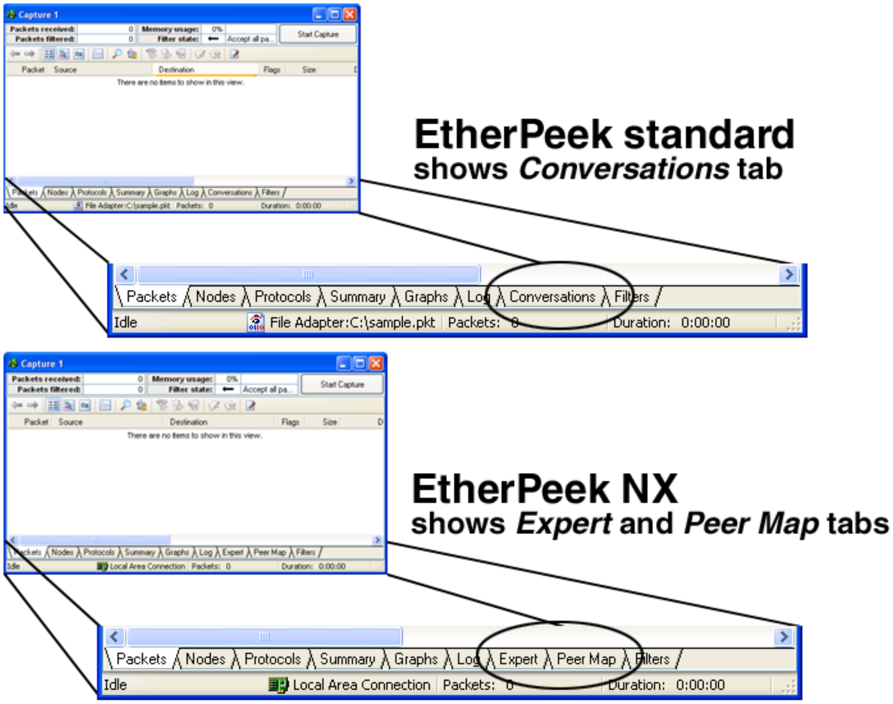


Figure 1.1 Detail of Capture window view tabs in EtherPeek standard and EtherPeek NX

EtherPeek standard includes the ***Conversations*** view and all its associated features, but does not include the ***Expert*** or the ***Peer Map*** views, nor any of the features associated with them. EtherPeek NX does not have a ***Conversations*** view, but has the expert analysis features associated with the ***Expert*** and ***Peer Map*** views.

Except when describing the ***Conversations*** view, the screenshots in this manual are taken from the EtherPeek NX version of the program. Most screenshots would appear identical whether taken from one version or the other. With the exception of the differences in view tabs (described above), where a screenshot is particular to one program version, the figure title shows this as “EtherPeek standard” or “EtherPeek NX.”

New features

A complete list of new features is available in the Readme file. The following section highlights some of the most important new features. Except as noted below, these features are new to both EtherPeek standard and EtherPeek NX.

Performance view

EtherPeek provides more flexibility than ever, allowing you to optimize both the Monitor statistics and packet capture functions for maximum performance in a particular environment. The new ***Performance*** view of the **Monitor Options** and **Capture Options** dialogs lets you selectively enable or disable individual program functions, either for Monitor statistics or for an individual Capture window. Now, for example, you can create capture templates that perform only the functions required for the task at hand. For more information, please see “Performance views” on page 25.

Enhanced statistics output options

Statistics output options have been changed to allow both the periodic output of statistics reports, and also the periodic creation of sets of reports on a variety of user-defined schedules. The **New File Sets Schedule** dialog allows you to create sets of reports covering an extended group of statistics output intervals of your choice. For more information, please see “Statistics output views” on page 173.

Repeat mode for triggers

Triggers can now be automatically reset to support continuous monitoring. In repeat mode, the Capture window will reset the start trigger each time the stop trigger is tripped. Repeat mode allows you to capture multiple occurrences of the same event with a single Capture window. For more information, see “About repeat mode” on page 230.

Copy selected packets to new window

In the ***Packets*** view of a Capture window or Packet File window, you can now choose **Copy Selected Packets to New Window** from the context menu to create a temporary Packet File window containing only the selected packets. You can repeat the process from within the new window, creating a cascade of Selection windows. The packets are renumbered, but the original packet order is retained. For more information, see "Copy selected packets to new window" on page 286.

Continuous expert and conversations analysis

The use of memory in Capture windows by the ***Expert*** view (in EtherPeek NX) and the ***Converstaions*** view (in EtherPeek standard) has been changed to improve efficiency and enhance performance in continuous monitoring. Expert and conversations analysis can now be used continuously in a Capture window, always presenting the most recent findings, and (for the Expert function only) logging the results to the Event Log. The Expert and conversations analysis functions in Packet File windows remain unchanged, and will always allocate whatever memory is required to process all the packets in a saved packet file. For more information, see "Expert memory allocation" on page 116.

New expert events

Expert events (formerly "problems") have always been ordered by OSI layer, but in EtherPeek NX they can now be selectively enabled and disabled by layer in a single click. EtherPeek NX also adds a few new expert events: EAP Authentication failure, HTTP Client Error, HTTP Request Not Found (404 error), HTTP Server Error, DNS Non-Existent Domain, and DNS Server Error. These new Expert events support a finer degree of user control over the Expert function.

New analysis modules

EtherPeek adds the NCP (Netware Control Protocol) and SCTP (Stream Control Transmission Protocol) Analysis Modules, providing additional information about packets of these types to the ***Summary*** column of the ***Packets*** view. For more information, see "NCP analysis module" on page 269 and see "SCTP analysis module" on page 270.

New and improved packet decoders

EtherPeek adds new and improved packet decoders, including: MMS/ICCP, RGMP, SNS, CGMP, ISDN Q.921, SCTP, VoIP (Q.850, H.245), Kerberos 5 gss-api, and more.

And more

These and all the features of EtherPeek standard and EtherPeek NX are covered in this manual. For a brief overview of some key program functions, please see the Quick Tour, available from the **Start Page**, or by choosing **Quick Tour** from the **Help** menu in the program main window.

Before beginning to use the software, please see Chapter 2, “Installing and Configuring” on page 9, to insure the program is installed in the proper system environment for maximum operability. Please refer to the Readme.htm file for other important information about program installation and use. Please visit the support pages on our website at http://www.wildpackets.com/support for the most current list of supported Ethernet adapters.

WildPackets Academy

WildPackets Academy offers comprehensive network analysis training centered on practical applications of protocol analysis techniques using EtherPeek NX and EtherPeek. Courses include the following topics:

- Foundations of Network Protocol Analysis
- Network Troubleshooting Methods
- Emerging Ethernet Technologies: VoIP, Full Duplex, Gigabit, and Switching
- TCP/IP Protocol Analysis Methods

Please see Appendix D, “Resources” on page A-21 for details on WildPackets Academy’s educational resources and a list of network analysis courses. For a complete course catalog, and information about web-delivered training and class schedules, please see the WildPackets Academy web site at: http://www.wildpackets.com/services.

Conventions used in this manual

This manual uses different typefaces to highlight elements that appear in the program user interface, and to distinguish these words and phrases from the rest of the text. In addition, keyboard shortcuts and function keys used to access program functions are set apart using different typefaces.

Table 1.1 **Typographical conventions**

Example	Uses
Edit Name dialog	The titles of dialogs and windows are shown in bold Helvetica.
File menu	Menu items are shown in bold Helvetica.
View > **Color** > **Destination**	A sequence of menu choices is sometimes shown using right angle brackets or “greater than” signs. The example at left means “From the **View** menu, choose **Color**, and within that, choose **Destination**.”
Type **Ctrl + M**, or click **F4**	Keyboard shortcuts and function keys are shown in bold Helvetica. The plus sign in keyboard shortcuts means you must hold down the Control Key (**Ctrl**) while typing the letter indicated. Keyboard shortcuts are not case sensitive. Those few keyboard shortcuts which require the **Shift** key show that fact explicitly in the notation.
Start button	Labels on buttons are shown in bold Helvetica.
Packets view	The names of individual views of windows or dialogs are shown in bold oblique Helvetica. Views provide different sets of information within a single window or dialog. They are usually accessed by clicking on a tab labeled with the name of the view.
Absolute Time column	Column headings are shown in bold oblique Helvetica.
Link speed	All other text appearing in windows and dialogs, including any text to be entered by the user, is shown in oblique Helvetica.

Installing and Configuring

EtherPeek is easy to install and configure for a variety of operating environments. If you have EtherPeek installed on a laptop, you can easily and quickly reconfigure the program to match new conditions as you move from one network segment to another.

This chapter describes the system requirements for EtherPeek, explains how to install the program, and lists the components EtherPeek installs on your computer. It explains how to set up and run the program for the first time, and how to use system memory and application configuration files to keep EtherPeek operating smoothly in all supported environments. The last section introduces Ethernet and explains the application of EtherPeek in different Ethernet network topologies.

Note: This manual does not describe how to install Ethernet network hardware and systems to create an Ethernet network. If you do not already have a functional Ethernet network, please see the documentation that accompanies your Ethernet hardware and computer.

Important! Under certain network or program configurations, EtherPeek can enable the user to monitor information that might be considered confidential. For example, some passwords may be viewable from EtherPeek. Because of this, you may want to prevent unauthorized access to the program. Consider limiting access to EtherPeek by not installing it on public machines and servers.

In this Chapter:

System requirements

This section describes the recommended system requirements for running EtherPeek. Please read this section before you launch the software.

The recommended (and minimum) system configuration for EtherPeek is:

- 1.7 GHz processor or better (minimum 600 MHz processor)
- 512 MB RAM or better (minimum 256 MB RAM)
- Windows 2000 (SP 3 or later) or Windows XP (SP 1 or later)
- Internet Explorer 5.5 or higher is required

Capture requires a supported Ethernet interface (see below).

Supported operating systems require users to have “Administrator” level privileges in order to load and unload device drivers, or to select a network adapter for the program’s use in capturing packets.

Ethernet interface requirements

EtherPeek will work with most NDIS 4 and NDIS 5 compatible interfaces that support promiscuous calls. EtherPeek uses its own software to bind to Ethernet cards and does not depend on vendor-supplied software drivers.

Error packet capture driver

Two types of information, error packets and FCS (Frame Check Sequence) bytes, are dependent on the native capabilities of the card, or of the card in combination with a card-specific WildPackets error capture driver. (FCS bytes are used to detect certain types of error packets.)

Cards operating under a WildPackets error capture driver pass both error packets and FCS bytes to EtherPeek. Error capture drivers are available for Xircom cardbus cards and cards based on the Digital (Intel) chipsets. The driver for Xircom cardbus cards is located in the Driver\Xircom subdirectory, under the directory in which you installed EtherPeek. The driver for cards using the Digital (Intel) chipsets is located in the Driver\Dc21x4 subdirectory. Please refer to the Readme file located in the appropriate directory for more details and driver installation instructions.

Tip The *Error Capture* property in all ***Adapter*** views shows whether or not the selected adapter supports this feature.

Cards operating under standard NDIS 3 (and higher) drivers (that is, those without a WildPackets driver) discard the FCS bytes and the error packets, and do not pass them to higher layers. The NDIS drivers do, however, permit error statistics to be passed, and many card manufacturers collect and pass some form of error statistics (typically CRC and Frame Alignment error counts) to higher layers. EtherPeek uses these statistics to update error counts in Monitor statistics when one of these adapters is selected as the Monitor Adapter. Statistics in Capture windows and Packet File windows are based solely on the contents of their buffers, so only error packets in the buffer will be counted in these windows. The **Packet Decode** window shows FCS bytes as *Calculated* when these bytes were not captured directly from the network.

For the most recent information on drivers, including their support for error capture, please see the Readme file located in the main program directory. You can also find the latest information about supported drivers for EtherPeek in the Supported Hardware section of our website, at http://www.wildpackets.com/support.

Memory

You should install EtherPeek on a workstation or laptop with 1 GB of RAM or more for best performance. The number of packets that can be kept in the capture buffer is limited by the amount of available RAM, so the more memory available to the program, the larger the number of packets that can be analyzed simultaneously in real time, or the larger the packet trace that can be loaded into the program's memory for post-capture analysis. You control the size and use of the capture buffer in the ***General*** view of the **Capture Options** dialog, accessible through the **Capture** menu.

Installing EtherPeek

This section describes how to install the EtherPeek software on your computer.

To avoid possible incompatibilities, it is recommended that you uninstall any earlier versions of EtherPeek before installing the latest version on your system. If EtherPeek detects an earlier version, the installer will offer you the option to uninstall it before installing the newer version. EtherPeek features a simplified setup procedure that automatically installs all of the program's components in their designated locations.

When you launch the Installer, the first window you will see is the Welcome screen, which tells you that EtherPeek is about to be installed on your machine.

The next screen contains the WildPackets Software License Agreement. Please read it carefully so that you understand our terms and conditions concerning possession and use of EtherPeek. You must accept the terms of the license agreement to continue the installation.

The next screen presents the Installation Notes from the Readme file.

The next screen in the setup program is the **User Information** dialog that requires you to enter a name, company name, and your serial number before the program can be installed and launched.

Next, the **Choose Destination Location** dialog suggests the default location in which to install EtherPeek. Use the **Browse** button to display the **Choose Folder** dialog, in which you can navigate to an alternate installation location.

If you have iNetTools installed, the **iNetTools Integration** dialog allows you to automatically set up EtherPeek's **Tools** menu to incorporate this IP test utility suite. For more on iNetTools, see "iNetTools" on page 41. Similarly, you will be offered the option to integrate other WildPackets tools, such as NetSense or ProConvert, if these products are already installed on your system. You can also add these and other software tools to the **Tools** menu after installation. Please see "Customizing the tools menu" on page 40 for details.

When the Installer has finished installing the program files, the final **Setup Complete** screen is displayed. From this dialog, you may choose to view the Readme file or launch the program when installation is complete.

EtherPeek components

The sections under the following headings describe each of the installed components. The location of these files and directories is described relative to the location in which you installed EtherPeek. The most typical location under Windows 2000 or XP would be C:\Program Files\WildPackets\EtherPeek, for EtherPeek standard, or C:\Program Files\WildPackets\EtherPeek NX for EtherPeek NX.

Alarms

The 1033\Alarms directory contains two sets of predefined alarms which you can load into the **Alarms** window using the **Import** button. You can also modify any of the alarms in either of these files. The two files are called Default Alarms.alm and Additional

Alarms.alm. For more information about creating, editing, and using Alarms, please see “Alarms” on page 231.

Utilities

Three command line utilities are included with EtherPeek and installed in the Bin directory where EtherPeek is installed. PeekCat concatenates packet files. TCPDumpFix addresses changes in RedHat Linux TCPDump file formats. VLANStrip removes 802.1Q (VLAN) tags from EtherPeek packet files (*.pkt). See the respective Readme files in the Bin directory for details and usage.

Packet decoders

The modules that decode packets are installed in the Decodes directory in the same location as EtherPeek. These modules provide EtherPeek with the instructions it needs to display packet contents, based on the types of protocols used.

EtherPeek currently provides decoders for over 1,000 protocols and sub-protocols, including IP, IPv6, AppleTalk, DECnet, Netware IPX/SPX, SNA, NetBEUI, and more. For a list of higher-level protocols decoded, please check the support pages at: http://www.wildpackets.com/support.

In addition to the protocol decoders provided as standard issue with EtherPeek, a decoder SDK (Software Development Kit) is available for customizing or adding to EtherPeek’s decoding capabilities. If you are interested in writing your own Packet Decoders, a document and source samples are provided in the 1033\Documents\Peek SDK directory,

Note: Some programming knowledge is required to create packet decoders using the SDK.

Documents

EtherPeek ships with a number of developer tools which are installed in the 1033\Documents directory. Software developers can use these tools to extend or customize ProtoSpecs, Decoders, and Analysis Modules for unique environments. For example, you can customize EtherPeek to recognize a proprietary protocol while it is still in development. Please see the Readme file in the Documents directory for details. The Documents directory also contains PDF versions of the EtherPeek manual and Quick Tour.

Drivers

The Driver directory contains the EtherPeek drivers for supported adapters and operating systems. Drivers are placed in subdirectories, each of which has its own Readme file detailing usage and installation instructions.

Filters

The 1033\Filters directory contains a file called Default.flt which is the default selection of filters for use with the program. You can create, modify or delete individual filters, and save and reload various assortments of filters in named *.flt files for use in different packet capture scenarios. The Filters directory also contains additional sets of filters which you may find useful.

Graphs

The 1033\Graphs directory contains the default set of graphs for the ***Graphs*** view of Capture windows and Packet File windows in a file called Default Graph.gph.

HTML

The 1033\HTML directory contains the Start Page and, in the QuickTour subdirectory, the Quick Tour.

Names

The 1033\Names directory contains configuration files for Name Table entries you might want to install. The Default.nam file provides a starting configuration for the Name Table, and includes a current list of the Vendor ID portion of MAC addresses. This allows you to substitute the name of the card manufacturer for the first three bytes of any Ethernet physical address.

Analysis modules

The Plugins directory contains files of Analysis Modules that enhance the program’s analyzing capabilities. For a complete description of the Analysis Modules currently available with the program and their use, please see “Analysis modules shipped with EtherPeek” on page 252.

The 1033\PluginRes directory contains resource files for use by the Analysis Modules.

Reports

The 1033\Reports directory contains XML, XSL and HTML templates along with related support files for use with the **Save Report** functions and with options available in the ***Statistics Output*** views of the **Capture Options** and **Monitor Options** dialogs. A Readme file in this directory contains detailed instructions for using and customizing output from statistics functions. Please see “Output from statistics” on page 172 for more details.

Samples

The Samples directory contains a variety of sample packet files in EtherPeek format and an associated name table file. You can use these files for testing, training, and to familiarize yourself with program function. Please see the Readme file in that directory for more details.

Application data

Application data (such as names, filters, log files, and more) is cached in the Application Data folder. The default location of the Application Data folder is different for different operating systems. Under Windows 2000 or Windows XP, the default location is in a directory in the root drive where the operating system is installed (typically C:\) with the path name: Documents and Settings\(user name)\Application Data.

EtherPeek creates a subdirectory structure within these locations to cache application data. That subdirectory structure is: WildPackets\EtherPeek (for EtherPeek standard) or WildPackets\EtherPeek NX (for EtherPeek NX). For example, the EtherPeek NX application data for the Administrator of a Windows XP system would be cached in: C:\Documents and Settings\Administrator\Application Data\WildPackets\EtherPeek NX.

Setup and configuration

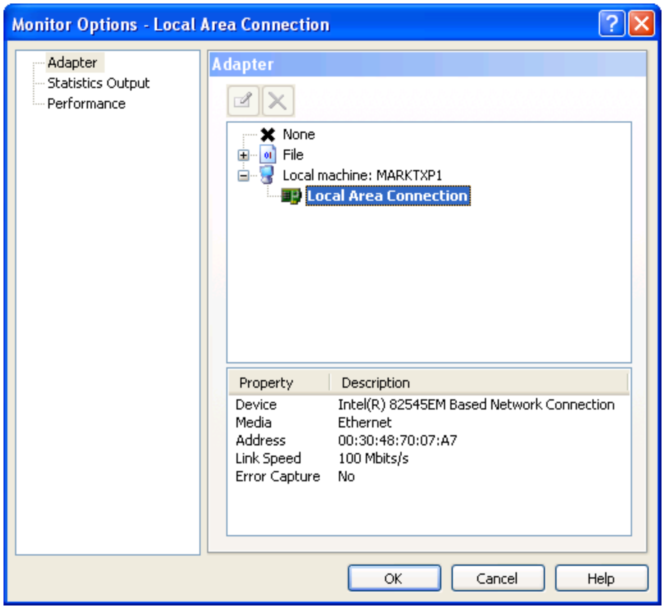
This section explains how to set up EtherPeek for the first time, how to set options for the workspace, list views, fonts, and warnings, how to optimize performance in heavy traffic environments, and how to alter the application’s use of memory.

Selecting an adapter for monitor statistics

When you launch the program, you will be asked to select an adapter to use in collecting Monitor statistics. By default, the program presents the ***Adapter*** view of the **Monitor Options** dialog (Figure 2.1) on program start up.

Tip You can customize this program start up behavior in the ***Workspace*** view of the **Options** dialog, available by choosing **Options…** from the **Tools** menu. Please see “Workspace view” on page 18 for details.

To choose an adapter as the source for Monitor statistics, select one of the choices in the upper pane of the ***Adapter*** view and click **OK**.



Property	Description
Device	Intel(R) 82545EM Based Network Connection
Media	Ethernet
Address	00:30:48:70:07:A7
Link Speed	100 Mbits/s
Error Capture	No

Figure 2.1 Adapter view of the Monitor Options dialog

The ***Adapter*** view of the **Monitor Options** dialog lists all available adapters, arranged hierarchically by type. Each installed NIC, for example, is listed as a separate *Local Area Connection* under the *Local Machine*.

The ***Adapter*** view also shows alternative adapter choices. You can, for example, start EtherPeek without binding Monitor statistics to any adapter by choosing *None* in the

Adapter view. You can choose to simulate network traffic by choosing a *File* as the adapter. If one or more remote (RMON) probes are network accessible, you can choose one of them as the monitor adapter by selecting it under *Module: RMONGrabber.* (For more about RMONGrabber, please see Chapter 14, “RMONGrabber” on page 273.)

When you select an adapter in the upper pane of the ***Adapter*** view, information about that adapter is presented in a table in the lower pane Depending on the type of adapter selected, the lower pane will show the *Device* type, its *Media* type, *Address*, *Link speed*, and whether or not the adapter supports *Error Capture*.

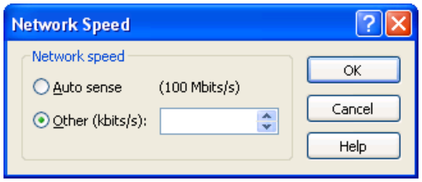
To choose a file as the adapter, expand the *File* item and select a previously used file or choose *New File Adapter.* Double-click on the item, or highlight it and click the **OK** button to make your choice. If you select *New File Adapter*, you will be asked to specify the file, using a standard file **Open** dialog. When you choose an EtherPeek packet file (one of those in the Samples directory, for example), the program cycles through the traffic captured in that file, treating it as live traffic for purposes of calculating Monitor statistics. By choosing a file as the adapter, you can simulate network conditions for training without being connected to a network, or indeed without even having a supported NIC installed on your computer.

EtherPeek remembers recently used file adapters and presents them in the ***Adapter*** view. To remove a file from the list, highlight the file and click the **Delete** button or right-click on the file and choose **Delete** from the context menu.

Tip To return to the ***Adapter*** view of the **Monitor Options** dialog, choose **Monitor Options...** from the **Monitor** menu, and click the *Adapter* item in the navigation pane to open the ***Adapter*** view, or double-click on the current Monitor statistics adapter, shown in the status bar at the bottom right of the main program window.

Network speed options

By default, EtherPeek auto-senses the network speed of the network adapter you select for its use. You may want to override this automatic behavior and set the network speed by hand in certain cases. Some statistics are derived from calculations based in part on the network speed. You may wish to set a nominal network speed for a particular adapter within EtherPeek to insure consistent statistics reporting.



Network Speed

Network speed

○ Auto sense (100 Mbits/s)

● Other (kbits/s):

Figure 2.2 Network Speed dialog

To override the automatic behavior and manually set the network speed EtherPeek should use in performing calculations based on a particular adapter, open the ***Adapter*** view in either the **Monitor Options** or the **Capture Options** dialog. Right-click on the adapter whose speed you wish to set, and choose **Network Speed…** from the context menu to open the **Network Speed** dialog (Figure 2.2) for that adapter. Click the radio button beside *Other (kbits/s)* and enter the speed in kilobits per second. Click **OK** to close the **Network Speed** dialog, and click **OK** again to close the parent dialog, accepting your changes. The same network speed is assigned for all uses of a particular adapter, whether it is selected for use by Monitor statistics, Capture window(s), or both.

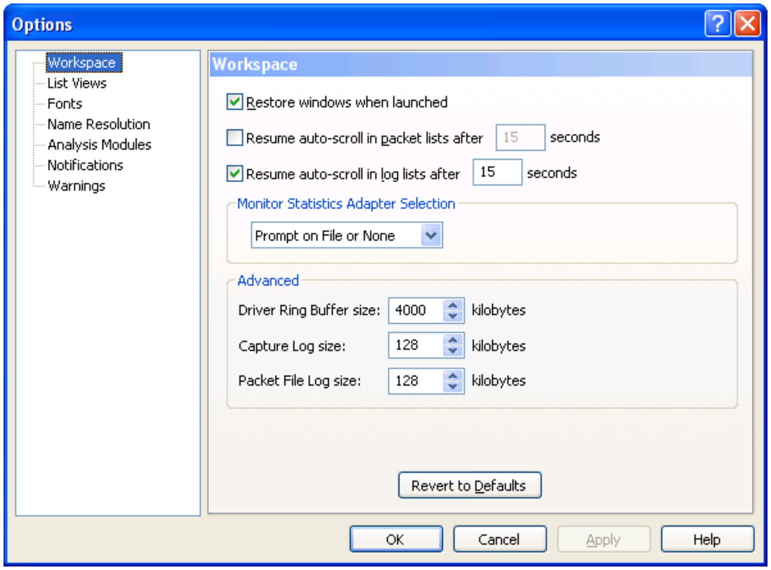
Options dialog

A number of options that apply to EtherPeek as a whole are set in the **Options** dialog. Choose **Options...** under the **Tools** menu to open the **Options** dialog. Click on the view names in the navigation pane to switch between views.

This dialog has seven views, only four of which (***Workspace***, ***List Views***, ***Fonts***, and ***Warnings***) are described in detail in this section. Other views are described in detail in their respective sections of this manual. For the ***Name Resolution*** view, see “Name resolution view of the options dialog” on page 134. For the ***Analysis Modules*** view, see “Enabling and configuring analysis modules” on page 248. For the ***Notifications*** view, see “Notifications” on page 237.

Workspace view

Choose **Options...** under the **Tools** menu and click the *Workspace* item in the navigation pane to open the ***Workspace*** view of the **Options** dialog (Figure 2.3).



Workspace

☑ Restore windows when launched

□ Resume auto-scroll in packet lists after 15 seconds

☑ Resume auto-scroll in log lists after 15 seconds

Monitor Statistics Adapter Selection

Prompt on File or None

Advanced

Driver Ring Buffer size: 4000 kilobytes

Capture Log size: 128 kilobytes

Packet File Log size: 128 kilobytes

Figure 2.3 Workspace view of the Options dialog

In the ***Workspace*** view, you can set default program behavior for scrolling, saving, and restoring open windows on program launch. The *Monitor Statistics Adapter Selection* drop-down list lets you control when (and whether) the program will present the ***Adapter*** view of the **Monitor Options** dialog on program launch. You can also restore EtherPeek to its initial default configuration by clicking the **Revert to Defaults** button in the ***Workspace*** view of the **Options** dialog.

CAUTION! When you **Revert to Defaults**, user-entered data will be lost.

Use the *Monitor Statistics Adapter Selection* drop-down list to tell EtherPeek whether it should *Always prompt* for a Monitor statistics adapter on program start up, *Prompt on File or None* (that is, only if the previous adapter selection was *File* or *None*), or *Never prompt* for selection of a Monitor statistics adapter. If you choose *Never prompt*, EtherPeek will attempt to use the previously selected adapter as the Monitor statistics adapter for new sessions of EtherPeek, but will never prompt for adapter selection. If the previously selected adapter is not found, EtherPeek starts silently with *None* as the adapter type.

In the *Advanced* section of the ***Workspace*** view, you can set the *Driver Ring Buffer size*, the *Capture Log size* and/or the *Log File size* by entering a new value in *KB* (kilobytes). Changes in the *Driver Ring Buffer size* take effect the next time the program is started.

The *Capture Log size* is the default size assigned to the Log function in all new Capture windows. The *Log File size* is the default size assigned to the Log function in all new Packet File windows.

Tip The global program log is distinct from the logs in Capture windows and Packet File windows. To set the maximum size of the global log file, right-click inside the **EtherPeek Log** window, choose **Maximum Log File Size…** from the **EtherPeek Log** window context menu, and enter a new value in kilobytes.

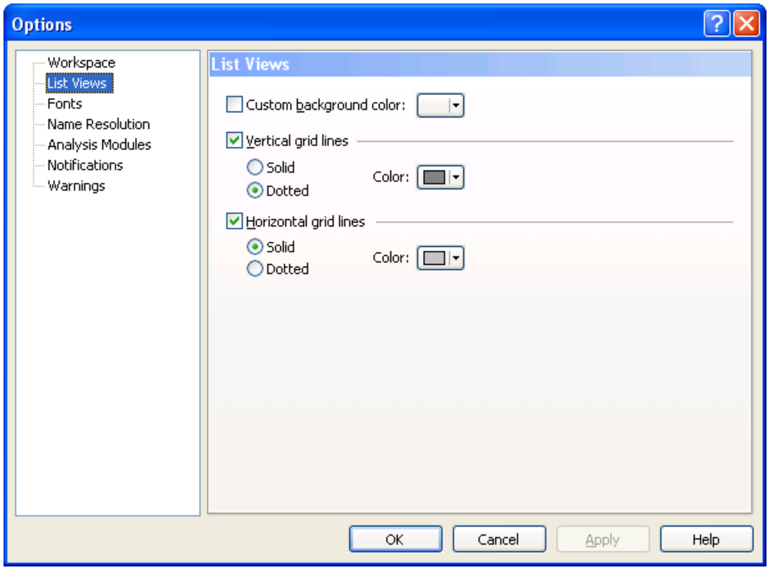
The *Driver Ring Buffer size* specifies the size of the ring buffer in the Peek driver. The Peek driver establishes a separate ring buffer of the size you specify in *Driver Ring Buffer size* for each adapter. The larger the driver ring buffer, the less chance of dropped packets in high traffic environments. but the greater the program's utilization of RAM. The maximum allowable setting for the *Driver Ring Buffer size* is 64 megabytes (*64000 KB*). The actual maximum available on a particular machine may be less, depending on the amount of installed RAM and the other uses of RAM, both by EtherPeek and by other applications and processes.

Important! The ring buffer referred to in the *Driver Ring Buffer size* item in the ***Workspace*** view of the **Options** dialog is ***NOT THE SAME*** as the ring buffer assigned to a Capture window, the size of which is specified in the *Buffer size* item in the **General** view of the **Capture Options** dialog. They are two distinct functions.

The *Driver Ring Buffer size* should be increased if the *Driver Statistics* in **Summary Statistics** indicate that packets are being dropped. Other factors can also affect performance. For a more complete discussion, please see “Optimizing performance” on page 24.

List views view

Choose **Options...** under the **Tools** menu and click the *List Views* item in the navigation pane to open the ***List Views*** view of the **Options** dialog (Figure 2.4).



List Views

☐ Custom background color:

☑ Vertical grid lines
○ Solid
● Dotted
Color:

☑ Horizontal grid lines
● Solid
○ Dotted
Color:

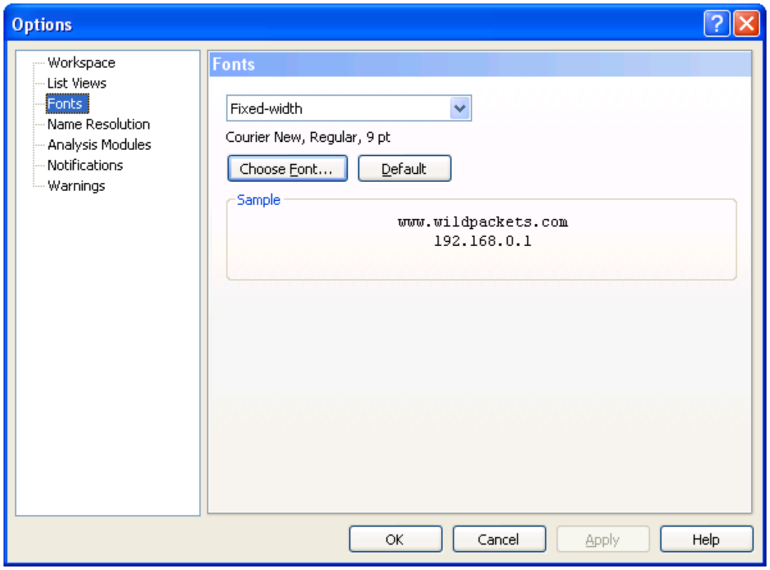
Figure 2.4 List Views view of the Options dialog

In the ***List Views*** view you can set the background color of list displays and the style and color of vertical and horizontal grid lines. When you have made your choices, click **Apply** to see them applied to the display. Click **OK** to accept your changes or click **Cancel** to close the dialog without making any changes.

Fonts view

Choose **Options...** under the **Tools** menu and click the *Fonts* item in the navigation pane to open the ***Fonts*** view of the **Options** dialog.

The ***Fonts*** view allows you to set the font, style, and size of the text used throughout the program to display information discovered by EtherPeek. Examples include the Packet List pane of the ***Packets*** view and all statistics views of Capture windows and Packet File windows, as well as data presented in Monitor statistics windows.



Fixed-width

Courier New, Regular, 9 pt

```
www.wildpackets.com
192.168.0.1
```

Figure 2.5 Fonts view of the Options dialog

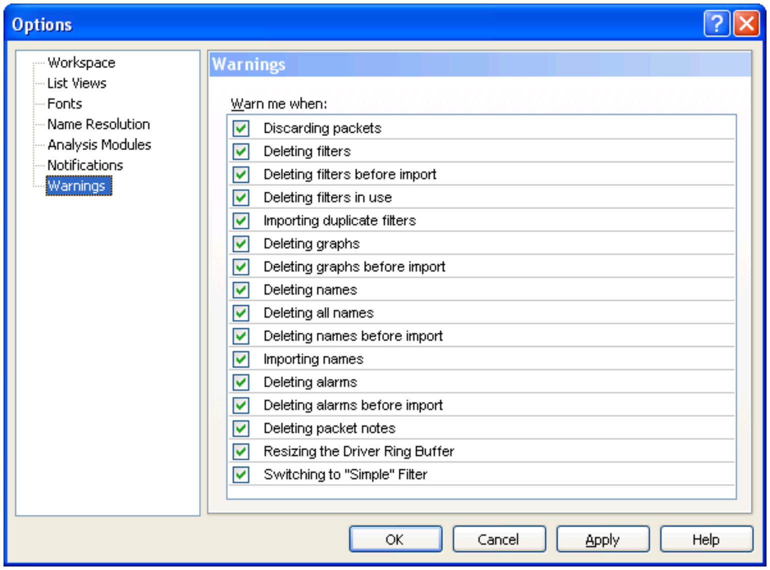
In the ***Fonts*** view, click the **Choose Font…** button to open the **Font** dialog, displaying the fonts installed on the local system. From this dialog you can choose any locally installed font, set the style (bold, italic, and so forth) and size, and choose the *Script* type (for example, *Western* for western languages such as English, German, and so forth). The **Font** dialog shows a sample of the new font. Click **OK** in the **Font** dialog to accept your changes or click **Cancel** to close without changing the font.

The ***Fonts*** view also shows a sample of the font currently in use. To restore the font selection to the program's initial default, click the **Default** button.

When you have made your choices, click **Apply** to see them applied to the display. Click **OK** to accept your changes or click **Cancel** to close the dialog without making any changes.

Warnings view

Choose **Options...** under the **Tools** menu and click the *Warnings* item in the navigation pane to open the ***Warnings*** view of the **Options** dialog.



Warnings

Warn me when:

- ☑ Discarding packets
- ☑ Deleting filters
- ☑ Deleting filters before import
- ☑ Deleting filters in use
- ☑ Importing duplicate filters
- ☑ Deleting graphs
- ☑ Deleting graphs before import
- ☑ Deleting names
- ☑ Deleting all names
- ☑ Deleting names before import
- ☑ Importing names
- ☑ Deleting alarms
- ☑ Deleting alarms before import
- ☑ Deleting packet notes
- ☑ Resizing the Driver Ring Buffer
- ☑ Switching to "Simple" Filter

Figure 2.6 Warnings view of the Options dialog

The ***Warnings*** view allows you to control the behavior of automatic warning dialogs that may appear in EtherPeek when you attempt to take certain actions. When the checkbox beside an item in the ***Warnings*** view is checked, the warning dialog will appear as normal when you attempt the specified action. When a particular type of warning is unchecked in the ***Warnings*** view, the attempted action will succeed without presenting any warning.

Most of the individual warning dialogs in EtherPeek also include a checkbox in the dialog itself which says *Do not ask again*. If you check this checkbox in an individual warning dialog, it has the same effect as disabling that type of warning in the ***Warnings*** view of the **Options** dialog. To restore the warning dialog for a particular type of action, place a check in the checkbox beside the particular type of action.

When you have made your choices, click **Apply** to see them applied to the display. Click **OK** to accept your changes or click **Cancel** to close the dialog without making any changes.

Optimizing performance

The performance of EtherPeek depends on many factors, some of which the user can control more easily than others. Understanding how EtherPeek works is important to getting the most out of the application, particularly in demanding environments.

Processor speed

Faster processors, those running at higher clock rates, help EtherPeek performance in two major ways: They process packets more quickly and they pass packets among drivers, applications and buffers more quickly. Both help prevent EtherPeek from dropping packets.

Peek driver ring buffer

The Peek driver ring buffer is used to optimize the capture and monitoring performance of the program when capturing from a particular adapter. Note that this is NOT the capture buffer. The larger the Peek driver ring buffer, the less chance of dropped packets in high traffic environments. but the greater the program's utilization of RAM. This parameter is set in the ***Workspace*** view of the **Options** dialog (available by choosing **Options…** from the **Tools** menu). For a complete discussion of this parameter, please see "Workspace view" on page 18.

Capture buffer and memory use

Packet capture in EtherPeek is handled by dedicated Capture windows, each with its own capture buffer of a user-defined size. In addition to setting the size of the capture buffer, the user also specifies how the Capture window will use that buffer. In the simplest arrangement, you can fill the buffer once and stop capture. Alternatively, you can use one of two methods to perform continuous capture. You can either discard all packets as the buffer becomes full, or you can use a ring buffer which continuously overwrites the same buffer, overwriting the oldest packets first. You can also save all packets captured with either of these continuous capture methods, periodically saving packets to disk before the buffer contents are discarded or overwritten. Each of these options, along with the size and number of capture buffers currently in use, has an effect on performance. Wrapping the buffer (emptying it in preparation for re-filling) can contribute to dropped packets, particularly when traffic volumes are high. Writing the contents of the capture buffer to disk can also allow some packets to be dropped in high traffic environments.

For a complete discussion of packet capture, including the **Capture Options** dialog, see Chapter 4, “Packet Capture” on page 45.

Performance views

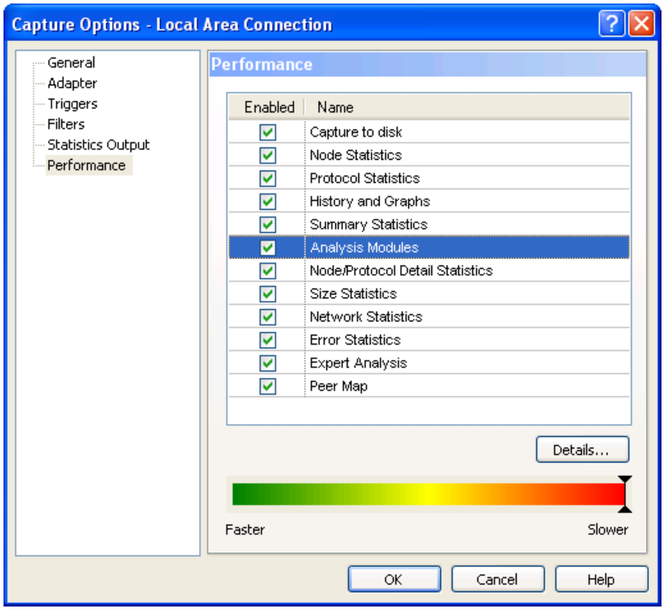
The ***Performance*** views of the **Monitor Options** and **Capture Options** dialogs allow you to selectively enable or disable individual program functions for a particular use of the selected adapter. For example, changes made to the ***Performance*** view of one Capture window affect only that window, not any others which may be open and capturing. This is true even when the other Capture windows are using the same adapter.

The ***Performance*** views of the **Monitor Options** and **Capture Options** dialogs present a list of program functions with checkboxes beside each. When a function is checked, it is enabled. To disable a function, uncheck the checkbox beside it. By default, all items are enabled.

Note: The *Expert Analysis* and *Peer Map* items are unique to EtherPeek NX. The *Conversations Analysis* item is unique to EtherPeek standard. Because *Capture To Disk*, *Expert Analysis*, *Peer Map*, and *Conversations Analysis* are not applicable to Monitor statistics, these items do not appear in the ***Performance*** view of the **Monitor Options** dialog.

Select the *Analysis Modules* item in the ***Performance*** view and click the **Details…** button to open the **Analysis Modules Performance** dialog. This dialog allows you to selectively enable (check) or disable (uncheck) the use of each individual Analysis Module in Monitor statistics or a particular Capture window. When some but not all of the individual Analysis Modules have been disabled, the check mark in the checkbox beside the *Analysis Modules* item in the ***Performance*** view turns gray.

Note: The **Analysis Modules Performance** dialog lists all the Analysis Modules installed by EtherPeek, regardless of whether they are enabled in the ***Analysis Modules*** view of the **Options** dialog. An Analysis Module must first be enabled in the ***Analysis Modules*** view of the **Options** dialog in order to be used by any function in EtherPeek. When an Analysis Module is checked in the **Analysis Modules Performance** dialog, it permits the use of the Analysis Module in the particular context (Monitor statistics or a particular Capture window). The Analysis Module will only actually be used if it is already enabled in the ***Analysis Modules*** view of the **Options** dialog.



Capture Options - Local Area Connection

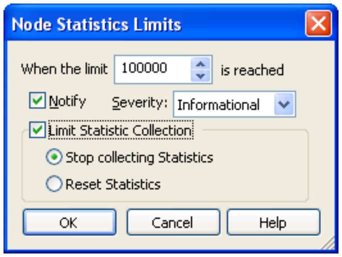
- General
- Adapter
- Triggers
- Filters
- Statistics Output
- Performance

Performance

Enabled	Name
☑	Capture to disk
☑	Node Statistics
☑	Protocol Statistics
☑	History and Graphs
☑	Summary Statistics
☑	Analysis Modules
☑	Node/Protocol Detail Statistics
☑	Size Statistics
☑	Network Statistics
☑	Error Statistics
☑	Expert Analysis
☑	Peer Map

Figure 2.7 Performance view of the Capture Options dialog

Three other items in the ***Performance*** views offer additional control over the use of resources by these functions. They are: *Node Statistics*, *Protocol Statistics*, and *Node/ Protocol Detail Statistics*. Select any of these items in the ***Performance*** view and click the **Details…** button to open a Limits dialog appropriate to the particular function. All the dialogs have the same layout and options as the **Node Statistics Limits** dialog, shown in Figure 2.8.



Node Statistics Limits

When the limit 100000 is reached

☑ Notify Severity: Informational

☑ Limit Statistic Collection

● Stop collecting Statistics

○ Reset Statistics

Figure 2.8 Node Statistics Limits dialog

The dialogs allow you to set a limit to the number of items that will be collected in each of these classes of statistics views, and determine what actions, if any, EtherPeek should take when this limit is reached. The items being limited are appropriate to each type of statistics view or window, as shown in Table 2.1.

Table 2.1 **Performance view -- limits dialogs**

Limits Dialog	Sets limits on
Node Statistics Limits	Limits the number of nodes (unique addresses), or (for the ***Hierarchical*** view of the ***Nodes*** view or **Node Statistics** window only) the number of node pairs, which can be displayed in the ***Nodes*** view (if you are setting **Capture Options**) or in the **Node Statistics** window (if you are setting **Monitor Options**).
Protocol Statistics Limits	Limits the number of unique protocols that can be displayed in the ***Protocols*** view (if you are setting **Capture Options**) or in the **Protocol Statistics** window (if you are setting **Monitor Options**).
Node/Protocol Detail Statistics Limits	Limits the number of conversations (address pairs using a particular port/protocol) that can be displayed in all **Detail Statistics** windows, whether opened from a Capture window or Monitor statistics windows.

The Limits dialogs are in the form of a statement, which begins *When the limit … is reached*. Use the combo box to enter the number of items to set as the upper limit. Optionally you may have EtherPeek *Notify* when this limit is reached, by checking that checkbox. Use the drop-down list at right to set a *Severity* for this notification. For more about notifications and levels of severity, please see “Notifications” on page 237. You can have EtherPeek *Limit Statistics Collection* when the limit you entered is reached, by checking that checkbox. When this option is checked, you may choose either of two ways of limiting statistics collection. You can either *Stop collecting Statistics* or *Reset Statistics*. Use the radio buttons to choose one of these options. Click **OK** to accept your changes or click **Cancel** to close the dialog without making any changes.

At the bottom of the ***Performance*** view is a spectral band labeled *Faster* at the left and *Slower* at the right. As you enable and disable program functions, an indicator moves along this band to give you a rough estimate of the relative impact of various combinations of features on the performance of Monitor statistics or the particular Capture window. The fewer functions enabled, the faster the performance; the more functions, the slower. The processing of *Size Statistics* has a negligible impact, for example, while the *Expert Analysis* and *Capture to disk* functions have substantial effects.

Starting EtherPeek from the command line

You can invoke EtherPeek from the command line using the following syntax:

```
Peek.exe [/autoload |/autostart ] [template1] [templateN]
```

The `/autoload` switch loads the specified Capture Template (*.ctf) file(s). The `/autostart` switch loads the specified template(s) and begins capture. Multiple templates may be listed, separated by a space. You can use the `*` (asterisk) character or the ? (question mark) character as wildcards in specifying template names, following standard Windows wildcard usage.

On a default install of EtherPeek NX, the command line would be started from:

```
C:\Program Files\WildPackets\EtherPeek NX
```

To automatically load template file capture1.ctf, for example, the command would be:

```
peek /autoload [template file location]\capture1.ctf
```

You can also invoke EtherPeek from the command line specifying an AutoCapture (*.wac) file as its object. For more about AutoCapture files, please see “AutoCapture” on page 88.

EtherPeek Menus and Toolbar

This chapter provides a complete list of EtherPeek menu commands, as well as an introduction to context menus and the EtherPeek toolbar. It also describes how to customize the **Tools** menu so you can launch such utilities as iNetTools directly from within EtherPeek.

In this Chapter:

EtherPeek menus

This is a listing of EtherPeek menus, with a brief description of each item's function. It is intended as a quick reference only, as several important caveats and other significant details are left out of the descriptions. Menu items followed by ellipses (for example, **New…**) open a dialog or window. Most sub-headings are either toggle choices (on or off, with a ✔ checkmark indicating the entry is on); or groups of alternative choices, only one of which may be active at a given time. The active choice is indicated by a checkmark.

You can make menu choices either by navigating the menus or by using the **Ctrl** key combinations shown beside certain items in the menus and in the list below.

File menu

New…	**Ctrl + N**	Creates a new Capture window.
New From Template ➤		Creates a new Capture window whose layout matches the template selected by one of the two methods below.
Choose…		Opens a file **Open** dialog wherein you can navigate to the Capture window template of your choice.
(recent templates)		A list of the most recently used Capture window templates. Choose one to create a new Capture window using this template.
Open…	**Ctrl + O**	Opens an EtherPeek packet file or other supported file type in a new Packet File window.
AutoCapture ➤		
Create New…		Opens an empty **AutoCapture File** window in which you can define the parameters for a new AutoCapture file.
Edit Existing…		Opens a file **Open** dialog in which you can navigate to the AutoCapture (*.wac) file of your choice.
Close		Closes the active window or file.

Save All Packets…	**Ctrl + S**	Opens the **Save** dialog to save all packets in the active window.
Save Selected Packets…		Opens the **Save** dialog to save selected packets in the active window. This item is displayed as **Save Filters…** when the **Filters** window is active, as **Save Graph…** when a **Graph** window or the ***Graphs*** view of a Capture window or Packet File window is active, as **Save Names…** when the **Name Table** window is active, or as **Save Log…** when the **Log** window is active. When a statistics window is active, it changes to allow you to save the active statistics window or view, and will appear as, for example: **Save Node Statistics…** or **Save Size Statistics…**, and so forth.
Save Report…		Opens the **Save Report** dialog to choose the file format and location in which to save a report on any of several collections of statistics for the current Capture window or Packet File window. Formats include text (*.txt, *.csv), HTML, or XML.
Save Capture Template…		Opens the **Save** dialog to save the Capture Options of the current Capture window as a capture template (*.ctf), so it can be used to format subsequent new Capture windows.
Print Setup…		Opens the **Print Setup…** dialog for configuring printer functions.
Print…	**Ctrl + P**	Prints the active window in a format appropriate to its type.
Print Selected Packets…		Opens the **Print** dialog to allow you to print the ***Decode*** view of the selected packets as a single document. That is, without page breaks between packets.
Recent File		Following the **Print Selected Packets…** command is a numbered list of recently opened packet files, with the most recently opened listed first. You can select a file from this list to open it.
Exit	**Alt + F4**	Quits EtherPeek.

Edit menu

Undo	**Ctrl + Z**	Undoes the last edit.
Cut	**Ctrl + X**	Cuts the highlighted item(s) and copies to the clipboard.
Copy	**Ctrl + C**	Copies highlighted item(s) to the clipboard.
Paste	**Ctrl + V**	Pastes the current contents of the clipboard.
Insert	**Ins**	When the **Filters** window is active, opens the **Edit Filter** dialog; when the **Name Table** window is active, opens the **Edit Name** dialog.
Delete	**Del**	Deletes the highlighted item(s).
Clear All Packets	**Ctrl + B**	Deletes all packets from the active Capture window.
Hide Selected Packets	**Ctrl + H**	Removes selected packets from the display without deleting them. Hidden packets are not processed further.
Hide Unselected Packets	**Ctrl + Shift + H**	Removes unselected packets from the display without deleting them. Hidden packets are not processed further.
Unhide All Packets	**Ctrl + U**	Restores all previously hidden packets to normal status.
Select…	**Ctrl + E**	Opens the **Select** dialog, where you can use filters, ASCII or hex strings, packet length, and Analysis Modules to select captured packets.
Select Related Packets ➤		Searches for and selects packets that provide best matches to the highlighted item(s), based on the set of characteristics chosen from the list below.
By Source		Chooses packets with matching source address.
By Destination		Chooses packets with matching destination address.

By Source and Destination		Chooses packets with matching source and destination addresses.
By Protocol		Chooses packets with matching protocol.
By Port		Chooses packets with matching port.
By Conversation		Chooses packets sent between two nodes, using the matching protocol.
Select All	**Ctrl + A**	Selects all packets, text, or items in a window.
Select None	**Ctrl + D**	Removes all highlighting and selection.
Invert Selection		Unselects items that were selected and selects items that were unselected.
Find Pattern	**Ctrl + F**	Opens the **Find Pattern** dialog to search for a user-defined string in specified parts of packets.
Find Next	**F3**	Finds the next match in sequence to the previous **Find Pattern** search.
Go To…	**Ctrl + G**	Opens the **Go To** dialog where you can choose a packet number to jump to. If packets are selected, the number of the first selected packet is shown.
Go To Next Selected	**Ctrl + J**	Jumps to the next selected packet.

View menu

Filters	**Ctrl + M**	Opens the **Filters** window.
Name Table		Opens the **Name Table** window.
Log Window	**Ctrl + L**	Opens the **Log** window.
Alarms		Opens the **Alarms** window.
Display Format ➤		The following options control display format for nodes:

Name Table Entry

Display using the names found in the Name Table when available (on by default).

Logical Address

Display using the logical address of the node where available (on by default).

Physical Address

Display using the hardware (MAC) address only.

Color ➤

The following options control the use of color in ***Packets*** views and other displays:

Source

Use the color assigned to the source node.

Destination

Use the color assigned to the destination node.

Protocol

Use the color assigned to the protocol.

Filter

Use the color assigned to the filter that allowed the packet to be captured.

Flag

Use the color assigned to flagged packets.

Independent

Each item uses its own color.

No Color

Use no color coding in ***Packets*** view and other displays.

Toolbar

Operates as a toggle setting. When enabled (the default), displays toolbar of convenient button versions of many of these menu commands.

Status Bar

Operates as a toggle setting. When enabled (the default), displays status alerts and the current adapter in a bar at the bottom of the main program window.

Capture menu

Start Capture	**Ctrl + Y**	Toggles the packet capture function. When capture is active, the item is displayed as **Stop Capture**. When the active window has a Start Trigger, this item can display as **Start Trigger** to start the trigger or **Abort Trigger** to abort the trigger process.
Capture Options…		Opens the **Capture Options** dialog, where you can use the various views to set ***General*** properties such as the capture buffer options, and specify the ***Adapter***, ***Triggers***, ***Filters***, and ***Statistics Output*** options for the active Capture window. In addition, you can selectively enable or disable individual program functions within the active Capture window to optimize ***Performance***.

Send menu

Initiate Send	**Ctrl + I**	Starts sending packets using the parameters you set in the **Send Window**.
Transmit One	**Ctrl + T**	Sends one copy of the designated Send Packet.
Send Selected Packets		Sends selected packets onto the network.
Set Send Packet		Designates a Send Packet.
Edit Send Packet...		Opens the designated Send Packet in a Decode window with edit capabilities.
Send Window		Opens the **Send Window**, where you can control transmissions from EtherPeek.

Monitor menu

Nodes	**Ctrl + 1**	Opens the monitor **Node Statistics** window.

Protocols	**Ctrl + 2**	Opens the monitor **Protocol Statistics** window.
Network	**Ctrl + 3**	Opens the monitor **Network Statistics** window.
Size	**Ctrl + 4**	Opens the monitor packet **Size Statistics** window.
Summary	**Ctrl + 5**	Opens the monitor **Summary Statistics** window.
History	**Ctrl + 6**	Opens the monitor **History Statistics** window.
Monitor Statistics		Operates as a toggle setting. When enabled (the default), collects all network statistics, independent of any Capture window.
Reset Statistics		This action clears all accumulated Monitor statistics information and resets all values to zero.
Monitor Options…		Opens the **Monitor Options** dialog. In the named views of that dialog, you can: select an ***Adapter*** for use in collecting Monitor statistics, set options for periodic ***Statistics Output***, and selectively enable or disable individual program functions as they apply to Monitor statistics to optimize ***Performance***.

Tools menu

Options…	Opens the **Options** dialog where you can specify default program behavior in the areas corresponding to each of this dialog's views: ***Workspace***, ***List Views***, ***Fonts***, ***Name Resolution***, ***Analysis Modules***, ***Notifications***, and ***Warnings***. From the ***Workspace*** view of this dialog you can also globally restore program defaults.
Customize…	Opens the **Customize Tools Menu** dialog from which you can add items to the **Tools** menu, allowing you to launch other programs from within EtherPeek. Use this dialog to add utilities from iNetTools, for example, to the EtherPeek menu.

Window menu

Cascade		Arranges all open windows one behind the other, with only the tops of those behind showing above the others.
Tile Vertically		Fills the screen with open windows, arranged side-by-side.
Tile Horizontally		Fills the screen with open windows, arranged one above the other.
Arrange Icons		Lines up the icons of minimized open files.
Next	**Ctrl + Tab**	Makes the next window in sequence the active window.
Previous	**Ctrl + Shift + Tab**	Makes the previous window in sequence the active window.
Close All		Closes all open windows.

At the bottom of the **Window** menu is a numbered list of open windows, with a checkmark beside the name of the active or front-most window. Selecting a window from this list makes it the active window and brings it to the front of the display.

Help menu

Help Topics	**F1**	Launches the Windows Help function for EtherPeek.
Show Start Page		Opens the Start Page.
Readme		Opens the Readme file, containing information about the program which may have appeared since the publication of the current manual.
Quick Tour		Opens the EtherPeek Quick Tour, introducing some of the key program features.

WildPackets on the Web ➤
The following indented items will launch the default Internet browser and load the appropriate page from the WildPackets website.

Product News
Loads the latest product news about EtherPeek and related WildPackets products.

Technical Support
Loads the technical support pages.

Training
Loads pages describing WildPackets' extensive courses in EtherPeek and related network troubleshooting tools and techniques.

WildPackets Home Page
Loads the WildPackets home page.

About EtherPeek
Appears as **About EtherPeek NX** in EtherPeek NX, and as **About EtherPeek** in EtherPeek standard. Displays the EtherPeek about box, including the last 10 characters of the serial number of your copy, and the support function, described below.

Support…
Click the **Support…** button in the **About EtherPeek** dialog to display key system and program information useful in troubleshooting and technical support. You can also save this information to a text file from this dialog.

Context menus

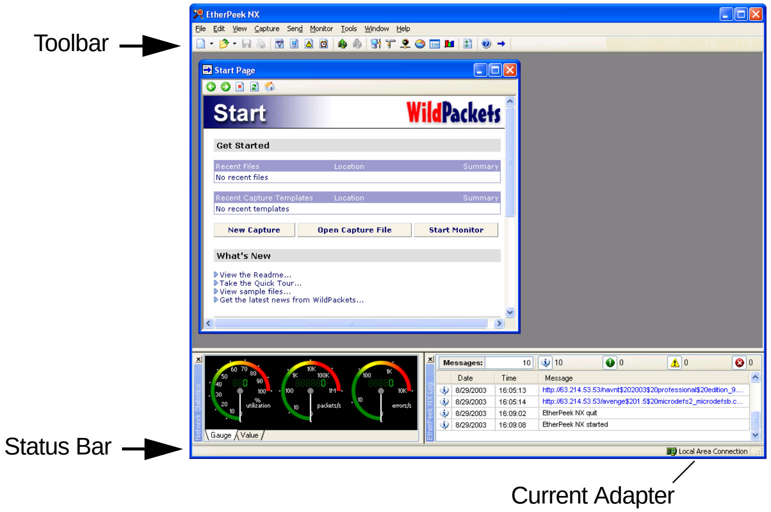
Context-specific menus are available in most windows of EtherPeek by right-clicking inside the window. The content of these menus changes with the active window and depends in some cases on whether or not items are selected.

Main program window start page and tools menu

This section describes two useful features of the EtherPeek main program window, the toolbar and status bar, and describes the **Start Page** which appears on program start-up under the default settings. It also shows how to customize the **Tools** menu by adding other programs, and describes one of those programs in detail: WildPackets' IP test suite, iNetTools.

EtherPeek toolbar

You can show or hide the toolbar for EtherPeek by selecting or deselecting the **Toolbar** item in the **View** menu. The toolbar provides button navigation for frequently-used tasks in EtherPeek. The name of each button's function appears when the cursor is moved over the button.



Get Started

Recent Files	Location	Summary
No recent files		

Recent Capture Templates	Location	Summary
No recent templates		

What's New

- View the Readme...
- Take the Quick Tour...
- View sample files...
- Get the latest news from WildPackets...

Messages: 10 ⓘ 10 ! 0 ⚠ 0 ✖ 0

	Date	Time	Message
ⓘ	8/29/2003	16:05:13	http://63.214.53.53/navnt$202003$20professional$20edition_9....
ⓘ	8/29/2003	16:05:14	http://63.214.53.53/avenge$201.5$20microdefs2_microdefsb.c...
ⓘ	8/29/2003	16:09:02	EtherPeek NX quit
ⓘ	8/29/2003	16:09:08	EtherPeek NX started

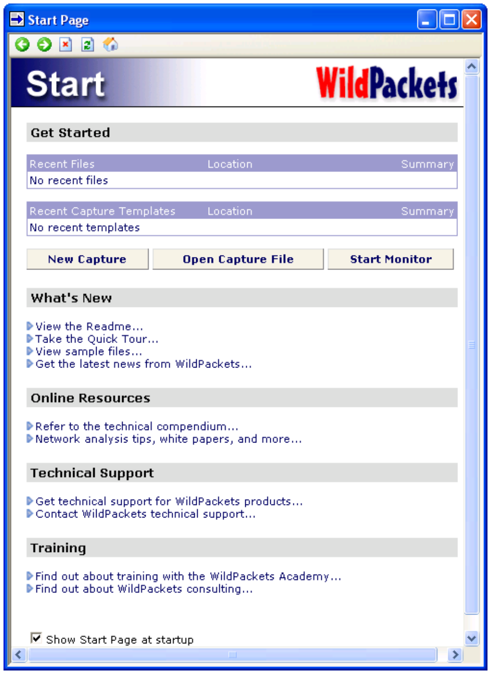
Figure 3.1 Main program window, showing location of Toolbar and Status Bar

EtherPeek program window status bar

Located at the bottom of the main program window, the main program status bar shows brief context-sensitive messages on the left and the current adapter on the right. You can click on the current adapter item to open the ***Adapter*** view of the **Monitor Options** dialog. Use this view to select an adapter for use in collecting Monitor statistics. Choose **Status Bar** under the **View** menu to toggle the display of this main program status bar. A checkmark appears beside this item when it is enabled, as it is by default.

Start Page

The first time you open EtherPeek, the **Start Page** appears. This is an HTML page with links to useful resources, both local and online. From the **Start Page**, you can open recently used Packet files, start a new Capture window, browse sample Packet files or other Packet files, view the Readme file or manual, take the Quick Tour, and more.



Get Started

Recent Files	Location	Summary
No recent files		

Recent Capture Templates	Location	Summary
No recent templates		

What's New

- View the Readme...
- Take the Quick Tour...
- View sample files...
- Get the latest news from WildPackets...

Online Resources

- Refer to the technical compendium...
- Network analysis tips, white papers, and more...

Technical Support

- Get technical support for WildPackets products...
- Contact WildPackets technical support...

Training

- Find out about training with the WildPackets Academy...
- Find out about WildPackets consulting...

☑ Show Start Page at startup

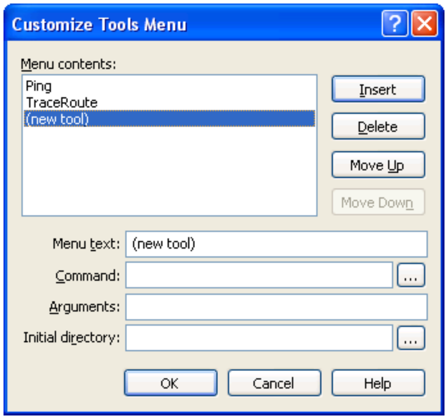
Figure 3.2 Start page

Check the checkbox at the bottom of the **Start Page** to *Show Start Page at start-up*. Alternatively, you can always view the **Start Page** by choosing **Show Start Page** from the **Help** menu.

Customizing the tools menu

You can add programs to the **Tools** menu, allowing you to launch them from within EtherPeek. Choose **Customize…** from the **Tools** menu to open the **Customize Tools Menu** dialog. This dialog lets you manage add-in items that appear in the **Tools** menu.

To add a program to the **Tools** menu, click the **Insert** button. This creates a blank item called *[new tool]*, highlighted in the *Menu contents* list. Use the text entry boxes to set the parameters for this new item. The *Menu text* field sets the name of the tool as it will appear in the **Tools** menu. In the *Command* field, type the path to the program, or use the **...** (ellipsis) button to navigate to its location. Optionally, you can enter any *Arguments* for the program, and set its initial directory by typing the path or using the **...** (ellipsis) button to navigate to its location.



Menu contents:
Ping
TraceRoute
(new tool)

Menu text: (new tool)
Command:
Arguments:
Initial directory:

Figure 3.3 Customize Tools Menu dialog

To remove an item, highlight its name in the *Menu contents* list and click the **Delete** button. Items appear at the end of the **Tools** menu in the same order in which they appear in the *Menu contents* list. To change the order, highlight an item and use the **Move Up** and **Move Down** buttons. When you have made your changes, click **OK** to accept your changes and close the dialog, or click **Cancel** to close the dialog and discard your changes.

An example of tools that can be added to the menu is the iNetTools suite of applications. Each individual part of the iNetTools suite can be added to the **Tools** menu as a separate menu choice. If you accept the option to add iNetTools to the **Tools** menu during installation, for example, all the parts of the suite will be added.

iNetTools

WildPackets' iNetTools is a collection of easy-to-use GUI-based tools for testing Internet and IP-based networks. The iNetTools suite is included on the EtherPeek distribution CD, and is also available from the WildPackets website at http://www.wildpackets.com. The current tools are shown in Table 3.1 below.

Table 3.1 **iNetTools components**

iNetTool	Description
Ping	uses the ICMP protocol to send echo request packets to a device and times the responses.
Ping Scan	pings a range of IP addresses to find out which addresses are currently in use.
Trace Route	traces the route packets take from your computer to any device with an IP address.
Name Lookup	resolves names to IP addresses and IP addresses to names.
Name Scan	performs a Name Lookup for a range of IP addresses.
DNS Lookup	provides detailed information on Internet hosts by querying Domain Name Servers.
Port Scan	scans ports on a machine to find supported services, such as HTTP, telnet, and FTP.
Service Scan	scans a range of IP addresses for services, such as FTP, HTTP, and telnet.
Finger	uses the finger protocol to get information about a user on a given server.
Whois	uses the WHOIS protocol to query database servers such as whois.internic.net for Internet directory information.
Throughput	connects to an FTP or an HTTP (Web) server to test download speed of FTP or Web files.

In addition to the above tools, iNetTools features the following reports and references:

Network/IP Configuration Information… uses 'IPCONFIG' under Windows 2000, or Windows XP.

Network Statistics… uses the NETSTAT command to display routing information and other network traffic details.

ARP Cache Content… uses the ARP command to list a system-cached record of associations between IP addresses and physical addresses.

Internet Port Description… lists Internet port numbers and descriptions, downloaded from the IANA (Internet Assigned Numbers Authority) site.

Packet Capture

EtherPeek can capture packets in multiple configurable Capture windows, each with its own selected adapter, its own dedicated capture buffer and its own settings for filters, triggers, and statistics output. Capture windows let you monitor, collect statistics, and capture from multiple adapters simultaneously. You can establish and view multiple Capture windows up to the limits of available memory and screen space.

Capture windows allow you to:

- view and monitor network traffic in real time
- use a different adapter for each Capture window
- apply filters, both before and after capture
- start and/or stop capture based on network events
- view statistics based on selected network traffic
- view packet contents, raw and/or decoded
- save packets for post-capture analysis in Packet File windows
- create snapshots of particular network conditions for future comparison
- enable all or only a few features in each window
- separate potential problems from severe ones
- use expert analysis to monitor and troubleshoot

This chapter explains how to set up a Capture window and configure its use of adapters and memory, how to customize its appearance, how to save packets and reload them in a Packet File window, and how to print out captured packets.

In this Chapter:

Capture window basics

This section presents the basic form and function of the Capture window. To capture packets in EtherPeek, you create a Capture window, set or accept its parameters, and click the **Start Capture** button. It's as simple as that. Because Capture windows can be configured to meet a variety of user needs, there are multiple ways to perform each of these functions. These are covered in detail in the sections below. The Capture window basics section ends with an overview of Capture window layout and structure.

Creating a new capture window

You can create a new Capture window in any of several ways. You can click the **New Capture** button on the **Start Page**. You can select **New…** from the **File** menu or type **Ctrl + N**. Alternatively, if you have created one or more capture templates, you can choose **New From Template** from the **File** menu to select a recently used capture template from the submenu list, or use the **Choose…** submenu item to navigate to a capture template (*.ctf) and open it as a new Capture window using a standard file **Open** dialog. Finally, if no Capture window is open, selecting **Start Capture** from the **Capture** menu or typing **Ctrl + Y** will also open a new Capture window.

The first time you open a Capture window, you will see the **Capture Options** dialog. The **Capture Options** dialog defines all the parameters for a Capture window. At a minimum, the definition of a Capture window requires a selected adapter, a memory allocation called a capture buffer, and a set of parameters defining how to use the buffer. All of these parameters must be set for each Capture window when it is created. You can set them by hand, accept the defaults, or use the settings stored in a capture template. Please see "Capture options: general" on page 52, for details about the **Capture Options** dialog and how to use it. Choose new values for the parameters or accept the defaults and click **OK** to create a new Capture window.

Using default settings and capture templates

If you do not want to be presented with the **Capture Options** dialog each time you open a new Capture window, you have two choices.

The first method is to set the parameters in the **Capture Options** dialog to the values you wish to use for all subsequent Capture windows and, in the ***General*** view uncheck the checkbox beside *Show this dialog when creating a new capture window.* Each time you create a new Capture window it will open immediately using these parameters. New windows will be named *Capture 1*, *Capture 2*, and so forth in sequence as each new window is created during a session of EtherPeek. To return to having the **Capture Options** dialog presented each time you open a new Capture window, make a Capture window the active (frontmost) window, choose **Capture Options…** from the **Capture** menu to open the **Capture Options** dialog, open the ***General*** view, and re-enable that option by checking the checkbox.

The second method is to create one or more capture templates and use them to create new Capture windows. Templates supply the **Capture Options** dialog settings for windows created from them. You can save any Capture window as a named capture template by making that Capture window the active window and choosing **Save Capture Template…** from the **File** menu. This opens a **Save As** dialog where you can choose the location in which to save the template and give the template a name. Save the template as a *Capture Template* format (*.ctf) file. A capture template contains all of the settings in the **Capture Options** dialog, and applies these to any Capture window created using the **New From Template…** command under the **File** menu. When you create a new Capture window from a template, the new window uses the Capture window title specified in the template, adding the numbers *1*, *2*, *3*… only when necessary to distinguish between multiple instances open at the same time. Capture windows created from templates are created without opening the **Capture Options** dialog, regardless of whether the checkbox labeled *Show this dialog when creating a new capture window* is checked or unchecked.

Tip You can also create a single named template that will create multiple Capture windows, each with its own individual capture options. Create or open the Capture windows you wish to include in the template. Make sure only the Capture windows you wish to include are open. Hold down the **Ctrl** key and choose **Save Capture Template…** from the **File** menu. The saved template will include all the open Capture windows.

Capture templates are also used when invoking EtherPeek from the command line. Please see “Starting EtherPeek from the command line” on page 28. The AutoCapture feature also allows you to create, import, and export settings from capture templates, and use them to programmatically invoke capture by EtherPeek, AiroPeek, GigaPeek NX, or Packet Grabber. Please see “AutoCapture” on page 88.

Important! The definition of a Capture window must include the selection of a valid adapter. If the adapter named in your default settings or capture template is not found, EtherPeek will present an error message. Click **OK** to clear this error message and bring up the ***Adapter*** view of the **Capture Options** dialog, from which you can select a valid adapter for the new Capture window.

Starting and stopping capture in a capture window

To start capturing packets, click the **Start Capture** button in the upper right of the Capture window (see Figure 4.1). The label on the button will change to **Stop Capture** when capture is under way. Alternatively you can use the **Start Capture** command from the **Capture** menu or press **Ctrl + Y** to start capture in whichever Capture window is the active (frontmost) window at the time. Both the **Ctrl + Y** sequence and the choices under the **Capture** menu act as toggles, starting or stopping capture depending on the state of the active Capture window. To toggle the start and stop of capture in all open Capture windows simultaneously, hold down the **Ctrl** key and choose **Start Capture** from the **Capture** menu or hold down the **Ctrl** key and click the **Start Capture** button on any open Capture window.

Tip You can also start and/or stop capture based on a time event or a filter match, by setting a trigger for the new Capture window. For more on triggers, please see “Triggers” on page 224.

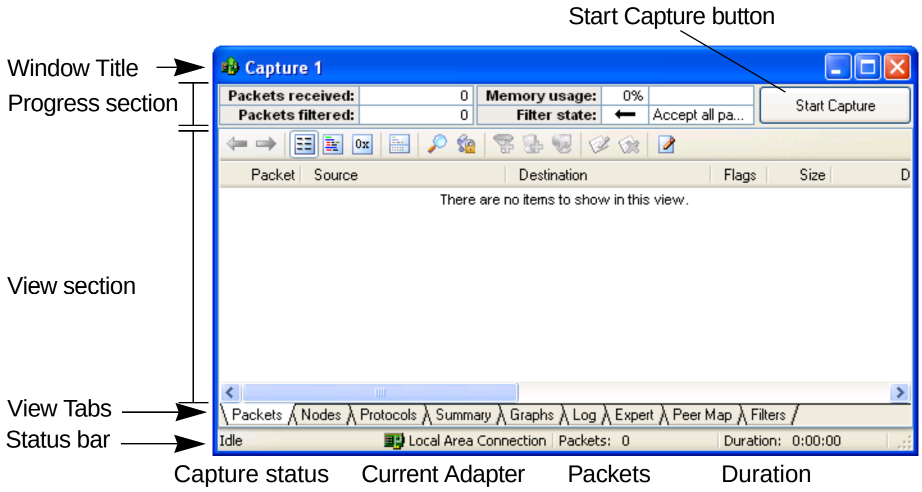
A progress bar labeled *Memory usage* tracks the percentage of that particular Capture window’s capture buffer that has been filled. You will notice that the *Memory usage* bar resets to zero when the buffer becomes full and is dumped to begin refilling. This can happen when the buffer wraps automatically under continuous capture, or when you clear the buffer manually, either by using the **Clear All Packets** command from the **Edit** menu, typing **Ctrl + B**, or by restarting capture in that window without saving already captured packets. The other indicators in the progress section (*Packets received* and *Packets filtered*) will continue to increment without interruption, even when the buffer wraps.

When you stop capture, all of the packets currently in the buffer for that Capture window are retained, and any statistics shown in any of the other views will be based on all the packets seen since capture was initiated for that Capture window. If you then restart capture for that Capture window, EtherPeek will clear the window’s buffer *and its statistics* and begin again from zero. The only way to restart capture in a Capture window without clearing the buffer (thus retaining any packets and any statistics collected so far) is to use **Shift + Click**. Hold down the **Shift** key while you click the **Start Capture** button to restart capture without clearing the existing contents of the buffer.

Because all packets and statistics will be lost when you close a Capture window without saving, EtherPeek warns you each time you close a Capture window. To change this and other default display behaviors, use the **Options** dialog, available by choosing **Options…** under the **Tools** menu.

Capture window structure

Each Capture window has a progress section at the top showing basic statistics for the window as a whole, and a lower section showing one of several different views selected by clicking the appropriate view tab.



Packets received: 0
Packets filtered: 0
Memory usage: 0%
Filter state: Accept all pa...

There are no items to show in this view.

Idle | Local Area Connection | Packets: 0 | Duration: 0:00:00

Figure 4.1 Parts of a Capture window

The parts of the Capture window common to every view are labeled in Figure 4.1 and described in Table 4.1. For a description of the individual views available in a Capture window please see “Capture window views” on page 62.

Table 4.1 Parts of a Capture window (see Figure 4.1)

Window part	Description
Capture window title	The user-defined (or default) title of the Capture window.
Start Capture	Click the **Start Capture** button to begin capturing packets. When capture is under way, the label on the button changes to **Stop Capture**. When a trigger is set for the Capture window, this button can be labeled in different ways. Please see “Triggers” on page 224 for details.

Table 4.1 **Parts of a Capture window (see Figure 4.1) (Continued)**

Window part	Description
Progress section	The progress section shows the following four parameters of capture activity:
Packets received	Shows total packets presented to the filters since capture was initiated for this window—essentially, the total number of packets on the network since capture was initiated for this Capture window.
Packets filtered	Shows, of those received, the total number of packets matching the filter or filters set for this window. If there are no filters, then *Packets Received* and *Packets Filtered* will be equal. One exception might be any packets dropped when the buffer is wrapping.
Memory usage	Shows the percentage of configured capture buffer memory used so far in packet capture for the current Capture window. This percentage is displayed as a number and graphically by a progress bar which fills more of the width of the display as memory is used. Also, as memory use approaches 100%, the color of the bar changes from blue to warmer colors, eventually showing red when 100% of the capture buffer is used.
Filter state	Summarizes any enabled filter conditions. For example, *Accept only packets matching any of two filters*. An icon indicates whether filters are set to accept or reject matching packets. Double-click in this area to open the ***Filters*** view of the **Capture Options** dialog.
View section	Shows the current view of the Capture window, which can be selected by using the view tabs located just below the view section near the bottom of the window.
View Tabs	Shows the current and the available views. The tab for the current view is shown in white or a foreground color. The others are shown in gray or a background color. Click on a tab to see a particular view of the Capture window displayed in the view section.
Status bar	At the bottom of the window, the following four items show the status of capture activity:

Table 4.1 Parts of a Capture window (see Figure 4.1) (Continued)

Window part	Description
Capture status	Shows the current state of the capture process for the Capture window. For example, *Idle* or *Capturing*.
Current Adapter	Shows the currently selected adapter. Double-click on this item to open the ***Adapter*** view of the **Capture Options** dialog, where you can select another adapter, set the network speed, and so forth.
Packets	Shows the number of packets in the buffer. When some packets have been hidden, shows, for example, *2 (of 48)*.
Duration	Shows the difference between the earliest and the most recent packet in the current window.

Capture options dialog

The **Capture Options** dialog defines all the parameters for a Capture window. The parameters are displayed in six views, accessible by clicking their names in the navigation pane: ***General***, ***Adapter***, ***Triggers***, ***Filters***, ***Statistics Output***, and ***Performance***. Each of these views and all their parameters are described below.

Note: At a minimum, you must set the capture buffer options (in the ***General*** view) and select a valid adapter (in the ***Adapter*** view) to define a Capture window. The other capabilities are optional.

Very briefly the functions of the views of the **Capture Options** dialog are as follows:

General — Set the size and method of use of the capture buffer for this Capture window. Also controls packet slicing (saving only the first *n* bytes of each packet). (Please see “Capture options: general” on page 52.)

Adapter — Choose the adapter from which this particular Capture window will capture packets. Choose any supported adapter: file or local Ethernet card. (Please see “Capture options: adapter” on page 58.)

Triggers Define time, network, or capture events to trigger the start and/or stop of capture in this Capture window. (Please see “Triggers” on page 224.)

Filters Select filters and define how they will be used to limit the packets captured into this Capture window. (Please see “Capture options: filters” on page 61.)

Statistics Output Choose from a variety of formats and statistics for periodic output from this Capture window, at a frequency you set. (Please see “Statistics output views” on page 173.)

Performance Selectively enable or disable individual program functions for a particular Capture window. (Please see “Performance views” on page 25.)

Each Capture window is defined by its own **Capture Options** settings. You can have multiple Capture windows open simultaneously, capturing and displaying in real time. You can quickly create a new Capture window using either your own or the factory default settings. You can also save **Capture Options** settings as a capture template and use these templates to create a fully configured Capture window—complete with triggers, filters, and statistics output options—in a matter of a few clicks or keystrokes. You can use these same templates as the basis for capture in PacketGrabber, or import them to AutoCapture and invoke capture in remote instances of EtherPeek, EtherPeek standard, EtherPeek NX, or PacketGrabber and have the resulting packet files emailed or FTP’ed to you as soon as capture is complete.

Capture options: general

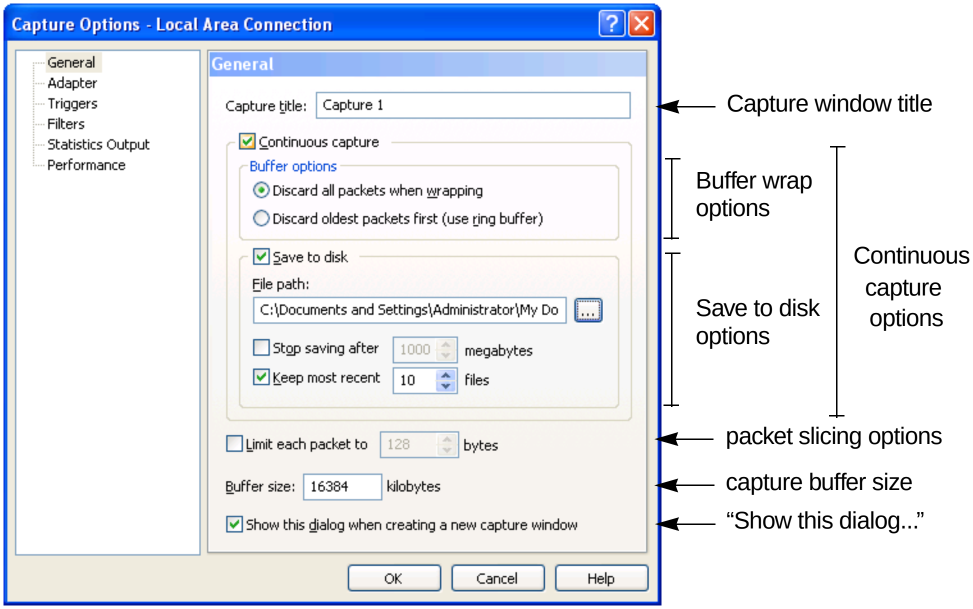
This section describes how to use the ***General*** view of the **Capture Options** dialog to set the capture buffer size and other important packet capture parameters for Capture windows.

Each Capture window has its own assigned memory allocation called a capture buffer, and a set of parameters telling the Capture window how to use that memory. In addition, a Capture window can be set to use a space-saving technique called packet slicing, in which it captures only a specified number of bytes from the beginning of each packet and ignores the rest. These parameters must be set for each Capture window when it is created, either directly by the user in the **Capture Options** dialog, or by configuring

EtherPeek to always use default Capture Options settings, or by using the Capture Options settings contained in a capture template. In addition, you can change the Capture Options settings for any Capture window by making it the active window and choosing **Capture Options…** from the **Capture** menu to bring up the **Capture Options** dialog and editing the values displayed there.

If the checkbox beside the *Show this dialog when creating a new capture window* item in the ***General*** view of the **Capture Options** dialog is checked, as it is by default, the **Capture Options** dialog will display each time you create a new Capture window using the **New…** command under the **File** menu or typing **Ctrl + N**. If no Capture windows are open, selecting **Start Capture** from the **Capture** menu or typing **Ctrl + Y** will do the same thing. This brings up the **Capture Options** dialog, shown in Figure 4.2.

In the ***General*** view of the **Capture Options** dialog, you can choose whether the Capture window will continuously capture packets (either discarding or saving previously captured packets each time the buffer becomes full), or simply stop capturing packets when all of its buffer memory has been used. The default setting is to stop capture when the buffer is full.



Capture Options - Local Area Connection

General
Capture title: Capture 1
☑ Continuous capture
Buffer options
● Discard all packets when wrapping
○ Discard oldest packets first (use ring buffer)
☑ Save to disk
File path: C:\Documents and Settings\Administrator\My Do
□ Stop saving after 1000 megabytes
☑ Keep most recent 10 files
□ Limit each packet to 128 bytes
Buffer size: 16384 kilobytes
☑ Show this dialog when creating a new capture window

Figure 4.2 General view of the Capture Options dialog

Your options are:

- **Capture until buffer is full:** This is the initial default setting. When the buffer is full, capture stops. When the *Continuous capture* checkbox is unchecked, capture will stop when the buffer becomes full.
- **Continuous Capture**: Periodically discards packets from the buffer to make room for new capture. When you check *Continuous capture*, capture does not stop until it is stopped manually by the user or by a stop trigger.
- **Continuous Capture, Save to Disk**: Periodically saves captured packets before emptying the buffer. You can limit the total disk space allocated for the saved files. When you check *Continuous capture*, and *Save to disk*, capture does not stop until it is stopped manually by the user or by a stop trigger.

Each of these options is explained in detail below.

You can change the settings in the **Capture Options** dialog for the active (frontmost) Capture window by choosing **Capture Options…** under the **Capture** menu.

Capture until the buffer is full

If you accept all the program's initial default settings, the new Capture window will stop capture when its buffer is full.

To create a new Capture window that will stop capture when its buffer becomes full:

1. Accept the default name for the new Capture window, or enter a new name in *Capture title*.
2. Make sure *Continuous capture* is disabled (unchecked), as it is by default.
3. Optionally, you can limit the amount of each captured packet to be saved. Please see "Using packet slicing" on page 57 for more details about this space-saving technique.
4. Accept the default *Buffer size: 16384 kilobytes*, or enter a new value for the buffer size.
5. When you have set all of the parameters, click **OK** to create the new Capture window.

Note: To avoid arbitrarily slicing the last packet captured, when you specify a buffer size in bytes, EtherPeek captures the whole packet that caused the specified *Buffer size* to be reached.

Continuous capture

When you check the checkbox beside *Continuous capture*, EtherPeek captures packets until capture is stopped manually by the user, or by a stop trigger. When the window's capture buffer is full, EtherPeek discards packets to make room for new ones. Continuous capture is useful when, for example, you are waiting for a stop trigger event or notification.

Important! Continuous capture, with or without the save options, means that the Capture window continues to capture until it is stopped manually by the user or until a user-defined stop trigger is tripped.

To create a new Capture window that will continuously capture, re-using the buffer space:

1. Accept the default name for the new Capture window, or enter a new name in *Capture title*.
2. Enable *Continuous capture* by checking that checkbox.
3. Use the radio buttons in the *Buffer options* section to *Discard all packets when wrapping*, or *Discard oldest packets first (use ring buffer)*. The first option fills the buffer completely, then dumps the whole contents. The second option, in effect, writes over the older entries with newer ones.

Note: When you select the ring buffer option, once the *Memory usage* item in the Capture window header section reaches *100%*, it will stay there. In the ring buffer, new packets are continuously replacing ones captured earlier. The ring buffer, once full, remains full throughout the capture process.

4. Optionally, you can limit the amount of each captured packet to be saved. Please see "Using packet slicing" on page 57 for more details about this space-saving technique.
5. Accept the default *Buffer size: 16384 kilobytes*, or enter a new value for the buffer size.
6. When you have set all of the parameters, click **OK** to create the new Capture window.

Tip The processing time needed for continuous capture may cause EtherPeek to miss or drop some packets when its memory becomes full. This loss can be minimized by disabling unneeded program functions using the ***Performance*** view, by not scrolling during capture, by closing any non-essential windows, and by exiting any other programs that may be running in the background, even if they are idle.

Important! When you choose *Continuous Capture*, statistics for the Capture window will reflect all of the packets seen since it last began capturing. If you did not also choose *Save to disk*, the packets themselves may no longer be available after the buffer has wrapped (that is, dumped its packets and begun to refill).

Continuous capture saving to disk

When you choose *Continuous capture*, *Save to disk*, capture continues until it is stopped manually or by a stop trigger. Saving can continue until either a set amount of space is filled or until all available disk space at the save location is used up, or it can continue endlessly, overwriting older files with newer ones.

While saving continues, the program saves to a new file each time the buffer wraps. Each file is saved under a unique name, made up of the name you specify in the *File path*, plus a timestamp showing the time at which the file was saved. The format of the timestamp is `_YYYY-MM-DD HH.MM.SS.mmm`, which corresponds to year, month, day, hour, minutes, seconds, and milliseconds. EtherPeek will append sequence numbers to the timestamp, beginning with `000`, only when necessary to prevent identical file names.

By default, the timestamp reflects local time and is placed immediately after the file name you entered. You can specify the location of the timestamp within the file name by using the # character (the number sign, sometimes called the pound sign) as a token for the timestamp. To have the timestamp written in Universal Time Code (UTC) instead of local time, place the letter *z* immediately after the number sign. When UTC (formerly known as Greenwich Mean Time) is in use, the letter `z` will appear at the end of the timestamp in the form in which you entered it (upper or lower case).

To create a new Capture window that will continuously capture, saving all packets to disk:

1. Accept the default name for the new Capture window, or enter a new name in *Capture title*.
2. Enable *Continuous capture* by checking that checkbox.
3. Use the radio buttons in the *Buffer options* section to *Discard all packets when wrapping*, or *Discard oldest packets first (use ring buffer)*. The first option fills the buffer completely, then dumps the whole contents. The second, in effect, writes over the older entries with newer ones.
4. Check the checkbox beside *Save to disk*.
5. Use the *File path* text entry box to specify the base file name, the directory in which to store the file(s), and the file format to use in saving the buffer’s contents. As described at the beginning of this section, you can also specify the position and format of the timestamp added to individual file names. You can enter the text directly, determining the

file format by entering the correct file extension (*.pkt), or you can click the **…** (ellipsis) button to open a **Save As** dialog in which you can specify all these parameters. When you choose *Continuous capture*, *Save to disk*, you must save the files in the native *EtherPeek Packet File (*.pkt)* format.

Important! If you do not use one of the following options to limit the space allocated to saved files, captured traffic can continue to be saved until all available disk space at the specified *File path* is used up.

6. To limit the amount of space which can be taken up by the captured files, you have two choices: set the total disk space, or set the number of files. When you limit the total disk space, capture will continue, but no further files will be saved after this space is filled. When you limit the number of files, capture will continue, and older saved files from this Capture window will be overwritten with newer ones.
7. To set the total disk space to be occupied by the captured files, check the checkbox beside *Stop saving after... megabytes* and use the data entry box to specify the maximum amount of disk space you wish to use for the saved files.
8. Alternatively, you can limit the disk space used for captured files by setting an upper limit on the number of files to be kept. Check the checkbox beside *Keep most recent … files* and use the data entry box to specify the number of files. The Capture window writes a new file each time the buffer become full. Each file will be roughly the size you specify in *Buffer size*, below. When you limit the number of files, the oldest file is replaced by the newest. The total space taken up by saved files will be approximately equal to the buffer size times the number of files to keep.
9. Optionally, you can limit the amount of each captured packet to be saved. Please see “Using packet slicing” on page 57 for more details about this space-saving technique.
10. Accept the default *Buffer size: 16384 kilobytes*, or enter a new value for the buffer size.
11. When you have set all of the parameters, click **OK** to create the new Capture window.

Using packet slicing

Use packet slicing to capture only a portion of each packet instead of the whole packet. This saves space in the capture buffer. The packet slicing option is found in the ***General*** view of the **Capture Options** dialog. If you have not changed the default program settings, the **Capture Options** dialog is opened each time you create a new Capture window. To enable packet slicing for an existing Capture window, make it the active window and choose **Capture Options…** under the **Capture** menu to open the **Capture**

Options dialog, and click the *General* item in the navigation pane to open the ***General*** view (shown in Figure 4.2).

To enable packet slicing, check the checkbox labeled *Limit Each Packet to…. Bytes* and enter a number of bytes in the edit field. For example, if you enter "*132*," EtherPeek saves only the first 132 bytes of each packet it captures.

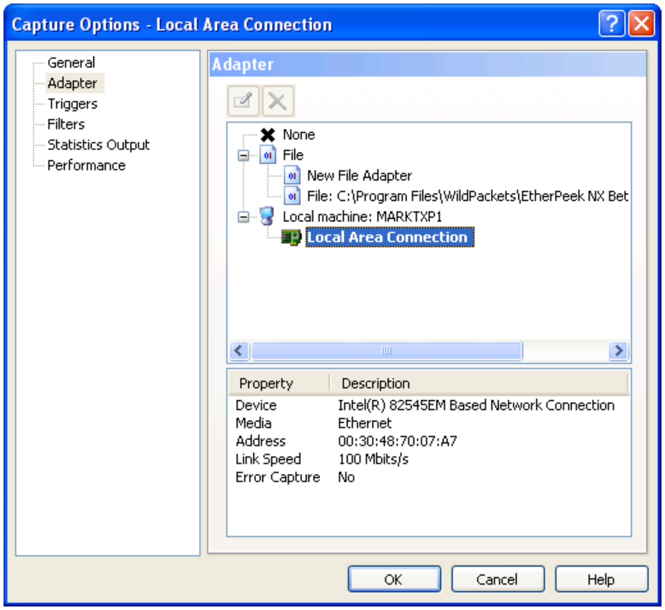
Note: You cannot enter a slice value of less than 14 bytes. In choosing a slice value, you should consider any filters and Name Table entries that you want to apply to your captured packets. Logical addresses and protocol fields both occur after the first 14 bytes. We recommend keeping the slice value at 128 bytes or greater. This typically will include all of the packet headers but little or no packet data. For more on the structure of Ethernet packets, please see Appendix A, "Packets and Protocols" on page A-3.

Capture filters are applied to packets before slicing occurs, so the slice value does not affect trigger events or filters enabled for a Capture window. However, any functions dependent on reading data from packets *after* they have been placed in the buffer *will* be affected. When used in the **Select** dialog, for example, Analysis Modules, filters, and other advanced functions read packets from the buffer, rather than directly from the network.

Capture options: adapter

Each Capture window must be assigned an adapter from which to capture network traffic. Multiple Capture windows can be assigned the same adapter, or each a different adapter, or any combination of shared or unique, so long as each Capture window has one valid adapter selected. You select an adapter in the ***Adapter*** view of the **Capture Options** dialog.

The ***Adapter*** view of the **Capture Options** dialog (Figure 4.3) displays all the network interface cards (NICs) installed on the local machine.



Property	Description
Device	Intel(R) 82545EM Based Network Connection
Media	Ethernet
Address	00:30:48:70:07:A7
Link Speed	100 Mbits/s
Error Capture	No

Figure 4.3 Adapter view of the Capture Options dialog

As an alternative to a locally installed NIC, the ***Adapter*** view lets you choose other sources of traffic as your adapter. You can, for example, choose a *File* as the adapter. If a remote (RMON) probe is network accessible, you can use that as the adapter.

To choose the adapter from which to capture packets, select a listed adapter or one of the alternate choices, then click **OK**.

When you select an adapter in the upper pane of the ***Adapter*** view, information about that adapter is presented in a table in the lower pane. Depending on the type of adapter selected, the lower pane will show the *Device* type, its *Media* type, *Address*, *Link speed*, and whether or not the adapter supports *Error Capture*.

To choose a file as the adapter, expand the *File* item and select a previously used file or choose *New File Adapter*. Double-click on the item, or highlight it and click the **OK** button to make your choice. If you select *New File Adapter*, you will be asked to specify the file, using a standard file **Open** dialog. When you choose an EtherPeek packet file (one of those in the Samples directory, for example), the program cycles through the traffic captured in that file, treating it as live traffic for purposes of this particular Capture window. By choosing a file as the adapter, you can simulate network conditions for training or open other packet traces without being connected to a network, or indeed without even having a supported NIC installed on your computer. EtherPeek remembers recently used file adapters and presents them in the ***Adapter*** view. To remove a file from the list, highlight the file and click the **Delete** button or right-click on the file and choose **Delete** from the context menu.

Note: If you have the separately purchased RMONGrabber Analysis Module installed, you will also see a heading for that Module and, under it, choices for a *New Remote Adapter*, or previously used remote adapters. For details about RMONGrabber, see Chapter 14, “RMONGrabber” on page 273.

Tip You can return to the ***Adapter*** view of the **Capture Options** dialog by double-clicking on the current adapter, shown in the status bar at the bottom right of the Capture window. Alternatively, make the Capture window the active window, and choose **Capture Options...** from the **Capture** menu to open the **Capture Options** dialog for that Capture window, then click the *Adapter* item in the navigation pane to open the ***Adapter*** view.

Network speed

EtherPeek auto-senses the network speed of the network adapter you select for its use, by default. You may want to expressly set the network speed in certain cases. The network speed applies to all uses of the selected adapter. For details on how to use this override function, please see “Network speed options” on page 17.

Default local adapter

The default choice in the ***Adapter*** view of the **Capture Options** dialog is the most recently selected adapter of any kind selected in the **Capture Options** dialog. (Creating a Capture window from a capture template does not affect the state of the **Capture Options** dialog, because the capture template bypasses the dialog, using its own stored options to create the new Capture window.) If there is no “most recently selected adapter” (you have never selected one, or the previously selected adapter is not found), the default adapter choice is the local NIC designated by EtherPeek as the “*default local adapter*.”

If you have only one supported NIC installed on the local machine, then that NIC is the default local adapter. If you have more than one NIC installed, then the default local adapter is the first supported NIC in the list of those shown under *Local machine* in the ***Adapter*** view.

Capture options: triggers

The ***Triggers*** view of the **Capture Options** dialog lets you control the start and or stop of capture in a particular Capture window by watching for a user-specified time, network, or capture event. For a complete discussion of trigger functions and the ***Triggers*** view of the **Capture Options** dialog, please see "Triggers" on page 224.

Capture options: filters

The ***Filters*** view of the **Capture Options** dialog shows a list of all available filters and allows you to choose which filters to enable for the current Capture window by checking the checkbox next to that filter's name. To choose how the filter(s) will be applied, use the **Accept Matching** or **Reject Matching** buttons at the top left of the ***Filters*** view. When you choose **Accept Matching**, only those packets which match the parameters of at least one of the enabled filters will be placed in the buffer. When you choose **Reject Matching**, only those packets which do not match any of the enabled filters will be entered in the buffer.

You can also set filters for a Capture window by using the ***Filters*** view of the Capture window itself.

Tip Use the ***Filters*** view of the **Capture Options** dialog to set the initial filter state for a new Capture window, or to build a capture template that includes filtering. Use the ***Filters*** view of the Capture window itself to make on-the-fly changes to filter settings. It's one click away, and the changes made there take effect immediately.

Double-click on any filter in any filter list in the program to open it in an **Edit Filter** dialog in which you can change or simply verify its parameters. For more about filters and how to use them, see Chapter 11, "Filters" on page 195.

To apply filters to packets already captured to a buffer, either in a Capture window or a Packet File window, use the **Select…** command from the **Edit** menu. For more on how to use filters to select captured packets, see "Select dialog: filters, analysis modules and more" on page 292.

Capture options: statistics output

Use the ***Statistics Output*** view of the **Capture Options** dialog to control the periodic output of statistics while the Capture window is open and capturing. Choose from several groups of statistics in a variety of report and file output formats. Save the files to any location at an interval you specify. A similar dialog view with similar choices is used to control the periodic output of Monitor statistics. A complete description of both views is provided in the “Statistics” chapter. Please see “Statistics output views” on page 173 for details.

Capture options: performance

Use the ***Performance*** view of the **Capture Options** dialog to selectively enable or disable individual program functions for a particular Capture window. The ***Performance*** view lets you streamline the resource utilization of a Capture window, optimizing it for a particular task. A similar dialog view with similar choices is used to control the performance of Monitor statistics, and a complete description of both views is provided in the “Installing and Configuring” chapter. Please see “Performance views” on page 25 for details.

Capture window views

The first time you launch EtherPeek, a new Capture window presents the ***Packets*** view in the view section by default. On subsequent start-ups, a new Capture window presents the view last seen in any Capture window. To move between views, click on the view tabs at the bottom of the Capture window.

When you click on a tab, it displays a different way of looking at the packets captured in this Capture window. The tab bar itself is not configurable and is the same for every Capture window. The appearance of individual views can be customized to a greater or lesser extent.

The views available in any Capture window, in order from left to right as they appear in the view tabs, are shown in Table 4.2.

Table 4.2 Views available in Capture windows

View	Description
Packets	This view shows a detailed list of all packets in the capture buffer, in the order they were received. You can choose which columns (what information) to display, as well as customize appearance. Packet Decodes are available.

Table 4.2 Views available in Capture windows (Continued)

View	Description
Nodes	This view shows traffic, in and/or out, aggregated by network node (physical address, logical address, and/or symbolic name). Display by any of several node types (*Physical*, *IP*, *IPX*, or others) or display all types ranged under physical addresses (*Hierarchical*). Considerable customization of information to be displayed is possible: choose from many columns, or, in Hierarchical view show packets received, packets sent, or both. Can limit display to highest traffic nodes. Customized appearance is available. Detailed views are available.
Protocols	This view shows traffic aggregated by protocol and subprotocol using ProtoSpecs technology. It has a customizable appearance, and detailed views are available.
Summary	This view shows a synopsis of the activity on the network since capture began: number of nodes, traffic volumes by type, and other summary statistics supplied by EtherPeek and Analysis Modules and Expert. Also shows Driver Statistics.
Graphs	This view presents a variety of graphs displaying statistics from the current window in real time. All graphs, including the default set, are editable and configurable. (Default graphs include equivalents to the **Size** and **History** graphs found in Monitor statistics, for example.) You can add to, delete, rearrange, create, edit, export, and import graphs of nearly any form, each based on single or multiple statistics from the current Capture window.
Log	This view logs events such as the start of capture and shows messages, primarily from any enabled Analysis Modules.
Conversations **(EtherPeek standard only)**	Unique to EtherPeek standard, this view shows statistics for traffic arranged by conversations between pairs of nodes, as well as data about the individual nodes in each conversation.
Expert **(EtherPeek NX only)**	Unique to EtherPeek NX, this view shows conversations, including detailed expert analysis of events and potential problems, as identified in the Expert EventFinder settings.

Table 4.2 **Views available in Capture windows (Continued)**

View	Description
Peer Map **(EtherPeek NX only)**	Unique to EtherPeek NX, this view shows a customizable graphical view of communications patterns between partners, based on the traffic in the current window.
Filters	This view shows a list of all available filters, showing which are enabled for this window. Enable and disable filters for the window in this view.

The rest of this section will take each of these views in order and describe their default appearance, any detailed views available, and any customizations that can be made to their appearance.

Packets view

The ***Packets*** view has three panes: the Packet List, Decode, and Hex view panes. You can display one, two, or all three of these panes in the ***Packets*** view at any time.

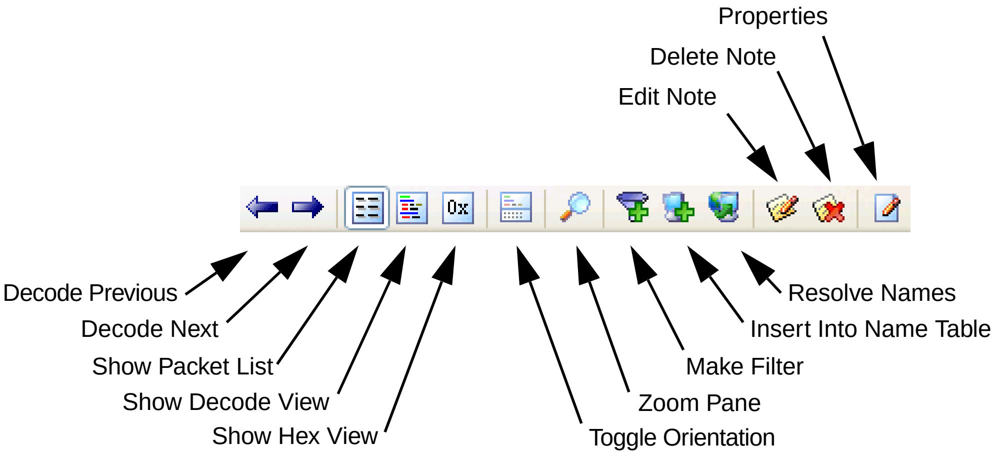
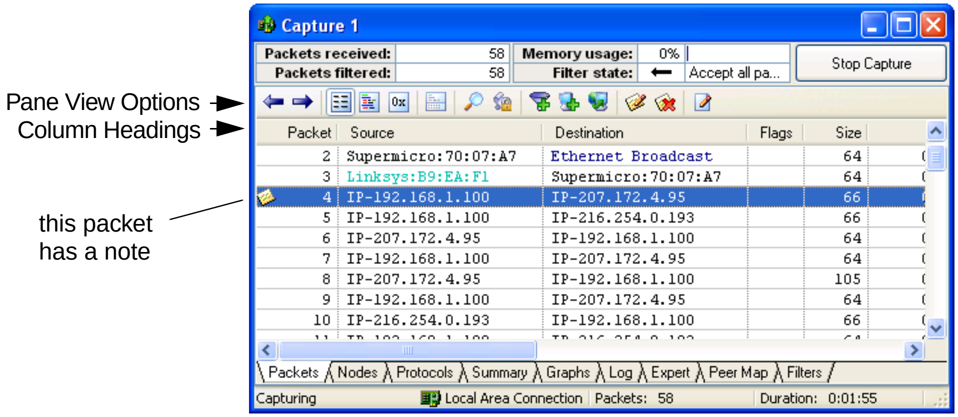


Figure 4.4 Detail of Pane View Options buttons in the Packets view

Use the Pane View Options buttons at the top of the ***Packets*** view (shown in detail in Figure 4.4) to select which panes will be visible. You can choose to **Show Packet List**, **Show Decode View** pane, and/or **Show Hex View** pane by toggling the appropriate button(s). When the Decode and Hex panes are both open, you can click the **Toggle Orientation** button to switch between having the Decode pane above and the Hex pane below, or the Decode pane at left and the Hex pane at right. When multiple panes are open, you can use the **Zoom Pane** button (or the **F4** function key) to toggle between viewing all panes (no zoom) or only the active pane (zoom). The active pane is the one in which you have highlighted some item.

The left and right arrow buttons step through the packets visible in the packet list backwards or forwards, respectively. As each packet in the Packet List is highlighted, the Decode and/or Hex view of that packet will appear in those panes, if they are open. You can use the function keys **F7** (previous) and **F8** (next), or use the keyboard combinations **Alt + left arrow** (previous) and **Alt + right arrow** (next) to accomplish the same thing.



Packets received: 58
Packets filtered: 58
Memory usage: 0%
Filter state: Accept all pa...

Packet	Source	Destination	Flags	Size
2	Supermicro:70:07:A7	Ethernet Broadcast		64
3	Linksys:B9:EA:F1	Supermicro:70:07:A7		64
4	IP-192.168.1.100	IP-207.172.4.95		66
5	IP-192.168.1.100	IP-216.254.0.193		66
6	IP-207.172.4.95	IP-192.168.1.100		64
7	IP-192.168.1.100	IP-207.172.4.95		64
8	IP-207.172.4.95	IP-192.168.1.100		105
9	IP-192.168.1.100	IP-207.172.4.95		64
10	IP-216.254.0.193	IP-192.168.1.100		66

Capturing | Local Area Connection | Packets: 58 | Duration: 0:01:55

Figure 4.5 Packet List in Packets view of a Capture window, with note

The Packet List pane is a table with user-configurable columns showing information about each packet on a single line. The next section, Packet list columns, describes each of the columns which can be used in the Packet List pane of the ***Packets*** view. For instructions on how to add or delete columns in a particular Packet List and how to change their order, please see “Customizing columns in the packet list” on page 76.

The Decode pane of the ***Packets*** view shows the information contained in a single packet, decoded and interpreted. The Hex view pane shows the information contained in a single packet as raw hexadecimal values on the left, and the same data expressed as ASCII characters on the right. The Decode and Hex panes of the ***Packets*** view are identical to the same views in the **Packet Decode** window. For a detailed description of the Decode and Hex view panes and how to use them, please see “The packet decode window” on page 298.

Tip EtherPeek produces live decodes of packets as they are captured, when either or both of the Decode and Hex panes are open and **Auto Scroll** is active. As each packet is captured, these panes are updated in real time with that packet’s information. The views are refreshed with the most recently captured packet, as long as **Auto Scroll** is enabled.

Packet list columns

Each column in the Packet List pane of the ***Packets*** view contains a particular type of information about the packet or a piece of information contained in the packet. Table 4.3 shows the columns available for use in the Packet List pane. Columns are included or excluded for a particular Capture window using the **Packet List Options** dialog. To open the **Packet List Options** dialog for a particular Packet List, click anywhere in the column headers of the list, or right-click in the display and choose **Packet List Options…** from the context menu.

The columns present by default when you use EtherPeek for the first time are shown in Table 4.3 with an **X** in the **Default** column. You can restore the default selection of columns at any time by clicking the **Defaults** button in the ***Columns*** view of the **Packet List Options** dialog.

Table 4.3 **Packet List Options columns, showing defaults**

Default	Column	Description
X	***Packet***	This column displays a packet number as determined by the time-sequential order in which the packets were captured.
X	***Source***	This column displays the source address. Depending upon the choice under **Display Format** in the **View** menu, this address may be a physical Ethernet address, a higher-level, logical address such as IP or AppleTalk, or a symbolic name.

Table 4.3 Packet List Options columns, showing defaults (Continued)

Default	Column	Description
	Source Logical	This column shows the logical address of the packet's source. Unlike the default ***Source*** column, this column's display is unaffected by any choice you make in **Display Format** under the **View** menu. This allows you to show different formats for a packet's source on a single line.
	Source Physical	This column shows the physical address of the packet's source. Unlike the default ***Source*** column, this column's display is unaffected by any choice you make in **Display Format** under the **View** menu. This allows you to show different formats for a packet's source on a single line.
	Source Port	This column displays the source port or socket, if any, in the notation appropriate for that protocol. For a definition of ports and sockets, please see "Ports and sockets" on page A-18.
X	***Destination***	This column displays the destination address. Depending upon the choice under **Display Format** in the **View** menu, this address may be a physical Ethernet address, a higher-level, logical address such as IP or AppleTalk, or a symbolic name.
	Destination Logical	This column shows the logical address of the packet's destination. Unlike the default ***Destination*** column, this column's display is unaffected by any choice you make in **Display Format** under the **View** menu. This allows you to show different formats for a packet's destination on a single line.
	Destination Physical	This column shows the physical address of the packet's destination. Unlike the default ***Destination*** column, this column's display is unaffected by any choice you make in **Display Format** under the **View** menu. This allows you to show different formats for a packet's destination on a single line.
	Destination Port	This column displays the destination port or socket, if any, in the notation appropriate for that protocol. For a definition of ports and sockets, please see "Ports and sockets" on page A-18.

Table 4.3 Packet List Options columns, showing defaults (Continued)

Default	Column	Description
X	***Flags***	This column contains flag characters indicating that a packet is in the IEEE 802.3 format, or is a particular type of error packet, or is a trigger packet. The characters used for flags are assignable using the ***Flags*** view of the **Packet List Options** dialog, available by left-clicking in the column headers of the Packet List pane of the ***Packets*** view of any Capture window or Packet File window. The default assignments are shown in Table 4.4 below.
X	***Size***	This column displays the length of the packet in bytes, including the packet header, FCS bytes, and any padding.
	IP Length	This column displays the total length of the IP datagram, in bytes. It includes the length of the IP header and data.
	IP ID	This column displays the IP ID (Identifier) of the packet. The IP ID uniquely identifies each IP datagram sent by a host. It normally increments by one each time a datagram is sent.
	Date	This column shows the date the packet was received.
	Absolute Time	This column displays the time-stamp assigned to each packet as the actual time of capture, according to the system clock of the computer on which EtherPeek is running (see note). Use the ***Format*** view of the **Packet List Options** dialog to set the display units for all time-stamps to milliseconds, microseconds, or nanoseconds.
X	***Delta Time***	This column shows the time-stamp of each packet as the elapsed time since the capture of the previous visible packet. (That is, if packets are hidden, the time shown is relative only to the previous *visible* packet.) Use the ***Format*** view of the **Packet List Options** dialog to set the display units for all time-stamps to milliseconds, microseconds, or nanoseconds.

Table 4.3 **Packet List Options columns, showing defaults (Continued)**

Default	Column	Description
	Relative Time	This column displays the time-stamp of each packet as the elapsed time since the start of the current EtherPeek session. You can set a particular packet as the “zero” time for all items in the ***Relative Time*** column. Packets captured before will show negative values, those after, positive values, all relative to the new zero time. To set a packet as the zero time by setting it as the Relative Packet, right-click on the packet’s line and choose **Set Relative Packet** from the context menu. Use the ***Format*** view of the **Packet List Options** dialog to set the display units for all time-stamps to milliseconds, microseconds, or nanoseconds.
	Cumulative Bytes	If no Relative Packet is set, this column shows the total bytes represented by all the visible packets from the first packet in the list to the current packet, inclusive. If you have set a Relative Packet, this column shows the total bytes from the Relative Packet to the current packet, inclusive. To set a packet as the Relative Packet, right-click on the packet’s line and choose **Set Relative Packet** from the context menu.
X	***Protocol***	This column displays the protocol type of the packet. This may be shown as an LSAP value, a SNAP value, or a ProtoSpec. If you have established a symbolic name for a protocol otherwise unknown to ProtoSpecs, that name may be taken from the Name Table and displayed here.
	Filter	This column displays the name of the filter that allowed the packet to be entered into the capture buffer.
X	***Summary***	This column lists any information provided about the packet by enabled Analysis Modules.
	Analysis Module Name	This column displays the name of the Analysis Module that supplied the information on that packet that is displayed in the ***Summary*** column.
	Note	This column shows the full text of any user-entered note associated with the packet.

Table 4.3 Packet List Options columns, showing defaults (Continued)

Default	Column	Description
X (EtherPeek NX only)	***Expert***	Presents data collected about the packet by the Expert Analysis tools. Typically, this is a short description of the type of problem found in the packet or a description of the event, and may include a measurement (such as response time since another named packet) which caused this packet to be identified as an event.
	Decode	This column displays a portion of the information present in the ***Decode*** view of the packet, when that information matches the most recently highlighted part of any decode of any packet in the Capture window. It shows the same part of the decode for every packet that contains the selected type of information. For example, if you highlight the *Ethernet Header* section of the ***Decode*** view of any packet in the Capture window, the column header of the ***Decode*** column will change to ***Ethernet Header*** and the Ethernet header information for each packet (destination address, source address, and the Type / Length field) will be shown in this column, just as it is in the ***Decode*** view. If you highlight an element of the decode that is not present in all other packets, the packets without the corresponding element will show nothing in the ***Decode*** column. Only those packets with a matching class of information will display data in the ***Decode*** column.

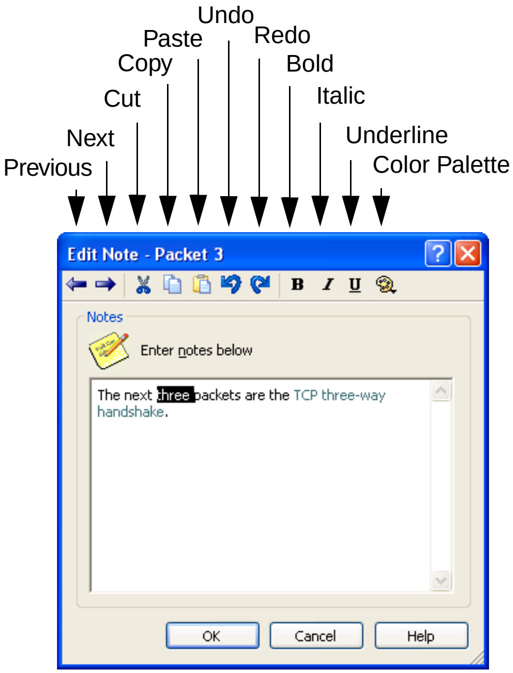
In order to see the right-most columns in the Packet List, you may need to use the scroll bars or resize the Capture window.

Making notes on packets and packet files

The Notes tools let you make notes on individual packets within the packet list, and the annotations will be preserved when you save the file as an EtherPeek packet file (*.pkt) or compressed packet file (*.wpz). In addition, you can make notes on the packet file as a whole by adding comments to the **Properties** dialog. These too will be preserved when the file is saved in either EtherPeek format.

To make a note on an individual packet, select the packet in either the ***Packets*** view or in its own **Packet Decode** window and click the **Edit Note** button in the header section of the view or window. (See Figure 4.4 on page 64 for a detail of the header section of the ***Packets*** view of a Capture window, with all the buttons labeled.) This brings up the **Edit Note** dialog (Figure 4.6). An icon representing a note appears in the ***Packet*** column (the column showing packet numbers) of the ***Packets*** view for any packet with an associated note. The optional ***Note*** column in the ***Packets*** view shows the full text of any note. This makes it easy to copy, save, or print any notes in a format that places the note on the same line as the packet data to which it refers. You can also use the **Find Pattern** dialog (available from the **Edit** menu) to search for text strings, and limit your search to the ***Note*** column.

Tip You can use standard keystroke combinations (**Ctrl + C**, **Ctrl +V**) to copy and paste tab-delimited text directly from the ***Packets*** view.



Enter notes below

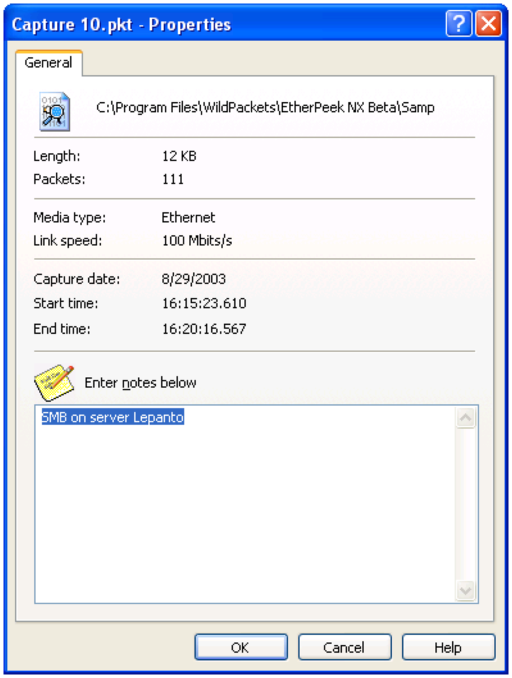
The next three packets are the TCP three-way handshake.

Figure 4.6 Edit Note dialog

To view or edit the contents of a note, highlight the packet to which it belongs (or open the packet in the **Packet Decode** window) and click the **Edit Note** button to open the **Edit Note** dialog. In addition to a full range of editing features, the **Edit Note** dialog allows you to step through the packets in the current selection. Use the **Next** and **Previous** buttons to steps forward or backward through the currently selected packets, in packet number order. You can keep the **Edit Note** dialog open, allowing you to review any existing notes or add notes to any packet in the selection.

To delete one or more notes, highlight the packet(s) to which they belong and click the **Delete Note** button.

You can also make a note on the contents of a Capture window or Packet File window as a whole by entering text in the **Properties** dialog. Click the **Properties** button in the header section of the ***Packets*** view to open the **Properties** dialog (Figure 4.7). In addition to providing a container for notes, the **Properties** dialog presents summary information, such as file size, number of packets, network type, and capture date and times. This information, along with any notes you have entered, will be saved and associated with the saved packet file.



C:\Program Files\WildPackets\EtherPeek NX Beta\Samp

Length:	12 KB
Packets:	111
Media type:	Ethernet
Link speed:	100 Mbits/s
Capture date:	8/29/2003
Start time:	16:15:23.610
End time:	16:20:16.567

Enter notes below

SMB on server Lepanto

Figure 4.7 Properties dialog for a Packet File window

Statistical display views

Capture windows offer four different displays of statistics: ***Node***, ***Protocol***, ***Summary***, and ***Graphs***. In addition, Capture windows in EtherPeek standard offer the ***Conversations*** view.

Statistics in Capture windows are calculated based on all the packets that have been accepted to the buffer since capture was initiated. If continuous capture is enabled and the buffer has wrapped, this may mean that the statistics are based on many more packets than are present in the buffer. If you use the Hide functions to alter the apparent contents of the buffer, it will force a recalculation of all the statistics to match the changed visible contents, and you will lose all accumulated data.

The ***Graphs*** and ***Conversations*** views, unlike the other statistical views, have no direct equivalents in Monitor statistics. The other statistics views of a Capture window or a Packet File window (the ***Node***, ***Protocol***, and ***Summary***, views) are substantially the same as the Monitor statistics windows of the same names. Please see Chapter 9, “Statistics” on page 143, for a detailed discussion of each of these types of statistical displays. For notes on important differences between Monitor statistics and statistics in Capture windows and Packet File windows, see “Statistics in capture windows” on page 166. For a complete discussion of the ***Graphs*** view, see Chapter 10, “Graphs of Monitor and Capture Statistics” on page 181. For a complete discussion of the ***Conversations*** view, see “Conversations” on page 168. For information on saving and printing statistics from these windows, see “Saving reports from capture windows” on page 172. You can also save statistics from Capture windows at set intervals by using the ***Statistics Output*** view of the **Capture Options** dialog. Please see “Statistics output views” on page 173 for details.

Graphs view

The ***Graphs*** view presents a variety of graphs displaying statistics from the current window in real time. All graphs, including the default set, are editable and configurable. (Default graphs include equivalents to the **Size** and **History** graphs found in Monitor statistics, for example.) You can add to, delete, rearrange, create, edit, export, and import graphs of nearly any form, each based on single or multiple statistics from the current Capture window.

For a complete discussion of the ***Graphs*** view and all its functions, please see Chapter 10, “Graphs of Monitor and Capture Statistics” on page 181.

Log view

The ***Log*** view contains information, primarily generated by Analysis Modules, about the packets in the buffer of the Capture window or Packet File window. The ***Log*** view also notes such things as start capture times. Log messages are not saved with saved packet files, but rather are regenerated each time the buffer is renewed—by opening the file or by hiding or unhiding packets, for example.

For more about the Log view, please see “Log views of capture and packet file windows” on page 141. For more about Analysis Modules and the types of messages they can write to the ***Log*** view, please see Chapter 13, “Analysis Modules” on page 247.

Conversations view

Unique to EtherPeek standard, the ***Conversations*** view groups the traffic in a Capture window or Packet File window into conversations between pairs of nodes. The ***Conversations*** view presents information about each conversation in the upper Conversations pane and additional information about each partner in the lower Naming and Statistics pane.

For more about the ***Conversations*** view and how to use it, please see “Conversations” on page 168.

Expert view

Unique to EtherPeek NX, the ***Expert*** view provides expert analysis of delay, throughput and a wide variety of network events in a conversation-centered view of traffic in a Capture window or Packet File window.

For more about the ***Expert*** view and the expert analysis it provides, please see Chapter 5, “Expert View and Expert EventFinder” on page 101.

Peer Map view

Unique to EtherPeek NX, the ***Peer Map*** view is a powerful tool for visualizing network traffic in a Packet File window or Capture window. The Peer Map displays the nodes in the current window around an elongated ellipse. Lines between communicating nodes (peers) represent the traffic. The line weight shows the volume of traffic between each pair of communicating peer nodes. The line color shows the protocol in use between each pair of nodes.

Like all other views of a Capture window or Packet File window, the ***Peer Map*** view is based on the packets that are visible in the ***Packets*** view. The ***Peer Map*** also contains its own independent tools to control the display of nodes and types of network traffic. This lets you quickly create a picture of all the traffic in a particular protocol, for example, or all the nodes sending or receiving multicast traffic.

For a detailed description of how to use the Peer Map and all the functions of the ***Peer Map*** view, please see Chapter 6, “Peer Map” on page 119.

Filters view

The ***Filters*** view of a Capture window shows a list of all available filters and allows you to choose which filters to enable for that Capture window by checking the checkbox next to that filter’s name. To choose how the filter(s) will be applied, use the **Accept Matching** or **Reject Matching** buttons at the top left of the ***Filters*** view. When you choose **Accept Matching**, only those packets which match the parameters of at least one of the enabled filters will be placed in the buffer. When you choose **Reject Matching**, only those packets which do not match any of the enabled filters will be entered in the buffer.

By double-clicking on any filter, you can open it in an **Edit Filter** dialog and change or simply verify its parameters. For more about filters and how to use them, see Chapter 11, “Filters” on page 195.

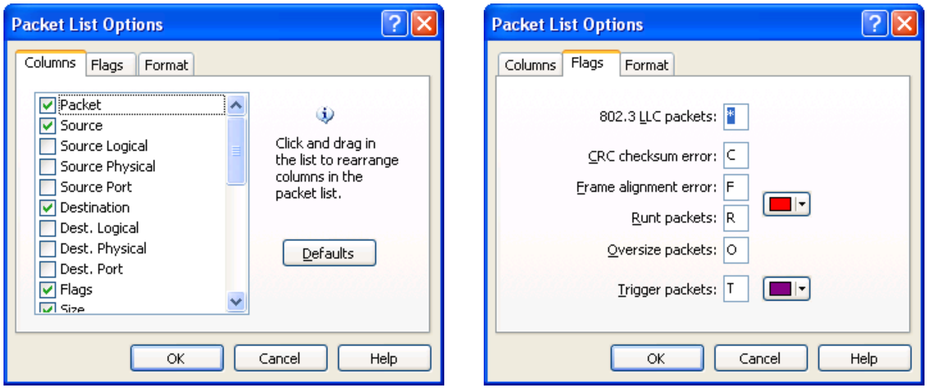
The ***Filters*** view only exists in Capture windows. To apply filters to packets already captured to a buffer, either in a Capture window or a Packet File window, use the **Select…** command from the **Edit** menu. For more on how to use filters to select captured packets, see “Select dialog: filters, analysis modules and more” on page 292.

Customizing views

You can customize the way in which certain types of information are displayed in the Packet List pane of the ***Packets*** view of Capture windows and Packet File windows using the **Packet List Options** dialog. Other data display characteristics can be customized in a way that affects the display of certain data in all windows, including Monitor statistics. These more general display parameters include the ***Fonts*** view of the **Options** dialog (available under the **Tools** menu) and the **Display Format** and **Color** submenus under the **View** menu. Each of these formatting tools is discussed in this section.

Packet list view options

The column content, color and format in which packet information is displayed in Capture windows and Packet File windows can be customized in the **Packet List Options** dialog. To open the **Packet List Options** dialog for a particular Packet List, click anywhere in the column headers of the list, or right-click in the display and choose **Packet List Options…** from the context menu.



Packet List Options

Columns | Flags | Format

☑ Packet
☑ Source
□ Source Logical
□ Source Physical
□ Source Port
☑ Destination
□ Dest. Logical
□ Dest. Physical
□ Dest. Port
☑ Flags
☑ Size

Click and drag in the list to rearrange columns in the packet list.

Defaults

OK | Cancel | Help

Packet List Options

Columns | Flags | Format

802.3 LLC packets: *
CRC checksum error: C
Frame alignment error: F 🟥
Runt packets: R
Oversize packets: O
Trigger packets: T 🟪

OK | Cancel | Help

Figure 4.8 Packet List Options dialog, Columns and Flags views

Customizing columns in the packet list

You can customize the information to be displayed about each packet in the Packet List pane of the ***Packets*** view by adding, deleting, or rearranging the columns. You can, for example, keep an inventory of the devices on a network segment which shows the physical, logical, and symbolic names for each device by creating a customized Capture window with only these columns.

Left-click anywhere in the Packet List column headings to bring up the **Packet List Options** dialog. Choose the ***Columns*** view (by clicking the labeled tab) to display a list of available column types. The columns currently used in the ***Packets*** view of the active Capture window will have a ✔ checkmark in the checkbox next to their entries in the scrollable list. Uncheck any you wish to remove and check any you wish to add to the Packet List of the active window. A descriptive list of all available column types is shown in Table 4.3.

When you save a packet file as a tab or comma-delimited file, the information is saved in the same column order as appears in the Packet List pane of the ***Packets*** view. You can rearrange the order of the columns in the Packet List pane using drag and drop in either the Packet List itself or in the list of columns shown in the ***Columns*** view of the **Packet List Options** dialog. To use drag and drop in the Packet List itself, click in the heading of the column you wish to move, and hold down the mouse button. You can drag the heading to any other position and drop it there by releasing the mouse button. You can use the same technique to rearrange the order in the list of column types in the ***Columns*** view of the **Packet List Options** dialog, but in this case you must also click **OK** in the **Packet List Options** dialog for the changes to take effect.

Packet list flag options

The ***Flags*** view of the **Packet List Options** dialog defines both the flag character and the color associated with flagged packets. These are: *802.3 LLC Packets*, error packets (*CRC Checksum Error*, *Frame Alignment Error*, *Runt Packets*, or *Oversize Packets*), and *Trigger Packets*. You can use the dialog to assign a flag character to any of these packet types, or to assign a color to all error packets or to trigger packets.

Table 4.4 Flag characters and colors, default values

Flagged Packet Type	Character	Color
802.3 LLC packets (all packets in the 802.3 format)	*	none
CRC checksum error (corrupt data)	*C*	Error color (red is default)
Frame alignment error (corrupt data)	*F*	Error color (red is default)
Runt packet (length < 64 bytes)	*R*	Error color (red is default)
Oversize packet (length > 1518 bytes)	*O*	Error color (red is default)
Trigger packets (match an enabled trigger)	*T*	Trigger color (purple is default)

Note: The 802.3 LLC packets cannot be associated with any color.

To assign a character, simply highlight the existing character and type over it. To assign a color, click on the color swatch to open a palette of alternative colors. If you choose **Flag** in **Color** under the **View** menu, the color associated with the packet type will be used for all information displayed about that type of packet.

Note: For the meaning of each error packet type, please see “Error types and error packets” on page 159.

Packet list format options

The ***Format*** view of the **Packet List Options** dialog allows you to set the *Time-Stamp format* to use *Milliseconds*, *Microseconds,* or *Nanoseconds* as its units, by choosing one of these from the drop-down list.

By checking the appropriate checkbox in the ***Format*** view, you can choose to *Show an ellipsis for truncated items* in Packet List columns and/or *Prefix addresses and protocols with the type* appropriate to them and/or *Show port names*. You can also choose to *Use protocol color for summary column* by checking that checkbox. When checked, this option displays the information provided by Analysis Modules and shown in the ***Summary*** column in the color assigned to the relevant protocol by ProtoSpecs.

Tip The same font is used throughout the program to display information discovered by EtherPeek. This font is used in the packet list and all other views of Capture windows, Packet File windows, and Monitor statistics. You can globally change this font in the ***Fonts*** view of the **Options** dialog. Please see “Fonts view” on page 21 for details.

Node display format options

The **Display Format** submenu is available from the **View** menu. At a minimum, packets are identified by the **Physical Address** of the source and destination nodes. If you choose **Name Table Entry** and there is a Name Table entry for a node, EtherPeek will use the node’s name instead of its address whenever it encounters packets to or from that node. The **Logical Address** item causes EtherPeek to show logical instead of physical addresses, wherever logical addresses are available. Before a packet is displayed, EtherPeek checks its protocol type. If it is one of the types that EtherPeek recognizes, it can replace the physical address with its logical address according to the protocol type.

Color display options

The **Color** submenu of the **View** menu determines how colors *already assigned in other dialogs* will be used in displaying packets, as well as node and conversation statistics in all displays. There are four sources of color assignments for elements of network traffic in EtherPeek:

- The ***Flags*** view of the **Packet List Options** dialog (available by left-clicking anywhere in the Packet List pane headers) determines the color associated with error or trigger packets. (These choices are not meaningful for statistics displays.)
- The **Edit Name** dialog in the **Name Table** can set the color for packets associated with a particular address (node), port, or protocol.
- ProtoSpecs assigns colors to all the protocols it can identify. ProtoSpecs color choices cannot be overridden.
- The **Edit Filter** dialog can set the color for any filter you create or edit. (These choices are not meaningful for statistics displays.)

The **Color** sub-menu of the **View** menu uses the color information from these other sources, and applies it to the display of packets in ***Packets*** view in the ways described in the table below for each of the available choices. A ✔ checkmark appears beside the enabled choice.

Table 4.5 View menu > Color menu choice items

Color Menu Item	Description
Source	This choice causes packets sent out by a particular node to be displayed in the color assigned to that node in the Name Table.
Destination	This choice causes packets destined for a particular node to be displayed in the color assigned to that node in the Name Table.
Protocol	This choice causes packets to be displayed in the color assigned to protocols by ProtoSpecs. If ProtoSpecs cannot identify the protocol *and* the protocol is listed in the Name Table and has a color assigned there, then the color assigned in the Name Table will be used.

Table 4.5 **View menu > Color menu choice items (Continued)**

Color Menu Item	Description
Filter	This choice causes packets that are captured through a filter to be displayed in the color assigned to that filter in the **Edit Filter** dialog. (This choice is not meaningful for statistics displays.)
Flag	This choice causes packets that have been flagged to be displayed in the color assigned to trigger, error, and other flagged packet types in the **Packet List Options** dialog. (This choice is not meaningful for statistics displays.)
Independent	This choice causes each of the above items to display in its own assigned color.
No Color	This choice turns off all color coding.

Scroll during capture

When **Auto Scroll** is enabled, the most recently captured packet will always appear as the last packet in the Packet List pane of the ***Packets*** view. Use the **Auto Scroll** button at the top of the ***Packets*** view of the Capture window to toggle this feature.

When this option is disabled, the Packet List pane does not change as packets are added to the buffer. The window does change however when continuous capture is enabled. The scroll bar at the right of the pane will move to show that it is keeping the same relative position in the whole buffer. As the buffer fills, the scroll bar will move up. If you chose to *Discard all packets when wrapping*, the scroll bar will move to the top of the display the first time the buffer is emptied, then stay there. If you chose *Discard oldest packets first (use ring buffer)*, the scroll bar will move up and down, following the relative position of the initial “end of file” marker.

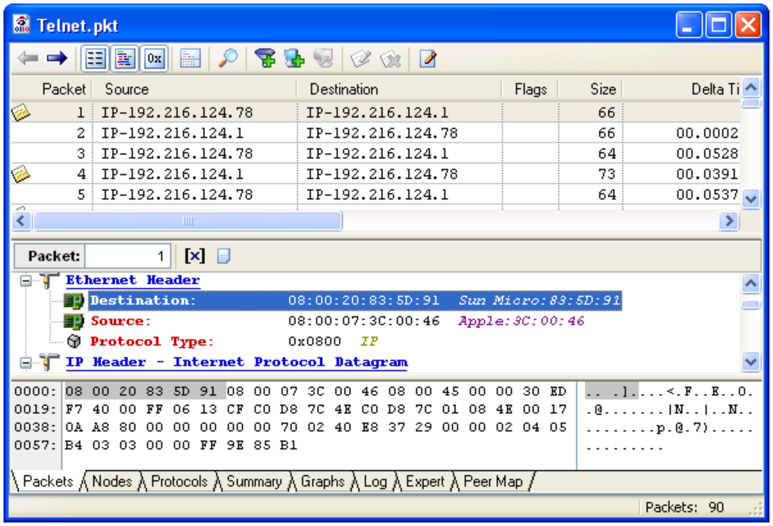
By default, when you stop the Auto Scroll function, you must restart it again manually. To have Auto Scroll resume automatically, choose **Options…** from the **Tools** menu to open the **Options** dialog. In the ***Workspace*** view, click the checkbox beside *Resume auto-scroll in the packet lists after … seconds*, and enter the number of seconds. Auto scroll does use some processor resources. For this reason, the auto-scroll resume feature is not enabled by default.

Packet file windows

Packet files in EtherPeek format are loaded into their own individual Packet File windows.

A Packet File window is very similar in structure, function and layout to a Capture window. With the important exceptions noted below, everything described in this chapter about Capture windows is also true of Packet File windows.

The differences between the two types of windows are due to their differences in function. There is no capture in a Packet File window and no loading of saved packets in a Capture window. The title of a Packet File window shows the name of the loaded file. The header section of the Packet File window shows no information relating to capture (no Progress section). It does, however, show a value for *Packets* (total packets in the file) in the window status bar. Figure 4.9 below shows the ***Packets*** view of a Packet File window with the Packet List, Decode and Hex panes all visible.



Packet	Source	Destination	Flags	Size	Delta Ti
1	IP-192.216.124.78	IP-192.216.124.1		66	
2	IP-192.216.124.1	IP-192.216.124.78		66	00.0002
3	IP-192.216.124.78	IP-192.216.124.1		64	00.0528
4	IP-192.216.124.1	IP-192.216.124.78		73	00.0391
5	IP-192.216.124.78	IP-192.216.124.1		64	00.0537

```
Ethernet Header
    Destination:      08:00:20:83:5D:91  Sun Micro:83:5D:91
    Source:           08:00:07:3C:00:46  Apple:3C:00:46
    Protocol Type:    0x0800  IP
IP Header - Internet Protocol Datagram
```

```
0000: 08 00 20 83 5D 91 08 00 07 3C 00 46 08 00 45 00 00 30 ED  .. .].....<.F..E..0.
0019: F7 40 00 FF 06 13 CF C0 D8 7C 4E C0 D8 7C 01 08 4E 00 17  .@.......|N..|..N..
0038: 0A A8 80 00 00 00 00 00 70 02 40 E8 37 29 00 00 02 04 05  ........p.@.7).....
0057: B4 03 03 00 00 FF 9E 85 B1                                .........
```

Figure 4.9 EtherPeek Packet File window, 3-pane view

There are, of course, no triggers or use of filters for capture in a Packet File window, but you can use filters as tests for selecting packets, using the **Select** dialog. (Please see “Select dialog: filters, analysis modules and more” on page 292.) You can include a ***Filter*** column in the Packet List pane of the ***Packets*** view, even though no filter information is saved with EtherPeek packet files. If a Packet File window has this column and you make a selection in the **Select** dialog using a filter match as the test, the name of the filter that allowed each packet to be selected will show up in the ***Filter*** column.

Note: Statistics in a Packet File window are calculated based on packets visible in the buffer. If you hide or unhide packets (using the commands from the **Edit** menu), it will force a recalculation of the statistics to reflect the changed visible contents of the buffer.

Saving, loading and printing captured packets

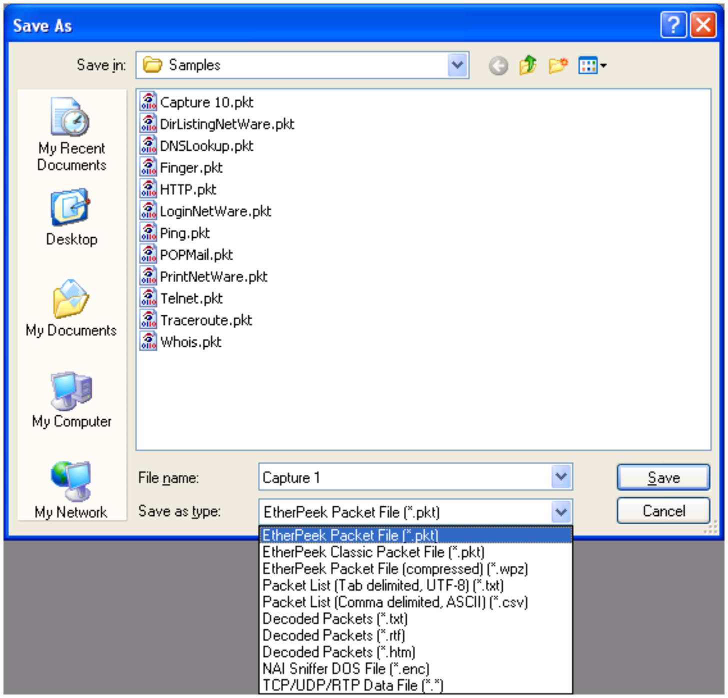
To quickly save all or a part of the information in the ***Packets*** view as tab-delimited text, you can copy and paste using standard keyboard combinations (**Ctrl + C**, **Ctrl + V**).

You can save the packets captured during a EtherPeek session for later examination and comparison. To save all captured packets, choose the **Save All Packets…** command in the **File** menu or type **Ctrl + S**. The **Save All Packets…** command saves all packets currently visible in the active window, whether selected or not. Any hidden packets will *not* be saved.

To save only certain packets, select the ones you want, then choose the **Save Selected Packets…** command in the **File** menu. **Save Selected Packets…** saves only the packets currently highlighted in the active Capture window or Packet File window.

For more information about selecting packets, please see Chapter 15, “Post-capture Analysis” on page 283.

Choosing either of the above commands opens the **Save As** dialog:



Save in: Samples

File name: Capture 1

Save as type: EtherPeek Packet File (*.pkt)

Figure 4.10 Saving packets as lists or as decoded packets

Save file formats

In the **Save As** dialog opened by choosing **Save All Packets** or **Save Selected Packets** from the **File** menu, you can assign a file name and choose among nine file formats. An additional choice, *TCP/UDP/RTP Data File*, is available only when you have opened the **Save As** dialog by choosing **Save Selected Packets**. The formats are:

- *EtherPeek Packet File (*.pkt)*, the default choice, saves to the native EtherPeek file format, with a *.pkt extension. This is also the native EtherPeek format.
- *EtherPeek Classic Packet File (*.pkt)*, saves to the older version of the EtherPeek packet file format, with a *.pkt extension. Use this format to make files readable by older programs, such as older versions of EtherPeek standard (5.0 and earlier), EtherPeek NX (2.0 and earlier), NetSense, and ProConvert.

- *EtherPeek Packet File (compressed) (*.wpz)*, saves to the native EtherPeek file format, but using file compression to save disk space. Uses a *.wpz extension. This is also a native EtherPeek format.
- *Packet List (Tab delimited, UTF-8) (*.txt)* creates a tab-delimited text file (*.txt), in UTF-8 encoding, containing only the information visible in the Packet List pane of the ***Packets*** view of the active Capture window or Packet File window. For a complete description of this option, please read the section entitled “Saving as packet list (comma- or tab-delimited)” below.
- *Packet List (Comma delimited, ASCII) (*.csv)* creates a comma-delimited text file (*.csv), in ASCII encoding, containing only the information visible in the Packet List pane of the ***Packets*** view of the active Capture window or Packet File window. For a complete description of this option, please read the section entitled “Saving as packet list (comma- or tab-delimited)” below.
- *Decoded Packets (*.txt)* saves the decoded packets as a plain text file (*.txt).
- *Decoded Packets (*.rtf)* saves the packets in an RTF file (*.rtf) that preserves the text formatting and page layout of the same packets in the ***Decode*** view of a EtherPeek **Packet Decode** window. For a complete description of this option, please see “Saving as decoded packets (RTF or HTML)” below.
- *Decoded Packets (*.htm)* saves the packets in an HTML file (*.htm) that preserves the text formatting and page layout of the same packets in the ***Decode*** view of a EtherPeek **Packet Decode** window. For a complete description of this option, please see “Saving as decoded packets (RTF or HTML)” below.
- *NAI Sniffer DOS file (*.enc)* saves the packets as a Sniffer® trace in DOS format with a *.enc file extension.
- *TCP/UDP/RTP Data File (*.*)* saves the part of the packet that is after the end of the TCP, UDP, or RTP header, up to and including the data at the offset specified by the Total Length field of the IP header. This part of the packet typically contains the application data for file transfers, for example. If multiple packets are selected, their contents are saved as one continuous file, in packet number order. You must supply a file name and file extension. This option is only available when you choose **Save Selected Packets**.

Saving as packet list (comma- or tab-delimited)

When you choose to save packets as *Packet List (Comma-delimited)* or *Packet List (Tab-delimited)*, the output file contains only the information shown in the Packet List pane of the ***Packets*** view of the active Capture window or Packet File window. By changing the information displayed in that view (by adding or subtracting columns, re-ordering columns, hiding or unhiding packets, and so forth), you can fully tailor the output to either of these file types. Comma-delimited and tab-delimited files are widely supported interchange formats among spreadsheet and database programs.

Note: Comma separated value (*.csv) files are available in ASCII encoding only. All other text files in EtherPeek are saved in UTF-8 encoding.

Saving as decoded packets (RTF or HTML)

EtherPeek can save decoded packets to RTF (Rich Text Format) or HTML (HyperText Markup Language) formats. Either of these text plus mark-up formats will preserve the text formatting and page layout used to present the decoded packets on the screen (for example, in the ***Decode*** view of a **Packet Decode** window or the Decode pane of the ***Packets*** view of a Capture window or Packet File window).

Choosing to save packets in either of these formats provides you with a file that includes information similar to that displayed in the **Packet Decode** window for each packet saved.

Deleting packets

To delete all packets, including any hidden packets, from a window, choose **Clear All Packets** from the **Edit** menu or press **Ctrl + B**.

Tip There is no direct command to delete selected packets. Instead, select the packets you wish to save, and save them to a new file. This can be done by a variety of methods. You can then either delete the original file (if it is a Packet File) or simply close the Capture window without saving.

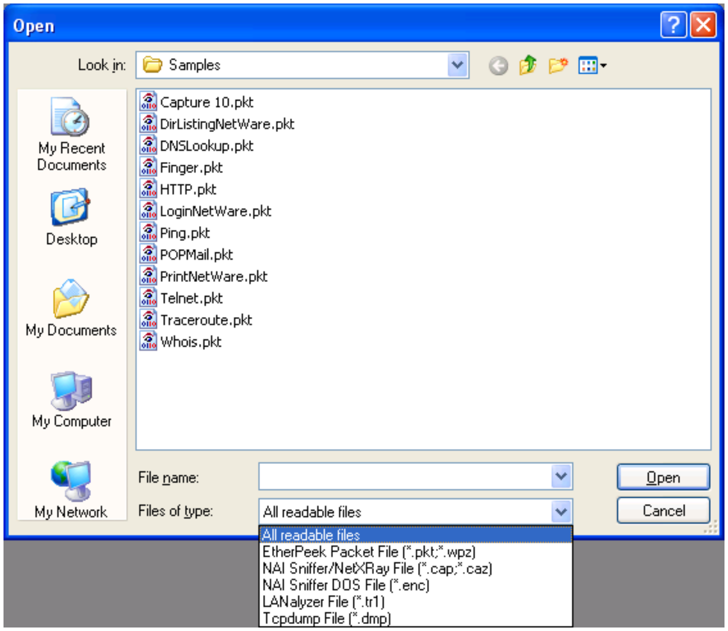
Loading packets from a file

If you save packets in a file format recognized by EtherPeek, you can open them again in a Packet File window using the **Open…** command in the **File** menu. A dialog opens in which you can select files of the following formats:

- **EtherPeek Packet File**: These are files created by using the **Save All Packets…** or **Save Selected Packets…** commands from the **File** menu in EtherPeek or in EtherPeek or saved in PacketGrabber. Files saved to this format from other versions of EtherPeek, including those running on the Macintosh, can also be opened by EtherPeek. Note that packet files must have a *.pkt or *.wpz (compressed format) extension in order to be recognized by EtherPeek.

- **NAI Sniffer/NetXray File**: These are files containing packets captured in the Sniffer® or NetXray® programs. For EtherPeek to recognize these files, they must have an extension of *.cap or *.caz.
- **NAI Sniffer DOS File**: These are files containing packets captured in the Sniffer® program for DOS. For EtherPeek to recognize uncompressed Sniffer files, the files must have an extension of *.enc.
- **LANalyzer File**: These are files containing packets captured in the LANalyzer® program. For EtherPeek to recognize LANalyzer files, the files must have an extension of *.tr1 (numeral one).
- **TCP Dump**: These are files containing packets captured using the open source TCP Dump program. These files must have an extension of *.dmp in order for EtherPeek to recognize them as TCP Dump files.

You can only load one file in a given Packet File window. You can use the PeekCat command line utility (located in the EtherPeek\Bin directory) to concatenate multiple EtherPeek packet files. Please see the PeekCat.txt file in the \Bin directory for more information.



Look in: Samples

File name:

Files of type: All readable files

Figure 4.11 File Open dialog

Note: ProConvert, the WildPackets packet trace conversion utility, can convert in either direction between a wide variety of packet traces including Sniffer® (compressed *.enc, *.cap), Wandel Goltermann, and so forth; and EtherPeek (*.pkt) formats. For more information on ProConvert, please visit the Product Information pages at: http://www.wildpackets.com.

PacketGrabber

PacketGrabber is a helper tool you may install on other machines for the purpose of capturing packets to analyze in EtherPeek. You can, for example, email PacketGrabber to someone who is having problems and ask them to email back the packets captured in PacketGrabber for you to analyze in EtherPeek.

PacketGrabber allows you to distribute network data capture throughout your organization. Because PacketGrabber is optimized for data capture and not for data analysis, you can distribute the program without raising security concerns. The program is compact, easy to install and easy to use. Complete instructions for installing and using

PacketGrabber are included in the documentation for the program, contained in the PacketGrabber directory on the distribution CD or the online image downloaded from the WildPackets ftp site. For more information on PacketGrabber, please visit the Product Information pages at: http://www.wildpackets.com.

Printing packet lists and packet decode windows

To print the complete list of packets shown in the ***Packets*** view of the active Capture window or Packet File window, choose the **Print…** command from the **File** menu.

Printing lists of selected packets

If you would like to print only some of the list of packets in a Capture window or Packet File window, use the functions under the **Edit** menu to hide everything except what you wish to print. When you choose the **Print…** command from the **File** menu, only the listings for the visible packets will be printed. For more on selecting, hiding and unhiding packets, please see Chapter 15, “Post-capture Analysis” on page 283.

To print in landscape format or to use other standard printer options, choose the **Print Setup…** command in the **File** menu.

Printing packet decode windows

To print individual decoded packets, select the packets you would like to print and choose **Print Selected Packets…** from the **File** menu. This will print out a formatted text version of *only* the decode portion of the selected packets. This will print the packet decodes as a single document without page breaks between the packets.

Tip An alternative is to save the decoded packets as RTF or HTML and print them from another application that can read and print those file types. This alternative preserves the formatting of the **Packet Decode** window and allows multiple packets to be printed on individual pages.

AutoCapture

The AutoCapture feature allows the user to set EtherPeek to automatically start multiple Capture windows, each with its own buffer size, adapter selection settings, save options, triggers, filters and performance settings. When capture in all windows is completed, the AutoCapture function sends the resulting capture files by a user-specified method, and checks for any Capture windows having triggers set for *Repeat mode*. If any Capture windows have triggers set for *Repeat mode*, the AutoCapture file resets the start trigger for these windows. If no Capture window has *Repeat mode* enabled, the AutoCapture file exits the application when the actions specified in the *Send options* are completed.

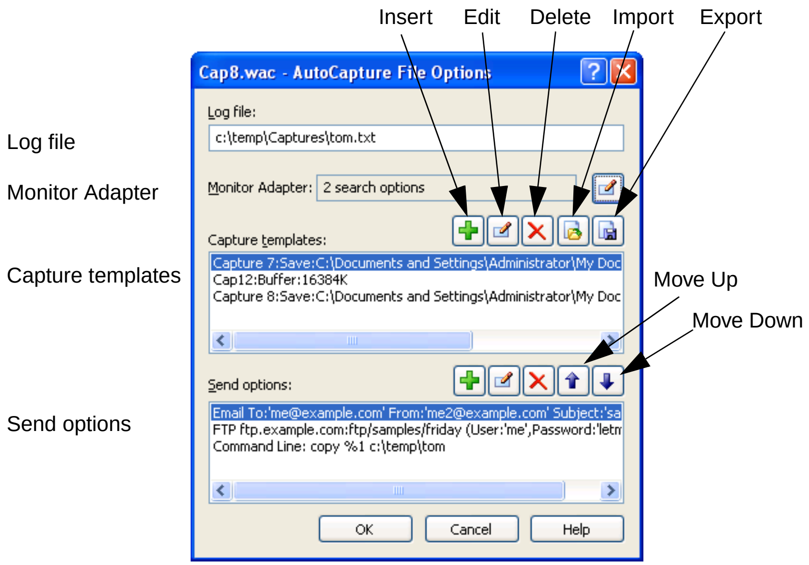
AutoCapture settings are saved in a file which can be sent to a remote user. Remote users can double-click on the file to run it immediately, or schedule EtherPeek to run using the Windows Scheduler.

Creating and editing AutoCapture files

To create or edit AutoCapture (*.wac) files, choose the **Create New…** or **Edit Existing…** sub-menu choices under **AutoCapture** in the **File** menu. This brings up the **AutoCapture File Options** dialog (Figure 4.12). When editing an existing file, the name of the *.wac file is shown in the dialog title. When creating a new file, the dialog title appears as **New AutoCapture File Options**. There are four sections in the **AutoCapture File Options** dialog: *Log file*, *Adapter search*, *Capture templates*, and *Send options*. Each of these is described below.

Log file

You can optionally specify the name and location of a text log file for an AutoCapture file. Each of the actions taken by the AutoCapture file will be appended to the end of the specified log file in text format.



Cap8.wac - AutoCapture File Options

Log file:
c:\temp\Captures\tom.txt

Monitor Adapter: 2 search options

Capture templates:

```
Capture 7:Save:C:\Documents and Settings\Administrator\My Doc
Cap12:Buffer:16384K
Capture 8:Save:C:\Documents and Settings\Administrator\My Doc
```

Send options:

```
Email To:'me@example.com' From:'me2@example.com' Subject:'sa
FTP ftp.example.com:ftp/samples/friday (User:'me',Password:'letm
Command Line: copy %1 c:\temp\tom
```

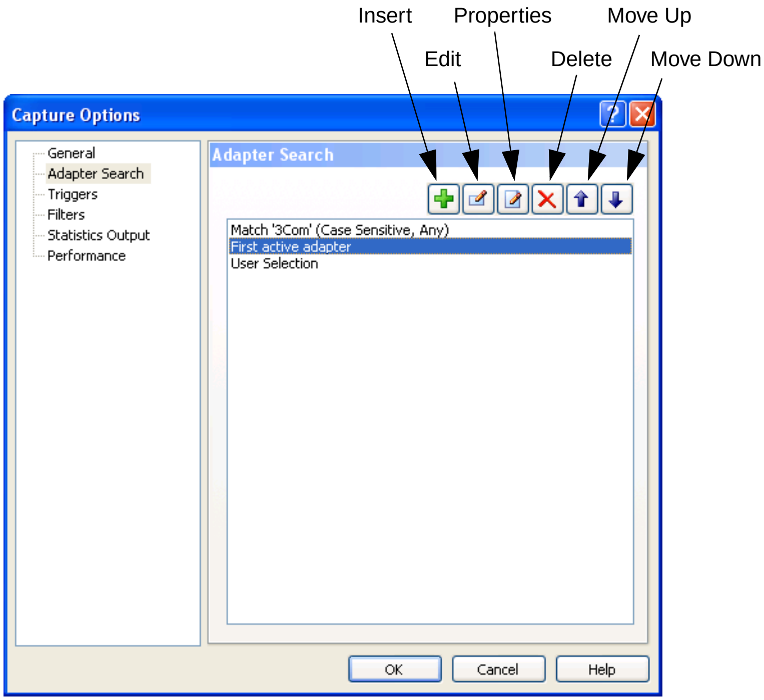
Figure 4.12 AutoCapture File Options window

Monitor adapter and adapter search

The AutoCapture file must be able to select an adapter for EtherPeek to use in capturing packets. You can use the program's default capture adapter, or you can specify one or more search methods for locating an adapter. The program's default adapter is the valid adapter (an actual NIC, not *File* or *None*) most recently selected as the Monitor Adapter in the **Monitor Options** dialog.

If you are unsure of the current default adapter for the target instance of EtherPeek, or if you want to specify the default adapter by setting your own choice for the Monitor Adapter on the target system, you can add one or more adapter search instructions to the *Monitor Adapter* section of the **AutoCapture File Options** dialog. Click the **Edit** button beside the *Monitor Adapter* text display box to open the special AutoCapture version of the **Capture Options** dialog (Figure 4.13). In the ***Adapter Search*** view of this dialog you can **Insert**, **Edit**, or **Delete** adapter search routines using the named buttons, or use the **Move Up** and **Move Down** buttons to change the order of adapter search routines.

EtherPeek will attempt to select a Monitor adapter based on each search method, in the order specified in the *Adapter search* section of the **AutoCapture File Options** dialog. EtherPeek will use the first usable adapter it finds, and ignore any further search methods.



Match '3Com' (Case Sensitive, Any)
First active adapter
User Selection

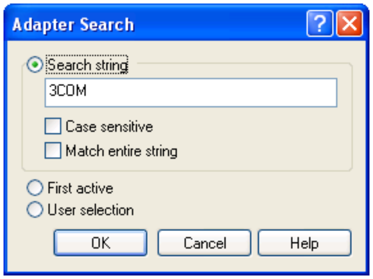
Figure 4.13 Adapter Search view of the special AutoCapture Capture Options dialog

Important! There are two levels of adapter search in an AutoCapture file. The settings in the *Monitor Adapter* section of the **AutoCapture File Options** dialog provide the default adapter for the AutoCapture file as a whole. The settings in the ***Adapter Search*** view of the **Capture Options** dialog for each separate capture template within the *.wac file define the method for selecting the adapter for the Capture window made from that template. The adapter selected for the AutoCapture file as a whole is treated as the default adapter by the ***Adapter Search*** settings of each individual capture template.

Note: After an AutoCapture (*.wac) file has been run successfully, it remembers the adapter it last used. The next time it is run, it first attempts to use that same adapter, regardless of any settings in the *Adapter search* section. If that attempt fails, it then runs through the choices, as if the AutoCapture file were being run for the first time. An AutoCapture file will only treat an actual NIC as the default adapter, never *File* or *None*.

Table 4.6 **Adapter search methods**

Search Method	Usage
Search string	Selects the first adapter whose description contains a match with the text in the user-supplied search string. You can constrain the search to be *Case sensitive* and/or to *Match whole string* by checking the checkbox beside either or both of those choices. You can see examples of the adapter descriptions over which this Adapter Selection method will search, in Windows **Device Manager** and in the *Device Description* in the lower pane of the ***Adapter*** view of either the **Monitor Options** or the **Capture Options** dialog.
First active	Selects the first active, usable adapter in the list of adapters installed on the host computer.
User selection	Opens the ***Adapter*** view of the **Capture Options** dialog, from which a user must actively choose an adapter. Note that if you use this method, EtherPeek will wait indefinitely for user input.
Default (blank)	If no specific adapter search method is listed in the *Monitor Adapter* section of the **AutoCapture File Options** dialog, EtherPeek will attempt to use its default adapter. (For details about the default adapter and how it is chosen for each local instance of EtherPeek, please see “Default local adapter” on page 60.) If any explicit Adapter Search methods are listed in the *Monitor Adapter* section, the AutoCapture file will attempt to use them first. That is, the search for the default adapter is always present, but is always last in the list of adapter search methods.



Adapter Search

● Search string

3COM

□ Case sensitive

□ Match entire string

○ First active

○ User selection

Figure 4.14 Adapter Search dialog

To define a new adapter search method, click the **Insert** button in the *Adapter search* section of the **AutoCapture File Options** dialog. This opens the **Adapter Search** dialog (Figure 4.14). Use the radio buttons to choose the adapter search method. Your choices are: *Search string*, *First active*, or *User selection*. Each of these methods is described in Table 4.6. When you have defined the new search method, click **OK** to add it to the list and close the dialog, or click **Cancel** to close the dialog without creating a new search method. New adapter search methods are added to the bottom of the list, and show as much of the method's parameters as can be displayed on a single line in the *Adapter search* section of the **AutoCapture File Options** dialog.

To edit a search method, highlight its entry in the *Adapter search* section of the **AutoCapture File Options** dialog and click the **Edit** button to bring up the **Adapter Search** dialog with that method's parameters displayed and ready to edit. Click **OK** to accept your changes or click **Cancel** to close the dialog without changing the adapter search method.

To delete a search method from the list, highlight its entry in the *Adapter search* section of the **AutoCapture File Options** dialog and click the **Delete** button.

EtherPeek or PacketGrabber will use the search methods in order from top to bottom as they appear in the *Adapter search* section of the **AutoCapture File Options** dialog. To change the list order, highlight a list item and use the **Move Up** or **Move Down** buttons to move the item.

Capture templates

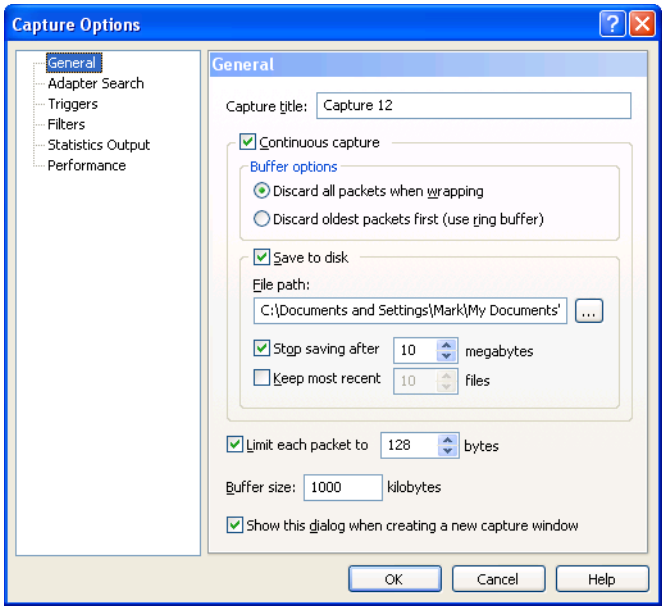
AutoCapture files use capture templates to create Capture windows. Each template creates one Capture window. A single AutoCapture file can have multiple capture templates and create multiple Capture windows. You can use existing capture templates, or you can create or modify capture templates from within the **AutoCapture File Options** dialog.

Note: In EtherPeek, a single capture template can define multiple Capture windows. This is not true inside an AutoCapture file. If you import the settings from a multi-window capture template, it will be read into the AutoCapture file as a distinct template for each Capture window.

A capture template specifies all the parameters found in the **Capture Options** dialog for a given Capture window. Although you can use capture templates created in other programs (GigaPeek NX, for example), capture templates used for AutoCapture have three special requirements:

- Because AutoCapture files are intended to be usable on remote machines, the ***Adapter*** view of an ordinary capture template is replaced by an ***Adapter Search*** view in the capture templates created in or imported into an AutoCapture file.
- You must save captured packets before they can be sent using the *Send options*. In practice, this means you should enable the *Continuous capture* and *Save to disk* options in the ***General*** view of the **Capture Options** dialog for each template.
- A stop trigger must be set for each capture template, or the capture will never terminate and no files will be sent. Capture must stop in *all* the Capture windows created by a given AutoCapture file before *any* files will be sent. Automatic saving of captured packets is only supported under the *Continuous capture* setting in the ***General*** view of the **Capture Options** dialog. Under the *Continuous capture* setting, only active user intervention or a stop trigger will stop capture.

To create a new capture template, click the **Insert** button in the *Capture templates* section of the **AutoCapture File Options** dialog. This opens the special version of the **Capture Options** dialog used for AutoCapture files (Figure 4.15). where you can specify the name, buffer usage, packet slicing, and other parameters for the Capture window created from this template. For a detailed discussion of the **Capture Options** dialog and how to use it, please see “Capture options dialog” on page 51. When you have specified the capture options for this template, click **OK** to add it to the list.



Capture Options

General
Adapter Search
Triggers
Filters
Statistics Output
Performance

General

Capture title: Capture 12

☑ Continuous capture

Buffer options

● Discard all packets when wrapping

○ Discard oldest packets first (use ring buffer)

☑ Save to disk

File path: C:\Documents and Settings\Mark\My Documents'

☑ Stop saving after 10 megabytes

□ Keep most recent 10 files

☑ Limit each packet to 128 bytes

Buffer size: 1000 kilobytes

☑ Show this dialog when creating a new capture window

Figure 4.15 Special Capture Options dialog for AutoCapture files, General view

Tip You can use the ***Performance*** view of the **Capture Options** dialog to selectively disable program functionality in a particular Capture window; turning off such functions as the Expert, Peer Map, Analysis Modules, and so forth. Because the primary purpose of AutoCapture is to collect packets for later analysis, you can typically disable all functions except capture itself. This reduces overhead and speeds operation. For details on how to use the Performance view, please see “Performance views” on page 25.

To add a previously saved capture template to the list, click the **Import** button to bring up a file **Open** dialog. Use the file **Open** dialog to navigate to the location of the capture template (*.ctf) file you wish to add. Choose the file and click **OK** to add it to the list.

To save a capture template from the *Capture templates* list as a free-standing capture template for later re-use, highlight its entry in the list and click the **Export** button. This brings up a **Save As** dialog which you can use to name the template and navigate to the location where you would like to save the capture template (*.ctf) file.

Note: When you use **Import** to add a previously existing capture template to an AutoCapture file, the template’s parameters are copied into the AutoCapture file. If you then modify these parameters from within the **AutoCapture File Options** dialog, only the

AutoCapture file's copy of the template parameters is modified. The original capture template remains unchanged. When you delete an imported capture template from the list, the template is removed from the AutoCapture file, but the original capture template (*.ctf) file is unaffected. Similarly, when you use **Export** to save a capture template, any further changes made to that template in the **AutoCapture File Options** dialog have no effect on the previously saved version.

To edit the capture options for a particular capture template, highlight the template in the *Capture templates* section of the **AutoCapture File Options** dialog and click the **Edit** button. This opens the **Capture Options** dialog with that template's parameters displayed and ready to edit. When you have made your changes, click **OK** to close the dialog and accept your changes, or click **Cancel** to close the dialog without changing the template's parameters.

Tip EtherPeek or PacketGrabber will automatically import a similarly named filter file found in the same location as the AutoCapture (.wac) file when starting an AutoCapture session. For example, if the AutoCapture file is named Agincourt.wac, EtherPeek will look in the same directory for a filter file named Agincourt.flt from which to import filters. EtherPeek adds the filters to the existing list, rather than replacing it. Duplicates of existing filters will be ignored if they have identical parameters as well as identical names. Filters with the same name but different parameters will be added with "*Copy of*" added to their names.

To delete a capture template from the list, highlight the listing for that template in the *Capture templates* section of the **AutoCapture File Options** dialog and click the **Delete** button.

Send options

When capture is stopped in all Capture windows, EtherPeek attempts to send a single packet file using the first send option listed in the *Send options* section of the **AutoCapture File Options** dialog. If the first send option fails, EtherPeek tries any remaining send options in the order in which they are listed in the *Send options* section. All packet files are sent using the first send option that succeeds, and any remaining send options are ignored. If no send option succeeds, no packet files are sent. There are three types of send option:

- *Email*
- *FTP*
- *Command line*

You can create multiple instances of the same basic type (for example, multiple Email send options, each using a different server), but only the first successful send option will actually be used by EtherPeek or PacketGrabber.

To create a new send option, click the **Insert** button in the *Send options* section of the **AutoCapture File Options** dialog. This brings up the **Send Options** dialog (Figure 4.16). Use the radio buttons to choose the type of option. Your choices are *Email*, *FTP* or *Command line*. Fill in the required information and any optional information for the chosen method, using the instructions in Table 4.7. Click **OK** to create the specified send option and close the dialog, or click **Cancel** to close the dialog without creating a new send option. New send options are added to the bottom of the list, and show as much of the option's parameters as can be displayed on a single line in the *Send options* section of the **AutoCapture File Options** dialog.

Note: The *Remove files after send completes* option is enabled or disabled for each send option individually. The files are only removed if this option is enabled (checked) in the particular send option ultimately used to send the files, and is ignored when it is enabled in a send option that is not used.

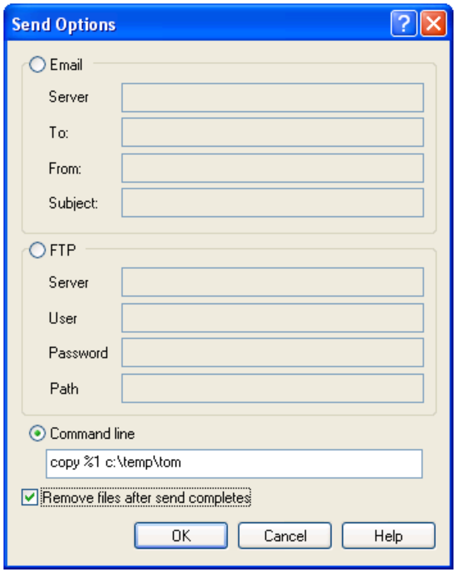
Table 4.7 **Send option usage**

Option	Usage
Email	Sends the packet (*.pkt) files as attachments in email, one file per email. You must specify a valid email *Server*, and valid email addresses in the *To* and *From* edit boxes. The *Subject* line is optional.
FTP	Copies the packet (*.pkt) files to the specified *Path* (directory) using FTP. You must specify a valid FTP *Server*, a valid *User* name and *Password* for that server, and the *Path* to a valid directory on that server. Note: because packet (*.pkt) files include a time signature in the file name, it is highly unlikely any two will ever have the same name, but not strictly impossible. In the unlikely event of identical file names, files will be overwritten only if the permissions for the *User* allow it.

Table 4.7 Send option usage (Continued)

Option	Usage
Command line	Executes the specified command line instruction on each packet (*.pkt) file in turn. Enter a valid command line in the text entry box, using the string `%1` as a substitute for the file names of the packet files. For example, to copy the files to the C:\temp\ directory, the command line would be: `copy %1 C:\temp\`
Remove files after send completes	The *Remove files after send completes* option removes each file after it is sent. Check to enable, uncheck to disable.

To edit a send option, highlight its entry in the *Send options* section of the **AutoCapture File Options** dialog and click the **Edit** button to bring up the **Send Options** dialog with that option's parameters displayed and ready to edit. Click **OK** to accept your changes or click **Cancel** to close the dialog without changing the send option.



● Command line

copy %1 c:\temp\tom

☑ Remove files after send completes

Figure 4.16 Send Options dialog

To delete a send option from the list, highlight its entry in the *Send options* section of the **AutoCapture File Options** dialog and click the **Delete** button.

EtherPeek or PacketGrabber will try the send options in order from top to bottom as they appear in the *Send options* section of the **AutoCapture File Options** dialog. To change the list order, highlight a list item and use the **Move Up** or **Move Down** buttons to move the item.

Using an AutoCapture file

To execute an AutoCapture, double-click on an AutoCapture (*.wac) file or specify the file on the command line. For example:

```
Peek.exe c:\temp\Poitiers.wac
```

When launched with an AutoCapture file as its object, EtherPeek will:

1. Establish a log file, if one is specified for the AutoCapture file.
2. Search the directory where the AutoCapture (*.wac) file is located, looking for a file of the same name but with the filter (*.flt) file extension. If it finds such a filter file in that directory, it will import it into the **Filters** window.
3. Run through the adapter search methods in the *Monitor Adapter* section of the AutoCapture file to select a valid adapter. If multiple methods are enabled, they will be tried in the order specified, and the first successful selection will set the Monitor Adapter.
4. Create the Capture window(s) specified by the capture template(s), executing the *Adapter search* methods (if any) specified by each individual capture template. The adapter found by the methods specified in the *Monitor Adapter* section of the AutoCapture file will become the fall-back or default adapter for each of these individual adapter searches.
5. Start capture or set the start/stop triggers for each Capture window.
6. Wait for all Capture windows to stop capturing.

Important! Be sure to enable the *Continuous capture* and *Save to disk* options and set a Stop Trigger for every capture template in the AutoCapture file. No files will be sent until capture is stopped in all Capture windows. Packets must be saved before they can be sent.

7. Run through the *Send options* to send or save any Packet Files. The first successful send option will be used to send all of the files.

8. Remove the sent or saved files if *Remove files after send completes* is selected for the Send Option used.
9. Check to see if any of the created Capture windows has triggers set to repeat mode. For any Capture window for which *Repeat mode* is enabled, the AutoCapture file will clear the capture buffer and return to step 5.
10. If no Capture window has triggers set to *Repeat mode*, the AutoCapture file will exit EtherPeek when the send options are completed.

EtherPeek can also be scheduled with the Windows Task Scheduler, available from the Windows **Control Panel**. The easiest way to use EtherPeek with an AutoCapture file as a scheduled task is to create a batch file (*.bat) with the desired command line, then schedule the batch file to run at a specified time in the Task Scheduler. For more about the command line, please see “Starting EtherPeek from the command line” on page 28.

Expert View and Expert EventFinder

Unique to EtherPeek NX, the ***Expert*** view provides expert analysis of delay, throughput and a wide variety of network events and potential problems in a conversation-centered view of traffic in a Capture window or Packet File window.

The Expert EventFinder scans traffic in a Capture window or Packet File window, looking for key events. You can configure the Expert EventFinder to be as narrowly or as broadly focused as you like. The EventFinder’s 91 separate events cover anomalies, sub-optimal performance and other key events at all layers of the network, from application to physical. The Expert monitors Client/Server delay and throughput as well.

You can enable and disable each test individually. In addition, many of the events have user-defined settings and thresholds, allowing you to fine-tune the Expert system to precisely fit your needs. You can save and reload Expert EventFinder settings for use in particular environments.

The ***Expert*** view provides aggregate EventFinder results, but it also provides a detailed view of every transaction, noting any events encountered in each individual conversation or flow.

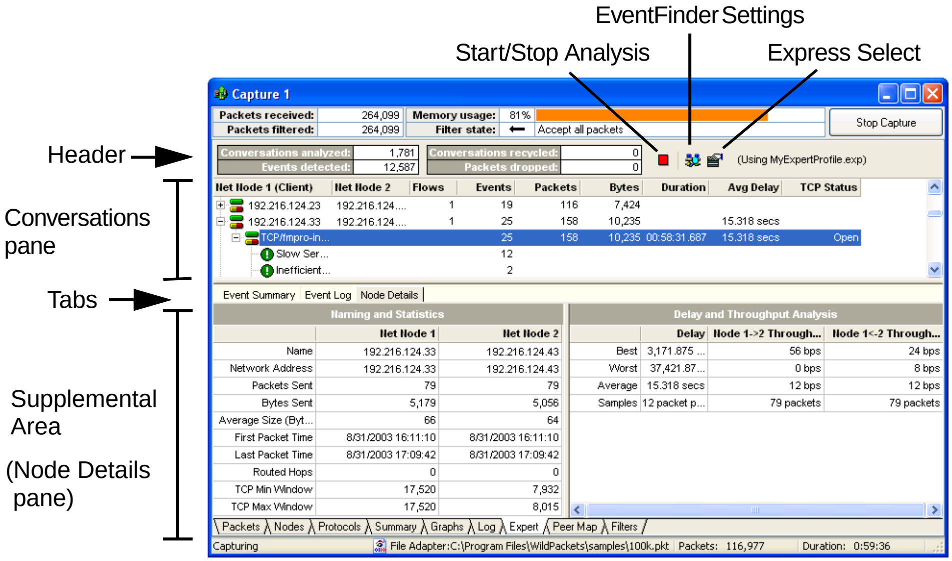
You can use the **Express Select** button to instantly highlight the packets associated with a particular event, or with any conversation in the ***Expert*** view.

The Expert EventFinder not only helps identify key events, but it also helps you understand the meaning, the typical causes, and the typical solutions to the problems it uncovers. Detailed information is only a click away.

In this Chapter:

Expert view

The ***Expert*** view has a header section and two data areas: the Conversations pane of the ***Expert*** view above, and a supplemental area below. The Conversations pane displays conversations or flows, nested under the address or name of the client node. The supplemental area can display one of three additional panes, accessible by clicking the labeled tabs. The tabs are: *Event Summary*, *Event Log*, and *Node Details*. Each of these elements of the ***Expert*** view is described in turn below.



Packets received: 264,099
Packets filtered: 264,099
Memory usage: 81%
Filter state: ← Accept all packets

Conversations analyzed: 1,781
Events detected: 12,587
Conversations recycled: 0
Packets dropped: 0
(Using MyExpertProfile.exp)

Net Node 1 (Client)	Net Node 2	Flows	Events	Packets	Bytes	Duration	Avg Delay	TCP Status
192.216.124.23	192.216.124....	1	19	116	7,424			
192.216.124.33	192.216.124....	1	25	158	10,235		15.318 secs	
TCP/fmpro-in...			25	158	10,235	00:58:31.687	15.318 secs	Open
Slow Ser...			12					
Inefficient...			2					

Naming and Statistics

	Net Node 1	Net Node 2
Name	192.216.124.33	192.216.124.43
Network Address	192.216.124.33	192.216.124.43
Packets Sent	79	79
Bytes Sent	5,179	5,056
Average Size (Byt...	66	64
First Packet Time	8/31/2003 16:11:10	8/31/2003 16:11:10
Last Packet Time	8/31/2003 17:09:42	8/31/2003 17:09:42
Routed Hops	0	0
TCP Min Window	17,520	7,932
TCP Max Window	17,520	8,015

Delay and Throughput Analysis

	Delay	Node 1->2 Through...	Node 1<-2 Through...
Best	3,171.875 ...	56 bps	24 bps
Worst	37,421.87...	0 bps	8 bps
Average	15.318 secs	12 bps	12 bps
Samples	12 packet p...	79 packets	79 packets

Capturing | File Adapter:C:\Program Files\WildPackets\samples\100k.pkt | Packets: 116,977 | Duration: 0:59:36

Figure 5.1 EtherPeek NX Expert view , showing Node Details pane

Important! The ***Expert*** view and its ability to write to the ***Expert*** column of the ***Packets*** view must be enabled in order to function. The ***Expert*** view is enabled by default. You can enable or disable the Expert function for EtherPeek NX as a whole in the ***Analysis Modules*** view of the **Options** dialog (available by choosing **Options…** from the **Tools** menu). When the Expert is enabled in the ***Analysis Modules*** view, you can turn the Expert function on or off for a particular Capture window by checking or unchecking the *Expert* item in the ***Performance*** view of the **Capture Options** dialog for that Capture window.

Expert view header

The header section of the ***Expert*** view (Figure 5.1) shows the number of *Conversations Analyzed* and the *Events Detected*. When the Expert runs in a Capture window, it uses a fixed block of memory allocated when the Capture window is created. The counts of *Conversations recycled* and *Packets dropped* relate to the Expert’s use of this memory. Please see “Expert memory allocation” on page 116 for details.

To the right of this information are three buttons: **Start/Stop Analysis**, **EventFinder Settings**, and **Express Select**.

The **Start/Stop Analysis** button displays as either a red square (click to stop) or a green arrow (click to start), depending on the current state of analysis.

Click the **EventFinder Settings** button to open the **Expert EventFinder Settings** window, where you can configure the individual Expert events.

Click the **Express Select** button to use the conversation currently selected in the Conversations pane as the basis for a **Select Related Packets** selection in the ***Packets*** view.

Expert view conversations pane

The Conversations pane of the ***Expert*** view shows the current conversations, with information about each conversation or flow displayed in a user-definable set of columns. Right-click in the Conversations pane to open the context menu and choose **Visible columns…** to select the columns you wish to display. Use drag and drop to change column order. To use drag and drop, click on a column heading, then drag the ghost image of the column heading to a new location and release the mouse button. The columns available in the Conversations pane of the ***Expert*** view are shown in Table 5.1. Columns present in the default Conversations pane layout show an **X** in the **Default** column of Table 5.1.

Table 5.1 Expert view, conversations pane columns

Default	Column	Description
X	***Net Node 1 (Client)***	The client or first peer in the selected conversation. NOTE: This column is always displayed, and cannot be toggled on or off like the other columns.
X	***Net Node 2***	The server or second peer in the selected conversation.
X	***Flows***	For a pair of nodes, shows the number of flows or conversations detected and detailed in the Conversations pane.
X	***Events***	Total number of events identified by the Expert EventFinder. Note that count of events is rolled up when the view is collapsed, such that higher levels of aggregations show totals for all sub-elements.
	Protocol	The protocol under which the packets in this conversation were exchanged.
	Hops	Number of hops separating the two end points of this conversation.
X	***Packets***	The number of packets in the selected exchange. Note that packet totals are rolled up when the view is collapsed, such that higher levels of aggregations show totals for all sub-elements.
X	***Bytes***	The total bytes represented by the packets which were a part of the selected conversation.
X	***Duration***	The elapsed time, from the first to the last packet of the selected exchange, represented in the form Hours:Minutes:Seconds:Milliseconds.
X	***Avg Delay***	For exchanges in which this parameter is relevant, shows the arithmetic average of all client/server response times or of latencies for the selected pair of nodes.
X	***TCP Status***	For exchanges that represent TCP transactions, notes whether the session is *Open* or *Closed*.

The Conversations pane of the ***Expert*** view of a Capture window or Packet File window provides a hierarchical view of all conversations contained in the visible packets in the buffer of the window. Each highest-level item in the display represents a single node acting as the Client or first peer in a particular conversation. When a group of conversations differ only in port number, they are ranged below the Client node in order by port number. Any events diagnosed by Expert EventFinder are shown in the next level of hierarchy below this one.

Note: The terms “conversation” or “flow” are equivalent, and have a precise meaning in the ***Expert*** view. For IP, the end-to-end IP address, and UDP or TCP ports form a unique conversation or flow for a given application. For IPX, the end-to-end IPX address, socket number, and connection IDs form a unique conversation or flow for a given application.

Items in the Conversations pane are color coded for easy scanning. When a conversation is still active, the color block beside that item is bright green. When the conversation is completed, the color block is dull green. When an event has been identified as being associated with that particular conversation, a yellow color block appears beside the Client node. If some of the events identified are classified as Major or Severe, the block will show part red and part yellow. If all of the events are Major or Severe, the whole block will be red.

Click on the + (plus) or - (minus) signs at the left margin to expand or collapse individual elements of the display. Alternatively, you can right-click anywhere in the Conversations pane to open the context menu and choose either **Expand All** or **Collapse All**.

Forcing server identification

The Expert makes its best attempt to determine which node is the client and which the server in each conversation. You can override this behavior by making entries in the Name Table telling the ***Expert*** view to always identify certain nodes as the server, regardless of the context. If a node is identified in the ***Expert*** view as a client and you wish to have that IP address always treated as a server, you can make an entry in the Name Table as follows:

1. In the Conversations pane of the ***Expert*** view, right-click on the conversation in which the node is identified as a client and choose **Insert Net Node 1 into Name Table…** from the context menu.
2. In the **Add Name** or **Edit Name** dialog which appears, accept the other entries, but set the *Node type* entry to *Server* by choosing from the drop-down list.
3. Click **OK** to make the change to the Name Table.

4. The Expert checks the Name Table to identify nodes, and your changes will be reflected in subsequent captures or on re-reading this captured file or doing post capture analysis.

   When you designate a node as a *Server* in the Name Table, all connections to that IP address will identify this address as a server, regardless of other contextual clues. To once again allow the Expert to determine from context whether this node is acting as the client or server, delete the node's entry from the Name Table or change its *Node type* to *Workstation* or *Unknown*.

Expert view supplemental information panes

The supplemental information area at the bottom of the ***Expert*** view provides summary counts of events and additional detail about the events and the participants in the conversations shown in the Conversations pane above it. The supplemental information area can show one of three panes, accessible by clicking on the labeled tabs.

The panes, and the data tables they contain, are:

Event Summary	***Event Log***	***Node Details***	
Event Summary table	*Event Log*	*Naming and Statistics* table	*Delay and Throughput Analysis* table

Event summary pane

The Event Summary pane contains the *Event Summary* table, showing the number of times each type of event was encountered. The header area of the Event Summary pane shows the *Total* number of events identified. The *Event Summary* table has four columns, as shown in Table 5.2. To sort by a particular column, click in the column header. An arrow shows the direction of the sort for the column.

Table 5.2 **Event Summary columns**

Column	Description
Severity Icon	The severity of the event, as set in the **Expert EventFinder Settings** window.
Layer	The network layer to which events of this type belong.
Event	The EventFinder event definition which identified this packet as an event (for example, *TCP Transport Retransmission*).
Count	The number of events of this type seen so far.

Right-click on any item in the *Event Summary* and choose **EventFinder Setting** from the context menu to open the **Expert EventFinder Settings** window with that particular event highlighted and its setting displayed. The **Expert EventFinder Settings** window shows a description of the event and a brief discussion of possible causes and possible remedies.

The context menu also allows you to save the *Event Summary* table, or individual lines from it, to a text file, or to copy them to the clipboard.

When you highlight a type of event in the *Event Summary* which is associated with conversations, the related conversations or flows in the Conversations pane are highlighted. From the *Event Summary* table, you can also choose **Select Related Packets** > **By Event Type** from the context menu (right-click).

Tip To see a list of all the individual instances of an event of a given type, switch to the Event Log pane and sort the *Event Log* by its ***Event*** column, to sort the list in alphabetical order by the name of the event type.

Event log pane

The Event Log pane (shown in Figure 5.2) contains the *Event Log*. The *Event Log* has a header area and a table.

The header area of the Event Log pane shows a count of total *Entries* in the log, and counts of events classified by their level of severity, shown beside the icon for that severity. In order from left to right, these are: Informational, Minor, Major, and Severe. Click on these icons to toggle the display of events of the selected severity in the table below. The counts will continue to update, even if you choose not to display events of a particular severity. When you toggle the display off for one of these levels of severity, the count beside the icon changes to indicate this fact, for example, *0 of 28*. The count of total *Entries* will change as well, showing, for example, *602 of 630*.

The *Event Log* can display up to 5,000 entries. When all levels of severity are visible, this will be the most recent 5,000 entries. In the background, the Expert keeps track of a somewhat larger number of entries. If the *Event Log* is becoming full, you may want to show only the entries with the highest levels of severity. For example, if 10,000 events have been identified, and 8,000 of these had a severity of Informational or Minor, you may want to toggle off the display of these less severe events in order to be able to see all of the more severe events, regardless of when they occurred.

The *Event Log* table shows the individual packets which generated an event notice, based on the settings in the **Expert EventFinder Settings** window. It shows one packet per line. The *Event Log* presents information for each packet in the columns shown in Table 5.3. You can sort the *Event Log* by any column by clicking in the column header. A triangle in the column header shows the order of the sort, ascending or descending. Click again to change the sort order.

Table 5.3 **Event Log columns**

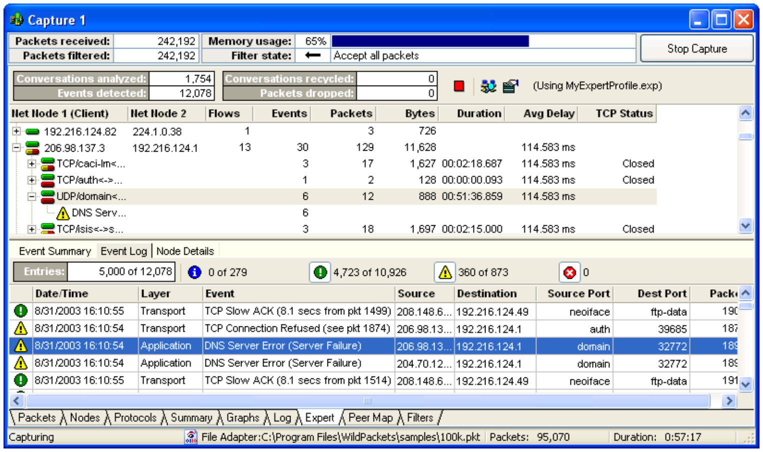
Column	Description
Severity Icon	The severity of the event, as set in the **Expert EventFinder Settings** window.
Date/Time	The date and time of capture for the packet, shown to the nearest whole second.
Layer	The network layer to which events of this type belong.

Table 5.3 **Event Log columns (Continued)**

Column	Description
Event	The EventFinder definition which identified this packet as an event (for example, *TCP Transport Retransmission*). The description may be modified to show additional information. For example, a packet which was identified as an event by the Expert EventFinder item called *TCP Reset Connection* might have an entry in the ***Event*** column of the *Event Log* such as *TCP Connection Reset by Client*. For packets identified by Expert EventFinder items having a user-definable *Setting* value, the description may be followed by the actual measurement which identified this packet as an event (for example *Low Server-to-Client Throughput (1,850 bps)*).
Source	The source address for this packet. The node is identified by its logical address or by the symbolic name for that address if one exists in the Name Table.
Destination	The destination address for this packet. The node is identified by its logical address or by the symbolic name for that address if one exists in the Name Table.
Source Port	The source port for this packet. If the port is a well known port, the protocol or application name will be shown instead of the port number.
Destination Port	The destination port for this packet. If the port is a well known port, the protocol or application name will be shown instead of the port number.
Packet	The packet number, as assigned in the ***Packets*** view of the Capture window or Packet File window. These numbers are assigned in sequence as the packets are captured or read into the buffer.

Click on an entry in the *Event Log* to highlight in the Conversations pane the conversation in which this event occurred. If the display of the Conversations pane is collapsed, clicking on an *Event Log* entry will expand the correct part of the Conversations pane to show the related conversation.

Note: If the entry in the *Event Log* does not apply to any particular conversation, there will be no conversation to highlight.



Capture 1

Packets received: 242,192
Packets filtered: 242,192
Memory usage: 65%
Filter state: ← Accept all packets

Conversations analyzed: 1,754
Events detected: 12,078
Conversations recycled: 0
Packets dropped: 0

(Using MyExpertProfile.exp)

Net Node 1 (Client)	Net Node 2	Flows	Events	Packets	Bytes	Duration	Avg Delay	TCP Status
192.216.124.82	224.1.0.38	1		3	726			
206.98.137.3	192.216.124.1	13	30	129	11,628		114.583 ms	
TCP/caci-lm<...			3	17	1,627	00:02:18.687	114.583 ms	Closed
TCP/auth<->...			1	2	128	00:00:00.093	114.583 ms	Closed
UDP/domain<...			6	12	888	00:51:36.859	114.583 ms	
⚠ DNS Serv...			6					
TCP/isis<->s...			3	18	1,697	00:02:15.000	114.583 ms	Closed

Entries: 5,000 of 12,078 | 0 of 279 | 4,723 of 10,926 | ⚠ 360 of 873 | 0

	Date/Time	Layer	Event	Source	Destination	Source Port	Dest Port	Pack
!	8/31/2003 16:10:55	Transport	TCP Slow ACK (8.1 secs from pkt 1499)	208.148.6...	192.216.124.49	neoiface	ftp-data	19
⚠	8/31/2003 16:10:54	Transport	TCP Connection Refused (see pkt 1874)	206.98.13...	192.216.124.1	auth	39685	187
⚠	8/31/2003 16:10:54	Application	DNS Server Error (Server Failure)	206.98.13...	192.216.124.1	domain	32772	18
⚠	8/31/2003 16:10:54	Application	DNS Server Error (Server Failure)	204.70.12...	192.216.124.1	domain	32772	18
!	8/31/2003 16:10:55	Transport	TCP Slow ACK (8.1 secs from pkt 1514)	208.148.6...	192.216.124.49	neoiface	ftp-data	191

Capturing | File Adapter:C:\Program Files\WildPackets\samples\100k.pkt | Packets: 95,070 | Duration: 0:57:17

Figure 5.2 EtherPeek NX Expert view, showing Event Log pane

When one or more log entries are highlighted, you can use the context menu to **Select Related Packets** in a number of different ways by choosing from the sub-menu. These methods and their results are described in Table 5.5 on page 113.

Right-click on any item in the *Event Log* and choose **EventFinder Setting** from the context menu to open the **Expert EventFinder Settings** window with the definition for that particular event highlighted and its setting displayed. The **Expert EventFinder Settings** window shows a description of the event and a brief discussion of possible causes and possible remedies.

The context menu also allows you to save the *Event Log* or individual lines from it to a text file, or to copy either the whole log or selected items to the clipboard.

Node details pane

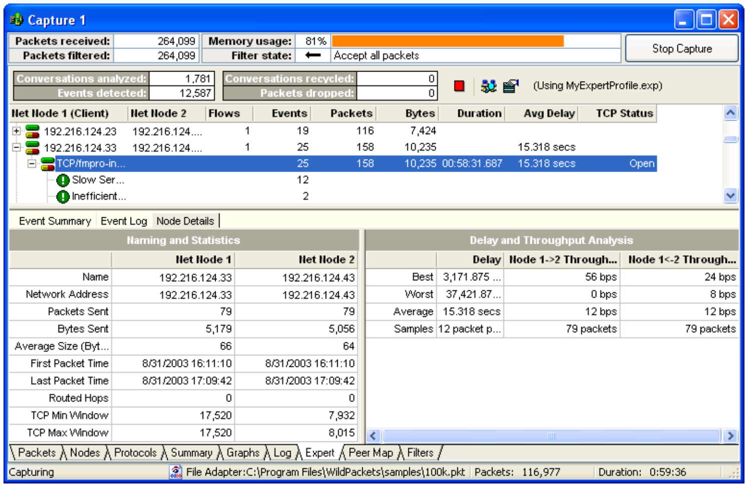
The Node Details pane (shown in Figure 5.3) contains two tables:

- the *Naming and Statistics* table (on the left)

- the *Delay and Throughput Analysis* table (on the right)

Note: Unlike the Event Summary and the Event Log panes, the information in the Node Details pane applies only to the currently selected conversation or flow, or to the item currently selected in the Conversations pane of the ***Expert*** view.

The *Naming and Statistics* table shows additional details for the participants in the selected conversation, identified as ***Net Node 1*** and ***Net Node 2***. The *Naming and Statistics* table shows the characteristics described in Table 5.4 for both Net Node 1 and Net Node 2.



Packets received: 264,099
Packets filtered: 264,099
Memory usage: 81%
Filter state: Accept all packets

Conversations analyzed: 1,781
Events detected: 12,587
Conversations recycled: 0
Packets dropped: 0
(Using MyExpertProfile.exp)

Net Node 1 (Client)	Net Node 2	Flows	Events	Packets	Bytes	Duration	Avg Delay	TCP Status
192.216.124.23	192.216.124....	1	19	116	7,424			
192.216.124.33	192.216.124....	1	25	158	10,235		15.318 secs	
TCP/fmpro-in...			25	158	10,235	00:58:31.687	15.318 secs	Open
Slow Ser...			12					
Inefficient...			2					

Naming and Statistics

	Net Node 1	Net Node 2
Name	192.216.124.33	192.216.124.43
Network Address	192.216.124.33	192.216.124.43
Packets Sent	79	79
Bytes Sent	5,179	5,056
Average Size (Byt...	66	64
First Packet Time	8/31/2003 16:11:10	8/31/2003 16:11:10
Last Packet Time	8/31/2003 17:09:42	8/31/2003 17:09:42
Routed Hops	0	0
TCP Min Window	17,520	7,932
TCP Max Window	17,520	8,015

Delay and Throughput Analysis

	Delay	Node 1->2 Through...	Node 1<-2 Through...
Best	3,171.875 ...	56 bps	24 bps
Worst	37,421.87...	0 bps	8 bps
Average	15.318 secs	12 bps	12 bps
Samples	12 packet p...	79 packets	79 packets

Capturing File Adapter:C:\Program Files\WildPackets\samples\100k.pkt Packets: 116,977 Duration: 0:59:36

Figure 5.3 EtherPeek NX Expert view, showing Node Details pane

Table 5.4 Naming and Statistics table parameters

Parameter	Description
Name	The name (or address) of each node. The node is identified by its logical address or by the symbolic name for that address if one exists in the Name Table.
Address	The logical address, in a format appropriate to the protocol of the conversation.
Packets Sent	The total number of packets sent by this node as a part of this conversation.
Bytes Sent	The total number of bytes sent by this node as a part of this conversation.

Table 5.4 Naming and Statistics table parameters (Continued)

Parameter	Description
Average Size	The average size of the packets sent by this node as a part of this conversation, in bytes.
First Packet Time	The date and time of capture (to the nearest second) of the first packet for this node in the current conversation.
Last Packet Time	The date and time of capture (to the nearest second) of the last packet for this node in the current conversation.
Routed Hops	The number of intervening router hops separating Net Node 1 and Net Node 2 in this conversation.
TCP Min Window	The minimum size of the TCP window during the course of this conversation.
TCP Max Window	The maximum size of the TCP window during the course of this conversation.

The *Delay and Throughput Analysis* table shows the *Best*, *Worst*, and *Average* measures of delay and throughput for the selected conversation, along with the number of *Samples* on which these figures are based. The table shows data in three columns: ***Delay*** (server response time, network latency, and so forth), ***Node 1->Node 2 Throughput*** (peer one to peer two, or client to server throughput), and ***Node 1<-Node 2 Throughput*** (peer two to peer one, or server to client throughput). Delay is shown in milliseconds. To set the units for throughput, choose **Throughput** from the context menu (right-click) and select one of the three sub-menu choices: **Bits/Second (bps)**, **kBits/Second (kbps)**, or **kBytes/Second (kBps)**. The current choice has a dot beside it.

Expert view packet selection

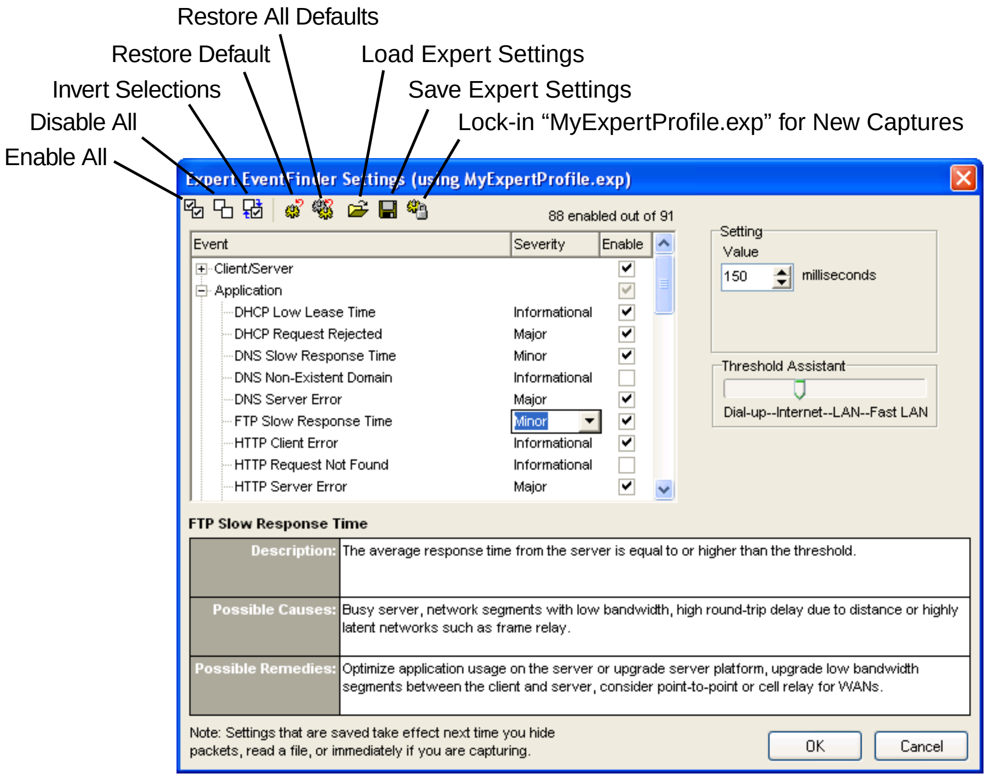
You can select related packets in the ***Expert*** view in a number of ways, depending on the context and the pane of the ***Expert*** view that is active. Right-click and choose **Select Related Packets** from the context menu, then choose a further option from the sub-menu. These sub-menu choices, and the pane in which they are available are shown in Table 5.5.

Table 5.5 Select Related Packets in the Expert view

Pane	Parameter	Action
Conversations pane	**Express Select button**	Click this button in the Expert view header section to use the conversation currently selected in the Conversations pane as the basis for a Select Related Packets By Conversation operation.
Conversations pane	**By Source and Destination**	Chooses packets with matching source and destination addresses.
Conversations pane	**By Conversation**	Chooses packets sent between two nodes (in either direction), using the matching protocol and port.
Event Summary pane	**By Event Type**	Chooses packets associated with events of the specified type.
Event Log pane	**Selected Entries**	Chooses only the individual packet identified with each highlighted entry in the Event Log. The Event Log shows one packet with one event in each log entry. Multiple log entries may be highlighted at once.
Event Log pane	**Selected Entries + “See” or “From Pkt”**	Chooses the individual packet identified with each highlighted entry in the Event Log, plus any packet referred to in the log entry in a phrase which begins “*See Packet…*” or “*From Packet….*” These log entries refer to another packet in the same conversation, such as a response or request packet, for example.

Expert EventFinder

The **Expert EventFinder Settings** window lets you enable or disable any of the 91 Expert EventFinder events individually or all together. Many of these events have user-definable settings which can be customized to match particular tasks or environments. Where settings are related to network bandwidth, the *Threshold Assistant* can help you choose the best setting. In addition, the **Expert EventFinder Settings** window shows the *Description*, *Possible Causes*, and *Possible Remedies* for each event it can diagnose.



88 enabled out of 91

Event	Severity	Enable
⊞ Client/Server		☑
⊟ Application		☑
DHCP Low Lease Time	Informational	☑
DHCP Request Rejected	Major	☑
DNS Slow Response Time	Minor	☑
DNS Non-Existent Domain	Informational	□
DNS Server Error	Major	☑
FTP Slow Response Time	Minor	☑
HTTP Client Error	Informational	☑
HTTP Request Not Found	Informational	□
HTTP Server Error	Major	☑

Setting
Value 150 milliseconds

Threshold Assistant
Dial-up--Internet--LAN--Fast LAN

FTP Slow Response Time

Description:	The average response time from the server is equal to or higher than the threshold.
Possible Causes:	Busy server, network segments with low bandwidth, high round-trip delay due to distance or highly latent networks such as frame relay.
Possible Remedies:	Optimize application usage on the server or upgrade server platform, upgrade low bandwidth segments between the client and server, consider point-to-point or cell relay for WANs.

Note: Settings that are saved take effect next time you hide packets, read a file, or immediately if you are capturing.

Figure 5.4 EtherPeek NX Expert EventFinder Settings window

Configuring expert events

The **Expert EventFinder Settings** window shows all of the available events in a table in the upper left of the window. Events are presented in a hierarchy, nested under their network layer. These layers are based on the OSI seven-layer model of networking. From top (closest to user interaction) to bottom (closest to the electrical impulses), the seven layers used by the Expert are: *Client/Server*, *Application*, *Session*, *Transport*, *Network*, *Data Link*, and *Physical*.

When you select an individual event, the rest of the **Expert EventFinder Settings** window changes to display the relevant characteristics for that event, including the descriptive and troubleshooting information and any settings. The table has three columns: ***Event***, ***Severity***, and ***Enable.***

The ***Event*** column shows the layers and, ranged under them, their events. The name of the event is expressed as a short description of the type of network event for which it tests.

The ***Severity*** column shows the level of severity of notification the Expert will send when it encounters a matching event. Click on the entry in the ***Severity*** column for any event to open a drop-down list where you can set the level of severity of these notifications. For more on notifications and their levels of severity, please see “Notifications” on page 237.

Check the checkbox in the ***Enable*** column to enable an individual event, or uncheck to disable it. You can also enable or disable all the events of a particular layer by checking or unchecking the checkbox in the ***Enable*** column for that layer. When only some events within a layer are enabled, a gray checkmark appears in the checkbox for that layer. You can also use the buttons at the top of the window to globally **Enable All** or **Disable All** events at once. To reverse the state of all events, enabling those currently disabled and disabling those currently enabled, click the **Invert Selections** button.

Event settings

The *Setting* area to the right of the event table shows the *Value* and units that mark the threshold of the condition for the selected event. In Figure 5.4, for example, the selected event, *FTP Slow Response Time*, shows a *Setting Value* of *150 milliseconds*. When this event is enabled, it will report any FTP response time greater than 150 milliseconds as an event. Note that not all events require a setting value. Some, such as *NCP Server Busy Reply,* simply check for a particular occurrence or packet type.

Threshold assistant

Many EventFinder event definitions look at characteristics of network traffic that can be expected to vary with network bandwidth. The Threshold Assistant helps you choose the right settings for these events in an intuitive way. In the example in Figure 5.4, the *Setting* for *FTP Slow Response Time* is set to *150 milliseconds*. The *Threshold Assistant* slider bar is aligned under the *Internet* marker, and the setting value of 150 milliseconds is appropriate for FTP connections over the Internet. If you move the slider bar to the left, the setting value increases, allowing for the slower FTP response times that you would expect over, for example, a *Dial-up* connection. If you move the slider bar to the right, the *Value* decreases, reflecting the faster FTP response times you would expect over a *LAN* or, further to the right, a *Fast LAN*. You can, of course, make changes to settings independent of the Threshold Assistant, but it often provides the quickest way to align several related events for optimum sensitivity and test accuracy.

Saving expert settings and restoring defaults

To restore the default setting values for an individual event, select that event in the table and click the **Restore Default** button at the top of the **Expert EventFinder Settings** window. To restore the default values to all the events, click the **Restore All Defaults** button.

You can also save and restore the entire collection of Expert EventFinder settings. To save the current Expert EventFinder settings under a new name for future use, click the **Save Expert Settings** button at the top of the window. This opens a **Save As** dialog in which you can name and choose a location for the saved settings file with its *.exp extension. To restore a previously saved group of settings, click the **Load Expert Settings** button at the top of the display. This opens a file **Open** dialog in which you can navigate to the location of and choose a settings file with a *.exp extension.

Tip You can load an alternative Expert EventFinder settings file in any particular Capture window or Packet File window and still use the default Expert EventFinder settings for all new Capture windows. Click the **Lock-in "MyExpertProfile.exp" for New Captures** button at the top of the **Expert EventFinder Settings** window to always use the settings in that particular *.exp file as the default Expert EventFinder settings for new Capture windows.

When you have made your changes to items in the **Expert EventFinder Settings** window, click the **OK** button to accept, or the **Cancel** button to reject your changes and close the window.

Expert memory allocation

When you create a new Capture window, a user-defined amount of memory is reserved for Expert analysis functions in that window. On a computer meeting the minimum system requirements, this default Expert reserved memory allocation can be set anywhere between 4 MB and 128 MB. Systems with more RAM may set a higher default Expert reserved memory allocation. When this memory is used up, the Expert begins to recycle the memory, using the methods described in the next section. You cannot change the amount of memory reserved for the Expert in an existing Capture window, as the memory is reserved when the Capture window is first created.

For complete details about how to use the ***Analysis Modules*** view of the **Options** dialog to set the default Expert reserved memory for new Capture windows, please see “Expert” on page 256.

Note: The reserved memory limitation only applies to Capture windows, not to Packet File windows. When you open a Packet File window, the Expert will consume as much memory as is required to analyze all the conversations in the saved file.

Continuous Expert use of allocated memory

In a Capture window, the Expert uses the allocated memory to hold packets, analyze conversations, display the findings in the Conversations pane, and hold entries for the Event Log. Packets are accepted, analyzed, and discarded on a continuous basis. As the number of conversations analyzed and the number of events in the Event Log grow, however, more memory is consumed by the stored results, leaving less memory for new analysis. Eventually, all the allocated memory can be consumed by the data presented in the Conversations pane and the Event Log.

When this happens, the Expert will recycle the conversations by deleting the oldest entries in the Conversations pane. The Expert will initially attempt to delete only conversations which are no longer active. If this does not free enough memory, the Expert will delete more conversations, deleting the oldest first. In very high traffic situations, even this may not free enough memory to allow the Expert to process all the packets presented in the capture buffer. In these high traffic situations, the Expert may also drop packets (that is, discard them without processing). The header section of the ***Expert*** view shows each of these parameters, in *Conversations recycled* and *Packets dropped*, respectively.

This re-use of memory allows the Expert to be used continuously, always presenting the most recent findings, and logging the results to the Event Log.

6

Peer Map

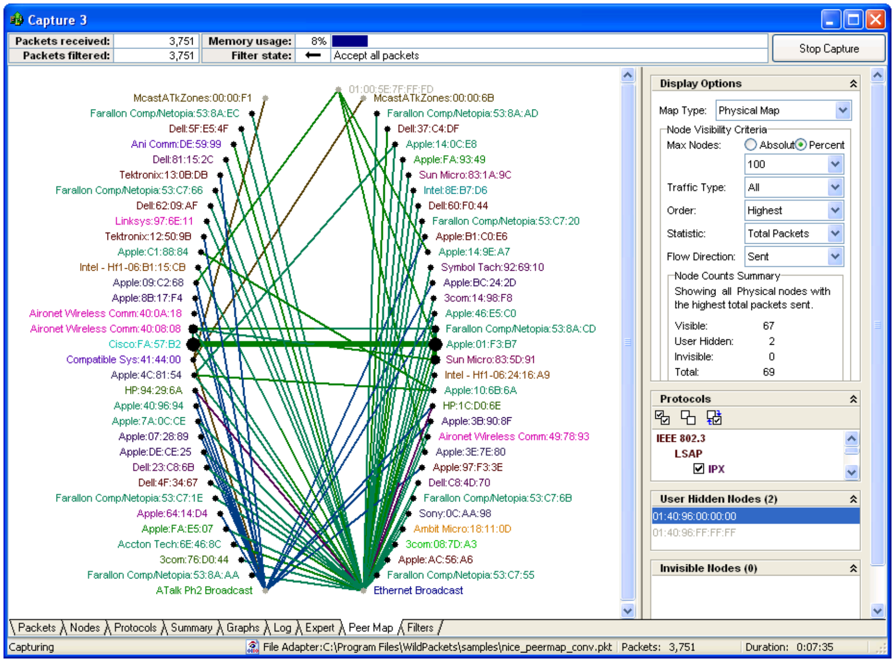
Unique to EtherPeek NX, the ***Peer Map*** view is a powerful tool for visualizing network traffic in a Packet File window or Capture window. The Peer Map uses line weight to show the volume of traffic between nodes, and uses line color to show the protocol in use between nodes. The nodes themselves can be color-coded and show icons for node type, based on Name Table data.

The ***Peer Map*** view contains its own tools to control the display of nodes and types of network traffic. This lets you quickly create a picture of all the traffic that is using a particular protocol, for example, or all the nodes sending or receiving multicast traffic.

The Peer Map displays the nodes around an elongated ellipse. Communications are shown by a line connecting each two peers. The color of the line denotes the protocol. The thickness of the line denotes the volume of traffic. When you drag nodes to new positions, the connecting lines rubber-band.

Nodes are labeled with their physical or logical address, depending on the layer you choose to view. You can optionally show nodes with their symbolic names and/or use icons to represent node types stored in the Name Table.

In this Chapter:



Capture 3

Packets received:	3,751	Memory usage:	8%
Packets filtered:	3,751	Filter state:	Accept all packets

Stop Capture

Display Options

Map Type: Physical Map

Node Visibility Criteria

Max Nodes: ○ Absolut ● Percent

100

Traffic Type: All

Order: Highest

Statistic: Total Packets

Flow Direction: Sent

Node Counts Summary

Showing all Physical nodes with the highest total packets sent.

Visible:	67
User Hidden:	2
Invisible:	0
Total:	69

Protocols

IEEE 802.3
LSAP
☑ IPX

User Hidden Nodes (2)

01:40:96:00:00:00
01:40:96:FF:FF:FF

Invisible Nodes (0)

Capturing | File Adapter:C:\Program Files\WildPackets\samples\nice_peermap_conv.pkt | Packets: 3,751 | Duration: 0:07:35

Figure 6.1 EtherPeek NX Peer Map view of a Capture window

The ***Peer Map*** view shows the Peer Map itself on the left and a series of panes on the right used to control the display of the Peer Map. The panes on the right, from top to bottom, are: *Display Options*, *Protocols*, *User Hidden Nodes*, and *Invisible Nodes*. You can collapse or expand the view of any of these panes by clicking the chevron (the double arrow) in the upper right-hand corner of the pane. You can also drag the edges of the whole area, or drag the bottom edge of any pane to resize. Each of the panes with its features and functions is described below.

Display options pane

The *Display Options* pane sets the basic parameters of the Peer Map. The *Map Type* drop-down list lets you choose whether to display nodes as a *Physical Map* (containing only physical addresses), an *IP Map* (containing only IP addresses), or an *IPX Map* (containing only IPX addresses). Note that the *Map Type* also limits the protocols which can be displayed, and changes the options in the *Protocols* pane as well. For example, choosing

IPX Address in the *Map Type* drop-down list will display only the nodes, traffic and protocols using IPX.

The *Node Visibility Criteria* section contains drop-down lists and checkboxes controlling what part of the traffic in the window’s buffer will be displayed in the Peer Map. The drop-down lists can be thought of as creating a simple description of the nodes to be displayed. In fact, such a description appears at the top of the *Node Counts Summary* section, immediately below the drop-down lists. You may see descriptions such as *Showing up to 50 unicasting IP addresses with the highest total bytes received*. The *Node Counts Summary* section also shows the number of nodes in the current view which are *Visible*, *User Hidden*, or *Invisible*, and gives the *Total*.

The *Max Nodes* text entry box lets you limit the display to no more than the specified number of nodes, expressed as an *Absolute* number or as a *Percent* of all nodes included in the Map Type for this buffer. The other parts of the *Node Visibility Criteria* section determine whether these are the nodes with the highest or the lowest values, and what aspects of network traffic to use as the test for inclusion.

The *Traffic Type* drop-down list lets you choose whether to show *All* nodes matching the other criteria, only those sending or receiving *Unicast* traffic, only those involved in *Multicast* traffic, only nodes with *Broadcast* traffic, or those with both *Multi- & Broadcast* traffic. When you choose any value other than *All* from the *Traffic* drop-down list, the nodes that do not meet your criteria are removed from the Peer Map and listed in a separate pane at the lower right of the ***Peer Map*** view called *Invisible Nodes*. If you choose *Multicast* from the *Traffic* drop-down list, for example, the *Invisible Nodes* pane will contain a list of all the nodes which neither sent nor received multicast traffic.

The *Order* drop-down list lets you choose whether you want the *Max Nodes* to represent the *Highest* or the *Lowest* values in the sample.

The *Statistic* drop-down list lets you choose the units to use when evaluating the *Max Nodes* and *Order* criteria, set above. You can choose to evaluate nodes based on their *Total Packets* or *Total Bytes*.

The last item in the *Node Visibility Criteria* section, the *Flow Direction* drop-down list, lets you choose whether to count the bytes or packets *Sent*, or those *Received*.

The three checkboxes in the *Node Appearance* area control the way in which nodes are displayed in the Peer Map. These choices are enabled when checked and disabled when unchecked.

Show Names replaces physical or logical addresses with symbolic names found in the Name Table.

Show Type Icons adds the icon appropriate to that node type (*Workstation*, *Router*, and so forth) to the display of any node that has a *Node Type* other than *Unknown* listed for it in the Name Table.

Use Colors uses the color assigned in Name Table entries to color code node names and addresses.

Protocols pane

The *Protocols* pane controls the display of the lines between the various peers in the Peer Map, which represent traffic in a particular protocol.

The *Protocols* pane shows a hierarchical list of protocols found in the Peer Map which use the address type chosen in the *Map Type* drop-down list in the *Display Options* pane above. Each of the protocols and sub-protocols has a checkbox beside it which lets you enable and disable the display of traffic in each protocol or sub-protocol independently. Each protocol has a color associated with it in ProtoSpecs. Both the entry in the *Protocols* pane and the traffic lines in the Peer Map use the same ProtoSpecs-assigned color to display each particular protocol.

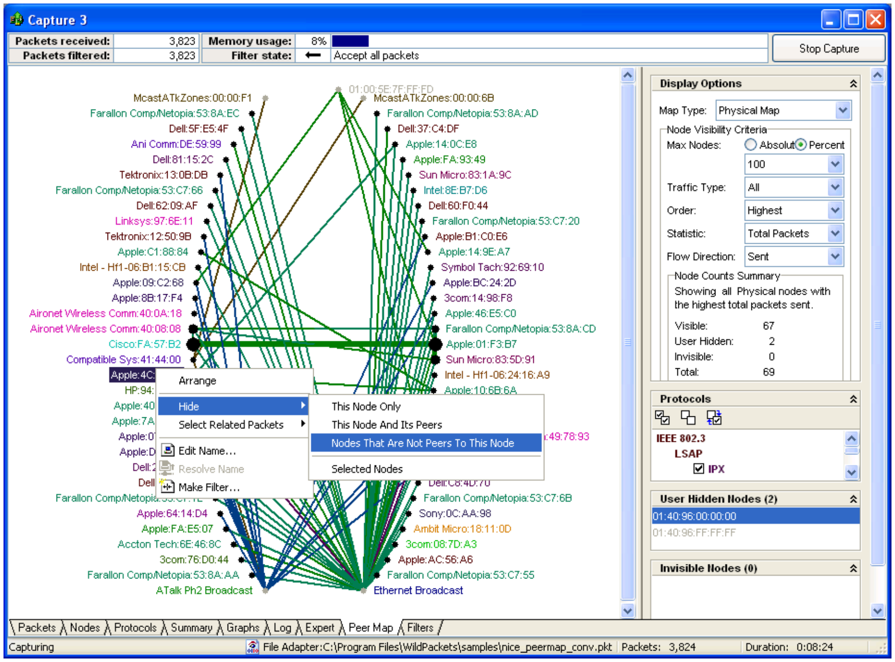
At the top of the *Protocols* pane are three buttons: **All On**, **All Off**, and **Invert All**. Click the **All On** button to enable the display of all protocols. Click the **All Off** button to disable the display of all protocols. Click the **Invert All** button to reverse the current enable/ disable choices, enabling any that were disabled and disabling any that were enabled.

Note: Some traffic, while clearly belonging to a particular network protocol such as IP, may not be assigned a sub-protocol under ProtoSpecs. When traffic of this type is present, the *Protocols* hierarchy will show an item called *Other* which includes all such sub-protocols.

User hidden nodes pane

You can temporarily remove individual nodes from the Peer Map by hiding them. The *User Hidden Nodes* pane shows a list of nodes you have removed. The number of hidden nodes is shown in the header of this pane in parentheses. From this pane, you can restore the selected (highlighted) nodes to the Peer Map by right-clicking in the pane and choosing **Show Selected Nodes** from the context menu, or restore all the hidden nodes by choosing **Show All Nodes**.

There are several ways to hide nodes. You can select one or more nodes and drag them to the *User Hidden Nodes* pane. Alternatively you can highlight one or more nodes and right-click to bring up the context menu. From the context menu, you can choose **Hide** and make a choice from the submenu, as shown in Figure 6.2. The submenu gives you the option to hide only the named node, to hide the named node and all its peers, to hide only nodes which are *not* peers of the named node, or to hide only the selected nodes.



Capture 3

Packets received: 3,823 | Memory usage: 8%
Packets filtered: 3,823 | Filter state: Accept all packets

Display Options
Map Type: Physical Map
Node Visibility Criteria
Max Nodes: ○ Absolut ● Percent
100
Traffic Type: All
Order: Highest
Statistic: Total Packets
Flow Direction: Sent
Node Counts Summary
Showing all Physical nodes with the highest total packets sent.
Visible: 67
User Hidden: 2
Invisible: 0
Total: 69

Protocols
IEEE 802.3
LSAP
☑ IPX

User Hidden Nodes (2)
01:40:96:00:00:00
01:40:96:FF:FF:FF

Invisible Nodes (0)

Capturing | File Adapter:C:\Program Files\WildPackets\samples\nice_peermap_conv.pkt | Packets: 3,824 | Duration: 0:08:24

Figure 6.2 EtherPeek NX Peer Map view showing Hide context menu submenu choices

Note: When more than one node is selected, only the node from which the context menu was invoked is named in the **Hide** submenu. That is, only the node over which the cursor was positioned when the right-click was made can be used as the basis for hiding …**Peers** or …**Not Peers**. Alternatively, you can choose **Hide All Nodes** from the background context menu. Right-click in the open area, away from all nodes, to open the background context menu. The Peer Map background context menu allows you to **Hide All Nodes**, **Arrange All Nodes**, **Resolve Names for All Nodes** (when available), or to copy the Peer Map to the clipboard.

Invisible nodes pane

The *Invisible Nodes* pane lists the nodes which have been temporarily hidden or removed from the Peer Map because they do not match the settings in the *Display Options* pane. Unlike the *User Hidden Nodes*, you cannot restore these nodes directly from the *Invisible Nodes* display. The header of the *Invisible Nodes* pane shows the number of invisible nodes in parentheses.

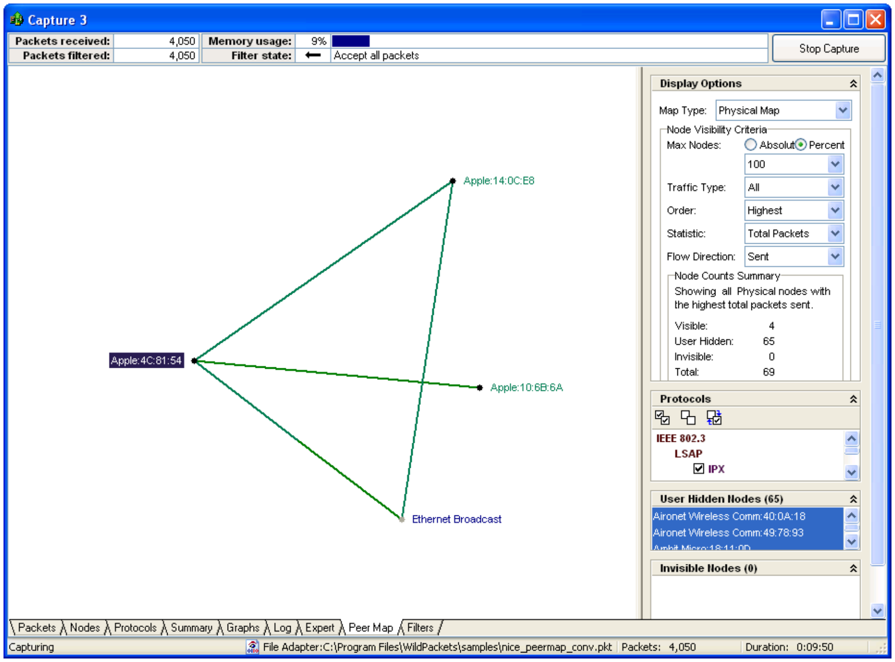
Using the peer map

The Peer Map is based on all the visible packets in the buffer of the Capture window or Packet File window, as further modified by the controls within the ***Peer Map*** view itself.

The tools for hiding and unhiding nodes described above in this chapter are particular to the Peer Map and have no effect on the ***Packets*** view or any of the other views.

Because the Peer Map reflects only the packets visible in the ***Packets*** view, you may also find it useful to switch back and forth between the ***Peer Map*** view and other views, hiding and unhiding packets to refine your picture of network traffic. When you right-click on a node in the Peer Map, the context menu allows you to **Select Related Packets**, using the current node **as Source**, **as Destination**, or **as Source or Destination**. These selection results are shown in the ***Packets*** view, as with any other **Select Related Packets** operation.

Each dot on the Peer Map represents a particular node. The size of the dot represents the packets sent from that node, as a percentage of total packets in the window. The lines between nodes represent the traffic between them. The color of the line represents the protocol. This matches the color shown for each protocol in the *Protocols* pane at the right of the ***Peer Map*** view. The thickness of the line represents the volume of the traffic. Specifically, the thickness of the line represents the volume in bytes of the traffic between two nodes, expressed as a percent of all the traffic in the buffer.



Packets received: 4,050
Packets filtered: 4,050
Memory usage: 9%
Filter state: ⟵ Accept all packets

Map Type: Physical Map

Node Visibility Criteria

Max Nodes: ○ Absolut ● Percent
100

Traffic Type: All

Order: Highest

Statistic: Total Packets

Flow Direction: Sent

Node Counts Summary

Showing all Physical nodes with the highest total packets sent.

Visible:	4
User Hidden:	65
Invisible:	0
Total:	69

IEEE 802.3
LSAP
☑ IPX

User Hidden Nodes (65)

Aironet Wireless Comm:40:0A:18
Aironet Wireless Comm:49:78:93
Ambit Micro:18:11:0D

Invisible Nodes (0)

Capturing | File Adapter:C:\Program Files\WildPackets\samples\nice_peermap_conv.pkt | Packets: 4,050 | Duration: 0:09:50

Figure 6.3 EtherPeek NX Peer Map, showing the results of hide non-peers from figure 6.2

Where screen space is limited, users may find the Peer Map is most useful when a smaller number of the most relevant nodes are displayed. Switching back and forth between various settings in the *Protocols* pane and choosing different *Traffic* options allows you to display the most interesting traffic quickly. Using the **Hide** functions from the context menu, you can further reduce the picture to only the most relevant nodes and traffic. At any time you can right-click in the white space of the Peer Map and choose **Arrange All Nodes** to restore the elliptical layout.

You can also drag nodes to clarify the picture of network traffic. You can drag a single node, or you can highlight multiple nodes and drag them all together. Use **Ctrl + Click** or **Shift + Click** to add unselected nodes to the selection, or to remove selected nodes from it. To move a single node back into the ellipse, select it and choose **Arrange** from the context menu. You can drag nodes to make any shape that suits your purpose, as shown in Figure 6.3.

Information about particular nodes

When you move the cursor over a particular node in the Peer Map, a tooltip appears containing information about that node.

Node	The label for this node in the current Peer Map. If the label is a symbolic name, the logical address is also shown in parentheses.
Type	The node type, as defined in the Name Table. For example, *Router*, *Workstation*, and so forth.
Protocols	All the protocols associated with this node in the current Map Type.
Packets Sent	Total, and percent of all packets this represents.
Packets Received	Total, and percent of all packets this represents.
Bytes Sent	Total, and percent of all traffic this represents.
Bytes Received	Total, and percent of all traffic this represents.
Identities	Other names and addresses by which this node is known.

To add any node in the Peer Map to the Name Table, right-click on the node and choose **Insert into Name Table…** from the context menu to open an **Edit Name** dialog with that node's characteristics already entered. To open an **Edit Name** dialog for any node in the Peer Map which already has a Name Table entry, choose **Edit Name…** from the context menu. When name resolution services are available, you can also choose **Resolve Name** from the context menu. For more about names, the Name Table and name resolution, see "Name table" on page 128.

To create a filter based on any node in the Peer map, right-click and choose **Make Filter** from the context menu.

Name Table

This chapter describes the Name Table in EtherPeek and its powerful tools for constructing and maintaining symbolic names for network devices and processes.

When you first start capturing packets, devices on your network will typically be identified in ***Packets*** views or in statistical displays by their logical or physical addresses. The Name Table lets you assign your own symbolic names to addresses, ports and protocols.

It is easy to create and update Name Table entries in EtherPeek. You can also save and restore (export and import) the contents of the Name Table. This allows you to keep separate Name Tables for different network segments or office locations.

EtherPeek can scan all traffic, searching for logical and symbolic names in the contents of passing packets. You can control how and whether EtherPeek adds these passively discovered names to the Name Table, and tell it how to automatically age these entries, deleting those that remain unused after a certain time.

Providing names in place of logical or physical addresses makes the task of identifying packets of interest much simpler.

In this Chapter:

Name table

The Name Table lets you assign your own symbolic names to addresses, ports and protocols. This is a simple but powerful way to make packet-related information immediately familiar and intelligible. It also lets you assign a *Node Type* to an address. This allows the program to identify nodes as a *Router* or *Server*, for example, and interpret traffic patterns accordingly. In EtherPeek NX, the Peer Map can also display nodes with the icons appropriate to the *Node Type* assigned in the Name Table.

Tip You can easily create a filter based on any entry in the Name Table. Highlight the entry and click the **Make Filter** button, or right-click and choose **Make Filter…** from the context menu. For more on creating and using filters, please see Chapter 11, “Filters” on page 195.

Adding entries to the name table

EtherPeek ships with a default Name Table. There are several ways to create new Name Table entries for your network devices. You can:

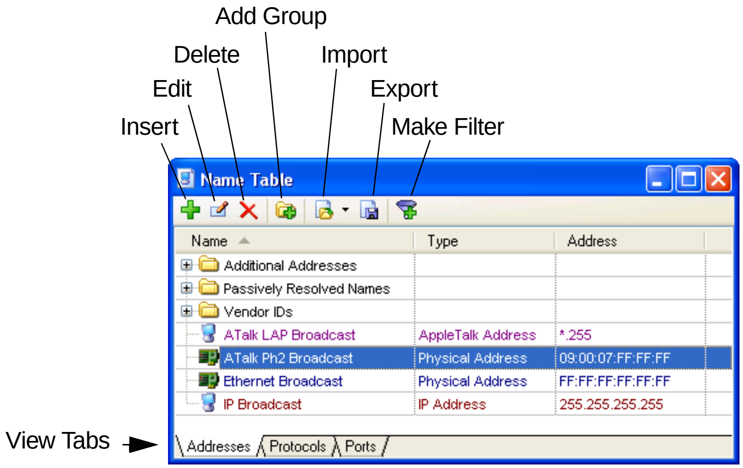
- Add names manually using the **Edit Name** dialog, displayed when you click the **Insert** button from the **Name Table** window.
- Highlight items in other views and click the **Insert Into Name Table** button, or right-click and use the **Insert Into Name Table…** command from the context menu.
- Highlight one or more items in other views and click the **Resolve Names** button, or right-click and use the **Resolve Names…** command from the context menu.
- Invoke the *Enable passive name resolution* function in the ***Name Resolution*** view of the **Options** dialog under the **Tools** menu to add WINS/NetBIOS, AppleTalk and IP names whenever EtherPeek encounters them in network traffic. This function is enabled by default.
- Use the **Import** button in the **Name Table** window to load previously saved versions of the Name Table. You can replace the whole Name Table, or add the contents of a saved Name Table to the existing one.

Figure 7.1 below shows the ***Addresses*** view of a Name Table set up with groups.

Important! Name Table entries are used in displaying packets and statistics only if **Name Table Entry** is enabled in the **Display Format** submenu of the **View** menu. There is a checkmark beside this menu item when it is enabled, as it is by default.

The name table window

The **Name Table** window has three views, accessed by clicking on the labeled tabs at the bottom of the window. The three views are: ***Addresses***, ***Protocols***, and ***Ports***.



Name	Type	Address
Additional Addresses		
Passively Resolved Names		
Vendor IDs		
ATalk LAP Broadcast	AppleTalk Address	*.255
ATalk Ph2 Broadcast	Physical Address	09:00:07:FF:FF:FF
Ethernet Broadcast	Physical Address	FF:FF:FF:FF:FF:FF
IP Broadcast	IP Address	255.255.255.255

Figure 7.1 Name Table window, Addresses view, showing Groups

Each of these views has three columns: ***Name***, ***Type***, and a third column that corresponds to the view: ***Address***, ***Protocol***, or ***Port***, respectively. ***Name*** is the symbolic name you assigned. ***Type*** is the type of address, type of port, or type of protocol. The third column shows the value that allows EtherPeek to identify the address, port, or protocol. This is written in the format of the specified type. As examples, an address of the ***Type*** *IP* will show a dotted decimal number in the ***Address*** column and a protocol of the ***Type*** *LSAP* will show the one-byte hexadecimal discriminator in the ***Protocol*** column.

Tip The Name Table allows you to sort entries in the table by the values in any column by clicking on the column headings in the **Name Table** window. A triangle appears in the column header to indicate that the display is being sorted by that column. The triangle points up if the sort is in ascending order and points down if the sort is in descending order.

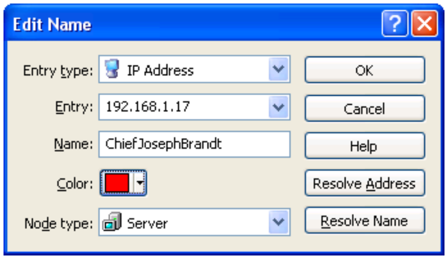
The **Name Table** window shows seven buttons. From left to right, their descriptions are shown in Table 7.1 below.

Table 7.1 **Name Table buttons**

Button	Description
Insert	This button opens the **Edit Name** dialog, in which you can enter all the parameters for the new name to be inserted in the Name Table.
Edit	When a name is highlighted, this button opens the **Edit Name** dialog with the details of the selected entry, ready to edit. When a Group is highlighted, it brings up the **Edit Group** dialog with the name of the highlighted Group ready to edit.
Delete	This button deletes the selected entry.
Add Group	This button opens the **Edit Group** dialog, in which you can name and create a new group. You can drag entries into and out of group folders. You can expand or collapse the view of group folder contents using the + plus sign or - minus sign in the left margin next to the folder icon.
Import	This button opens a dialog in which you can specify the Names file to load into the Name Table.
Export	This button opens a **Save** dialog allowing you to save the contents of the Name Table.
Make Filter	This button opens the **Edit Filter** dialog with an untitled filter matching the information in the selected Name Table entry.

Adding and editing name table entries manually

While EtherPeek offers many time-saving ways to populate the Name Table, some entries will always need to be entered by hand. Examples include symbolic names for routers and bridges, multicast addresses, loopback addresses, not well known ports, and protocols not defined in ProtoSpecs.



Entry type: IP Address

Entry: 192.168.1.17

Name: ChiefJosephBrandt

Color: 🟥

Node type: Server

Figure 7.2 Edit Name dialog

Choose **Name Table** from the **View** menu to open the **Name Table** window. To add a new entry, click the **Insert** button in the **Name Table** window. This opens the **Edit Name** dialog. To edit an existing entry, select the entry you wish to edit and click the **Edit** button. Both the **Insert** and **Edit** buttons open the **Edit Name** dialog.

To enter the complete device or protocol entry manually:

1. Open the **Edit Name** dialog.
2. Use the *Entry type* drop-down list to select the type of entry you want to add to the Name Table.

Note: The only wildcard is the asterisk (*), and it stands for zero or more alphanumeric characters. It cannot substitute for any form of punctuation.

3. Enter the numeric designation for the entity you wish to add to the Name Table in the *Entry* edit field.
4. Enter a name in the *Name* field or, if you have entered an IP address, you can use the **Resolve Address** button in this dialog to query domain name services for a name for your specified address. Alternatively, you can specify a symbolic name and EtherPeek will attempt to resolve the name to find its address when you click the **Resolve Name** button.

Note: EtherPeek actually queries name services over the network (DNS for IP addresses), so these services must be reachable for the name and address resolution functions to work properly.

5. Accept the default, or assign a new color for your Name Table entry, by clicking in the color swatch.

6. Use the *Node Type* drop-down list to set the node type for this entity, if you wish. Your choices are: *Unknown*, *Workstation*, *Server*, *Router*, *Switch*, *Repeater*, *Printer*, or *Access Point*.

Note: If the *Node Type* of a device is set to *Router* in the Name Table, EtherPeek will suppress duplicate address notifications associated with this node. If the *Node Type* is set to *Server*, the Expert function will identify this node as the server in all client/server interactions, regardless of any contrary indications contained in the packets of a particular flow or conversation.

7. Click **OK** to add the entry to the Name Table and close this dialog.

Example: adding a protocol name

To add a protocol to the Name Table:

1. Select the type of entry you want to add to the Name Table from the *Entry type* drop-down list. For example, choose *802.2 SNAP ID* to add an entry for the DECnet DNA Naming Service protocol.
2. Enter the hexadecimal representation of that protocol, *08-00-2B-80-3C,* in the *Entry* edit field.
3. Enter the name *DECnet DNA Naming Service* for the protocol in the *Name* field.
4. Assign a new color for your Name Table entry, if you wish.
5. Click **OK** to add the entry to the Name Table and close this dialog.

Note: Symbolic names assigned to protocols in the Name Table will *not* override names provided by ProtoSpecs.

Adding names from other windows

You can add to the Name Table or change name assignments for addresses by choosing device and protocol entries from a variety of other displays in EtherPeek. Basically, any window that can show individual devices can be used as a source of names for the Name Table. This includes the **Node Statistics** window, as well as the ***Packets*** and ***Nodes*** views in Capture windows or Packet File windows and **Packet Decode** windows. In the EtherPeek standard version of the program only, you can also use the ***Conversations*** view. In the EtherPeek NX version of the program only, you can also use the ***Expert*** and ***Peer Map*** views.

To add information from selected items to the Name Table:

1. Select an item in one of the appropriate views to be entered into the Name Table.

Note: Only those protocols not already identified by ProtoSpecs can be entered into the Name Table.

2. Click the **Insert Into Name Table** button, or right-click and use the **Insert Into Name Table…** command from the context menu.
3. This opens a dialog identical to the **Edit Name** dialog in form and function, but with a different dialog title. The title of the dialog that opens will depend on the nature of the item selected for insertion into the Name Table. Conversations, for example, include two addresses, each of which will be presented in turn. When you choose a packet from the ***Packets*** view for example, the first dialog that opens will be titled **Source Address**. The second will be titled **Destination Address**. In all cases, the dialog opens with the *Entry Type* and the *Entry* edit fields already filled in for the individual potential Name Table entry implied in your selection that is named in the dialog title. The *Name* field or other fields may also be filled in, depending on the information available in your selection.
4. Follow the instructions for making manual entries and edits to the Name Table given above.
5. You can only apply the **Insert Into Name Table** command to one entry at a time. If your selection presents an opportunity for adding or reviewing the settings of multiple Name Table entries, each one will be brought up in turn in a separate dialog. Click **Cancel** to close the dialog for any potential entry you do not wish to enter into the Name Table, or for any existing entries you do not want to modify.

Resolving names and addresses

EtherPeek can actively resolve IP device or host names on your network if DNS is reachable. Once names are resolved, they can be added automatically to your Name Table, where the names will be available to replace logical address entries for devices in any EtherPeek displays. Remember that name substitutions will only appear in displays if you choose the **Name Table Entry** option in the **Display Format** submenu of the **View** menu. You can set rules governing how newly discovered names and addresses are written to the Name Table using the ***Name Resolution*** view of the **Options** dialog, described in the next section.

To resolve names manually:

1. Select the nodes or packets whose addresses you wish to resolve. You can do this directly in any window that shows the individual nodes, whether it is a ***Packets*** view, a Monitor statistics window, or one of the statistics views of a Capture window or Packet File window.

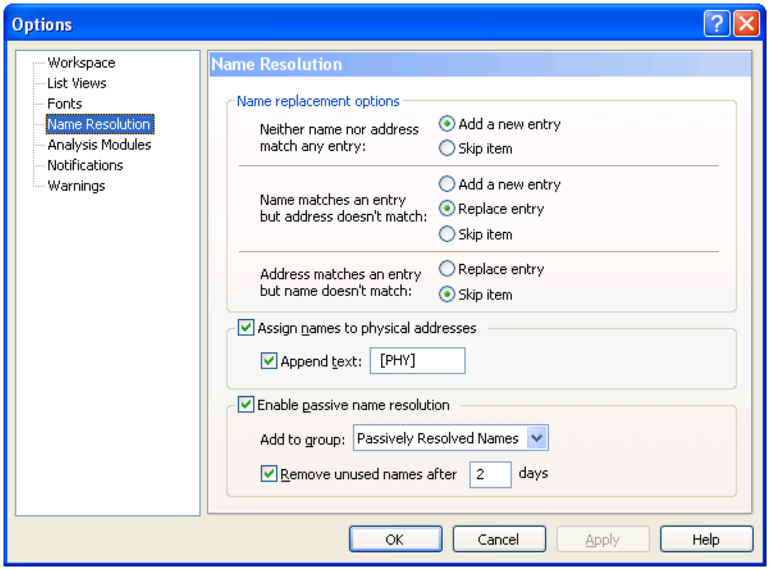
2. Click the **Resolve Names** button in the header of the window in which you've selected the items, or right-click and use the **Resolve Names…** command from the context menu.

   EtherPeek will use your network to find the names of the IP addresses of the selected packets. DNS must be reachable over the network, as EtherPeek uses this service to resolve names. Once names have been resolved, you will see name entries substituted for logical addresses in all EtherPeek displays.

   You may also look up the address of an IP name by clicking the **Resolve Name** button in the **Edit Name** dialog.

Name resolution view of the options dialog

Name and address resolution is controlled through the ***Name Resolution*** view of the **Options** dialog. Choose **Options…** from the **Tools** menu to open this dialog, and click the *Name Resolution* item in the navigation pane to open this view. Use the radio buttons in the *Name replacement options* section to determine how EtherPeek will use new information about names and addresses to automatically update the Name Table.



Options

- Workspace
- List Views
- Fonts
- Name Resolution
- Analysis Modules
- Notifications
- Warnings

Name Resolution

Name replacement options

Neither name nor address match any entry: ● Add a new entry ○ Skip item

Name matches an entry but address doesn't match: ○ Add a new entry ● Replace entry ○ Skip item

Address matches an entry but name doesn't match: ○ Replace entry ● Skip item

☑ Assign names to physical addresses

☑ Append text: [PHY]

☑ Enable passive name resolution

Add to group: Passively Resolved Names

☑ Remove unused names after 2 days

Figure 7.3 Name Resolution view of the Options dialog

Click the *Assign names to physical addresses* checkbox to automatically add names for the physical addresses found in the same packet as the logical addresses being resolved. Entries for these hardware addresses will be added to the Name Table following the same rules defined in *Name replacement options*. You may choose to add a short text string to the end of all names assigned by this function.

Note: Before resolving names and automatically assigning names to physical addresses, it is recommended that you manually add names for the physical address of intermediate link devices such as routers.

When *Enable passive name resolution* is checked, EtherPeek examines all incoming packets for symbolic names it can add to the Name Table. It adds these names according to the rules you set down in the *Name replacement options* section. Accept the default group *Passively Resolved Names*, or choose another Name Table group from the drop-down list as the location in which to put all name and address pairs discovered by passive name resolution. This is particularly useful when much of the traffic from outside the local network uses symbolic names, as Web traffic does.

In some environments, very large numbers of new names may be discovered each day through passive name resolution. Web browsing, for example, generates packet traffic containing many more unique names than just the base URLs apparent to the casual user. To keep the Name Table from becoming overgrown with unnecessary data, check the checkbox beside *Remove unused names after*, and enter a number of *days.* Names added by passive resolution will be removed from the Name Table when they go without being detected in network traffic for the specified time. If a name is encountered before its time is up, the clock for this item is restarted. In this way, you can ensure that all passively added names in the Name Table have been seen in network traffic at some time during, for example, the past two days.

Note: When you use the **Insert Into Name Table** command to add names to the Name Table, these names are not considered to have been added passively, but actively. For details, see “Adding names from other windows” on page 132.

Loading and saving name table data

You can load and save the contents of the Name Table, allowing you to keep descriptions of different segments, or to simply store and retrieve different ways of looking at the same segment.

Note: When you import items into the Name Table, a dialog asks if you want to *Delete all entries before importing?* the new names. If you click **Yes**, the imported names will be the only ones in the new Name Table, and all of the previous entries will be deleted. If you click **No**, the new names will be added to the Name Table alongside the existing entries. Only exact duplicates of existing entries will be ignored.

Loading a previously saved name table

You can load the contents of previously built and saved Name Tables, including any Name Table files you may have created manually or exported using other WildPackets analyzers.

Note: In order for EtherPeek to recognize a file as a Name Table, the file must have a *.nam file extension.

To load the names from another Name Table into the current Name Table:

1. Open the current Name Table by choosing **View** > **Name Table** from the main menu.
2. Click the **Import** button in the **Name Table** window.

Tip Alternatively, you can choose a previously used Name Table from the drop-down list beside the **Import** button.

3. In the dialog which asks if you want to *Delete all entries before importing?*, click **Yes** if you would like to replace the existing Name Table with the imported names. Alternatively, you could add the imported names to the current Name Table by clicking **No** in the warning dialog.
4. Use the resulting **Open** dialog to navigate to the location of the file you wish to load.
5. Choose this file and click **OK** to add the contents of this file to the current contents of the Name Table, or to replace the current contents, if you chose to replace the whole contents with the new Name Table file.

Tip You can also use drag and drop to add the contents of a saved Name Table (*.nam) file to the existing Name Table. Drag the Name Table file onto the open **Name Table** window. When you use this method, the names are always added to the existing Name Table.

Saving the name table

You can save all or a selected subset of the Name Table to a new file. If your work involves managing several networks, it can often be useful to build and store Name Tables for each of the networks you support. Then, when you visit each network site, you can import into the Name Table the device and protocol names relevant to that environment.

To save the entire contents of the current Name Table under a new name:

1. Open the Name Table by choosing **View** > **Name Table** from the main menu.
2. Click the **Export** button in the **Name Table** window.
3. Use the resulting **Save** dialog to name the file and choose a location in which to save it.
4. Click **OK** to save the file.

You can also save selected names from the Name Table. Group folder information is preserved when exporting either individual entries or the entire Name Table.

To save selected names from the current Name Table into a new Name Table file:

1. Open the Name Table by choosing **View** > **Name Table** from the main menu.
2. Select the entries you wish to export. You can use the standard Windows **Ctrl + click** and **Shift + click** to highlight multiple entries.
3. Right-click and choose **Export Selected…** from the context menu.
4. Use the resulting **Save** dialog to name the file and choose a location in which to save it.

5. Click **OK** to save the file.

Note: When a Name Table group folder is highlighted, the **Export Selected…** function will export the whole contents of the folder only if no individual entries within the folder are selected. If entries within the folder are highlighted, then only those highlighted entries will be exported, and not the whole contents of the folder. The group folder is preserved, whether you have selected the entire group or a single entry within it.

Log File

8

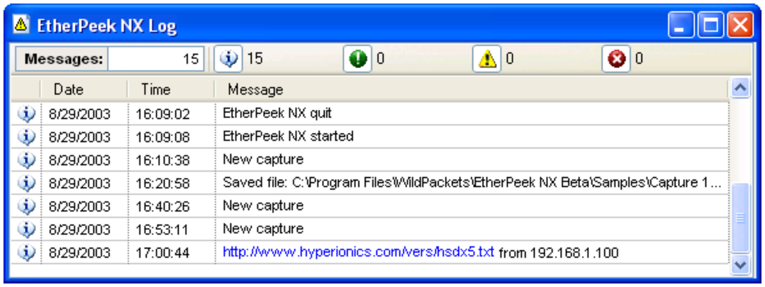
EtherPeek has a global log for the program as a whole, as well as individual log files for each Capture window and Packet File window. This chapter describes the functions of the log files.

In this Chapter:

EtherPeek log

When EtherPeek is launched, an EtherPeek Log file (called Peek.log) is created in the Application Data folder. This log is referred to in this manual as the global log file, the EtherPeek Log, or the **EtherPeek Log** window. The title of the window itself will appear as **EtherPeek Log** in the EtherPeek standard version of the program and as **EtherPeek NX Log** in the EtherPeek NX version.

Three types of events can result in items being written to this EtherPeek Log. A few events, such as the starting or stopping of EtherPeek or the creation of a new Capture window, always send a message to the EtherPeek Log. Some events, such as the writing of statistics from the ***Statistics Output*** view function, will create an entry in the EtherPeek Log if the user specifies in the function’s set-up dialog that it should do so. Other events are noted in the EtherPeek Log only when they send a notification which has as one of its actions the Log type action, as notifications of all levels of severity do by default. Analysis Modules, triggers and alarms are examples of this type. Alarms always send a notification, but the notification must have the Log type action associated with it (in the ***Notifications*** view of the **Options** dialog) in order for a message to be posted to the EtherPeek Log.



Messages: 15

	Date	Time	Message
ⓘ	8/29/2003	16:09:02	EtherPeek NX quit
ⓘ	8/29/2003	16:09:08	EtherPeek NX started
ⓘ	8/29/2003	16:10:38	New capture
ⓘ	8/29/2003	16:20:58	Saved file: C:\Program Files\WildPackets\EtherPeek NX Beta\Samples\Capture 1...
ⓘ	8/29/2003	16:40:26	New capture
ⓘ	8/29/2003	16:53:11	New capture
ⓘ	8/29/2003	17:00:44	http://www.hyperionics.com/vers/hsdx5.txt from 192.168.1.100

Figure 8.1 EtherPeek NX Log window

The header area of the **EtherPeek Log** window shows the total number of messages in the log and their breakdown by level of severity of notification (represented by their icons). You can toggle between hiding and showing the notifications of any level of severity by clicking on their icon at the top of the window.

To view the contents of the EtherPeek Log, choose the **Log Window** command from the **View** menu or press **Ctrl + L**.

The Web Analysis Module writes URLs it discovers in network traffic to the EtherPeek Log. You can access that Internet resource by double-clicking on the URL directly in the **EtherPeek Log** window. This launches your default Internet browser and opens the selected URL.

By default, the EtherPeek Log is limited to 4MB. When this limit is reached, the EtherPeek Log will delete older entries to make room for new ones. To change this upper limit, choose **Maximum Log File Size…** from the **EtherPeek Log** window context menu (right-click inside the **EtherPeek Log** window). This opens a dialog in which you can enter the new maximum size for the Log file, in kilobytes. Click **OK** to accept your changes or click **Cancel** to close the **Maximum Log File Size** dialog without making any changes.

To save the EtherPeek Log as a text file (tab-delimited or comma separated values), right-click in the **EtherPeek Log** window and choose **Save Log…** from the context menu. To copy individual lines from the EtherPeek Log to the clipboard as tab-delimited text, highlight the lines and choose **Copy** from the context menu. You can also choose to **Select All** lines by choosing that item in the context menu.

To clear or empty the EtherPeek Log, right-click in the **EtherPeek Log** window and choose **Clear Log** from the context menu.

To print the EtherPeek Log, right-click in the **EtherPeek Log** window and choose **Print Log…** from the context menu. To alter default print settings, choose **Print Setup…** from the **File** menu.

You can toggle the Auto Scroll feature of the **EtherPeek Log** window by choosing the **Auto Scroll** item from the context menu. A checkmark appears next to that item when it is enabled.

Log views of capture and packet file windows

Individual Capture windows and Packet File windows also each have a view called ***Log*** which accepts the same classes of data from the same notifications as the global EtherPeek Log (see “EtherPeek log” on page 140). There are two main differences between the global log file and the ***Log*** views of Capture windows and Packet File windows.

First, the ***Log*** view of a Capture window or a Packet File window contains only the items that are relevant to that particular window. For example, the ***Log*** view of a Capture window will show results from any enabled Analysis Modules processing just those packets that are entered into the buffer of that window. The EtherPeek Log, in contrast, contains the results from any enabled Analysis Modules processing the packets used to calculate Monitor statistics.

Second the entries in the ***Log*** view of a Capture window or Packet File window are temporary. The log is created when the window is opened and is not saved when the window is closed or saved.

The ***Log*** view of a Capture window or a Packet File window has only 128K bytes of memory under the program's default settings. Older entries are discarded to make room for new entries. You can change the default memory allocated to the log function in new Capture windows, Packet File windows, or both, Choose **Options...** under the **Tools** menu and click the *Workspace* item in the navigation pane to open the ***Workspace*** view of the **Options** dialog. In the *Advanced* section of that view, the *Capture Log size* is the default size assigned to the Log function in all new Capture windows, the *Log File size* is the default size assigned to the Log function in all new Packet File windows. Change the default *Capture Log size* and/or the *Log File size* by entering a new value in *KB* (kilobytes). Click **OK** to accept your changes.

Statistics

For monitoring, baselining, or troubleshooting network problems of all kinds, statistics are a vital tool.

EtherPeek calculates a variety of key statistics in real time. It presents these statistics in intuitive graphical displays. You can save, copy, print, and/or automatically generate periodic reports on these statistics in a variety of formats.

Node Statistics and **Protocol Statistics** offer detailed views of any item in their main displays with a double-click of the mouse. You can also create a separate graph of items in these or the **Summary Statistics** display quickly and easily. You can create snapshots of your network in **Summary Statistics** and save them for later side-by-side comparison with current conditions.

You can control how statistics are presented in each window, allowing you to quickly isolate anomalies and potential problems.

You can also set sophisticated multi-stage alarms based on most items in Monitor statistics displays. You can further key these alarms to notifications whose severity and type of response action you control. For more on Alarms, please see “Alarms” on page 231. For more on Notifications, please see “Notifications” on page 237.

This chapter describes Monitor statistics in general, then describes each type of statistic in detail. It notes differences between Monitor statistics and those found in Capture windows and Packet File windows. The chapter ends with a look at printing, saving, and other outputs of statistics.

In this Chapter:

General overview of statistics windows

Under its default settings, EtherPeek calculates Monitor statistics based on all the traffic seen on the adapter you chose in the ***Adapter*** view of the **Monitor Options** dialog. It begins doing so as soon as the program is launched, and continuously updates its statistics as long as the program is running. More precisely, when the **Monitor Statistics** item under the **Monitor** menu is enabled (as it is by default), EtherPeek analyzes all network traffic continuously in the background from the moment the program loads and the adapter for Monitor statistics is chosen until you quit the program or disable the **Monitor Statistics** item.

All packets read from the network by the Monitor statistics functions are processed and then discarded. The Monitor statistics functions of EtherPeek keep only the aggregate information needed to provide an updated tally of all the tracked parameters. Monitor statistics are not altered by filters, triggers, or any other such function. Monitor statistics are simply on or off.

Because the packets used to calculate Monitor statistics are not saved, they do not function like packets in a Capture window or Packet File window. They cannot be examined individually or used for other purposes. To actually capture packets and make them available for individual decoding, you must use a Capture window. Packet File windows and Capture windows offer most of the statistical displays found in Monitor statistics, but base their calculations on the contents of their own buffers. For more on the distinction between Monitor statistics and the statistics in Capture windows and Packet File windows, please see “Statistics in capture windows” on page 166.

Monitor Adapter

To collect Monitor statistics, you must first select an adapter to use as the source of network data for this function. By default, the program presents the ***Adapter*** view of the **Monitor Options** dialog on program start-up, so you can choose an adapter. Also by default, the program silently starts with the most recently selected adapter on all subsequent program start-ups, if that adapter is other than *File* or *None*.

To open the ***Adapter*** view, double-click on the adapter listing in the status bar at the bottom of the main program window, or choose **Monitor Options…** from the **Monitor** menu to open the **Monitor Options** dialog, then click the *Adapter* item in the navigation pane.

For a complete discussion of selecting a Monitor Adapter, please see “Selecting an adapter for monitor statistics” on page 16.

For instructions on how to set program behavior in presenting the ***Adapter*** view of the **Monitor Options** dialog on program start-up, please see “Workspace view” on page 18.

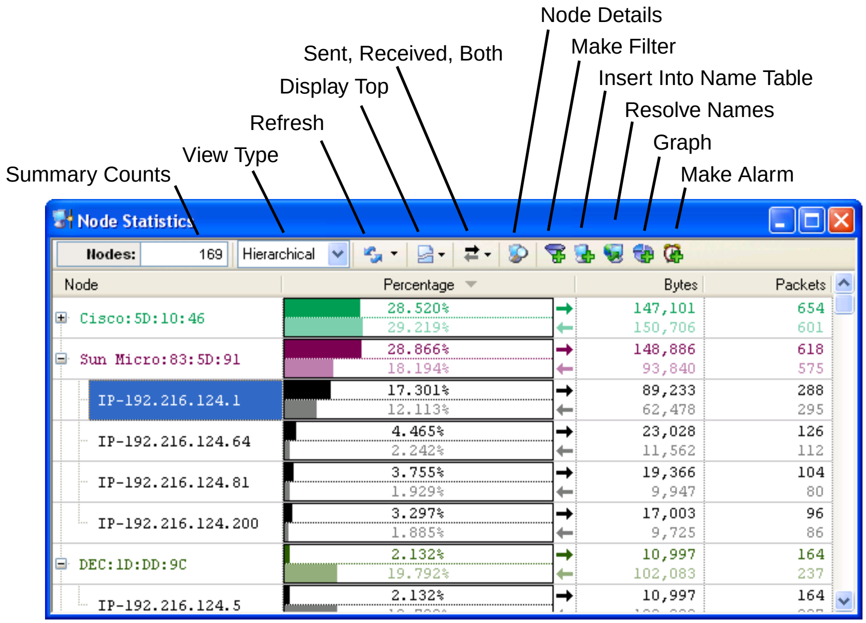
Start, stop and reset monitor statistics

By default, Monitor statistics begin calculation as soon as an adapter is selected and continue to accumulate data as long as EtherPeek is running. From the **Monitor** menu, you can change either of these defaults. Select the toggle choice labeled **Monitor Statistics** (enabled by default) to stop or start the collection of Monitor statistics. A ✓ checkmark indicates this item is enabled and Monitor statistics are being collected. Choose the item labeled **Reset Statistics** to discard all the Monitor statistics data accumulated to that moment and return all Monitor statistics displays to their zero or empty state.

In the ***Statistics Output*** view of the **Monitor Options** dialog, you can also set a schedule on which Monitor statistics are periodically reset. Please see “Statistics output views” on page 173 for details.

Statistics window headers and display controls

This section describes the various elements of statistics windows and statistics views, both for Monitor statistics and for those in Capture windows and Packet File windows. The following table (Table 9.1) describes the function of typical features of statistics windows. Please refer to Figure 9.1 for examples of most of these items.



Nodes: 169

Hierarchical

Node	Percentage	Bytes	Packets
Cisco:5D:10:46	28.520% 29.219%	147,101 150,706	654 601
Sun Micro:83:5D:91	28.866% 18.194%	148,886 93,840	618 575
IP-192.216.124.1	17.301% 12.113%	89,233 62,478	288 295
IP-192.216.124.64	4.465% 2.242%	23,028 11,562	126 112
IP-192.216.124.81	3.755% 1.929%	19,366 9,947	104 80
IP-192.216.124.200	3.297% 1.885%	17,003 9,725	96 86
DEC:1D:DD:9C	2.132% 19.792%	10,997 102,083	164 237
IP-192.216.124.5	2.132%	10,997	164

Figure 9.1 Node Statistics window, showing window element labels

Table 9.1 Statistics window elements

Element	Usage
Summary counts	Several statistics windows, including **Node**, **Protocol** (and their **Detail Statistics** windows), and **Network Statistics**; show summary counts for a few key items. **Protocol Statistics**, for example, shows the total *Protocols seen* in the upper left of the window.
View Type	The **View Type** drop-down list in the **Node Statistics** window lets you choose between a *Hierarchical* view of network nodes, in which logical addresses and symbolic names are nested beneath their physical addresses, and a variety of flat (that is, un-hierarchical) tabular displays of nodes defined by a particular address type. The column headings also change with the **View Type** choice.

Table 9.1 **Statistics window elements (continued)**

Element	Usage
Refresh rate drop-down list and button	In several statistics windows, including **Node**, **Protocol**, and **History** statistics, you can set the display refresh interval by selecting values from a drop-down list. You can click the **Refresh** button at any time to update the display. If the interval is set to *Manual*, the display will update only when you click the **Refresh** button.
Display top	For **Node** statistics, you can use the drop-down list to limit the display to the top *5*, *10, 20*, *50*, or *100* nodes seen, as measured by traffic volume. Alternatively, you can use the drop-down list to choose to display *All*.
Display Sent/Received/Both	Unique to the ***Hierarchical*** view of **Node** statistics, this drop-down list allows you to limit the display to packets *Sent*, or packets *Received*, or to show both by choosing *Sent and Received*.
Units	**History** and **Summary** statistics each have a drop-down list used to select the units in which their statistics are displayed. **History** statistics can be displayed as a percent of network bandwidth *Utilization*, or as *Bytes/second* or *Packets/second*. **Summary** statistics can be displayed in either of these last two units, or in *Packets*, *Bytes*, or a percentage of either. Other statistics windows present information in a variety of units within a single display.
Snapshot button	Unique to the **Summary Statistics** window, the **Snapshot** button saves the current statistics values for side by side comparison with future values.
Detail button	Opens **Detail Statistics** windows for all selected items. Available for **Node** and **Protocol Statistics** windows.
Pause button	Operates as a toggle to temporarily suspend scrolling or screen re-draw due to data update in the statistics list or graph. Available for **Size**, **Summary**, and **History Statistics** windows. This button is also used in all statistics **Graph** windows.

Table 9.1 Statistics window elements (continued)

Element	Usage
Graph button	Opens the **Graph Data Options** dialog to create a graphical representation of the selected item. Please see “Creating and controlling graph windows” on page 182 for more details.
Alarm button	Opens the **Make Alarm** dialog to define the parameters for establishing and resolving alarm conditions based on the selected statistics item. Available for **Node**, **Protocol**, and **Summary Statistics** windows. Please see “Alarms” on page 231 for more details.

Display options for statistics windows

In the ***List Views*** view of the **Options** dialog, you can customize background color and the style of vertical and horizontal lines in all list displays. In the ***Fonts*** view of the **Options** dialog, you can specify the font and style of the data text in all views of the program. To change these and other default aspects of window display, use the **Options** dialog, available by choosing **Options…** under the **Tools** menu.

You can also sort, collapse or expand statistics displayed as lists or tables, and change the way colors are applied to various elements of statistics displays. These features are described in the following sections.

Sorting, collapsing and expanding lists

You can change the sort order of statistics presented in a table (**Node** and **Protocol**), and collapse or expand those listed in a hierarchy (**Node**, **Protocol**, and **Summary**). To change the sort order of any list of statistics, click in the heading of the column by which you want to sort the display. Click in the column header again to toggle between ascending and descending order. A triangle in the column header indicates the sort order. In the ***Hierarchical*** view of the **Node Statistics** window, you can use the drop-down list to choose whether to display statistics about packets *Sent*, *Received*, or both.

Note: Hierarchical lists are sorted within their own level of the hierarchy.

To expand or collapse individual groups in hierarchical lists, click on the + plus or - minus sign in the left margin beside any group entry. Right-click to bring up a context menu with options to **Collapse All** or **Expand All** hierarchical items.

Controlling color in statistics lists

The **Color** sub-menu of the **View** menu determines how colors *already assigned in other dialogs* will be used in displaying data in the ***Hierarchical*** view of **Node Statistics**. There are only two sources of color assignments for elements of network traffic in EtherPeek that have an effect on the **Node Statistics** display:

- The **Edit Name** dialog in the **Name Table** can set the color for packets associated with a particular address (node), port, or protocol.
- ProtoSpecs assign colors to all the protocols they know how to identify, and their color choices cannot be overridden.

The **Color** sub-menu of the **View** menu uses the color information from these other sources, and applies it to the display of nodes and protocols in statistics lists. For more about how colors are assigned to packet lists and statistics displays, please see “Color display options” on page 79.

Monitor statistics

You can open any or all Monitor statistics windows from the **Monitor** menu: **Node**, **Protocol**, **Network**, **Size**, **Summary**, and/or **History Statistics**. Each of these is described in detail in this section.

Tip All of the Monitor statistics windows can be displayed at the same time. However, if they are all displaying information in real-time during capture and the network is very busy, EtherPeek might not have enough time to process captured packets. This can cause statistics to lag behind actual network activity or cause packets to be dropped.

Node statistics

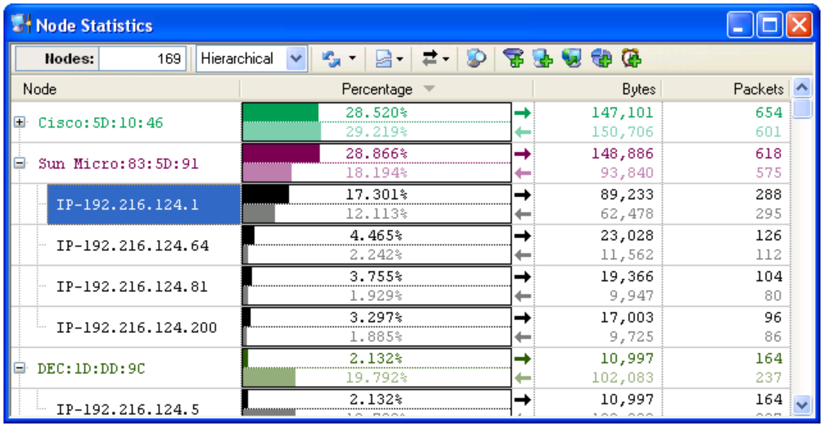
To open the **Node Statistics** window, choose **Nodes** from the **Monitor** menu or press **Ctrl + 1**. The **Node Statistics** window displays real-time data organized by network node.

The *View Type* drop-down list in the **Node Statistics** window lets you choose between a *Hierarchical* view of network nodes (in which logical addresses are nested beneath their physical addresses), and a variety of flat (that is, not hierarchical) tabular displays of nodes defined by a particular address type. The column headings also change with the *View Type* choice.

Hierarchical view of node statistics

The ***Hierarchical*** view shows network nodes or devices identified by their physical address, with any associated logical addresses nested underneath. The header of the window shows a count of the total network *Nodes* seen. For each node and unique address, the ***Hierarchical*** view can present information about traffic sent, received, or both, depending on your selection from the **Sent, Received, Both** drop-down list in the window header. For each line, the ***Hierarchical*** view shows the total ***Bytes*** and ***Packets***, plus a ***Percentage*** column showing graphically and numerically the total bytes for this line, expressed as a percentage of total bytes for all lines in the ***Hierarchical*** view.

Use the drop-down lists at the top of the window to control the display. These items are labeled in Figure 9.1, and Table 9.1 describes how to use each of these elements to control the display of statistics and other functions.



Nodes: 169

Hierarchical

Node	Percentage	Bytes	Packets
Cisco:5D:10:46	28.520% 29.219%	147,101 150,706	654 601
Sun Micro:83:5D:91	28.866% 18.194%	148,886 93,840	618 575
IP-192.216.124.1	17.301% 12.113%	89,233 62,478	288 295
IP-192.216.124.64	4.465% 2.242%	23,028 11,562	126 112
IP-192.216.124.81	3.755% 1.929%	19,366 9,947	104 80
IP-192.216.124.200	3.297% 1.885%	17,003 9,725	96 86
DEC:1D:DD:9C	2.132% 19.792%	10,997 102,083	164 237
IP-192.216.124.5	2.132%	10,997	164

Figure 9.2 Node Statistics window

The ***Node*** column shows a hierarchical address list showing the physical address and any associated logical addresses (or their symbolic names) for each node being monitored. To set the window to show only the nodes generating or receiving the most traffic, select a value from the drop-down list at the top of the window labeled *Display top*. You can choose to display the top *5*, *10*, *20*, *50*, or *100* nodes; or you can choose to display *All*.

The ***Node*** address list can be set to look at the Name Table and replace physical or logical addresses with the symbolic names (and associated colors) stored there. To toggle the **Node Statistics** display's use of the Name Table, go to the **View** menu, pull down

Display Format and choose **Name Table Entry**. A checkmark appears beside the choice when it is enabled.

The ***Percentage*** bar graph represents the bytes sent (top bar) and/or received (bottom bar) by each node. Use the drop-down list at the top of the window to display only *Sent*, only *Received*, or display both by choosing *Sent and Received*.

The ***Bytes*** column shows the total bytes, sent and/or received, for each node. The ***Packets*** column displays the number of packets, sent and/or received, for each node.

To change the sort order of any list of statistics, click in the heading of the column by which you want to sort the display. Click in the column header to toggle between ascending and descending order. The sort order is indicated by a small triangle pointing up or down, shown in the header of the column by which the display is sorted.

If you intend to keep the window open for some length of time, you may want to select a longer refresh interval. You can set the refresh interval for this window by using the drop-down list at the top of the display. This applies only to refresh of the display, as calculation goes on continuously in the background. Nevertheless, longer refresh intervals do save processing time for other tasks, such as processing packets. Click the **Refresh** button at any time to manually refresh the display.

Flat views of node statistics

In addition to the ***Hierarchical*** view, the **Node Statistics** window can present data in a variety of flat tables which list nodes of a particular type in the left-most column and data about the traffic of those nodes in a series of columns to the right. These flat views each correspond to one particular protocol or address type. The columns shown in the **Node Statistics** window change to match the view type. The available flat view types for **Node Statistics** are: ***Physical***, ***IP***, ***IPv6***, ***AppleTalk***, ***DECnet***, and ***IPX***.

Table 9.2 lists and describes the columns common to all of the flat view types and notes for each whether it is present by default. To change which columns are visible in any particular flat table view of the **Node Statistics** window, right-click in any column header to bring up a list of all available columns. Visible columns show a checkmark beside them. Click on any column name to toggle its state between shown and not shown.

Table 9.2 Columns for all node statistics flat view types

Default	Column	Description
X	***Node***	The address or name of the node, in the format appropriate to the view type.
X	***Total Bytes***	Total bytes sent and received by this node.
	Bytes Sent	Total bytes sent by this node.
	Bytes Received	Total bytes received by (or addressed to) this node.
	Total Packets	Total packets sent and received by this node.
X	***Packets Sent***	Total packets sent by this node.
X	***Packets Received***	Total packets received by (or addressed to) this node.
	Broadcast Packets	Total broadcast packets sent by this node.
	Broadcast Bytes	Total broadcast bytes sent by this node.
	Multicast Packets	Total multicast packets sent by this node.
	Multicast Bytes	Total broadcast and multicast packets sent by this node.
X	***Broadcast/Multicast Packets***	Total broadcast and multicast packets sent by this node.
	Broadcast/Multicast Bytes	Total multicast packets sent by this node.
	Min. Size Sent	The size of the smallest packet sent by this node.
	Max. Size Sent	The size of the largest packet sent by this node.
	Avg. Size Sent	The average size of the packets sent by this node.
	Min. Size Received	The size of the smallest packet received by this node.

Table 9.2 **Columns for all node statistics flat view types (continued)**

Default	Column	Description
	Max. Size Received	The size of the largest packet received by this node.
	Avg. Size Received	The average size of the packets received by this node.
	First Time Sent	Time stamp of the first packet sent by this node.
	Last Time Sent	Time stamp of the most recent packet sent by this node.
	First Time Received	Time stamp of the first packet received by this node.
	Last Time Received	Time stamp of the most recent packet received by this node.
	Duration	The difference between the time stamp of the earliest sent or received packet and that of the most recent sent or received packet.

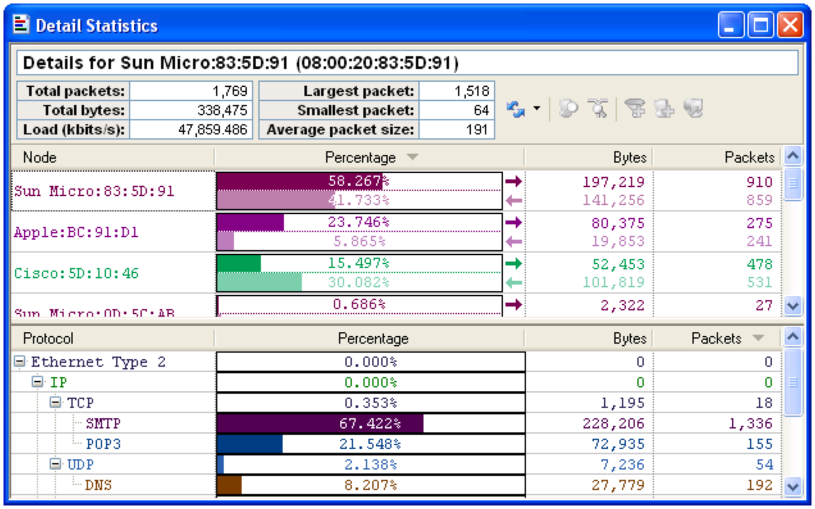
Viewing details for a network node

Double-click the entry to see more detail about the activity for the selected node and the protocols they are using. A window similar to that shown in Figure 9.3 opens.

The additional detail includes:

- Details of communications partners for this node.
- A hierarchical list of protocols used by this node and its communications partners. For details on display conventions, see “Protocol utilization statistics” on page 157.
- The *Total packets* and *Total bytes* for this node.
- Network *Load (kbits/s)* attributed to this node.
- *Largest packet*, *Smallest packet* and *Average packet* size for the specific node or protocol.

Click the **Refresh** button to update the display. Alternatively, you can use the **Refresh** drop-down list to set a refresh interval for the **Detail Statistics** window.



Details for Sun Micro:83:5D:91 (08:00:20:83:5D:91)

Total packets:	1,769	Largest packet:	1,518
Total bytes:	338,475	Smallest packet:	64
Load (kbits/s):	47,859.486	Average packet size:	191

Node	Percentage		Bytes	Packets
Sun Micro:83:5D:91	58.267% 41.733%	→ ←	197,219 141,256	910 859
Apple:BC:91:D1	23.746% 5.865%	→ ←	80,375 19,853	275 241
Cisco:5D:10:46	15.497% 30.082%	→ ←	52,453 101,819	478 531
Sun Micro:0D:5C:AB	0.686%	→	2,322	27

Protocol	Percentage	Bytes	Packets
⊟ Ethernet Type 2	0.000%	0	0
⊟ IP	0.000%	0	0
⊟ TCP	0.353%	1,195	18
SMTP	67.422%	228,206	1,336
POP3	21.548%	72,935	155
⊟ UDP	2.138%	7,236	54
DNS	8.207%	27,779	192

Figure 9.3 Node Detail Statistics window.

Note: Node **Detail Statistics** windows show both sent and received traffic, regardless of the settings in the main **Node Statistics** window.

Protocol statistics

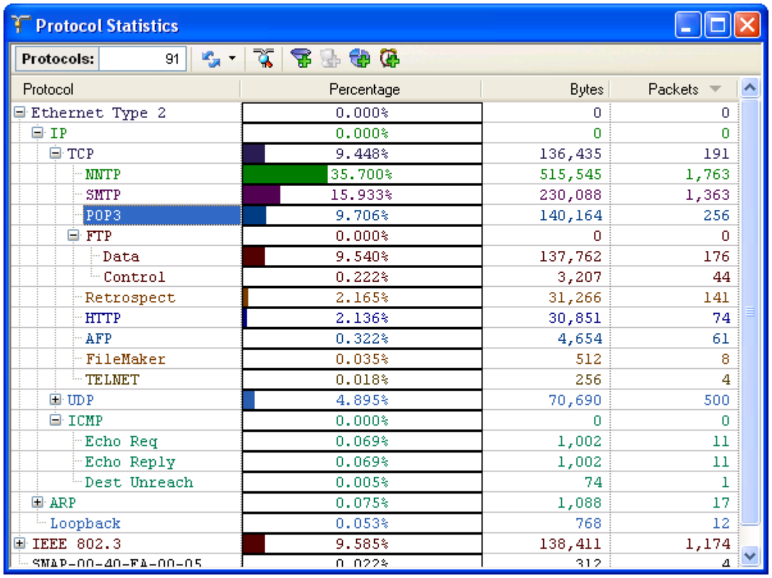
To open the **Protocol Statistics** window, choose **Protocols** from the **Monitor** menu or press **Ctrl + 2**.

The **Protocol Statistics** window shows network traffic volume, in packets and in bytes, broken down by protocol and sub-protocol.

This window is useful in determining which protocols or sub-protocols are generating a high percentage of the overall network traffic.

The ***Percentage*** bar graph represents the percentage of bytes for each protocol and sub-protocol type.

The ***Bytes*** column shows the total bytes used by that protocol. The ***Packets*** column displays the number of packets transmitted and received by all nodes combined for that protocol.



Protocols: 91

Protocol	Percentage	Bytes	Packets
Ethernet Type 2	0.000%	0	0
IP	0.000%	0	0
TCP	9.448%	136,435	191
NNTP	35.700%	515,545	1,763
SMTP	15.933%	230,088	1,363
POP3	9.706%	140,164	256
FTP	0.000%	0	0
Data	9.540%	137,762	176
Control	0.222%	3,207	44
Retrospect	2.165%	31,266	141
HTTP	2.136%	30,851	74
AFP	0.322%	4,654	61
FileMaker	0.035%	512	8
TELNET	0.018%	256	4
UDP	4.895%	70,690	500
ICMP	0.000%	0	0
Echo Req	0.069%	1,002	11
Echo Reply	0.069%	1,002	11
Dest Unreach	0.005%	74	1
ARP	0.075%	1,088	17
Loopback	0.053%	768	12
IEEE 802.3	9.585%	138,411	1,174
SNAP-00-40-FA-00-05	0.022%	312	4

Figure 9.4 Protocol Statistics window

The *Protocols* item at the top of the display shows the total number of different protocols encountered.

To change the sort order, click in the heading of the column by which you want to sort the display to toggle between ascending and descending order. The order is indicated by a small triangle pointing up or down, shown in the header of the column by which the display is sorted.

If you intend to keep the window open for some length of time, you may want to select a longer refresh interval. To set the refresh interval for this window, use the drop-down list at the top of the display. This applies only to refresh of the display, as calculation goes on continuously in the background. Longer refresh intervals save resources for other tasks, such as processing packets. You can click the **Refresh** button in the window header at any time to immediately refresh the display.

ProtoSpecs™

ProtoSpecs™ is an exclusive feature that quickly and accurately identifies the protocols nested within Ethernet packets.

ProtoSpecs use multiple identifiers within a packet to create a tree-structure that specifies a top-level or parent protocol (such as IP) and sub-protocols that it contains (such as FTP or SNMP). The protocol list in the **Protocol Statistics** window uses a hierarchical structure. Click the + plus sign or - minus sign preceding a name to expand or collapse the selection, or right-click to access the context menu, where you can choose to **Expand Selection**, **Collapse Selection**, **Expand All**, or **Collapse All**.

ProtoSpecs recognize hundreds of different protocols and sub-protocols. Nevertheless, there are still some protocols that are not identified by name in the program. EtherPeek will list unidentified Ethernet Type 2 (two-byte), LSAP (one-byte) and SNAP (five-byte) protocol types by their numeric value in hexadecimal. You may add these to the Name Table to assign them a symbolic name.

When EtherPeek cannot identify a sub-protocol, it lists the protocol with other unidentified types at the highest known protocol level. For example, UDP port 1378, which is reserved for the Elan License Manager, is not uniquely identified by EtherPeek. Instead, the packet statistics associated with this protocol are collected under the identified name of UDP protocol statistics.

For a more detailed look at protocols: what they are, how they work, and how they are handled in EtherPeek, see Appendix A, “Packets and Protocols” on page A-3.

You can add new protocol discrimination definitions to the ProtoSpecs hierarchy. Instructions can be found in a file called ProtoSpecsXML.pdf, located in the 1033\Documents\Peek SDK directory under the directory in which you installed EtherPeek.

Note: ProtoSpecs protocol discriminators test for particular values at specified locations (offset, or offset and mask) within packets. They also rely on the hierarchical relationship between protocols (encapsulation) for proper functioning. Writing protocol discriminators requires a good understanding of protocol characteristics and packet structure, as well as some knowledge of XML syntax.

Protocol information

For a quick refresher on the meaning and usage of a particular protocol or sub-protocol, highlight the protocol in any window where it is shown, right-click and choose **Protocol Info…** from the context menu. Brief descriptions of hundreds of protocols and sub-protocols are stored here for ready reference.

Protocol utilization statistics

When the hierarchical view is collapsed (with a plus sign + in front of the protocol name), the utilization statistics show the sum of all sub-protocols within that protocol. When the hierarchical view is expanded (with a minus sign - in front of the protocol name), utilization statistics are broken out by individual sub-protocol. The top-level protocol (such as IP) then shows statistics only for itself and for any sub-protocols that seem to be a part of the top-level protocol, but that are not uniquely defined by ProtoSpecs. Statistics that do not belong to any of the recognized sub-protocols are added to the totals for the parent protocol. This allows statistics for unrecognized sub-protocols to be included in the totals with as much precision as possible.

Viewing details for a protocol

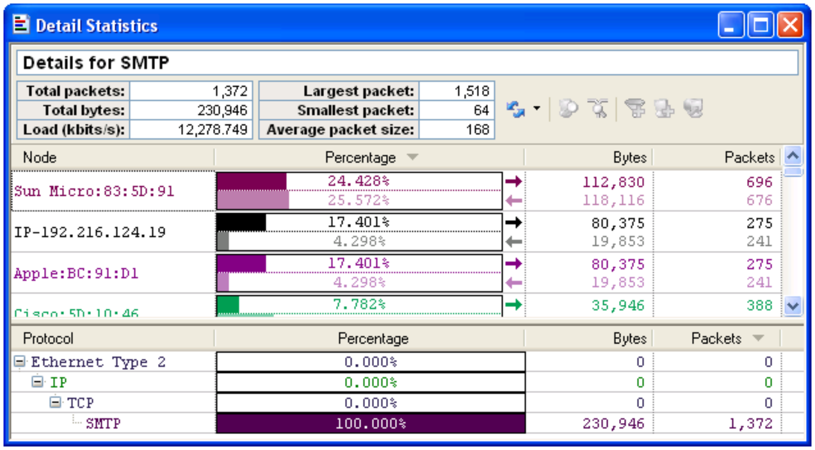
To view more detail about the traffic in a particular protocol or sub-protocol, double-click the protocol or sub-protocol name. This opens a **Detail Statistics** window.

This window displays more detail about nodes generating the selected protocol. The additional detail includes:

- Details for nodes communicating in this protocol (and its sub-protocols, if any).
- The relative percentage of traffic represented by any sub-protocols.
- The *Total packets* and *Total bytes* of traffic for this protocol.
- Network *Load (kbits/s)* used by the protocol (and its sub-protocols, if any).
- *Largest packet*, *Smallest packet* and *Average packet size* for the protocol.

The bar graph in this detail window lists all nodes receiving or sending packets of the selected protocol type, their respective percentage share of the protocol traffic, and the number of packets that percentage represents.

Click the **Refresh** button to update the display. Alternatively, you can use the **Refresh** drop-down list to set a refresh interval for the **Detail Statistics** window.



Details for SMTP

Total packets:	1,372	Largest packet:	1,518
Total bytes:	230,946	Smallest packet:	64
Load (kbits/s):	12,278.749	Average packet size:	168

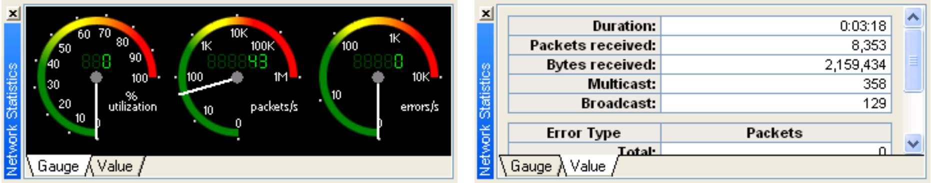
Node	Percentage	Bytes	Packets
Sun Micro:83:5D:91	24.428% 25.572%	112,830 118,116	696 676
IP-192.216.124.19	17.401% 4.298%	80,375 19,853	275 241
Apple:BC:91:D1	17.401% 4.298%	80,375 19,853	275 241
Cisco:5D:10:46	7.782%	35,946	388

Protocol	Percentage	Bytes	Packets
Ethernet Type 2	0.000%	0	0
IP	0.000%	0	0
TCP	0.000%	0	0
SMTP	100.000%	230,946	1,372

Figure 9.5 Protocol Detail Statistics window

Network statistics

To open the **Network Statistics** window, choose **Network** from the **Monitor** menu or press **Ctrl + 3**.



Duration:	0:03:18
Packets received:	8,353
Bytes received:	2,159,434
Multicast:	358
Broadcast:	129

Error Type	Packets
Total:	0

Figure 9.6 Network Statistics window, Gauge and Value views

The default ***Gauge*** view of the **Network Statistics** window shows network utilization (as a percent of capacity), traffic volume (in packets per second), and error rate (total errors per second) as analog dials with corresponding digital displays at their centers.

The ***Value*** tab at the bottom of the window opens an alternate view showing two tables. The first shows duration, traffic volumes and utilization. The lower table shows counts of error packets, both total and by each of the four error types. The upper table lists five parameters that show total counts from the time EtherPeek began collecting Monitor statistics to the current second. They are:

Duration This parameter shows elapsed time in “days: hours: minutes: seconds:” format since you started collecting Monitor statistics.

Packets received This parameter shows packets received since you started collecting Monitor statistics.

Bytes received This parameter shows bytes received since you started collecting Monitor statistics.

Multicast This parameter shows packets addressed to multicast addresses since you started collecting Monitor statistics.

Broadcast This parameter shows packets addressed to broadcast addresses since you started collecting Monitor statistics.

The lower table in the ***Value*** view of the **Network Statistics** window shows error counts for *Total Errors* and for each of the four types individually since you began collecting Monitor statistics.

Error types and error packets

EtherPeek recognizes four error types, shown in the table below:

Table 9.3 **Error Types**

Error Type	Description
CRC Error	At the end of the packet, four bytes are transmitted which force the checksum to a known constant. If the recipient does not compute the same constant after receiving the four bytes, the packet must have been corrupted. A CRC error occurs when the CRC (Cyclic-Redundancy Check) fails. These bytes are referred to as a Frame Check Sequence or FCS.
Frame Alignment Error	Each byte is transmitted onto Ethernet a bit at a time, and the Ethernet receiver hardware collects the bits back into 8-bit bytes. A Frame error is detected at the end of a packet when the number of bits received is not a multiple of eight, that is, when the number of bits does not collect evenly into a number of 8-bit bytes.

Table 9.3 Error Types (continued)

Error Type	Description
Runt Packet	A Runt packet is a packet which is at least 8 bytes but fewer than 64 bytes long, and is otherwise well-formed.
Oversize Packet	By definition, ordinary Ethernet packets must be between 64 and 1,518 bytes long. For VLAN-tagged packets (802.1Q/802.3ac), the maximum length is 1522 bytes. For packets tagged as Jumbo packets, the maximum size is 9022 bytes. A packet is Oversize when its length is greater than the limits appropriate to its packet type, and it is otherwise well-formed.

Note: Some network adapter and driver configurations report none or only some of these error statistics. Please see "Ethernet interface requirements" on page 10 for details.

When EtherPeek captures error packets, they are treated exactly like any other packet except they are flagged as an error. However, any data in an error packet, including the source and destination physical addresses, should be viewed with caution since it may have little correspondence to what was originally transmitted.

Size statistics

To open the **Size Statistics** window, choose **Size** from the **Monitor** menu or press **Ctrl + 4**.

The *Packet Size Distribution* graph sets up size classes for packets (their length in bytes) and shows what percentage of the packets on the network are in each size class.

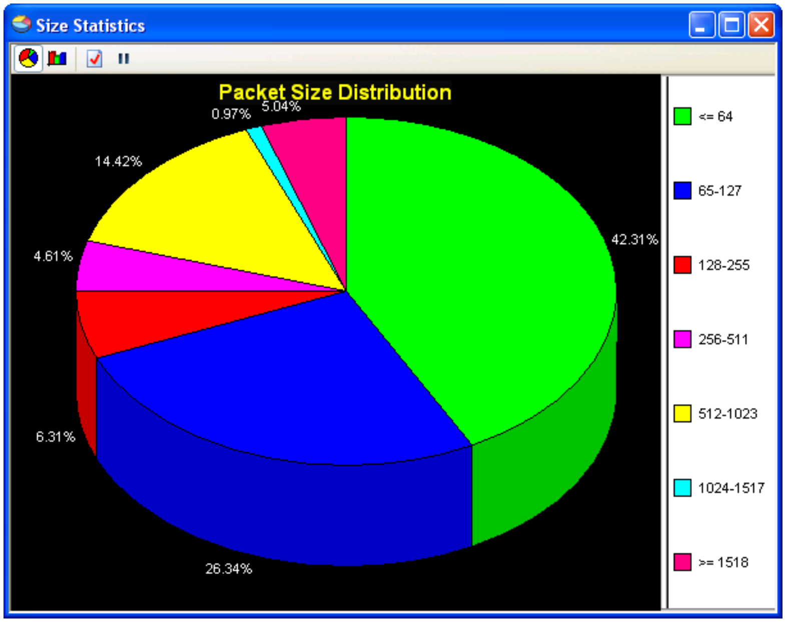
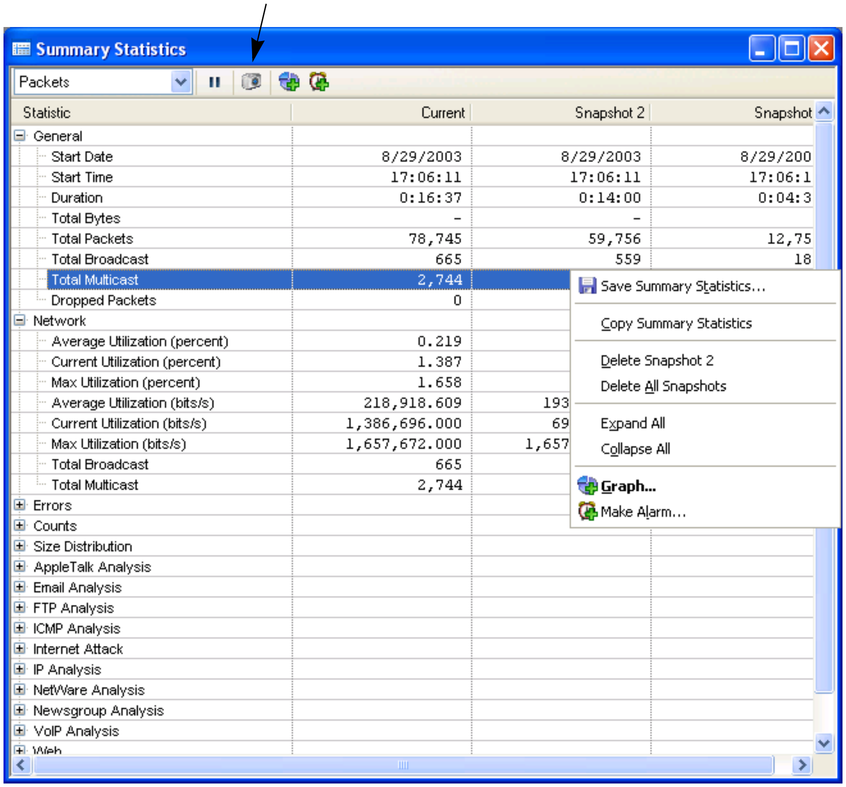


Figure 9.7 Size Statistics window

You can choose a pie chart or a bar chart display format by clicking the **Pie** chart or the **Bar** chart button in the upper left hand corner of the **Size Statistics** window. Click the **Options** button to choose additional options for color, borders, and three-dimensional or two-dimensional display. Click the **Pause** button to temporarily suspend chart updates.

Summary statistics

To open the **Summary Statistics** window, choose **Summary** from the **Monitor** menu or press **Ctrl + 5**.



Statistic	Current	Snapshot 2	Snapshot
General			
Start Date	8/29/2003	8/29/2003	8/29/200
Start Time	17:06:11	17:06:11	17:06:1
Duration	0:16:37	0:14:00	0:04:3
Total Bytes	-	-	
Total Packets	78,745	59,756	12,75
Total Broadcast	665	559	18
Total Multicast	2,744		
Dropped Packets	0		
Network			
Average Utilization (percent)	0.219		
Current Utilization (percent)	1.387		
Max Utilization (percent)	1.658		
Average Utilization (bits/s)	218,918.609	193	
Current Utilization (bits/s)	1,386,696.000	69	
Max Utilization (bits/s)	1,657,672.000	1,657	
Total Broadcast	665		
Total Multicast	2,744		
Errors			
Counts			
Size Distribution			
AppleTalk Analysis			
Email Analysis			
FTP Analysis			
ICMP Analysis			
Internet Attack			
IP Analysis			
NetWare Analysis			
Newsgroup Analysis			
VoIP Analysis			
Web			

Figure 9.8 Summary Statistics window, showing context menu for Snapshot 2

The **Summary Statistics** window allows you to monitor key network statistics in real time and save those statistics for later comparison. To create a new Summary Statistics Snapshot, click the **Snapshot** button at the top of the window. The new column labeled *Snapshot 1* will appear immediately to the right of the column labeled *Current*. As you take additional snapshots, any previous snapshots will be pushed further to the right, so that the most recent (highest numbered) is next to the current statistics. If you had three snapshots, for example, the columns, reading from left to right, would be named *Current*, *Snapshot 3*, *Snapshot 2*, *Snapshot 1*. To delete a particular snapshot, right-click in the column you wish to delete and choose **Delete Snapshot #** (where # is the number of the particular snapshot). Alternatively, you can choose **Delete All Snapshots** to clear all.

Use the snapshot feature to baseline normal network activity, save the data as a snapshot, and then compare these saved statistics with those observed during periods of erratic network behavior to help pinpoint the cause of the problem.

Summary statistics are also extremely valuable in comparing the performance of two different networks or network segments. For example, a field support engineer could compare the real-time statistics on a client's network with a saved healthy snapshot and easily diagnose or eliminate the source of inconsistent or poor performance.

Click on the plus sign + or minus sign - in the margins beside the major headings to expand or collapse the view of that section of the hierarchy. Details are hidden when the hierarchy is collapsed and no summary of those hidden details is provided at higher levels. Right-click to bring up a context menu with options to **Expand All** or **Collapse All** hierarchical items.

To set the display units for the **Summary Statistics** window, choose from the drop-down list in the upper left. Your choices are: *Packets*, *Bytes*, *Percent of Packets*, *Percent of Bytes*, *Packets per second*, or *Bytes per second*.

Many of the statistics reported in **Summary Statistics** are provided by Analysis Modules and the Expert. These functions can be enabled or disabled individually (globally in the Analysis Modules view of the Options dialog, and for Monitor statistics in particular in the ***Performance*** view of the **Monitor Options** dialog). These functions must be enabled (as most of them are by default) in order to be presented in the **Summary Statistics** window.

Tip When you have a supported adapter selected, the **Summary Statistics** window also displays *Driver* statistics detailing performance, as reported by the driver for your adapter. This is available for Monitor statistics and in the ***Summary*** view of Capture windows.

History statistics

To open the **History Statistics** window, choose **History** from the **Monitor** menu or press **Ctrl + 6**.

The **History Statistics** window shows a graph of network performance at selected intervals over time. You can choose to measure that performance as *Utilization* (percent of capacity as set in the **Network Speed** dialog), or as *Packets/second* or *Bytes/second* by choosing from the drop-down list in the upper left of the **History Statistics** window, as shown in Figure 9.9. The scale at the left can be fixed, or it can be dynamically adjusted to cover only the range of values encountered so far.

You can choose how the historical data is displayed by selecting a sampling interval from the drop-down list. The drop-down list sets the displayed sampling interval for the **History Statistics**. The choices are described in Table 9.4.

Table 9.4 History statistics sampling intervals

Sampling Interval	Description
1 sec. / 30 min.	Takes the average over every one second to produce a graph that covers a total of 30 minutes.
5 sec. / 2 hr.	Takes the average over every five seconds to produce a graph that covers a total of two hours.
15 sec. / 6 hr.	Takes the average over every 15 seconds to produce a graph that covers a total of six hours.
30 sec. / 12 hr.	Takes the average over every 30 seconds to produce a graph that covers a total of 12 hours.
60 sec. / 24 hr.	Takes the average over every one minute to produce a graph that covers a total of 24 hours.

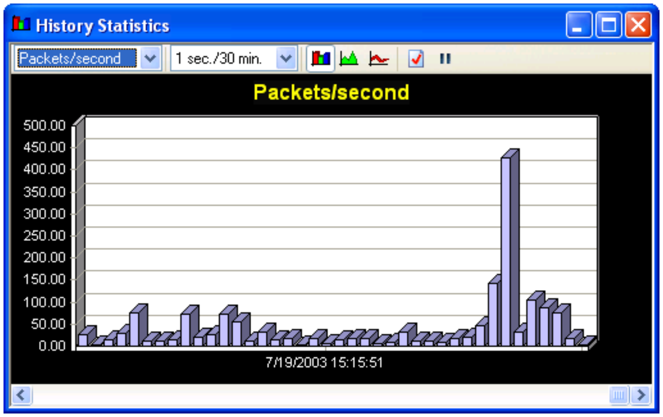
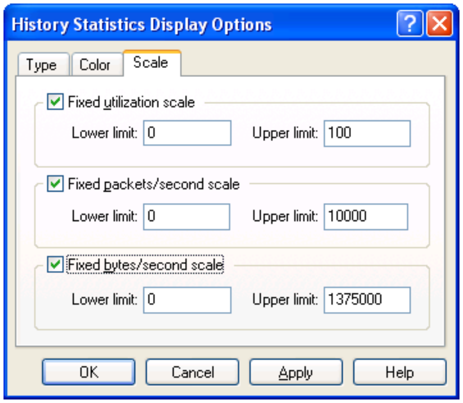


Figure 9.9 History Statistics

The first three buttons to the right of the interval drop-down list show: a bar graph, an area graph, and a line graph. You can quickly change the display format of the **History Statistics** to any one of these formats by clicking on its button.



History Statistics Display Options

☑ Fixed utilization scale

Lower limit: 0 Upper limit: 100

☑ Fixed packets/second scale

Lower limit: 0 Upper limit: 10000

☑ Fixed bytes/second scale

Lower limit: 0 Upper limit: 1375000

Figure 9.10 Scale view of the History Statistics Display Options dialog

The last two buttons at the far right of the **History Statistics** window are the **Options** button and the **Pause** button. The **Pause** button, at the far right, temporarily stops the otherwise continuous scrolling of the display. Click the **Pause** button when you want to go back and review an earlier time segment and temporarily suspend screen updates and the scrolling they entail. Calculations will go on uninterrupted in the background. Scrolling will resume when you unclick the **Pause** button or when you close and re-open the **History Statistics** window.

The **Options** button is the second button in from the right of the row of buttons, immediately to the left of the **Pause** button. This button opens the **History Statistics Display Options** dialog, where you can set the appearance of the History Statistics graph. The **History Statistics Display Options** dialog has three views, ***Type***, ***Color***, and ***Scale***, accessible by clicking their respective tabs. The first two views, ***Type*** and ***Color***, are common to other statistics display options and are described elsewhere (please see “Controlling the graph display” on page 184).

The ***Scale*** view (Figure 9.10) allows you to use a fixed scale (checked) or a dynamically adjusted scale based on the largest values seen so far (unchecked), for each of three parameters: *Utilization*, *Packets/second*, and/or *Bytes/second*. Use the text entry boxes to set a *Lower limit* and an *Upper limit* for any enabled fixed scale. Click **Apply** to see the effect of your changes. Click **OK** to accept the changes, or click **Cancel** to close the dialog without making any changes.

Statistics in capture windows

While Monitor statistics offer a continuous view of all network traffic on the selected Monitor adapter, Capture windows can be used to collect statistics on a more narrowly defined aspect of network traffic. Capture windows allow you to filter traffic before statistics are calculated, and they allow you to select groups of packets and save them for later analysis. Unlike Monitor statistics, Capture windows allow you to save and analyze individual packets. This can be crucial in understanding what certain nodes are attempting to do on the network, for example.

Please see Chapter 15, “Post-capture Analysis” on page 283, for a more detailed view of the analytical tools available for looking at traffic that has been captured and saved. This section just gives the basic distinctions between the statistics available in Monitor statistics and those available in Capture windows and Packet File windows.

Monitor vs. capture or packet file window statistics

The primary difference between Monitor statistics and those calculated in Capture windows or Packet File windows is that statistics in these windows are based on a subset of network traffic. If the capture options for a window are set to *Continuous capture* and the buffer has wrapped (that is, been emptied and begun to re-fill, or begun to overwrite older entries), statistics are still based on all packets seen since capture began, even though much of the traffic may no longer be visible in the ***Packets*** view. If packets are hidden using any of the Hide functions from the **Edit** menu, however, statistics are re-calculated based on the remaining, visible packets. Unhiding the packets will cause the statistics to once again be re-calculated.

Hide and Unhide have no effect on Monitor statistics.

Statistics in the views of a Capture window or a Packet File window are calculated based on the packets that are visible and in the buffer at the time the statistics are calculated. Filters can control what packets are placed in the buffer of a Capture window, and packet slicing can affect the contents of packets in either type of buffer. Please see “Using packet slicing” on page 57 for more information about packet slicing.

While you can create a new alarm from within any Capture window or Packet File window, the alarm itself will always watch Monitor statistics only. If **Monitor Statistics** is turned off, a message appears in the **Alarms** window warning you that alarms cannot function properly without Monitor statistics.

For creating reports of statistics from Capture windows or Packet File windows, please see “Saving reports from capture windows” on page 172. You can also periodically output statistics from any open Capture window using the ***Statistics Output*** view of the **Capture Options** dialog. Please see “Statistics output views” on page 173 for details.

Nodes

The ***Nodes*** view in a Capture window or in a Packet File window presents essentially the same view and provides the same customization features, detailed views, and calculations for the subset of traffic in its window that the Monitor statistics’ **Node Statistics** window does for all traffic seen on the Monitor statistics adapter. Please see “Node statistics” on page 149 for details.

Protocols

The ***Protocols*** view in a Capture window or in a Packet File window presents essentially the same view and provides the same customization features, detailed views, and calculations for the subset of traffic in its window that the Monitor statistics’ **Protocol Statistics** window does for all traffic seen on the Monitor statistics adapter. Please see “Protocol statistics” on page 154 for details.

Network statistics equivalents

There is no Network Statistics view for Capture windows or Packet File windows. The instantaneous measure of network performance makes no sense in a Packet File window, as no packets are being received. For Capture windows, however, the ***Graphs*** view lets you create one or more graphs that would show much the same information.

Error counts equivalents

There is no Error counter as such in Capture windows or Packet File windows. Error counts do appear in the ***Summary*** view, and advanced filters allow you to capture any combination of the supported error packet types (see “Error filter nodes” on page 219). In addition, the ***Graphs*** view lets you graph any combination of statistics from the ***Summary*** view in a variety of formats.

Alternatively, after capture you could use the **Select Related Packets**, or the **Select…** function from the **Edit** menu to select only the types of error packets you want from a more heterogeneous group of captured traffic.

Size and history equivalents

The ***Graphs*** view of a Capture window or Packet File window provides a number of default graphs which present the same information for the subset of traffic in their window that the Monitor statistics' **Size Statistics** and **History Statistics** windows do for all traffic seen on the Monitor statistics adapter.

The default *Size* graph is equivalent to the **Size Statistics** display in Monitor statistics. Please see "Size statistics" on page 160 for more details.

Most of the various functions of the **History Statistics** window in Monitor statistics are covered in two default graphs in the ***Graphs*** view: *Bytes/Second* and *Packets/Second*. Please see "History statistics" on page 163 for more details.

For more information about the ***Graphs*** view, please see "Graphs view of capture windows and packet file windows" on page 189.

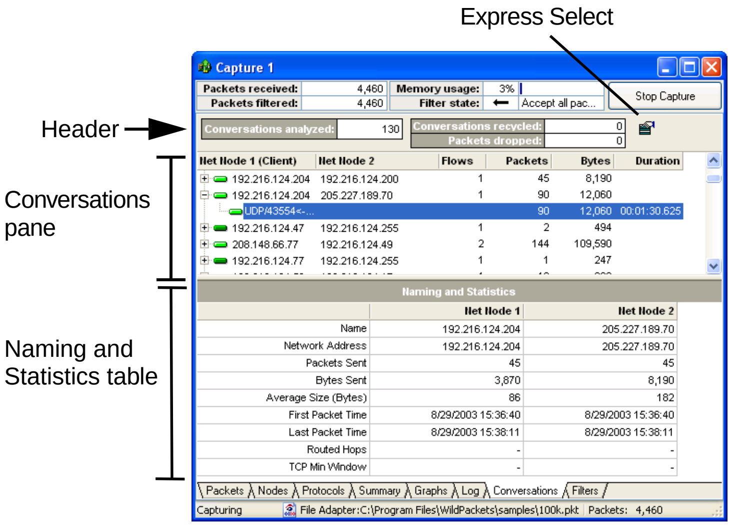
Summary

The ***Summary*** view in a Capture window or in a Packet File window presents essentially the same view and provides the same customization features, detailed views, and calculations for the subset of traffic in its window that the Monitor statistics' **Summary Statistics** window does for all traffic seen on the Monitor statistics adapter. Please see "Summary statistics" on page 161 for more details.

Conversations

The ***Conversations*** view in a Capture window or in a Packet File window has no equivalent in Monitor statistics, and is unique to EtherPeek standard.

The ***Conversations*** view (Figure 9.11) groups traffic in a Capture window or Packet File window into conversations between pairs of network nodes. The ***Conversations*** view presents information about each conversation in tabular form in the upper Conversations pane, and additional information about each peer in the selected conversation in the *Naming and Statistics* table in the lower pane.



Packets received:	4,460	Memory usage:	3%
Packets filtered:	4,460	Filter state:	⟵ Accept all pac...

Conversations analyzed:	130	Conversations recycled:	0
		Packets dropped:	0

Net Node 1 (Client)	Net Node 2	Flows	Packets	Bytes	Duration
⊞ 192.216.124.204	192.216.124.200	1	45	8,190	
⊟ 192.216.124.204	205.227.189.70	1	90	12,060	
UDP/43554<-...			90	12,060	00:01:30.625
⊞ 192.216.124.47	192.216.124.255	1	2	494	
⊞ 208.148.66.77	192.216.124.49	2	144	109,590	
⊞ 192.216.124.77	192.216.124.255	1	1	247	

Naming and Statistics

	Net Node 1	Net Node 2
Name	192.216.124.204	205.227.189.70
Network Address	192.216.124.204	205.227.189.70
Packets Sent	45	45
Bytes Sent	3,870	8,190
Average Size (Bytes)	86	182
First Packet Time	8/29/2003 15:36:40	8/29/2003 15:36:40
Last Packet Time	8/29/2003 15:38:11	8/29/2003 15:38:11
Routed Hops	-	-
TCP Min Window	-	-

Capturing File Adapter:C:\Program Files\WildPackets\samples\100k.pkt Packets: 4,460

Figure 9.11 Conversations view of a Packet File window in EtherPeek standard

The header section of the ***Conversations*** view shows the number of *Conversations*. When Conversations analysis runs in a Capture window, it uses a fixed block of memory allocated when the Capture window is created. The counts of *Conversations recycled* and *Packets dropped* relate to the use of this memory. A similar memory allocation system is used for the Expert fuinctions in EtherPeek NX. Please see “Expert memory allocation” on page 116 for details.

To the right of this information is the **Express Select** button. Click the **Express Select** button to use the currently selected conversation as the basis for a **Select Related Packets** selection in the ***Packets*** view.

The Conversations pane of the ***Conversations*** view shows the current conversations, with information about each conversation displayed in a user-definable set of columns. Right-click in the Conversations pane to open the context menu and choose **Visible columns…** to select the columns you wish to display. Use drag and drop to change column order. To use drag and drop, click on a column heading, then drag the ghost image of the column heading to a new location and release the mouse button. The columns available in the Conversations pane of the ***Conversations*** view are shown in Table 9.5. Columns present in the default Conversations pane layout show an **X** in the **Default** column of Table 9.5.

Table 9.5 Conversations view, conversations pane columns

Default	Column	Description
X	***Net Node 1 (Client)***	The client or first peer in the selected conversation.
X	***Net Node 2***	The server or second peer in the selected conversation.
X	***Flows***	For a pair of nodes, shows the number of flows or conversations detected and detailed in the Conversations pane.
	Protocol	The protocol under which the packets in this conversation were exchanged.
X	***Packets***	The number of packets in the selected exchange. Note that packet totals are rolled up when the view is collapsed, such that higher levels of aggregations show totals for all sub-elements.
X	***Bytes***	The total bytes represented by the packets which were a part of the selected conversation.
X	***Duration***	The elapsed time, from the first to the last packet of the selected exchange, represented in the form Hours:Minutes:Seconds:Milliseconds.

The Conversations pane of the ***Conversations*** view of a Capture window or Packet File window provides a hierarchical view of all conversations contained in the visible packets in the buffer of the window. Each highest-level item in the display represents a single node acting as the Client or first peer in a particular conversation. When a group of conversations differ only in port number, they are ranged below the Client node in order by port number.

Note: The terms “conversation” or “flow” are equivalent, and have a precise meaning in the ***Conversations*** view. For IP, the end-to-end IP address, and UDP or TCP ports form a unique conversation for a given application. For IPX, the end-to-end IPX address, socket number, and connection IDs form a unique conversation for a given application.

Items in the Conversations pane are color coded for easy scanning. When a conversation is still active, the color block beside that item is bright green. When the conversation is completed, the color block is dull green.

Click on the + (plus) or - (minus) signs at the left margin to expand or collapse individual elements of the display. Alternatively, you can right-click anywhere in the Conversations pane to open the context menu and choose either **Expand All** or **Collapse All**.

When one or more conversations are highlighted, you can use the context menu to **Select Related Packets** either **By Source and Destination**, which chooses packets with matching source and destination addresses, or **By Conversation**, choosing packets sent between two nodes in either direction, with the matching protocol and port.

The *Naming and Statistics* table shows additional details for the participants in the selected conversation, identified as ***Net Node 1*** and ***Net Node 2***. The *Naming and Statistics* table shows the characteristics described in Table 9.6 for both Net Node 1 and Net Node 2.

Table 9.6 Naming and Statistics table parameters

Parameter	Description
Name	The name (or address) of each node. The node is identified by its logical address or by the symbolic name for that address if one exists in the Name Table.
Network Address	The logical or physical address, as appropriate to the conversation.
Packets Sent	The total number of packets sent by this node as a part of this conversation.
Bytes Sent	The total number of bytes sent by this node as a part of this conversation.
Average Size (Bytes)	The average size of the packets sent by this node as a part of this conversation, in bytes.
First Packet Time	The date and time of capture (to the nearest second) of the first packet for this node in the current conversation.
Last Packet Time	The date and time of capture (to the nearest second) of the last packet for this node in the current conversation.

Table 9.6 Naming and Statistics table parameters (continued)

Parameter	Description
Routed Hops	The number of intervening router hops separating Net Node 1 and Net Node 2 in this conversation.
TCP Min Window	The minimum size of the TCP window during the course of this conversation.

Output from statistics

You can save statistics to text files, print them out, or save them automatically to HTML or XML files using customized templates. Many graphical displays such as **Size Statistics** and the contents of the ***Graphs*** view of Capture windows can also be saved as images. This section describes the most important methods of saving statistics from Monitor statistics and from statistics views in Capture windows and Packet File windows.

Saving statistics

When a statistics window or view other than **Network Statistics** is the active or front-most window, the **File** menu changes to allow you to save the active window, showing, for example, **Save Node Statistics…** or **Save Size Statistics…** as a choice under the **File** menu. You can choose to save the file as either a tab-delimited (*.txt) or a comma-delimited (*.csv) text file which can be read by most database, spreadsheet and charting programs. Statistics presented in graphical form, such as **Size** and **History** statistics and any separately created **Graph** windows, can also be saved as an image of the current display in either a bitmapped image (*.bmp) or Portable Network Graphic (*.png) format.

Saving reports from capture windows

When a Capture window or Packet File window is the active or front-most window, you can choose **Save Report…** from the **File** menu to create an integrated collection of documents in XML, HTML, or a variety of text formats, reporting statistics from that window. Note that statistics calculations in Capture windows and Packet File windows follow slightly different rules than those in Monitor statistics. Please see “Monitor vs. capture or packet file window statistics” on page 166 for details.

The statistics reports include the data from the ***Nodes***, ***Protocols***, and ***Summary***, views, and the statistics associated with any graphs in the ***Graphs*** view. See the sections at the end of this chapter for more details on the structure of each of these report output formats.

Use the drop-down list to choose a *Report type* and choose a *Report folder* in which to save the report. Click **Save** to create the specified report. The resulting XML or HTML reports are viewable in Internet Explorer 5.5 or compatible browsers.

Printing statistics

To print a statistics window or view, make it the front-most or active window and choose **Print…** from the **File** menu. You can access all standard printer functions from the **Print Setup…** command under the **File** menu. You can print any statistics window or details window except the **Network Statistics** window.

Statistics output views

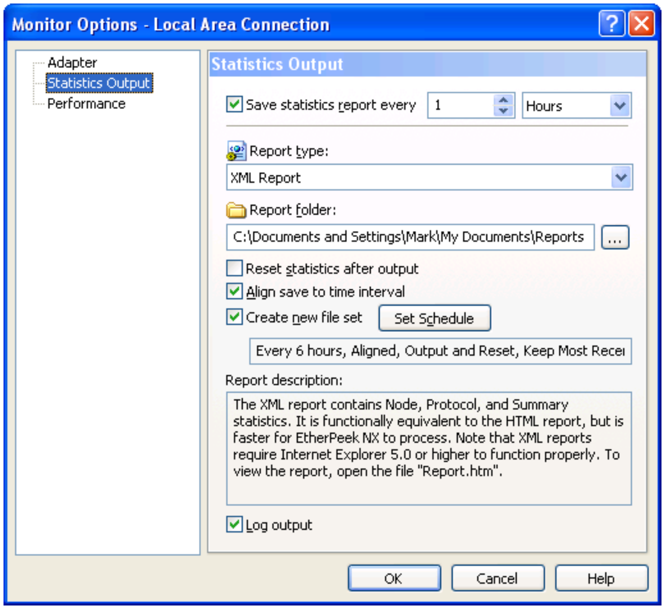
Statistics from open Capture windows or open Monitor statistics windows can be periodically saved as XML, HTML or in a variety of text formats.

To periodically save a particular set of statistics to text, HTML, or XML files:

1. Open the source for the statistics: either the Monitor statistics windows or the Capture window whose statistics you want to save.
2. If your source is Monitor statistics, choose **Monitor Options…** under the **Monitor** menu and choose the *Statistics Output* item in the navigation pane to open the ***Statistics Output*** view of the **Monitor Options** dialog (Figure 9.12). If your source is a Capture window, open its **Capture Options** window and choose the *Statistics Output* item in the navigation pane to open the ***Statistics Output*** view. These views have the same name and offer identical choices. Only the source of statistics is different.
3. To enable saving statistics, check the checkbox in the upper left, labeled *Save statistics report every … …*.
4. Set the frequency with which you want to update the statistics files, setting the interval in the first text entry box (whole numbers only) and the units of time in the box at the far right. Your choices are *Seconds*, *Minutes*, *Hours*, or *Days*. A new statistics report of the type specified below will be written out at the interval you specify here. With the exceptions noted below, each new report is written over the previous report, replacing it.

Tip The minimum interval is *5 Seconds*. Such a very short interval may be impractical, except in the lightest traffic.

5. Choose the report *Type* from the drop-down list. The *HTML Report*, *XML Report*, *Text Report (tab-delimited)*, *CSV Report (comma-delimited)* and *CSV Row Report (comma-delimited)* include all data from the **Node**, **Protocol**, and **Summary Statistics** windows or views. In addition, the tab-delimited (*.txt) and *.csv reports also include data from the **Size** and **History** statistics windows of Monitor statistics, or the statistics used by the graphs of the ***Graphs*** view of a Capture window. The *CSV Row Report* outputs the current values from the **Node**, **Protocol**, and **Summary Statistics**, windows or views and **Size Statistics** in a single row, appending to the same files each time statistics are written. See the sections at the end of this chapter for more details on the structure of each of these report output formats.



☑ Save statistics report every 1 Hours

Report type: XML Report

Report folder: C:\Documents and Settings\Mark\My Documents\Reports

☐ Reset statistics after output

☑ Align save to time interval

☑ Create new file set

Every 6 hours, Aligned, Output and Reset, Keep Most Recei

Report description:

The XML report contains Node, Protocol, and Summary statistics. It is functionally equivalent to the HTML report, but is faster for EtherPeek NX to process. Note that XML reports require Internet Explorer 5.0 or higher to function properly. To view the report, open the file "Report.htm".

☑ Log output

Figure 9.12 Statistics Output view of the Monitor Options dialog

6. Choose a *Report folder* location for the statistics output. Click the **...** (ellipsis) button to open a **Browse For Folder** dialog in which you can navigate to the location of the report folder.

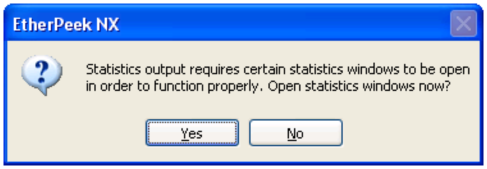
7. You can *Reset statistics after output* by checking the checkbox beside this item. Resetting statistics returns the counts to zero in the source of statistics (Monitor statistics or Capture window) and begins a fresh count. This is useful for creating a series of snapshots of network conditions.

8. You can *Align save to time interval* by checking the checkbox beside this item. When you check this option new output occurs at the nearest whole unit of time by the clock. For example, if your interval is set to some number of *Hours*, the output will occur on the hour. When this option is not checked, the count begins as soon as you click **OK** to accept the settings in the dialog, and output occurs when the first interval is reached.

9. With the exception of the *CSV Row Report*, which appends new entries to a single file, each new statistics report is written over any previous report at the save file location. To allow you to create a series of statistics output reports, EtherPeek can create new folders and write the statistics reports to these. Check the checkbox beside *Create new file set*. When this item is checked, reports are written to new file folders, created at an interval you specify in the **New File Set Schedule** dialog, available by clicking the **Set Schedule** button. For more details about this option, please see “New file set schedule” on page 176.

10. If you want a message placed in the global log file each time statistics are output, check the *Log output* checkbox at the bottom left of the dialog. Log entries include the path name of the output folder.

Note: Although you can enable the periodic output of Monitor statistics at any time, the output will only contain data if **Monitor Statistics** is enabled under the **Monitor** menu. The required statistics windows must also be open, although they may be reduced to icons. Similarly, periodic output from a Capture window can take place only when the window is open and capturing.



EtherPeek NX

Statistics output requires certain statistics windows to be open in order to function properly. Open statistics windows now?

Figure 9.13 Statistics output requires source windows to be open

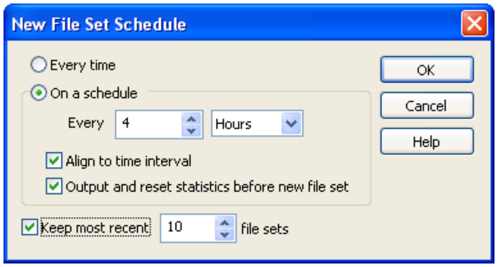
11. When you have set the parameters for statistics output, click the **OK** button to accept your changes, or click **Cancel** to close the dialog without making any changes.

New file set schedule

Each time statistics are output, the new report is written over any previous report that exists at the same file save location. (The one exception to this rule is the *CSV Row Report*, which appends new entries to a single file.) If you wish to create a series of statistics output reports, check the checkbox beside *Create new file set*. When this item is checked, EtherPeek creates a series of new file folders, one at a time, at the intervals you specify. As each new folder is created, statistics reports begin to be written to the new folder, leaving the last report written to each previously created folder intact.

Folder names have the form: `FolderName-YYYY-MM-DD hh.mm.ss`, where `FolderName` is the name you specified in *Report folder*, and the timestamp shows the year, month, day, hour, minute, and second at which the folder was created. (More precisely, the timestamp shows the time at the beginning of the output interval of the first statistics report which is to be written to the folder.) The timestamp is always local time for the machine on which EtherPeek is running.

When you check *Create new file set*, the **Set Schedule** button becomes available. Click this button to open the **New File Set Schedule** dialog (Figure 9.14), in which you can control the frequency at which new report folders are created and other parameters.


New File Set Schedule

○ Every time

● On a schedule

Every 4 Hours

☑ Align to time interval

☑ Output and reset statistics before new file set

☑ Keep most recent 10 file sets

Figure 9.14 New File Set Schedule dialog for statistics output

You have a choice of two schedule approaches: *Every time* or *On a schedule*. Click the *Every time* radio button to create a new folder each time a new statistics report is generated. If you choose this option, the timestamp of each folder will show the time at which each statistics report was created.

Alternatively, you can click the *On a schedule* radio button to establish a schedule just for the creation of new folders, and hence new file sets. In the line *Every* … …, use the data entry box to enter the number, and the drop-down list to set the units of time (*Days*, *Hours*, *Minutes*, or *Seconds*). When you choose this option, the timestamp on each file folder will show the time at which the folder itself was created. Statistics reports continue to overwrite one another in this folder until a new folder is created.

For example, if you set the ***Statistics Output*** view to *Save statistics report every 1 Hours* and set the **New File Set Schedule** dialog to create a new file set *On a schedule Every 6 Hours*, after 13 hours, there would be three folders, with one statistics report in each. The file creation time on the statistics reports in the first two folders would be 6 hours later than the timestamp in the folder name. The creation date on the statistics report in the most recently created folder would be only one hour later than the timestamp in the folder name, as only a single report output would have occurred since the creation of the folder.

Check the *Align to time interval* checkbox to have the creation of new folders occur on the nearest whole unit of clock time (for example, on the hour).

Check the checkbox beside *Output and reset Statistics before new file set* to output the next scheduled statistics report, then reset statistics before each new folder is created.

Whether you have set a separate schedule for the creation of new file sets or are using the interval already set in the main ***Statistics Output*** view, you can check the checkbox beside *Keep most recent … file sets* and enter a number in the data entry box.When this item is enabled, EtherPeek will keep only the specified number of files, discarding older files and folders to make room for newer ones.

Click **OK** to accept your changes to the **New File Set Schedule** dialog and return to the ***Statistics Output*** view, or click **Cancel** to close the dialog and return without making any changes. The current settings for the **New File Set Schedule** dialog appear in the ***Statistics Output*** view in the box immediately below the **Set Schedule** button.

XML output

Choose *XML Report* from the *Report type* drop-down list to output statistics as XML. The *XML Report* includes **Node**, **Protocol** and **Summary Statistics** information. When this report type is generated for a Capture window or Packet File window, it also includes statistics for all graphs in the ***Graphs*** view. The report is written to a file called StatsReport.xml in the directory you specified in *Report folder.* Supporting files are also written to this directory, including an HTML presentation of the data called Report.htm, the XSL style sheets used to present the report, and a copy of the XML Schema. You can view the formatted output in Report.htm in Internet Explorer, version 5.5 and above.

XML Report provides the same detail as the HTML output formats (except **History Statistics**), but with less processing demand on the program. In addition, XML provides a structured output for data interchange. For more detail about the structure of XML output, please see the Readme file located in the 1033\Reports directory where you installed EtherPeek and StatsReportSchema.xml, located in the 1033\Reports\Auxiliary subdirectory.

HTML output

The *HTML Report* includes **Node**, **Protocol**, **Summary** and **History Statistics** information. When this report type is generated for a Capture window or Packet File window, there are no **History Statistics** as such, but the report does include statistics for all graphs in the ***Graphs*** view.

EtherPeek outputs statistics to HTML files, one file for each statistics window or view. All of the output files are linked through a page called Stats.htm, and all are written to the directory you specified in *Report folder.* EtherPeek creates HTML files using templates. The HTML template contains keywords for the various parameters of each statistical display. These keywords are then replaced by the values returned by the statistics function each time it saves. The result is written to a standard HTML file and placed in the *Report folder* or directory of your choosing. You can use the templates supplied or create your own templates. Detailed instructions are included in the Readme file located in the 1033\Reports directory within the directory where you installed EtherPeek.

Text output

Choose *Text Report (tab-delimited)* from the *Type* drop-down list to output statistics as tab-delimited text (*.txt) or choose *CSV Report (comma-delimited)* to output text with comma separated values (*.csv). Either function creates files of the specified type, one for each statistics window or view, and places them in the directory you specified in *Report folder.* These text formats output **Node**, **Protocol**, **Summary**, **Size**, and **History** statistics for Monitor statistics. For output from Capture windows, periodic output in these formats includes statistics from the ***Nodes***, ***Protocols***, and ***Summary*** views (one file per view) and from the ***Graphs*** view (one file per graph).

Row report

Choose *CSV Row Report (comma delimited)* to periodically append the current values from the **Node**, **Protocol**, **Summary**, and **Size Statistics** windows or views to a single set of files, one for each statistics window or view. Unlike the other reports, the Row Report does not overwrite the target files when statistics are output. Instead it adds a new row to the end of each target *.csv file each time statistics are output. Each such row contains the whole contents of the ***Current*** column of the **Summary Statistics** window (or view), or the current values for the other statistics, as comma-separated values. You can import CSV format files to spreadsheet and database programs for trending and other analysis.

Graphs of Monitor and Capture Statistics

10

In addition to the standard statistical displays, EtherPeek offers multiple methods for displaying individual statistical items or groups of statistics in user-defined graphs.

From creating instant graphs to defining complex suites of graphical displays, EtherPeek offers speed, power, and flexibility in the display of statistics. This chapter explains the tools for graphing statistics from Monitor statistics and from Capture windows and Packet File windows.

In this Chapter:

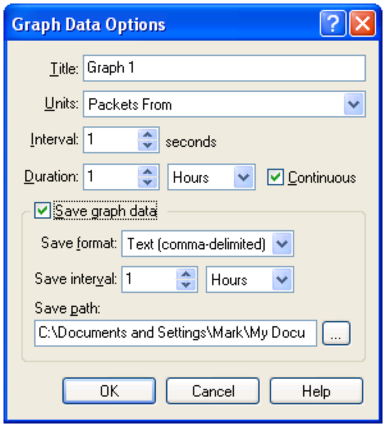
Creating and controlling graph windows

Individual items from the **Node**, **Protocol** and **Summary Statistics** windows (or from the analogous views of a Capture window) can be displayed graphically in real time. The data from these graphs can also be saved as tab-delimited or comma-delimited text, or as XML. This section describes how to create and modify the appearance of statistics graphs. The **Size** and **History Statistics** windows are displayed graphically by default. With the noted exceptions, what is said below about graph display options also applies to these windows.

Creating a new graph window

You can create a graphic view, updated in real time, of any item in the **Node**, **Protocol** or **Summary Statistics** windows (or from the analogous views of a Capture window) by selecting the item and clicking the **Graph** button at the top of the display.

Note: The **Graph Data Options** dialog will appear with added options in the lower half of the display when you create a new **Graph** window from items in a Capture window or Packet File window. These additional options relate to adding statistics items to the ***Graphs*** view of Capture windows or Packet File windows and are covered in their own section. Please see “Graphing statistics from capture and packet file windows” on page 187 for details.



Title: Graph 1

Units: Packets From

Interval: 1 seconds

Duration: 1 Hours ☑ Continuous

☑ Save graph data

Save format: Text (comma-delimited)

Save interval: 1 Hours

Save path: C:\Documents and Settings\Mark\My Docu

Figure 10.1 Graph Data Options dialog for Monitor statistics

To create a real-time graph of an item in a statistics display:

1. Open an appropriate statistics window or statistics view of a Capture window. You can create graphs of items in the **Node**, **Protocol** or **Summary Statistics** windows or from items in the analogous views of a Capture window.
2. Select the item you wish to graph and click the **Graph** button at the top of the statistics window (or view), or right-click and choose **Graph…** from the context menu.
3. This opens the **Graph Data Options** dialog, shown in Figure 10.1.

Note: The version of the **Graph Data Options** dialog presented when graphing a statistic from a Capture window offers additional options, but can be used as described here.

4. Fill in the **Graph Data Options** dialog. The table below (Table 10.1) describes each of the parameters used to set up a statistics graph and to save data from it.
5. When you have chosen the parameters, click **OK** to create the new graph and begin displaying data, or click **Cancel** to close the dialog and return to the statistics display.

Table 10.1 Graph Data Options dialog parameters

Parameter	Usage
Title	The title of the graph.
Units	The units to be graphed. The dialog is aware of the statistic to be graphed, and will only present those units which make sense in context.
Interval	Enter a number to set the refresh and sampling interval, in *seconds*.
Duration	The total length of time to be covered by the graph. Choose units of *Minutes*, *Hours,* or *Days* from the drop-down list.
Continuous	If this checkbox is checked, the graph will represent a moving window of the size specified in *Duration* above. If the checkbox is unchecked, graphing will stop when the *Duration* time is reached.

Table 10.1 Graph Data Options dialog parameters (Continued)

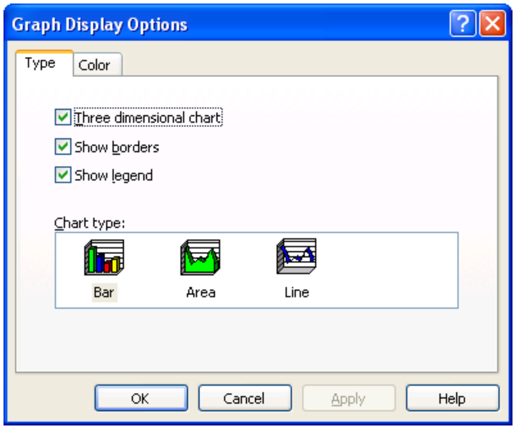
Parameter	Usage
Save graph data	Check this checkbox to enable the remainder of this dialog, specifying the format, interval and path with which to save graph data. Uncheck this checkbox to disable saving graph data.
Save format	Choose from the drop-down list one of the three supported formats: comma-delimited text (*.csv), tab-delimited text (*.txt), or XML (*.xml)
Save interval	Specify the frequency with which graph data is written to the specified file. Enter a number and use the drop-down list to choose units of *Minutes*, *Hours* or *Days*.
Save path	Choose the directory in which the graph data files should be saved. To browse, choose the button marked with the **…** ellipses.

Controlling the graph display

Statistics graphs, including the **History Statistics** graph, scroll each time data is refreshed so the most recent data appears at the far right of the screen. To temporarily suspend scrolling and make it possible to view data which has scrolled off-screen to the left, click the **Pause** button, located at the top of the **Graph** window.

Note: The scroll bar represents the position within a window of the size you set in the *Duration* parameter. For example, if you set a duration of one hour and have been graphing statistics for only ten minutes, only the right-most portion of the scroll bar will show any graphed data.

You can quickly change from one display type to another by clicking the icons representing **Bar**, **Area**, and **Line** display types, located at the top of the **Graph** window. For a finer control of the appearance of the graph, click the **Options** button at the top of the **Graph** window to open the **Graph Display Options** dialog, described below.



Graph Display Options

☑ Three dimensional chart
☑ Show borders
☑ Show legend

Chart type:

Figure 10.2 Graph Display Options dialog, Type view

Note: The ***Type*** and ***Color*** views of the **History Statistics Display Options** and **Size Statistics Display Options** dialogs are nearly identical to the **Graph Display Options** dialog, differing primarily in the *Chart type* choices available in each.

The **Graph Display Options** dialog presented for free-standing **Graph** windows has two tabs. From left to right, they are:

Type In this view (Figure 10.2), you can set the display format to *Bar* graph, *Area* graph or *Line* graph by choosing the appropriate labeled icon. The choice of graph types is context sensitive, and only those choices applicable to the graph being modified are available. You can also turn borders on or off by checking or unchecking the *Show borders* checkbox. Borders are on by default. Change the graph display from two-dimensional to three-dimensional by checking the *Three dimensional chart* checkbox. Toggle the display of the key or legend by checking or unchecking the *Show legend* checkbox.

Color In this view (Figure 10.3), you can click in the color swatches to change the color of any of the listed display elements. Clicking in the swatch opens a small palette as a new window. Choose from this palette or click the **Other…** button at the bottom of the new window to open the **Color** dialog where you can create custom colors.

Click **Apply** to see the effect of your changes on the graph, click **OK** to accept changes, or click **Cancel** to return to the graph without making any changes.

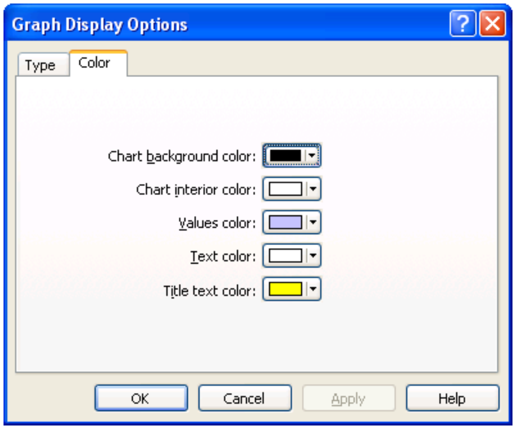


Figure 10.3 Graph Display Options dialog, Color view

Saving graph windows

When a **Graph** window is the active or frontmost window, you can choose **Save Graph…** from the **File** menu to open a standard **Save As** dialog, from which you can save either the graph data or the current image of the **Graph** window itself. To save the graph data give the file a name, and choose a *Save file format* of *Text (tab delimited)(*.txt)*, *CSV (comma delimited)(*.csv)*, or *XML (*.xml)* from the drop-down list. To save the current image of the **Graph** window itself, give the file a name and choose either *Bitmap image (*.bmp)* or *PNG image (*.png)* from the *Save file format* drop-down list.

These options are separate from any settings you may have made in the **Graph Data Options** dialog to periodically *Save graph data*.

Monitor statistics graphs and alarms

Click the **Alarm** icon in the header of any statistics **Graph** based on Monitor statistics to create an alarm based on that statistic. Clicking on the **Alarm** icon opens the **Make Alarm** dialog where you can specify all characteristics of the alarm. Please see “Alarms” on page 231 for details.

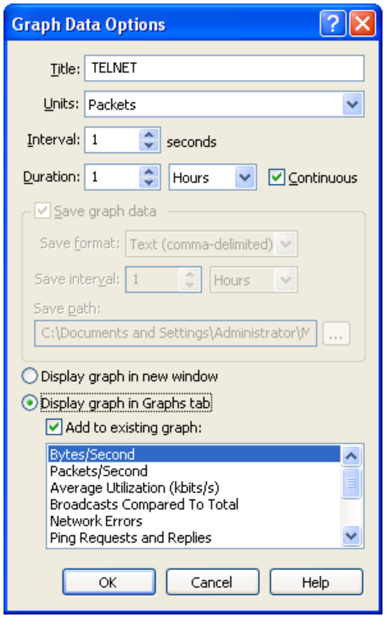
Important! Alarms only watch Monitor statistics. They never watch the statistics from a Capture window or Packet File window.

Graphing statistics from capture and packet file windows

You can graph any statistics item calculated in a particular Capture window or Packet File window, by creating the graph from within the window in either of two basic ways:

- Create a new statistics **Graph** window showing just the selected statistic
- Create or add to a graph displayed in the ***Graphs*** view

In either case, the graph displays the statistics calculated in the Capture window or Packet File window from which it was created. The main distinction between the two types of graphs is in their formatting options and the ability to save and retrieve these formats.


Graph Data Options

Title: TELNET

Units: Packets

Interval: 1 seconds

Duration: 1 Hours ☑ Continuous

☑ Save graph data

Save format: Text (comma-delimited)

Save interval: 1 Hours

Save path:
C:\Documents and Settings\Administrator\M

○ Display graph in new window

● Display graph in Graphs tab

☑ Add to existing graph:

Bytes/Second
Packets/Second
Average Utilization (kbits/s)
Broadcasts Compared To Total
Network Errors
Ping Requests and Replies

Options unique to
Capture window
or Packet File window
statistics graphs

Figure 10.4 Graph Data Options dialog for Capture window or Packet File window statistics

To create a graph of a statistics item in the ***Nodes***, ***Protocols***, or ***Summary*** views of a Capture window or Packet File window, highlight the item and click the **Graph** button at the top of the view, or right-click and choose **Graph…** from the context menu. This opens the **Graph Data Options** dialog (Figure 10.4). Note that the top half of this dialog is identical to the dialog of the same name used to control graphs made from Monitor statistics. The bottom half presents options for adding the statistic to the ***Graphs*** view.

These options are unique to graphs created from Capture windows or Packet File windows.

Use the radio buttons to choose whether to *Display graph in new window* or *Display graph in Graphs tab.*

Display graph in new window

If you choose to *Display graph in new window*, your options are identical to those available in creating a similar new **Graph** window for a Monitor statistics item. Only the source of statistics is different. A new window is created, showing a single statistic. When you close the source of statistics (in this case, the Capture window or Packet File window), the **Graph** window disappears. A new **Graph** window created from a Capture window or Packet File window offers the same formatting and data saving options as a **Graph** window created for a Monitor statistics item. Please see “Creating and controlling graph windows” on page 182 for details.

Important! Alarms only watch Monitor statistics. They never watch the statistics from a Capture window or Packet File window. You cannot create an alarm based on a graph from a Capture window or Packet File window.

Adding a statistic to the graphs view

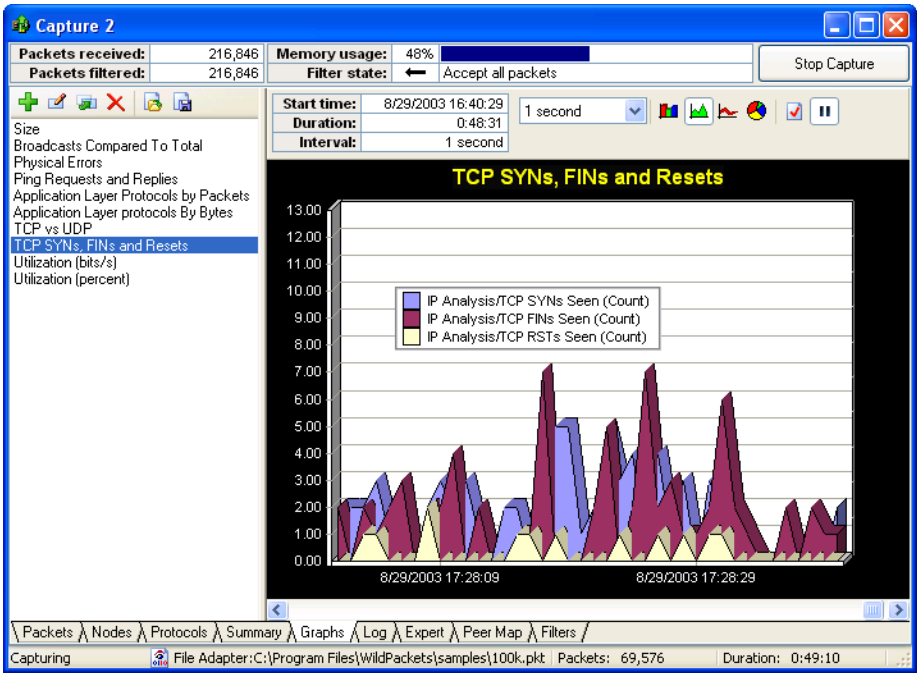
If you chose to *Display graph in Graphs tab*, you have two options. If you make no further selection, the new graph will be created and added to those listed in the ***Graphs*** view. Its name will be added to the list of graphs already displayed there. To see the graph, select its name from the list at the left side of the ***Graphs*** view. The graph will be displayed on the right.

Alternatively, you can add the selected statistics item to one of the graphs that already exists in the ***Graphs*** view. Click the *Display graph in Graphs tab* radio button. Check the checkbox labeled *Add to existing graph*, and choose the target graph by highlighting its title in the list shown below. When you click the **OK** button, the statistics item you selected will be displayed under its default parameter name as a new item in the graph you selected. To view this item, select the title of the graph to which you added the statistic, using the list at the left of the ***Graphs*** view. The graph with the new statistics item will appear at the right. You can add up to 20 statistics items to a single graph in the ***Graphs*** view, although for ease of reading you may want to keep to a smaller number.

When you choose to *Display graph in Graphs tab*, the *Save graph data* section of the **Graph Data Options** dialog becomes grayed out. This is because, unlike separate **Graph** windows, the graphs in the ***Graphs*** view are treated as a part of the Capture window or Packet File window, and their data is saved using the same methods as other items in those windows. Briefly, you can use the **File** > **Save Report…** menu option for either Capture windows or Packet File windows. For Capture windows only, you can also use the ***Statistics Output*** view of the **Capture Options** dialog to set parameters for periodic output of statistics, including all statistics from graphs in the ***Graphs*** view. For details on these methods, please see “Output from statistics” on page 172.

Graphs view of capture windows and packet file windows

Click the *Graphs* tab to open the ***Graphs*** view (Figure 10.5) of any Capture window or Packet File window. The ***Graphs*** view contains a number of default graphs, including *Size*, *Utilization (percent)*, *Utilization (bits/s)*, and many more.



Capture 2

Packets received: 216,846
Packets filtered: 216,846
Memory usage: 48%
Filter state: Accept all packets

Start time: 8/29/2003 16:40:29
Duration: 0:48:31
Interval: 1 second

Capturing | File Adapter:C:\Program Files\WildPackets\samples\100k.pkt | Packets: 69,576 | Duration: 0:49:10

Stop Capture
Size
Broadcasts Compared To Total
Physical Errors
Ping Requests and Replies
Application Layer Protocols by Packets
Application Layer protocols By Bytes
TCP vs UDP
TCP SYNs, FINs and Resets
Utilization (bits/s)
Utilization (percent)
1 second
TCP SYNs, FINs and Resets
IP Analysis/TCP SYNs Seen (Count)
IP Analysis/TCP FINs Seen (Count)
IP Analysis/TCP RSTs Seen (Count)
13.00
12.00
11.00
10.00
9.00
8.00
7.00
6.00
5.00
4.00
3.00
2.00
1.00
0.00
8/29/2003 17:28:09
8/29/2003 17:28:29
Packets
Nodes
Protocols
Summary
Graphs
Log
Expert
Peer Map
Filters

Figure 10.5 Graphs view of a Capture window showing TCP SYNs, FINs. and Resets

The ***Graphs*** view allows great flexibility in the display of statistics. You can add to, delete, rearrange, create, edit, export, and import graphs of a wide range of formats, each based on single or multiple statistics from the current Capture window. This section explains how to manage graphs in the ***Graphs*** view.

The ***Graphs*** view is divided into a list pane on the left and a display pane on the right. The list pane presents the list of available graphs. The title of the currently visible graph is shown by a highlight in this list. The graph itself appears on the right. Select any title from the list to display that graph. Right-click on any title to open a context menu which mirrors the buttons at the top of the list pane. The buttons (or context menu items) and their functions are described in Table 10.2 below.

Table 10.2 Buttons in list pane of Graphs view

Button	Usage
Insert	Opens the **New Graph: Pick a Statistic** dialog, presenting: above, a scrollable hierarchical list of all statistics in the ***Summary*** view, and below, a drop-down list for choosing the *Units* of display for the highlighted statistics item. Choose any statistics item. If alternative units are possible for the selected item, you can choose them from the drop-down list. Click **OK** to add the new graph to the ***Graphs*** view.
Edit	Opens the **Graph Display Options** dialog for the selected graph.
Duplicate	Creates a copy of the selected graph and adds it to the list pane, with the word *Copy* added to its name.
Delete	Deletes the selected graph.
Import	When you click **Import**, the program first asks if you would like to *Delete all graphs before importing?* If you choose **Yes**, all the graphs currently shown in the ***Graphs*** view will be deleted and replaced by the contents of the imported *.gph file. If you choose **No**, the graphs you import will be added to the current list. Use the file **Open** dialog to navigate to the location of the *.gph file you wish to import, and click **OK**.
Export	You can export the entire contents of the ***Graphs*** view to a *.gph file, which is a set of parameters for defining all the graphs currently in the ***Graphs*** view. This allows you to create and maintain groups of graphs for particular troubleshooting tasks, or for particular environments.

Tip You can restore the default ***Graphs*** view by importing the Default Graph.gph file, located in the 1033\Graphs directory in the directory where you installed EtherPeek.

Controlling display of graphs in the graphs view

Graphs in the ***Graphs*** view have a standard basic layout.

A header section at the top of the display contains a drop-down list for setting the display interval, buttons for choosing a graph style, a **Pause** button to temporarily halt the scrolling of the display, and an **Options** button to open the **Graph Display Options** dialog for the graph. Some graphs may also show tabular data at the left of this header area, as appropriate to the statistics being displayed.

Tip You may need to increase the width of the Capture window or Packet File window in order to see all the items in the graph display pane header area.

Below this header area is the graph itself. You can choose whether the graph key or legend is displayed within the graph area or at the right side. Double-click in the legend to toggle its placement.

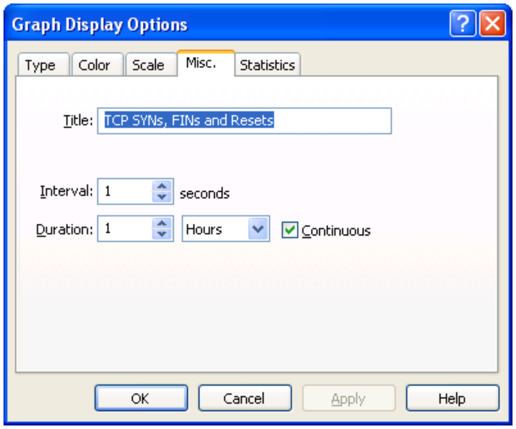
There are three basic sets of tools for controlling graph display. The first is the tools in the header section. The second is the **Graph Display Options** window, available by clicking the **Options** button in the graph display pane, or by clicking the **Edit** button in the list pane. The third is the **Chart FX Properties** dialog, available by double-clicking within any graph display.

The header options and the first two panes of the **Graph Display Options** dialog (the graph ***Type*** and ***Color*** views) are essentially identical to the analogous options for graphs created for Monitor statistics. Please see “Controlling the graph display” on page 184 for details. The remaining tools, the last three views of the **Graph Display Options** dialog and the **Chart FX** dialog, are unique to graphs created in the ***Graphs*** view of Capture windows and Packet File windows. These additional tools are described below.

Graph display options for the graphs view

The appearance of graphs is controlled by the **Graph Display Options** dialog. When graphs are displayed as a separate **Graph** window, this dialog only shows the first two tabs and views: ***Type*** and ***Color***. When graphs are displayed in the ***Graphs*** view, three more tabs or views are added to this dialog: ***Scale***, ***Misc.***, and ***Statistics***. These are described below.

The ***Scale*** view controls the scale used for the Y-axis (vertical scale) of the graph. Check the *Logarithmic* checkbox to plot the data against a logarithmic Y-axis. Check the *Fixed scale* checkbox and enter a *Minimum* and a *Maximum* value to force the Y-axis to this scale. If the *Fixed scale* checkbox is unchecked (the default), EtherPeek attempts to dynamically adjust the scale to match the data.



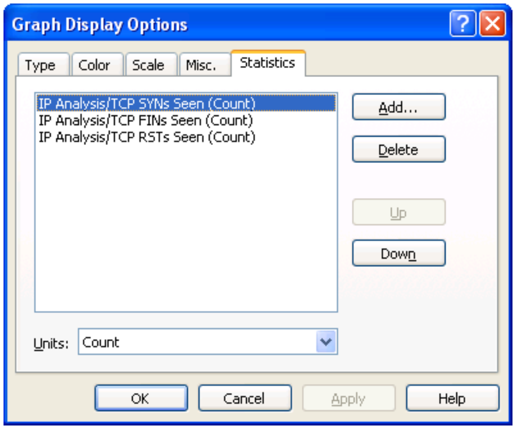
Title: TCP SYNs, FINs and Resets

Interval: 1 seconds

Duration: 1 Hours ☑ Continuous

Figure 10.6 Misc. view of the Graph Display Options dialog

Use the ***Misc.*** view to edit the *Title* of the graph, or set the sampling *Interval* by entering a number of *seconds*. You can set the *Duration* of the graph by entering a value in the text entry box and specifying the units (*Minutes*, *Hours*, or *Days*) by using the drop-down list. Check *Continuous* to restart collection when the *Duration* is reached, or leave *Continuous* unchecked to stop graphing when the *Duration* value is first reached. Note that the *Duration* sets the nominal width of the graph window.



IP Analysis/TCP SYNs Seen (Count)
IP Analysis/TCP FINs Seen (Count)
IP Analysis/TCP RSTs Seen (Count)

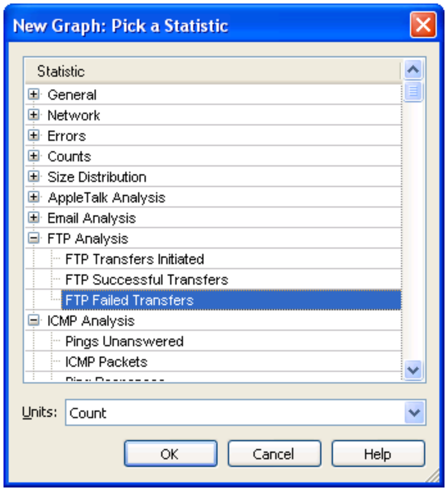
Units: Count

Figure 10.7 Statistics view of the Graph Display Options dialog

The ***Statistics*** view of the **Graph Display Options** dialog (Figure 10.7) presents a list of each statistics item displayed in the current graph. The drop-down list at the bottom of the display presents alternative choices for the *Units* used to measure the selected statistics item. If alternate units are available, you can choose them from this list.

Use the buttons at the right of the ***Statistics*** view to **Add…** a new statistics item to the list, to **Delete** an item, or to move the selected item **Up** or **Down** in the display. When you click the **Add…** button, it opens the **Add Statistic** dialog (Figure 10.8). This dialog presents a scrollable hierarchical list of all statistics in the ***Summary*** view, and below, a drop-down list for choosing the *Units* of display for the highlighted statistics item. Choose any item. If alternative units are possible, you can choose them from the drop-down list. Click **OK** to add the new statistics item to the list of those shown in the ***Statistics*** view.

Tip You can also add statistics items from the ***Nodes*** or ***Protocols*** views to any graph in the ***Graphs*** view. Please see “Adding a statistic to the graphs view” on page 188 for details.



Units: Count

Figure 10.8 Add Statistic dialog

Chart FX display options in the graphs view

Double-click on the graph display area of any graph in the ***Graphs*** view to open the **Chart FX Properties** dialog for that graph. The **Chart FX Properties** dialog offers a wide range of tools for fine tuning and customizing the appearance of graphs and charts. The ***General*** view of the **Chart FX Properties** dialog lets you set styles for axes, grid lines and general appearance qualities such as color schemes and fill patterns. The ***Series*** view offers control over color and style for individual statistics items within a graph. The ***Axes*** view offers a range of options for controlling the appearance of tick marks, value labels, and so forth. The ***3D*** view can set angle, shading, and perspective for three-dimensional graph views.

Filters

This chapter describes how to create, edit, and use filters in EtherPeek. Filters work by testing packets against the criteria specified in the filter. Packets whose contents or other attributes meet these criteria are said to “match” the filter.

When you use a filter to limit the flow of packets into a Capture window, or to select packets already captured, you can specify whether you want to see all the packets that match the filter, or only those packets which do not match. You can also use a filter match as the test condition for a trigger that will start or stop capture in a Capture window.

Filters are so easy to create in EtherPeek that you can often create a custom filter on-the-fly while analyzing suspect traffic on your network and use that filter to narrow your search in real time.

Filters are discrete individual tools that can be saved, imported, exported, edited, and used in combination with one another. You can build filters to test for just about anything found in a packet: addresses, protocols, sub-protocols, ports, error conditions, and more. This chapter explains how.

In this Chapter:

Using filters

Filters are used to isolate particular types of traffic on the network for troubleshooting, analysis, and diagnostics. Filters can be used singly or in groups. If multiple filters are used together, EtherPeek treats them as being OR'ed together. That is, a packet matching any *one* of the enabled filters is treated as a match.

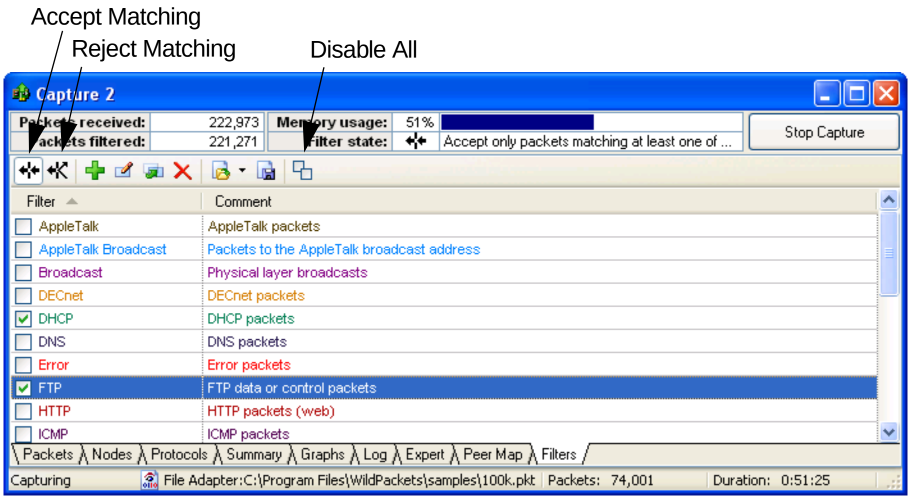
Note: Filters never apply to Monitor statistics, which are always calculated on the basis of *all* network traffic. Filters can only be used either to restrict the flow of packets into a Capture window or to select packets already captured to a buffer, either in a Capture window or from a saved packet file in a Packet File window.

Enabling filters in a capture window

To set one or more filters to control capture into a particular Capture window:

1. Click the *Filters* tab to open that window's ***Filters*** view (Figure 11.1).
2. Check to enable, or uncheck to disable any listed filter(s).
3. Use the buttons at the top of the view to choose how the Capture window should apply the filters. Choose either to **Accept Matching** or to **Reject Matching** packets.

   When you enable multiple filters, they are logically OR'ed together. If you choose **Accept Matching**, only those packets matching any one of the selected filters are captured into the buffer. If you choose **Reject Matching**, packets matching any one of the enabled filters will be rejected, and only those packets not matching any of the enabled filters will be captured into the buffer.



Packets received: 222,973
Packets filtered: 221,271
Memory usage: 51%
Filter state: Accept only packets matching at least one of ...

	Filter	Comment
□	AppleTalk	AppleTalk packets
□	AppleTalk Broadcast	Packets to the AppleTalk broadcast address
□	Broadcast	Physical layer broadcasts
□	DECnet	DECnet packets
☑	DHCP	DHCP packets
□	DNS	DNS packets
□	Error	Error packets
☑	FTP	FTP data or control packets
□	HTTP	HTTP packets (web)
□	ICMP	ICMP packets

Capturing | File Adapter:C:\Program Files\WildPackets\samples\100k.pkt | Packets: 74,001 | Duration: 0:51:25

Figure 11.1 Enabling filtering in a Capture window: the Filters view

You can also set filters in the similar ***Filters*** view of the **Capture Options** dialog. All available filters are shown in all filter lists. Changes made in the ***Filters*** view of the Capture window take effect immediately. If you use the **Capture Options** dialog to manually change the filter settings for a Capture window, the changes take place only when you click **OK** to accept the dialog's settings. The ***Filters*** view of the **Capture Options** dialog allows you to include filter settings in capture templates and AutoCapture files.

To view the details of any particular filter, double-click on the filter to open it in its appropriate **Edit Filter** dialog. This displays that filter's attributes, ready for editing. Click **Cancel** to close the filter without making any changes. For more details, see "Editing and duplicating filters" on page 202.

Tip You can create a filter testing for nearly any attribute of network traffic, including packet details, in a matter of two clicks using the **Make Filter** command. Please see "Make filter command" on page 200 for details.

Using filters as a trigger test

You can use one or more filters as the test for a trigger that will start or stop capture in a Capture window. When a packet is found that matches one of the filters set as a test parameter, the trigger trips, either starting or stopping capture. Any filter in the filter list can be used in this way, including one newly created by using the **Make Filter** command. Note that assigning a filter as a test parameter in the trigger does not enable that filter for use in controlling capture into the Capture window. To use a filter for that purpose, you must enable it separately, and expressly for packet capture, as described above.

For more about triggers, see “Triggers” on page 224.

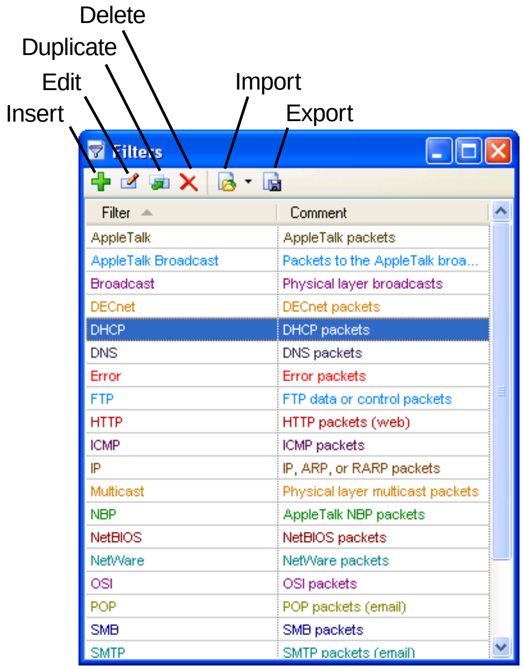
Using filters as a selection test

Filters you create or import can be used as selection criteria in the **Select** dialog, available by choosing **Select…** from the **Edit** menu. For more on using the **Select** dialog, see “Select dialog: filters, analysis modules and more” on page 292.

Filter resources in EtherPeek

EtherPeek includes a number of resources for filtering packets. The central resource is the **Filters** window. To open the **Filters** window, choose **Filters** from the **View** menu or press **Ctrl + M**.

The **Filters** window lists all currently loaded filters. From the **Filters** window, you can create a new filter by clicking the **Insert** button. When one of the existing filters in the window is selected, you can use the buttons to **Edit**, **Delete**, or **Duplicate** that filter. You can **Export** to save all existing filters to a *.flt file, or use the **Import** button to add the contents of any *.flt file to the existing filters. For a detailed discussion of each of these functions, please see “Creating and editing filters” on page 202.



Filter	Comment
AppleTalk	AppleTalk packets
AppleTalk Broadcast	Packets to the AppleTalk broa...
Broadcast	Physical layer broadcasts
DECnet	DECnet packets
DHCP	DHCP packets
DNS	DNS packets
Error	Error packets
FTP	FTP data or control packets
HTTP	HTTP packets (web)
ICMP	ICMP packets
IP	IP, ARP, or RARP packets
Multicast	Physical layer multicast packets
NBP	AppleTalk NBP packets
NetBIOS	NetBIOS packets
NetWare	NetWare packets
OSI	OSI packets
POP	POP packets (email)
SMB	SMB packets
SMTP	SMTP packets (email)

Figure 11.2 Filters window

Ready-made filters

EtherPeek ships with a number of filters already made and loaded, by default, into the **Filters** window. These may be used as they are, or they can provide a start for creating your own more precise filters. The **Filters** window in Figure 11.2 shows the list of ready-made filters. These ready-made filters are in the file Default.flt in the 1033\Filters directory in the directory where you installed EtherPeek, and can be loaded into the **Filters** window using the **Import** button. Please see “Saving and loading filters” on page 220.

Simple filter

The ***Simple*** view of the **Edit Filter** dialog (the default view) allows you to create filters based on address, protocol, and/or port. Double-click on an existing simple filter or click the **Insert** button in the **Filters** window to open the ***Simple*** view of the **Edit Filter** dialog. Please see “Simple filters” on page 203.

Advanced filter

Double-clicking on an existing advanced filter or choosing *Advanced* from the drop-down list in the upper right of the **Edit Filter** dialog opens the ***Advanced*** view of the **Edit Filter** dialog. Here you can create more complex filters with a wider range of filter parameters (including specific offsets and string values). In addition, the ***Advanced*** view allows you to construct a single filter based on a chain of filter properties connected by logical AND, logical OR, and logical NOT statements. Please see “Advanced filters” on page 208.

Make filter command

An easy way to create a new filter is to use the **Make Filter** command, available as the **Make Filter** button in many windows, or from the context menu (right-click) where applicable. The **Make Filter** command creates a filter based on the selected packet or statistics item. **Make Filter** can also be used in the Name Table to create a filter based on the selected named node, protocol, or port. It can also be used in the **Packet Decode** window (or the decode panes of the ***Packets*** view of a Capture window or Packet File window) to create a filter based on the selected data item.

When you use the **Make Filter** command, an unnamed filter is created matching the parameters of the selected packet, node, protocol, conversation, or packet decode item. An **Edit Filter** dialog will open with the parameters for your selection already loaded. Use this dialog to make any additional changes, and save the filter under a new name.

If multiple items are selected, the **Make Filter** command will attempt to create a filter for each one.

Using multiple filters simultaneously

When multiple filters are enabled simultaneously, EtherPeek considers them to be connected by logical OR statements. That is, packets matching any one of the enabled filters is considered a match, and will pass or be rejected, depending on whether you chose to accept or reject matching packets.

Filter parameters

Filters can operate on the properties of packets shown in Table 11.1 below. As the table shows, filters created in either view of the **Edit Filter** dialog can test for address, protocol and/or port. The additional parameters are available only for filters constructed in the ***Advanced*** view of the **Edit Filter** dialog.

Table 11.1 **Filter parameters**

Filter Parameter	Simple	Advanced	Description
Address	**yes**	**yes**	Tests the identity of the network node, either receiving or sending, for that packet. This can be a physical address, or a logical address under a particular protocol.
Protocol	**yes**	**yes**	EtherPeek can filter for protocols and for many of the individual types of traffic within a given protocol family, which we call sub-protocols. For example, FTP is a sub-protocol of TCP, which is itself a sub-protocol of IP.
Port	**yes**	**yes**	Tests for a port (or socket) within a particular protocol. IP, AppleTalk, and NetWare provide services at different ports or sockets on the server. The default port for Web traffic under TCP, for example, is port 80. ProtoSpecs assume that subprotocols are using the standard default ports (well known ports in TCP and UDP, for example), but you can also set filters to test explicitly for traffic to and/or from particular ports.
Value		**yes**	Tests the numerical value of a particular part of each packet (at a particular offset with a particular mask) for its relation (greater than, less than, equal to, and so forth) to the value you specify.
Pattern		**yes**	Tests for the presence of a particular character string (hexadecimal or ASCII) in each packet. Can be constrained to search within a specified location for greater efficiency.
Length		**yes**	Tests the length of the packet and matches those within the range you set, specified in bytes.

Table 11.1 Filter parameters (Continued)

Filter Parameter	Simple	Advanced	Description
Error		**yes**	Tests for one or more of four error conditions: CRC errors, Frame Alignment errors, Runt packets, and Oversize packets.
Analysis Module		**yes**	Packets handled by the specified Analysis Module will match the filter.

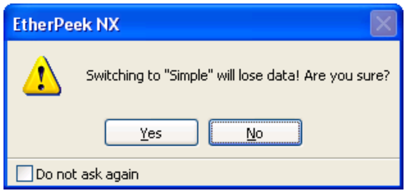
Creating and editing filters

This section describes the details of how to build filters, from the simple to the advanced. It also describes how to export, duplicate, import and edit filters.

Editing and duplicating filters

Editing an existing filter uses all the same tools as creating a new filter. Select the filter you wish to edit and click the **Edit** button, or double-click on any named filter to open the **Edit Filter** dialog. The dialog will open in the ***Simple*** view or the ***Advanced*** view automatically, depending on the filter you chose to edit. The filter's parameters will be displayed, ready to edit. You can make changes to the filter and click **OK** to save it.

To make a new filter based on an existing filter, you must first duplicate the existing filter, then edit the duplicate. To duplicate a filter, highlight the filter and click the **Duplicate** button. A copy will appear with the words "*Copy of*" prepended to the filter name. Edit the copy and save it under a new name.



Switching to "Simple" will lose data! Are you sure?

□ Do not ask again

Figure 11.3 Some filter types can only be created in the Advanced view

Note: You can switch back and forth between the ***Simple*** and ***Advanced*** views of the **Edit Filter** dialog while editing a filter. If, on moving from the ***Advanced*** to the ***Simple*** view, you are in danger of losing the ability to specify parameters you have already entered, a warning will be displayed and you will be given the opportunity to abort switching to the ***Simple*** view of the dialog.

Simple filters

Simple filters can test for address, protocol and port in a single filter. When multiple parameters are chosen they are connected by logical AND statements. That is, packets must match all of the conditions in order to match the filter.

To open the edit filter dialog simple filter view

To create a new simple filter, choose **Filters** from the **View** menu (or press **Ctrl + M**) to view the **Filters** window. Click the **Insert** button to bring up the **Edit Filter** dialog in its default ***Simple*** view (Figure 11.4).

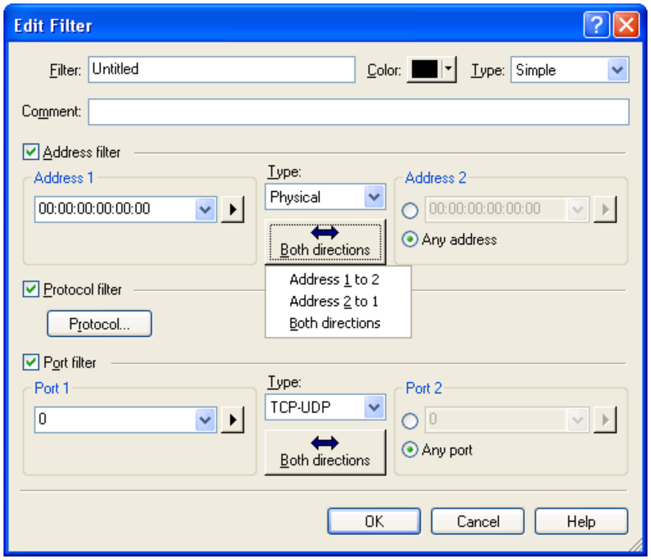
The default name "*Untitled Filter*" shows in the *Filter* text entry box where you can enter a new name. The color assigned to this filter (black is the default for a new filter) is shown in the color swatch at the top of the **Edit Filter** dialog, to the right of the *Filter* box. Click the arrow to the right of this color swatch to open the drop-down list of color choices.

In addition to its name, you can enter a *Comment* for the filter. This comment appears in the **Filters** window and in all filter lists, and allows you, for example, to create a more complete description of the filter's properties. You can sort any list of filters by either the ***Filter*** name or the ***Comments*** column.

Specify the parameters for *Address Filter*, *Protocol Filter*, and/or *Port Filter* according to the directions given below and click **OK** to create the new filter. The new filter will appear in the **Filters** window and all other filter lists and can be enabled by checking the box beside the filter's name. (For more, please see "Using filters" on page 196.) To edit an existing filter, double-click on its name in any filter list to open the **Edit Filter** dialog with that particular filter's parameters displayed.

Specifying address filter parameters

To specify an address filter, check the checkbox to the left of the *Address Filter* section of the **Edit Filter** dialog. Notice that there is room for two addresses. Between these two address text entry boxes are two drop-down lists.



Filter: Untitled

Color: ■

Type: Simple

Comment:

☑ Address filter

Address 1: 00:00:00:00:00:00

Type: Physical

Address 2: ○ 00:00:00:00:00:00 ● Any address

☑ Protocol filter

☑ Port filter

Port 1: 0

Type: TCP-UDP

Port 2: ○ 0 ● Any port

Figure 11.4 Edit Filter dialog, Simple view

The topmost specifies the *Type* of addresses you want to enter. Both addresses must be of the same type and must be entered in the correct format for the address type you have selected in this drop-down list. For more on addresses and their notation formats, see Appendix B, “Addresses and Names” on page A-13.

The second drop-down list specifies the send/receive relationship between the two addresses. The default value is to match all packets going in either direction between *Address 1* and *Address 2*. You could instead match only traffic going from *Address 1* to *Address 2*, or match only traffic going the other direction.

You must enter a valid address in *Address 1*, but you can choose either a particular address for *Address 2* or simply choose *Any Address* by clicking the radio button beside that choice. This allows you to match all traffic of a particular *Type* to and/or from a single address or address range.

The drop-down list immediately to the right of each address text entry box contains the most recently used addresses. The drop-down list arrows further to the right of each *Address* box allow you to specify an address by reference to either the Name Table or any reachable name resolution servers. Selecting *Name Table* from this drop-down list takes you to the **Select Name** dialog, where you can select any address stored in the

Addresses view of the **Name Table** by clicking on its entry. If you choose *Resolve*, EtherPeek will query the appropriate name service to attempt to find a name for the address, or an address for the name entered in the edit box.

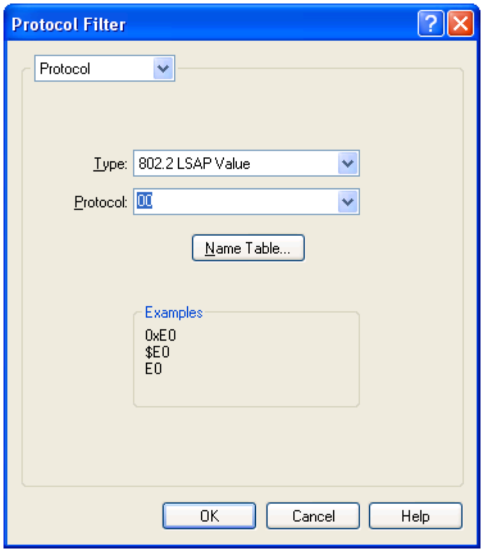
You can use the asterisk * character as a wildcard when specifying addresses. The program will replace the asterisk with its most inclusive equivalent. For example, if you specified an IP address of 192.216.124.* the program would interpret the wildcard to mean “all possible values for this element.” If you save and reopen a filter with this example, you will see that the program has interpreted the address as 192.216.124.0/24, which is standard dotted decimal/subnet notation for all addresses within the specified Class C network.

Note: Address filters support CIDR for the IP address space.

Specifying protocol filter parameters

To specify a protocol filter, check the checkbox to the left of the *Protocol Filter* section of the **Edit Filter** dialog. Click the **Protocol…** button to bring up the **Protocol Filter** dialog. At the top of the **Protocol Filter** dialog is a drop-down list whose default value is *ProtoSpec*. This allows you to choose the method EtherPeek will use to define and test for the protocol you select.

Note: In general, ProtoSpecs provides the easiest path to nearly every protocol and sub-protocol type. The secondary edit capability is provided for new or unusual protocol situations and also for backward compatibility.



Protocol

Type: 802.2 LSAP Value

Protocol: 00

Examples
0xE0
$E0
E0

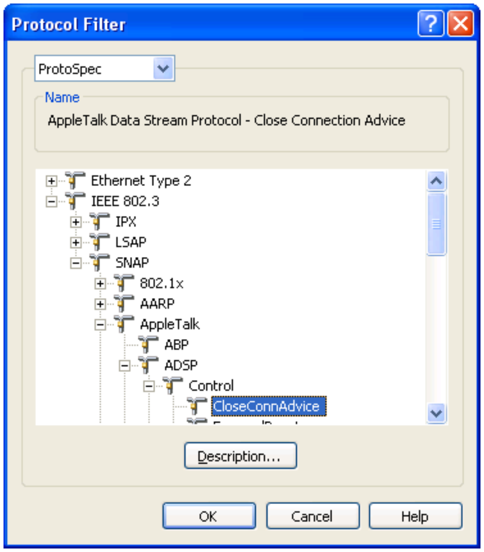
Figure 11.5 Specifying a protocol using the encoding in the 802.2 LLC header

If you choose *Protocol* rather than *ProtoSpec* as your protocol definition method, the dialog switches to its ***Protocol*** view (Figure 11.5). From this view, you can choose *Ethernet Protocol*, *802.2 LSAP Value*, or *802.2 SNAP ID* from the *Type* drop-down list.

Each of these choices represents a distinct method for denoting the protocol of the network data framed by the packet. Each has its own format for representing these protocols. Choose the type of protocol and enter a value in the appropriate format. You can use a wildcard in these entries. The asterisk character (*) is a wildcard and stands for zero or more alphanumeric characters.

You may also select a protocol from the Name Table by clicking the **Name Table…** button and choosing from the protocols listed there. In this case, the name of the protocol selected rather than the discrimination values will appear in the *Protocol* text entry box. For more information about Ethernet frames and protocols, see Appendix A, “Packets and Protocols” on page A-3.

If you choose *ProtoSpec* as your protocol definition method, your protocol choices are listed in the default ***ProtoSpec*** view of the **Protocol Filter** dialog (Figure 11.6).



Protocol Filter

ProtoSpec

Name
AppleTalk Data Stream Protocol - Close Connection Advice

Figure 11.6 Specifying a protocol from the ProtoSpecs list as the object of a filter

The protocols are listed in two hierarchies: IEEE 802.3 and Ethernet Type 2, corresponding to the newer and the older Ethernet standards, respectively. For example, IP protocol stacks are nearly always written to use the older Ethernet Type 2 packets, while AppleTalk is usually framed in 802.3 headers and trailers. Still, both protocols are represented under both hierarchies, as other implementations are possible.

Click on the + plus or - minus signs to expand or collapse the hierarchical list of protocols.

To specify a protocol, highlight it in the list. The active choice will appear above the list box in the space between the list box and the *ProtoSpec/Protocol* drop-down list. If the protocol you select has other sub-protocols listed under it, your filter will match any of these sub-protocols as well.

To finish choosing the protocol, click **OK** at the bottom of the **Protocol Filter** dialog to return to the **Edit Filter** dialog. Your protocol choice will be shown in the *Protocol Filter* section of the **Edit Filter** dialog.

Protocol descriptions

To find more information about a particular protocol, select it in the list and click the **Description…** button at the bottom of the **Protocol Filter** dialog. Brief descriptions of many of the most commonly used protocols are included with EtherPeek and will appear in a new **Protocol Description** dialog when you click the **Description…** button. For more on how EtherPeek and ProtoSpecs deal with protocols, see Appendix A, “Packets and Protocols” on page A-3.

Specifying port filter parameters

To specify a port filter, check the *Port Filter* checkbox. Notice that there is room for two ports. Between these text entry fields are two drop-down lists. The topmost, whose default value is *TCP-UDP*, specifies the protocol which uses the ports you want to enter. Both ports must be of the same type and must be entered in the correct format for the type you have selected in this drop-down list. For more on ports, sockets, and their notation formats, see “Ports and sockets” on page A-18.

The second drop-down list specifies the source/destination relationship between the two ports. The default value is to match all packets going in either direction between *Port 1* and *Port 2*. You could instead match only traffic going from *Port 1* to *Port 2*, or match only packets going the other direction.

Enter a port in *Port 1*. Alternatively, you can choose a recently used port definition from the drop-down list, or click the arrow further to the right to open a **Select Name** dialog, in which you can choose a port from the list of those entered in the Name Table.

You must enter a valid port designation in *Port 1*, but you can choose either a particular port for *Port 2* or simply choose *Any Port* by clicking the radio button beside that choice. This allows you to match all traffic to and/or from a specific port.

Advanced filters

The ***Advanced*** view of the **Edit Filter** dialog allows you to create filters that match any of the filter parameters supported by EtherPeek (see “Filter parameters” on page 201). In addition, it allows multiple parameters to be joined with logical AND, logical OR, and logical NOT statements to create very precise tests in a single named filter.

To open the edit filter dialog advanced filter view

To create a new advanced filter, choose **Filters** from the **View** menu (or press **Ctrl + M**) to view the **Filters** window. Click the **Insert** button to bring up the **Edit Filter** dialog in its default ***Simple*** view. Choose *Advanced* from the *Type* drop-down list in the upper right to switch to the ***Advanced*** view of the **Edit Filter** dialog.

The default name "*Untitled Filter*" shows in the *Filter* text entry box where you can enter a new name. The color assigned to this filter (black is the default for a new filter) is shown in the color swatch at the top of the **Edit Filter** dialog, to the right of the *Filter* box. Click the arrow to the right of this color swatch to bring up the drop-down list of color choices.

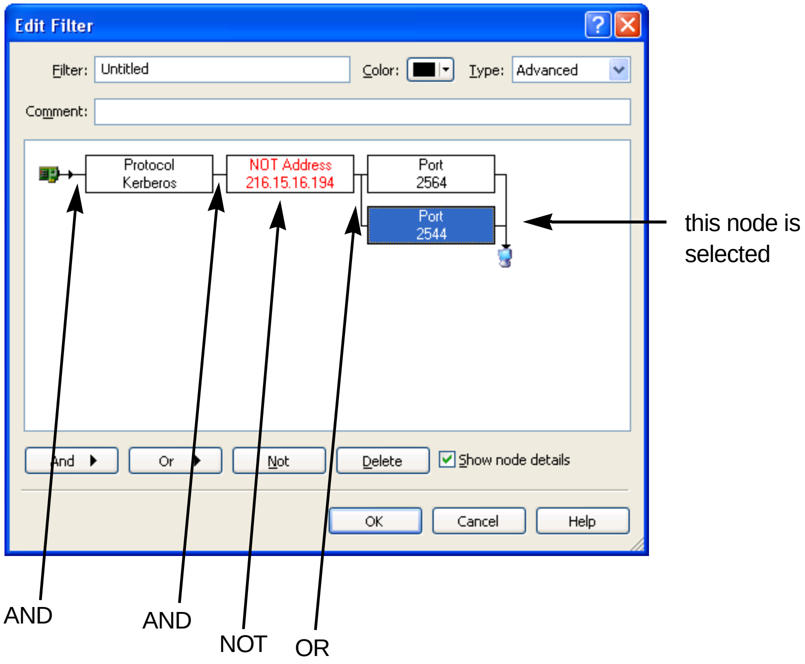
In addition to its name, you can enter a *Comment* for the filter. This comment appears in the **Filters** window and in all filter lists, and allows you, for example, to create a more complete description of the filter's properties. You can sort any list of filters by either the ***Filter*** name or the ***Comments*** column.

Specify the parameters for the new filter according to the directions given below and click **OK** to create the new filter. The new filter will appear in all the filter lists.

To edit any existing filter, select the filter and click the **Edit** button, or simply double-click on its name in any filter list to open the **Edit Filter** dialog with that particular filter's parameters displayed and ready to edit.

Logical AND, OR, and NOT operators in advanced filters

When you open the ***Advanced*** view of the **Edit Filter** dialog, you will see a screen with an icon in the upper left corner representing a network adapter. When you add the first node to the filter, a new icon will appear representing the computer or its capture buffer, and an arrow will appear connecting the card to the computer. The arrow points from the network adapter icon to the icon for the computer on which EtherPeek is installed. As you add sets of filter parameters, called *filter nodes*, the relationship between and among these filter nodes is displayed on this screen in a logical tree or flow diagram, starting from the network side and building toward the computer icon. Each filter node you define is treated as a building block and displayed as a labeled rectangle. The internal logic of an advanced filter is that of a pass filter. That is, any packet which could pass through the criteria established in the flow diagram is said to match the advanced filter.



Edit Filter

Filter: Untitled

Color:

Type: Advanced

Comment:

☑ Show node details

Figure 11.7 Advanced view shows nodes joined by logical AND, OR, and NOT

The *Show node details* checkbox causes the rectangles representing filter nodes to display an approximation of the logical content of each filter node. If this checkbox is unchecked, nodes will display only their parameter type.

To view the details of any node, double-click on the rectangle that represents it. This will open the appropriate edit dialog, displaying all the specifications for that node. You can click **Cancel** if you do not wish to make any changes.

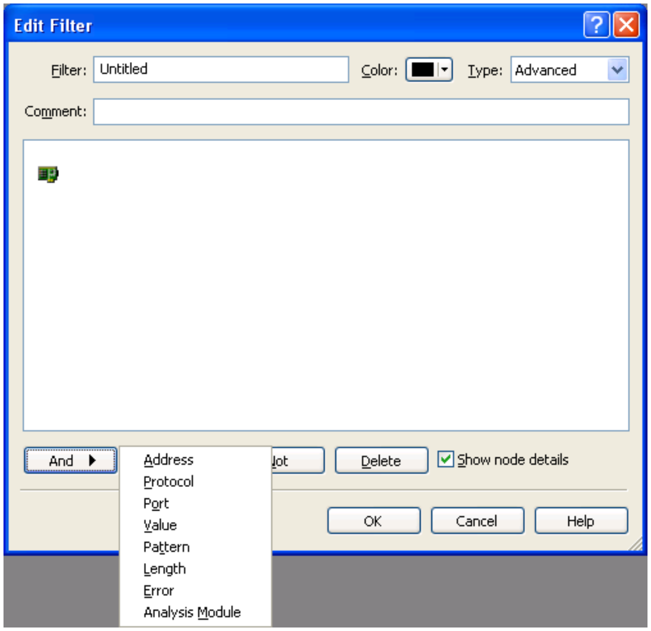
The graphic display helps to make clear the logical relationship of the various filter nodes you create in the ***Advanced*** view. The relationships are limited to three simple choices represented in the buttons at the bottom of the **Edit Filter** dialog: **And**, **Or**, and **Not**. Figure 11.7 shows how these relationships are represented graphically in the ***Advanced*** view. Table 11.2 describes the meaning and use of each of these three buttons, plus the **Delete** button, in detail.

Table 11.2 Advanced view of the Edit Filter dialog, buttons and functions

Button	Description
And	Use this button to create the first node of a new advanced filter. Clicking **And** creates a new node just after (to the right of) the currently selected node and establishes an **And** relationship with the argument of that node. That is, a packet must meet both the previous node's criteria and the newly added node's criteria in order to match the filter.
Or	Clicking the **Or** button creates a new filter node in parallel with the node that was selected when you pressed the **Or** button. That is, the new filter node will get the same inputs as the filter node that was selected when you pressed the **Or** button, and packets meeting the criteria of either filter node will pass through, or match this stage. Please note that when you have created a set of filter nodes that is several stages deep, choosing an early node (one far to the left) and pressing the **Or** button will create a parallel path that bypasses any nodes further to the right. In other words, the new OR statement will create a node on a path that is parallel to the whole of the remaining structure, not just to the single node selected when you pressed **Or**.
Not	Negates or inverts the filter node selected when you pressed the **Not** button, changing it from a pass node to a blocking node. All packets *except* those matching the criteria inside the negated node will now be passed to the next stage.
Delete	Deletes the selected filter node.

Adding a filter node

Click on the **And** or the **Or** button in the ***Advanced*** view of the **Edit Filter** dialog and choose a filter type from the drop-down list to begin specifying the parameters of the new filter node. The following sections describe each of the filter parameters you can use for a node in the ***Advanced*** view of the **Edit Filter** dialog



Filter: Untitled

Color: ■

Type: Advanced

Comment:

☑ Show node details

Figure 11.8 Choosing the filter node type for a new advanced filter

Address filter nodes

To specify an address filter node, choose *Address* from the drop-down list to open the **Address Filter** dialog. This dialog offers exactly the same choices as the *Address Filter* section of the ***Simple*** view of the **Edit Filter** dialog, but laid out in a slightly different order, as shown in Figure 11.9.

Set the parameters for the address filter node. For detailed instructions, see “Specifying address filter parameters” on page 203.

When you have finished specifying the filter node, click **OK** to return to the ***Advanced*** view of the **Edit Filter** dialog. The filter node you have just created will be selected, with the first address in the filter displayed, or, if *Show node details* is unchecked, with the simple label: *Address*.

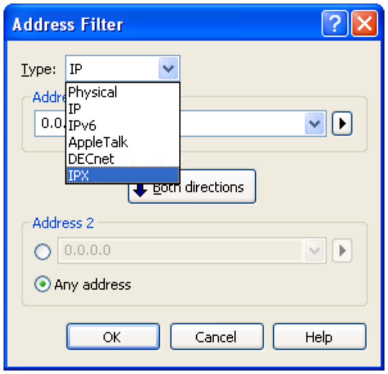


Figure 11.9 Advanced filters: the Address Filter dialog

Protocol filter nodes

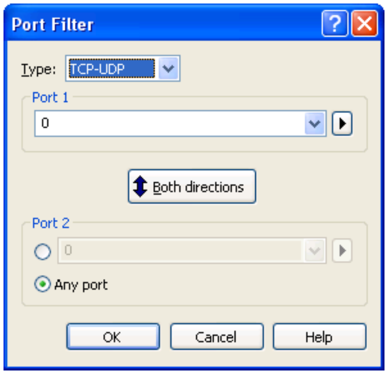
To specify a protocol filter node, choose *Protocol* from the drop-down list to open the **Protocol Filter** dialog. At the top of the **Protocol Filter** dialog is a drop-down list offering the choice of *Protocol* or *ProtoSpec*. This allows you to choose the method EtherPeek will use to define and test for the protocol you select.

For a detailed description of the **Protocol Filter** dialog and how to use it, please see “Specifying protocol filter parameters” on page 205.

When you have selected the protocol, click **OK** at the bottom of the **Protocol Filter** dialog to return to the ***Advanced*** view of the **Edit Filter** dialog. The node you have just created will be selected and show the name of the protocol that describes the parameter of the newly constructed node, or, if *Show node details* is unchecked, it will simply be labeled *Protocol*.

Port filter nodes

To specify a port filter node, choose *Port* from the drop-down list to open the **Port Filter** dialog. This dialog offers exactly the same choices as the *Port Filter* section of the ***Simple*** view of the **Edit Filter** dialog, but laid out in a slightly different order, as shown in Figure 11.9.



Type: TCP-UDP

Port 1

0

Port 2

○ 0

● Any port

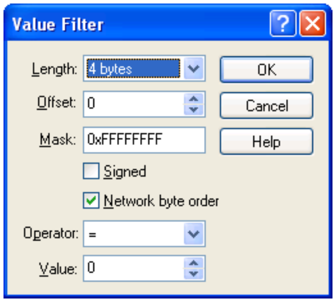
Figure 11.10 Advanced filters: the Port Filter dialog

Set the parameters for the port filter node. For detailed instructions, see “Specifying port filter parameters” on page 208.

When you have finished specifying the filter node, click **OK** to return to the ***Advanced*** view of the **Edit Filter** dialog. The filter node you have just created will be selected, with the first port in the filter displayed, or, if *Show node details* is unchecked, with the simple label: *Port*.

Value filter nodes

To specify a value filter node, choose *Value* from the drop-down list to open the **Value Filter** dialog.



Length: 4 bytes

Offset: 0

Mask: 0xFFFFFFFF

□ Signed

☑ Network byte order

Operator: =

Value: 0

Figure 11.11 Editing a Value Filter in the Advanced view of the Edit Filter dialog

A value filter is used to test whether the specified bits at a specified location in a packet have the specified relationship to a numerical value you set. If the particular part of a tested packet has a numerical value with the relationship you specified to the numerical value you set, the packet matches the filter.

Tip You can quickly create a value filter node matching any item in the ***Decode*** view of a **Packet Decode** window or the Decode Pane of any ***Packets*** view, by highlighting the item and clicking the **Make Filter** button, or by choosing **Make Filter…** from the context menu (right-click).

The **Value Filter** dialog can be understood as an IF:THEN statement of the following form:

Default values:

If the number of *Length* “*4 bytes*”
at *Offset* “*0*”
with a *Mask* of “*0xFFFFFFFF*”,
where the packet value [unchecked means “is not”] *Signed*
and it [✓ means “is in”] *Network byte order*,
has the relationship defined by the *Operator* “=”
to the *Value* of “*0*”
then the packet matches the filter.

Parameters:

If the number of length (*Length*)
at offset (*Offset*)
with a mask of (*Mask*)
where the packet value [is/is not] *Signed*
and it [is in/is not in] *Network byte order*,
has the relationship (*Operator*)
to the value (*Value*)
then the packet matches the filter.

Taking each of these parameters in turn, you need to specify:

Length

What is the size of the number you wish to test? The choices are: *4 bytes*, *2 bytes*, or *1 byte*. Remember that the mask specified below must be of the correct format to properly mask a number of this length.

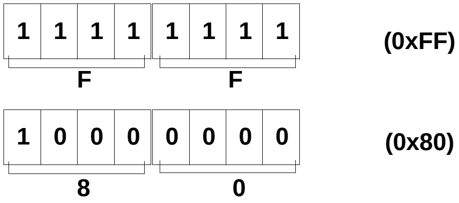
Offset

What is the location in the packet of the beginning of the first byte of the number you want to test? The location is specified as the distance (in bytes) from the beginning of the packet to the beginning of the first byte of the number you wish to test, or its *offset* from the first byte of the packet. If you want to test the first byte, it begins 0 bytes away from the beginning of the packet, so enter an offset of "*0*." The second byte of the packet begins 1 byte away, so it is at offset 1, and so on. Enter a decimal number or a hex number with the "*0x*" prefix for the offset.

To see the offset and mask for any element in a packet ***Decode*** view, click the **Show Offsets** button.

Mask

The number in this field is used to isolate particular bits inside of the byte or bytes you specified in the Length and Offset parameters. The value of the Mask is logically AND'ed with the value present in the byte or bytes you choose to test, and the result is examined. If you choose to test a one-byte number and enter a mask of "0xFF", EtherPeek will examine all of the bits in the byte. With a mask of "0x80" EtherPeek would examine only the most significant bit of that byte, as shown below:



You can enter a mask value in hex (*0x* prefix) or in decimal format, but it will display in hex format when the filter is re-opened for editing.

Signed

Click the checkbox labeled *Signed* if the number at the offset you chose to test is signed. If it is not signed, leave the box unchecked. For example, unsigned "0xFF" is decimal 255, but signed "0xFF" is decimal -1 (minus one).

Network byte order

EtherPeek must be told in what order to evaluate the bytes at the offset you specified. Make sure the checkbox beside *Network byte order* is checked (the default) if the bytes are in network byte order, as they usually are for most network packets. Uncheck this checkbox if the bytes at the specified offset are *not* in network byte order.

Operator What is the relationship of the number you are testing to the value you have chosen? Remember that the mask will be applied to the bytes you specified before their value is calculated. Your choices are: equal to, greater than, less than, greater than or equal to, less than or equal to, or not equal to. Choose a relationship from the drop-down list.

Value The number in this field is the constant that EtherPeek compares to the value it obtains by applying the *Mask* to the byte or bytes specified in *Length* and beginning at the location in the packet specified in *Offset*. If that calculated value has the specified relationship to the value you enter here, then the packet matches. You can enter a number in hex (with the 0x prefix), binary (with the % prefix, such as *%1101*), or in decimal format.

Note: Network byte order, also known as Big Endian (most significant byte first), is the form in which most protocols write their data. Little Endian (least significant byte first), or not network byte order, is the native form for Intel machines. When they communicate, however, they typically write the data in network byte order to insure compatibility with others. A few protocols such as SMB (a part of NetBIOS) may encode data in Little Endian, or not network byte order. SMB data can ride inside an IP packet. In such cases the Ethernet header and the IP header would be in network byte order, but the SMB portion of the packet would be in Little Endian, or not network byte order.

When you have finished specifying the filter node, click **OK** to return to the ***Advanced*** view of the **Edit Filter** dialog. The filter node you have just created will be selected and show the value and relationship for which this node is testing, or, if *Show node details* is unchecked, it will simply be labeled *Value*.

Pattern filter nodes

To specify a pattern filter node, choose *Pattern* from the drop-down list to open the **Pattern Filter** dialog.

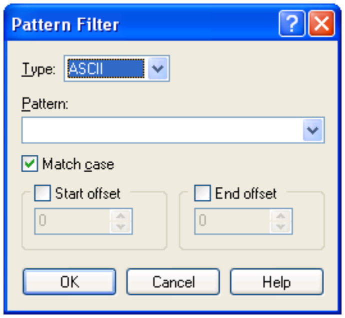


Figure 11.12 Advanced filters: the Pattern Filter dialog

Pattern filter nodes test packets for the presence of a specific character string within the bounds of a packet. Enter a character string up to 255 characters long in the *Pattern* box. Use the *Match case* checkbox to match case as well as character form of the string. Specify where in the packet you want EtherPeek to start or end the search by specifying either a *Start offset*, an *End offset*, or both. If you select neither of these offset options, EtherPeek will search the whole packet. Limiting the area of search can speed performance.

Note: Offset is a measure of the distance in bytes from the beginning of the packet. The first byte of the packet begins 0 bytes away from the first byte of the packet, and is therefore at offset 0. The second byte of the packet begins one byte away at offset 1, the third byte begins at offset 2, and so on. To see the offset and mask for any element in a packet decode, click the **Show Offsets** button.

Any packet containing the pattern you specified (and in the exact case, if you specified *Match case*) will match the filter if it can be found within the offsets you specified.

When you have finished specifying the filter node, click **OK** to return to the ***Advanced*** view of the **Edit Filter** dialog. The filter node you have just created will be selected and labeled with as much of the search pattern as will fit, or, if the *Show node details* checkbox is unchecked, with the simple label: *Pattern*.

Length filter nodes

To specify a length filter node, choose *Length* from the drop-down list to open the **Length Filter** dialog.

Specify the length range (in bytes) of the packets you wish to match this filter by checking *Maximum length* and/or *Minimum length* and entering a value in bytes in the respective text entry boxes.

When you have finished specifying the filter node, click **OK** to return to the ***Advanced*** view of the **Edit Filter** dialog. The filter node you have just created will be selected and show a representation of the length range you have chosen, or, if *Show node details* is unchecked, it will simply be labeled *Length*.

Error filter nodes

To specify an error filter node, choose *Error* from the drop-down list to open the **Error Filter** dialog.

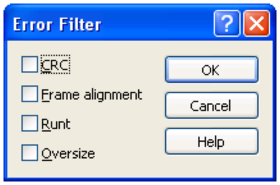


Figure 11.13 Adding an error filter node to an advanced filter

Choose the type of errors you would like to capture with this filter by using the checkboxes beside each type.

CRC Cyclic Redundancy Check is a type of error that indicates data was corrupted in transmission.

Frame alignment This is another indication of corrupted data.

Runt Runt packets are less than 64 bytes in length.

Oversize Oversize packets are over 1518 bytes in length for ordinary Ethernet packets.

Unlike all other types of filters, the error filter node allows you to connect the internal parameters of a single filter node with logical OR statements. Normally, a packet would have to pass all of the tests within a single node in order to match that filter node. The error filter node, in contrast, will match packets that match at least one of the criteria you enable by selecting their checkbox.

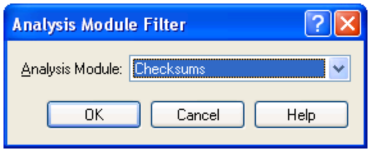
For more on error types, see "Error types and error packets" on page 159.

Note: Error packets may not be passed to EtherPeek in some computer setups. For more information, see "Ethernet interface requirements" on page 10.

When you have finished specifying the filter node, click **OK** to return to the ***Advanced*** view of the **Edit Filter** dialog. The filter node you have just created will be selected and show the initial of each error type you chose, or, if *Show node details* is unchecked, the node will have the simple label: *Error.*

Analysis Module filter nodes

You can use Analysis Modules to filter packets. Packets which are handled by the Analysis Module named in the filter will match the filter. To specify an Analysis Module filter node, choose *Analysis Module* from the drop-down list to open the **Analysis Module Filter** dialog. For details on the behavior of individual Analysis Modules and the kinds of packets each one is designed to handle, see Chapter 13, "Analysis Modules" on page 247.



Analysis Module: Checksums

Figure 11.14 Adding an Analysis Module filter node to an advanced filter

Saving and loading filters

You can save and load filters. This allows you to create multiple sets of filters for different requirements. Click the **Export** button in the **Filters** window to save the whole set of filters under a new name. Alternatively, you can save a selection of filters by highlighting them and using the **Export Selected…** command from the context menu (right-click). Either action brings up a **Save As** dialog, in which you can specify the name and path under which to save the file. All filter files must use the *.flt file extension.

To save the existing set of filters under a new name:

1. Open the **Filters** window by choosing **View** > **Filters** or typing **Ctrl + M**.
2. Click the **Export** button to open the **Save** dialog.
3. Give the file a name and save it by clicking the **Save** button.

When you import a previously saved group of filters into the **Filters** window, it adds them to the filters already there.

To import filters from another *.flt file into the existing **Filters** window:

1. Click the **Import** button at the left of the **Filters** window.
2. Use the dialog to navigate to the saved filters file of your choice.
3. Click **Open** to load the selected file.

Tip Alternatively, you can choose a recently used filters file from the drop-down list beside the **Import** button.

Note: Imported filters are added to the existing filters list. Duplicates of existing filters will be ignored if they have identical parameters as well as identical names. Filters with the same name but different parameters will be added with “*copy*” added to their names.

12

Triggers, Alarms and Notifications

The evidence of network problems is often fleeting. EtherPeek provides a variety of real-time monitoring tools to help you automate the search for anomalies and problem conditions.

EtherPeek can be programmed to take a variety of actions based on network traffic or statistical events. There are four classes of actions that can be automated:

- Actions you assign directly to a trigger using one of the **Trigger** dialogs
- Actions you assign directly to an Analysis Module using the ***Analysis Modules*** view of the **Options** dialog
- Actions you associate with notifications using the ***Notifications*** view of the **Options** dialog
- Notifications sent when user-defined **Alarm** conditions are detected in statistics outputs

Triggers, alarms and Analysis Modules can be configured to make notifications. When they do, these notifications will execute any action(s) you have assigned to that particular level of severity of notification: write to the Log file, send an email message, play a sound file, or run a program.

Triggers and Analysis Modules scan all incoming packets for matching conditions. Alarms periodically query statistics functions to find their specified conditions. This chapter describes the creation and function of triggers, alarms, and Notifications. For more on Analysis Modules, see Chapter 13, “Analysis Modules” on page 247.

In this Chapter:

Triggers

Triggers are used to start or stop capture in a Capture window at a specified time or network event. They are very useful for pinpointing the origins of intermittent network problems. For example, you can set a start trigger so that capture begins when a problem occurs. Conversely, you can stop capturing when the problem occurs so that you can see exactly what happened just prior to the observed symptom. Alternatively, if you know that problems occur at a particular time, you can set a time event to begin capturing packets during that time. Start and stop triggers can help you uncover many hard-to-find network problems.

Trigger events

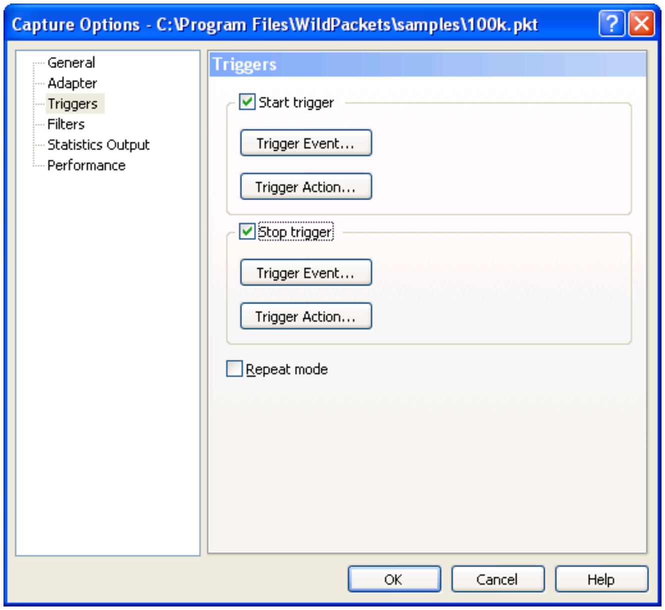
A trigger event can be one of the following:

- A user-specified time occurs.
- A packet matches one of any number of user-specified filters.
- A stop trigger may be set to trip when a specified number of bytes are captured.

In addition, when both a start and stop trigger are specified, you can use a repeat mode which resets the start trigger each time the stop trigger is tripped.

Note: Although the same list of filters is used for capture filtering, triggering, and post-capture selection in the **Select** dialog, enabling the use of a filter in one area does not enable that filter in any other area. For example, specifying a filter in a trigger does not mean that the filter will be applied during capture activity.

To create a trigger, you must make the Capture window for which you want to create the trigger the active window. If it is an existing Capture window, you must also stop capture before creating a new start trigger. With the target Capture window active, choose **Capture Options…** from the **Capture** menu, or double-click the current adapter listing in the status bar of the Capture window to open the **Capture Options** dialog. Click the *Triggers* item in the navigation pane to display the ***Triggers*** view.



☑ Start trigger

☑ Stop trigger

□ Repeat mode

Figure 12.1 Triggers view of the Capture Options dialog

From the ***Triggers*** view of the **Capture Options** dialog you can set a Start Trigger, a Stop Trigger, or both; define the triggering event(s), and specify what action(s) the trigger(s) will take. Details for each of these are discussed below.

About start triggers

A start trigger instructs a Capture window to remain idle, reviewing but not capturing packets, until a specified event occurs. When the trigger event occurs, you can specify that the Capture window:

- begin capture (according to the set-up of the Capture window it triggers).
- send a notification of the specified severity.
- do both of the above.

While idle, all packets on the network are reviewed but not captured. The start trigger tests all network traffic against any filters you have set as trigger events, but ignores any filters enabled for the Capture window itself. Once the start trigger event occurs, the configuration you set for the Capture window itself takes over; including any enabled filters, packet slicing options, use of buffer memory, columns to be displayed, and so forth.

When you have finished specifying the start trigger and you click **OK** in the ***Triggers*** view of the **Capture Options** dialog, the **Start Capture** button in the active Capture window changes to **Start Trigger**. The trigger will not begin reviewing incoming packets or checking to see if its assigned time has arrived until you click this button.

When you click on the **Start Trigger** button it changes to **Abort Trigger** and the start trigger begins searching the incoming packets and/or the system clock for the event(s) you specified. When any one of the specified events occurs, the actions you specified are performed and the button changes back to **Stop Capture**. Clicking on the **Abort Trigger** button at any time will stop the process and return the Capture window to its normal state.

Tip If you have already captured traffic in the current window and wish to add the new capture to the old, hold down the **Shift** key when you click the **Start Trigger** button. This will bypass the warning dialog asking if you wish to save the existing contents of the Capture window. When the start trigger is tripped, capture will resume just as it does when you use **Shift + Click** to restart capture manually.

When you first open the Capture window, the status bar at the bottom will show *Idle*. When you press the **Start Trigger** button, the status bar will show *Waiting for Start Trigger*. When the trigger event occurs, the status bar will show *Capturing*. If, instead, you press the **Abort Trigger** button before the start trigger is tripped, the status bar message will return to *Idle*.

Creating a start trigger

To specify a start trigger:

1. Open a new Capture window, or make an existing Capture window the active window and make sure capture in that window is stopped.
2. Choose **Capture Options…** from the **Capture** menu to open the **Capture Options** dialog, and click the *Triggers* item in the navigation pane to open the ***Triggers*** view.
3. Check the *Start trigger* checkbox in the ***Triggers*** view to enable the **Trigger Event…** and the **Trigger Action…** buttons.
4. Click the **Trigger Event…** button to specify the event which will trip the trigger. You can specify a time event, a filter event, or both. If you specify both, the first one to occur will trip the trigger. Please see “Setting a time trigger event” below and “Setting a filter trigger event” on page 228 for details.

5. Click the **Trigger Action…** button to define what will take place when the trigger is tripped. You can choose to begin capture in the selected Capture window, to send a notification of a specified severity, or both. Please see “Specifying trigger actions” on page 228 for details.

6. When you have specified both the event(s) and the action(s) for the trigger, click the **OK** button in the **Capture Options** dialog to create the trigger for the active Capture window.

Setting a time trigger event

To create a trigger that will trip at a specified time, or date and time:

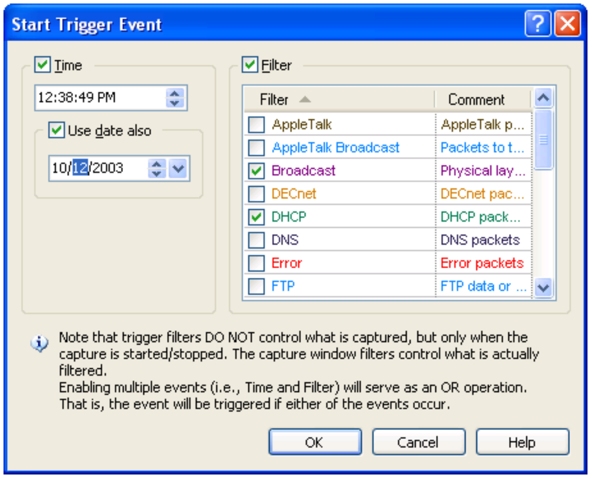
1. From the ***Triggers*** view of the **Capture Options** dialog, click the **Trigger Event…** button to open the **Trigger Event** dialog.

2. Click in the *Time* checkbox in the **Trigger Event** dialog.

3. Edit the time directly or use the arrows at the right of the time box to set the time for the trigger event. When this time is reached, the trigger will trip.

4. Click the *Use date also* checkbox and enter the date in a similar fashion if you want to include this parameter. At the far right of the date text entry box is a drop-down list that allows you to choose the date by reference to monthly calendars.

5. Click **OK** to return to the ***Triggers*** view.



☑ Time

12:38:49 PM

☑ Use date also

10/12/2003

☑ Filter

	Filter	Comment
□	AppleTalk	AppleTalk p...
□	AppleTalk Broadcast	Packets to t...
☑	Broadcast	Physical lay...
□	DECnet	DECnet pac...
☑	DHCP	DHCP pack...
□	DNS	DNS packets
□	Error	Error packets
□	FTP	FTP data or ...

Note that trigger filters DO NOT control what is captured, but only when the capture is started/stopped. The capture window filters control what is actually filtered.
Enabling multiple events (i.e., Time and Filter) will serve as an OR operation. That is, the event will be triggered if either of the events occur.

Figure 12.2 Start Trigger Event dialog

Setting a filter trigger event

The **Trigger Event** dialog includes the same list of filters that appears in the **Filters** window. To view the details of filters or to make edits or duplicates of any filter, choose **Filters** from the **View** menu or press **Ctrl + M** to open the **Filters** window. You can also open any filter in its appropriate view of the **Edit Filter** dialog by double-clicking on its listing in this or any other list of filters.

Note: Although the same list of filters is available for use in capture filtering and triggering, enabling the use of a filter in one area does not enable that filter in any other area. For example, specifying a filter in a trigger does not mean that the filter will be applied during capture activity.

To set a trigger based on a filter:

1. Check the checkbox beside any filter or filters you wish to enable. The checkbox labeled *Filter* will be checked (or unchecked) automatically, to show that this option is enabled.
2. You can set one or more filters, or you can enable both filter and time events in a single trigger. Each enabled trigger event is independent of the others that have been enabled; that is, the trigger action is started if any one of the enabled trigger events occurs.
3. When you have selected the trigger event(s), click **OK** to return to the ***Triggers*** view of the **Capture Options** dialog.

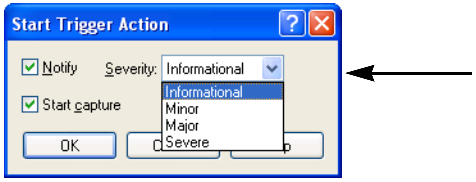
Specifying trigger actions

In a start trigger, you can perform one or more of the following actions when the trigger event occurs:

- Start capturing packets in this Capture window, with all its enabled filters, packet slicing options, or any other options you have enabled for it.
- Send a notification of the *Severity* you specify using the drop-down list. The default value is *Informational*, the lowest level.

Note: If you have already assigned actions to these severity levels in the ***Notifications*** view of the **Options** dialog, then the actions assigned there will be executed when notification occurs. For more on Notifications and their associated actions, see “Notifications” on page 237.

To choose one or more of these trigger actions, check the appropriate checkboxes in the **Start Trigger Action** dialog. When you have specified the action(s), click **OK** to return to the ***Triggers*** view of the **Capture Options** dialog.



☑ Notify Severity: Informational

☑ Start capture

Notification Severity Levels drop-down list

Figure 12.3 Start Trigger Action dialog

About stop triggers

A stop trigger tells a Capture window to stop capture, send a notification, or both when a specified event occurs. When you use a stop trigger, you should consider what you want to happen if the buffer becomes full before the trigger event occurs. Please see “Capture options: general” on page 52 for details.

When a stop trigger is active, the message *(Stop Trigger Active)* appears in the status bar at the bottom of the Capture window.

When the trigger event occurs, you can specify that EtherPeek:

- stop capture.
- send a notification of the specified severity.
- perform both of these actions.

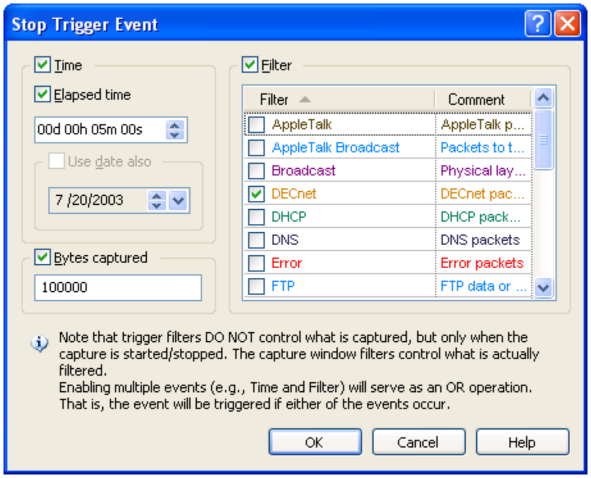
Important! When a Capture window is to be used for AutoCapture, you must set a stop trigger.

Creating a stop trigger

The process of creating a stop trigger is virtually identical to the one described for “Creating a start trigger” above. Click the *Stop trigger* checkbox in the ***Triggers*** view of the **Capture Options** dialog. **Trigger Event…** offers the same choices as presented for start triggers above, with two important additions. The stop trigger event can be based on *Elapsed time*, specified in the form *00d 00h 00m 00s*, for days, hours, minutes, and seconds from the moment the stop trigger is enabled. You can also base the stop trigger on the number of *Bytes captured*. Check the checkbox beside either or both of these parameters and fill in the text entry box in the appropriate format.

Tip If you choose *Bytes captured*, EtherPeek has the intelligence to capture all the bytes in the last packet—the packet that causes the counter to reach your *Bytes captured* limit.

Because of this, the number of bytes actually captured will be the value you set in *Bytes captured*, minus as little as one byte, plus the length of the last packet captured.



☑ Time

☑ Elapsed time

00d 00h 05m 00s

□ Use date also

7 /20/2003

☑ Bytes captured

100000

☑ Filter

Filter	Comment
□ AppleTalk	AppleTalk p...
□ AppleTalk Broadcast	Packets to t...
□ Broadcast	Physical lay...
☑ DECnet	DECnet pac...
□ DHCP	DHCP pack...
□ DNS	DNS packets
□ Error	Error packets
□ FTP	FTP data or ...

Note that trigger filters DO NOT control what is captured, but only when the capture is started/stopped. The capture window filters control what is actually filtered.
Enabling multiple events (e.g., Time and Filter) will serve as an OR operation. That is, the event will be triggered if either of the events occur.

Figure 12.4 Stop Trigger Event dialog

In **Trigger Action…** for a Stop trigger, instead of the possible action of starting to capture packets, you can click the *Stop capture* checkbox to stop capturing packets when the specified event occurs.

About repeat mode

When both a *Start trigger* and a *Stop trigger* are enabled, you can set the Capture window to apply these triggers in *Repeat mode* by checking the checkbox beside that item in the ***Triggers*** view of the **Capture Options** dialog (Figure 12.1 on page 225). In repeat mode, the Capture window will reset the start trigger each time the stop trigger is tripped. Any stop trigger actions are completed normally. If capture restarts when *Repeat mode* is enabled, the buffer contents are retained and any new packets are added to those already in the buffer. Repeat mode allows you to capture multiple occurrences of the same event(s) with a single Capture window.

Note: Repeat mode operates in a slightly different way when used in a Capture window or capture template that is part of an AutoCapture (*.wac) file. In this case, the start trigger will not be reset until after capture is completed for all Capture windows in the AutoCapture file and any actions specified in the *Send options* section of the

AutoCapture file have been completed. The buffer is also refreshed (cleared) if capture restarts. For more about AutoCapture files, please see “AutoCapture” on page 88.

Alarms

Alarms query a specified Monitor statistics function approximately once per second, testing for the user-specified alarm condition(s), and/or for the user-specified alarm resolution condition. On matching any of these tests, the alarm function sends a notification of a user-specified severity.

Unlike triggers, filters and Analysis Modules, alarms do not query all incoming packets directly. Instead, alarms query statistics functions, looking for the occurrence of the user-specified statistical values and their persistence over a specified length of time. This allows multiple alarms to be set without adding packet processing overhead, thus speeding program performance.

Alarms can be created for items in the **Node**, **Protocol**, and **Summary Statistics** windows. You can also create an alarm from items in the ***Node***, ***Protocol***, and ***Summary*** views of any Capture window or Packet File window, or from any open statistics **Graph** window.

Important! No matter where or how an alarm is created, it only watches Monitor statistics. This means the **Monitor Statistics** option under the **Monitor** menu must be enabled in order for alarms to work.

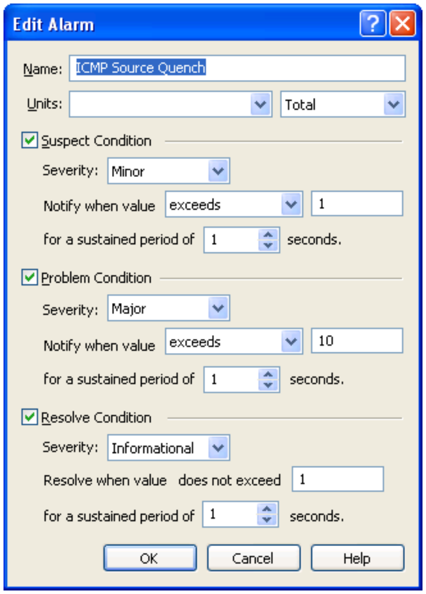
Predefined alarms

EtherPeek includes two sets of ready-made alarms for your convenience. The first set is loaded on installation. These are located in the 1033\Alarms directory where you installed EtherPeek, in a file called Default Alarms.alm. A second, larger set of alarms is included in a file in the same directory called Additional Alarms.alm. The default set of alarms covers the most frequently encountered network problem conditions. The additional alarms generally include normal network conditions which you may want to monitor for particular purposes. You can load these or any other saved set of alarms using the **Import** button in the **Alarms** window.

Creating and editing alarms

To create a new alarm:

1. Open one of the statistics windows, statistics views, or statistics graphs offering the Make Alarm function. Alarms can be created for items in the **Node**, **Protocol**, or **Summary Statistics** windows; or from items in the analogous views of any Capture window or Packet File window; or from any open statistics **Graph** window.
2. Select the item to be monitored.
3. Click the **Alarms** button at the top of the window, or right-click on the item and choose **Make Alarm…** from the context menu, to open the **Make Alarm** dialog (Figure 12.5).
4. Fill in the parameters for the alarm, following the usage shown in Table 12.1. Note that a single alarm can test for two distinct levels, identified in the **Make Alarm** dialog as *Suspect Condition* and *Problem Condition*. Both sets of conditions share the same *Resolve Condition*. This allows you to create a yellow alert / red alert / stand down for the same statistics parameter in a single alarm. Alternatively, you can specify only the *Suspect Condition* or only the *Problem Condition* for this alarm.
5. When you have chosen all the parameters, click **OK** to create and enable the alarm, or click **Cancel** to close the **Make Alarm** dialog without creating the alarm.



Edit Alarm

Name: ICMP Source Quench

Units: Total

☑ Suspect Condition

Severity: Minor

Notify when value exceeds 1

for a sustained period of 1 seconds.

☑ Problem Condition

Severity: Major

Notify when value exceeds 10

for a sustained period of 1 seconds.

☑ Resolve Condition

Severity: Informational

Resolve when value does not exceed 1

for a sustained period of 1 seconds.

Figure 12.5 Make Alarm dialog

The following table (Table 12.1) lists the user-definable elements in the **Make Alarm** dialog (and the identical **Edit Alarm** dialog), and describes their usage.

Table 12.1 Make Alarm and Edit Alarm dialog parameters

Parameter	Usage
Name	The name by which this alarm will be known in the **Alarms** window, and which will be used in the message portion of any notifications. The dialog is context aware. By default, any new alarm is named for the statistical item to be monitored. You may modify or add to this name.

Table 12.1 Make Alarm and Edit Alarm dialog parameters (Continued)

Parameter	Usage
Units	This two part entry sets the units in which the statistical value for which the alarm is testing will be measured. The dialog is context sensitive and the choices in the first drop-down list change according to the statistical parameter chosen. Typical choices are: *Count*, *Packets*, or *Bytes*. Alarms created in the **Node Statistics** window add the concept of direction, giving *Bytes From*, *Bytes To*, *Total Bytes*, and the same for packets. The second drop-down list (on the right) determines whether these units are to be counted *Per second* or in *Total* over the time periods set for each *Condition* below. In general, only alarms set to watch statistics which are themselves already measured in units per second should be set to *Total*. Alarms for all other statistics should be set to the default *Per second*.
Suspect Condition	Check this checkbox to specify the parameters for a *Suspect Condition* for the current statistics parameter. Suspect conditions are typically used to note less severe states.
Severity	Choose the severity of the notification to be sent when the Suspect conditions are met. For more about notifications, see “Notifications” on page 237.
Notify when value	Choose *exceeds* or *does not exceed* from the drop-down list and enter a value in the adjacent text entry box.
for a sustained period of … seconds	Enter a value in *seconds.*
Problem Condition	Check this checkbox to specify the parameters for a *Problem Condition* for the current statistics item. Problem conditions are typically used to note more severe states.
Severity	Choose the severity of the notification to be sent when the Suspect conditions are met. For more about notifications, see “Notifications” on page 237.

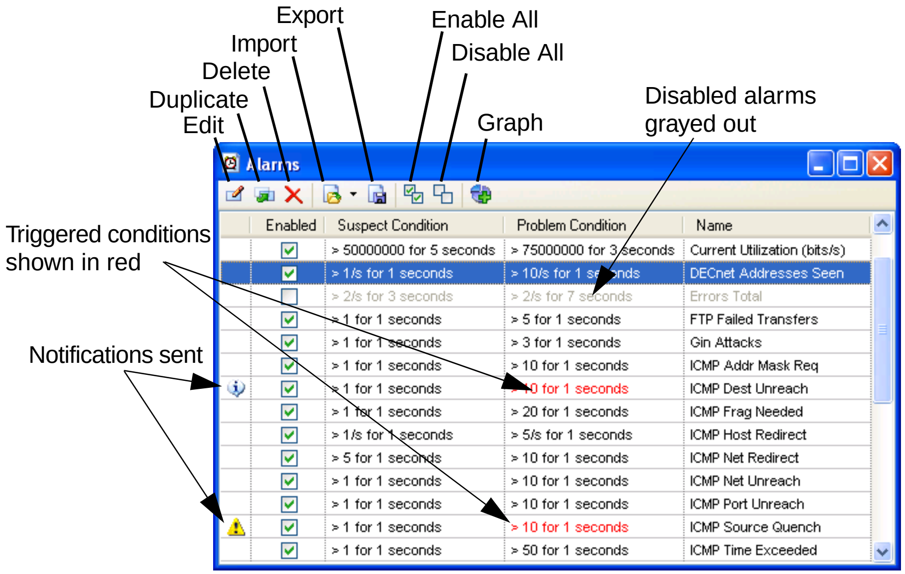
Table 12.1 Make Alarm and Edit Alarm dialog parameters (Continued)

Parameter	Usage
Notify when value	Choose *exceeds* or *does not exceed* from the drop-down list and enter a value in the adjacent text entry box.
for a sustained period of … seconds	Enter a value in *seconds.*
Resolve Condition	When these conditions are met, the alarm is “stood down” or resolved. The resolve condition is identical for either or both the *Suspect Condition* and *Problem Condition* in a given alarm.
Severity	Choose the severity of the notification to be sent when the resolve conditions are met. For more about notifications, see “Notifications” on page 237.
Resolve when value exceeds **/** *does not exceed*	The wording and sense of this resolve condition is automatically set to the opposite sense entered for the *Suspect Condition* and *Problem Condition* in a given alarm. Enter a value in the text entry box.
for a sustained period of … seconds	Enter a value in *seconds.*

Important! Alarms set to watch the *Total* value of a statistic which never goes down in value will not resolve until the statistics buffer is cleared (for example, when EtherPeek is restarted or when Monitor statistics are reset, either by a ***Statistics Output*** specification or manually by the user). Only a few statistics, such as *Average Utilization (kbits/s)* in **Summary Statistics** and some of the statistics captured by some Analysis Modules require the use of the *Total* feature. Most statistics require the default value of *Per second* when setting the conditions for any alarm.

The alarms window

When an alarm is first created, it is automatically enabled. To review the existing alarms, to enable or disable, duplicate, modify or delete them, or to create a real-time graph of the Monitor statistics parameters they are monitoring; open the **Alarms** window by choosing **Alarms** under the **View** menu.



	Enabled	Suspect Condition	Problem Condition	Name
	☑	> 50000000 for 5 seconds	> 75000000 for 3 seconds	Current Utilization (bits/s)
	☑	> 1/s for 1 seconds	> 10/s for 1 seconds	DECnet Addresses Seen
	□	> 2/s for 3 seconds	> 2/s for 7 seconds	Errors Total
	☑	> 1 for 1 seconds	> 5 for 1 seconds	FTP Failed Transfers
	☑	> 1 for 1 seconds	> 3 for 1 seconds	Gin Attacks
	☑	> 1 for 1 seconds	> 10 for 1 seconds	ICMP Addr Mask Req
ℹ	☑	> 1 for 1 seconds	> 10 for 1 seconds	ICMP Dest Unreach
	☑	> 1 for 1 seconds	> 20 for 1 seconds	ICMP Frag Needed
	☑	> 1/s for 1 seconds	> 5/s for 1 seconds	ICMP Host Redirect
	☑	> 5 for 1 seconds	> 10 for 1 seconds	ICMP Net Redirect
	☑	> 1 for 1 seconds	> 10 for 1 seconds	ICMP Net Unreach
	☑	> 1 for 1 seconds	> 10 for 1 seconds	ICMP Port Unreach
⚠	☑	> 1 for 1 seconds	> 10 for 1 seconds	ICMP Source Quench
	☑	> 1 for 1 seconds	> 50 for 1 seconds	ICMP Time Exceeded

Figure 12.6 Alarms window

The **Alarms** window (Figure 12.6) has five columns. The first or left-most column is unlabeled. From left to right, the remaining columns are: ***Enabled***, ***Suspect Condition***, ***Problem Condition***, and ***Name***.

The first column (unlabeled) displays the icon for the type of notification sent by any alarm that is in a triggered state.

The ***Enabled*** column shows a checkmark if the alarm is enabled. Check the checkbox in this column to enable or uncheck to disable individual alarms. When an alarm is disabled it is shown in grey.

The ***Suspect Condition*** and ***Problem Condition*** columns show a shorthand version of the statistics measurements required to trigger this part of the alarm. This value is set in the **Make Alarm** dialog and can be modified in the **Edit Alarm** dialog. Alarm conditions which have been triggered are shown in red.

The ***Name*** column shows the name of the alarm, which by default is the name of the statistic to be monitored. This value is set in the **Make Alarm** dialog and can be modified in the **Edit Alarm** dialog.

Double-click on any alarm to open the **Edit Alarm** dialog with all that alarm's properties shown and ready to edit. You can also open the **Edit Alarm** dialog by selecting an alarm from the list and clicking the **Edit** button at the left of the **Alarms** window. The **Edit Alarm** dialog is identical to the **Make Alarm** dialog in appearance and function.

To make a copy of an alarm, select the alarm in the **Alarms** window and click the **Duplicate** button. To delete an alarm, select the alarm in the **Alarms** window and click the **Delete** button. To create a graph showing the current values for the statistics being monitored by any alarm, select the alarm from the **Alarms** window and click the **Graph** button. Graphs created from an alarm will show a red line indicating the value set as the alarm's Problem Condition and an orange line for its Suspect Condition. You can **Enable All** or **Disable All** alarms at once by clicking those buttons. All of these functions are also available from a context menu by right-clicking on any alarm.

Importing and exporting alarms

You can save and reload the whole contents of the **Alarms** window, using the **Export** and **Import** buttons in the **Alarms** window. When you load an alarms file, you can choose whether to add to the existing list or replace it with the contents of the new file.

Note: If you re-install EtherPeek, neither the Default nor the Additional alarms will be loaded if the **Alarms** window already contains entries.

Notifications

Notifications are messages sent from triggers, alarms, Analysis Modules and other parts of the program to announce and describe the occurrence of specified events. Under the default settings, all notifications are sent using the same method or Action—they use an Action called *Log* to send their notifications to the Log file. This section describes how to use the ***Notifications*** view of the **Options** dialog to create other Actions and to associate these Actions with notifications of a particular severity. These Actions can be used either in addition to or instead of the standard Action of writing to the Log file.

Four types of Actions can be configured and associated with notifications in the ***Notifications*** view of the **Options** dialog. They are:

Log	Sends the notification to the Log File.
Email	Sends the notification in email.
Execute	Launches a program of your choice.

Sound	Plays a specified *.wav file on the local machine.

Individual sections following this introduction describe how to create each of these types of actions.

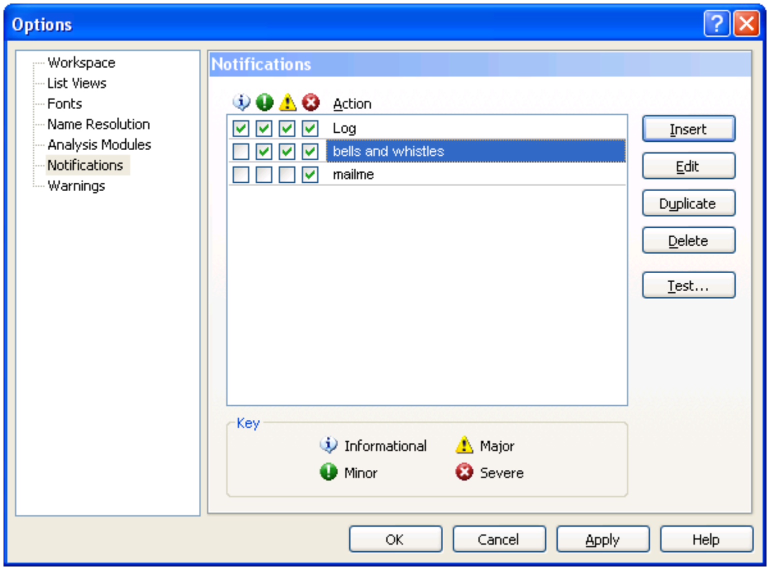
Notifications have an attribute called *level of severity*. Notifications can have one of four levels of severity. From least to greatest, they are:

- Informational
- Minor
- Major
- Severe

The level of severity is set by the function generating the notification. For triggers, alarms and some Analysis Modules the user can set the level of severity directly. Other Analysis Modules are coded to always assign a certain severity to notifications of a particular event. Analysis Modules can also be limited to a capped range of severities, overriding their internal coding. Please see these other sections (“Triggers” on page 224, “Alarms” on page 231, and “Analysis Modules” on page 247) for details about how each of these other functions generates notifications.

The ***Notifications*** view of the **Options** dialog controls how notifications of a given severity will be delivered, where they will be sent, and what (if any) other actions will be taken.

To open the ***Notifications*** view of the **Options** dialog, choose **Options…** from the **Tools** menu. Click the *Notifications* item in the navigation pane to bring up the ***Notifications*** view, shown in Figure 12.7.



ℹ	●	⚠	✖	Action
☑	☑	☑	☑	Log
☐	☑	☑	☑	bells and whistles
☐	☐	☐	☑	mailme

Key

ℹ Informational ⚠ Major

● Minor ✖ Severe

Figure 12.7 Notifications view of the Options dialog

The main pane of the ***Notifications*** view shows all the defined notification Actions, one Action per line. The name of each Action is shown in the column labeled ***Action***. The four left-most column headings are icons of the various levels of severity. Their meanings are shown in the *Key* at the bottom of the dialog. From left to right, the icons represent: ***Informational***, ***Minor***, ***Major***, and ***Severe***.

When a checkbox under one of these levels of severity is checked, the notification Action on that line will be invoked each time a notification of that severity is generated by any other function in the program. If the checkbox is unchecked, then a notification of that level of severity will not invoke the Action shown on that line.

When you first open the ***Notifications*** view of the **Options** dialog in EtherPeek, the only ***Action*** that is defined is called *Log* and the checkboxes under all four levels of severity are checked. This means that the *Log* action will be invoked when a notification of any of the four severity levels is generated from any source.

On the right hand side are five buttons used to maintain the notification actions. From top to bottom, they are as shown in Table 12.2 below.

Table 12.2 Notifications view buttons

Button	Description
Insert	Opens an **Edit Action** dialog with *Action* (the name of the Action) set to "*Untitled Action*" and the *Type* parameter set to the default *Log*. Select the type of action you want to create from the *Type* drop-down list. The **Edit Action** dialog view for that type of Action will appear, ready to be filled in.
Edit	Opens an **Edit Action** dialog for the selected Action, with all the information for that Action already filled in. (Double-clicking on an Action also opens the **Edit Action** dialog.)
Duplicate	Creates a copy of the selected Action.
Delete	Deletes the selected Action.
Test	Opens a dialog which allows you to edit the long and short messages of a sample notification, set the severity of the test notification, then test the notification settings for that severity level.

To create a new notification Action:

1. Choose **Options…** from the **Tools** menu to open the **Options** dialog.
2. Click the *Notifications* item in the navigation pane to open the ***Notifications*** view.
3. Click **Insert** to open the **Edit Action** dialog.
4. Use the *Type* drop-down list to choose the type of Action you wish to create. Your choices are *Log*, *Email*, *Execute*, or *Sound*.
5. Fill in the parameters for the Action. The following sections describe the parameters for each of the possible types of notification Actions.
   - Write the notification to the log file
   - Send the notification as email
   - Execute a program upon notification

- Play a sound file upon notification

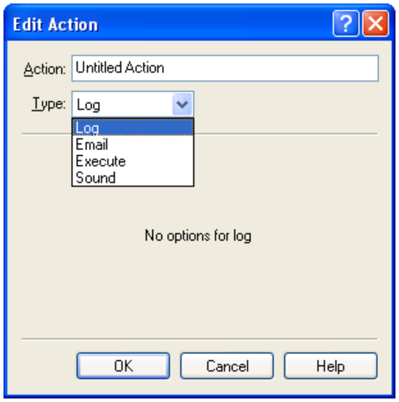
6. When you have filled in the parameters for the particular type of Action, click **OK** in the **Edit Action** dialog to close the dialog and return to the ***Notifications*** view, where your new Action will appear under the name you assigned.

7. Choose the levels of severity of notification for which this Action should be invoked. Check the checkbox under each level of severity that should use this Action.

8. Click **Apply** to implement your changes and leave the dialog open, or click **OK** to accept your changes and close the **Options** dialog.

Write the notification to the log file

The Log type action writes notifications to the EtherPeek Log. When the notification is generated by an event associated with a particular window, the Log type action will also write the notification to the ***Log*** view of that Capture window or Packet File window.



Action: Untitled Action

Type: Log

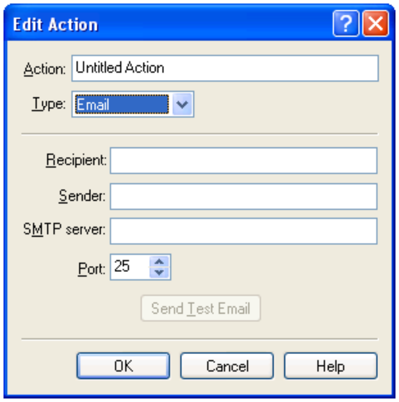
No options for log

Figure 12.8 Insert brings up the Edit Action dialog in default Log view

If you select the Action called *Log* and click the **Edit Action** button, it will bring up the **Edit Action** dialog in the correct view for this action. You will see that, as its name suggests, the Action is of the *Type Log*. Its name (in the box labeled *Action*) is *Log*, and there are, as the message says, *No options for log* type actions.

Send the notification as email

The Email type Action sends notifications as email messages with the text of the notification in the body of the email.



Action: Untitled Action
Type: Email
Recipient:
Sender:
SMTP server:
Port: 25

Figure 12.9 Edit Action dialog for Email action type

To create an Action of the type Email:

1. Click the **Insert** button. In the **Edit Action** dialog that appears, select *Email* from the *Type* drop-down list to switch to the ***Email*** view of the **Edit Action** dialog (Figure 12.9).
2. Fill in the options for the Email type action as shown in Table 12.3

Table 12.3 Options for Email type notification action

Option	Description
Recipient	Fill in the address to which you want the notifications to be sent.
Sender	Fill in the return address of the email message.
SMTP Server	Fill in the mail server on your network.
Port	The port on which the SMTP services are offered. The standard port for SMTP is port 25.

Tip You can use the *Sender* portion of the notification emails to sort the messages in the email program at the receiving end.

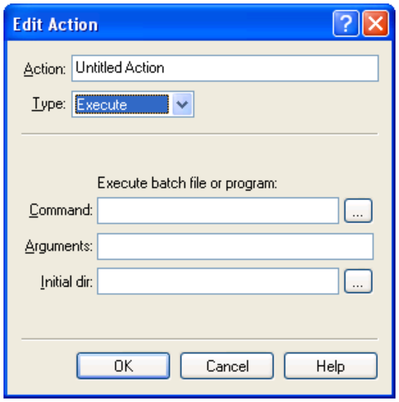
3. Optionally, you can test the email notification by clicking the **Send Test Email** button.
4. Give this Action a name (in the box labeled *Action*) and click **OK** to add it to the list of possible actions in the ***Notifications*** view.
5. Select which levels of severity of notification you would like to automatically perform this action, using the checkboxes to the left of the Action's name. Alternatively, you can leave all the checkboxes unchecked to simply hold this action in reserve without applying it at the moment to any Notifications.

Execute a program upon notification

You can run a program of your choice either instead of or in addition to any other notification actions.

To create an Action of the type Execute:

1. Click the **Insert** button. In the **Edit Action** dialog that appears, select *Execute* from the *Type* drop-down list to switch to the ***Execute*** view of the **Edit Action** dialog (Figure 12.10).



Edit Action

Action: Untitled Action

Type: Execute

Execute batch file or program:

Command:

Arguments:

Initial dir:

Figure 12.10 Edit Action dialog for Execute action type

2. Fill in the *Command* text entry box or click the button on the right marked with the ellipses **...** to browse your system to locate and select the program or batch file you wish to run when this Action is invoked.

3. Use the *Arguments* text entry box to specify any argument or command line parameters to use in invoking this program.

4. If the program requires an initial directory, you can specify this in the *Initial Dir* text entry box, or use the button marked with the ellipses **…** to browse your system to locate and select the initial directory.

5. Give this Action a name (in the box labeled *Action*) and click **OK** to add it to the list of possible actions in the ***Notifications*** view.

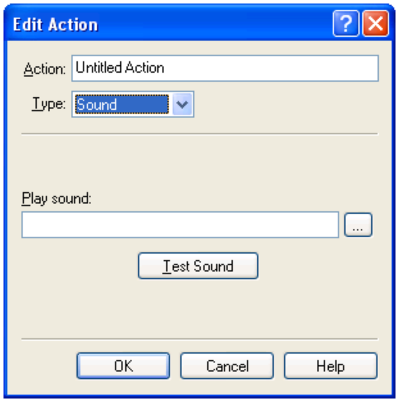
6. Select which levels of severity of notification you would like to automatically perform this action, using the checkboxes to the left of the Action's name. Alternatively, you can leave all the checkboxes unchecked to simply hold this action in reserve without applying it at the moment to any Notifications.

Play a sound file upon notification

You can play a sound of your choice in *.wav file format, either instead of or in addition to any other notification actions. The system on which EtherPeek is running must have the ability to play sound files in *.wav format in order to use this type of Action.

To create an Action of the type Sound:

1. Click the **Insert** button. In the **Edit Action** dialog that appears, select *Sound* from the *Type* drop-down list to switch to the ***Sound*** view of the **Edit Action** dialog (Figure 12.11).



Edit Action

Action: Untitled Action

Type: Sound

Play sound:

Figure 12.11 Edit Action dialog for Sound action type

2. Fill in the *Play sound* text entry box or click the button on the right marked with the ellipses **…** to browse your system to locate and select the *.wav file you wish to play when this Action is invoked.
3. Give this Action a name (in the box labeled *Action*) and click **OK** to add it to the list of possible actions in the ***Notifications*** view.
4. Select which levels of severity of notification you would like to automatically perform this action, using the checkboxes to the left of the Action's name. Alternatively, click no checkboxes to simply hold this action in reserve without applying it at the moment to any Notifications.

Analysis Modules

13

Analysis Modules are external modules that provide additional highly focused analysis features to the program. An Analysis Module tests network traffic and provides detailed summaries and counts of key parameters of one specific type of traffic, posting its results in the **Summary Statistics** window and/or in the ***Summary*** column of the ***Packets*** view of Capture windows and Packet File windows.

Enabled Analysis Modules are applied to traffic captured in real-time and to packets in the buffer of a Capture window or a Packet File window. You can enable and disable Analysis Modules individually. In addition, many Analysis Modules have user-configurable options, which can be used to further refine the data you collect about your network.

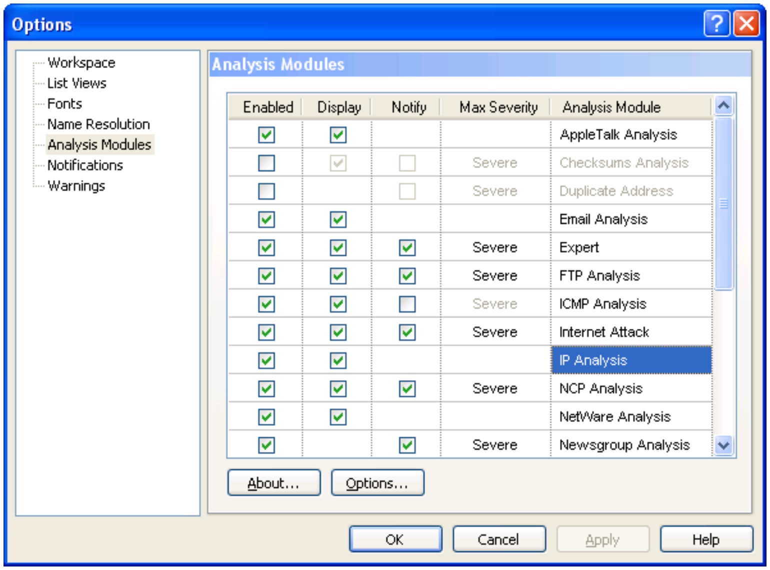
The Analysis Modules shipped with EtherPeek cover a wide range of the most common protocols and network applications. Users with some programming knowledge can use the accompanying SDK to write their own Analysis Modules, for example, to report on proprietary protocols or applications, or to present statistics of particular interest in their environment.

This chapter describes how to use Analysis Modules, and describes each of the Analysis Modules shipped with EtherPeek in detail.

In this Chapter:

Enabling and configuring analysis modules

To open the ***Analysis Modules*** view of the **Options** dialog, choose **Options…** from the **Tools** menu and click the *Analysis Modules* item in the navigation pane. The ***Analysis Modules*** view of the **Options** dialog shows a list of available Analysis Modules.



Enabled	Display	Notify	Max Severity	Analysis Module
☑	☑			AppleTalk Analysis
☐	☑	☐	Severe	Checksums Analysis
☐		☐	Severe	Duplicate Address
☑	☑			Email Analysis
☑	☑	☑	Severe	Expert
☑	☑	☑	Severe	FTP Analysis
☑	☑	☐	Severe	ICMP Analysis
☑	☑	☑	Severe	Internet Attack
☑	☑			IP Analysis
☑	☑	☑	Severe	NCP Analysis
☑	☑			NetWare Analysis
☑		☑	Severe	Newsgroup Analysis

Figure 13.1 Analysis Modules view of the Options dialog for EtherPeek NX

Important! Unlike triggers, capture buffers, and many other functions in EtherPeek, Analysis Modules are enabled and disabled *globally*. When an Analysis Module is enabled in the ***Analysis Modules*** view of the **Options** dialog, it is enabled simultaneously for any function in EtherPeek that could use the Analysis Module's added functionality. This includes Monitor statistics, Capture windows and Packet File windows. When any item is disabled in the ***Analysis Modules*** view of the **Options** dialog, it is unavailable to ALL parts of the program. The ***Performance*** view of either the **Monitor Options** or a **Capture Options** dialog can restrict the use of some or all Analysis Modules for a particular purpose, but it cannot enable an Analysis Module that has been disabled in the ***Analysis Modules*** view of the **Options** dialog.

Analysis Modules process packets each time the packets are loaded into a buffer. This means the same Analysis Module may process the same packet several times, but with the results posted to different places in EtherPeek, depending on which buffer is involved.

EtherPeek maintains one buffer for Monitor statistics, and separate buffers for individual Capture windows or Packet File windows.

The buffer for Monitor statistics is the simplest, in that it is either on or off. Any time **Monitor Statistics** is enabled and an adapter is selected for Monitor statistics, EtherPeek captures Monitor statistics, and does so continuously while the program is running. The buffer for Monitor statistics is not affected by filters or packet slicing. It is simply on or off. Any enabled Analysis Module will have the opportunity to process the packets in this buffer exactly once: when they first enter the buffer.

The buffers for individual Capture windows and Packet File windows are different. Any enabled Analysis Modules are applied to packets as they arrive in the Capture window buffer from the network, or as they are loaded into a Packet File window buffer from a file. Analysis Modules are also re-applied each time the contents of the buffer is changed in any of these windows by hiding or unhiding packets. Filters can restrict which packets are accepted into the buffer of a Capture window. Packet slicing, by capturing only a part of each packet, can limit the information available to Analysis Modules.

Enable/disable the analysis module

To enable or disable an Analysis Module, check or uncheck the left-most checkbox beside its name, in the column labeled ***Enabled***.

Analysis module info in packet list summary columns

To allow the Analysis Module to write details about the packet to the ***Summary*** column of any Capture window or Packet File window, check the checkbox in the column labeled ***Display***.

Enable/disable notification

Enable notifications by checking the checkbox in the column labeled ***Notify***. This tells the Analysis Module to send notifications when it detects certain events. For more on associating notifications with actions, see “Notifications” on page 237. Notifications can be set to perform one or more of the following types of actions:

Log	Sends the notification to the Log File.
Email	Sends the notification in Email.

Execute	Executes a program of your choice.
Sound	Plays a sound file in *.wav file format on the local machine.

Set maximum severity of notification

Each Analysis Module assigns its own level of severity to each type of event it is able to detect. It tries to assign that pre-determined severity to any notification of that event. The last column of the ***Analysis Modules*** view of the **Options** dialog, labeled ***Max severity*** allows you to set an upper limit for the severity of the notifications coming from each particular Analysis Module, regardless of the level of severity the Analysis Module itself might have assigned to some event. The four levels of severity, from least to greatest are: *Informational*, *Minor*, *Major*, and *Severe*. If you enable notification for an Analysis Module and set the maximum severity to *Minor*, then notifications coming from that Analysis Module will be capped at *Minor*. If the Analysis Module then tries to send notifications of *Severe*, *Major*, or *Minor* severity; they will *all* be treated as *Minor*. If the Analysis Module sends a notification with a severity of *Informational*, it will be treated as *Informational*.

This capability is important for keeping notifications to a manageable level when many Analysis Modules are enabled. It also provides essential flexibility in using notifications to launch a variety of actions. For instance, although many administrators might find it convenient to have a log of all Web URLs accessed in the course of a day, few would want to be paged each time a new URL or web page is seen on the network. They might, however, want to be paged in the event of a notice of *Severe* from the InternetAttack Analysis Module. Please see “Notifications” on page 237 for more detail on associating notification severity levels with the different types of actions available in EtherPeek.

Configuring options for an analysis module

Some Analysis Modules have configurable options. For example, the Duplicate Address Analysis Module allows you to suppress redundant reports and, through its options, to enter physical addresses that you would like to have ignored. When any of these Analysis Modules with user-configurable options is highlighted, the **Options…** button at the bottom of the ***Analysis Modules*** view of the **Options** dialog will no longer be greyed out. Click the **Options…** button to open the **Options** dialog for the selected Analysis Module.

Quick info on analysis modules

The **About…** button in the lower left of the ***Analysis Modules*** view of the **Options** dialog displays an About Box for the selected Analysis Module. For information on the capabilities of each Analysis Module, see “Analysis modules shipped with EtherPeek” on page 252.

Apply analysis module command

Normally, Analysis Modules are applied to packets as they arrive in the buffer from the network, or as they are loaded from a file. Analysis Modules are also re-applied each time the contents of the buffer is changed by hiding or unhiding packets.

There are circumstances where it is useful to be able to apply Analysis Modules to one or more packets that are already in the buffer without having to re-apply all Analysis Modules to all packets.

For example, if you have just enabled a particular Analysis Module and you want to see its results for a group of packets but do not want to re-apply all enabled Analysis Modules to all packets in the buffer, select the packets to which you would like to apply the new Analysis Module, right-click and choose that Analysis Module’s name from the **Apply Analysis Module** list in the context menu.

A second reason to use the **Apply Analysis Module** command has to do with the mechanics of how Analysis Modules operate with respect to the ***Summary*** column in the ***Packets*** view of a Capture window or a Packet File window. There is only room for information from a single Analysis Module in the ***Summary*** column. When multiple Analysis Modules are enabled, they are applied in order, and the first Analysis Module to write to the ***Summary*** column is the only one whose information actually appears there. For example, the Web Analysis Module is normally applied before the IP Analysis Module. If both are enabled, you would normally not see any IP Analysis information for any packet that showed information in the ***Summary*** column provided by the Web Analysis Module. To overcome this, you can use the **Apply Analysis Module** command.

To apply the IP Analysis Module to selected packets in a Packet List:

1. Select the packet(s) to which you would like to apply the IP Analysis Module.
2. Right-click, choose the **Apply Analysis Module** command from the context menu and select **IP Analysis** from the sub-menu.

3. This applies the IP Analysis Module to the selected packet(s) and allows the Analysis Module to write to the ***Summary*** column. Any other actions specified for this Analysis Module will also be taken, based on the results of processing the selected packets.
4. A message dialog appears showing the number of your selected packets which were processed by the Analysis Module you applied. If the dialog shows less than the whole number (for example *2 of 3 packets applied*), it means that the Analysis Module you applied did not find the information for which it was designed to test in some of the packets you selected.
5. Click **OK** to close the message dialog.

Note: The order in which Analysis Modules are applied for purposes of writing to the ***Summary*** column, depends on how far into the packet the Analysis Module must reach to find the information for which it is testing. The deeper into the packet the Analysis Module must reach (or the larger the offset of the data for which the Analysis Module is testing), the earlier the Analysis Module is applied. The closer to the beginning of the packet (or the lower the offset of) the data for which the Analysis Module is testing, the later it will be applied.

For information on using Analysis Modules to select captured packets, see “Select based on analysis modules” on page 294.

Analysis modules shipped with EtherPeek

Note: Registered users of EtherPeek are provided with a Software Development Kit (SDK) for Analysis Modules. Analysis Modules can be written by any user with some programming knowledge. If you are interested in writing your own Analysis Modules, you can find the Analysis Modules SDK in the 1033\Documents directory in the directory where you installed EtherPeek.

AppleTalk analysis module

The AppleTalk Analysis Module keeps track of and displays information about AARP requests, AARP responses, AARP probes, unanswered AARP requests, and the number of AppleTalk multicasts on your network. In addition, the AppleTalk Analysis Module shows details for NBP, ATP, and ASP. The AARP request shows AppleTalk address requested. The AARP response shows address and name. An ATP request shows transaction ID and Bitmap. An ATP response shows transaction ID and sequence number. ASP shows transaction ID, sequence number, and session ID. The results of the

AppleTalk Analysis Module are displayed in the ***Summary*** column of the ***Packets*** view of any Capture window or Packet File window, and its counts are also used as some of the key baseline traffic elements provided in the **Summary Statistics** window.

Checksums analysis module

Many network error detection and correction techniques are based on checksums. The sender performs a computation on the data to be sent and the result, the checksum, is included with the transmission. The receiver performs the same computation on the data it receives and compares its results to the sender's checksum. If a difference exists, the data is most likely corrupted, and the sender is asked to retransmit the data.

The Checksums Analysis Module verifies checksums and keeps track of the total number of invalid checksums for IP headers and data (including ICMP, IGMP, TCP, and UDP), and AppleTalk DDP data. Invalid checksums can be displayed in Capture and Packet File windows. This Analysis Module can send Notifications.

Conversations

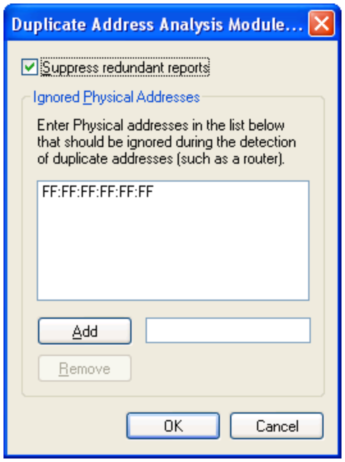
The *Conversations* which appears in the ***Analysis Modules*** view of the **Options** dialog in EtherPeek standard is the ***Conversations*** view of Capture windows and Packet File windows. While not an Analysis Module in the ordinary sense, the ***Conversations*** view makes use of the Analysis Modules architecture to allocate memory resources to the Conversations Analysis function and to allow users to selectively enable and disable ***Conversations*** view functionality. A similar memory allocation system is used for the Expert fuinctions in EtherPeek NX. For a detailed description of how to set the default memory allocation for the ***Conversations*** view, please see "Expert" on page 256.

For complete details about the use of the ***Conversations*** view in EtherPeek standard, please see "Conversations" on page 168.

Duplicate address analysis module

The Duplicate Address Analysis Module displays and logs instances when two or more network devices are using the same IP address. When two distinct and separate physical addresses are noted by the Duplicate Address Analysis Module to be using the same logical IP address, the Analysis Module produces a Notification. The Duplicate Address Analysis Module also adds a count of duplicate IP addresses detected to the **Summary**

Statistics window and the ***Summary*** column of the ***Packets*** view of any Capture window or Packet File window.



Duplicate Address Analysis Module...

☑ Suppress redundant reports

Ignored Physical Addresses

Enter Physical addresses in the list below that should be ignored during the detection of duplicate addresses (such as a router).

FF:FF:FF:FF:FF:FF

Figure 13.2 Duplicate Address Analysis Module Options dialog

To change options for this Analysis Module, select it in the ***Analysis Modules*** view of the **Options** dialog and click the **Options…** button. To suppress redundant reports, enter the physical addresses of devices that should be ignored. By default, duplicate reports for the physical hardware broadcast address are suppressed.

The Duplicate Address Analysis Module is disabled by default. For the most accurate results, you should use the Name Table to identify routers on the local segment before enabling the Duplicate Address Analysis Module.

Note: Duplicate IP address notifications are usually caused by multiple routers. Because routers forward traffic from other networks at OSI Layer 3, the logical address (IP) is forwarded unchanged but the physical address (MAC) is changed to that of the router doing the forwarding. When there is more than one router on the local segment, EtherPeek may see multiple physical addresses associated with a single logical address, and send a Duplicate Address notification accordingly. To prevent these notifications from being triggered by legitimate traffic from local routers, you have two choices. You can enter the physical address of each router in the **Duplicate Address Analysis Module Options** dialog *Ignored Physical Addresses* list and check the *Suppress redundant reports* checkbox. Alternatively, you can use the Name Table to identify each router as such by assigning it a *Node Type* of *Router* in the **Edit Name** dialog. Please see Chapter 7, “Name Table” on page 127 for details. The program checks the Name Table for nodes identified as routers before generating a duplicate address notification.

Email analysis module

The Email Analysis Module displays SMTP and POP3 commands that can be helpful in debugging Internet mail problems. The Email Analysis Module reports on client/server connections by counting the number of mail transfers initiated, the number of successful transfers, and the number of failed transfers. It then delivers this information to the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window, and to the **Summary Statistics** window.

SMTP specifies the exact format of messages a client on one machine uses to transfer mail to a server on another. Communication between a client and a server consists of readable ASCII text.

First, the client establishes a reliable stream connection to the server and then waits for the server to send a 220 READY FOR MAIL message. If the server is overloaded, it may delay sending the 220 message temporarily. Once the 220 message is received by the client, the client sends a HELO command.

The server responds by identifying itself. Once communication has been established, the sender can transmit one or more mail messages, terminate the connection, or request the server to exchange the roles of sender and receiver so messages can flow in the opposite direction. The receiver must acknowledge each message. It can also suspend the entire connection or the current message transfer.

Mail transactions begin with the MAIL command that provides the sender identification as well as a FROM: field that contains the address to which errors should be reported. A recipient prepares its data structures to receive a new mail message and replies to a MAIL command by sending the response 250, which means all is well. The full response consists of the text 250 OK. As with other application protocols, programs read the abbreviated commands and 3-digit numbers at the beginning of lines; the remaining text is intended to help debug mail software.

After a successful MAIL command, the sender issues a series of RCPT commands that identify recipients of the mail message. The receiver must acknowledge each RCPT command by sending 250 OK or by sending the error message 550 No Such User Here.

After all RCPT commands have been acknowledged, the sender issues a DATA command. In essence, a DATA command informs the receiver that the sender is ready to transfer a complete mail message. The receiver responds with message 354 Start Mail Input and specifies the sequence of characters used to terminate the mail message. The termination sequence consists of 5 characters: carriage return, line feed, period, carriage return, and line feed.

Although clients can suspend the delivery completely if an error occurs, most clients do not. Instead, they continue delivery to all valid recipients and then report problems to the sender.

Usually, the client reports errors using electronic mail. The error message contains a summary of the error as well as the header of the mail message that caused the problem.

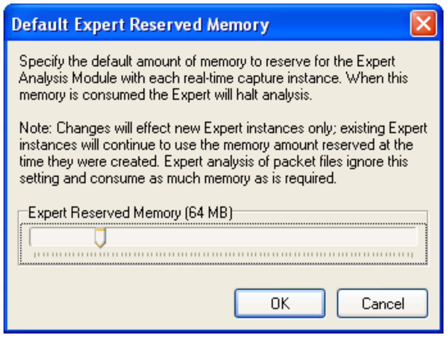
Once the client has finished sending all the mail messages to a particular destination, the client may issue the TURN command to turn the connection around. If it does, the server responds 250 OK and assumes control of the connection. With the roles reversed, the side that was originally the server sends back any waiting mail messages. Whichever side controls the interaction can choose to terminate the session by issuing a QUIT command. The other side responds with command 221, which means it agrees to terminate. Both sides then close the TCP connection.

Expert

The *Expert* which appears in the ***Analysis Modules*** view of the **Options** dialog in EtherPeek NX is the ***Expert*** view of Capture windows and Packet File windows. While not an Analysis Module in the ordinary sense, the ***Expert*** view makes use of the Analysis Modules architecture to allow users to allocate memory resources to the Expert and to selectively enable and disable part or all of the Expert Analysis functionality.

Findings from the Expert are displayed in the ***Expert*** column of the ***Packets*** view of any Capture window or Packet File window, and summaries of its findings are displayed in the **Summary Statistics** window. The Expert can send notifications.

From the main program menu, choose **Tools** > **Options…** to open the **Options** dialog, then click the *Analysis Modules* item in the navigation pane to open the ***Analysis Modules*** view.



Default Expert Reserved Memory

Specify the default amount of memory to reserve for the Expert Analysis Module with each real-time capture instance. When this memory is consumed the Expert will halt analysis.

Note: Changes will effect new Expert instances only; existing Expert instances will continue to use the memory amount reserved at the time they were created. Expert analysis of packet files ignore this setting and consume as much memory as is required.

Expert Reserved Memory (64 MB)

Figure 13.3 Default Expert Reserved Memory dialog

To set the amount of memory reserved for Expert functions in new Capture windows, select *Expert* in the ***Analysis Modules*** view of the **Options** dialog and click the **Options…** button. This opens the **Default Expert Reserved Memory** dialog. Use the slider bar to set the *Expert Reserved Memory* and click **OK** to accept your setting. The amount of memory you assign here will be reserved for Expert analysis functions in each separate Capture window you create. The total memory used will depend on how many Capture windows are open and performing Expert analysis. You cannot change the *Expert Reserved Memory* settings for an existing Capture window.

Expert memory is reserved for each Capture window individually. When its reserved memory is consumed, the Expert will begin to re-use the memory, dropping the oldest conversations first. (For details, see “Continuous Expert use of allocated memory” on page 117.) The greater the available memory, the greater the number of conversations that can be analyzed and kept in the Conversations pane. The default settings in the **Default Expert Reserved Memory** dialog are conservative and designed to accommodate multiple simultaneous captures. If you have only one capture, you can certainly increase the default memory allocation.

Note: Expert analysis in Packet File windows is not affected by the settings in the **Default Expert Reserved Memory** dialog. The Expert will consume as much memory as is required to analyze all the conversations present in any saved packet file.

For complete details about the use of Expert Analysis in EtherPeek NX, please see Chapter 5, “Expert View and Expert EventFinder” on page 101.

FTP analysis module

The FTP Analysis Module provides the ability to:

- Report the number of successful file transfer initiations, completions and failures.
- Report and display the names of files that are being uploaded or downloaded.
- Report and display ftp commands (for example, ls, cd, and so forth).

The FTP Analysis Module also watches FTP control traffic for status messages that signal the successful start and end of a file transfer. A count is then added to the **Summary Statistics** window for these values. The FTP Analysis Module can also write these control messages to the ***Summary*** column of the ***Packets*** view of Capture windows and Packet File windows.

FTP can send an unsuccessful termination message. This condition is rare, but can be of interest to a network manager, especially if there is a high incidence of terminated sessions. Normally, failed FTP transactions are due to unexpected network delays or disruptions. Because a status packet does not usually accompany termination, the only way for a network manager to be aware of this condition is by monitoring the difference between the successful start and end of file transfers. A high discrepancy can signal not only potential network problems, but also additional loss of bandwidth due to unsuccessful transfers.

ICMP analysis module

ICMP (Internet Control Message Protocol) is defined as a maintenance protocol that handles error messages to be sent when packets are discarded or when systems experience congestion. For instance, the classic TCP/IP test command is PING. It sends an ICMP Echo Request to a remote system. If the system responds, the link is operational. If it fails to respond to repeated pings, something is wrong.

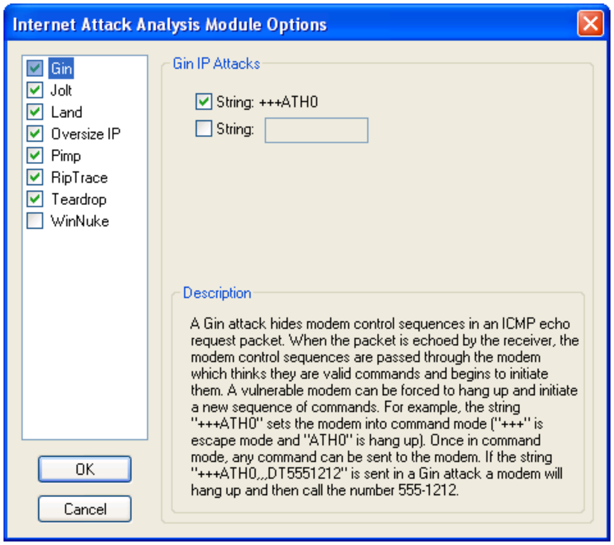
Another important function of ICMP is to provide a dynamic means to ensure that your system has an up-to-date routing table. ICMP is part of any TCP/IP implementation and is enabled automatically. ICMP messages provide many functions, including route redirection. If your workstation forwards a packet to a router, for example, and that router is aware of a shorter path to your destination, the router sends your workstation a redirection message informing it of a shorter route.

The ICMP Analysis Module keeps track of and displays information about ICMP destination unreachables, ICMP redirects, ICMP address mask replies, ICMP source quenches, and more. The Analysis Module can display ICMP type and code in the ***Summary*** column of the ***Packets*** view of any Capture window or Packet File window, as well as in the **Summary Statistics** window. This Analysis Module can send Notifications.

To change options for this Analysis Module, select it in the ***Analysis Modules*** view of the **Options** dialog and click the **Options** button. You can choose to log or to ignore ping (echo) packets because they are quite common. The default is to ignore echo packets, and the option is therefore unchecked.

InternetAttack analysis module

The InternetAttack Analysis Module collects eight common types of attacks and their variations into a single multi-view dialog. The **InternetAttack Analysis Module Options** dialog allows you to enable testing for all attacks, or to enable or disable individual parts of the Analysis Module. For flexibility of application, some attack Analysis Modules feature user-definable test parameters.


Internet Attack Analysis Module Options

☑ Gin
☑ Jolt
☑ Land
☑ Oversize IP
☑ Pimp
☑ RipTrace
☑ Teardrop
☐ WinNuke

Gin IP Attacks

☑ String: +++ATH0
☐ String:

Description

A Gin attack hides modem control sequences in an ICMP echo request packet. When the packet is echoed by the receiver, the modem control sequences are passed through the modem which thinks they are valid commands and begins to initiate them. A vulnerable modem can be forced to hang up and initiate a new sequence of commands. For example, the string "+++ATH0" sets the modem into command mode ("+++" is escape mode and "ATH0" is hang up). Once in command mode, any command can be sent to the modem. If the string "+++ATH0,,,DT5551212" is sent in a Gin attack a modem will hang up and then call the number 555-1212.

Figure 13.4 InternetAttack Analysis Module Options dialog, Gin IP Attacks view

The InternetAttack Analysis Module can write to the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window. It also adds a count of packets for each enabled type of attack to the **Summary Statistics** window. This Analysis Module can send Notifications.

Each type of attack covered by the Analysis Module is described below. Each section shows the protocol used, the date when the attack first appeared, and the types of systems reported as vulnerable to the attack. (These vulnerable systems are generally specified in the source code of the attack as tested by the author of each attack. These were not tested and verified by WildPackets.) For each type of attack the description shows the incidence of false positives—legitimate traffic which happens to match the test criteria. Each describes the working of the attack and its results, then lists the packet characteristics and contents which will generate a positive match. These criteria are listed in the order in which they will be tested. All of the listed test criteria must be met, for the Analysis Module to respond with a notification of an attack of this type.

To change the options for the InternetAttack Analysis Module:

1. Select it in the ***Analysis Modules*** view of the **Options** dialog and click the **Options** button to open the **InternetAttack Analysis Module Options** dialog (Figure 13.4).
2. Enable or disable testing for each individual attack type, by checking or unchecking the checkbox next to the name of each.
3. Highlight the name of any individual attack to bring up all user-definable parameters for that test, along with a brief description of the type of attack.
4. Make any changes to the parameters for individual attack tests. Please see the description of the attack in the following section for details of user-definable parameters.
5. Click **OK** to accept your changes and close the **InternetAttack Analysis Module Options** dialog.

Gin IP attacks

Protocol: ICMP (Internet Control Message Protocol) **Date**: June 6, 1999

Vulnerable system configurations:

Systems on which all of the following are true:

- The system does not filter ICMP echo request (Ping) packets.
- The system knows how to reply to ICMP echo request (Ping) packets.
- The system is using a modem.
- The modem's guard time is set extremely low.

False Positives: None, for default character string. Modem control sequences are not a legitimate part of ICMP packets.

Description: A Gin attack hides modem control sequences in an ICMP echo request packet. When the packet is echoed by the receiver, the modem control sequences are passed through the modem which thinks they are valid commands and begins to act on them. A vulnerable modem can be forced to hang up and initiate a new sequence of commands. Once in command mode, any command can be sent to the modem, including instruction to dial any number.

Results: Attacker control of the defender's modem.

Analysis Module tests for:

- ICMP packet
- ICMP Echo Request
- The character string "+++ATH0" (user definable)

Jolt IP attacks

Protocol: ICMP (Internet Control Message Protocol) **Date**: 1997

Vulnerable system configurations:

- Windows 95
- Old versions of Mac OS

False Positives: Very Rare. Any fragmented ICMP packet with the specific values of Identifier = 4321 and Fragmentation Offset = 45216 bytes (or the values defined by the user) will be marked as a Jolt attack. False positives are possible but unlikely, since only 1 in 536,870,912 fragmented ICMP packets will randomly have these values.

Description: A Jolt attack sends a large number of spoofed, fragmented, oversized ICMP packets. To deal with possible future variations on this attack, two key parameters are user definable.

Results: System freeze.

Analysis Module tests for:

- ICMP packet
- Fragmentation flag = 1
- Identifier = 4321 (user definable)

- Fragmentation Offset = 45216 bytes (user definable)

Land TCP attacks

Protocol: TCP/IP **Date**: November 20, 1997

Vulnerable system configurations:

- BSDI 2.1 (vanilla)
- FreeBSD 2.2.2-RELEASE
- FreeBSD 2.2.5-RELEASE
- FreeBSD 2.2.5-STABLE
- FreeBSD 3.0-CURRENT
- HP-UX 10.20
- MacOS 8.0 (TCP/IP stack crashed)
- NetBSD 1.2
- NeXTSTEP 3.0
- NeXTSTEp 3.1
- OpenBSD 2.1
- Solaris 2.5.1 (conflicting reports)
- SunOS 4.1.4
- Windows 95 (vanilla)
- Windows 95 + Winsock 2 + VIPUPD.EXE

Not vulnerable:

- BSDI 2.1 (K210-021,K210-022,K210-024)
- BSDI 3.0
- Digital UNIX 4.0
- IRIX 6.2
- Linux 2.0.30
- Linux 2.0.32
- Novell 4.11
- OpenBSD 2.2 (Oct31)
- SCO OpenServer 5.0.4

False Positives: Rare. TCP/ IP packets should not have their source and destination addresses set to the same value. Packets detected by this Analysis Module may not be Land attacks, but they are still improperly formed.

Description: A Land attack is a flood of packets with the Synchronize (SYN) flag set and the source IP Address and Port Number spoofed to be the same as the destination. Vulnerable systems can neither resolve these circular synchronize requests nor discard them quickly enough to avoid overload. When a large number of TCP open requests are left in the SYN state, TCP networking locks up on affected systems. UDP, including ICMP, continues to work, however.

There are three variations of the Land attack: Land, Blat, and LaTierra. In addition to setting the SYN flag, Blat also sets the Urgent (URG) flag and LaTierra also sets the Push (PSH) flag.

Results: TCP networking locks up.

Analysis Module tests for:

Land:

- TCP/IP packet
- Source and Destination IP Addresses are the same
- Source and Destination Ports are the same
- Synchronize flag (SYN) = 1

Blat (same as LAND, plus):

- Urgent flag (URG) = 1

LaTierra (same as LAND, plus):

- Push flag (PSH) = 1

Oversize IP attacks

Protocol: IP **Date**: January, 1997

Vulnerable system configurations:

- Older (pre-1998) operating systems
- Older (pre-1998) TCP/IP stacks

False Positives: None. The existence of oversized packets may not constitute an attack, but it is always an error.

Description: An oversize IP packet occurs when a packet's IP data size + Fragmentation Offset is greater than 65535. When attempting to reassemble such packets, some operating systems and TCP/IP stacks crash.

The maximum size of an IP packet is (2^16)-1 octets, or 65535 bytes. Because many network systems cannot accept packets this large (Ethernet, for example, sets a maximum packet size of 1500 bytes), IP allows packets to be fragmented and reassembled at the receiving end. Each fragment is assigned an offset to define its place in the original packet. The offset of the first fragment is 0, the offset of the second fragment is the length (in bytes) of the first fragment, and so on. It is possible to create a fragment which is itself of a normal size, but which has an offset such that the size of the rogue packet plus its offset is greater than 65535 bytes. Many older implementations of TCP/IP do not attempt to reassemble packets until all the fragments have been received. When these systems attempt to reassemble an oversized packet, processing overflows occur which freeze or otherwise derange the system.

Early Microsoft implementations of the Ping (ICMP echo request) tool were likely to generate such illegally large IP packets. Other versions of these tools could easily be modified to do so deliberately. Although oversize IP packets and the attacks built around them are certainly not limited to Ping, these events from around 1996 and 1997 gave the name “Ping of Death” to this type of attack.

Results: Varies by system: system crash, system freeze, reboot, etc.

Analysis Module tests for:

- IP packet
- IP data size + Fragmentation Offset > 65535

Pimp IP attacks

Protocol: IGMP (Internet Group Management Protocol) **Date**: June 4, 1999

Vulnerable system configurations:

- Windows 95

False Positives: Rare. Any fragmented IGMP packet with the specific values of IP Address equal to 96.37.250.127, Identifier equal to 17664, and Fragmentation Offset equal to 7400 bytes (or the values defined by the user) will be marked as a Pimp attack. False positives are possible but unlikely, since only 1 in 2.30 x10^18 fragmented IGMP packets will randomly have these values.

Description: A Pimp attack sends a large number of spoofed, fragmented, oversized, IGMP packets. The default values represent the version of this attack seen today, but several key values are user-definable to enable the Analysis Module to be modified to address possible future variations on this attack.

Results: System crash

Analysis Module tests for:

- IGMP packet
- Identifier = 17664 (user definable)
- Fragmentation Offset = 7400 bytes (user definable)
- IP Address = 96.37.250.127 (user definable)

RipTrace IP attacks

Protocol: UDP (User Datagram Protocol) **Date**: 1997

Vulnerable system configurations:

- Linux 2.0.x
- RedHat - Routed checks if RIP packet comes from a valid router. Can always spoof the router's IP.

Not vulnerable:

- Solaris 2.6 - ignores the packet and returns the following error:
  ```
  in.routed[6580]: trace command from 1.2.3.4 - ignored
  ```

False Positives: None, for default character string.

Description: A RipTrace attack is a special RIP (Router Information Protocol) packet that commands routed (the UNIX routing daemon) to be in debug mode. Once in this mode, routed can be commanded to append to any file on the file system. By default, the Analysis Module tests for the “/” character used to begin a UNIX file path. To accommodate other operating systems and environments, this value is user definable.

Results: Any file on the attacked system can be appended to. Extremely dangerous!

Analysis Module tests for:

- UDP
- Source port = 520 (RIP)
- Destination port = 520 (RIP)

- Trace mode is on
- The character string “/” (the beginning of a file path) (user definable)

Teardrop IP attacks

Protocol: UDP (User Datagram Protocol) **Date**: March 11, 1997

Vulnerable system configurations:

- Windows 95
- Windows NT 4.0 w/ Service Pack 3
- Linux (1.x - 2.x, including the development kernels)

False Positives: Rare. A fragmentation offset less than 6 is valid but extremely unlikely. A fragmentation offset of 6 would mean the packet passed over a network segment with a *maximum* frame size of 42 bytes. This is less than Ethernet's *minimum* frame size of 64 bytes.

Description: A Teardrop attack sends two fragmented packets designed such that the fragmentation offset plus the UDP data size of the second packet is less than the size of the first packet. Thus, the end of the second packet is inside the first packet.

The attack is successful on systems that reassemble packet fragments without carefully checking the end points. These systems blindly subtract the second endpoint from the first, which, in this attack, results in a negative number. The computer considers the negative number unsigned, which means it is actually so large that it overflows the memory buffer set aside for packet fragment reassembly.

Results: System crash.

There are several variations of the Teardrop attack:

Analysis Module tests for:

- Teardrop: UPD packet, UDP Length =48, Fragmentation Offset = between 0 and 6
- Newtear: UPD packet, Fragmentation Offset = between 0 and 6
- SSPing: UPD packet, Fragmentation Offset = 1, Type of Service (ToS) =%00000000 (Precedence: Routine, Normal Delay, Normal Throughput, Normal Reliability)
- Flushot: UPD packet, Fragmentation Offset = 1, Type of Service (ToS) =%00000010 (Maximum Reliability)

- Nestea: UPD packet, Fragmentation Offset = 6
- Bonk: UPD packet, Port = 53, Fragmentation Offset = 4
- Boink: UPD packet, Fragmentation Offset = 4

WinNuke TCP attacks

Protocol: TCP/IP **Date**: May 7, 1997

Vulnerable system configurations:

- Windows 95 without Windows Sockets Version 3
- Windows NT 3.51 without Service Pack 5 and the “oob-fix” hot fix
- Windows NT 4.0 without Service Pack 3 and the “teardrop2-fix” hot fix

False Positives: Likely. The use of Out of Band data is valid, and the TCP protocol provides for this with the Urgent flag. Such packets are a normal if not frequent part of network traffic. If no vulnerable machines are on the network, the Analysis Module can and probably should be disabled.

Description: A WinNuke attack sends a few bogus TCP/IP packets followed by one with the Urgent (URG) flag set. Windows networking did not handle URG flags and either lost connection to the network or crashed the whole system.

The Urgent flag, along with the Urgent Pointer, are the TCP mechanism for sending “Out of Band” (OOB) data which provides a way for a packet to hop the queue and be immediately processed. This is a useful way of allowing an interrupt signal to stop the processing of network data, or for control commands to be sent to an application while its buffers are full.

A WinNuke attack must be sent to an open port on the defender's computer. This is usually port 139 (NetBIOS) but can be any open port. Other commonly attacked ports are 113 (Ident) and 135 (Epmap). WinNuke is also referred to as WinBlow, which is a version of WinNuke written to run on Windows, to attack other Windows OS machines.

Results: Lost network connection or system crash.

Analysis Module tests for:

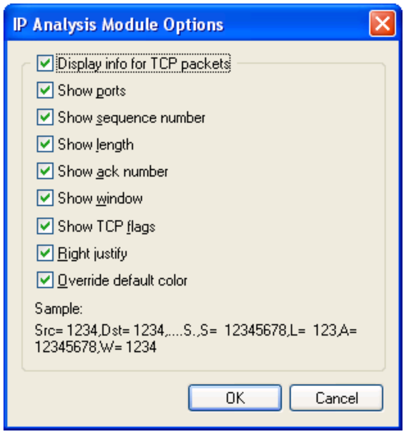
- TCP/IP packet
- Urgent flag (URG) = 1

Note: Because of the relatively higher chance of false positives, the WinNuke is disabled by default.

IP analysis module

The IP Analysis Module keeps track of and displays information about requests and responses from ARP, RARP, DHCP, and DNS; and TCP sequence numbers, acknowledgement numbers, windows, and flags, as well as TCP and UDP port numbers.

Address Resolution Protocol (ARP) dynamically discovers the physical address of a device, given its IP address. Reverse Address Resolution Protocol (RARP) enables a device to discover its IP address by broadcasting a request on the network. Dynamic Host Configuration Protocol (DHCP) provides clients with a dynamically assigned IP address and other network configuration setting parameters. Domain Name System (DNS) is a set of distributed databases providing information such as the IP addresses corresponding to network device names, and the location of mail servers.



IP Analysis Module Options

☑ Display info for TCP packets
☑ Show ports
☑ Show sequence number
☑ Show length
☑ Show ack number
☑ Show window
☑ Show TCP flags
☑ Right justify
☑ Override default color

Sample:
Src= 1234,Dst= 1234,....S.,S= 12345678,L= 123,A= 12345678,W= 1234

Figure 13.5 IP Analysis Module Options dialog

A Sequence number is a 32-bit field of a TCP header. If the segment contains data, the Sequence number is associated with the first octet of the data. TCP requires that data is acknowledged (given an Acknowledgement number) before it is considered to have been transmitted safely. TCP maintains its connections within a series of TCP windows established by the protocol. TCP packets may contain flags to denote a variety of conditions or protocol functions.

Results of the IP Analysis Module are displayed in the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window, and its counts are used as some of the key baseline traffic elements provided in the **Summary Statistics** window.

Options for this Analysis Module, all of which are enabled by default, are to show: ports, sequence number, length, ack number, window and TCP flags. Also enabled by default are the display options of *Right justify*, which makes the numbers line up correctly when seen in the ***Packets*** view, and *Override default color*, which shows information from this Analysis Module in grey in the ***Summary*** column of the ***Packets*** view.

NCP analysis module

The NCP Analysis Module collects request commands and response completion codes found in NCP (Netware Core Protocol) headers and posts this information to the ***Summary*** column of the ***Packets*** view of Capture windows and Packet File windows. NCP defines a set of request and reply packets used in support of file and print services, originally over IPX, but now also over IP.

NetWare analysis module

The NetWare Analysis Module provides information on unanswered RIP, SAP, and NCP requests to the **Summary Statistics** window and displays hop and tick counts for RIP packets, Sequence and Acknowledgement numbers for SPX, function and return codes for NCP packets, and service names for SAP packets in the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window.

Newsgroup analysis module

The Newsgroup Analysis Module displays and logs accesses to newsgroups and provides these counts to **Summary Statistics** and the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window. Anytime a newsgroup is accessed over the network by way of NNTP, the Analysis Module will generate a Notification noting the specific newsgroup name and the date and time of the access event.

Peer map

The *Peer Map* which appears in the ***Analysis Modules*** view of the **Options** dialog in EtherPeek NX is the ***Peer Map*** view of Capture windows and Packet File windows. While not an Analysis Module in the ordinary sense, the ***Peer Map*** view makes use of the Analysis Modules architecture to allow users to selectively enable and disable Peer Map functionality.

For complete details about the use of the ***Peer Map*** view in EtherPeek NX, please see Chapter 6, “Peer Map” on page 119.

RADIUS analysis module

The RADIUS Analysis Module provides statistics and decode summaries for Remote Access Dial-up User Services (RADIUS) and RADIUS accounting packets, including summaries for Access Request, Accept, and Reject packets; Accounting Request and Response packets; Access Challenge; and RADIUS Start and Stop packets. The Analysis Module provides this information to **Summary Statistics** and the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window.

SCTP analysis module

The SCTP Analysis Module collects information on the chunk type found in SCTP (Stream Control Transmission Protocol) headers and posts this information to the ***Summary*** column of the ***Packets*** view of Capture windows and Packet File windows.

SCTP (rfc 2960) provides reliable simultaneous transmission of multiple data streams between two nodes on an IP network. Either or both of the end points may be multi-homed. The original purpose of SCTP was to make IP networks capable of establishing the types of connections required for telephone service. Telephone service relies on SS7 (Signalling System 7), which sends signalling information (that is, information about the connection) along with the voice or other data at the same time. Sometimes referred to as next generation TCP (TCPng), SCTP was designed for broad application, and is not limited to telephone service over IP.

SMB analysis module

The SMB Analysis Module tracks many of the most common commands, status messages, and other responses for the Server Message Block protocol. It displays information about these SMB transactions in the ***Summary*** column of the ***Packets*** view of any Capture window or Packet File window. SMB is essentially an extended and enhanced file management protocol. Conceptually, the protocol treats files, printers, and named pipes as file objects which can be opened, closed, and modified.

Check the checkbox in the **SMB Analysis Module Options** dialog to *Show SMB command descriptions* in the ***Summary*** column in the ***Packets*** view and in the **Summary Statistics** window.

SQL analysis module

The SQL Analysis Module provides decode summaries for TNS and TDS traffic. Structured Query Language (SQL) is a widely used standard for querying databases. When using SQL over a network, the queries and data are carried within special protocols, where the type of protocol used depends on the type of database environment. Oracle environments use Transparent Network Substrate (TNS). Sybase and Microsoft SQL Server environments use the Tabular Data Stream protocol (TDS).

The module provides TDS descriptions including Login, RPC, and SQL summary strings. For TNS, the module provides decode summaries for TNS Connect, Accept, Refuse, Redirect, Data, Abort, Resend, Marker, and Control packets. The Analysis Module provides this information to the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window.

Telnet analysis module

The Telnet Analysis Module displays the contents of telnet sessions in the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window.

Telnet is a TCP/IP protocol that enables a terminal attached to one host to log in to other hosts and interact with their resident applications.

VoIP analysis module

The VoIP Analysis Module provides detailed information on traffic related to Voice over IP (VoIP). Specifically, the module provides statistics and decode summaries for MGCP, SIP, RTCP, G.723, H.323, H.225, G.711 traffic. The VoIP Analysis Module also follows H.245 connections based on H.323 port/IP connection data to provide statistics and decode summaries. The Analysis Module provides this information to **Summary Statistics** and the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window.

Web analysis module

The Web Analysis Module displays and logs access to World Wide Web resources. Anytime a Web URL is accessed over the network, the specific website location can be logged in the log file, noting date and time, and an email can be sent to inform the network manager of the access event (all by way of Notifications). The results can also be displayed in the ***Summary*** column in the ***Packets*** view of any Capture window or Packet File window.

The Web Analysis Module also adds a count of URLs accessed in the **Summary Statistics** window.

Tip Double-click on any URL posted to the Log file by the Web Analysis Module to open that resource in your default browser.

Note: In environments with significant Web traffic, the Web Analysis Module can write substantial amounts of information to the EtherPeek Log. You may want to disable the Web Analysis Module in such cases to prevent the Log file from growing too large, too quickly.

RMONGrabber

RMONGrabber is a separately purchased add-on feature. RMONGrabber lets you connect to an RMON probe on a remote network segment, collect (and optionally filter and/or slice) packets there, then view and analyze the packets in a local Capture window using all the EtherPeek tools. RMONGrabber ships with its own documentation. This chapter only describes the program features briefly.

To purchase a copy of RMONGrabber, or to find more information about this and other Analysis Modules, please see our website at http://www.wildpackets.com.

In this Chapter:

RMONGrabber Overview

The RMONGrabber Module is a separately purchased product which takes advantage of the Analysis Modules architecture to add new capabilities to EtherPeek. RMONGrabber follows the SNMP and RMON 1 standards to interact with remote probes. These standards allow RMONGrabber to set capture buffer and filter options on the remote probe, as well as to control the flow of packets back to EtherPeek. Standards compliance allows RMONGrabber to work with any RMON-compliant probe.

Note: RMONGrabber 1.0 supports RMON1, but not the RMON2 spec.

How RMONGrabber works

The RMONGrabber Module extends the troubleshooting capabilities of EtherPeek to remote segments of the network. Network traffic is captured from a remote probe and displayed instantly within an EtherPeek Capture window.

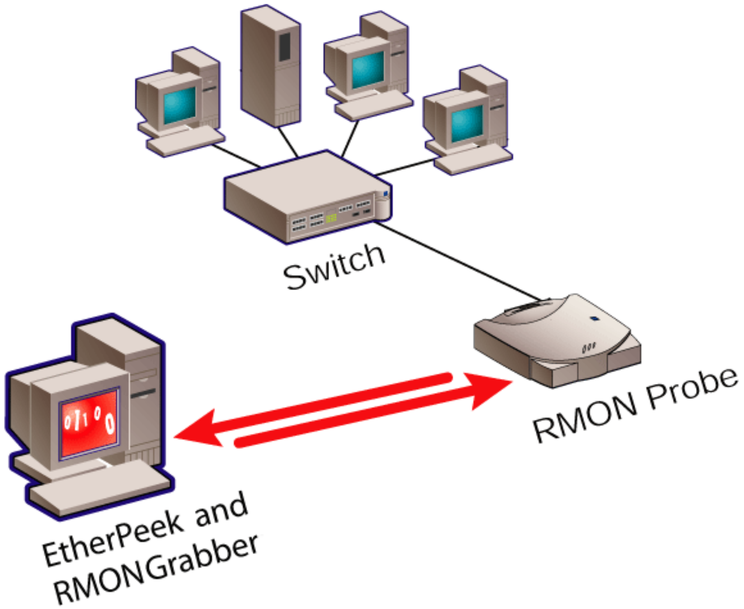


Figure 14.1 RMONGrabber collects network data from an RMON probe

System requirements and installation

Important! Please check our product pages at http://www.wildpackets.com/products for information on the latest version of RMONGrabber compatible with a particular version of EtherPeek.

RMONGrabber requires EtherPeek, and will run on any system meeting the system requirements for that program. EtherPeek should be installed before RMONGrabber, in order to create the directory structure RMONGrabber expects to find on the target system. In addition, remote capture requires a supported, standards-compliant RMON probe. RMONGrabber has been tested with Cisco, NetScout, 3Com, and Network Instruments RMON probes.

On installation, RMONGrabber creates an RMONGrabber folder in the directory where EtherPeek is installed, and places the RMONGrabber.dll in the EtherPeek Plugins folder, also located in the main EtherPeek directory. When installation is complete, you will be given the option of starting EtherPeek.

Important! Please keep your RMONGrabber serial number in a safe place. You will need it to reinstall RMONGrabber (for example, when you upgrade to a new version of EtherPeek).

Probe licenses

RMONGrabber is licensed under a variety of arrangements, some of which specify the number of probes which can be connected. For example, a 5 probe license will allow up to 5 probes to be added. There is no limit on the number of simultaneous connections that can be made to these 5 probes. When adding a probe to RMONGrabber with a 5 probe license, a dialog shows the number of licenses remaining.

Unlimited licenses place no limit on the number of probes that can be added. When installed under such a license, RMONGrabber does not display information about licenses remaining when a new probe is added.

RMONGrabber as an analysis module

RMONGrabber uses the Analysis Modules architecture to interact with EtherPeek. When you install RMONGrabber, it will appear in the EtherPeek ***Analysis Modules*** view of the **Options** dialog. You can enable or disable the module as a whole from within EtherPeek using this view. In EtherPeek, choose **Options…** from the **Tools** menu to open the **Options** dialog, then click the *Analysis Modules* item in the navigation pane to open the ***Analysis Modules*** view. To enable or disable RMONGrabber, check or uncheck the left-most checkbox beside its name, in the column labeled ***Enabled***. Click **OK** to exit the dialog, accepting your changes.

Using RMONGrabber

When you install RMONGrabber, it becomes an integral part of EtherPeek and appears in its own ***RMONGrabber*** view in every Capture window. RMONGrabber allows EtherPeek to acquire packets from RMON probes on remote network segments and capture them directly into local Capture windows, where you can use all the EtherPeek tools for filtering, decoding, selection, analysis, and more.

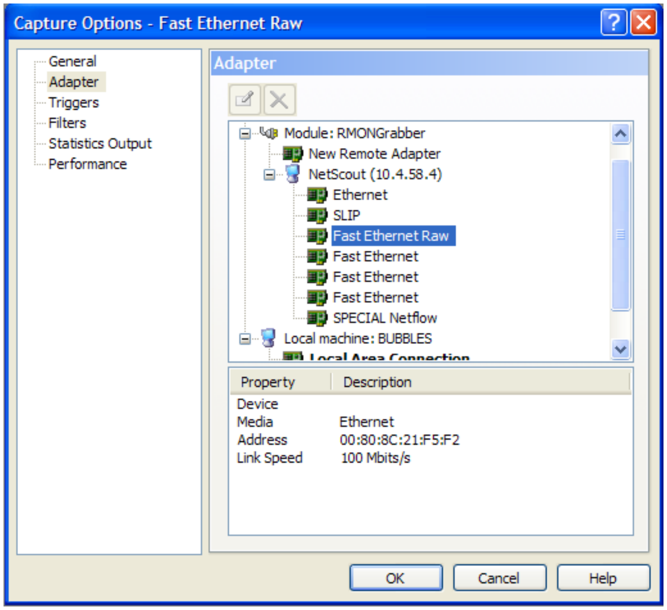
This section explains how to use RMONGrabber with EtherPeek to connect to an RMON probe, set capture options and filters for that probe, and collect packets from remote network segments.

This section covers only the functions unique to RMONGrabber, or those which are changed when RMONGrabber is installed with EtherPeek.

Connecting to an RMON probe

A Capture window in EtherPeek must, at a minimum, have its own capture buffer options and a valid adapter selected for its use. Other parts of this manual describe the options for creating a Capture window and setting its use of capture buffers, filters, and other functions in detail.

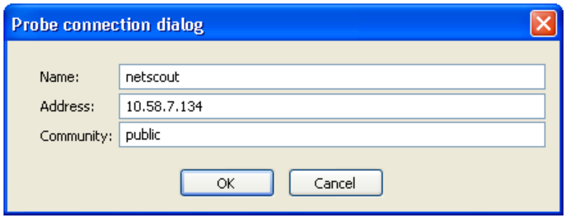
When preparing to capture using RMONGrabber, you must first create a new Capture window. Set the window’s capture buffer and packet slicing options in the ***General*** view of the **Capture Options** dialog. These should match your expectations of volume and timing for the remote capture.



Property	Description
Device	
Media	Ethernet
Address	00:80:8C:21:F5:F2
Link Speed	100 Mbits/s

Figure 14.2 RMONGrabber module in Adapters view of Capture Options dialog

Select the *Adapter* item in the navigation pane to open the ***Adapter*** view of the **Capture Options** dialog. Double-click on the *New Remote Adapter* item under *Module: RMONGrabber*. This will bring up the **Probe connection dialog**.



Name: netscout
Address: 10.58.7.134
Community: public

Figure 14.3 Probe connection dialog

Enter the *Name*, the IP *Address*, and SNMP *Community* name of the probe from which you want to capture. Note that the SNMP *Community* name (for example, *public*) must match what the probe expects, in order for the probe to permit a connection. Click **OK**. RMONGrabber attempts to connect to the probe using the information you supplied. If the connection is made, the probe will appear in the ***Adapter*** view. Select the newly listed adapter and click **OK** to accept your selection and close the **Capture Options** dialog.

The label on the **Start Capture** button at the upper right of the Capture window now appears as **Start Remote Capture**. When you click this button, the remote capture session will begin.

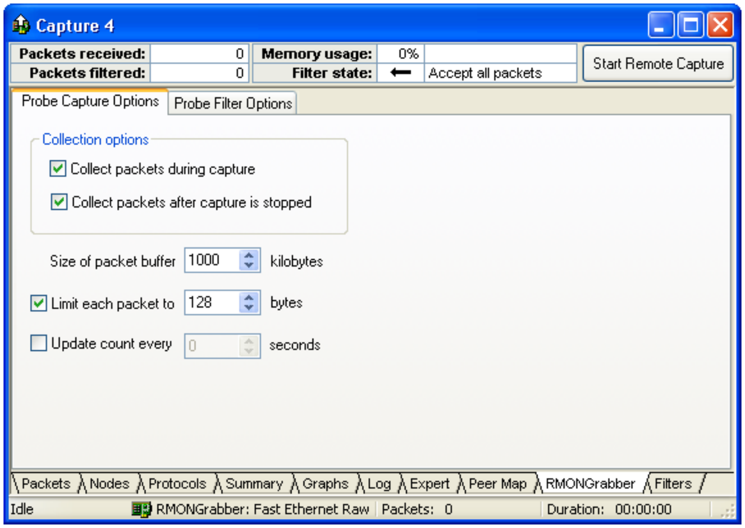
Note: You must set all RMONGrabber options before you begin the remote capture. Unlike capture from a local adapter, none of the options for remote capture can be changed during capture. Instead you will need to stop capture, make changes, then re-start capture to change the options for an RMONGrabber capture session.

RMONGrabber view of a capture window

Click on the *RMONGrabber* tab in the Capture window to open the ***RMONGrabber*** view. The ***RMONGrabber*** view contains two tabs: *Probe Capture Options* and *Probe Filter Options*. Each is described below. As those names imply, the options in the ***RMONGrabber*** view have to do primarily with settings on the remote probe. Just as the **Capture Options** dialog configures and controls the Capture window, so the ***RMONGrabber*** view controls the RMON probe.

Probe capture options

The ***Probe Capture Options*** view of the ***RMONGrabber*** view contains packet slicing and capture buffer options for the remote probe, and controls how captured packets will be sent from the probe and accepted by the local Capture window.



Packets received: 0
Packets filtered: 0
Memory usage: 0%
Filter state: ⟵ Accept all packets

Collection options

☑ Collect packets during capture

☑ Collect packets after capture is stopped

Size of packet buffer 1000 kilobytes

☑ Limit each packet to 128 bytes

□ Update count every 0 seconds

Idle | RMONGrabber: Fast Ethernet Raw | Packets: 0 | Duration: 00:00:00

Figure 14.4 Probe Capture Options in the RMONGrabber view

Collection options

In RMONGrabber, *Collection options* control the process of sending the captured packets from the RMON probe to the local EtherPeek Capture window. You must check one or both of the checkboxes in the *Collection options* section.

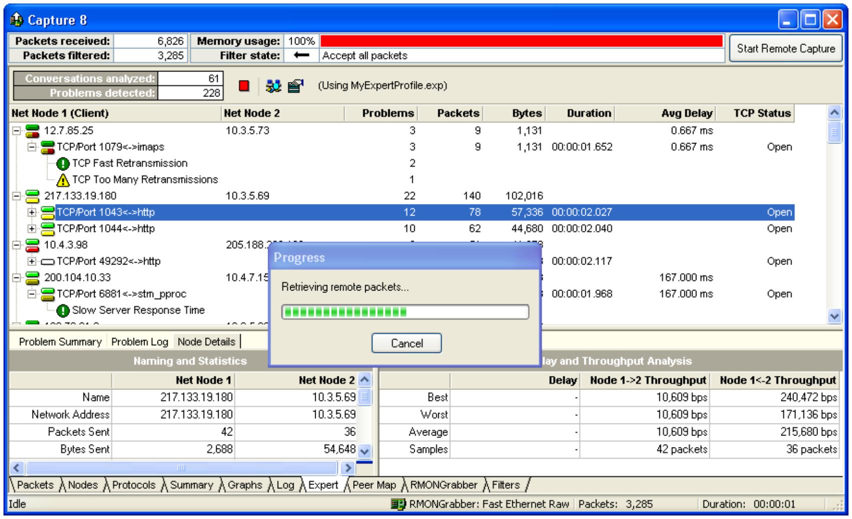
If you check *Collect packets during capture*, the RMON probe will send captured packets to EtherPeek while the remote capture is still under way. EtherPeek is expecting the packets, and will treat them like any other live traffic as they arrive. If this option is not checked, no packets will be seen in the local Capture window until the remote capture is stopped.

If you check *Collect packets after capture is stopped*, the RMON probe will wait until capture is stopped before sending any captured packets to EtherPeek. The local Capture window expects this behavior, and will accept the packets into the buffer as the RMON probe sends them.

The remote capture is normally stopped in one of two ways. Either the remote buffer is full and the configured capture is completed, or you clicked the **Stop Remote Capture** button and told the RMON probe to stop capture.

Equivalents to clicking the **Stop Remote Capture** button will also stop the remote capture--for example, a Stop Trigger set to a time or number of bytes captured, or the local Capture window's buffer becoming full when the *Continuous capture* option is not enabled. In this last case, the packets from the RMON probe will not be entered into the Capture window buffer, even if *Collect packets after capture is stopped* is enabled, since no space remains in the local buffer.

If the *Collect packets after capture is stopped* option is not checked, then stopping capture in the local Capture window will immediately stop the entry of new packets into the local buffer. This allows you to stop the flow of packets when you have found what you are looking for.



Packets received:	6,826	Memory usage:	100%
Packets filtered:	3,285	Filter state:	Accept all packets

Conversations analyzed:	61
Problems detected:	228

(Using MyExpertProfile.exp)

Net Node 1 (Client)	Net Node 2	Problems	Packets	Bytes	Duration	Avg Delay	TCP Status
12.7.85.25	10.3.5.73	3	9	1,131		0.667 ms	
TCP/Port 1079<->imaps		3	9	1,131	00:00:01.652	0.667 ms	Open
TCP Fast Retransmission		2					
TCP Too Many Retransmissions		1					
217.133.19.180	10.3.5.69	22	140	102,016			
TCP/Port 1043<->http		12	78	57,336	00:00:02.027		Open
TCP/Port 1044<->http		10	62	44,680	00:00:02.040		Open
10.4.3.98	205.188.						
TCP/Port 49292<->http					00:00:02.117		Open
200.104.10.33	10.4.7.15					167.000 ms	
TCP/Port 6881<->stm_pproc					00:00:01.968	167.000 ms	Open
Slow Server Response Time							

Retrieving remote packets...

Naming and Statistics

	Net Node 1	Net Node 2
Name	217.133.19.180	10.3.5.69
Network Address	217.133.19.180	10.3.5.69
Packets Sent	42	36
Bytes Sent	2,688	54,648

lay and Throughput Analysis

	Delay	Node 1->2 Throughput	Node 1<-2 Throughput
Best	-	10,609 bps	240,472 bps
Worst	-	10,609 bps	171,136 bps
Average	-	10,609 bps	215,680 bps
Samples	-	42 packets	36 packets

Idle | RMONGrabber: Fast Ethernet Raw | Packets: 3,285 | Duration: 00:00:01

Figure 14.5 RMONGrabber collecting packets after capture stopped

When both these options are checked, the RMON probe will begin sending packets as soon as they are captured, and the local Capture window will process them as they arrive. The probe will continue to send, and the local Capture window will continue to process packets after the capture has stopped. This is a useful setting for high traffic environments, where the RMON probe can capture at network speed and capture runs ahead of the probe's ability to transmit the captured packets to EtherPeek. Enabling both options lets you see packets sooner, and allows all the packets in the probe's buffer to be sent with greater certainty.

Other probe capture options

Packet capture with RMONGrabber actually takes place on the remote RMON probe. You must set the size of the probe's buffer. In the *Size of packet buffer … kilobytes* line, enter a value in kilobytes.

RMON probes are small devices with relatively small amounts of memory. When you set the buffer size, RMONGrabber connects to the probe and requests a buffer of the size you entered. If the probe cannot allocate that much space, it will refuse the request and RMONGrabber will display an error message. Different probes have different capabilities, but a buffer of 50 MB would be a very large request, while the RMONGrabber default buffer of 2 MB (*2000 kilobytes*) is a very modest one.

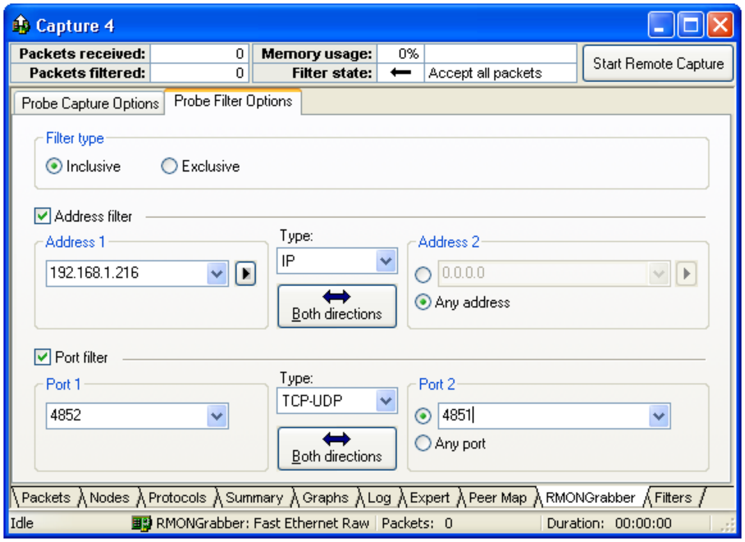
To minimize the use of scarce buffer space on the RMON probe, you may want to use packet slicing. Packet slicing captures only the first *n* bytes of each packet and discards the rest. Ethernet address information is contained in the first 14 bytes, for example. The protocol information for most protocols is contained in the first 128 bytes, with the remainder of the packet containing application data such as web pages and database records. If you set the slice value too low, capturing too little of the packet, you may miss the information you need to troubleshoot. Even a generous slice value, however, can greatly reduce the amount of space required in the buffer of the RMON probe.

To enable packet slicing on the RMON probe, check the checkbox beside *Limit each packet to … bytes* and enter the number of bytes of each packet you wish to keep.

When you are capturing from a remote probe, RMONGrabber updates the *Packets received* item in the Capture window header to indicate how many packets have been captured at the RMON probe. It does this by querying the probe periodically. You can control this frequency by checking the checkbox beside *Update count every … seconds* and entering a value in seconds.

Probe Filter Options

The ***Probe Filter Options*** view of the ***RMONGrabber*** view allows you to define an address filter and/or a port filter to limit the packets captured into the buffer of the remote RMON probe.



Packets received: 0
Packets filtered: 0
Memory usage: 0%
Filter state: Accept all packets

Filter type
● Inclusive ○ Exclusive

☑ Address filter
Address 1: 192.168.1.216
Type: IP
Address 2: ○ 0.0.0.0 ● Any address

☑ Port filter
Port 1: 4852
Type: TCP-UDP
Port 2: ● 4851 ○ Any port

Idle | RMONGrabber: Fast Ethernet Raw | Packets: 0 | Duration: 00:00:00

Figure 14.6 Probe Filter Options in the RMONGrabber view

At the top of the ***Probe Filter Options*** view in the *Filter type* section is a pair of radio buttons that tell the probe how to use the filters defined below in this view. Click the *Inclusive* radio button to have the probe accept all packets matching the filters defined below. Click the *Exclusive* radio button to have the probe reject all packets matching the filters.

Define the filters using the controls presented in the remainder of the view. These controls are laid out like the ***Simple*** view of the **Edit Filter** dialog in EtherPeek, but offer only two address types (*Physical* or *IP*) and one port type (*TCP-UDP*). For help in entering the Address filter information, please see “Specifying address filter parameters” on page 203. For help in specifying the Port filter parameters, please see “Specifying port filter parameters” on page 208.

Post-capture Analysis

Much of the work of troubleshooting problems on a network is a process of narrowing down the possibilities, examining first one set of clues and then another. EtherPeek provides a number of tools for analyzing packets, for selecting, grouping, and sorting them by a variety of attributes. This chapter starts with the most basic selection methods and concludes with the more sophisticated tools for evaluating groups of packets.

The ***Expert***, ***Peer Map,*** and statistical views of Capture windows and Packet File windows are recalculated and redrawn each time there is a change in the visible packets in the ***Packets*** view. By selecting, hiding and unhiding packets, a user can perform sophisticated analysis on captured traffic quickly and easily.

This chapter explains how to select, group, manipulate and process captured packets in Packet File windows and in Capture windows.

In this Chapter:

Captured and saved packets

The techniques described in this chapter are applied to packets that have already been captured and are in the buffer of either a Packet File window or of a Capture window. There is no requirement for these packets to have been saved in a Packet File, but they must be present in the buffer in order to be the object of selection, hiding, unhiding, and so forth.

Although it is possible to use many of the packet selection techniques while capture is still under way, other key functions are not available in a Capture window until capture has been stopped. The **Edit** menu functions that allow you to hide and unhide packets, and the **File** menu choices of **Save All Packets…** and **Save Selected Packets…** are not available in a Capture window until capture in that window is stopped.

Tip You can, however, use the context menu in the Packet List pane of the ***Packets*** view to **Copy Selected Packets to New Window** while capture is underway. For details, see “Copy selected packets to new window” on page 286.

For a complete discussion of saving and reloading packets, please see “Saving, loading and printing captured packets” on page 82.

Tip The **Save All Packets…** command saves all packets currently visible in the active window, whether selected or not. Any hidden packets will *not* be saved.

Using basic select and hide functions

When items are *selected*, that state is shown by the fact that they are highlighted. You can select items in any of the following views of a Capture window or a Packet File window:

- ***Packets***
- ***Nodes***
- ***Protocols***
- ***Conversations*** (this view exists in EtherPeek standard only)
- ***Expert*** (this view exists in EtherPeek NX only)
- ***Peer Map*** (this view exists in EtherPeek NX only)

While you can select the line entries in any of these views, the only place that *packets* are actually selected is in the Packet List pane of the ***Packets*** view. (Please see “Select related packets” on page 287, below.)

Basic selection

You can use all the standard selection techniques to choose items in any of the windows that allow selection. To highlight a single item, click on it. Clicking on another item highlights it instead. To highlight multiple items, hold down the **Ctrl** key when you click. To unhighlight any one item, hold down the **Ctrl** key and click on it again. To highlight a contiguous group of items, click on the first item, then hold down the **Shift** key when you click on the last item in the sequence. Everything between the two clicks (inclusive) will be highlighted.

The **Edit** menu adds a few more simple techniques. To highlight everything in the view, choose **Select All** from the **Edit** menu or press **Ctrl + A**. To remove all highlighting, choose **Select None** from the **Edit** menu or press **Ctrl + D**.

Choose **Invert Selection** from the **Edit** menu to reverse the highlighting.

Hide and unhide

Hiding packets removes them from view without actually deleting them. It is a handy way to quickly reduce the clutter of the ***Packets*** view. Hide functions are disabled for Capture windows when capture is under way.

Hidden packets are not processed by Analysis Modules or statistics, are not printed when the contents of the window are printed, and are not saved when you choose **Save All Packets…** from the **File** menu. They are, however, deleted when you select **Clear All Packets** from the **Edit** menu or press **Ctrl + B**.

To hide the selected packets, choose **Hide Selected Packets** from the **Edit** menu or press **Ctrl + H**. Alternatively, you can choose **Hide Unselected Packets** or type **Ctrl + Shift + H**. To restore all hidden packets to view, choose **Unhide All Packets** from the **Edit** menu or type **Ctrl + U**. You can continue to add to the hidden packets, hiding some now and more later, but there is no way to selectively unhide.

Note: Hiding or Unhiding causes all packets in the Capture window or Packet File window to be reprocessed by any enabled Analysis Modules and causes statistics to be recalculated based on the changed visible contents of the window’s buffer.

Hidden packets are a part of the total packets, but are not processed by any Analysis Modules, statistics, or further selections.

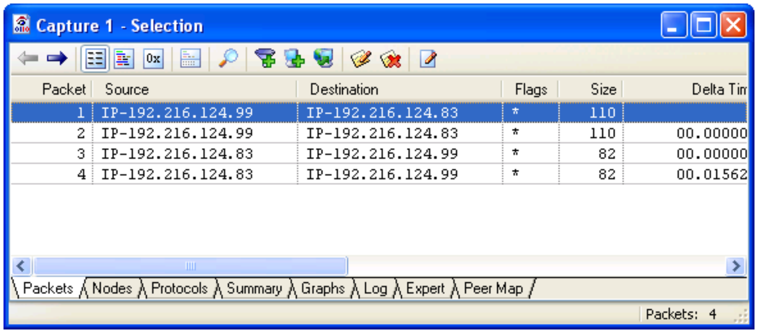
Navigating within selections

The **Go To…** and **Go To Next Selected** functions open the next packet in the selection in the **Packet Decode** window. They also move to that packet's listing in the ***Packets*** view of the active Capture window or Packet File window. Choose **Go To…** from the **Edit** menu or press **Ctrl + G** to bring up the **Go To** dialog. Fill in the number of the packet to which you want to jump. Choose **Go To Next Selected** from the **Edit** menu or press **Ctrl + J** to jump to the next packet in the selection.

Copy selected packets to new window

When one or more packets are selected in the ***Packets*** view of a Capture window or Packet File window, you can right-click on any part of the selection and choose **Copy Selected Packets to New Window** from the context menu. This creates a temporary Packet File window containing only the selected packets. The packets are renumbered, but the original packet order is retained.

The title bar of the window shows the name of the Packet File or Capture window from which the original selection was copied, with the word **Selection** added. You can continue the process, copying a further selection from the selection window, or copy a new selection from the original window. All of these windows will have the same name in the title bar, indicating the original file from which the first copy was made.



Packet	Source	Destination	Flags	Size	Delta Tir
1	IP-192.216.124.99	IP-192.216.124.83	*	110	
2	IP-192.216.124.99	IP-192.216.124.83	*	110	00.00000
3	IP-192.216.124.83	IP-192.216.124.99	*	82	00.00000
4	IP-192.216.124.83	IP-192.216.124.99	*	82	00.01562

Packets: 4

Figure 15.1 Temporary Packet File window created from selection

Each selection window is a fully functional Packet File window, but it is temporary. If you close any of these selection windows without saving, the information will be discarded. If you attempt to save any of these copied sets of packets, the **Save As** dialog will default to the name of the original source of packets. If you ignore the warning dialog, it is possible to replace the original file with the selection window.

Select related packets and find pattern

These more sophisticated selection tools essentially create pattern matching tools and apply them to the packets in the window.

Select related packets

The **Select Related Packets** command allows you to find packets that are like, or related to, the packet or data item currently selected. **Select Related Packets** presents a submenu of choices (shown in Table 15.1) allowing you to define which aspect(s) of the currently selected item you want this new selection to match. **Select Related Packets** creates a detailed set of selection criteria based on the parameter you choose and on the values found in the currently selected item. It then tests all the visible packets in the ***Packets*** view of the Capture window or Packet File window against those criteria and selects the ones that match.

To select related packets:

1. Highlight an item in the ***Packets***, ***Nodes***, or ***Protocols*** view of a Capture window or Packet File window. In EtherPeek standard only, you can also highlight items in the ***Conversations*** view. In EtherPeek NX only, you can also highlight items in the ***Expert***, or ***Peer Map*** views.
2. Choose **Select Related Packets** from the **Edit** menu, or right-click and choose **Select Related Packets** from the context menu.
3. From the submenu (shown in Table 15.1), choose the particular parameter set by which you want to define the relationship. Note that the submenu is context-sensitive and will only show the parameters that make sense for the item you initially highlighted.
4. If the current Capture window or Packet File window contains any related packets, the **Selection Results** dialog will open, showing the number of *packets selected*.

5. Use the **Selection Results** dialog to **Hide Selected** or **Hide Unselected**, or to do neither by clicking on **Close**.

Table 15.1 Submenu choices for Select Related Packets command

Parameter	Action
By Source	Chooses packets with matching source address.
By Destination	Chooses packets with matching destination address.
By Source and Destination	Chooses packets with matching source and destination addresses.
as Source or Destination	Unique to the ***Peer Map*** view, chooses packets showing the current node as either the source or destination address.
By Protocol	Chooses packets with matching protocol.
By Port	Chooses packets with matching port.
By Conversation	Chooses packets sent between two nodes (in either direction), using the matching protocol and port.
By Event Type	Unique to the Event Summary pane of the ***Expert*** view, this chooses all packets flagged with the particular event highlighted in the Event Summary.
By Flow	Unique to the Event Log pane of the ***Expert*** view, this item chooses packets sent between two nodes (in either direction), using the matching protocol and port.
Selected Entries	Unique to the Event Log pane of the ***Expert*** view, this item chooses only the individual packet identified with each highlighted entry in the Event Log. The Event Log shows one packet with one event in each log entry. Multiple log entries may be highlighted at once.
Selected Entries + "See" or "From Pkt"	Unique to the Event Log pane of the ***Expert*** view, this item chooses the individual packet identified with each highlighted entry in the Event Log, plus any packet referred to in the log entry in a phrase which begins "*See Packet…*" or "*From Packet….*" These log entries refer to another packet in the same conversation, such as a response or request packet, for example.

The **Select Related Packets** sub-menu of commands is available from the **Edit** menu, or from the context menu (right-click) where applicable. Not every submenu choice is available in every view. When you highlight a particular item in a statistical view, the **Select Related Packets** sub-menu items will change to match the context. EtherPeek standard and EtherPeek NX offer different views. Table 15.2 shows which sub-menu commands may be available in each of the four views found in EtherPeek standard: ***Packets***, ***Nodes***, ***Protocols***, and ***Conversations***. Table 15.3 shows which sub-menu commands may be available in each of the five views found in EtherPeek NX: ***Packets***, ***Nodes***, ***Protocols***, ***Expert***, and ***Peer Map***. As a more general guide, remember that the highlighted item must contain some value for the parameter by which you wish to select. This explains why you cannot select **By Source** address when you have highlighted an item in the ***Protocols*** view, nor select **By Protocol** when you have highlighted an item in the ***Nodes*** view.

Table 15.2 **Select related packets, parameter availability by view**

EtherPeek standard	*Packets*	*Nodes*	*Protocols*	*Conversations*
By Source	yes	yes		
By Destination	yes	yes		
By Source and Destination	yes	yes		yes
By Protocol	yes		yes	
By Port	yes			
By Conversation	yes			yes

Table 15.3 Select related packets, parameter availability by view

EtherPeek NX	*Packets*	*Nodes*	*Protocols*	*Expert*	*Peer Map*
By Source	**yes**	**yes**			**modified***
By Destination	**yes**	**yes**			**modified***
By Source and Destination	**yes**	**yes**		**yes**	**modified***
By Protocol	**yes**		**yes**		
By Port	**yes**				
By Conversation	**yes**			**yes**	
By Event Type				**yes**	
Selected Entries				**yes**	
Selected Entries + “See” or “From Pkt”				**yes**	

***** The ***Peer Map*** view offers a modified version of the Select Related Packets function. You can use the highlighted node **as Source**, **as Destination**, or **as Source or Destination** for a Select Related Packets function. Note that there is no selection **By Source and Destination**, only selection using the current node **as Source or Destination**.

The **Select Related Packets** command creates the most specific match it knows how to make, based on the parameters you chose and the item you selected. For example, if you highlight a single ARP request packet in ***Packets*** view and choose **Select Related Packets** > **By Protocol**, you will find the selection includes no ARP response packets, only requests. If you go to the ***Protocols*** view and select the ARP protocol itself, which includes both requests and responses, and invoke **Select Related Packets** > **By Protocol** from there, you will find all the ARP traffic highlighted in the ***Packets*** view.

When you use the **Select Related Packets** command, a dialog appears telling how many packets EtherPeek selected and offering to **Hide Selected** or **Hide Unselected**, or to do neither by clicking on **Close**.

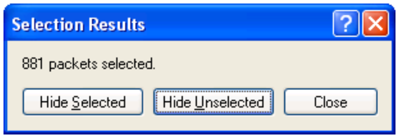


Figure 15.2 Selection Result dialog offers to hide, or just select with Close

Find pattern and find next

The **Find Pattern** and **Find Next** commands are a matched pair of tools. **Find Pattern** finds matches of a user-defined string at a user-defined location. To open the **Find Pattern** dialog, choose **Find Pattern** from the **Edit** menu or press **Ctrl + F**. You must limit the area and type of search, by choosing from the *Find in* drop-down list. Your choices are:

Packet ASCII data Searches for a match with an ASCII string found anywhere in the raw data of the packet.

Packet Hex Data Searches for a match with a hex string found anywhere in the raw data of the packet.

Packet List Headers Searches for a match with a string found in the packet list headers; that is, with the text shown in the current set of columns in the Packet List pane of the ***Packets*** view for that packet.

Decoded Text Searches for a match with a string found in the text of the decoded packet. This is like doing a text search in the ***Decode*** view portion of the text file which would be created by choosing Save Selected Packets as Text for the currently selected packets.

Packet notes Searches for a match with a string found in any Note associated with any packet in the Packet List pane. This is like doing a search in the optional ***Notes*** column of the ***Packets*** view.

Enter a string and choose whether the search should be case sensitive. The first packet matching these criteria will be highlighted in the ***Packets*** view. To find the next matching packet in sequence, choose **Find Next** from the **Edit** menu or press **F3**.

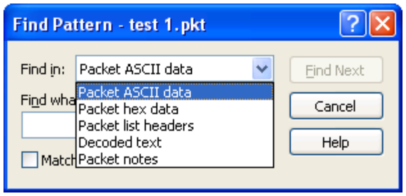


Figure 15.3 The Find Pattern dialog, showing the find in drop-down list

Tip The **Find Pattern** and **Find Next** commands search the packets in packet number order, starting from, but not including, the currently highlighted packet. The search does not wrap. In practice, this means that any matches in packet number 1 will not be found.

Select dialog: filters, analysis modules and more

The **Select…** command from the **Edit** menu brings up the **Select** dialog that allows you to use existing filters to select captured packets, to select based on string content or packet length, or to select based on Analysis Modules. You can select either all packets matching your criteria or all those not matching. The **Select** dialog only applies to visible packets in the active Capture window or Packet File window.

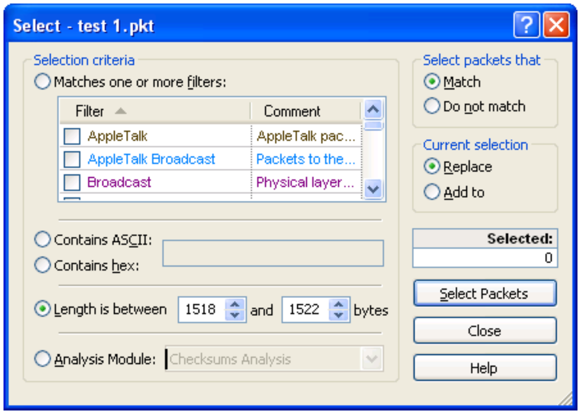
The **Select** dialog is also the only selection tool (other than the standard **Ctrl + click**) that allows you to add to an existing selection. Alternatively, you can choose to replace the current selection with the results of the new selection, as is the case with all other selection tools from the **Edit** menu.

Important! Packet slicing can affect the operation of some selection tools. When used from the **Select** dialog, filters, Analysis Modules and other selection tools read packet contents from the *captured* packets to determine protocols, addresses and related information. If the packet slice value was set in such a way as to discard some of the information these tools expect to find, they will not be able to identify packet attributes correctly.

To use the **Select** dialog to select packets in the Packet List of the active window:

1. Choose **Select…** from the **Edit** menu to open the **Select** dialog (Figure 15.4).
2. In the *Selection criteria* section, use the radio buttons to choose the method you will use to select the packets. Fill in the parameters for the chosen selection criteria. Each of the methods is described in its own section below. Your choices are:
   - Select based on filters
   - Select based on ASCII or hex character string

- Select based on packet length
- Select based on analysis modules



Select - test 1.pkt

Selection criteria

○ Matches one or more filters:

Filter	Comment
□ AppleTalk	AppleTalk pac...
□ AppleTalk Broadcast	Packets to the...
□ Broadcast	Physical layer...

○ Contains ASCII:

○ Contains hex:

● Length is between 1518 and 1522 bytes

○ Analysis Module: Checksums Analysis

Select packets that

● Match

○ Do not match

Current selection

● Replace

○ Add to

Selected: 0

Figure 15.4 Select dialog

3. Use the radio buttons marked *Match* or *Do not match* to choose whether to *Select packets that Match* the criteria you chose or packets that *Do not match* the selection criteria.
4. Use the radio buttons in the *Current selection* section to decide whether the results of this operation will *Replace* or *Add to* the *Current selection*.
5. Click the **Select Packets** button to perform the selection.

   A pane immediately above the **Select Packets** button shows the number of packets *Selected*. If any packets were selected, a **Selection Results** dialog will appear, noting how many packets were selected and offering the option to **Hide Selected**, **Hide Unselected**, or click **Close** to simply close the dialog without further action.
6. You can leave the **Select** dialog open and perform another selection, either adding to or replacing the current selection, or you can close the dialog by clicking the **Close** button.

Select based on filters

To select using one or more existing filters, click the *Matches one or more filters* radio button and check one or more filters from the list to enable them for selection.

Note: When multiple filters are enabled simultaneously, they are considered to be OR'ed together. That is, a packet matching any one of the enabled filters will be considered a match.

Select based on ASCII or hex character string

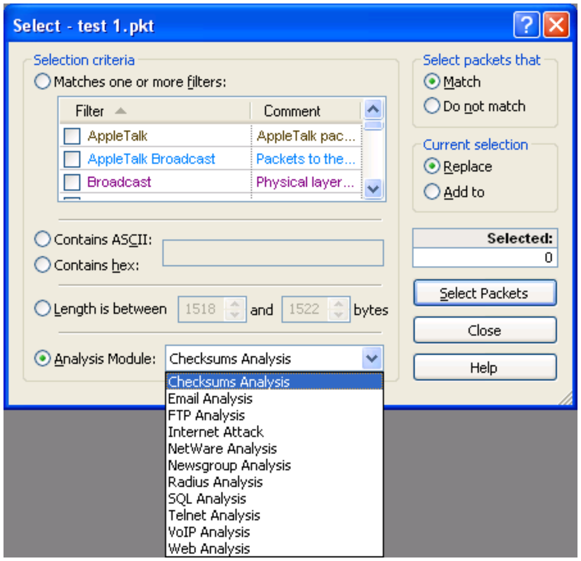
You can select packets which match a specified string found anywhere within the packet. To create a string selection, choose the appropriate radio button and enter the string for which you want to test. Choose either *Contains ASCII* for a text string, or *Contains hex* for a hexadecimal value.

Select based on packet length

Selecting by length checks for packet size, measured in bytes. To use this selection method, click the radio button beside *Length is between*. The default values in the dialog are set to *64* bytes and *1518* bytes, the minimum and maximum sizes, respectively, for ordinary Ethernet packets. You may set values outside this range if you wish. The upper and lower limit values are included in the search. Setting both values to the same number of bytes selects packets of that length only.

Select based on analysis modules

Analysis Modules can perform many different functions. Not all Analysis Modules support the select feature. Those that do are accessible in the **Select** dialog (Figure 15.5). Choose **Select…** from the **Edit** menu to bring up the **Select** dialog for the active window. In the *Selection criteria* section, click the *Analysis Module* radio button and choose an Analysis Module from the drop-down list. An Analysis Module will match a packet if it finds any of the data for which it tests in that packet.



Select - test 1.pkt

Selection criteria

○ Matches one or more filters:

	Filter	Comment
□	AppleTalk	AppleTalk pac...
□	AppleTalk Broadcast	Packets to the...
□	Broadcast	Physical layer...

○ Contains ASCII:
○ Contains hex:

○ Length is between 1518 and 1522 bytes

● Analysis Module: Checksums Analysis

Select packets that
● Match
○ Do not match

Current selection
● Replace
○ Add to

Selected: 0

Figure 15.5 Analysis Module choices in the Select dialog

Decoding Packets

When troubleshooting your network, tracking down a security breach, or simply gaining a better knowledge of protocols and network services; looking into the packets themselves is often very useful. When troubleshooting network applications, it is sometimes the only way to identify the real root of a problem.

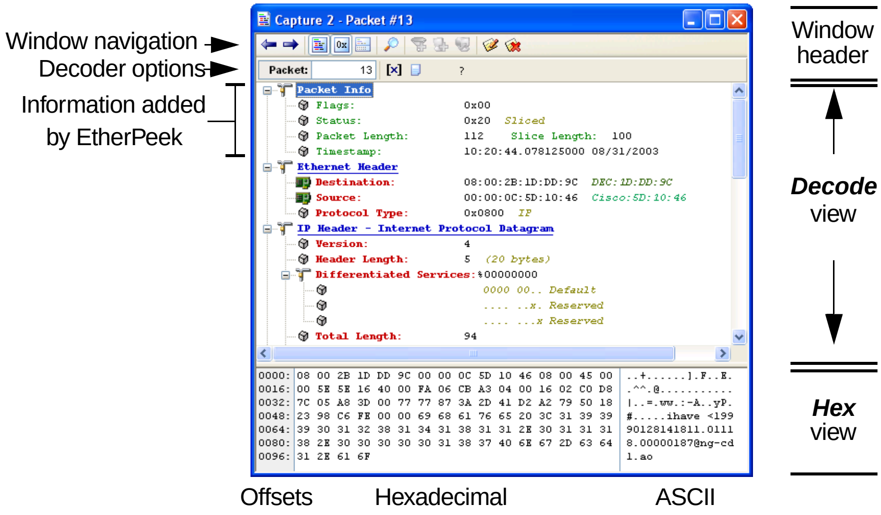
This chapter describes how to decode packets and read the packet headers, how to customize the way EtherPeek displays packet decodes, navigate through multiple selected packets and reconstruct the threads of network conversations.

CAUTION! Many protocols, especially the older Internet protocols such as HTTP, POP3, FTP, Telnet, and others transmit packet data in plain ASCII text. Controlling access to EtherPeek should be a normal part of your security routine.

In this Chapter:

The packet decode window

Double-click on any packet in a Packet List to open it in the **Packet Decode** window and see the data it contains as decoded information. The **Packet Decode** window makes packet headers readable and understandable. There are three basic parts to the display of a **Packet Decode** window: the window header, the ***Decode*** view and the ***Hex*** view. These are shown in Figure 16.1. Each of the parts of the **Packet Decode** window is described below.



Capture 2 - Packet #13

Packet: 13

```
Packet Info
    Flags:                    0x00
    Status:                   0x20  Sliced
    Packet Length:            112     Slice Length:  100
    Timestamp:                10:20:44.078125000 08/31/2003
Ethernet Header
    Destination:              08:00:2B:1D:DD:9C  DEC:1D:DD:9C
    Source:                   00:00:0C:5D:10:46  Cisco:5D:10:46
    Protocol Type:            0x0800  IP
IP Header - Internet Protocol Datagram
    Version:                  4
    Header Length:            5  (20 bytes)
    Differentiated Services:%00000000
                                 0000 00.. Default
                                 .... ..x. Reserved
                                 .... ...x Reserved
    Total Length:             94
```

```
0000: 08 00 2B 1D DD 9C 00 00 0C 5D 10 46 08 00 45 00  ..+......].F..E.
0016: 00 5E 5E 16 40 00 FA 06 CB A3 04 00 16 02 C0 D8  .^^.@...........
0032: 7C 05 A8 3D 00 77 77 87 3A 2D 41 D2 A2 79 50 18  |..=.ww.:-A..yP.
0048: 23 98 C6 FE 00 00 69 68 61 76 65 20 3C 31 39 39  #.....ihave <199
0064: 39 30 31 32 38 31 34 31 38 31 31 2E 30 31 31 31  90128141811.0111
0080: 38 2E 30 30 30 30 30 31 38 37 40 6E 67 2D 63 64  8.00000187@ng-cd
0096: 31 2E 61 6F                                      1.ao
```

Figure 16.1 Parts of a Packet Decode window

Packet decode window navigation

The **Packet Decode** window header contains the window title bar and the **Packet Decode** window view and navigation buttons. The window title bar shows the name of the file (Capture window or Packet File window) from which the displayed packet was taken, and the number of the packet in that Packet List.

The buttons immediately below the title bar allow you to move backward and forward through the active Packet List (**Decode Previous** and **Decode Next**), and to control which views of the **Packet Decode** window will be displayed. You can choose to **Show Decode View**, **Show Hex View**, or enable both. Click the **Toggle Orientation** button to switch between having the ***Decode*** view above and the ***Hex*** view below, or the ***Decode*** view at left and the ***Hex*** view at right. Click the **Zoom Pane** button to make the active view (the one with the current active highlight) the only visible view. Click the **Zoom Pane** button again to toggle back to the previous appearance. These window navigation buttons are shown in a detail of a **Packet Decode** window in Figure 16.3 on page 302.

You can step through the packets in the active Packet List in a number of ways. You can use the **Decode Previous** and **Decode Next** buttons as described above, or you can do the same thing using the function keys **F7** (previous) and **F8** (next), or use the keyboard short-cuts **Alt + left arrow** to decode the previous packet and **Alt + right arrow** to decode the next packet. Note that, whatever the method of stepping through, only the packets visible in the Packet List are available for decode. Packets hidden using any of the Hide functions on the **Edit** menu cannot be decoded in the **Packet Decode** window.

You can open individual **Packet Decode** windows for up to 10 packets at once. When multiple packets are selected in the active Packet List, click **Enter** to open them all. If more than 10 packets are selected, EtherPeek will display a message noting how many packets were selected and reminding you that only the first ten can be opened.

To open and view the contents of selected packets one at a time, select the packets and choose the **Go To…** command from the **Edit** menu, or press **Ctrl + G**. The **Go To** dialog opens, showing the packet number of the first packet in the current selection. Press **ENTER** (or click **OK**) to open the first selected packet. You can then use **Go To Next Selected** in the **Edit** menu or press **Ctrl + J** to close the **Packet Decode** window for the current packet and open a new one for the next packet in sequence in the current selection.

Tip The **Go To…** command finds the first packet of a selection for you. There is no need to scroll and look for it, as its number is displayed in the **Go To** dialog when it opens.

For a more complete view of selection options and techniques for navigating through selected packets, see “Navigating within selections” on page 286.

Decode view

The larger upper view of the **Packet Decode** window (shown in Figure 16.1) contains the ***Decode*** view, including the buttons controlling the application of decoder options. This section describes the ***Decode*** view. The decoder options are described in “Packet decoder options” below.

At the top of the data portion of the ***Decode*** view, the topmost fields are created internally by EtherPeek as it controls the Ethernet card. Most of these items relate to packet capture or to the state of the adapter, and are described in Table 16.1, below.

Table 16.1 Packet Decode information added by EtherPeek

Parameter	Description
Flags	Denotes errors and frame type.
Status	Indicates any one of several conditions, including that the packet was truncated or sliced. Shows a value of *0x00* when the packet does not have any of these other conditions.
Packet Length	The number of bytes that the adapter retrieved off the network for this packet, including all header information and FCS.
Slice Length	When *Slice Length* appears, it indicates the number of bytes of the packet which were captured. This is shown only if packet slicing was used on a packet, or if data was truncated because it was unavailable.
Timestamp	The time the packet was received.

The decoded packet data is presented in byte order from top to bottom. Click on the - minus or + plus signs in the margin to collapse or expand the view of any header section.

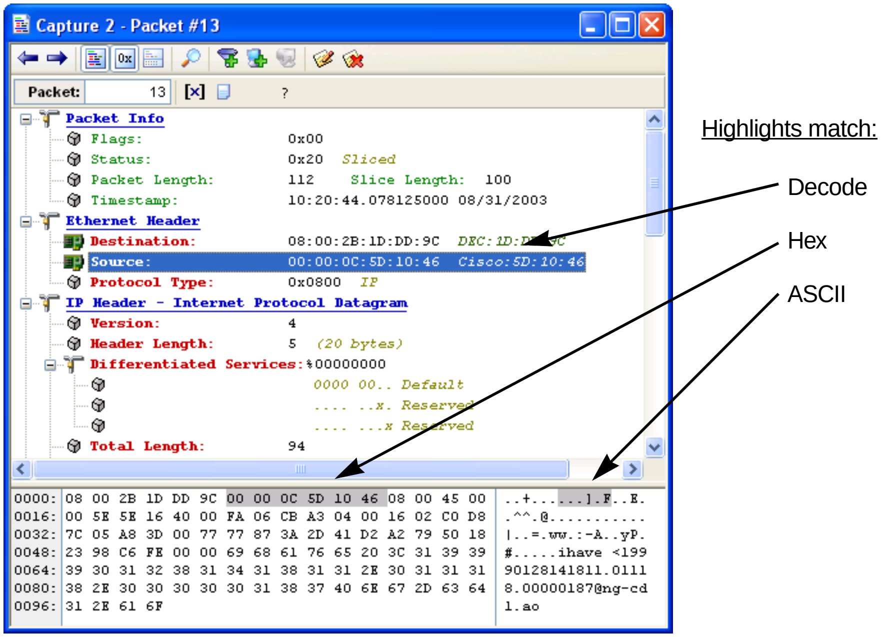
EtherPeek decodes many hundreds of network, transport, application and device control protocols, displaying both the commands and their meaning in English. When the data portion of the packet is listed toward the end of the ***Decode*** view simply as *data*, however, EtherPeek has reached a layer of the packet that it cannot decode with the current or default decoder. For details about selecting an alternative decoder, see “Choose decoder” on page 304. If you are writing your own protocols and wish to write your own decoders, please see “Writing your own decoders” on page 307.

Hex view: hex and ASCII packet contents

The bottom view pane of the **Packet Decode** window is the ***Hex*** view and contains the actual packet contents in raw hexadecimal on the left and its ASCII (or EBCDIC) equivalent on the right.

EtherPeek graphically links the ***Decode*** view with the ***Hex*** view for both hex and its ASCII equivalent. When you highlight a section of the ***Decode*** view, the corresponding portion of the hex data and the ASCII data in the ***Hex*** view is also highlighted, as shown in Figure 16.2. The reverse is also true. When you highlight an element in the ***Hex*** view, the corresponding element is highlighted in the ***Decode*** view.

When you right-click in the ***Hex*** view, it opens a context-sensitive menu with alternative display choices. The first permits you to toggle between displaying the text portion of the ***Hex*** view as **ASCII** or as **EBCDIC**. The second set of choices changes the notations at the left of the hex portion of the ***Hex*** view between **Decimal Offsets** and **Hexadecimal Offsets**. The third set of choices allow you to **Show Offsets**, **Show Hex** and/or **Show ASCII**. Each of these is a toggle, and has a checkmark beside it when enabled. The last item in the context menu, **Bytes Per Row**, opens a submenu of choices controlling the width of the ***Hex*** view. The choices are **Auto**, **8**, **10**, **16**, or **32** bytes per row. When you choose **Auto**, the ***Raw*** view expands to fill the space available in the current window. If Hex and ASCII are both shown, they retain their line-for-line symmetry.



```
Packet Info
   Flags:                     0x00
   Status:                    0x20  Sliced
   Packet Length:             112    Slice Length:  100
   Timestamp:                 10:20:44.078125000 08/31/2003
Ethernet Header
   Destination:               08:00:2B:1D:DD:9C   DEC:1D:DD:9C
   Source:                    00:00:0C:5D:10:46   Cisco:5D:10:46
   Protocol Type:             0x0800   IP
IP Header - Internet Protocol Datagram
   Version:                   4
   Header Length:             5   (20 bytes)
   Differentiated Services:%00000000
                                 0000 00.. Default
                                 .... ..x. Reserved
                                 .... ...x Reserved
   Total Length:              94

0000: 08 00 2B 1D DD 9C 00 00 0C 5D 10 46 08 00 45 00  ..+......].F..E.
0016: 00 5E 5E 16 40 00 FA 06 CB A3 04 00 16 02 C0 D8  .^^.@...........
0032: 7C 05 A8 3D 00 77 77 87 3A 2D 41 D2 A2 79 50 18  |..=.ww.:-A..yP.
0048: 23 98 C6 FE 00 00 69 68 61 76 65 20 3C 31 39 39  #.....ihave <199
0064: 39 30 31 32 38 31 34 31 38 31 31 2E 30 31 31 31  90128141811.0111
0080: 38 2E 30 30 30 30 30 31 38 37 40 6E 67 2D 63 64  8.00000187@ng-cd
0096: 31 2E 61 6F                                      1.ao
```

Figure 16.2 Highlights match: Decode, Hex, and ASCII data in a Packet Decode window

Packet decoder options

At the top of the ***Decode*** view of the **Packet Decode** window is a small header section showing the packet number and, to the right of that, buttons controlling the decoder options for the current packet. These buttons and their labels are shown in Figure 16.3. Each of these decoder options is discussed below.

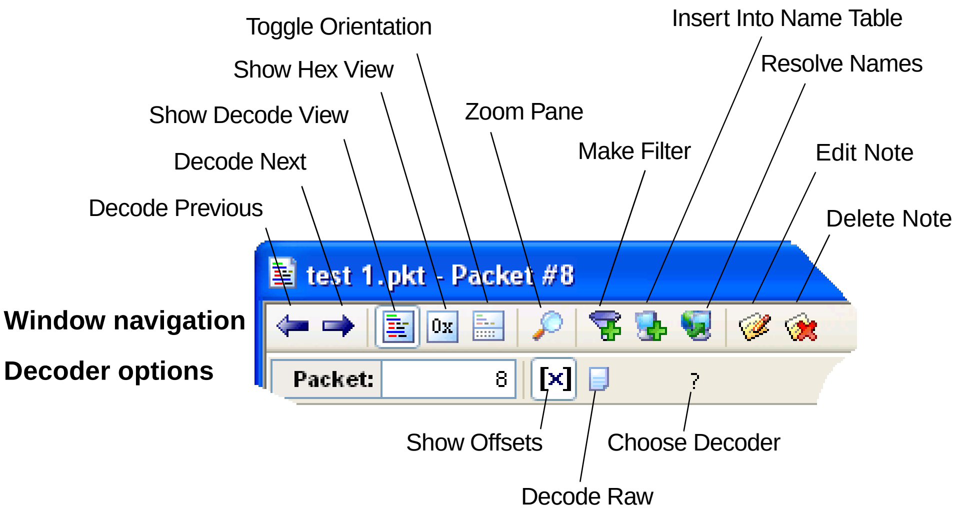


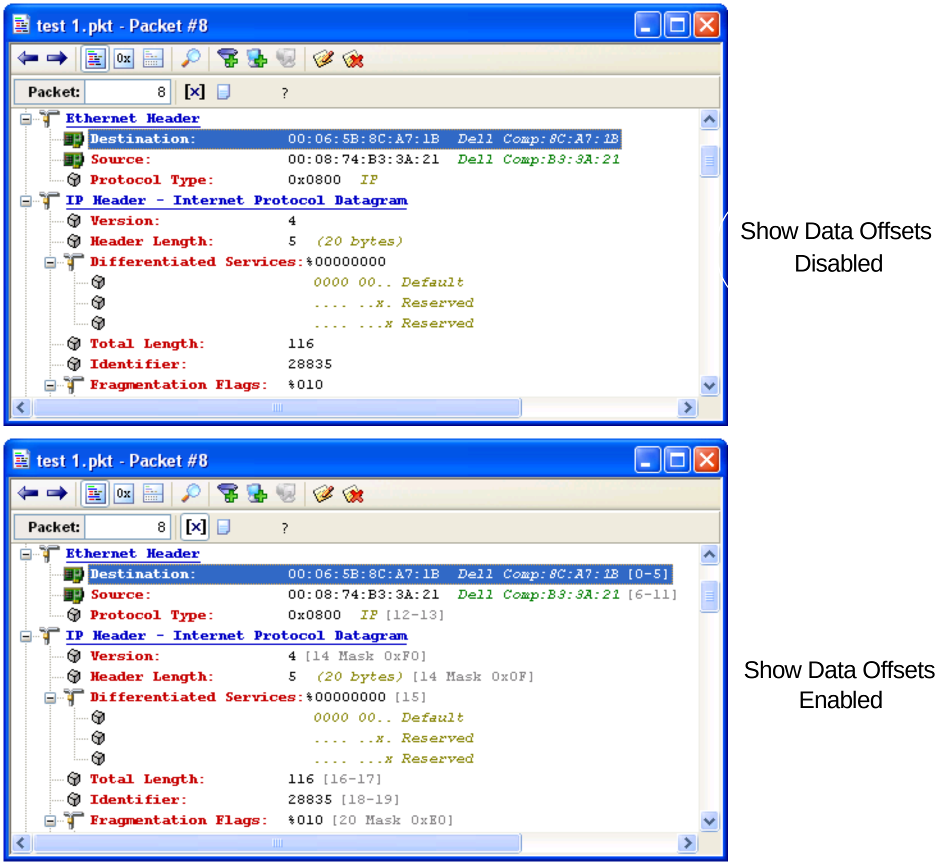
Figure 16.3 Detail of Packet Decode window: navigation and decoder options buttons

Show data offsets

The **Show Offsets** button toggles the display of data offset and mask information for all individual items in the ***Decode*** view. Offset is a measure of location within a packet, counted as the distance in bytes from the first byte of the packet. The offset of the first byte is “0,” that of the second byte is “1,” and so on. The mask is a mathematical way of defining a particular bit or bits within a byte. The offset and mask information is especially useful when developing protocols, constructing filters, and in a variety of other detailed packet analysis tasks.

Tip You can quickly create a filter that matches the value found at a particular point in a packet, directly from the ***Decode*** view or Decode pane. Highlight the item you wish to match and click the **Make Filter** button, or right-click and choose **Make Filter…** from the context menu. This opens the ***Advanced*** view of the **Edit Filter** dialog with a *Value* filter node matching the value, offset, and mask of the item you selected. You can give the new filter a name and click **OK** to save it. If you wish to edit the details of the filter, double-click on the new node to open it in the **Value Filter** edit dialog. For more information about Value filters, please see “Value filter nodes” on page 214.

The same packet is shown first without, and then with offsets in Figure 16.4.



```
Ethernet Header
    Destination:               00:06:5B:8C:A7:1B   Dell Comp:8C:A7:1B
    Source:                    00:08:74:B3:3A:21   Dell Comp:B3:3A:21
    Protocol Type:             0x0800   IP
IP Header - Internet Protocol Datagram
    Version:                   4
    Header Length:             5   (20 bytes)
    Differentiated Services:%00000000
                                 0000 00.. Default
                                 .... ..x. Reserved
                                 .... ...x Reserved
    Total Length:              116
    Identifier:                28835
    Fragmentation Flags:       %010
```

```
Ethernet Header
    Destination:               00:06:5B:8C:A7:1B   Dell Comp:8C:A7:1B [0-5]
    Source:                    00:08:74:B3:3A:21   Dell Comp:B3:3A:21 [6-11]
    Protocol Type:             0x0800   IP [12-13]
IP Header - Internet Protocol Datagram
    Version:                   4 [14 Mask 0xF0]
    Header Length:             5   (20 bytes) [14 Mask 0x0F]
    Differentiated Services:%00000000 [15]
                                 0000 00.. Default
                                 .... ..x. Reserved
                                 .... ...x Reserved
    Total Length:              116 [16-17]
    Identifier:                28835 [18-19]
    Fragmentation Flags:       %010 [20 Mask 0xE0]
```

Figure 16.4 Show Data Offsets—disabled above, enabled below

Decode raw data only

Click the **Decode Raw** button to present only the raw data found in the packet. Ordinarily, when you choose **Print Selected Packets…** from the **File** menu, or use **File** > **Save Selected Packets…** and choose any of the *Decoded Packets* formats, only the contents of the ***Decode*** view is printed or saved. If you wish to print or save the hexadecimal and ASCII contents of the ***Hex*** view of a packet, first click the **Decode Raw** button. Only the information added by EtherPeek and the contents of the ***Hex*** view will be printed or saved.

Choose decoder

You can open the **Select Decoder** window for certain packets by clicking the **Choose Decoder** button. The **Choose Decoder** button appears as a question mark (?) when this option is available for the current packet.

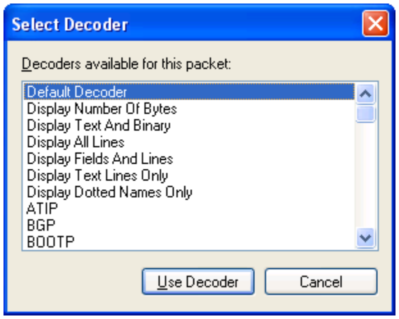


Figure 16.5 Select Decoder dialog

The **Select Decoder** window shows a context-sensitive list of decoders which can be applied to the current packet. If the packet contains TCP or UDP, this list will include generic line decoders such as *Display Number Of Bytes*. See Table 16.2 for a list of the available line decoders and their behavior. Alternatively or in addition, the **Select Decoder** window may present decoders for protocols which, because of their lack of uniquely identifying attributes, can often be mistaken for one another. Examples include particular types of RPC (Remote Procedure Call), TFTP (Trivial File Transfer Protocol), and others.

To use a particular decoder to decode the current packet *and all subsequent packets of the same type*, select the decoder from the list presented in the **Select Decoder** window and click the **Use Decoder** button at the bottom of the window.

If you wish to apply a different decoder to the same packet, or to all subsequent packets of this type, click the **Choose Decoder** button to re-open the **Select Decoder** window, choose the new decoder, and click **Use Decoder**. When the program believes that it knows how to decode the current packet properly, the **Select Decoder** window will present the *Default Decoder* choice at the top of the list of available decoders. You can choose this decoder to apply or re-apply the program's default decode behavior to the current packet (and all subsequent packets of the same type) at any time.

Note: Decoders only affect the display of data in the ***Decode*** view of **Packet Decode** windows and the Decode pane of Capture windows and Packet File windows. The ***Hex*** view or Hex pane always shows the actual packet data in hex and ASCII.

The **Choose Decoder** function is particularly useful in environments where new protocols are under development, or where TCP or UDP applications are using non-standard ports.

Table 16.2 Line decoders for TCP and UDP packets

Decoder	Shows
Default Decoder	When you select this decoder, the program returns to its default behavior when decoding packets of the current type. Use this selection to stop using any decoder previously selected in the **Select Decoder** window and restore the program's ability to choose its own decoder.
Display Number Of Bytes	This line decoder displays only the number of bytes in the UDP or TCP payload of the packet.
Display Text And Binary	This line decoder displays 0x00 through 0x1F as their code equivalents (0x00, for example, is *<NULL>*), displays (non-extended) ASCII characters as ASCII text, and displays any other values as a dot (.). In comparison, the ASCII part of the ***Hex*** view displays the extended ASCII character set (which includes accented characters, for example) and displays all non-ASCII values as dots.

Table 16.2 Line decoders for TCP and UDP packets (Continued)

Decoder	Shows
Display All Lines	This line decoder displays only (non-extended) ASCII characters, plus line feed / carriage return (0x0D and 0x0A). When it encounters the first value outside this set, the decoder stops and displays the number of bytes remaining in the payload portion of the UDP or TCP packet.
Display Fields And Lines	This line decoder searches for lines containing semi-colons (;). Each line with a semi-colon is split in two, with the part before the semi-colon treated as the label and the part to the right of the semi-colon treated as the data. Lines containing text without semi-colons are treated as for the *Display All Lines* decoder above. That is, non-extended ASCII text is displayed until the first non-ASCII character is reached. The decoder then displays the number of bytes remaining in the payload of the TCP or UDP packet. This decoder is particularly useful in quickly scanning through the Label;Value pairs found in HTTP and FTP packets, particularly when the transactions are taking place on ports other than the default port 80 (HTTP) or port 21 (FTP).
Display Text Lines Only	This line decoder displays all the non-extended ASCII characters, plus line feeds and carriage returns (LF/CR), ignoring all other characters. If no LF/CR is encountered, lines are automatically wrapped at 120 characters.
Display Dotted Names Only	This line decoder searches for lines of non-extended ASCII text containing the period character(.). It displays each such line. All other lines are ignored. This decoder is useful when scanning for file names and IP names and addresses that use dotted notation.

Important! When you choose a decoder in the **Select Decoder** window, EtherPeek will continue to use that decoder every time it encounters a packet of the same type. To restore the program's ability to choose its own decoder, select a packet of the same type, open the **Select Decoder** window, choose *Default Decoder* from the list, and click the **Use Decoder** button.

Writing your own decoders

If you find proprietary protocols on your network for which EtherPeek does not supply decoders, or if you are developing your own protocols, you may want to write your own decoders for use with EtherPeek.

EtherPeek lets you write your own packet decoders and add them to the Decodes directory for use with the application. Documentation on writing decoders is included in the 1033\Documents directory in the directory where you installed EtherPeek.

Note: Writing packet decoders requires programming knowledge.

Printing, saving and copying

To print decoded packets, open a **Packet Decode** window and make it the front-most or active window. From the **File** menu choose **Print** to print out a formatted version of *only* the ***Decode*** view of the **Packet Decode** window. An alternative is to save the decoded packets as RTF or HTML and print them from another application that can read and print those file types. This alternative preserves the formatting of the **Packet Decode** window. To print the decode portion of multiple packets as a single file, select the packets and choose **Print Selected Packets…** from the **File** menu.

To save packets in their decoded form, select the packets (highlight them) in the Packet List pane of the ***Packets*** view of a Capture window or a Packet File window. From the **File** menu, choose **Save Selected Packets** to open the **Save** dialog. In the **Save** dialog, choose a file type of plain text, RTF or HTML. Give the file a name and click **Save** to save the files to your chosen location.

To save or print the hexadecimal and ASCII contents of the Hex pane, click the **Decode Raw** button before saving or printing. For details, see “Decode raw data only” on page 304.

You can copy an individual line from any pane of a **Packet Decode** window to the clipboard and paste it into another application as plain text by using standard editing keystroke combinations.

Decode reassembled PDU

In the (right-click) context menu of the Packet List pane of the ***Packets*** view of a Capture window or Packet File window, you can choose to **Decode reassembled PDU**. The PDU is the Protocol Data Unit—roughly, the payload of a network application packet. When a web page, for example, is sent over the Internet, the page is broken into convenient sized pieces and transmitted in a series of packets. If you right-click on a packet containing one of the fragments of the web page and choose **Decode Reassembled PDU** from the context menu; EtherPeek will attempt to locate all the other pieces of this page, decode them, and present the results in a single temporary **Packet Decode** window. The window title bar of the resulting **Packet Decode** window will show a packet number, followed by the phrase **(Reassembled PDU)**. The packet number is the packet EtherPeek identified as the one containing the first part of the PDU.

To save or print the decode of the individual **Packet Decode** window containing the reassembled PDU, make it the active window, and choose **Save Packet…**, or **Print** from the **File** menu. For details of formats and file types, please see the previous section.

The **Packet Decode** window containing the decoded reassembled PDU is temporary. If you close the window without saving, the information will be discarded. In any case, creating a reassembled PDU does not change the contents of any of the packets in the Capture window or Packet File window.

Using thread intelligence in EtherPeek

Packets usually contain the information EtherPeek requires to decode them into their protocol components. For some protocols, however, the required information is not contained in the packet itself, but in a previous packet exchanged between the same two nodes. EtherPeek supports thread intelligence for some protocols, including Simple Network Management Protocol (SNMP), Simple Mail Transfer Protocol (SMTP), AppleTalk Session Protocol (ASP), Printer Access Protocol (PAP), NetWare Core Protocol (NCP), and others.

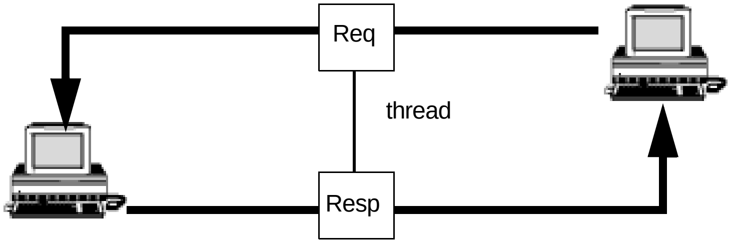


Figure 16.6 Protocol decode thread maintained by EtherPeek

When two or more packets are related to the same session in one of these protocols, EtherPeek can pre-decode them in the order in which they arrived, allowing the Request/

Response pairs to be connected. This provides a richer set of decode information than would otherwise be available. This relationship between packets is called a thread, and the pre-decoding done to establish the thread is called making a thread.

To make threads, select the packets in the Packet List among which you believe threads may exist. You can use **Ctrl + A** to select all packets. Right-click and choose **Make Threads** from the context menu.

EtherPeek uses threads to keep track of the protocol type in decoding Response packets associated with a particular Request.

There are two ways to employ thread intelligence in EtherPeek:

- The **Select Related Packets** command—to find possibly related threads.
- The **Make Threads** command—to automatically create any threads from packets near the selected packets.

Manually selecting further decode options

If you view the Request packet first, EtherPeek keeps track of the thread when you open the corresponding Response and Release packets. However, if you view a Response packet before you have opened a preceding Request, no thread will have been started, and EtherPeek will simply show a question mark (**?**) instead of the protocol type at the top of the **Packet Decode** window.

You can click on the **Choose Decoder** button (a question mark) to open the **Select Decoder** dialog, and then manually choose the decoder to use.

As an alternative to manually selecting options for further decoding packets, you can instruct EtherPeek to make threads before opening any packets. This ensures that the threads will exist even if you open a Response packet first. To make threads in the background before you open packets, use the **Select Related Packets** command or **Select All Packets** (either from the **Edit** menu or from the context menu), and then choose the **Make Threads** command from the context menu (right-click). You can then view packets in any order.

Sending Packets

EtherPeek is capable of sending as well as receiving packets.

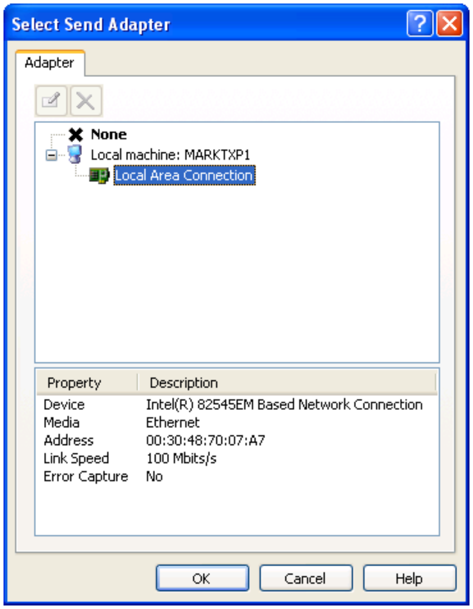
You can use the packet transmission feature to generate network traffic or to probe specific computers to observe their reactions. You can also check network connections by using the send function at the computer being checked, while using a second computer running EtherPeek to observe the resulting activity. Developers can also use the send features to test protocol implementations.

You can send a single packet, a set of bursts at intervals, or a single burst of packets. You can send a generic TCP/IP packet, or select any captured packet as the Send Packet. You can also edit the contents of the Send Packet.

The Send function lets you test potential problems actively, without having to wait for events to reveal a possible source of trouble.

In this Chapter:

Select send adapter



Property	Description
Device	Intel(R) 82545EM Based Network Connection
Media	Ethernet
Address	00:30:48:70:07:A7
Link Speed	100 Mbits/s
Error Capture	No

Figure 17.1 Select Send Adapter dialog

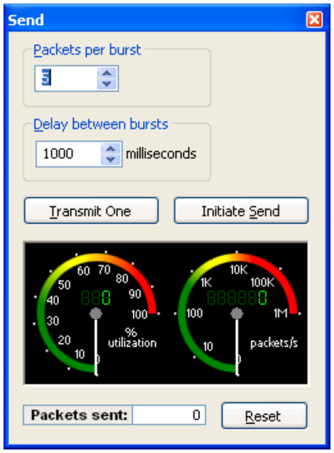
In order for EtherPeek to send packets, you must first select an adapter to use for this purpose. Under the **Send** menu, choose **Select Send Adapter…** to open the **Select Send Adapter** dialog. Select a valid NIC as the adapter and click **OK** to make your choice.

The send packet

EtherPeek ships with a generic Ethernet packet already set as the default Send Packet. Alternatively, you may choose another packet to send onto the network. You can select any packet from the ***Packets*** view of any active window and set it as the Send Packet by selecting the packet and choosing **Set Send Packet** from the **Send** menu.

To send

To send traffic onto the network from EtherPeek or to set the parameters for send events, choose **Send Window** from the **Send** menu.



Send

Packets per burst

3

Delay between bursts

1000 milliseconds

Packets sent: 0

Figure 17.2 Send window

At the bottom of the **Send** window (Figure 17.2) are two dials with digital readouts. They show the *% utilization* (percent of utilization of maximum network bandwidth) and *packets/s* (packets per second) represented by the packets being sent in the current send event. The **Send** window also shows the total *Packets sent* in the current send event.

There are several ways to send packets:

- send a single copy of the Send Packet out on the network.
- send bursts of multiple copies of the Send Packet at specified intervals.
- send a selected packet or group of packets in a single burst.

Transmit one

The simplest form of sending a packet is to use **Transmit One**. Select **Transmit One** from the **Send** menu, click the **Transmit One** button in the **Send** window, or simply type **Ctrl + T**. This causes the immediate transmission of exactly one of the specified Send Packet.

Send multiple copies of a packet at specified intervals

The second way to generate traffic is to transmit copies of the Send packet in bursts at specified intervals. Use the text entry boxes in the **Send** window to establish the number of *Packets per burst* and the *Delay between bursts*, in milliseconds. The text entry boxes can be edited directly or you can use the arrows at the right to set these numbers. Note that the minimum delay between bursts is one millisecond.

When you have set these parameters, you can initiate the send process by selecting the **Initiate Send** command from the **Send** menu, clicking the **Initiate Send** button in the **Send** window, or simply typing **Ctrl + I** (letter “i”). When you initiate a send, the **Initiate Send** command in the **Send** menu changes to **Halt Send**.

CAUTION! Sending large volumes of traffic onto the network can slow down service for other users. Also, if you set the **Send** window to send a large number of packets with too small an interval, you may prevent your computer from doing any of the other tasks that it does normally. If this happens, the computer will seem sluggish, and in the most severe cases, the computer may not even respond to your attempts to stop transmission or to quit EtherPeek.

Sending selected packets

The **Send Selected Packets** command in the **Send** menu is enabled when you are in the ***Packets*** view of a Capture window or a Packet File window and a packet or packets are selected. The selected packets will be sent in a single burst at one millisecond intervals between packets.

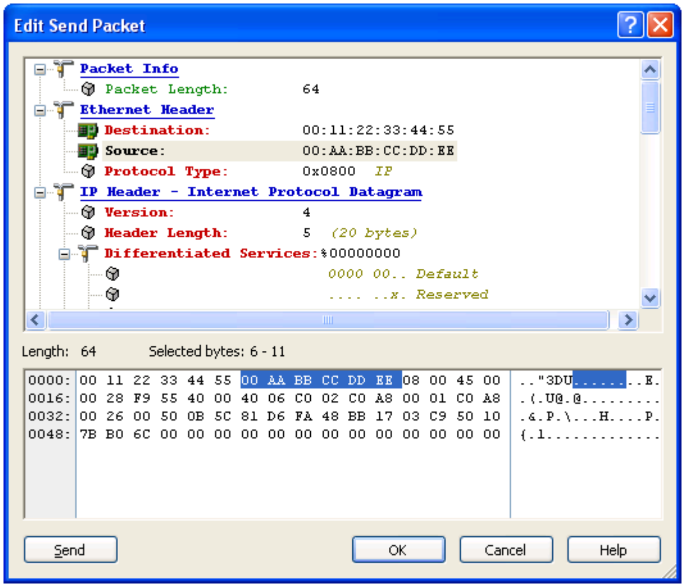
Editing send packet contents

EtherPeek ships with a generic Ethernet packet as the default Send Packet. Alternatively, you may select another packet to send onto the network. You can choose any packet from the ***Packets*** view of any active window and set it as the Send Packet by selecting the packet and choosing **Set Send Packet** from the **Send** menu.

CAUTION! Setting a Broadcast or a Multicast packet as the Send Packet will cause all nodes to process this packet and force switches to forward the packet onto all segments.

To edit the contents of the Send Packet:

1. Choose **Edit Send Packet** from the **Send** menu to open the **Edit Send Packet** window (Figure 17.3).



```
Packet Info
    Packet Length:            64
Ethernet Header
    Destination:              00:11:22:33:44:55
    Source:                   00:AA:BB:CC:DD:EE
    Protocol Type:            0x0800   IP
IP Header - Internet Protocol Datagram
    Version:                  4
    Header Length:            5   (20 bytes)
    Differentiated Services:%00000000
                                 0000 00.. Default
                                 .... ..x. Reserved
```

Length: 64 Selected bytes: 6 - 11

```
0000: 00 11 22 33 44 55 00 AA BB CC DD EE 08 00 45 00  .."3DU........E.
0016: 00 28 F9 55 40 00 40 06 C0 02 C0 A8 00 01 C0 A8  .(.U@.@.........
0032: 00 26 00 50 0B 5C 81 D6 FA 48 BB 17 03 C9 50 10  .&.P.\...H....P.
0048: 7B B0 6C 00 00 00 00 00 00 00 00 00 00 00 00 00  {.l.............
```

Figure 17.3 Editing a Send Packet

2. The layout of the **Edit Send Packet** window is similar to that of the **Packet Decode** window, with a ***Decode*** view above and a ***Raw Data*** view below. Each line of the ***Raw Data*** view begins at the left with the offset of the first character of that line, followed by 16 bytes of hex data (one two-digit hexadecimal number per byte) followed by the same 16 bytes represented in ASCII characters (one character per byte).

3. Each ASCII character is the equivalent of its corresponding hexadecimal pair on the left hand side. You can edit either of the representations by directly overwriting the contents.

   The highlighting in the three parts of the **Edit Send Packet** window makes it easy to keep track of where in the packet your edits are being made.

4. In the display area between the ***Decode*** view and the ***Raw Data*** view, the **Edit Send Packet** window shows the *Length* of the packet (including all headers) and the data offsets of the *Selected bytes*.

5. In the ***Decode*** view of the **Edit Send Packet** window in Figure 17.3 above, the decode still shows an accurate decoding of this part of the packet. As editing proceeds, the ***Decode*** view attempts to update on-the-fly and show an accurate decode of the Send Packet, as edited.

6. When you have finished editing the Send Packet, choose **OK** to use the changes you have made, or click **Cancel** to ignore any changes and leave the Send Packet as it was.

Appendices

Packets and Protocols

The following section is a brief introduction to the concepts of packets and protocols. For a list of recommended readings on networking topics, please visit our website at: http://www.wildpackets.com/support/resources.

About Ethernet

Ethernet is the most popular LAN technology in the world. It is an easy, relatively inexpensive way to provide high-performance networking to all different types of computer equipment.

Ethernet was invented at Xerox PARC and developed jointly by Digital Equipment Corporation, Intel and Xerox. Introduced in 1980, Ethernet was distinguished by its high speed (10 Mbps), its unusual signaling methodology (the latest version of which is now referred to as Carrier Sense Multiple Access with Collision Detection or CSMA/CD), and by the physical medium on which it ran: a thick, high-quality coaxial cable with a bright yellow braided sheath.

Today, the term Ethernet refers to a whole family of closely related protocols characterized by their raw data rates (10 Mbps, 100 Mbps, 1 Gbps or 10 Gbps) and the physical medium on which they operate. Ethernet now runs on a wide variety of physical media. Among the most common are: coaxial cable (thick or thin), many types of copper cable called twisted pair, and several types of fiber-optic cables using a variety of signalling methods and light wavelengths.

The CSMA/CD approach is used by any form of Ethernet operating in *half duplex* mode—that is, the mode in which transmit (Tx) and receive (Rx) signals can be sent on the same wire or data path. In *full duplex* mode, transmit and receive signals are separated onto dedicated, one-way channels. This eliminates the need for CSMA/CD, as all the transmissions on a single data path will be coming from a single device. Half duplex mode is seldom used in versions of Ethernet running on fiber, and is not supported at all in the 10 Gbps standards.

What is a packet?

Each piece of information transmitted on an Ethernet network is sent in something called a *packet*. A packet is simply a chunk of data enclosed in one or more wrappers that help to identify the chunk of data and route it to the correct destination. *Destination* in this sense means a particular application or process running on a particular machine. These wrappers consist of *headers*, or sometimes headers and *trailers*. Headers are simply bits of data added to the beginning of a packet. Trailers are added to the end of a packet.

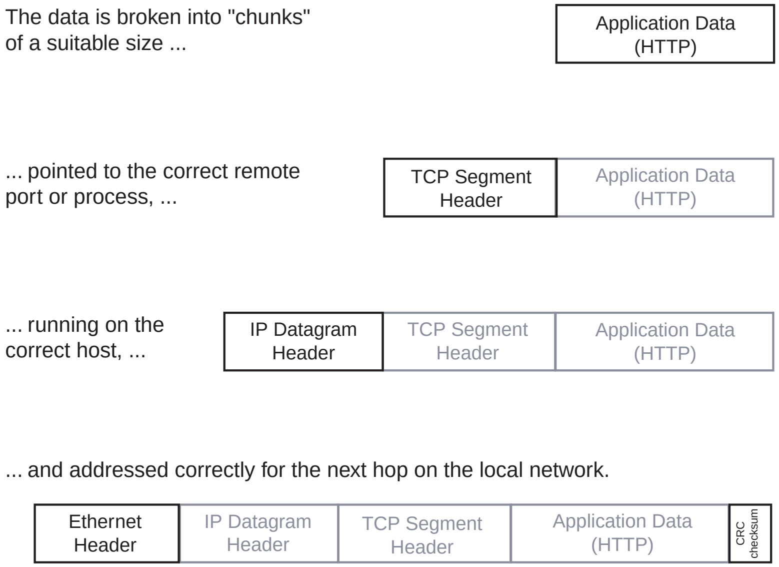

The data is broken into "chunks" of a suitable size ...
Application Data (HTTP)
... pointed to the correct remote port or process, ...
TCP Segment Header
Application Data (HTTP)
... running on the correct host, ...
IP Datagram Header
TCP Segment Header
Application Data (HTTP)
... and addressed correctly for the next hop on the local network.
Ethernet Header
IP Datagram Header
TCP Segment Header
Application Data (HTTP)
CRC checksum

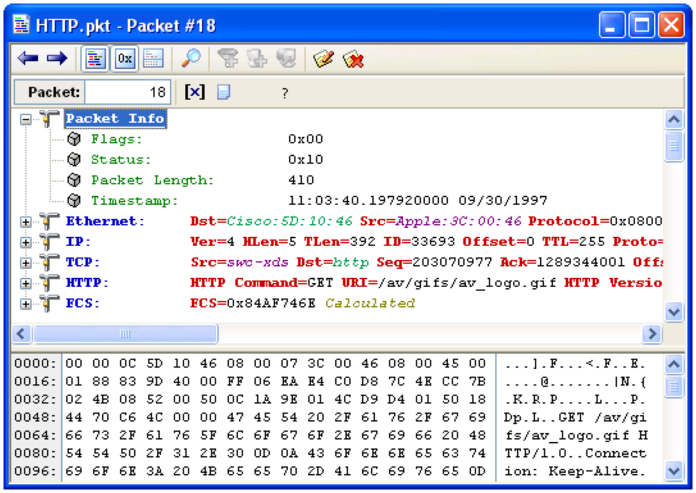
Figure A.1 Constructing a network data packet (here, a piece of a web page)

Packets are created at the machine sending the information. The application generating the data on the sending machine passes the data to a *protocol stack* running on that machine. The protocol stack breaks the data down into chunks and wraps each chunk in one or more wrappers that will allow the packets to be reassembled in the correct order at the destination. The protocol stack on the sending machine then passes the packets to the Ethernet hardware: the *NIC* (Network Interface Card). The Ethernet hardware adds its own wrapper (the Ethernet header and trailer) to each packet to direct it to the correct destination on the local network.

If the packet’s ultimate destination is somewhere off the local network, the Ethernet header added by the sending machine will point to a router or switch as its destination address. The router will open the packet, strip off the Ethernet wrapper, read far enough to find the ultimate destination address and re-wrap the packet, giving it a new header that will send it on the next hop of its journey.

At the receiving end, the process is reversed. The packet is read by the NIC at the receiving machine which strips off the Ethernet header and passes the packet up to the appropriate protocol stack. The protocol stack reads and strips off its headers and passes the remaining packet contents on up to the application or process to which it was addressed, reassembling the chunked data in the correct order as it arrives.

The packet diagramed in Figure A.1 above is shown in a **Packet Decode** window in Figure A.2 below. The ***Decode*** view shows four fields calculated by EtherPeek at the top of the window, then shows each of the layers of the packet. In this view, the + plus signs in the margin indicate that the details for each part of the packet are hidden under their headings. EtherPeek displays the packet contents in the same order in which it appears in the packet: Ethernet header, IP header, then the TCP and the HTTP payload.



```
⊟ Packet Info
     Flags:                      0x00
     Status:                     0x10
     Packet Length:              410
     Timestamp:                  11:03:40.197920000 09/30/1997
⊞ Ethernet:         Dst=Cisco:5D:10:46 Src=Apple:3C:00:46 Protocol=0x0800
⊞ IP:               Ver=4 HLen=5 TLen=392 ID=33693 Offset=0 TTL=255 Proto=
⊞ TCP:              Src=swc-xds Dst=http Seq=203070977 Ack=1289344001 Offs
⊞ HTTP:             HTTP Command=GET URI=/av/gifs/av_logo.gif HTTP Versio
⊞ FCS:              FCS=0x84AF746E Calculated
```

```
0000: 00 00 0C 5D 10 46 08 00 07 3C 00 46 08 00 45 00  ...].F...<.F..E.
0016: 01 88 83 9D 40 00 FF 06 EA E4 C0 D8 7C 4E CC 7B  ....@.......|N.{
0032: 02 4B 08 52 00 50 0C 1A 9E 01 4C D9 D4 01 50 18  .K.R.P....L...P.
0048: 44 70 C6 4C 00 00 47 45 54 20 2F 61 76 2F 67 69  Dp.L..GET /av/gi
0064: 66 73 2F 61 76 5F 6C 6F 67 6F 2E 67 69 66 20 48  fs/av_logo.gif H
0080: 54 54 50 2F 31 2E 30 0D 0A 43 6F 6E 6E 65 63 74  TTP/1.0..Connect
0096: 69 6F 6E 3A 20 4B 65 65 70 2D 41 6C 69 76 65 0D  ion: Keep-Alive.
```

Figure A.2 EtherPeek decode of an HTTP packet (collapsed view)

What is a protocol?

A *protocol* is a set of rules governing communications.

Networking protocols specify what types of data can be sent, how each type of message will be identified, what actions can or must be taken by participants in the conversation, precisely where in the packet header or trailer each type of required information will be placed, and more.

EtherPeek understands protocols by examining the contents of the packets those protocols create. Each protocol has a variety of forms of headers and sometimes trailers that it uses, either to transmit data for other applications, or to transmit control and information messages that support its own functionality. The exact form of these wrappers or headers tends to be unique, not only among functions within a given protocol, but also across protocols.

EtherPeek essentially acts as if it were a combination of *all* of the various protocol stacks: AppleTalk, TCP/IP, DECnet, NetWare or others. Instead of acting on the messages, EtherPeek decodes the packets in order to identify as precisely as possible what function each packet serves within its protocol. WildPackets' ProtoSpecs™ technology refers to these various functions as *sub-protocols*. In other, more formal views of networking, TCP and UDP may be seen as protocols in their own right, HTTP may be seen as an application running under TCP/IP, and so on. ProtoSpecs side-steps all these largely formal naming conventions and simply treat all of them—UDP, TCP, and HTTP—as sub-protocols of IP. ProtoSpecs does preserve the correct functional relationships among the various sub-protocols, however. HTTP, for instance, is shown as a sub-protocol of TCP which is itself a sub-protocol of IP.

Ethernet protocols and ProtoSpecs™

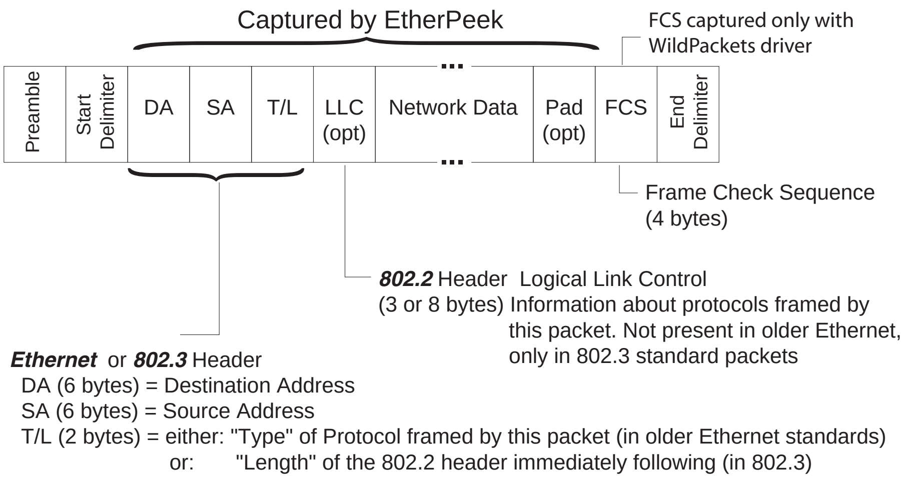
Ethernet is the collective name for a variety of closely related network standards. As a network standard, each version of Ethernet includes specifications for the physical network layer: how the signals will be sent and received. Protocols like IP or NetWare, in contrast, define communications without reference to the physical transport medium. EtherPeek is only interested in that aspect of each version of Ethernet that is reflected in the construction of Ethernet packets or network frames. It treats the various Ethernet standards as if they were protocols, discriminating among them based on the unique form each one gives to packet headers and trailers.

ProtoSpecs nests protocols, one under the other, in a hierarchy from broadest to most specific. ProtoSpecs places the various Ethernet standards at the top of the hierarchy of protocols, as they are certainly the broadest. The 802.3 Ethernet standard and the older Ethernet Type 2 standard form parallel hierarchies in ProtoSpecs. Many protocol stacks are still written for the older Ethernet Type 2. Most IP implementations, for example, use the older standard. The net effect is that protocols may appear twice in the ProtoSpecs hierarchy: once under the older Ethernet Type 2 and again under the newer 802.3 standard.

Ethernet frames and packet headers

This section describes the various types of Ethernet packet headers and the clues they contain to the protocols found in the network data which they frame. Ethernet packets use a format like that shown in Figure A.3.

As Figure A.3 shows, EtherPeek captures all of the packet except the hardware preamble, packet start delimiter and end delimiter bytes. EtherPeek captures FCS bytes only from adapters that are under the control of a WildPackets driver. The majority of supported interfaces operate under Windows NDIS drivers which do not pass FCS bytes to higher layers. EtherPeek calculates the FCS bytes for packets captured on these adapters. The **Packet Decode** window shows FCS bytes as *Calculated* when these bytes were not captured directly from the network. Please see “Ethernet interface requirements” on page 10 for details about WildPackets drivers.



802.2 Header Logical Link Control
(3 or 8 bytes) Information about protocols framed by this packet. Not present in older Ethernet, only in 802.3 standard packets

Ethernet or ***802.3*** Header
DA (6 bytes) = Destination Address
SA (6 bytes) = Source Address
T/L (2 bytes) = either: "Type" of Protocol framed by this packet (in older Ethernet standards)
or: "Length" of the 802.2 header immediately following (in 802.3)

Figure A.3 Ethernet packet format

Ethernet packets are sometimes called *network frames* because they add both a header and a trailer to the packets, thus framing the network data being transmitted. The older Ethernet standards and the newer 802.3 standard are largely the same. Both types begin with a 6-byte destination (MAC) address followed by a 6-byte source (MAC) address, and both add a 4-byte frame check sequence (FCS) to the end of the packet to help detect any errors introduced during packet transmission.

Note: The MAC (Media Access Control) address is the physical address of a particular Network Interface Card (NIC) or other Ethernet device. For more information on addressing, please see Appendix B, “Addresses and Names” on page A-13.

The difference between the two standards is in how they describe the contents of the packet itself. The older standard uses a 2-byte hexadecimal number to denote the protocol Type of the network data framed by the packet. This information is placed in a 2-byte field at offset 12, immediately following the source address.

The 802.3 standard takes advantage of further work done by the IEEE in establishing more powerful tools for describing the contents and function of Ethernet packets. This work resulted in the 802.2 standard for Logical Link Control and created a new part of an Ethernet packet known as the LLC header. The 802.2 standard permits this field to be of varying length (3 bytes or 8 bytes), so 802.3 packets use the old Type field at offset 12 to describe the length of this new header.

Only Ethernet packets following the 802.3 standard can take advantage of the newer 802.2 specifications. Their two basic forms are described below. Please see “802.2 headers” on page A-9.

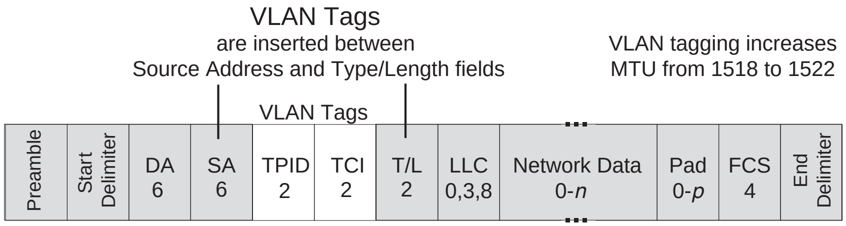
Frame length

The standard Ethernet frame (Figure A.3) is from 64 to 1518 bytes in length, excluding the preamble, start delimiter, and end delimiter. The maximum transmission unit (MTU) is sometimes expressed as 1500 bytes, but this excludes the destination and source address fields (six bytes each), the length/type field (two bytes), and the four bytes of FCS.

Packets smaller than the 64 byte minimum are described as runt packets. Those larger than 1518 bytes (with the exceptions noted below) are described as oversized. The vast majority of Ethernet implementations in the field today will reject packets outside the 64-1518 byte range.

As Ethernet data rates increased, vendors began to consider using Ethernet beyond the LAN in metropolitan area networks (MANs) and wide area networks (WANs). The conditions in these new environments prompted two changes in the Ethernet standards, each of which permitted longer packets.

The first change was the adoption of an optional set of fields in the Ethernet header to accommodate virtual local area networks (VLANs). This method is covered in the IEEE 802.1Q and 802.3ac standards. The two fields, shown in Figure A.4, increase the Ethernet MTU from 1518 to 1522 bytes for protocol stacks that support the new option. Packets that conform to this standard are sometimes referred to as Baby Jumbo or Baby Giant frames.



802.3ac VLAN Tagged Header
TPID (2 bytes) = Tag Protocol Identifier (always = 0x8100)
TCI (2 bytes) = Tag Control Information:

Priority	3 bits
CFI	1 bit (always = 0)
VLAN ID	12 bits

Figure A.4 VLAN tagged 802.3ac Ethernet packet, showing VLAN headers

So-called Jumbo frames have a theoretical MTU of 9180 bytes. This is the largest packet that can be verified using a four byte FCS. The actual maximums vary from one vendor to another, with many vendors choosing the intermediate size of 4470 bytes, which is compatible with FDDI/IP. On networks with a very high data rate (> 1 Gbps), the use of Jumbo packets can reduce overhead and improve throughput.

As a practical matter, the largest packets found on any network path tend to conform to the smallest MTU permitted by any router or switch on that network path. Even on Ethernet backbone segments that are “Jumbo clean” (that is, those on which all directly connected devices are able to send and receive Jumbo frames), it is not unusual to find very few, or even *no* frames larger than 1518 bytes.

802.2 headers

The 802.2 Header, usually called the LLC (Logical Link Control), contains information about the protocol type of the packet. These 802.2 headers are either 3 bytes or 8 bytes long. The first section of the LLC header is 3 bytes long and contains two LSAP values and one LSAP command. These LSAP values can either contain information about the protocol of the packet, or they can point to the optional 5-byte SNAP section that follows. If they point to the SNAP section of the header then the protocol is described by this 5-byte Protocol Discriminator or SNAP ID.

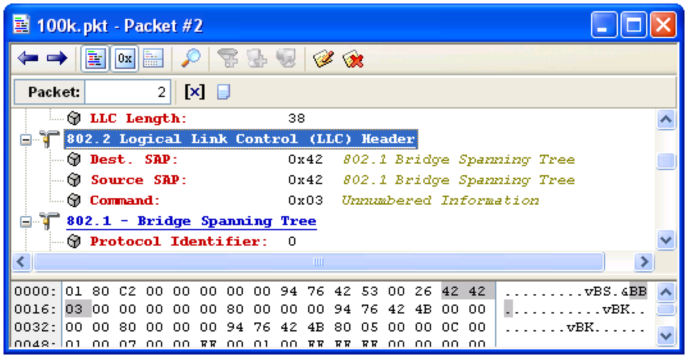
The LSAP Values and the SNAP IDs are described in the next two sections.

802.2 LSAP values

In EtherPeek, the 1-byte protocol type specifications found in the first 3-byte section of the 802.2 LLC header are referred to as 802.2 LSAP values.

The first three bytes of an 802.2 LLC header are as follows:

- The first byte is the Destination Service Access Point (DSAP), which designates a destination protocol.
- The second byte is the Source Service Access Point (SSAP), which designates a source protocol, most often set to the same value as the DSAP.
- The third byte is a control byte that indicates the data format in the packet. This byte is ignored by most protocols (except SNA).



```
LLC Length:           38
802.2 Logical Link Control (LLC) Header
  Dest. SAP:          0x42  802.1 Bridge Spanning Tree
  Source SAP:         0x42  802.1 Bridge Spanning Tree
  Command:            0x03  Unnumbered Information
802.1 - Bridge Spanning Tree
  Protocol Identifier: 0
```

```
0000: 01 80 C2 00 00 00 00 00 94 76 42 53 00 26 42 42  .........vBS.&BB
0016: 03 00 00 00 00 00 80 00 00 00 94 76 42 4B 00 00  ...........vBK..
0032: 00 00 80 00 00 00 94 76 42 4B 80 05 00 00 0C 00  .......vBK......
0048: 01 00 07 00 00 FF 00 01 00 FF FF FF 00 00 00 00
```

Figure A.5 802.2 LSAP values in the 802.2 LLC Header

The DSAP and SSAP fields are referred to *collectively* as the LSAP (Link Layer Service Access Point).

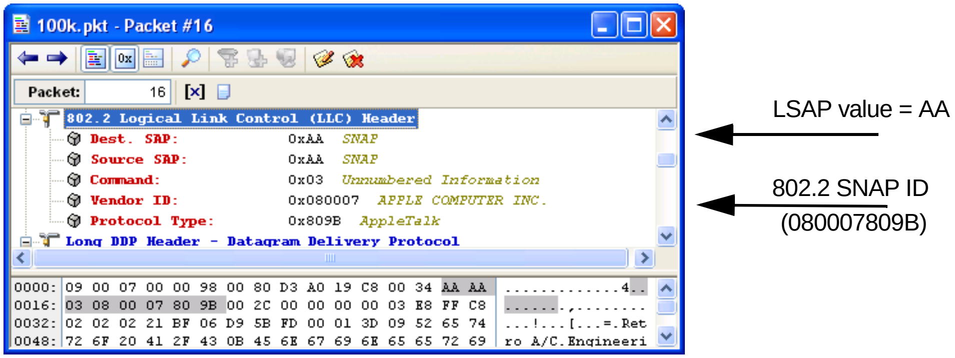
Note: The 1-byte hexadecimal number in these fields can be used to identify the specific 802.2 LSAP protocol in a filter. For example, XNS uses this LSAP value: 0x80.

802.2 SNAP ids

When both the DSAP and SSAP are set to 0xAA, the type is interpreted as a protocol not defined by IEEE and the LSAP is referred to as SubNetwork Access Protocol (SNAP).

In SNAP, the 5 bytes that follow the DSAP, SSAP, and control byte are called the *Protocol Discriminator.*

In EtherPeek, protocol type specifications found in this optional 5-byte SNAP section of the 802.2 header are referred to as 802.2 SNAP IDs. The following figure shows an example of an 802.2 header with a SNAP ID.



```
802.2 Logical Link Control (LLC) Header
    Dest. SAP:          0xAA   SNAP
    Source SAP:         0xAA   SNAP
    Command:            0x03   Unnumbered Information
    Vendor ID:          0x080007  APPLE COMPUTER INC.
    Protocol Type:      0x809B  AppleTalk
Long DDP Header - Datagram Delivery Protocol

0000: 09 00 07 00 00 98 00 80 D3 A0 19 C8 00 34 AA AA  ..............4..
0016: 03 08 00 07 80 9B 00 2C 00 00 00 00 03 E8 FF C8  .......,.........
0032: 02 02 02 21 BF 06 D9 5B FD 00 01 3D 09 52 65 74  ...!...[...=.Ret
0048: 72 6F 20 41 2F 43 0B 45 6E 67 69 6E 65 65 72 69  ro A/C.Engineeri
```

Figure A.6 802.2 Header with SNAP ID

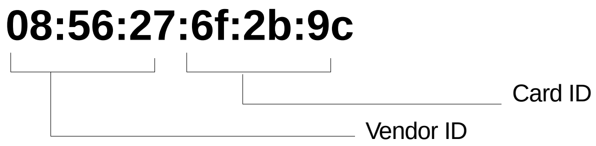
Addresses and Names

The basic concept of Ethernet networking is that packets are given destination addresses by senders, and those addresses are read and recognized by the appropriate receivers. Devices on the network check every packet, but fully process only those packets addressed either to themselves or to some group to which the device belongs.

EtherPeek recognizes three types of addresses: physical addresses, logical addresses, and symbolic names assigned to either of these.

Physical addresses

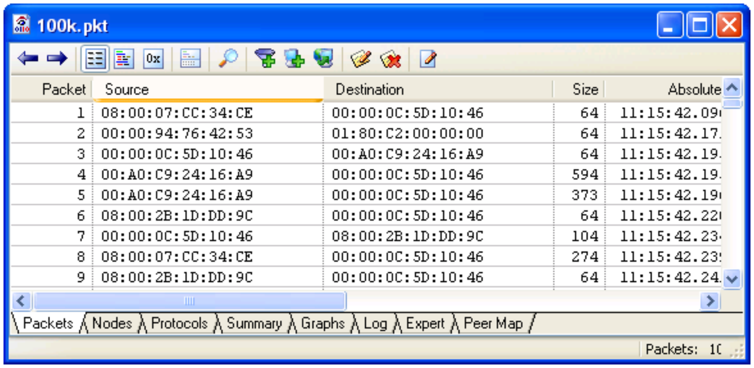
A physical address is the hardware-level address used by the Ethernet interface to communicate on the network. Every device must have a unique physical address. This is often referred to as its MAC (Media Access Control) address. An Ethernet physical address is six bytes long and consists of six hexadecimal numbers, usually separated by colon characters (:). For example:



Typically, a hardware manufacturer obtains a block of physical address numbers from the IEEE and assigns a unique physical address to each card it builds. The vendor block of addresses is designated by the first three bytes of the six-byte physical Ethernet address. In this way, Ethernet physical addresses are generally distinct from each other, although some networks and protocols will override this built-in mechanism with one of their own.

Note: A current list of vendor IDs is included in the default EtherPeek Name Table.

The following figure shows captured packets that use physical addresses to represent the source and destination:



Packet	Source	Destination	Size	Absolute
1	08:00:07:CC:34:CE	00:00:0C:5D:10:46	64	11:15:42.09
2	00:00:94:76:42:53	01:80:C2:00:00:00	64	11:15:42.17.
3	00:00:0C:5D:10:46	00:A0:C9:24:16:A9	64	11:15:42.19.
4	00:A0:C9:24:16:A9	00:00:0C:5D:10:46	594	11:15:42.19.
5	00:A0:C9:24:16:A9	00:00:0C:5D:10:46	373	11:15:42.19
6	08:00:2B:1D:DD:9C	00:00:0C:5D:10:46	64	11:15:42.22
7	00:00:0C:5D:10:46	08:00:2B:1D:DD:9C	104	11:15:42.23
8	08:00:07:CC:34:CE	00:00:0C:5D:10:46	274	11:15:42.23
9	08:00:2B:1D:DD:9C	00:00:0C:5D:10:46	64	11:15:42.24.

Figure B.1 Physical addresses displayed in a Packet File window

Logical addresses

A logical address is a network-layer address that is interpreted by a protocol handler. Logical addresses are used by networking software to allow packets to be independent of the physical connection of the network, that is, to work with different network topologies and types of media. Each type of protocol has a different kind of logical address, for example:

- an IP address (IPv4) consists of four decimal numbers separated by period (.) characters, for example:

  `130.57.64.11`

- an AppleTalk address consists of two decimal numbers separated by a period (.), for example:

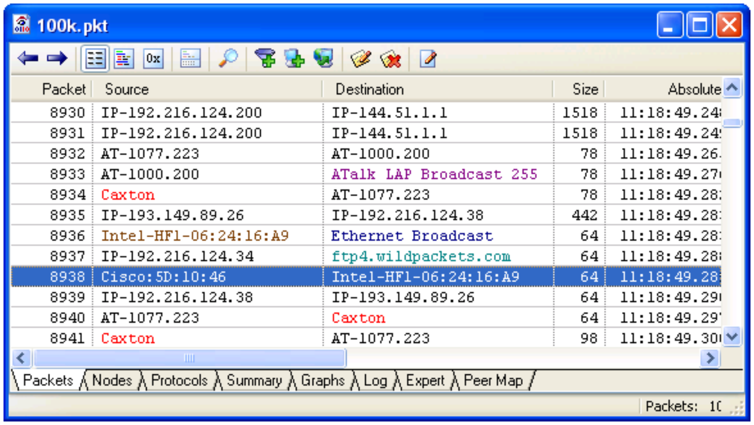
  `2010.42`

  `368.12`

Depending on the type of protocol in a packet (such as IP or AppleTalk), a packet may also specify source and destination logical address information, either as extensions to the physical addresses or as alternatives to them.

For example, in sending a packet to a different network, the higher-level, logical destination address might be for the computer on that network to which you are sending the packet, while the lower-level, physical address might be the physical address of an inter-network device, like a router, that connects the two networks and is responsible for forwarding the packet to the ultimate destination.

The following figure shows captured packets identified by logical addresses under two protocols: AppleTalk (two decimal numbers, separated by a period) and IP (four decimal numbers from 0 to 255 separated by a period). It also shows symbolic names substituted for IP addresses (*www0.wildpackets.com* and *ftp4.wildpackets.com*) and for an AppleTalk address (*Caxton*).



Packet	Source	Destination	Size	Absolute
8930	IP-192.216.124.200	IP-144.51.1.1	1518	11:18:49.24
8931	IP-192.216.124.200	IP-144.51.1.1	1518	11:18:49.24
8932	AT-1077.223	AT-1000.200	78	11:18:49.26
8933	AT-1000.200	ATalk LAP Broadcast 255	78	11:18:49.27
8934	Caxton	AT-1077.223	78	11:18:49.28
8935	IP-193.149.89.26	IP-192.216.124.38	442	11:18:49.28
8936	Intel-HF1-06:24:16:A9	Ethernet Broadcast	64	11:18:49.28
8937	IP-192.216.124.34	ftp4.wildpackets.com	64	11:18:49.28
8938	Cisco:5D:10:46	Intel-HF1-06:24:16:A9	64	11:18:49.28
8939	IP-192.216.124.38	IP-193.149.89.26	64	11:18:49.29
8940	AT-1077.223	Caxton	64	11:18:49.29
8941	Caxton	AT-1077.223	98	11:18:49.30

Figure B.2 Logical AppleTalk and IP addresses and symbolic names

Symbolic names

The strings of numbers typically used to designate physical and logical addresses are perfect for machines, but awkward for human beings to remember and use. Symbolic names stand in for either physical or logical addresses. The domain names of the Internet are an example of symbolic names. The relationship between the symbolic names and the logical addresses to which they refer is handled by DNS (Domain Name Services) in IP (Internet Protocol). EtherPeek takes advantage of these services to allow you to resolve IP names and addresses either passively in the background or actively for any highlighted packets.

In addition, EtherPeek allows you to identify devices by symbolic names of your own by creating a Name Table that associates the names you wish to use with their corresponding addresses.

To use symbolic names that are unique to your site, you must first create Name Table entries in EtherPeek and then instruct EtherPeek to use names instead of addresses when names are available.

To learn more about correlating names and addresses, see Chapter 7, “Name Table” on page 127.

Other classes of addresses

When one says “address,” one typically thinks of a particular workstation or device on the network, but there are other types of addresses equally important in networking. To send information to everyone, you need a *broadcast* address. To send it to some but not all, a *multicast* address is useful. If machines are to converse with more than one partner at a time, the protocol needs to define some way of distinguishing among services or among specific conversations. *Ports* and *Sockets* are used for these functions. Each of these is discussed in more detail below.

Broadcast and multicast addresses

It is often useful to send the same information to more than one device, or even to all devices on a network or group of networks. To facilitate this, the hardware and the protocol stacks designed to run on the IEEE 802 family of networks can tell devices to listen, not only for packets addressed to that particular device, but also for packets whose destination is a reserved broadcast or multicast address.

Broadcast packets are processed by every device on the originating network segment and on any other network segment to which the packet can be forwarded. Because broadcast packets work in this way, most routers are set up to refuse to forward broadcast packets. Without that provision, networks could easily be flooded by careless broadcasting.

An alternative to broadcasting is multicasting. Each protocol or network standard reserves certain addresses as multicast addresses. Devices may then choose to listen in for traffic addressed to one or more of these multicast addresses. They capture and process only the packets addressed to the particular multicast address(es) for which they are listening. This permits the creation of elective groups of devices, even across network boundaries, without adding anything to the packet processing load of machines not interested in the multicasts. Internet routers, for example, use multicast addresses to exchange routing information.

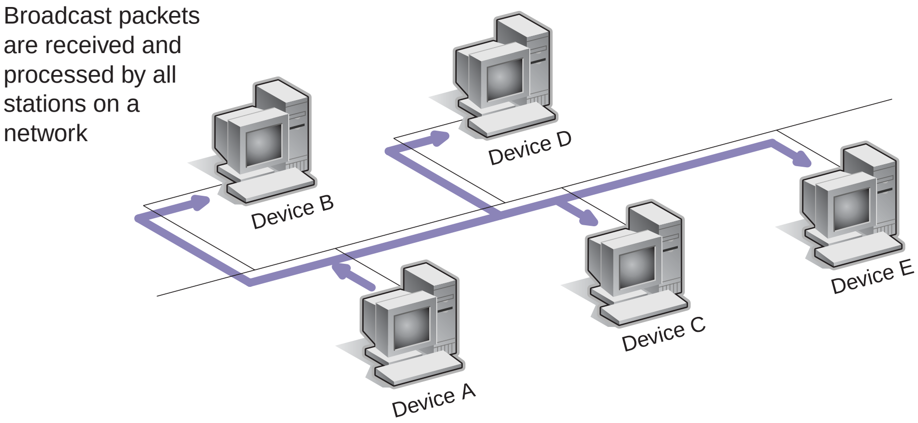


Figure B.3 Broadcast packets are processed by all nodes on the network

Hardware Broadcast Address. The following destination physical address is the Ethernet Broadcast address:

`FF:FF:FF:FF:FF:FF`

A packet with this destination address will be accepted by all devices on the network.

Some protocol types have logical Broadcast addresses. When an address space is subnetted, the last (highest number) address is typically reserved for broadcasts. For example:

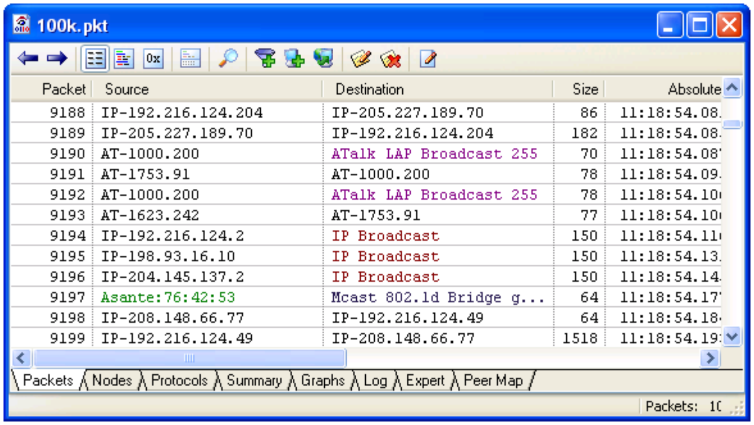
IP Broadcast Addresses typically uses 255 as the host portion of the address; for example:

`130.57.255.255`

AppleTalk Broadcast Addresses use 255 as the node portion of the address:

`200.255`

While conceptually very powerful, broadcast packets can be very expensive in terms of network resources. Every single node on the network must spend the time and memory to receive and process a broadcast packet, even if the packet has no meaning or value for that node.



Packet	Source	Destination	Size	Absolute
9188	IP-192.216.124.204	IP-205.227.189.70	86	11:18:54.08
9189	IP-205.227.189.70	IP-192.216.124.204	182	11:18:54.08
9190	AT-1000.200	ATalk LAP Broadcast 255	70	11:18:54.08
9191	AT-1753.91	AT-1000.200	78	11:18:54.09
9192	AT-1000.200	ATalk LAP Broadcast 255	78	11:18:54.10
9193	AT-1623.242	AT-1753.91	77	11:18:54.10
9194	IP-192.216.124.2	IP Broadcast	150	11:18:54.11
9195	IP-198.93.16.10	IP Broadcast	150	11:18:54.13
9196	IP-204.145.137.2	IP Broadcast	150	11:18:54.14
9197	Asante:76:42:53	Mcast 802.1d Bridge g...	64	11:18:54.17
9198	IP-208.148.66.77	IP-192.216.124.49	64	11:18:54.18
9199	IP-192.216.124.49	IP-208.148.66.77	1518	11:18:54.19

Figure B.4 AppleTalk broadcast and multicast packets

Multicast Address. In Ethernet, addresses in which the first byte of the address is an odd-number are reserved for multicasting. In IPv4, all of the Class D addresses have been reserved for multicasting purposes. That is, all the addresses between 224.0.0.0 and 239.255.255.255 are associated with some form of multicasting. Multicasting under AppleTalk is handled by an AppleTalk router which associates hardware multicast addresses with addresses in an AppleTalk *Zone*.

Ports and sockets

Network servers, and even workstations, need to be able to provide a variety of services to clients and peers on the network. To help manage these various functions, protocol designers created the idea of logical *ports* to which requests for particular services could be addressed.

Ports and *sockets* have slightly different meanings in some protocols. What is called a port in TCP/UDP is essentially the same as what is called a socket in IPX, for example. EtherPeek treats the two as equivalent. ProtoSpecs uses port assignments and socket information to deduce the type of traffic contained in packets.

Product Support and Maintenance

Providing quality technical support to our customers is very important to us! Our online technical support form provides our customers with a standard format for reporting product issues and comments, while giving our staff the information required to deliver expeditious responses to specific issues and product feature requests.

EtherPeek is available with two levels of maintenance. Standard Maintenance is available for twelve or twenty-four months and can be purchased with your product on our Web site. Premium Maintenance is available for twelve months and can be purchased by contacting sales@wildpackets.com.

Standard Maintenance (available for 12 or 24 months)

- Priority technical support via telephone, electronic mail, fax
- Automatic notification of and on-line access to product updates and upgrades as available
- Password access to the maintenance area at wildpackets.com
- Free documentation updates
- Online technical reference materials
- Free utility software
- Qualification for pre-release product testing

Premium Maintenance

- Additional 12 months Standard Maintenance benefits
- One Remote Trace File Analysis (next business day response)
- 1 class seat in a WildPackets Academy 3-day class
- 1 companion seat at 50% discount in any WildPackets Academy 2-day class

Technical support

If you have a problem with EtherPeek, please fill out the web-based technical support form located at http://www.wildpackets.com/support/contact, or call (800) 466-2447.

Resources

WildPackets Academy

WildPackets Academy offers a structured educational curriculum centered on practical applications of protocol analysis techniques using EtherPeek and AiroPeek. Introductory courses in the basic concepts of protocol analysis provide the foundation for a full range of advanced offerings in specialized topics. See http://www.wildpackets.com/services for a full course catalog, current public course scheduling, web-delivered courses, and on-site course delivery information.

Network Analysis Courses

WP-100	Foundations of Network Protocol Analysis
WP-101	Network Troubleshooting Methods Using EtherPeek
WP-102	Emerging Ethernet Technologies: VoIP, Full Duplex, Gigabit, and Switching
WP-103	TCP/IP Protocol Analysis
WP-104	Advanced TCP/IP Protocol Analysis
WP-105	AppleTalk, AppleShare IP, and Mac OS X Network Analysis
WP-106	Wireless LAN Administration

Live Online QuickStart e-Seminars

QuickStart e-Seminars are hour-long programs focusing on detailed aspects of using EtherPeek and AiroPeek, led by a WildPackets Academy instructor. See our website at http://www.wildpackets.com for current scheduling information.

T.E.N. Video Workshop

The Technology, Engineering, and Networking Video Workshop is a 5-Session, 14-Module self-paced program covering the major components of protocol analysis. Participants complete each module by working though exercises and submitting answers to a professional instructor at WildPackets Academy. The modules in the T.E.N. program are consistent with the material tested in the NAX certification program. Visit our website at: http://www.wildpackets.com/services/video for more information.

NAX™ Certification

WildPackets Academy provides instruction and testing for the NAX (Network Analysis Expert) Certification. A Network Analysis Expert certificate is confirmation by WildPackets Academy that an individual is fully qualified to perform Ethernet or 802.11 Wireless network protocol analysis.The NAX certification program is completely vendor-neutral and is positioned as an industry-standard method for demonstrating protocol analysis expertise. For complete details, see http://www.nax2000.com.

Consulting Services

WildPackets offers a full spectrum of expert network analysis consulting services, available on-site, online or through remote dial-in service:

- On-Site Consulting
- Performance Baseline and Network Capacity Planning Report
- Infrastructure Design Analysis Services
- Remote Consulting Services

For complete details, see http://www.wildpackets.com/services/consulting.

White papers

WildPackets offers a number of white papers on network management topics, ranging from basic approaches to network monitoring, troubleshooting, and security to switched network management and remote analysis. To obtain copies of these white papers, please visit: http://www.wildpackets.com/support/resources.

Software License Agreement

E

SOFTWARE LICENSE AGREEMENT

Please read this license carefully.

You are purchasing a license to use the WildPackets Software. The Software is owned by and remains the property of WildPackets, Inc., is protected by international copyrights, and is transferred to the original purchaser and any subsequent owner of the Software media for his/her use only according to the license terms set forth below. Opening the packaging and/or using the Software indicates your acceptance of these terms. If you do not agree to all of the terms and conditions herein, return the Software, manuals and any partial or whole copies you have made within thirty days of purchase to the party from whom you purchased it for a refund, subject to our restocking fee.

1. Grant of License:

WildPackets, Inc., (WildPackets), grants the original purchaser (Licensee) the limited rights to possess and use WildPackets Software (Software) and User Manual, on the terms and conditions specifically set out in this License.

2. Term:

This License is effective as of the time Licensee receives the Software, and shall continue in Effect until Licensee ceases all use of the Software and returns or destroys all copies thereof, or until automatically terminated upon the failure of Licensee to comply with any of the terms of this License.

3. Your Agreement:

SINGLE USER LICENSE

The Software is provided under a Single User License. This means that one specific individual ("Licensee") is licensed to install and use the Software on a single hard disk at one time. Neither simultaneous use by more than one individual nor multiple installation of the Software is permitted under the terms of this Single User License. The Licensee may also make ONE BACKUP COPY for the pupose of restoring the Software should he/she experience a loss of the originally installed Software image. If the Software has the capacity for multiple simultaneous capture sessions with the use of multiple network adapters, then the Licensee is permitted to use the Software from their installed platform to conduct multiple simultaneous captures.

If the Software is installed on a networked system, or on a computer connected to a file server or other system that physically allows shared access to the Software, Licensee agrees to prevent use of the Software by more than one user.

MULTIPLE USER LICENSE

If you want to install the Software on a network and provide access for more than one user, you can purchase additional single-user licenses. Each additional single-user license allows one other specific individual to install and use the Software. There is no limit to the number of additional single-user licenses that may be purchased.

Additional single-user licenses are not concurrent-user licenses (that is, each additional single-user license is associated with a specific individual). There is no restriction on the number of additional single-user licensees who may access the Software at any given time. A group of 50 users who want access to a single copy of the Software must purchase 49 additional single-user licenses so the entire work group has access, for instance.

One machine-readable copy of the software may be made for BACK-UP PURPOSES ONLY, and the copy shall display all proprietary notices, and be labeled externally to show that the back-up copy is the property of WildPackets, and that its use is subject to this License. Documentation in whole or part may not be copied.

Licensee may transfer its rights under this License, PROVIDED that the party to whom such rights are transferred agrees to the terms and conditions of this License, and written notice is provided to WildPackets. Upon such transfer, Licensee must transfer or destroy all copies of the Software.

Licensee agrees and certifies that neither the Software nor any software product containing code generated by the Software: (a) is being or will be shipped, transferred or re-exported, directly or indirectly, into any country prohibited by the United States Export Administration Act and the regulations thereunder, or (b) will be used for any purpose prohibited by same.

Except as expressly provided in this License, Licensee may not use, copy disseminate, modify, distribute, sub-license, sell, rent, lease, lend, give, or in any other way transfer, by any means or by any medium, including electronic, the Software. This license is for machine readable object code only, and Licensee will use its best efforts and take all reasonable steps to protect the Software from unauthorized use, copying or dissemination, and will maintain all proprietary notices intact.

4. LIMITED WARRANTY:

WildPackets warrants the Software media to be free of defects in workmanship for a period of ninety days from purchase. During this period, WildPackets will replace at no cost any such media returned to WildPackets, postage prepaid. This service is WildPackets' sole liability under this warranty. LICENSE FEES FOR THE SOFTWARE DO NOT INCLUDE ANY CONSIDERATION FOR ASSUMPTION OF RISK BY WILDPACKETS OR ITS LICENSOR, AND WILDPACKETS AND ITS LICENSOR DISCLAIM ANY AND ALL LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR OPERATION OR INABILITY TO USE THE SOFTWARE, OR ARISING FROM THE NEGLIGENCE OF WILDPACKETS AND ITS LICENSOR, OR THEIR EMPLOYEES, OFFICERS, DIRECTORS, CONSULTANTS, OR DEALERS, EVEN IF ANY OF THESE PARTIES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. FURTHERMORE, LICENSEE INDEMNIFIES AND AGREES TO HOLD WILDPACKETS AND ITS LICENSOR HARMLESS FROM SUCH CLAIMS.

THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE SOFTWARE IS ASSUMED BY THE LICENSEE. THE WARRANTIES EXPRESSED IN THIS LICENSE ARE THE ONLY WARRANTIES MADE BY WILDPACKETS AND ITS LICENSOR, AND ARE IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND OF FITNESS FOR A PARTICULAR PURPOSE. THIS WARRANTY GIVES YOU SPECIFIED LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM JURISDICTION TO JURISDICTION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF WARRANTIES, SO THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.

5. General:

This license is the complete and exclusive statement the agreement of the parties. Should any provision of this License be held to be invalid by any court of competent jurisdiction, that provision will be enforced to the maximum extent permissible, and the remainder of the License shall nonetheless remain in full force and effect. This License shall be controlled by the laws of the State of California, and the United States of America.

6. United States Government Restricted Rights:

Use of the Software by any department, agency or other entity of the United States Federal Government is limited as follows:

(1) The Software and User Manual are provided with RESTRICTED RIGHTS, and are trade secrets of WildPackets for all purposes of the Freedom of Information Act.

(2) Use, duplication or disclosure is subject to restrictions set forth in subparagraph (c)(I)(ii) of the Rights in Technical Data and Computer Software clause at 252.227-7013 or in subparagraphs (c)(1) and (2) of the Commercial Computer Software - Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer: WildPackets, Inc., 1340 Treat Boulevard, Suite 500, Walnut Creek, California, 94597.

F

Contacting WildPackets

During normal business hours, we are available by phone. You can also contact us by fax or email, and we will usually get back to you by the next business day.

Phone	(925) 937-3200
Domestic	(800) 466-2447
FAX	(925) 937-3211
Email	techsupport@wildpackets.com
	sales@wildpackets.com
Web	http://www.wildpackets.com

Our address:

WildPackets, Inc.
1340 Treat Blvd., Suite 500
Walnut Creek, CA 94597

Training and Certification:

WildPackets Academy
(800) 466-2447
http://www.wildpackets.com/services

Index