

# Open Transport 1.3: Technical Information

This document contains technical information about Open Transport 1.3 that may be useful to network managers and administrators. You do not need to read this document in order to use Open Transport.

This article has been archived and is no longer updated by Apple.

## Contents

### Introduction

#### Files added by the Open Transport installer

#### Open Transport AppleTalk features

- Static and dynamic AppleTalk address allocation
- Use of parameter RAM

#### Open Transport TCP/IP features

- DHCP server support
- DHCP address lease support
- Windows NT advanced server support
- BootP support
- Local Hosts file support
- Single Link Multi-homing System Setup
- MacTCP <sup>3</sup>server<sup>2</sup> addressing support
- MacTCP <sup>3</sup>dynamic<sup>2</sup> addressing support
- MacIP support
- PPP connectivity

#### Memory requirements

#### Application compatibility guidelines

#### Performance

---

### Introduction

Open Transport is modern networking and communications system software for the Mac OS. It is based on industry standards and brings a new level of networking connectivity, control, and compatibility to Mac OS computers, while preserving built-in support for easy-to-use networking. Open Transport is supported on system software versions 7.1, 7.1.1, 7.1.2, 7.5.3, 7.5.5 or later. System software version 7.5.3 or later is recommended. No other system software versions are supported.

For more important information about system requirements, compatibility, and known incompatibilities and limitations, read the "About Open Transport 1.2" document.

#### Files added by the Open Transport installer

Open Transport installs the AppleTalk and TCP/IP control panels into the Control Panels folder inside the System Folder of your startup disk. The Open Transport Installer also adds the following files to the Extensions folder inside the System Folder:

Shared Library Manager

Shared Library Manager PPC

These extensions implement a shared library mechanism on 68K and PowerPC Mac OS computers, respectively. Both extensions are required on PowerPC-based computers in order to support both emulated and native shared libraries.

OpenTransportLib

Open Transport Library

These shared libraries implement core Open Transport services on both 68K and PowerPC-based computers. The first library contains the PowerPC implementation and an interface for native PPC applications. The second library contains the built-in Open Transport modules and an interface for emulated 68K applications.

OpenTptAppleTalkLib

Open Tpt AppleTalk Library

These shared libraries implement Open Transport AppleTalk services on both 68K and PowerPC-based computers. The first library contains the PowerPC implementation and an interface for native PPC applications. The second library contains the built-in Open Transport modules and an interface for emulated 68K applications.

OpenTptInternetLib

# TA37352\_Open\_Transport\_Technical\_Information.pdf

## Open Tpt Internet Library

These shared libraries implement Open Transport TCP/IP services on both 68K and PowerPC-based computers. The first library contains the PowerPC implementation and an interface for native PPC applications. The second library contains the built-in Open Transport modules and an interface for emulated 68K applications.

## Open Transport 68K Library

This shared library implements core Open Transport services on 68030- and 68040-based computers. It is installed instead of Open Transport Library if "Open Transport for 68k Macintoshes" is selected in a Custom Install.

## Open Tpt ATalk 68K Library

This shared library implements Open Transport AppleTalk services on 68030- and 68040-based computers. It is installed instead of Open Tpt AppleTalk Library if "Open Transport for 68k Macintoshes" is selected in a Custom Install.

## Open Tpt Inet 68K Library

This shared library implements Open Transport TCP/IP services on 68030- and 68040-based computers. It is installed instead of Open Tpt Internet Library if "Open Transport for 68k Macintoshes" is selected in a Custom Install.

## OpenTransportLib.68K

This shared library implements CFM 68k API for Open Transport. Used by Apple and 3rd party network applications on 68030- and 68040-based computers that use the Code Fragment Manager. It is installed instead of OpenTptInternetLib if "Open Transport for 68k Macintoshes" is selected in a Custom Install.

## Open Transport AppleTalk Features

### Static and dynamic AppleTalk address allocation

Open Transport AppleTalk supports static assigned (manually administered) protocol addresses as well as the dynamic addressing provided by AppleTalk Address Resolution Protocol (AARP). Static addressing allows AppleTalk nodes to be managed using the protocol address as a unique and stable identifier.

It is important that all nodes on each individual AppleTalk subnet (a given cable segment assigned a unique network number or network number range) be administered consistently (either all with dynamic addressing or all with pre-assigned static addresses). This avoids a potential conflict when a new dynamic node acquires an address assigned to manually-addressed node that is not connected or is off line. Administrators can enforce the addressing policy for a subnet by locking the addressing mode. As a precaution, Open Transport AppleTalk checks for duplicate protocol addresses on the LAN even when static addressing is configured.

### Use of parameter RAM

Under classic AppleTalk, AppleTalk's on/off state, the selected network interface, the previous network (protocol) address, and the previous AppleTalk zone name are saved in persistent memory (parameter RAM) for reuse at startup. To ensure backward compatibility, this information is still stored and retrieved on systems using Open Transport AppleTalk. However, the following differences are found with Open Transport:

At startup, Open Transport reads the current AppleTalk configuration file to determine if AppleTalk should be turned on or off. This value overrides the value saved in parameter RAM. The user is not explicitly notified when this occurs.

If the network interface specified in the current AppleTalk configuration file is locked and the specified port is not available or cannot be initialized, Open Transport will not automatically switch the port back to LocalTalk. Instead, AppleTalk will remain off. The user sees a dialog box with this information.

## Open Transport TCP/IP features

Open Transport brings a workstation-class implementation of TCP/IP protocols to Mac OS. As with MacTCP, Open Transport TCP/IP is a full 32-bit stack. Open Transport TCP/IP adds support for:

- dynamic path MTU discovery, for more efficient network use in heterogeneous network topologies.
- Dynamic Host Configuration Protocol (DHCP), for centralized IP address configuration management. DHCP is an Internet Engineering Task Force (IETF) standards-track protocol.
- IP multicast, for participation as an Mbone client for applications written to the native Open Transport API.
- simultaneous TCP connections for applications written to the native Open Transport API are limited only by installed memory and processor power, for increased functionality as an Internet or other TCP/IP network server.
- a new, more robust and standards-compliant domain name resolver (a caching stub DNR).
- support for developer access to raw IP services, as well as TCP and UDP.
- Ethernet Version 2.0 and IEEE 802.3 framing, for better interoperability with a wider range of TCP/IP hosts.
- implicit and explicit domain name search paths, for increased control of domain name resolution.

# TA37352\_Open\_Transport\_Technical\_Information.pdf

- use of multiple IP routers with fail-over, for increased robustness in mission-critical applications.
- TCP wildcard source port assignments start at 2048, increasing from that point.

## DHCP server support

Apple's implementation conforms to the current versions of the applicable specification documents (RFCs). To date, Open Transport TCP/IP has been tested with the following DHCP server implementations:

Competitive Automation (<http://www.join.com>)

FTP Software (<http://www.ftp.com>)

Hewlett Packard HP-UX (<http://www.hp.com>)

Microsoft Windows NT Advanced Server (<http://www.microsoft.com>)

Silicon Graphics (<http://www.sgi.com>)

Sun Solaris and SunOS (<http://www.sun.com>)

TGV (<http://www.tgv.com>)

## DHCP address lease support

Open Transport TCP/IP supports DHCP address leases. Open Transport TCP/IP automatically attempts to renew any address lease that reaches its renewal interval (by default, the renewal interval is reached when the lease is halfway completed). The renewal interval may be configured to a different value by making changes to the configuring DHCP server. Renewal is attempted regardless of how many times the lease has already been renewed. Lease rebinding is also supported. Should an interface's IP address lease expire, the interface is closed down. Open Transport TCP/IP does not currently support the DHCP Client ID option or the DHCPINFORM message, nor will it currently try to use the remainder of a previous but unexpired lease upon rebooting.

## Windows NT advanced server support

With Open Transport 1.1 and later, Mac OS clients are interoperable with the Windows NTAS DHCP server on LAN links. However, Mac OS clients cannot acquire configuration information from an NT DHCP server across a dialup (PPP) link because there is not yet an accepted industry standard for DHCP over dialup. The NT implementation is based on proprietary Microsoft extensions. Mac OS clients cannot acquire configuration information nor register with a Microsoft WINS server. WINS is also dependent on Microsoft extensions to TCP/IP (requiring NetBIOS support).

The Internet Engineering Task Force (IETF) is developing a cross-platform industry standard technology for dynamic registration and look-up of IP names through the Dynamic Service Location working group. Apple has no current plans to implement the WINS extensions. Instead, we are fully committed to implementation of the applicable IETF standards as they emerge.

Macintosh clients running versions of Open Transport prior to 1.1 could experience some interoperability problems due to other significant differences between the Microsoft implementation and that of a typical UNIX-based server.

## BootP support

Open Transport 1.1 and later fully supports Bootstrap Protocol (BootP). Versions of Open Transport prior to 1.1 failed to accept a BootP Reply sent to the unicast (subnet broadcast) address, (for example, xxx.xxx.xxx.255. Replies sent to the all-nets broadcast address (for example, 255.255.255.255) were handled properly. Also, earlier versions of Open Transport required that the BootP gateway be zero hops away.

## Local Hosts file support

Open Transport TCP/IP supports a Hosts file that may be used to supplement and/or customize the Domain Name Resolver's initial cache of information. The Hosts file is normally stored in the Preferences folder in the active System Folder. When Open Transport TCP/IP is initialized, it reads the Hosts file (if any). As in MacTCP, the supported Hosts file features follow a subset of the Domain Name System Master File Format (RFC 1035).

Open Transport TCP/IP is more stringent regarding the content and format of the Hosts file than was MacTCP, which permitted violation of the FQDN requirement for <domain-name>. For instance, the format:

```
charlie A 128.1.1.1
```

which was acceptable to the MacTCP DNR, is no longer permitted because of the use of domain search lists in Open Transport/TCP (³charlie² could potentially exist in any or all of the configured domains). To accomplish the same effect, use this format instead:

```
charlie CNAME myhost.mydomain.edu  
myhost.mydomain.edu A 128.1.1.1
```

# TA37352\_Open\_Transport\_Technical\_Information.pdf

This associates the local alias charlie with the fully qualified domain name myhost.mydomain.edu, and resolves it to the address 128.1.1.1. Use of local aliases is limited to CNAME entries; NS and A entries must use fully qualified domain names.

You can create a Hosts file with any text editor or word processor (the Hosts file must be stored in text format). If you use a Hosts file, keep it as short as possible, and include only entries that will be accessed frequently. This reduces the memory required to cache the DNS information and minimizes the need to maintain and update Hosts files as system information changes.

Open Transport TCP/IP automatically uses a Hosts file stored in the Preferences folder of the active System Folder. If no Hosts file is found in the Preferences folder, Open Transport TCP/IP searches the active System Folder for a Hosts file. You can specify a particular Hosts file to use with a specific configuration. For example, one Hosts file might be set up for a user connecting via Ethernet, and another set up for when that user connects via modem.

For additional information about the Hosts file design of the Open Transport Domain Name Resolver, see the document <sup>3</sup>Apple Open Transport Reference Q&A.<sup>2</sup> The latest version of the Q&A is in the Open Transport Extras Folder.

## Single Link Multi-homing System Setup

Open Transport 1.3 introduces Single Link Multi-homing, a mechanism by which Open Transport can support multiple IP addresses on the same hardware interface, at the same time. Synonyms for this feature include IP Aliasing, Secondary IP address Support, IP Masquerading, "Multihoming", IP Multinode support. This is useful for sites like Internet Service Providers (ISPs), that want to give each of their clients a distinct IP address, without requiring separate computers for each address. Web server software packages, or server plug-ins that utilize this feature can offer virtual domain support that supports all web browsers. You will need to contact your application developer to see if they are supporting this new feature of Open Transport. This feature is only available with Open Transport 1.3 and later.

You configure a system to use multiple IP addresses as follows:

1. The TCP/IP Control Panel must be set for manual addressing.
2. You create a text file with the required name "IP Secondary Addresses" and put it into the Preferences folder in the System Folder.

Each line of the IP Secondary Addresses file contains a secondary IP address to be used by the system, and an optional subnet mask and router address for the secondary IP address. If there is no subnet mask entry, then a default subnet mask for the IP address class will be used. If there is no router address entry, then the default router associated with the primary address will be used.

Each secondary address entry must be prefixed by "ip=". Each subnet mask entry must be prefixed by "sm=". Each router address entry must be prefixed by "rt=". Lines preceded by a ; are ignored. An example of the contents of the IP Secondary Addresses file follows.

- 'ip=' for ip address, 'sm=' subnet mask, 'rt=' router address. There should be no commas in this line as shown between entries.

**Note:** no space in 'ip=192.168.22.200'

IP address	Subnet Mask	router addresses
ip=192.168.22.200	sm=255.255.255.0	rt=192.168.20.1
ip=192.168.22.201	-	rt=192.168.20.1
ip=192.168.22.202	-	-

The order of the entries is important. The "rt=" entry must follow the "sm=" entry if used.

When Open Transport activates TCP/IP, the primary address will be obtained from the TCP/IP Control Panel setting. Open Transport then looks for the IP Secondary Addresses file in the Preferences folder to determine if additional addresses should also be configured. If there are duplicate IP address entries in the IP Secondary Addresses file, the duplicate addresses will be ignored. When Open Transport binds a TCP/IP connection, if there is an address conflict with either the primary or any secondary addresses with another host, Open Transport will present an error message using a dialog box and unload Open Transport/TCP from memory. The error dialog will display the conflicting IP address, the hardware address of the conflicting machine and note that your TCP/IP network interface has been shut down.

To resolve the conflict, quit all of the TCP/IP applications on both conflicting machines, reconfigure TCP/IP on one of the machines so there is no longer an address conflict, then relaunch your TCP/IP applications.

## MacTCP "server" addressing support

Open Transport TCP/IP supports both Bootstrap Protocol (BootP) and Reverse Address Resolution Protocol (RARP) configuration methods. MacTCP Server mode addressing was a combination of BootP and RARP. When Server mode was selected, MacTCP used BootP to attempt to acquire an IP address. If that failed, MacTCP tried RARP. Whichever protocol was successful was stored as a preference, and was used first the next time the computer started up. In Open Transport, you must choose BootP or RARP explicitly.

## MacTCP "dynamic" addressing

Open Transport does not support MacTCP "dynamic" addressing. MacTCP dynamic mode addressing was based on an Apple-proprietary extension to TCP/IP protocols. It applied the address negotiation and assignment rules used by the AppleTalk protocols to TCP/IP networks, making it very easy to set up a Macintosh-only standalone TCP/IP network. Use of this dynamic addressing method in other scenarios, however, could create additional work for a network administrator.

The Internet community (IETF) has since developed a multivendor standard for the dynamic assignment of IP addresses, known as Dynamic Host Configuration Protocol (DHCP). Open Transport TCP/IP supports the industry standard DHCP.

## MacIP support

MacIP is a protocol specification developed for carrying TCP/IP traffic on AppleTalk-only networks, originally LocalTalk networks. MacIP is today frequently used with AppleTalk Remote Access Protocol (ARAP) to provide mobile users access to TCP/IP network services.

Use of MacIP typically requires a gateway, which strips off the AppleTalk encapsulation and places the IP packet on the TCP/IP LAN. When packets are sent back to the MacIP end-node, the gateway replaces the AppleTalk encapsulation. MacIP gateway support is most frequently offered as an integrated service within a multiprotocol router. The gateway (router) attaches to both an AppleTalk and a TCP/IP network.

Open Transport supports MacIP in the TCP/IP control panel. Once selected, TCP/IP data is encapsulated in AppleTalk packets, and is sent through the selected network interface.

## PPP connectivity

PPP (Point to Point Protocol) connectivity for Open Transport is currently based STREAMs modules or on the use of third-party software extensions known as "MDEVs." Early versions of some MDEVs may not be compatible with Open Transport. For information about MDEV compatibility, see both parts of the "Open Transport 1.3 ReadMe" document.

STREAMs based PPP solutions are available from Apple and selected 3rd parties. Open Transport/PPP and Apple Remote Access 3.0 both use STREAMs. For more information about PPP see both parts of the "Open Transport 1.3 ReadMe" document.

## Memory requirements

Open Transport provides many new features and capabilities to Mac OS customers and, in general, will require more system memory (RAM) than does classic networking. However, the actual memory requirements of Open Transport vary depending upon the networking services in use at a given time. This is different from classic networking, which allocates memory to networking services and keeps it allocated even after networking services are no longer in use.

Factors contributing to differences in memory requirements include:

- Open Transport provides implementations of networking as both 680x0 and native PowerPC code. PowerPC code is typically larger (but also faster).
- Open Transport provides "mixed-mode" applications support, making it possible for both PowerPC native and 680x0 applications to use native networking on PowerPC-based computers.
- Open Transport includes both the new implementations of networking and the libraries required to provide backward compatibility support for the older AppleTalk and MacTCP programming interfaces.
- Open Transport requires less memory on systems with virtual memory enabled. Classic networking has about the same memory requirements regardless of the VM setting.
- Open Transport is based on the cross-platform standard STREAMs environment, which requires more memory than classic networking.

The difference in memory requirements depends upon the configuration of your computer. Some examples of base memory requirements include:

Computer System	Virtual Memory	Classic AppleTalk & MacTCP	Open Transport
PowerPC-based	On	350K to 450KB	200KB
PowerPC-based	Off	350K to 450KB	up to 1.2MB
680x0-based	Off	350K to 450KB	700KB to 800KB

## Application compatibility guidelines

Apple has defined three levels of interoperability with Open Transport. The first, "Open Transport Compatible," is used to describe network applications originally developed for "classic" AppleTalk or MacTCP programming interfaces that now take advantage of Open Transport Compatibility Services. These applications automatically gain the benefits associated with the Open Transport control panels. However, they will not realize a significant performance increase on Power Macintosh systems, nor can they take advantage of Open Transport's transport-independence capabilities.

"Open Transport Ready" applications have adopted the new Open Transport APIs. They are PowerPC native, in addition to running on 680x0-

# TA37352\_Open\_Transport\_Technical\_Information.pdf

based Macintosh systems. Open Transport-ready applications benefit from the new control panels and may also realize a significant performance boost when running on PowerPC-based computers.

The highest category of interoperability is "Open Transport Enhanced." In addition to adopting the new Open Transport APIs and being PowerPC native, these applications can be dynamically configured to support AppleTalk, TCP/IP, or serial communication.

Applications that rely on undocumented APIs or examine private data structures in AppleTalk or MacTCP may not be fully compatible with Open Transport. Updated versions of these software products will be required for full compatibility.

## **Performance**

Open Transport is designed to take advantage of the PowerPC processor. For maximum performance, however, networking applications must also take advantage of the PowerPC processor, and should adopt the new Open Transport programming interfaces.

In general, current Mac OS networking applications are written for the 680x0 processor and use the "classic" (680x0-based) networking programming interfaces. These applications can still be used with Open Transport, and may perform somewhat better. Networking applications that are PowerPC-native but not Open Transport-ready may yield better performance, but still fall short of the maximum potential performance because they make use of Open Transport backward compatibility rather than its full capabilities.

Performance improvements will be greater with protocols that use larger datagram sizes. For example, TCP/IP users will see greater improvements than AppleTalk users, because AppleTalk has a fixed and limited datagram size. On high-speed datalinks such as fast Ethernet, FDDI, and ATM, the performance of the network interface card (NIC) driver code is also a significant factor.

Overall performance also depends on the amount of RAM available. Larger packet sizes and higher throughput place increased demand on the buffering system of Open Transport. If Open Transport becomes low on memory, throughput decreases to accommodate the limitation.

Published Date: Feb 18, 2012