

# *CanOPI 1.4*

## *Technical Guide*

For the Apple Network Server (AIX 4.1.x)

### **Information Presentation Technologies, Inc.**

994 Mill Street, Suite 200,  
San Luis Obispo, CA 93401  
+1 (805) 541-3000

### **Information Presentation Technologies Europe BV**

Turfpoortstraat 66, 1411 E H Naarden Vesting,  
The Netherlands,  
+31 35 694 9499

### **Information Presentation Technologies Ltd.**

Suite 15, 8 Kings Court, Newmarket,  
Suffolk, UK, CB8 7SG  
+44 (1638) 663999

Manual version 1.7

## Limited Warranty on Media and Replacement

All implied warranties on the media and manuals, including implied warranties of merchantability and fitness for a particular purpose, are limited in duration to ninety (90) days from the date of the original retail purchase of this product.

Even though Information Presentation Technologies, Inc. (IPT) has tested the software and reviewed the documentation, IPT makes no warranty or representation, either express or implied, with respect to the software, its quality, performance, merchantability, or fitness for a particular purpose. As a result, this software is sold "as is," and you, the purchaser, are assuming the entire risk as to its quality and performance.

In no event will IPT be liable for direct, indirect, special, incidental, or consequential damages resulting from any defect in the software or its documentation.

The warranty and remedies set forth above are exclusive and in lieu of all others, oral and written, express or implied.

We are grateful to Frame Technology Corporation for supplying FrameMaker software with which we produced the camera-ready pages of this manual.

©1995-1996 Information Presentation Technologies, Inc.

AppleShare, AppleTalk, LaserWriter, SimpleText, EtherTalk, LocalTalk, and Macintosh are registered trademarks of Apple Computer, Inc.

Mercutio MDEF from Digital Alchemy

Copyright © Ramon M. Felciano 1992-1995, All Rights Reserved

PostScript is a registered trademark of Adobe Systems, Inc.

All other brand and product names are trademarks of their respective owners.

## ***Welcome to CanOPi***

What is CanOPi? .....	Preface-2
Product Components.....	Preface-3
Conventions in this Guide.....	Preface-4
Typeface Conventions.....	Preface-4
Other Conventions.....	Preface-5

## ***Processes***

Controlling Processes .....	Processes-8
Starting Processes.....	Processes-8
Stopping Processes .....	Processes-8
Process Abbreviations.....	Processes-9
How it Works .....	Processes-9
Autostarting Processes .....	Processes-10
Process Control Commands .....	Processes-10
Notifying Clients of File Server Shutdown.....	Processes-11

## ***AppleShare File Server Service***

Changes in Version 1.4.....	AFP-14
Changes in Caching.....	AFP-14
Other Changes .....	AFP-14
Managing CanOPi's AFP File Servers .....	AFP-16
Configuring Your File Servers.....	AFP-16
How it Works.....	AFP-16
Advanced Configuration.....	AFP-17
Advanced Configuration.....	AFP-19
Default Volumes.....	AFP-24
Setting a Volume Password .....	AFP-25
Read-Only Volumes .....	AFP-25
Deleting MacOS Files.....	AFP-25
UNIX Permissions and MacOS Privileges .....	AFP-26
Setting up AppleShare Passwords - Random Number Exchange.....	AFP-27

Automatic Assignment of Finder Information .....	AFP-29
Using CanOPI's AppleShare File Servers.....	AFP-30
Preparing Your MacOS.....	AFP-30
Logging in to a CanOPI File Server.....	AFP-31
Working with AppleShare Volumes .....	AFP-33
Disconnecting from a File Server .....	AFP-34

### ***Print Spooler Service***

Preparing Your MacOS Computer.....	Spooler-35
Managing CanOPI Print Spoolers .....	Spooler-36
PPD files and the LaserWriter 8.x driver .....	Spooler-36
AppleTalk to UNIX Spooler .....	Spooler-39
UNIX to AppleTalk Spooler .....	Spooler-41
Editing the usps_options File .....	Spooler-44
Printing PostScript to a File.....	Spooler-50
Using CanOPI Print Spoolers .....	Spooler-51

### ***CanOPI Files***

Location of CanOPI Files .....	Files-54
CanOPI Files in /ushare/bin.....	Files-54
CanOPI Files in /ushare/etc .....	Files-54
afp.dirs .....	Files-54
afp.servers.....	Files-55
fonts.plus .....	Files-55
lw .....	Files-55
packages .....	Files-55
qcap.....	Files-55
CanOPI Files in /ushare/etc.local .....	Files-56
keys.....	Files-56
passwd .....	Files-56
CanOPI Files in MacOS Users' Home Directories.....	Files-57
afp.dirs .....	Files-57

## CanOPI Files in Directories Published as AppleShare Volumes .. Files-58

.desktop/desktop .....	Files-58
.dirids/cache .....	Files-58
.rsrc .....	Files-58
.volrsrc .....	Files-58

## **CanOPI Commands**

Syntax of CanOPI Commands .....	Commands-60
build_cache.x .....	Commands-61
cached .....	Commands-62
cachevol .....	Commands-63
docman .....	Commands-64
fontqry .....	Commands-65
lwdq.x .....	Commands-66
lwpr .....	Commands-67
lwstatus .....	Commands-68
opistub .....	Commands-69
pso .....	Commands-71
sendattn.x .....	Commands-72
usafpatr .....	Commands-73
usrestart .....	Commands-75
usrm .....	Commands-76
ussetkey .....	Commands-77
usstart .....	Commands-78
usstop .....	Commands-79
vpasswd .....	Commands-80

## **Compatibilities**

Using CanOPI with Aldus FreeHand DocumentsCompatibilities-82
Using CanOPI with ccMail..... Compatibilities-84
Using CanOPI with GroupWise ..... Compatibilities-86
Using CanOPI with EPOCH file migration ..... Compatibilities-87

Using CanOPI with Microsoft Excel Documents created by PC computers ..... Compatibilities-88

Using UNIX to AppleTalk Print Spoolers With Microsoft Windows Compatibilities-89

## ***Appendix A-***

### ***Suggested Reading***

Minimum UNIX Knowledge Prerequisites..... Appendix A-ii

Minimum AppleTalk Knowledge PrerequisitesAppendix A-iii

A Brief Introduction to AppleTalk Networking .... Appendix A-iv

Basic Terminology and Concepts ..... Appendix A-iv

Network Cables and AppleTalk Protocols Appendix A-iv

Internets and Routers ..... Appendix A-v

Seed and Non-Seed Routers ..... Appendix A-v

Router Configuration..... Appendix A-v

Network Numbers and AppleTalk Zones. Appendix A-vi

Node Addresses and NBP ..... Appendix A-vi

AppleTalk Nodes and Entity Types ..... Appendix A-vii

## ***Appendix B -***

### ***Advanced Configuration of uShare***

Does uShare Have to be a Router? ..... Appendix B-ii

What is Your Network's Routing Table?...Appendix B-v

Configuring uShare's Default Interface and ZoneAppendix B-viii

## ***Appendix C-***

### ***CanOPI's File Tree & Printing Paths***

## ***Appendix D-***

### ***CanOPI Related Processes***

Server Processes ..... Appendix D-ii

*Glossary*

*Index*



# ***Welcome to CanOPI***

This preface introduces CanOPI and describes it's server and client applications.

## What is CanOPI?

CanOPI is IPT's integrated software system for storing, retrieving, and printing high-resolution images.

CanOPI enables MacOS clients to maintain a central archive of images. By using CanOPI's **Drag & Drop** feature, users can have low-resolution versions of high-resolution images generated automatically when the high-resolution image is placed on a network file server. Users may then use these low-resolution images as usual in their documents, clients print through CanOPI **print spoolers**.

These print spoolers include an **OPI** filter that interprets OPI comments in client print jobs and automatically substitutes high-resolution versions of the included images. Clients can monitor and control their print jobs with CanOPI's **PrintQMgr** application, and the **AlertDriver** init sends any printer error messages directly to the client originator of the print job.

Each element of the CanOPI system is designed for a multi-user production environment, so user names, passwords, and access rights are honored and easily managed with CanOPI's System Management module called **EasyAdmin**.

This integrated system, through which clients manipulate low-resolution versions of images, results in dramatic reduction of network traffic and of processing overhead on client machines. And the easy access to this system provided by CanOPI's client applications further expands the resources and productivity of your network and users.

## Product Components

IPT's family of services integrate your MacOS and UNIX networks. Some services are enabled separately as indicated (contact your sales representative for details):

### **uShare**

This module provides basic AppleTalk connectivity for your UNIX host. uShare's **file server** service lets your UNIX host appear to your MacOS users as an AppleShare file server. With uShare's **uPrint spooler**, your Apple Network Server and MacOS users can share printers on your network. Your CanOPI administrator configures and controls uShare services with the **EasyAdmin** application for the MacOS.

### **Drag & Drop Low Resolution Placement**

This module allows users to move their high-resolution images to a network file server and have low-resolution OPI For Position Only (FPO) images automatically generated for use in page layouts.

### **OPI Processor**

This module interprets OPI comments in clients' print jobs and automatically substitutes high-resolution versions of included images. Clients access this feature simply by printing through CanOPI print spoolers from applications that support OPI.

### **EasyAdmin**

EasyAdmin is part of IPT's graphical interface for system and network administration. EasyAdmin provides an easy to use, MacOS based set

of tools, that let you configure and control uShare services and manage user accounts and groups.

### **uPrint Print Spooler Service**

uShare's uPrint Print Spooler service lets MacOS and UNIX users share PostScript printers on your AppleTalk and UNIX networks. The MacOS users select uShare spoolers with the Chooser desk accessory, while UNIX users use their familiar print tools and commands.

### **PrintQManager**

PrintQManager is part of IPT's graphical interface for print queue administration. PrintQManager provides an easy to use, MacOS based set of tools, that allows administrative control of CanOPI print spoolers. PrintQManager also allows individual users to control their own jobs in the printing queue, as well as resubmit finished jobs.

## **Conventions in this Guide**

### **Typeface Conventions**

Throughout this guide, several conventions are followed:

- UNIX commands you enter at a command line appear in the **Courier Bold** typeface and are preceded with a pound sign (#) representing the UNIX shell prompt, for example:  

```
# cd /ushare/bin
```
- File names and DOS commands in the text appear in **Courier bold**, for example:  

```
/ushare/etc/mtab
```

- Keys you are prompted to press appear in bold face type, for example: Press the **‘Shift’** key.
- UNIX shell output and the contents of editable files appear in the **Courier bold** typeface.
- The chapter section headings are indicated by **Helvetica Bold** typeface, aligned flush with the left side of the page.
- Each chapter section is split into individual topics, as indicated by the **Helvetica Bold Underline** typeface, at the beginning of each new topic.
- *NOTES*, that include helpful advice, pertinent information about configuration, and warnings, are indicated by the *times italic* typeface.
- Menu names and options are indicated in bold type, for example: Choose the **‘New’** option from the **‘File’** menu.
- Buttons within dialog boxes and the like appear in bold type, for example: When you have selected the desired options in the **‘Print’** dialog box, click **‘OK’**.
- References to other sections in the guide are printed in bold, italic typeface, for example: see the **‘Advanced Settings’** section of this guide.

### **Other Conventions**

- When you're instructed to “edit the **<namedfile>** file...” you are expected to open the named file with a text editor such as **vi** or Open Windows' **Text Editor** program, make the indicated changes, and save the file.

- When you're instructed to “issue the following command at a UNIX command line,” you are expected to be running a C-shell, or the like, on your Apple Network Server and issue the indicated command.

# *Processes*

You provide CanOPI services to your network by controlling processes on your UNIX host and configuring each service for your particular needs. This chapter gives instructions for using UNIX commands to control CanOPI processes. It is generally preferable to perform these administrative tasks by using CanOPI's MacOS GUI "EasyAdmin", as discussed in the "EasyAdmin" chapter of the User Guide.

## Controlling Processes

Each CanOPI service that you have licensed from IPT or your CanOPI reseller has a related process that you must control. This section gives instructions for manually and automatically starting and stopping CanOPI processes.

### Starting Processes

1. Edit the `/ushare/etc/packages` file and uncomment the abbreviation for each process you want to start. For example, to start CanOPI's File Server Spooler and OPI processes, make the `packages` file look like this:

Apple File Service (enabled) → `afp`  
Spooling service (enabled) → `splr`  
OPI service (enabled) → `opi`

Process abbreviations preceded by a pound sign (#) are commented out and are ignored by the `usstart` command. Read "Process Abbreviations" on page **Processes-3** for a description of these process abbreviations.

2. Issue the following command at a UNIX command line:

```
# /ushare/bin/usstart
```

This command starts all CanOPI processes whose abbreviations are uncommented in `/ushare/etc/packages`.

### Stopping Processes

To stop CanOPI processes, issue the following command at a UNIX command line:

```
# /ushare/bin/usstop
```

This command stops all CanOPI processes.

### Process Abbreviations

These abbreviations are used as arguments to the **usstart**, **usstop**, and **usrestart** commands and as entries in `/ushare/etc/packages`.

Abbreviation	Service
afp (AFP)	AppleShare File Server Service
splr	uPrint-Print Spooler Service
opi	Low Resolution file generation

**NOTE** The abbreviations "afp" and "AFP" are substantially different. Performing a **usstop** on "afp" kills the file server process, while performing a **usstop** on "AFP" kills the file server process and all existing connections as well.

### How it Works

The **usstart**, **usstop**, and **usrestart** commands are shell scripts. The **usstart** script (1) loads CanOPI device drivers if necessary, (2) starts CanOPI's essential processes (nbpd, atnbpd, and epd) if necessary, and (3) starts service processes passed as arguments or listed in `/ushare/etc/packages`. The **usstop** script kills CanOPI's essential processes and service processes passed as arguments (when no arguments are passed, **usstop** kills all service processes). The **usrestart** script runs **usstop** and **usstart** in succession.

## Autostarting Processes

To configure your UNIX host to autostart CanOPI processes each time it is booted, issue the appropriate command listed below:

### **To enable autostarting CanOPI**

1. Issue the following command at a UNIX command line:

```
# /ushare/bin/usautostart on
```

2. Edit the `/ushare/etc/packages` file and uncomment the abbreviation for each process you want to autostart. For example, to automatically start CanOPI's File Server and Spooler processes, make the `packages` file look like this:

Apple File Service (enabled) → `afp`  
Spooling service (enabled) → `splr`  
OPI service (enabled) → `opi`

Process abbreviations preceded by a pound sign (#) are commented out and are ignored by the **usstart** command.

### **To disable autostarting CanOPI**

1. Issue the following command as "root" at a UNIX prompt:

```
/ushare/bin/usautostart off
```

## Process Control Commands

You can manually start, stop, and restart CanOPI processes individually or severally with the **usstart**, **usstop**, and **usrestart** commands in the form:

```
# command <abbrev> <abbrev>...
```

where: `<command>` is "usstart," "usstop," or "usrestart" and `<abbrev>` is a CanOPI process abbreviation.

For example, to start CanOPI's File Server and OPI processes, issue the following command at a UNIX command line:

```
# /ushare/bin/usstart afp opi
```

When you issue the **usstart** command without arguments, it determines what processes to start by reading the `/ushare/etc/packages` file. For example, to start all processes whose abbreviations are un-

commented in the `/ushare/etc/packages` file, issue the following command at a UNIX command line:

```
# /ushare/bin/usstart
```

When you issue the **usstop** command without arguments, it stops all CanOPI processes without consulting the `/ushare/etc/packages` file.

The **usrestart** command executes the **usstop** and **usstart** commands in succession. For example, to restart CanOPI's Print Spooler process, issue the following command at a UNIX command line:

```
# /ushare/bin/usrestart splr
```

When you issue the **usrestart** command without arguments, it stops all CanOPI processes and then starts those processes whose abbreviations are uncommented in the `/ushare/etc/packages` file.

### **Notifying Clients of File Server Shutdown**

To stop CanOPI's AppleShare File Server service and notify logged-in users that the server is shutting down, issue the following command at a UNIX command line:

```
# /ushare/bin/usstop AFP
```

This command immediately stops the File Server process (`afpd`) and sends a special signal to the user log-in processes (`afpsess.x`). The log-in processes are killed after a period of time specified in the `/ushare/etc/shutdowntime` file. The `/ushare/etc/shutdowntime` file is a single-line file containing shutdown time data in the form:

```
<TIME>,<INTERVAL>
```

where: **<TIME>** is the number of minutes until shutdown, and **<INTERVAL>** is the interval (in minutes) between warning messages sent to logged-in clients.



# *AppleShare File Server Service*

CanOPI's AppleShare File Server service lets you define AppleShare (AFP) file servers and publish portions of your UNIX file system as AppleShare volumes. This chapter gives instructions for managing and using CanOPI's AFP file servers from the UNIX command line. It is generally preferable to perform these administrative tasks by using the MacOS GUI "EasyAdmin", as discussed in the "**EasyAdmin**" chapter of the User Guide.

Make sure you start the AppleShare File Server service as described in the "Processes" chapter of this guide.

## Changes in Version 1.4

### Changes in Caching

Version 1.4 of CanOPI includes important changes in the file server service, as compared to 1.3 and earlier versions of CanOPI. Caching has been improved to comply with AFP 2.1. Improvements in overall performance have increased average throughput 25-30% over previous versions of CanOPI.

In previous versions of CanOPI, caching was an option that could be disabled. In version 1.4, caching is always enabled.

In previous versions of CanOPI, the file cache was held in `.rsrc/:#dcache.afp` in files named with a hexadecimal representation of the cached file's inode. In version 1.4, the file cache for a volume is held in that volume's `.dirids/cache` file.

If you mount a read-only volume (one that is not writable by anyone, including root), or a volume over NFS, the cache file will be created in the directory `/ushare/etc.local/caches` on your host machine. Caches in this directory are created with the nomenclature: `cache_0001`, `cache_0002`... etc. To determine which cache belongs to which volume, type:

```
# cat /ushare/bin/etc.local/caches/index
```

The output looks similar to:

```
/nfs/iptoffice:/ushare/etc.local/caches/cache_0001  
/nfs/testing:/ushare/etc.local/caches/cache_0002
```

Each line describes a CanOPI volume path, and the associated cache files, separated by a colon.

You can mount read-only devices such as CD-ROM drives (although file-typing will not be available for such devices).

### Other Changes

The `/ushare/etc/afp.servers` file has two new options in version 1.4: "vers" and "auth," both of which are explained on page **AFP-18**.

The "Publish If Already Published" volumes option has been discontinued in CanOPI 1.4.

One-way and two-way random-number exchange password authentication is supported by the server. For more detail see "Setting Up AppleShare Passwords" on page **AFP-27**.

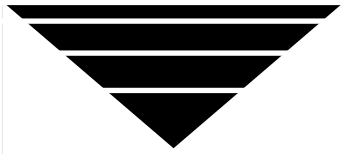
## Managing CanOPI's AFP File Servers

This section describes how to manage CanOPI's AFP file servers by editing configuration files and issuing UNIX commands. It is generally preferable to perform these administrative tasks by using CanOPI's MacOS GUI "EasyAdmin", as discussed in the "**EasyAdmin**" chapter of the User Guide.

**NOTE** Do not edit CanOPI configuration files while "EasyAdmin" is running.

### CanOPI QuickTIP

If you are configuring your CanOPI server to work in conjunction with products like Groupwise, ccMail, EPOCH file migration, or any product that handles files on both the UNIX and MacOS sides, read the section entitled "Compatibility" before proceeding with your configuration.



### Configuring Your File Servers

When you install and start CanOPI, a default file server "CanOPI\_<hostname>" is available to AppleShare clients on your network. This file server allows up to 64 AppleShare clients to simultaneously log in. To change the name and maximum users setting for this default file server, complete the following steps:

1. Edit the line of the `/ushare/etc/afp.servers` file that looks like this:  
`nm=CanOPI_yourhost:mx=64:`
2. Change this line to set your file server's name and maximum number of users:  
`nm=CanOPI1_nevis:mx=128` (128 is the recommended maximum):
3. Restart CanOPI's file server process by issuing the command:  
`# /ushare/bin/usrestart afp`

### How it Works

For each file server defined in `/ushare/etc/afp.servers`, the **start afp** script starts an **afpd** (Apple Filing Protocol Daemon). Each time a client logs in to a server, the server's **afpd** starts an **afpsess.x** (AFP session) process that is owned by the client. File system calls from AppleShare clients are converted to analogous UNIX calls by means of the **afpsess.x**, resulting in a remote file subsystem under UNIX for AppleShare files.

### Advanced Configuration

You can define any number of file servers by making entries in `/ushare/etc/afp.servers` and assign a different set of attributes to each file server by means of parameter assignments in this form:

**param1=val1:param2=val2: ... :paramN=valN:**

For example, to define a file server that is available to only one zone of your network, make an entry like this:

**nm=CanOPII\_lorien:mx=12:zn=EtherTalkZone:**

The complete set of parameters appears in the following table (you must set the **mx** and **nm** parameters; all other parameters are optional).

## File Server Attributes

Attribute	Description
nm	The name registered with NBP that appears in the MacOS Chooser. It must be unique on your network and no more than 32 characters long.
zn	<i>This value is only valid when using EtherTalk Phase I Protocol.</i> By setting this value, the file server is available to clients residing in this zone; default = "*" (all). Only "*" or a single zone name are valid. Specifying a zone name causes the file server to be invisible to all MacOS clients except those residing in the specified zone.
vl	The pathname of the file that contains the list of global volumes (directories that are available to all MacOS clients).
mx	The maximum number of MacOS users that can simultaneously log in to the server; default = 64. It must be set to no more than 128 (as defined by AppleTalk protocol—theoretical limit is ≤ 253 sockets/machine); your CanOPI authorization key may restrict usage to a smaller number of users.
rts	<i>This value should not be changed unless you are requested to do so by an ipt Support Technician.</i> Values 1-8 define the number of 576 byte packets per transaction; values above 16 define the number of bytes per transaction (setting a value of 1 is the same as setting a value of 576).
wt	<i>This value should not be changed unless you are requested to do so by an ipt Support Technician.</i> Values 1-8 define the number of 576 byte packets per transaction; values above 16 define the number of bytes per transaction (setting a value of 1 is the same as setting a value of 576).
vers	The latest AFP version supported (1=1.0, 2=2.0 or 21=2.1); default is 21.
auth	What sort of authentication will be allowed during password entry, and whether or not guest accounts are allowed; Default is "guest,clear,1way,2way". Eliminate any of these fields to disallow the associated function.

## File Server Attributes

Attribute	Description
-S	By appending the -S to a server definition, you may use an alternate afpsess.x file to publish your CanOPI server. This is necessary when using the 3.07 afpsess.x to maintain compatibility with a particular software package (see the Compatibilities section for more detail). The syntax of the entry is: <code>-S /ushare/bin/&lt;afpsessalternate&gt;</code> , where <code>&lt;afpsessalternate&gt;</code> is the afpsess.x file you wish to use in place of the standard daemon.
VERS=	It is only necessary to set this value when it is used in conjunction with the -S entry described above, while using the 3.07 style afpsess.x daemon (usually to maintain compatibility with other software; see the Compatibilities section for more detail). It dictates whether the 3.07 afpsess.x will use AFP version 1.0, AFP version 2.0, or AFP version 2.01. Unless you are using an old MacOS application that cannot communicate using AFP version 2.01, you should set this value to "21". Specifically, the syntax is: <code>VERS=&lt;#&gt;</code> , where <code>&lt;#&gt;</code> is the value "1", "2", or "21", which represent AFP version 1.0, AFP version 2.0, and AFP version 2.01 respectively.

**NOTE** *The Read and Write Transaction Size values should almost never need to be changed. They are included in case you are using an old or very slow router on the network, in which case they may be helpful.*

### Advanced Configuration

You can define any number of volumes by making entries in `/ushare\`  
`/etc/afp.dirs` and `~username/afp.dirs`, and each volume can have a different set of attributes. If, when defining your file server, you used the `vl` argument to specify a global volumes file, edit the specified file instead of `afp.dirs`. All changes to `afp.dirs` take effect the next time a client logs in to a CanOPI file server.

Entries in `afp.dirs` are made in the form:

```
path:name:sort:bits:field #5:field #6:field #7
```

- **path** is the pathname of a UNIX directory to be published. When defining private volumes, you can use short-cut notations: "**~username**" specifies a user's home directory. Pathnames without a preceding slash (/) are relative to the user's home directory.

**NOTE** *Symbolic links and NFS mount points within a published volume are discouraged because they may appear invisible to MacOS clients.*

- **name** is the volume name seen by clients. If you omit the name, CanOPI uses the last 27 characters of the path.
- **sort** is the sorting value. Lower numbers are given to users first. A value of -2 prevents a volume from appearing to clients. Values other than -2 are ignored by the System7 MacOS Chooser which always sorts volume names alphabetically.
- **bits** is the volume attribute bits (read "Volume Bit Settings" on page **AFP-22** for a description of these bits). To specify more than one attribute for a volume, OR together the bits of the attributes (i.e., add the hexadecimal numbers). For example, to turn off text translation and record locking, enter: **0x0011**.
- **field #5**, if specified, is the number of cache records to preallocate. The default is 1000. Users who maintain file systems with many files may want to allocate a larger number of cache records.
- **field #6**, if specified, contains a comma separated list of group names. Only members of these groups will see this volume at login time. If not specified, any user will see the volume in the list. For details on defining a group, and adding a user to that group, see "Users and Groups Admin" on page **EasyAdmin-26** of the User Guide.
- **field #7**, if specified, should contain either "**adf**", "**asf**".  
If "**asf**" is specified, the volume file format for writing is set to "AppleSingle".  
If "**adf**" is specified, the volume file format is set to AppleDouble format.  
Otherwise, it is set to IPT format.

If the files are in "EtherShare" format, CanOPI will be able to read them automatically, but will write any changes in the format specified in this field.

- **field #8**, if specified, contains a comma separated list of user names. Only these users will see this volume at login time. If not specified, any user will see the volume in the list. For details on adding a user, see "Users and Groups Admin" on page **EasyAdmin-26** of the User Guide.

## Volume Bit Settings

Attribute	Description
0x0001	<b>Disable translation of end-of-line characters</b> - MacOS text files have Carriage Return characters at line ends; UNIX text files have Line Feed characters. This attribute disables translation of these characters when a text file is written to or read from the volume (read "Automatic Assignment of Finder Information" on page <b>AFP-29</b> for more information).
0x0002	<b>Do not allow this volume to be mounted. (Apple Sort Value)</b>
0x0004	<b>Disable reading and writing of Finder Information</b> - Setting this attribute prevents CanOPI from reading and writing Finder information to the resource forks of MacOS files on the volume. Because this attribute results in all MacOS files having generic document icons, it is useful only in rare cases and should not be set unless you first contact IPT.
0x0008	<b>Deactivate the Binder</b> - Setting this attribute disables CanOPI's Binder program that reads the first 64 bytes and the file names of un-typed files to determine their AFP Finder information (e.g., type and creator); read "Automatic Assignment of Finder Information" on page <b>AFP-20</b> for more information.
0x0010	<b>Disable Record Locking</b> - Setting this attribute prevents CanOPI from passing MacOS record-locking calls to the UNIX system and is helpful when file systems don't support record locking or when record locking is buggy (e.g. some versions of NFS).
0x0020	<b>Disable Folder Caching</b> - No longer an option in CamOPI versions 1.4 and above. Setting this option has no effect.
0x0040	<b>Mark opened files "busy"</b> - Setting this attribute causes CanOPI to mark opened files as busy with an flock bit for AFP. This bit will only affect files that are opened with write access. Files that are opened as read only will not be affected by this setting.
0x0080	<b>Don't count the offspring of subdirectories</b> - This will speed up opening a folder that has subfolders with LOTS of items in them. But this flag is also incompatible with some applications, e.g. DiskTop.

## Volume Bit Settings

Attribute	Description
0x0100	<b>Support file systems larger than 2GB</b> - The Mac will only support up to 2GB on one volume. With this flag on, CanOPI will manipulate the disk used and disk free sizes reported by the MacOS so that it will work with these large file systems. The MacOS will still report only a maximum of 2GB of space on the volume, but will write to the volume's capacity.

## Default Volumes

Each of your file server's users will have a volume by default, "YourHome(username\_yourhost)." This volume is automatically defined by CanOPI and can be removed only by editing `/usr/share/etc/afp.dirs`.

To remove the default volume for a single user, make the following entry in the `/usr/share/etc/afp.dirs` file:

```
~username/.:HOME:-2
```

This entry defines a sorting value of "-2" for a volume "HOME" that publishes the specified user's home directory. Because a sorting order of "-2" causes a volume to be invisible, this entry effectively removes the default volume for the specified user.

To remove the default volume for all users, make the following entry in the `/usr/share/etc/afp.servers` file:

```
-h
```

You can make similar entries to change the attributes of the default volume; e.g., to disable record locking for the default volume for all users, make the following entry in the `/usr/share/etc/afp.dirs` file:

```
~/.:HOME::0x0010
```

## Setting a Volume Password

In order to control access to published volumes, you can set passwords for public and private volumes. When a MacOS client attempts to mount a volume that has a password, the client must enter the volume password. To set a password for a volume, issue the following command at a UNIX command line:

```
# vpasswd <VOLDIR>
```

Where **<VOLDIR>** is the directory of your UNIX file system published as the AFP volume whose password you want to set. You are prompted to enter a password for the volume. The password you enter is stored in encrypted form in the **.password** file in **<VOLDIR>**.

To remove a volume password, simply remove the **.password** file:

```
# rm <VOLDIR>/ .password
```

## Read-Only Volumes

To make a MacOS volume read-only, it is not enough to set MacOS privileges to disallow writing to the volume's files and folders. You must also make the volume's desktop read-only by issuing the following command at a UNIX command line:

```
# chmod ugo -w <VOLDIR>/ .desktop
```

Where **<VOLDIR>** is the pathname of the UNIX directory that is published as an AFP volume.

## Deleting MacOS Files

To delete a MacOS file from a UNIX command line, you must delete both the file's data fork (residing in **<PATH>**) and the file's resource fork (residing in **<PATH>/ .rsrc**); e.g., to delete a file named **myfile**, issue the following commands at a UNIX command line:

```
# rm <PATH>/myfile
# rm <PATH>/ .rsrc/myfile
```

CanOPI's **usrm** command removes both forks; e.g., to delete a file named **myfile**, issue the following command at a UNIX command line:

```
# /ushare/bin/usrm <PATH>/myfile
```

### UNIX Permissions and MacOS Privileges

When working with files residing on AFP volumes published with CanOPI, you must deal with the differences between UNIX permissions and MacOS (AFP) privileges:

- All UNIX files (including directories) have ownership and permissions, but under AFP only directories (Folders) have privileges
- AFP Folders have three privileges (See Folders, See Files, Make Changes) while UNIX files have three permissions (Read, Write, Execute). Only the AFP privilege "Make Changes" and the UNIX permission "Write" correspond exactly

In order to maintain UNIX permissions while representing proper AFP privileges, CanOPI stores the AFP privileges of a MacOS folder in the header of the folder's resource fork (**./rsrc/<FOLDER>**).

CanOPI does a bitwise AND of the stored AFP privileges and the UNIX permissions, thereby honoring the UNIX permissions if they are more restrictive than the AFP privileges. So, if a UNIX user changes a directory's permissions to be less restrictive, the AFP privileges do not change.

A file residing on a CanOPI-published AFP volume inherits the ownership and permissions of the file's parent directory. This is only true, however, of files that are operated upon by AFP commands. CanOPI does not change the ownership/permissions of a pre-existing file, or a file modified only by a UNIX user, unless an action caused by an AFP client specifically sets the file's attributes. For example if a UNIX user, User1, changes the permissions of a file "**<FILE>**" so that they no longer match the AFP privileges of **<FILE>**'s parent directory, CanOPI honors the new permissions of **<FILE>** and does not change them unless:

- root, User1, or a member of User1's group overwrites **<FILE>**

- root logs in from a MacOS and changes the privileges of <FILE>'s parent directory
- User1 logs in from a MacOS and changes the privileges of <FILE>'s parent directory, and /ushare/bin/uscog is SUID to root and executable by user1

### Setting up AppleShare Passwords - Random Number Exchange

If you want to use password authentication, edit or create the /ushare/etc.local/passwd file (see "passwd" on page). If you don't use password authentication, cleartext passwords will be transmitted over the network when users log in to a CanOPI server.

#### **IMPORTANT!:**

*Make sure that /ushare/etc.local/passwd is owned by root and that its permissions are set to rw----- (i.e. only root can read the file or write to it). You may need to use the "chmod" command to set the file permissions. Use the command:*

```
# ls -l /ushare/etc.local/passwd
```

to see the file's ownership and permissions.

#### **CAUTION:**

*If you are going to use password authentication (either one-way or two-way random-number exchange), you must make entries in this file for all users who are going to be using the file server.*

If you are going to be using one-way random-number exchange, in which the MacOS sends the server a random number and then uses that random number to encrypt the password before transmitting it, entries should be of the form:

```
username:0x0000:
```

Where 'username' is the login name of the user on the filesystem.

If you are going to be using two-way random-number exchange with cleartext passwords in the passwd file, in which the UNIX server and

the MacOS agree on a random number with which the password is encrypted before being transmitted, entries should be of the form:

**username:0x0000:password**

Where ‘username’ is the login name of the user on the filesystem, and ‘password’ is that user’s cleartext password for that filesystem.

### Automatic Assignment of Finder Information

When a UNIX file is created in a directory that is available to MacOS clients through a CanOPI volume, the file lacks a MacOS resource fork (in which CanOPI stores the file's Finder Information, including the type and creator). When the directory is enumerated by a MacOS (i.e., when a MacOS does a getvolinfo request, e.g., when a MacOS closes and opens the folder), CanOPI determines if the file is a text file or a data file. If CanOPI determines that the file is a text file, it creates a resource fork for the file and assigns the type and creator "TEXT-TUNIX." If CanOPI determines that the file is a data file, it uses the Binder program to determine the type and creator, and if the Binder fails to find an appropriate pattern CanOPI assigns the type and creator "DATAUNIX."

UNIX users can assign Finder Information from a UNIX command line by building a resource fork for an untyped file. E.g., if a UNIX file "doc1" resides in a directory "/folder1," issue the following command to build a resource fork:

```
# echo -n <TYPECREA> > /folder1/.rsrc/doc1
```

Where <TYPECREA> is the MacOS type and creator. For example, to make a UNIX file look like a Microsoft Word text file, issue the following command:

```
# echo -n TEXTMSWD > /folder1/.rsrc/doc1
```

**NOTE** *CanOPI performs end-of-line character conversion for a text file only if it has a corresponding resource fork (located in the .rsrc directory) that types the file as "TEXT." One known limitation with this conversion appears when applications by Microsoft (e.g., Word, Excel) create a new text file, they write the data fork before creating the file's resource fork; CanOPI is therefore unable to recognize the file as text until it is too late to perform translation.*

## Using CanOPI's AppleShare File Servers

CanOPI's AppleShare File Server service lets MacOS users access files that reside on a UNIX host by running Apple Computer's AppleShare software. This section gives instructions for:

- preparing your MacOS computer
- logging in to a CanOPI file server
- working with AppleShare volumes
- disconnecting from a file server


### **Preparing Your MacOS**

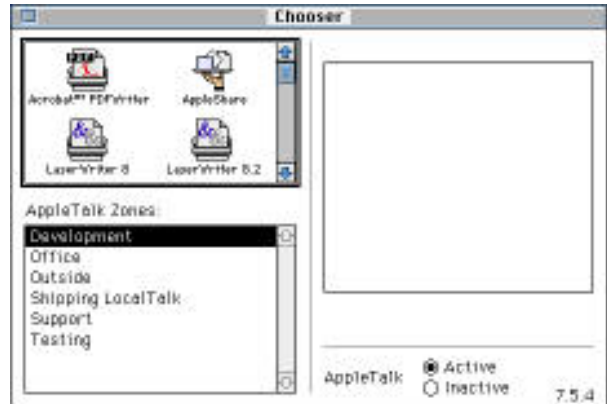
To use CanOPI's File Server service, you must first do the following:

- Use your MacOS System installation disks to install AppleShare Workstation software. This is usually already installed.
- Ask your network administrator for your **Username** and **Password**.
- Ask your network administrator for the name of a CanOPI file server to which you will log in. The default name of CanOPI's file server is "CanOPI\_<hostname>", but your network administrator may have changed the file server's name as described in "Configuring Your File Servers" on page **AFP-16**.
- If your network has AppleTalk zones, ask your administrator for the name of the zone in which the CanOPI file server resides.

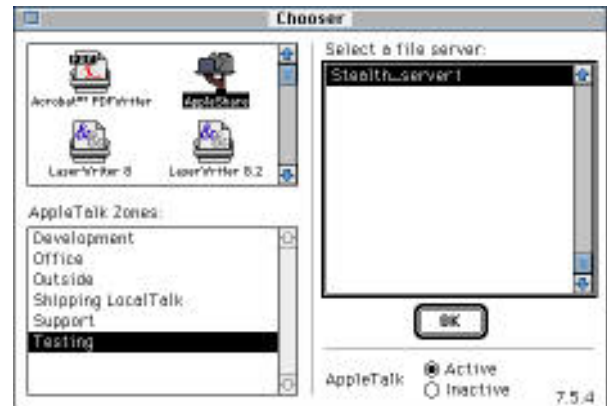
## Logging in to a CanOPI File Server

Use the Chooser desk accessory to log in to a CanOPI file server and mount AppleShare volumes.

1. Select "Chooser" from the  menu. The Chooser window appears, with icons for AppleShare and possibly for printers and other network services. AppleTalk zones may appear:



2. Select the AppleShare icon and a zone (if necessary). A list of available file servers appears:



3. From the list of available file servers, select a CanOPI file server, and click the "OK" button. A login screen appears:



4. Enter your user name and password (your password appears as dots so no one else may read it) and click the "OK" button. A dialog box appears and presents a list of available volumes:



5. Select the volume(s) you want to mount and click the "OK" button. Hold the Command key while clicking to select multiple volumes. The number of volumes you can access at one time depends on how much memory your MacOS has.

You can also click checkboxes of volumes to mount automatically when you start-up your MacOS. If you click a checkbox, two additional choices appear, "Save My Name Only" and "Save My Name and Password". If you select the first of these boxes, entering your password upon system start-up will automatically mount the checked volume. If you select the second of these boxes, the checked volume will automatically mount upon system start-up.

**NOTE** If you select the "Save my Name and Password" option, anyone who starts up your MacOS will have access to your files.

6. Close the Chooser window.

For each volume you selected, an **AppleShare Volume** icon appears on your desktop. Use these icons like any other MacOS disk:



## **Working with AppleShare Volumes**

This section provides a brief introduction to working with AppleShare volumes. Read Apple Computer's *AppleShare User Guide* for detailed instructions. Each AppleShare volume, and each folder on an AppleShare volume, has separate access privileges for its owner, for its group, and for everyone else. Folders on a volume have distinct appearances, depending on the access privileges you have:

- If you own a folder, a black line appears on its tab.



- If you can add things to a folder but can't see its contents, the folder is grey with an arrow above it.



- If you have no privileges, the folder is grey.



By default, when you create a folder on an AppleShare volume, only you, the owner, can See Folders, See Files, and Make Changes. You

can view the access privileges of any folder, and change them if you are the folder's owner, by choosing "Get Privileges..." from the "File" menu (on a System7 MacOS, you must choose "Sharing" from the file menu).

**NOTE** *Only the owner of a folder or volume can permanently rearrange the position of icons or change the default view for the folder or volume.*

### **Disconnecting from a File Server**

To disconnect from a CanOPI file server, drag the icon for each volume you have mounted into the "Trash." You can also disconnect by restarting your MacOS computer.

# ***Print Spooler Service***

CanOPI's Print Spooler service lets MacOS and UNIX users share PostScript printers on your AppleTalk and TCP/IP networks. Because the spooling of print jobs is done by the UNIX host, MacOS users don't suffer the loss of performance associated with direct printing or background printing.

## **Preparing Your MacOS Computer**

To use CanOPI's print spooler service, you must first do the following:

- Make sure you have LaserWriter version 5.2 or higher and Laser-Prep version 6.0 or higher in your MacOS Computer's System Folder.
- Ask your Administrator for the name of the CanOPI print spooler you should use.
- If your network has multiple AppleTalk zones, ask your administrator for the name of the zone in which the spooler resides.

Make sure you start the Print Spooler service as described in the "Processes" chapter.

## Managing CanOPI Print Spoolers

CanOPI print spoolers let UNIX users print to AppleTalk printers and MacOS users print to UNIX printers. By combining these two capabilities (i.e., passing MacOS print jobs to a UNIX queue and back out to an AppleTalk printer), you can make your MacOS users' print jobs accountable. This section describes how to manage CanOPI's print spoolers by editing configuration files and issuing UNIX commands. It is generally preferable to perform these administrative tasks by using CanOPI's MacOS "EasyAdmin" GUI, as discussed in the "EasyAdmin" section. See page **EasyAdmin-23** of the User Guide for more details.

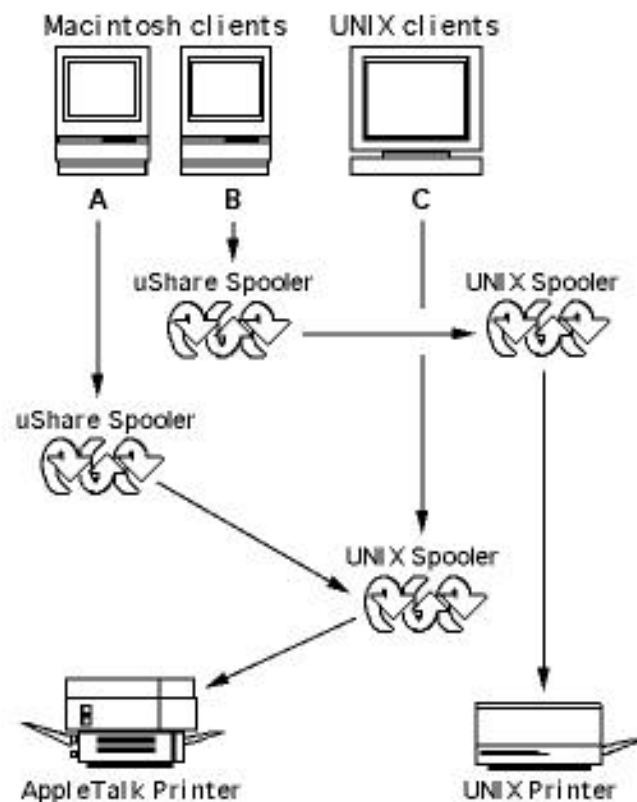
**NOTE** *Do not edit CanOPI configuration files while either the CanOPI Manager or EasyAdmin are running.*

### PPD files and the LaserWriter 8.x driver

Note that for the LaserWriter 8.x, the printer can be queried, giving valuable information to the querying application (e.g. whether the printer is black & white or color, what the status of the print trays are, etc.). To take advantage of this capability using CanOPI, copy the printer's PPD file (this file should come with the software accompanying the printer, or alternately can be found as part of some software packages such as Adobe PageMaker) to the spooler directory

`/usr/spool/ushare/<atlkspoolername>`

and rename it `ppd`. Then use the Chooser to select the LaserWriter 8.x and configure it to query the printer.



This illustration shows the four basic types of spooler configurations (each will be discussed in detail in this section):

- A) AppleTalk to UNIX to AppleTalk
- B) AppleTalk to UNIX
- C) UNIX to AppleTalk

**NOTE** For a flowchart describing the order in which print jobs are handled by CanOPT's print daemons and scripts, see **Appendix C**.

### **AppleTalk to UNIX to AppleTalk Spooler**

This spooler appears in the MacOS Chooser. Print jobs sent to this spooler pass through CanOPI's spooler to a UNIX spooler and then to an AppleTalk printer on your network. This is the recommended method for printing to an AppleTalk printer via a CanOPI spooler.

**NOTE** *For the LaserWriter 8.x, the printer can be queried, giving valuable information to the querying application. To take advantage of this capability using CanOPI, see "PPD files and the LaserWriter 8.x driver" on page **Spooler-2**.*

To define an AppleTalk to UNIX to AppleTalk spooler:

1. Follow the instructions for Defining a UNIX to AppleTalk Spooler on page **Spooler-7** and remember the name you set.
2. Follow the instructions for Defining an AppleTalk to UNIX Spooler on page **Spooler-5**, and at step 2 place the name (remembered from step 1) into the -LLpr and -LLn fields:

```
SPN=MySpooler4:-Lp /ushare/bin/us2np:\
-LLpr test:-LLn test:\
-LLz UNIX:-LLpt ts:
```

3. Restart the spooler process:

```
# usrestart splr
```

This command kills CanOPI's spooler process if it is running and then starts the process. Be careful not to restart the spooler process while a client's job is printing, because the print job will die with the process.

## **AppleTalk to UNIX Spooler**

This spooler appears in the MacOS Chooser. MacOS print jobs sent to this spooler pass through CanOPI's spooler to an already defined UNIX printer. Before defining your spooler, make sure you can print a PostScript file to the printer. Remember the name of your printer, because you must use the same name when defining your spooler.

**NOTE** *For the LaserWriter 8.x, the printer can be queried, giving valuable information to the querying application. To take advantage of this capability using CanOPI, see "PPD files and the LaserWriter 8.x driver" on page.*

To define an AppleTalk to UNIX spooler:

1. Create, in the `/ushare/etc` directory, a file that contains a list of your printer's resident fonts. You can create a new file or copy one of the example fonts files, e.g.,  
`/ushare/etc/fonts .newsprint`  
 (Remember the name of this file as you must enter it in step 3).

```
AvantGarde-Book
AvantGarde-BookOblique
AvantGarde-Demi
Bembo
.
.
.
*
```

Make sure the last line of this fonts file contains an asterisk (\*).

2. Edit `/ushare/etc/qcap`, and find near the end of the file the sample entry for an AppleTalk to UNIX spooler:

```
SPN=DemoSpoolerX:-Lp /ushare/bin/us2np:\
-LLpr YourPrinter:-LLn YourPrinter:\
-LLz UNIX:-LLpt ts:
```

Using this sample entry as a model, make a new entry to define your spooler:

- replace "DemoSpoolerX" with the name you want MacOS users to see in the Chooser; make sure the spooler name contains

no spaces;

- replace the two occurrences of "YourPrinter" with your UNIX printer's name as defined in the `/etc/qconfig` file.

You must put single quotes around any names that contain spaces. If your entry doesn't fit on one line, use the backslash (\) to indicate continuation on the next line. **Make sure the `-LLpr` argument is the third parameter in your entry.** Your new `qcap` entry should look like this:

```
SPN=MySpooler2:-Lp /ushare/bin/us2np:\
-LLpr 'MyLaserWriter':-LLn 'MyLaserWriter':\
-LLz UNIX:-LLpt ts:
```

3. Edit `/ushare/etc/lw`, and find the sample entry:

```
# np@UNIX
# /ushare/etc/fonts.np
```

Using this entry as a model, make a new, two-line entry that points to the fonts file you created in step 1. The first line of your entry is the name of your UNIX printer followed by "@UNIX." The second line of your entry is the pathname of the fonts file you created in step 1. Your new `lw` entry should look like this:

```
MyLaserWriter@UNIX
/ushare/etc/MyFonts
```

Do not put single quotes around names that contain spaces, and do not use the backslash (\) to indicate continuation on the next line.

4. Restart the spooler process:

```
# /ushare/bin/usrestart splr
```

This command kills CanOPI's spooler process if it is running and then starts the process. Be careful not to restart the spooler process while a client's job is printing, because the print job will die with the process.

## UNIX to AppleTalk Spooler

This spooler does not appear in the MacOS Chooser. UNIX user's print jobs sent to this spooler pass through a UNIX spooler to an Apple printer on your network. Both text files and PostScript files can be sent to this spooler. To define a UNIX to AppleTalk spooler:

1. Determine your AppleTalk printer's name and zone (if any).  
The easiest way to do this is to use a MacOS Chooser to select your printer; carefully note the printer's name and the zone in which it resides.
2. Make sure you can print directly to the printer.
  - Create a PostScript file with CanOPI's **pso** command:  

```
# /ushare/bin/pso /etc/passwd > /tmp/passwd.ps
```

 This command converts your Sun host's **passwd** file to a PostScript file **/tmp/passwd.ps**.
  - Print this PostScript file with CanOPI's **lwdq.x** command:  

```
# /ushare/bin/lwdq.x -n '<printer>' \
  -z '<xzone>' < /tmp/passwd.ps
```

 Where **<printer>** is your printer's name and **<zone>** is the zone in which your printer resides. If your AppleTalk network has no zones omit this field.
3. Create ushare spool directory for the configuration files to be copied to. (For this example we will create a spooler called **test**)  

```
# mkdir /usr/spool/lpd/printers/test
```
4. Create an interface script for this spooler.  

```
# cd /usr/spool/lpd/printers/test
# cp /ushare/bin/iptmodel .
# chmod 555 iptmodel
```
5. Edit the **/etc/qconfig** to define your UNIX spooler. At the end of the file make an entry that resembles the following:  

```
test
    device = testdev
testdev
    backend = /var/spool/lpd/printers/test/iptmodel
```

6. Copy the following files to the spool directory created in item 3.

```
# cd /usr/spool/lpd/printers/test
# cp -p /ushare/bin/alert.sh .
# chmod 555 ./alert.sh
# cp -p /ushare/bin/usps_options .
```

7. Edit the `/ushare/bin/usps_options` file you created in step 6 and find these lines near the bottom of the file:

```
LWN='Laserwriter Name'
LWZ='Laserwriter Zone'
```

Replace "Laserwriter Name" with your printer's name and "Laserwriter Zone" with the zone in which your printer resides. If your AppleTalk network has no zones, use an asterisk (\*) as the zone name. Make sure to enclose the asterisk in quotes.

8. Test the new spooler by printing a text file from the command line.

```
# lp -d test /etc/passwd
```

This command sends the `/etc/passwd` file to your printer via your new spooler.

## Parameters for Entries in */ushare/etc/qcap*

Parameter	Description
SPD=	Directory for temporary files; must be unique for each spooler; default = <code>/usr/spool/ushare/&lt;SPOOLERNAME&gt;</code>
SPQ=	Program to run as the spooler; default = <code>lwq</code>
SPN=	Name of the spooler as it appears in the MacOS Chooser (up to 32 characters long). It must contain <u>no</u> spaces
-d	Debugging level for the spooler (0-99)
-n	Maximum number of simultaneous jobs; default = 16
-p	Program to run when job starts; normally <code>lwp</code>
-s	Program to run to scan for DCS in job; normally <code>lws</code>
-t	Entity type of spooler; default = LaserWriter
-z	<i>This option is only effective when using EtherTalk Phase II Protocol. It sets the zone the spooler serves; default = * (all). Entering a zone name causes the spooler to be invisible to all AppleShare clients except those residing in the specified zone</i>
-Ld	Debugging level for the job process (0-99)
-Lp	Program to de-queue the job to the printer; default = <code>lwdq.x</code>
-LLcvs	Conversion(s) performed by the <code>us2np</code> script; combinations of 1, 2, and 3; or 4; default = 1. For details, see the following page.
-LLd	Debugging level for <code>lwdq</code> (0-99)
-LLg	Name of the file where accounting information for the LaserWriter goes
-LLn	Name of the printer to which jobs are sent
-LLpr	Name of the printer to which jobs are sent (matches entry in the <code>lw</code> file)
-LLpt	Type of destination printer; " <code>ts</code> " for PostScript printers; default = <code>ts</code>
-LLt	Entity type of destination device; default = LaserWriter
-LLz	AppleTalk zone in which the destination device resides

When making entries in the `/ushare/etc/qcap` file, you can use the **-LLcvs** argument to specify the conversion(s) performed by the **us2np** script. You can assign four values to the **-LLcvs** argument:

- "1" This value causes the PostScript Dictionary (i.e., LaserWriter Prep file) to be prepended to each job in an encapsulated form.
- "2" This value activates the translation of **carriage return** characters to **line feed** characters. This translation causes problems if the PostScript contains binary encoded images.
- "3" This value causes the `/ushare/bin/wordwrap` program to limit the maximum line length to 240 characters. This modification causes problems if the PostScript contains binary encoded images.
- "4" This value causes the PostScript Dictionary to be sent to the printer as a permanent dictionary when the spooler is started. Do not combine this value with other cvs values.

You can assign combinations of these values, but "4" cannot be combined with other values, i.e., the only valid assignments are:

```
-LLcvs 1, -LLcvs 2, -LLcvs 3, -LLcvs 12,  
-LLcvs 13, -LLcvs 23, -LLcvs 123, -LLcvs 4
```

The default **cvs** value "1" is optimum in almost all cases, and you should not assign other values without good cause.

### Editing the `usps_options` File

You can customize your UNIX to AppleTalk spoolers by modifying parameters in CanOPI's `usps_options` file, e.g., to control the behavior of a spooler's .finished queue. The complete set of modifiable parameters is presented in "Parameters in the `usps_options` File" on page **Spooler-11**.

## Parameters in the usps\_options File

Parameter	Description
LWN	Name of the destination printer
LWZ	AppleTalk zone in which the destination printer resides
LWT	Entity type; default = 'LaserWriter'
PSDMAN	<b>PSDMAN</b> defines the program used to convert text on stdin to PostScript on stdout. The default is <code>/ushare/bin/pso</code> .
PSDMAN_FLAGS	Flags to pass to the text to PostScript converter
comm	Program that sends PostScript on stdin to the destination printer; default = <code>/ushare/bin/lwdq</code> . (For applications such as Hyphen's RIP, that expect to read PostScript output from a file before sending it to a printer, you can set <code>comm=pr2file</code> and set the parameters as defined in comments in the <code>usps_options</code> file)
debug	<b>true</b> or <b>false</b> . If <b>true</b> , verbose messages are sent to the debug device. The default is <b>false</b> .
minfree	Specifies the minimum available disk space required for activation of the <b>.finished</b> queue, in megabytes (to be used in conjunction with the PrintQManager). The default value is <b>100</b> .
maxjobs	This entry determines how many finished print jobs will be held in the spooler's <b>.finished</b> queue, so that they may be re-submitted by the PrintQManager application. When the number of finished print jobs matches the value for this field, subsequent finished jobs will "push" the oldest saved job out of the <b>.finished</b> queue (see the section "PrintQManager" for more details). The default value is <b>20</b> .
maxentries	This determines how many completed print jobs will be held in the list of finished jobs, as reported by PrintQManager. When used in conjunction with the Max Jobs entry, it is a convenient way to keep track of many finished jobs, while only actually saving the most recent jobs for re-submission. The default value is <b>30</b> .
maxspc	Specifies the maximum disk space consumed by the <b>.finished</b> queue in megabytes (To be used in conjunction with the PrintQManager). The default value is <b>100</b> .

## Parameters in the usps\_options File

Parameter	Description
lognotice	This is the facility and level to report printing errors to. The default is <b>"local0.err"</b> .
includefonts	This option enables uShare's ability to insert pre-cached fonts into a document if they are not permanently downloaded to the output device already. This saves time by eliminating the need for the MacOS machine to download the font description with the print job (once the font has been cached). The default is <b>"false"</b> .
extractfonts	This option enables CanOPI's ability to extract, or "learn" about fonts that the output device doesn't already have permanently downloaded. When a print job is sent to the spooler, CanOPI determines which fonts need to be downloaded from the MacOS machine to the printer, and caches this information. By doing this, subsequent print jobs using the same fonts will print without having to download the same font information from the MacOS again. The default is <b>"false"</b> .
pshdrfn	This field is used to define a PostScript file that is to be prepended to each print job as it passes through the spooler. This is helpful when you need to pass specific printer options via PostScript, to the printing device. To set the value, enter the absolute path to the PostScript header file you wish to use.
do_docman	Set to <b>true</b> to enable OPI substitution, <b>false</b> to disable. The default is false.
do_docpipe	Set to <b>true</b> to enable piping the output of docman to the lwdq process, <b>false</b> to disable. The default is true.
do_docmem	Set to <b>true</b> to enable Shared Memory Interface, <b>false</b> to disable. The default is false.
docmemflags	Use this option to set the number and size of shared memory windows. It is possible to optimize output by adjusting these values. The default is set to <b>2 windows</b> of size <b>512K</b> , and this has been found to be very effective.

## Parameters in the usps\_options File

Parameter	Description
docflags	<p>These are the flags to pass to the <b>docman</b> program (the program that handles low-res image replacement) and consist of:</p> <p><b>-a</b> Enables support for Scitex APR (Automatic Picture Replacement) PSImage, version 1.2. If you are using a Scitex RIP in conjunction with the Adobe PhotoShop plug-in "PSImage Exporter v2.5", use this flag. It is important to also specify the search directory with the -s command (described below). This is necessary because although APR does include image path information, this version of the PhotoShop plug-in does not pass this information to the image replacement engine (<b>docman</b>).</p> <p><b>-b</b> Enables binary representation of image data. This is set as a default.</p> <p><b>-c</b> Enables CanOPI's internal caching feature. This provides for faster OPI processing and should only be removed for debugging purposes.</p> <p><b>-m</b> Enables <b>docman's</b> DCS merge feature. This allows for color separated jobs to be printed to a composite only device.</p> <p><b>-M</b> Enable support for MultiAd Creator™.</p> <p><b>-R</b> Enable support for Recursive OPI (embedded EPS).</p> <p><b>-s</b> Allows you set alternate search paths for docman. These should be expressed as: -s /directory1/directory2 -s /directory3/directory4</p> <p><b>-t</b> Enables the transparency option for DTP applications. This will allow backgrounds to be set to none for positioned images. It is recommended that this option is set at all times, but it must be noted that it may cause problems with older versions of the DTP applications.</p>

### Working with an 8-bit data stream

In some cases in which MacOS print jobs are sent via CanOPI's AppleTalk to UNIX spooler to a serially-connected printer, special characters (e.g., bullets, trademarks) are not printed correctly. This problem arises because many printers' serial ports can not accept an 8-bit data stream (e.g., they might use the 8th bit for flow control). When a MacOS computer sends PostScript data to a LaserWriter (or a CanOPI spooler), special characters are encoded in eight-bit binary form (if the PostScript is saved to disk, the special characters are encoded in an ASCII Octal form, so the problem does not arise when the PostScript is sent to a printer).

The best solution to this problem is to configure the printer to support an 8-bit data stream. Many printers have a DIP switch setting that achieves this support. Other printers require software configuration (i.e., downloading of PostScript code) to achieve this support. A possible solution is to create and download to the printer a PostScript file that may allow the printer to accept an 8-bit data stream:

1. Restart the printer.
2. Create a text file `/usr/spool/ushare/high_bit_fix` that contains the following text:

```
%!  
0000000000  
serverdict begin exitserver  
statusdict begin  
25 9600 3  
setscbatch  
end
```
3. Send this file to the printer by issuing the following command at a UNIX command line:

```
# lpr -P<PRINTER> /usr/spool/ushare/high_bit_fix
```
4. Try printing a MacOS job (that contains problematic characters) through the CanOPI spooler.

Another possible solution is to incorporate into CanOPI's spooler a program that converts the high-bit characters to a digestible form. This procedure may allow Mac print jobs to include special characters, but because it will cause CanOPI's spooler to expand high-bit set bytes

into escaped-octal ascii, MacOS computer print jobs containing binary-encoded images will probably fail to print (binary encoding of images is an application-specific procedure, so the frequency of problems printing to the modified CanOPI spooler is unpredictable):

1. Edit the `/ushare/bin/us2np` file and find a line resembling the line in boldface below. You should see lines like this:

```
370 # Sometimes, software RIPS need cr<->lf translation.
371 # But this does not work if the PostScript contains
372 # any Binary encoded images.
373 tr '\015' '\012' <$t1 >>$t7
374 mv $t7 $t1
```

2. Modify that line to incorporate CanOPI's ex8 program:

```
370 # Sometimes, software RIPS need cr<->lf translation.
371 # But this does not work if the PostScript contains
372 # any Binary encoded images.
373 tr '\015' '\012' <$t1 |/ushare/bin/ex8 >>$t7
374 mv $t7 $t1
```

3. Edit the `/ushare/bin/qcap` file and modify the entry for your AppleTalk to UNIX spooler so that the cvs value is "12":

```
SPN=LaserWriter@gop32:-Lp /ushare/bin/us2np\
-LLpr PostScript:\-LLn PostScript:-LLz gop32\
-LLpt ts:-LLcvs 12:
```

4. Restart CanOPI's print spooler, and try printing a MacOS computer job that contains problematic characters.

## Printing PostScript to a File

It is sometimes necessary to send a print job to a file as opposed to an actual output device. With CanOPI, you have the ability to create an AppleTalk to UNIX to AppleTalk, or UNIX to AppleTalk print spooler that sends the print job to a directory. This is useful when a spooled print job needs to be 'handled' by another application (like an OPI, or software RIP), before being sent off to the print device. In an instance where your OPI or RIP can be set to look for print jobs in a certain directory, you may configure your CanOPI spooler to 'drop off' these print jobs.

To set this option, use EasyAdmin's "Spooler Manager". Choose "more options" from the "Edit Spooler Options" window. Then, under "UNIX spooler", set the "Print Program" to **pr2file**. The **path to print** the file should be set to your destination path for the print jobs. **Print file name** should be set to a name pattern you prefer. A \$ should be used to represent a unique integer in the pattern of the file name. Use the pattern `<desired string>_$$` to write a file with a name `<desired string>` followed by a unique two-digit integer. For a complete description, see page **EasyAdmin-29** of the User Guide.

## Moving CanOPI's Spool Directory

In the event that you need to move CanOPI's spool directory to a location with more space available, you may do so by creating a symbolic link in /usr/spool that points to the new location. To do this:

An "l" indicates that /usr/spool/ushare is actually a link to the directory the arrow (->) points to.

1. Stop your CanOPI spooling:

```
# /ushare/bin/usstop splr
```

2. Check to see where your CanOPI spool directory is located:

```
# ls -l /var/spool | grep ushare
```

if it looks like:

```
lrwxrwxrwx 1 root system 4 Mar 22 04:26
ushare@ -> /usr/ushare.spool
```

then, as "root" issue:

```
# mv <actldir> <newdest>
```


where `<actldir>` is the actual location of your uShare spool directory and `<newdest>` is the new destination of your spool directory.

```
# unlink /var/spool/ushare
# ln -s <newdest> /var/spool/ushare
```

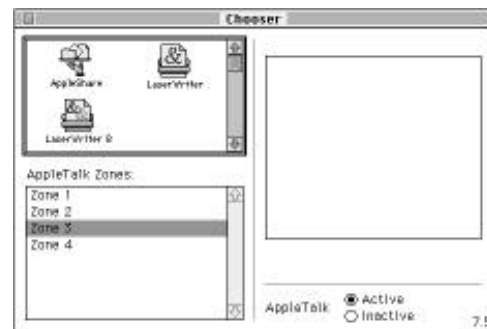
## Using CanOPI Print Spoolers

### Selecting a CanOPI Print Spooler

You select a CanOPI print spooler by using the Chooser.

1. Select "Chooser" from the  menu.

The Chooser window appears, with icons for LaserWriter, AppleShare and possibly for other network services. AppleTalk zones may also appear.



2. Select the LaserWriter icon and AppleTalk Zone (if necessary).  
A list of available LaserWriters appears.
3. From the list of available LaserWriters, select the name of a CanOPI spooler, and close the Chooser window.

Now that you have chosen the CanOPI spooler, your print jobs will automatically spool each time you choose "Print" from the "File" menu of any application.



# *CanOPI Files*

When you install CanOPI software on your UNIX host, a number of files are placed on your UNIX file system. You should not execute or modify most of these files, because they are reserved for use by CanOPI software. You can, however, edit some of CanOPI's files in order to configure CanOPI's services manually. Other CanOPI files are executable programs and shell scripts that let you control CanOPI processes, configure CanOPI services, and diagnose network problems.

**NOTE** *Read the "CanOPI Commands" chapter for detailed descriptions of CanOPI's executable files.*

## Location of CanOPI Files

Most CanOPI-related files are contained in the **/ushare** directory, while other files are contained in directories modified by CanOPI. The files described in this chapter reside in:

- subdirectories of **/ushare** (**MacApps**, **bin**, **etc**, and **etc.local**);
- MacOS users' home directories;
- directories published as AppleShare volumes.

### CanOPI Files in **/ushare/bin**

This directory contains all of CanOPI's executable files (programs and shell scripts). Most of the files in this directory are used by CanOPI processes and you should not directly execute them. The following files may, however, be executed during troubleshooting, configuration, or process control operations:

build_cache.x	opistub	usstop
cached	pso	vpaswd
cachevol	sendattn.x	
docman	usafpatr	
fontqry	usrestart	
lwdq.x	usrm	
lwpr	ussetkey	
lwstatus	usstart	

**NOTE** Read the "CanOPI Commands" chapter for detailed descriptions of these files.

### CanOPI Files in **/ushare/etc**

This directory contains CanOPI configuration files. All users must have read and search access to this directory.

#### **afp.dirs**

This file contains entries that define AppleShare volumes. Any file name can be used if it matches the **vl** parameter of an entry

**afp.servers**

This file contains entries that define CanOPI file servers and their attributes. Read the "AppleShare File Server Service" chapter in the AFP section of the Technical Guide for instructions for making entries in this file.

**fonts.plus**

This file contains a list of all fonts resident on an Apple LaserWriter Plus. Read the instructions for defining an AppleTalk to UNIX Spooler in the "Print Spooler Service" chapter in the Spooler section of the Technical Guide for information about managing fonts files.

**lw**

This file contains entries used by CanOPI print spoolers to obtain a list of a printer's resident fonts. CanOPI automatically makes entries in this file for spoolers that print to printers on an AppleTalk network. You must manually make entries in this file for spoolers that use printers not on an AppleTalk network, e.g., any PostScript or NeWSprint printer connected to your UNIX host via a serial or parallel interface. Read the instructions for defining an AppleTalk to UNIX Spooler in the "Print Spooler Service" chapter in the Spooler section of the Technical Guide for information about making entries in this file.

**packages**

This file contains the list of CanOPI process abbreviations used by the **usstart**, and **usrestart** commands when they are issued with no parameters. Read the "Processes" chapter in the Spooler section of the Technical Guide for details about editing this file.

**qcap**

Entries in this file define CanOPI print spoolers and their attributes. Read the "Print Spooler Service" chapter in the Spooler section of the Technical Guide for instructions for making entries in this file.

## CanOPI Files in /ushare/etc.local

This directory contains your UNIX host's local configuration information. This directory should never be shared (e.g., exported with NFS). All users must have read and search permissions for this directory.

### keys

This file contains your CanOPI authorization key. You can edit this file with the Key Manager, manually, or by using the **ussetkey** command.

### passwd

This file exists to support password authentication. If you want to use password authentication, edit or create the **/etc/local/passwd** file (see “Setting up AppleShare Passwords - Random Number Exchange” on page **AFP-27**), making entries in this file for everyone who is going to be using the file server, in the form:

**username:flags:password**

Where *username* is the login name of the user on the filesystem, and *password* is that user's (potentially encrypted) password for that filesystem. If you are not using random-number exchange, leave the *password* field blank.

**NOTE** *If you are going to use password authentication (either one-way or two-way random-number exchange), you must make entries in this file for all users who are going to be using the file server. This will be done automatically when setting a password with EasyAdmin.*

## CanOPI Files in MacOS Users' Home Directories

If you use CanOPI's File Server service, you may work with some files in MacOS users' home directories (a user's home directory is referred to in this guide as **~username**).

### **afp.dirs**

Entries in this file define a user's AppleShare volumes.

## CanOPI Files in Directories Published as AppleShare Volumes

Any directory of your UNIX file system that you publish as an AppleShare volume with CanOPI's AppleShare File Server service contains several CanOPI-related files.

### **.desktop/desktop**

This file contains information that lets CanOPI associate appropriate icons to MacOS files. The .desktop/desktop file is a database of MacOS CREATORS (i.e., an application's unique four-character code), and each time a MacOS file is placed on the CanOPI volume, a new entry is made in this file (unless an entry corresponding to the file's CREATOR already exists). Each CREATOR entry is linked to the file TYPES recognized by the CREATOR; and then these TYPE entries are linked to the actual MacOS icon data.

### **.dirids/cache**

This directory contains a volume-wide shared cache of files denoted by UNIX inode number.

### **.rsrc**

This directory contains the resource forks of files in the parent directory of **.rsrc**.

### **.volrsrc**

This file contains a volume's FinderInfo.

# *CanOPI Commands*

CanOPI provides a number of user-level commands that let you control CanOPI processes, configure CanOPI services, and diagnose network problems.

This chapter summarizes the syntax of the following commands:

build_cache.x	opistub	usstop
cached	pso	vpasswd
cachevol	sendattn.x	
docman	usafpatr	
fontqry	usrestart	
lwdq.x	usrm	
lwpr	ussetkey	
lwstatus	usstart	

## **Syntax of CanOPI Commands**

When you issue a CanOPI command at a UNIX command line, you must prepend the path `/ushare/bin` to each command (e.g., to start CanOPI, you must type `/ushare/bin/usstart`) unless you have added `/ushare/bin` to your Shell environment's path (the `$PATH` variable).

## **build\_cache.x**

build\_cache.x is used to update or rebuild an existing cache to an entire CanOPI volume. If you have created a new volume that is publishing directories that already have files in them, it is a good idea to build the cache before publishing the volume. To build the cache file for all volumes that do not yet have one defined, see the command "cachevol", on page **Commands-63**.

### **Syntax**

To use this command, issue the following at a UNIX command prompt, while logged in as "root":

```
# /ushare/bin/build_cache.x -v <volume path> -i 2 -r
```

where <volume path> is the path to the root level of the CanOPI volume.

### **Example**

```
# /ushare/bin/build_cache.x -v /usr/workspace /files -i 2 -r
```

This will build a cache file for the CanOPI volume whose UNIX path is /usr/workspace/files.

## **cached**

cached is a handy tool for forcing CanOPI to update the cache file for an existing directory. It may be used in conjunction with UNIX shell scripts that place files in CanOPI volume directories, so that MacOS users will recognize the updated contents immediately.

### **Syntax**

To use this command, follow the instructions below for the appropriate file system:

**1.# cd <directory path>**

Where <directory path> is the path of the directory for which you would like to update the cache file.

**2.# touch .**

**3.# /ushare/bin/cached <directory path>**

### **Example**

**# cd /macvols/Applications/Word\_Docs**

**# touch .**

**# /ushare/bin/cached /macvols/Applications \**  
**/Word\_Docs**

This will update the cache file for the CanOPI volume sub-directory "Word\_Docs", whose UNIX path is /macvols/Applications/Word\_Docs.

**cachevol**

cachevol is used to create a cache file for any CanOPI volume that has been defined, but not yet published. It is handy when you are publishing a volume whose directories already contain many files. This gives CanOPI the opportunity to build a cache before users begin to use the volume, which will improve performance significantly.

**Syntax**

To use this command, issue the following while logged in as "root" at a UNIX prompt:

```
# /ushare/bin/cachevol
```

## **docman**

**docman** is CanOPI's OPI processor, responsible for replacing low-res image PostScript information with its high-res counterpart in OPI documents. **docman** is generally called only when printing to a CanOPI spooler that has been configured to perform image replacement (e.g. **docman-true** in the spooler's **usps\_options** file). However, **docman** may also be executed so that it will replace low-res image PostScript information, and write the output to a file.

### **Syntax**

To have **docman** perform image replacement in a file, issue the following at a UNIX command prompt:

```
docman <docflags> <<inputfilename>> > <outputfilename>
```

The values for **<docflags>** are described below;

**<inputfilename>** represents the name of the file you wish to convert, and **<outputfilename>** represents the name of the result -ing file.

### **<docflags>**

For a full list of available flags please see "Parameters in the **usps\_options** file" on page 13 of the Print Spooler Service section.

### **Example**

```
# docman -b -c < newsletter.low > newsletter.high
```

This will create a file called **newsletter.high**, which is a PostScript file containing the same information as **newsletter.low**, plus the hi-res image data necessary to print to a high resolution output device.

## **fontqry**

This command queries an AppleTalk printer and builds a file containing a list of the printer's resident fonts. This command is automatically executed when you start CanOPI's Print Spooler process, but you may use it manually to build font lists for your printers.

### **Syntax**

```
fontqry -z <zone> -n <name> -t <type> -l <file> \  
-q <query>
```

### **Options**

-z <zone>	zone in which the printer resides
-n <name>	name of the printer to query
-t <type>	type of printer, usually "LaserWriter"
-l <file>	file to which the font list is written
-q <query>	PostScript file that queries the printer, usually "/ushare/etc/fontsqry"

### **Example**

```
# fontqry -z EtherZone -n LaserWriter\  
-t LaserWriter -l /ushare/etc/fonts.LW\  
-q /ushare/etc/fontsqry
```

**lwdq.x**

This command prints a PostScript file to an AppleTalk printer. This command assumes the file is on standard input, so you must use redirection to print a file.

**Syntax**

```
lwdq.x -n <name> [option...]
```

**Options**

-z <zone>	zone in which the printer resides
-n <name>	name of the printer
-t <type>	type of printer, usually "LaserWriter"
-w	wait (remain connected until successful completion of job)

**Example**

```
# lwdq.x -n 'MyLaserWriter' -z 'MyZone' < MyFile
```

This command prints the file **MyFile** to "MyLaserWriter" in "MyZone."

**lwpr**

This command converts text files to PostScript and prints them to an AppleTalk LaserWriter. Before issuing this command, you must edit the `/ushare/bin/lwpr` file and enter your LaserWriter's name and zone as indicated in the script.

**Syntax**

```
lwpr filename(s)
```

**lwstatus**

This command reports the status of a LaserWriter.

**Syntax**

```
lwstatus [option...]
```

**Options**

-z <zone>	zone in which the printer resides; default is the local zone
-n <name>	name of the printer; the default is all names
-t <type>	type of printer, usually "LaserWriter"

**Example**

```
# lwstatus -n 'LaserWriter Plus' -z 'MyZone'
job: user1; document: salesletter; status: printing
```

**opistub**

This CanOPI daemon is used to create low-res representations of high-res images. Which in turn, are used for placement in documents that will be printed in conjunction with CanOPI's OPI processor **docman**. **opistub** is normally executed when performing a Drag & Drop to a folder configured for low-res image generation. However, **opistub** may also be executed manually for the same purpose.

**Syntax**

To generate a low-res representation of a high-res image file, issue the following at a UNIX command prompt:

**/ushare/bin/opistub <options>** (all options must be defined)

**Options**

<b>-l &lt;UNIX path to Mac volume&gt;</b>	Specifies the full UNIX pathname that is the 'root', or top level of the AFP volume, as defined in the volumes description file <b>afp.dirs</b> .
<b>-u &lt;UNIX path to hi-res image&gt;</b>	Specifies the full UNIX pathname to the hi-res source image.
<b>-i &lt;user id&gt;</b>	Specifies the user id to use when creating the placement image. This is used to locate the user's home volume list (the <b>afp.dirs</b> file in the user's home directory), as well as to set the ownership of the placement image. Unless <b>opistub</b> is run as "root", this should reflect the id of the user running the opistub program.
<b>-a &lt;address of Mac client (use 0)&gt;</b>	Specifies the AppleTalk address of the Macintosh client to associate errors with. When issuing this command manually, simply use "0".

**Example**

```
# opistub -l /usr/images -u /usr/images/cowimage \
-i 12 -a 0
```

This will create a low-res version of the image **cowimage** in the Macintosh volume who's UNIX path is **/usr/images**, for user id **12** (no Mac address is specified (0)).

**pso**

This command converts files to PostScript and prints them to standard output. To print to a file, you must redirect the output.

**Syntax**

```
pso <file>... [option...]
```

**Options**

-f <name>	name of file to print in header
-fh <header>	the entire header to print
-tn <spaces>	number of spaces to expand tabs into (default=8)
-nn	turns off line numbering (off by default)
-n	turns on line numbering (off by default)
-npr	turns off the <code>pr</code> filter
-l <lines>	specifies number of lines per page (default=76)
-w <chars>	specifies number of characters per line (default=96)
-tp <converter>	specifies text-to-PostScript converter (default=/usr/bin/text2ps)

**Example**

```
#  pso MyFile > MyFilePS
```

**sendattn.x**

This command sends a message string to an alert box on a MacOS on the network. The MacOS must have the Extension "Alert Driver" installed before it will receive messages from the sendattn.x command. The volume "MacApps" contains a copy of this Extension (for more details on mounting the MacApps volume see the page **AFP-22**). The "Alert Driver" may be installed on the MacOS's local HD by dragging it to the System Folder.

When the command is executed, a message appears on the MacOS and the user must click on the alert box's "OK" button to dismiss the alert.

**Syntax**

```
sendattn.x "<message>" <net>.<node>,68
```

**Options**

<message> is an ascii string of 255 or fewer characters.

<net>.<node> is the AppleTalk address of the MacOS.

68 is the appropriate socket number to use for this command.

**Example**

```
# sendattn.x "Server going down" 9.219.56,68
```

**usafpattr**

This command lets you view/modify the Finder information for a file or folder within a CanOPI volume.

**Syntax**

```
usafpattr [flags] <volume path> pathname ...
```

**Options**

"<volume path>" is the pathname to the root of the afp volume; if your current directory is the parent of the MacOS file you are manipulating, you can use dot (.) instead of supplying the volume path;

"pathname" is the pathname of the MacOS file you are manipulating;

"[flags]" can be:

```
-s [icmrSDLnb] : set flag bits
-u [icmrSDLnb] : clear flag bits
  i : invisible bit
  c : copy protect bit
  m : multiuser bit
  r : read only bit
  s : system bit
  d : delete-inhibit bit
  l : lock/protect bit
  n : rename-inhibit bit
  b : last backup bit
-c <creator> : set icon creator name
-t <type> : set icon type name
-v <view> : print Finder info
-p <mode> : set privilege of Folder
  mode = [ugo][+-][rwx] or octal mode
  u:user g:group o:other
  +:set -:clear
  r:see files
  w:make changes
  x:See folders
```

**Examples**

To view the Finder information for a file in your current directory:

```
#  usafpattr -v . filename
```

To copy-protect a file in your current directory:

```
#  usafpattr -s c . filename
```

To set the TYPE and CREATOR for a file in your current directory:

```
#  usafpattr -c MSWD -t WDBN . filename
```

**usrestart**

This command executes the **usstop** and **usstart** commands in succession.

**Syntax**

```
usrestart <abbrev> <abbrev>...
```

Where **<abbrev>** is a CanOPI process abbreviation. Read “Process Abbreviations” on page **Processes-9** for the complete list of CanOPI process abbreviations.

**Example**

To restart CanOPI's Print Spooler process, issue the following command at a UNIX command line:

```
# usrestart splr
```

When you issue the **usrestart** command without arguments, it stops all CanOPI processes and then starts those processes whose abbreviations are uncommented in the `/ushare/etc/packages` file.

**usrm**

This command is the equivalent for MacOS files of the UNIX **rm** command. This command removes a file's data and resource forks.

**Syntax**

```
usrm [-rif] <file>...
```

**Options**

-r	indicates to recursively delete the contents of a directory, its subdirectories, and the directory itself
-i	indicates to ask whether to delete each file, and, under -r, whether to examine each directory
-f	indicates to remove specified file(s) without displaying permissions, asking questions or reporting errors

**Example**

```
# usrm <path>/myfile
```

This command deletes a file named **<path>myfile** and the file's resource fork, **<path>/.rsrc/myfile**.

**ussetkey**

This command places your key in the `/ushare/etc.local/keys` file.

**Syntax**

```
ussetkey key=<YOUR KEY>
```

where `<YOUR KEY>` = your key.

**Example**

```
# ussetkey key=37f38ac36bd44d871
```

**usstart**

This command starts CanOPI processes.

**Syntax**

```
usstart <abbrev> <abbrev>...
```

Where <abbrev> is a CanOPI process abbreviation. Read “Process Abbreviations” on page **Processes-9** for the complete list of CanOPI process abbreviations.

**Example**

To start CanOPI's File Server and Print Spooler processes, issue the following command at a UNIX command line:

```
# usstart afp splr
```

When you issue the **usstart** command without arguments, it starts all processes whose abbreviations are uncommented in the `/ushare/etc/packages` file.

**usstop**

This command stops CanOPI processes.

**Syntax**

```
usstop <abbrev> <abbrev>...
```

Where <abbrev> is a CanOPI process abbreviation. Read “Process Abbreviations” on page **Processes-9** for the complete list of CanOPI process abbreviations.

**Example**

To stop CanOPI's File Server and Print Spooler processes, issue the following command at a UNIX command line:

```
# usstop afp splr
```

When you issue the **usstop** command without arguments, it stops all CanOPI processes without consulting the `/ushare/etc/packages` file.

**vpaswd**

This command sets an AppleShare volume password for a directory that is a CanOPI volume.

**Syntax**

```
vpaswd <directory>
```

If no directory is specified, the current working directory is assumed.

**Example**

```
# vpaswd /mnt/user1/notes
Setting volume password for /mnt/user1/notes
Enter password: secret
Re-enter password: secret
```

# *Compatibilities*

## **Adjusting CanOPI to Work with Other Products**

Many products allow users the ability to open their documents on multiple platforms. With CanOPI, you have the ability to use a common storage area (a UNIX directory published as a CanOPI volume) to keep these shared files. To maintain compatibility with these products, CanOPI sometimes requires specific configuration settings. This chapter describes the necessary configurations for CanOPI to work with the products listed below:

- Aldus FreeHand
- ccMail
- GroupWise
- EPOCH File migration software
- Microsoft Excel (when sharing files with PC users)
- Microsoft Windows (when printing to a UNIX spooler)

## Using CanOPI with Aldus FreeHand Documents

When saving Aldus FreeHand documents directly to the CanOPI volume, it is necessary to turn off the CR to LF translation on the CanOPI volume where this information is being stored. (This is not necessary if files are just copied to the server). This is because the FreeHand software writes files to a volume as type TEXT, until the file is complete, then changes the type to what is appropriate for the document. This causes CanOPI to perform a carriage return to linefeed translation when the file is written initially, but not during subsequent reads of the file, after the type has been altered. To configure a volume to support this type of file:

**NOTE** *This installation must take place on a newly defined volume to avoid difficulties with existing MacOS files. The CR to LF's option must be turned off before using the volume to store files from the MacOS side. Also note that if these files are saved locally and copied to the server it is not necessary to create a special volume.*

### Create a new folder using EasyAdmin

1. Launch the EasyAdmin application (either from your HD, or from the volume "MacApps" that is published by your default file server, see the "EasyAdmin" chapter for more detail).
2. Click on the "File Sharing" button.
3. Select the uShare server you would like to publish this new folder for you, and select the "New Shared Folder" option from the File menu.
4. A new, untitled, shared folder is created. Double-click on it to open it's "Options" box. Select a location on the hardrive for it.
5. Click on the "Translate Text Files" checkbox. It must be unchecked.
6. Close the "Options" window to save your new folder definition.
7. You may now use the newly defined volume as usual. When FreeHand files are written to the UNIX directory being published as a

CanOPI volume, MacOS users will be able to access them without errors.

## Using CanOPI with ccMail

By using a CanOPI volume as a storage facility, ccMail users have the ability to share a Post Office between MacOS computers, and PC compatibles using PCNFS. The following steps must be taken to ensure compatibility among platforms.

**NOTE** This installation must take place on a newly defined volume to avoid difficulties with existing MacOS files. The CR to LF's option must be turned off before installing the ccMail Post Office; instructions are below:

### Create a new volume using EasyAdmin

1. Launch the EasyAdmin application (either from your HD, or from the volume "MacApps" that is published by your default file server, see the "EasyAdmin" chapter for more detail).
2. Click on the "File Sharing" button.
3. Select the uShare server you would like to publish this new folder for you, and select the "New Shared Folder" option from the File menu.
4. A new, untitled, shared folder is created. Double-click on it to open it's "Options" box. Select a location on the hardrive for it.
5. Click on the "Translate Text Files" checkbox. It must be unchecked.
6. Close the "Options" window to save your new folder definition.
7. You may now mount the newly defined folder and install your cc-Mail software.

**NOTE** *PCTCP users: PCTCP v1.3 for OS/2 does not support record locking, which is required by ccMail.*

### When using a ccMail PO with CanOPI and PCNFS:

1. When mounting a drive (`NET USE <drive> <host path>`), it is necessary to use the `/MS` switch at the end of the command line so the command would appear: `NETUSE <drive> <host path> /MS`

2. The **SHARE.EXE** DOS executable must be loaded to provide DOS level file and record locking.

## Using CanOPI with GroupWise

In order for the GroupWise product to recognize incoming mail and attachments, it is necessary to install the Post Office on a server that is using the 3.07 CanOPI daemons. This version of CanOPI provides a different method of file caching that appropriately recognizes when new information is written to a uShare volume. You may change the configuration of an existing server, or create a new server using EasyAdmin, then follow the steps below (see the chapter "EasyAdmin" for more information on how to create a new server).

1. Login as "root" on your uShare host and issue the following command at a prompt:

```
# cd /ushare/etc
```

2. Using a file editor (like vi), edit the file `afp.Servers` (for more information on vi, type `man vi` at the command prompt or see your Network Administrator).
3. Towards the bottom of the file, you will see a line describing the server you plan to set up for GroupWise, it will look something like:

```
nm=uShare_everest:mx=64:
```

(some servers will have more fields defined than others, the order is irrelevant)

4. Append `-s /ushare/bin/afpsess_3.07.x:VERS=21:SO` that your entry now looks like:

```
nm=uShare_everest:mx=64 \  
:-s /ushare/bin/afpsess_3.07.x\  
:VERS=21:
```

5. Write and quit the file.
6. Restart CanOPI file sharing by issuing:  

```
# /ushare/bin/usrestart afp
```
7. Groupwise will now function properly.

## Using CanOPI with EPOCH file migration

EPOCH file migration may be used on a UNIX host running CanOPI software provided the EPOCH software is prevented from "stubbying" CanOPI cache files. To prevent this:

For each published local uShare volume issue the following command at a UNIX prompt:

```
epchmod -l '<volume path>/dirids/cache'
```

where **<volume path>** is the full path to the root level of the CanOPI volume.

To effect all "remote" (e.g. NFS) caches at once issue:

```
epchmod -l /ushare/etc.local/caches/*
```

The epchmod command stands for "epoch change mode".

With the options:

-r -- recursive

-l -- lock a file onto magnetic media so it will never  
be staged (copied to optical media).

If necessary, you can prevent an entire volume or sub-folder from being staged by issuing:

```
epchmod -rl '<volume-path>'
```

Or,

```
epchmod -rl '<directory>'
```

## Using CanOPI with Microsoft Excel Documents created by PC computers

When using MS Excel documents created by PC computers, it is necessary to turn off the CR to LF translation on the CanOPI volume where this information is being stored. To do this:

**NOTE** *This installation must take place on a newly defined volume to avoid difficulties with existing MacOS files. The CR to LF's option must be turned off before using the volume to store files from the MacOS side.*

### Create a new volume using EasyAdmin

1. Launch the EasyAdmin application (either from your HD, or from the volume "MacApps" that is published by your default file server, see the "EasyAdmin" chapter for more detail).
2. Click on the "File Sharing" button.
3. Select the uShare server you would like to publish this new folder for you, and select the "New Shared Folder" option from the File menu.
4. A new, untitled, shared folder is created. Double-click on it to open it's "Options" box. Select a location on the hardrive for it.
5. Click on the "Translate Text Files" checkbox. It must be unchecked.
6. Close the "Options" window to save your new folder definition.
7. You may now use the newly defined folder as usual. When MS Excel files are written to the UNIX directory being published as a CanOPI volume (from a PC), MacOS users will be able to access them without errors.

## Using UNIX to AppleTalk Print Spoolers With Microsoft Windows

If your Windows machine has been configured so that it may directly access a UNIX printer, it will also have the ability to print to a CanOPI UNIX to AppleTalk spooler as well. However, it is necessary to configure the Windows machine so that it does not send Control-D (^D) characters at the beginning of print jobs. This character will flush a print job that is being received by an AppleTalk printer. To prevent this:

1. On the Windows machine, edit the file `/win/win.ini` (choose the "Run" option from the File Manager's "File" menu and type "sysedit").
2. Search for the word "postscript". Following this word, there may be an entry that reads:  
`ctrlD=x`, where `x=1` (default) or `0`.
3. Change or create the entry to read `ctrlD=0`.
4. This should be done for ALL printer setups in this file.
5. Save and close the file.

This will prevent print jobs that originate from the Windows machine from sending the Control-D character when printing to a UNIX spooler.



# ***Appendix A-***

## ***Suggested Reading***

This appendix provides information regarding:

- Minimum UNIX Knowledge Prerequisites
- Minimum AppleTalk Prerequisites
- A Brief Introduction to AppleTalk Networking.

**Minimum UNIX Knowledge Prerequisites**

In order to successfully install and configure uShare software, you should have the following knowledge of UNIX and your UNIX host:

- You should be familiar with the concepts of "logging in" and the "login prompt." You must know how to log in as "root" and must have the proper password.
- You should know what a "UNIX command line" is and how to enter simple commands at a UNIX command line.
- You should know how to list and change directories with the "ls" and "cd" commands and examine text files with the "more" command.

### **Minimum AppleTalk Knowledge Prerequisites**

In order to successfully install and configure AppleTalk and uShare software, you should have the following knowledge of AppleTalk and MacOS Computers:

- You should be familiar with the terms "EtherTalk" and "LocalTalk" and know how to determine what protocol(s) your MacOS Computers are using.
- You should know how to physically connect your MacOS computer(s) and other AppleTalk devices to your UNIX host.

**NOTE** *If your MacOS Computers are using their built-in LocalTalk interfaces, you must connect them to your UNIX host by using an Ethernet to LocalTalk gateway (e.g., Shiva FastPath, Cayman GatorBox). If your MacOS Computers are using Ethernet interfaces, they must be attached to the same Ethernet cable as your UNIX host, or they must be connected to a cable that is connected to your UNIX host's cable via an EtherTalk router.*

- If your MacOS Computers have Ethernet interfaces, you should know how to use the MacOS Network Control Panel to select the EtherTalk driver. (MacOS version 7.5.2 has an AppleTalk Control Panel instead.)

**NOTE** *There are two versions of EtherTalk; Phase I and Phase II. Phase II is the most widely used version, but devices on your network may use Phase I. Make sure all your MacOS Computers use the same version by selecting the proper EtherTalk driver in each MacOS clients Network Control Panel (the Phase I icon has two single-headed arrows and the Phase II icon has two doubleheaded arrows):*



## A Brief Introduction to AppleTalk Networking

This section briefly introduces the basic terminology and concepts with which you should be familiar to successfully configure uShare's AppleTalk router. For detailed information about AppleTalk networking, protocols, and routing, read Apple Computer, Inc.'s *Planning and Managing AppleTalk Networks*.

### **Basic Terminology and Concepts**

An AppleTalk network consists of network devices, **and nodes** (e.g., MacOS computers, laser printers, AppleTalk PCs), connected with network **cables**. Nodes communicate with each other by sending **packets** of data over the cables. In order for nodes to understand each others packets, the format of the packets conforms to one or more **protocols** (a protocol is like a language spoken by network devices). To insure delivery of packets to the proper destination, each node has a unique address consisting of two parts: the **network number** (the network's address) and the **node number** (the node's address within the network).

### **Network Cables and AppleTalk Protocols**

The most common types of network cables used to connect AppleTalk nodes are **LocalTalk** and **Ethernet**. LocalTalk is the native hardware of MacOS computers (a MacOS computer's printer port is also its LocalTalk port). Ethernet is the native hardware of a UNIX host (all UNIX hosts have a built-in Ethernet port). The protocol used by nodes on an AppleTalk network is determined in part by the type of network cables connecting the nodes. When AppleTalk packets travel over LocalTalk cables, they conform to the **LocalTalk** protocol. When AppleTalk packets travel over Ethernet cables, they conform to the EtherTalk protocol (there are two versions of the EtherTalk protocol: **EtherTalk Phase I**, and **EtherTalk Phase II**). Each node on an AppleTalk network must be equipped with hardware or software that allows it to use the correct protocol—this hardware or software is called an **AppleTalk Driver**.

## Internets and Routers

Two or more AppleTalk networks can be combined to form one large **internet**. The individual networks within an internet are called **subnets**. Subnets are connected to the internet by devices called **routers**. If the subnets are using different protocols, routers translate the protocols when passing packets between subnets. To insure delivery of packets throughout the internet, routers assign a unique network number (or range of numbers) to each subnet. The complete list of network numbers assigned to an internet's subnets is called the **routing table**. AppleTalk routers can also assign to each subnet one or more **zone names** that provide a means for logically organizing nodes within the internet.

## Seed and Non-Seed Routers

If a subnet has more than one router attached to it, only one of the attached routers need assign a network number to the subnet; all other attached routers can acquire the network number from this router. A router that actively assigns a network number to a subnet is called a **seed router**; a router that acquires a network number from a seed router is called a **non-seed router**. Each subnet in an internet must have at least one attached seed router. If a subnet has two or more attached seed routers, these routers must assign the same network number to their common subnet.

## Router Configuration

When a seed router connects a subnet to an internet, it must assign to the subnet a unique network number and one or more AppleTalk zone names. The list of network numbers and zone names assigned by a seed router to its attached subnets is called the **router's configuration**. A seed router's configuration is dependent, in part, upon what protocols are being used on the attached subnets. Under the LocalTalk and EtherTalk Phase1 protocols, a router can assign to a subnet only one network number and one zone name. Under the EtherTalk Phase2 protocol, a router can assign to a subnet a range of network numbers (**cable range**) and a list of zone names (**zone list**).

## Network Numbers and AppleTalk Zones

If there are no routers on an AppleTalk network, the network has a **default network number**. The default network number of an AppleTalk network is dependent upon what protocol is being used on the network. Under the LocalTalk and EtherTalk Phase1 protocols, the default network number is 0 (zero). Under the EtherTalk Phase2 protocol, the default network number is the cable range 65280 through 65534. In the absence of a router, an AppleTalk network has no associated zone names.

When there are routers on an AppleTalk network, they must assign to each subnet of the internet a unique network number or cable range (valid network numbers are 1 to 65279). The routers can optionally assign to each subnet one or more AppleTalk zone names. Zone names provide a logical organization to the internet making it easier for users to locate particular nodes. For instance, when a MacOS user selects a LaserWriter printer with the Chooser application, the Chooser displays a list of the internet's zones; by selecting an entry from the list of zones, the user can view only those printers that reside in the selected zone. Without the assignment of zones, the user would have to view a list of *all* printers on the internet.

When a node is added to an AppleTalk internet with multiple zones (e.g., when you start a MacOS computer), it must determine its **default zone** (the zone in which it resides). If the subnet to which the node is attached is using the LocalTalk or EtherTalk Phase1 protocol, the node's default zone is the zone name assigned to the subnet (because under these protocols, only one zone name can be assigned to a subnet). If the attached subnet is using the EtherTalk Phase2 protocol, and the subnet has been assigned a list of zones by an EtherTalk Phase2 router, the node must select its default zone from the subnet's zone list.

## Node Addresses and NBP

When a node is added to an AppleTalk network, it must acquire a unique address consisting of a network number and a node number. A node determines the network number (or cable range) of the network to which it is attached by broadcasting packets that request this data of

the nearest router. If no router responds to the node's request, the node acquires the default network number. If a router responds, the node acquires an appropriate network number. After a node acquires the proper network number, it must acquire a node number that is unique within its network. A node acquires a unique number by randomly choosing a number and broadcasting packets that inform other devices on the network of this randomly chosen number. If another device has already acquired this number, the node chooses another number and repeats its broadcast. Eventually, the node chooses a number that is unique within its network, and that number becomes the node's address.

Generally, people using AppleTalk nodes do not have to deal with network numbers and node numbers—Only when configuring an AppleTalk router does a user have to be sensitive to these numeric addresses. Instead, users can refer to nodes by **name**. A portion of the AppleTalk protocol known as **NBP** (Name Binding Protocol) handles the association of node names with numeric addresses. A single node can have more than one name, in which case each name is referred to as an **entity**.

### AppleTalk Nodes and Entity Types

In addition to having a name, each AppleTalk node has a **type** that identifies its role within the network (for instance, a device of type "LaserWriter" is recognized by other nodes on the network as a printing device). When a single node has more than one name, each entity has its own type (for instance, a Macintosh IIfx computer running File Sharing appears to the network as an entity of type "AFPServer" and an entity of type "MacintoshIIfx"). When your UNIX host is running uShare software, it typically appears to your network as several entities depending on what uShare services you are using (for instance, if you're using uShare's File Server and Print Spooler services, your UNIX host appears as one or more entities of type "AFP Server" and one or more entities of type "LaserWriter").



# ***Appendix B - Advanced Configuration of uShare***

This appendix provides advanced configuration instructions for uShare. This includes:

- defining uShare's AppleTalk driver configuration
- activating network interfaces for uShare
- enabling uShare's AppleTalk router
- activating uShare's AppleTalk router

## **Advanced Configuration of uShare**

To allow your UNIX host to communicate with your AppleTalk network, uShare loads an AppleTalk driver into your UNIX host's kernel. The type of AppleTalk driver loaded depends upon what protocol uShare will use over your UNIX host's network interface(s) (only one protocol may be used at a time). uShare can use either the EtherTalk Phase I or EtherTalk Phase II (default) protocol over your UNIX host's Ethernet interface(s). If your UNIX host has more than one network interface (including any combination of Ethernet interfaces), you must decide what interface(s) to activate for uShare (by default your UNIX host's built-in Ethernet interface is active). If you decide to activate two or more network interfaces for uShare, you may have to configure uShare's AppleTalk router.

To configure uShare's drivers and AppleTalk router you must edit several uShare configuration files. If you use this feature, you must take great care to insure that uShare's router configuration does not conflict with your network's existing routing table.

### **Before Configuring uShare**

To configure uShare's drivers and AppleTalk router, you should be prepared to answer the following questions:

- Does uShare have to be a router?
- Are there AppleTalk routers on your network?
- What is your network's routing table?
- How are network numbers expressed?

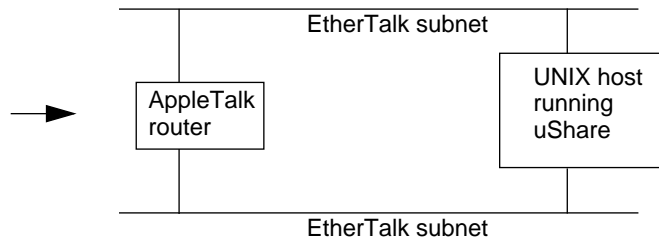
### **Does uShare Have to be a Router?**

If your UNIX host has only one network interface over which you want to provide uShare services, you do not have to configure uShare as a router. You do have the option of configuring uShare as a router over a single Ethernet interface if your AppleTalk network is using the EtherTalk Phase II protocol; e.g., if you want to define multiple Ap-

AppleTalk zones for organizational purposes, or if you have more than 254 AppleTalk devices on your network.

If your UNIX host has more than one network interface over which you want to provide uShare services, you **must** configure uShare as a router unless another AppleTalk router has already connected the subnets to which your UNIX host is attached. e.g., if your UNIX host has two Ethernet interfaces connected to independent AppleTalk networks, uShare must be configured as a seed router unless another AppleTalk router has connected the two subnets and is the "seed" router:

If an AppleTalk router has connected multiple subnets to which your UNIX host is attached, uShare does not have to be configured as a seed router.

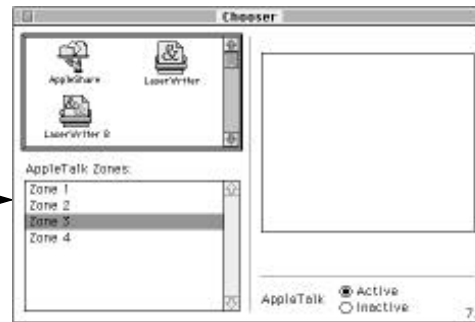


### Are there AppleTalk Routers on your Network?

When adding a router to your AppleTalk network, you must determine if there are other AppleTalk routers on the network. If a Network Administrator is responsible for this information, contact this person. If you are responsible for managing your AppleTalk network, you can use several methods to determine if there are AppleTalk routers on your network:

- **Method 1** - Open the Chooser on a MacOS computer. If you see a list of AppleTalk zones, your network definitely has one or more AppleTalk routers:

If the MacOS Chooser displays a list of AppleTalk zones, your network definitely has one or more AppleTalk routers.



If the MacOS Chooser does not display a list of zones, this fact does not prove the absence of AppleTalk routers on your network—your network's AppleTalk routers might be assigning a single zone name to the entire network, in which case the MacOS Chooser does not display the zone name.

- **Method 2** - Perform an analysis of the AppleTalk traffic on your network and look for packets of type RTMP (Routing Table Maintenance Protocol). If you see RTMP packets, your network definitely has one or more AppleTalk routers. This method is suitable only for experts in network traffic analysis, so you should talk to your network administrator.
- **Method 3** - Look at uShare's network address using a uShare command. Because this method reports the network address that uShare has acquired from your network, its accuracy depends on the integrity of uShare's physical and logical connection to your network—this method can be misleading if uShare has failed to communicate with your network because of a bad physical connection or improper AppleTalk driver configuration. Make sure uShare is running and issue the following command:

```
# /ushare/bin/usiface
```

The output of this command looks like this:

```
Number of Interfaces registered : 1
=====
AT_if  UNIX_if  Address(Net.Node)  Protocol
=====
*  et0      1e0        6.253             EtherTalk Phase 2 (range: 6-9)
=====
```

In the output above, uShare's network address is 6 indicating that the attached network has one or more AppleTalk routers. If uSh-

are's AppleTalk address is any number other than default network numbers reserved for networks that have no router, your network definitely has one or more AppleTalk routers.

**NOTE** *The default network number of an AppleTalk network is dependent upon what protocol is being used on the network. Under the LocalTalk and EtherTalk Phase I protocols, the default network number is 0 (zero). Under the EtherTalk Phase II protocol, the default network number is the cable range 65280 through 65534.*

### What is Your Network's Routing Table?

If your network has one or more AppleTalk routers, the complete list of network numbers assigned to the internet's subnets is called the **routing table**. When adding a router to your internet, you must properly configure the new router so that it does not conflict with the existing routing table. If someone else is responsible for this information, contact this network administrator. If you are responsible for managing your AppleTalk network, you can use several methods to determine your network's routing table:

- **Method 1** - The best way to determine your AppleTalk network's routing table is to document the configuration of each AppleTalk router. This method is preferred, because a careful analysis of each router's configuration is the only way to insure that no conflict exists with your network's routing table:
  - Draw a schematic of your network and indicate the location of each AppleTalk router
  - Determine the configuration of each AppleTalk router. The procedure for determining a router's configuration differs according to the type of router. Shiva FastPath routers, for instance, are configured with the "FastPath Manager" application on a MacOS computer.
- **Method 2** - Another way to determine your network's routing table is to perform an analysis of the AppleTalk traffic on your network and decipher packets of type RTMP (Routing Table Maintenance Protocol). This method is suitable only for experts in

network traffic analysis, so you should talk to your network administrator.

- **Method 3** - A third way to view your network's routing table is by using a uShare command that queries the nearest AppleTalk router for a routing table report. Because this method receives routing table data from only one router, it does not necessarily reveal conflicts that may exist in the routing table. Make sure uShare is running and issue the following command:

```
# /ushare/bin/usgz
```

The output of this command looks like this:

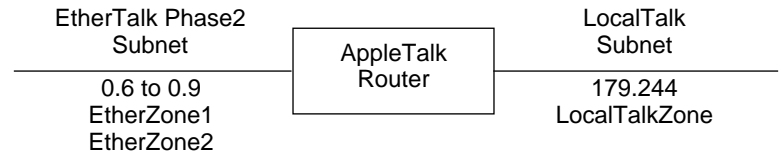
```
gateway at 6.234 reports 3 zones:
zone (EtherZone1) has nets:
0.6 (6, 0x0006) to 0.9 (9, 0x0009) hops= 0
zone (EtherZone2) has nets:
0.6 (6, 0x0006) to 0.9 (9, 0x0009) hops= 0
zone (LocalTalkZone) has nets:
179.244 (46068, 0xb3f4) hops= 0

Flags: 0x00 <zone (EtherZone1) is
valid,multicast,more than one zone>
Cable Range is 6(0x0006,0.6), 9(0x0009,0.9)
MultiCast address is 09:00:07:00:00:8 for zone
(EtherZone1)

Flags: 0x00 <zone (EtherZone2) is
valid,multicast,more than one zone>
Cable Range is 6(0x0006,0.6), 9(0x0009,0.9)
MultiCast address is 09:00:07:00:00:a for zone
(EtherZone2)
```

In the output above, the routing table report indicates that three zone names are associated with the network, and the zone names' associated network numbers are displayed. This output describes a network with two subnets: an EtherTalk Phase II subnet (network address 0.6 to 0.9, zone names "EtherZone1" and "EtherZone2") and a LocalTalk subnet (network address 179.244,

zone name "LocalTalkZone"). This network can be illustrated by the following schematic:



### How are Network Numbers Expressed?

There are three common ways to express AppleTalk network numbers:

- as **decimal** (base 10) numbers from 1 to 65279
- as **hexadecimal** (base 16) numbers from 0001 to FEFF
- as two-byte **dotted** numbers from 0.1 to 254.255

When configuring uShare's AppleTalk router, you will deal mainly with decimal and dotted notations. To convert a decimal number to its dotted form:

- divide the number by 256. The resulting whole number is the first byte in its dotted form.
- multiply the first byte of the dotted form (derived in the previous step) by 256 and subtract the resulting number from the original decimal number. The resulting number is the second byte in its dotted form.

For example, the decimal number 46068 is expressed in dotted notation as 179.244; here's the calculation performed to derive this dotted form:

$$46068 \div 256 = 179.953\dots$$

(179 is the first byte of the dotted form.)

$$46068 - (179 * 256) = 244$$

(244 is the second byte of the dotted form.)

To convert a network address in dotted form to its decimal equivalent, multiply the first byte of the dotted form by 256 and add to the resulting number the second byte of the dotted form. E.g., the dotted number 179.244 is expressed as 46068 in decimal notation; here's the calculation performed to derive this decimal form:

$$(179 * 256) + 244 = 46068$$

### **Configuring uShare's Default Interface and Zone**

Each node on an AppleTalk internet resides in one of the internet's zones—the zone in which an AppleTalk node resides is referred to throughout this guide as the node's **default zone**. Most AppleTalk nodes have a single active network interface, so the node's default zone is the zone assigned to the interface's attached subnet—the interface whose attached subnet contains the node's default zone is referred to throughout this guide as the node's **default interface**. If a node is functioning as an AppleTalk router and has more than one active network interface, the node must select one of its active interfaces as its default interface, so the node's default zone is the zone assigned to the default interface's attached subnet.

If a node's default interface is attached to a subnet using the LocalTalk or EtherTalk Phase I protocol, the node resides in the zone assigned to that subnet (because under these protocols, only one zone name can be assigned to a subnet). If a node's default interface is attached to a subnet using the EtherTalk Phase2 protocol and a list of zones has been assigned to the subnet by an EtherTalk Phase II router, the node must select the zone in which it resides.

You may have to set uShare's default interface/zone under two conditions:

- your UNIX host has multiple interfaces activated for uShare
- uShare's default interface is attached to a subnet that has been assigned a list of zones by an EtherTalk Phase2 router

**NOTE** *After setting uShare's default interface/zone, you must restart all uShare processes for the change to take affect.*

## **Steps for Configuring uShare's Default Interface and Zone**

To configure uShare's default interface and zone, perform the following steps:

1. Edit the `/ushare/etc.local/flags` file and set uShare's default interface by making an entry in the form:

```
-i <interface>
```

Where `<interface>` is uShare's name for an active interface (e.g., `et0`, `et1`).

The `flags` file can also contain an entry that activates uShare's AppleTalk router as described in "Activating uShare's AppleTalk Router" on the previous page. If the `flags` file contains a `"-g"` to activate the router, make your entry in the form:

```
-g -i <interface>
```

### **NOTE**

*If the default interface's attached subnet is using the LocalTalk or EtherTalk Phase1 protocol, setting uShare's default interface also sets uShare's default zone, so you can skip to step 3 (because under these protocols, only one zone name can be assigned to a subnet). If the default interface's attached subnet is using the EtherTalk Phase2 protocol, and the subnet has been assigned a list of zones by an EtherTalk Phase2 router, you must complete step 2 to select uShare's default zone from the subnet's list of zones.*

2. Edit the `/ushare/etc/zones` file and set uShare's default zone by making an entry in the form:

```
<zone>:<interface>:*
```

Where `<interface>` is uShare's name for the default interface you set in step 1, and `<zone>` is the name of a zone assigned to the default interface's attached subnet.

The `zones` file can also contain entries related to uShare's AppleTalk router as described in "Configuring uShare's AppleTalk Router" on page **Appendix B-11**. If you have previously config-

When uShare's AppleTalk router is assigning a zone list to an EtherTalk Phase2 subnet attached to the default interface, set uShare's default zone by appending ".\*"

ured uShare as a seed router for an EtherTalk Phase II subnet attached to the default interface and you have assigned a list of zones to the attached subnet, you select one of these zones as uShare's default zone by appending ".\*" to an entry in the **zones** file; e.g., if uShare is a seed router for two subnets using EtherTalk Phase II, each attached to an Ethernet interface on your UNIX host, and uShare's default zone is "EtherZone3", the **zones** file might look like this:

→ **EtherZone1:et0**  
**EtherZone2:et1:.\***  
**EtherZone3:et1**

In this example, ".\*" is appended to an entry in the **zones** file, so the "EtherZone2" zone is set as uShare's default zone.

If you will be making additional changes to your uShare configuration, you may skip this step until the configuration is complete.

**3.** Restart all uShare processes by issuing the following commands:

```
# /ushare/bin/usstop ALL
# /ushare/bin/unload_drivers.sh
# /ushare/bin/usstart
```

Read the "Processes" chapter for more information about uShare's processes.

It is only necessary for users who have changed their AppleTalk Phase to issue this command.

If this is a new installation, you may now proceed to the "EasyAdmin" section for details on how to customize your uShare files servers and volumes.

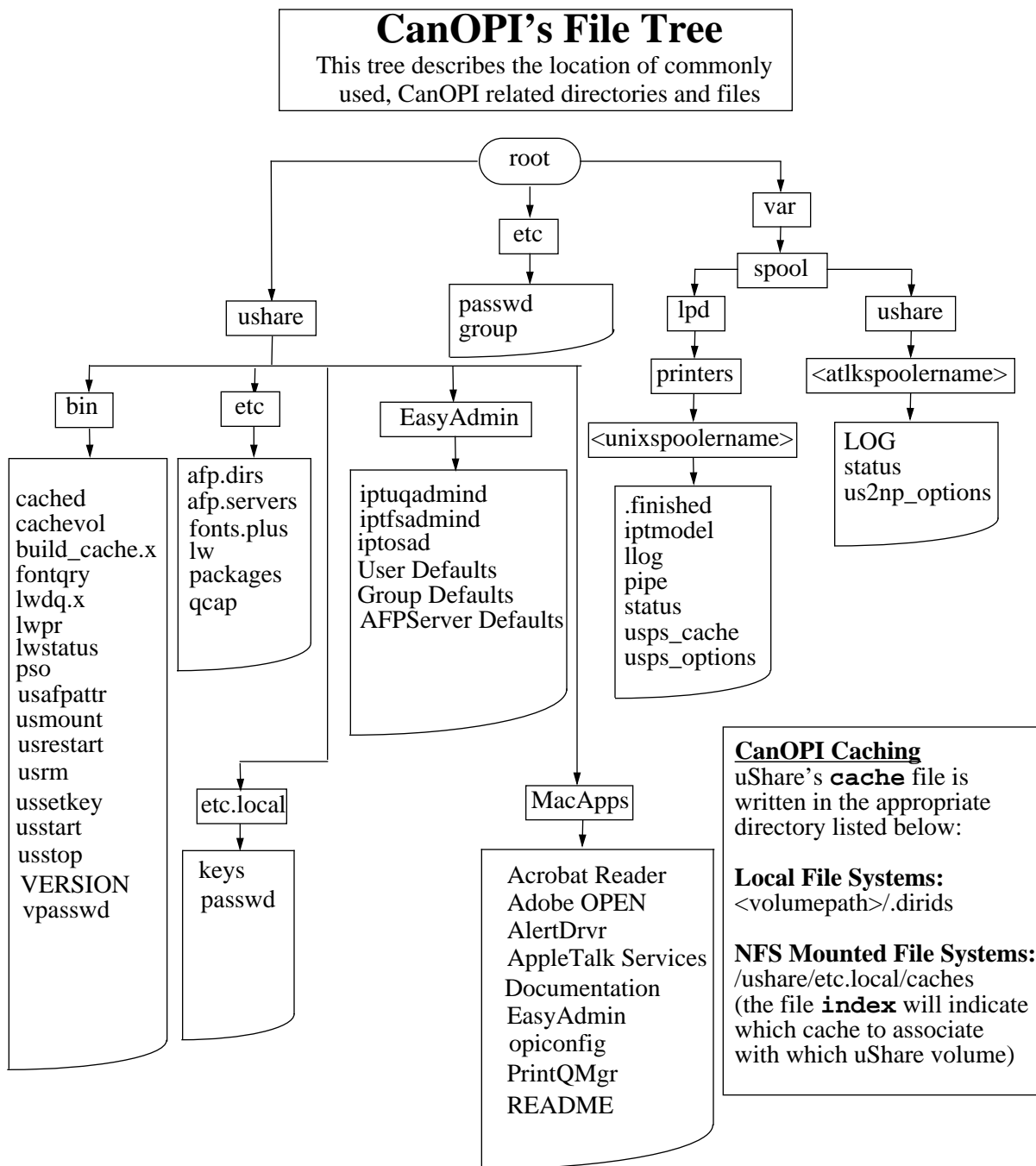


# *Appendix C-*

## *CanOPI's File Tree & Printing Paths*

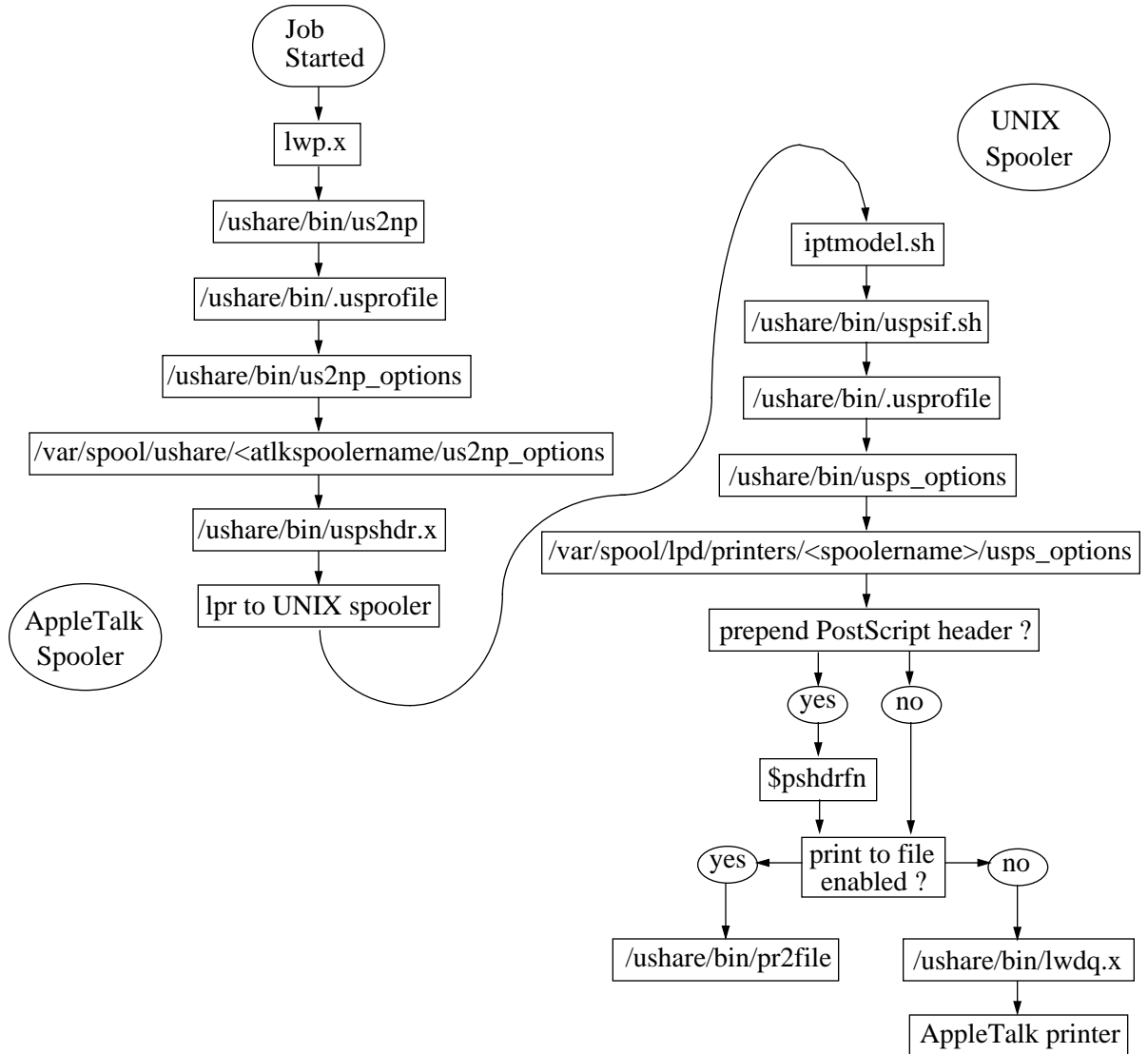
This appendix provides information regarding:

- A file system tree describing the location of commonly used, CanOPI-related files and directories
- A flowchart describing the order in which an AppleTalk to UNIX to AppleTalk print job is handled by a CanOPI spooler
- A flowchart describing the order in which an AppleTalk to UNIX print job is handled by a CanOPI spooler
- A flowchart describing the order in which a UNIX to AppleTalk print job is handled by a CanOPI spooler



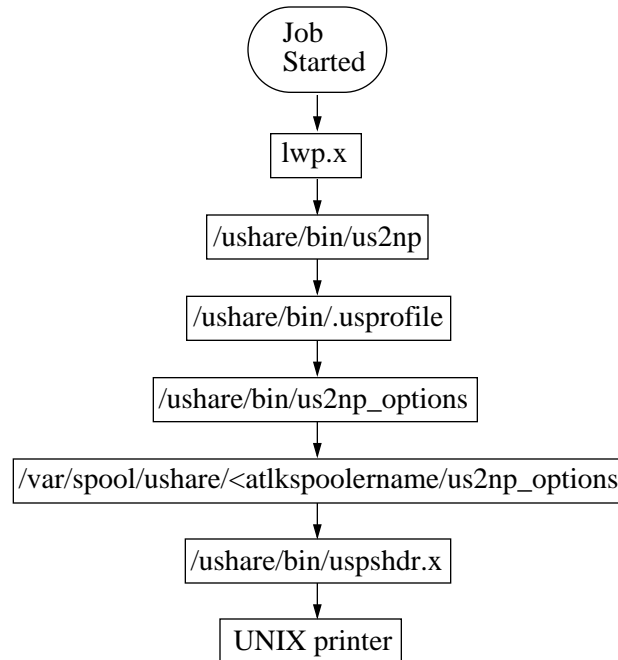
## AppleTalk to UNIX to AppleTalk Printing

This chart describes the order in which an AppleTalk to UNIX to AppleTalk print job is handled by CanOPI's daemons and shell scripts.



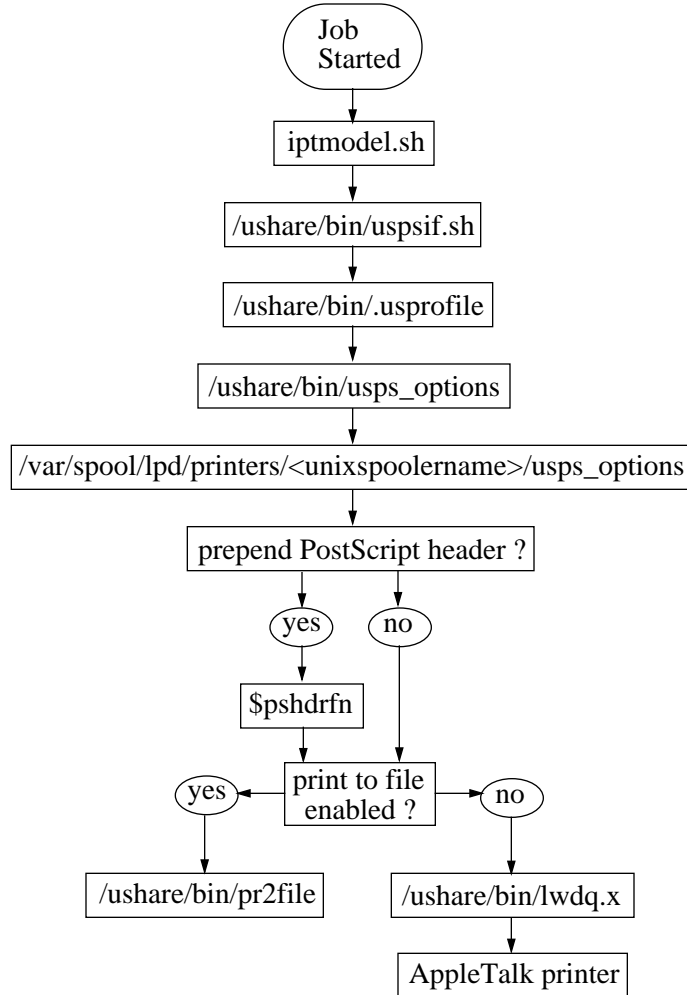
## AppleTalk to UNIX Printing

This chart describes the order in which an AppleTalk to UNIX print job is handled by CanOPI's daemons and shell scripts.



## UNIX to AppleTalk Printing

This chart describes the order in which a UNIX to AppleTalk print job is handled by CanOPI's daemons and shell scripts.





# *Appendix D-*

## *CanOPI Related Processes*

Your uShare system consists of server processes that run on a UNIX host and client applications for Macintosh users. The majority of uShare's server processes run for a brief time — they are active just long enough to perform a specific task. Some of the server processes, however, run continuously. Each of uShare's client applications communicates with one or more of these continuous server processes. The following table shows the relationship between uShare's client applications and server processes:

Client Application	Server Process(es)
Drag & Drop	iptopisrvrd iptevtlogd
PrintQMgr	IPTPrintServ
AlertDriver	errmgr
EasyAdmin	iptfsadmind iptuqadmind iptosad

## Server Processes

All of the following server processes start automatically when you complete the installation and start-up steps described in the Quick Start chapter, and each time you restart your CanOPI software:

- **IPTPrintServ** - This process handles requests from the Print-QMgr application.
- **epd, nbpd, atnbpd** - These processes, along with the EtherTalk driver kernel module, provide basic AppleTalk connectivity for your UNIX host.
- **afpd** - This process publishes an AppleShare file server for MacOS clients (uShare).
- **afpsess.x** - This process, a child process of afpd, performs AppleShare transactions and maintains the connection to MacOS clients (uShare).
- **lwq** - This process provides a print spooler to MacOS clients.
- **ipfsadmin** - File services daemon.
- **ipuqadmin** - Users and Groups daemon.
- **iposad** - Script interpreter.

# Glossary

- AFP:** (Apple Filing Protocol) Apple Computer, Inc.'s network file service protocol that allows remote access to data and applications on a server.
- alert:** a box that appears on the screen in a graphical user interface which warns the user of some condition or requests confirmation of a potentially destructive act.
- AppleTalk:** Apple Computer, Inc.'s network protocol provided with all MacOS computers.
- application:** a computer program that performs a specific task, such as word processing.
- archive:** a backup of a file, several files, or a whole system, often stored separately from that system on long-term external storage.
- Authorization Key:** see "key."
- autostart:** to have a CanOPI process start automatically when the server is booted.
- batch:** to give a series of commands to a computer which it receives all at once and then processes one-by-one.
- Binder:** the MacAdmin tool that allows files stored on a UNIX workstation to have customized MacOS *Type* and *Creator* information.
- booting:** the process whereby a computer loads its operating system into memory, or, more generally, starting or restarting a computer.
- cable range:** a range of network numbers assigned by a router under EtherTalk Phase II.
- checkbox:** in a graphical user interface, a small box which can be blank or contain an X, the state of which can be changed by a mouse-click.
- Chooser:** a MacOS desk accessory (or application under System7) used to choose network services (e.g., printers and file servers).
- client:** a computer that requests and receives a service (such as file or mail service) from a server.

- command line:** in UNIX systems, the state of a shell when it is waiting for a user command.
- configure:** the process of customizing computer hardware and software to suit a specific need (e.g., configuration of a CanOPI file server entails modification of several files that describe the file server and its volumes).
- daemon:** see "process."
- database:** a structured collection of data.
- data fork:** MacOS files have a data fork and a resource fork, UNIX files are not split in this manner.
- default:** a value that a program assumes in the absence of a user-assigned value.
- dequeuing:** the process of removing a print job from a queue and sending it to a device.
- device:** a peripheral such as a disk drive or printer that is attached to a computer.
- device driver:** software responsible for a computer's I/O interface with a specific peripheral device (e.g., a network interface board).
- dialog:** in a graphical user interface, a dialog is a box or window in which the user can enter data or answer questions.
- directory:** a file that contains other files. Within the MacOS environment, a directory is called a "Folder".
- dotted notation:** two numbers (between 0 and 255) separated by a dot, e.g. 179.244. This is one way of expressing a network number.
- driver:** see "device driver."
- drop folder:** on the MacOS, a drop folder is one which you can add files to but cannot look inside of.
- entity:** a name of a node with more than one name.
- Ethernet:** a coaxial cable system for network communications.
- EtherTalk:** a variety of the AppleTalk protocol for communication over Ethernet hardware.
- FastPath:** a gateway product from Kinetics Inc.
- file server:** a process, on a networked computer, that allows client computers to store and access files over the network; or, the networked computer itself.

- flag:** 1) A dash-prefixed command-line addition to a UNIX command that specifies some non-default action. 2) A notice attached to a file indicating that it has been checked out.
- flock bit:** "file lock bit" — a flag indicating that a file is in use.
- folder caching:** uShare places information about MacOS files and folders in a temporary storage area, or cache, allowing for improved system performance.
- gateway:** a device that routes packets between networks, especially networks using dissimilar protocols.
- global volume:** a server volume available to all MacOS clients.
- hardware:** a physical element of a computer or network (e.g., cables, keyboards).
- home directory:** the directory of a file system in which a user's private files and directories are located.
- host:** a computer that provides services to a network.
- Host ID:** an eight-character machine identification.
- hostname:** a symbolic representation of a UNIX host's address.
- icon:** a graphical representation of an element of a computer or network (e.g., the MacOS and OpenWindows environments use icons to represent files, disks, etc.).
- incremental backup:** a backup only of files that have been modified since a previous backup was performed.
- interface:** 1) the meeting point of two machines, networks or subsystems, 2) that part of a computer program that communicates directly with the user.
- internet:** the combination of two or more AppleTalk networks.
- interoperability:** a strategy of computer and network design that emphasizes adherence to the native environments of dissimilar computing platforms while enabling cross-platform resource sharing (e.g., CanOPT's print spooler service lets MacOS and UNIX users share printing resources without having to abandon their familiar printing tools).
- job:** an encapsulated task or file in the process of having something done to it (e.g. a script file being run, or a text file that has been sent to a printer or spooler).

- kernel:** the core of a UNIX host's operating system that executes low-level hardware control.
- key:** an authorization key, available from IPT, allows you to get access to the full range of CanOPI features.
- LaserWriter:** a laser printer from Apple Computers.
- LocalTalk:** the native network hardware of MacOS computers and other network devices from Apple Computers. Also the name of AppleTalk when implemented over this hardware.
- log file:** each spooler has a log file to which is written informational and error messages over time. These files can be truncated when they grow too large.
- log in:** the procedure by which a user establishes a work session on a computer (this procedure usually requires a name and password).
- log out:** the procedure by which a user terminates a work session on a computer.
- media:** refers to a method of external storage, such as tape, floppy disk or optical disk.
- mount:** to make a file system available for reading and writing.
- Name Binding Protocol:** the AppleTalk protocol that specifies dynamic assignment of addresses to AppleTalk nodes.
- NBP:** see "Name Binding Protocol."
- network:** a system of connected computers and other devices in communication with each other.
- network address:** (also "network number") The portion of an address that specifies the network as a whole rather than an individual node.
- network number:** see "network address."
- node:** a network device that sends or receives data.
- node number:** a node's address within a network.
- OPI:** Open Pre-press Interface (low-res image generation and replacement).
- packet:** a structured group of bits sent as a meaningful unit.
- PAP:** (Printer Access Protocol); The AppleTalk protocol that manages interaction between print servers and clients.

- Partner:** IPT's family of software that allows a UNIX host to be both host and client to an AppleTalk network.
- path:** a list of directories a UNIX shell examines when a command is issued at a command line.
- pathname:** the complete description of a UNIX file or directory, including what directory it is in, e.g. "/usr/local/bin/xform" as opposed to "xform."
- permissions:** in the UNIX file system, permissions determine who can read, write, or execute a file.
- pixel:** "picture element;" the smallest "dot" size on the screen.
- polling:** the act of periodically checking for something, e.g. checking to see if new mail has arrived for a user.
- PostScript:** a programming language by Adobe for formatting printer output.
- print spooler:** see "spooler."
- print queue:** see "queue."
- privileges:** the AFP equivalent of UNIX permissions, privileges are a way of making a folder available to users other than the owner in three ways: See Folders, See Files, and Make Changes.
- process:** a program running on a UNIX host.
- protocol:** a set of rules governing the transmission of data over a network.
- queue:** a list of jobs, such as print jobs, which are processed in order.
- radio button:** in graphical user interfaces, a radio button is a small circle that can be empty or can contain a black dot. The state of the button can be changed by a mouse-click on the button's location.
- resource fork:** MacOS files have a resource fork and a data fork. UNIX files are not split in this manner.
- root:** the login id for the superuser of a UNIX machine.
- router:** routers connect subnets to an internet and translate between the protocols of different subnets.
- routing table:** the complete list of network numbers assigned to the internet's subnets.
- RTMP:** Routing Table Maintenance Protocol. RTMP packets on your AppleTalk network are evidence of one or more AppleTalk routers.

- script:** in UNIX, a set of commands in a programming-language-like structure which are executed together as previously prepared in a script file.
- sed:** in UNIX, a batch editor that uses a script to modify textual input.
- seed router:** the router in a subnet that assigns a network number to that subnet. Each subnet has at least one seed router.
- server:** a host computer running software to serve clients.
- shell:** a UNIX program that accepts and interprets user commands (e.g., a C shell).
- Shell Tool:** allows an OpenWindows user to access a UNIX command line.
- software:** a set of instructions carried out by computing hardware (e.g., programs, scripts).
- spooler:** a program that queues print requests and forwards them to a printer as it becomes available.
- subnet:** an individual network within an AppleTalk internet.
- superuser:** the UNIX system administrator, who has full access to all system files and resources.
- symbolic link:** in the UNIX file system, a symbolic link is an entry in a directory that resembles a file, but actually points to an actual file or directory in another directory path.
- system administrator:** a computer user with administrative privileges and responsibilities.
- System Folder:** a special folder on a MacOS computer that contains the System and other files that provide basic MacOS functionality.
- terminal emulation:** behavior by a remote node as though it were a specific type of terminal connected directly to a host.
- UNIX:** a multiuser, multitasking operating system.
- uShare:** IPT's family of software that allows a UNIX host to provide services to an AppleTalk network.
- vi:** (Visual Display Editor); a standard UNIX text processing program.
- volume:** a file system within the MacOS environment— a volume is usually an entire storage device (e.g., a hard disk). CanOPI's File Server service lets you publish directories in a UNIX file system as AppleShare volumes.

**watch cursor:** when the MacOS is busy and cannot accept input from the user, it displays the watch cursor, an icon resembling a wristwatch.

**zone:** a group of AppleTalk networks within an internet.



# *Index*

## **Symbols**

.desktop file  
described **Files-58**  
.dirids file  
described **Files-58**  
.rsrc file  
described **Files-58**  
.ussetup file  
modifiable parameters in  
**Spooler-45**  
.volrsrc file  
described **Files-58**

## **A**

AFP volumes  
attributes of **AFP-19**  
setting **AFP-20**  
default private **AFP-24**  
removing **AFP-24**  
deleting Macintosh files  
**AFP-25, Commands-76**  
mounting with Chooser  
**AFP-32**  
naming **AFP-20**  
permissions and privileges  
**AFP-26**  
read only **AFP-25**  
setting password **AFP-25,**  
**Commands-80**  
sorting value **AFP-20**  
working with **AFP-33**

afp.dirs file  
described **Files-54,**  
**Files-57**  
syntax for entries **AFP-19**  
afp.servers file  
described **Files-55**  
parameters for entries **AFP-18**  
syntax for entries **AFP-17**  
version 4.0 new options  
**AFP-14**  
afpd  
process **AFP-16**  
afpsess.x  
process **AFP-16**  
AppleShare Workstation software  
explained **AFP-30**  
AppleTalk  
introduction to **Appendix A-iv**  
minimum knowledge prerequisites **Appendix A-iii**  
AppleTalk devices  
locating **Commands-69**  
pinging **Commands-61**  
atping  
command  
described  
**Commands-61**  
autostart  
processes **Processes-10**

**B**

binary encoded images  
and print spoolers **Spooler-44**

**C**

caching  
in version 4.0 **AFP-14**  
CanOPi  
defined **Preface-2**  
carriage return characters  
and print spoolers **Spooler-44**  
Chooser  
logging in to file server  
**AFP-31**  
mounting AFP volumes  
**AFP-32**  
selecting print spoolers  
**Spooler-51**  
commands  
atping  
described  
**Commands-61**  
fontqry  
described  
**Commands-65**  
lwdq.x **Spooler-41**  
described  
**Commands-66**  
lwpr  
described  
**Commands-67**  
lwstatus  
described  
**Commands-68**  
nbplkup **Spooler-41**

described

**Commands-69**

pso **Spooler-41**

described

**Commands-71**

usrestart **Processes-11**

described

**Commands-75**

usrm **AFP-26**

described

**Commands-76**

ussetkey

described

**Commands-77**

usstart **Processes-8,**

**Processes-10**

described

**Commands-78**

usstop **Processes-8**

described

**Commands-79**

vpasswd **AFP-25**

described

**Commands-80**

**D**

desktop

.desktop file **Files-58**

desktop file

and AFP volumes **AFP-25**

disconnecting

from file server **AFP-34**

**F**

file server **AFP-30**

attributes **AFP-17**

- attributes of volumes **AFP-19**
  - setting **AFP-20**
- deleting Macintosh files on volumes **AFP-25, Commands-76**
- explained **AFP-16**
- logging in **AFP-31**
- logging out **AFP-34**
- naming volumes **AFP-20**
- permissions and privileges on volumes **AFP-26**
- read only volumes **AFP-25**
- setting password for volumes **AFP-25, Commands-80**
- sorting value for volumes **AFP-20**
- files
  - in /ushare/bin **Files-54**
  - in /ushare/etc **Files-54**
  - in /ushare/etc.local **Files-56**
  - location of **Files-54**
  - overview **Files-53**
- fontqry
  - command described **Commands-65**
- fonts
  - fonts.newsprint file **Files-55**
  - fonts.plus file **Files-55**
  - lw file **Files-55**
  - providing list to print spoolers **Spooler-39**
- fonts.newsprint file described **Files-55**
- fonts.plus file described **Files-55**
- I**
- interfaces
  - choosing default **Appendix B-viii**
  - Macintosh network ports **Appendix A-iii**
- internet defined **Appendix A-v**
- K**
- key
  - keys file **Files-56**
- keys file described **Files-56**
- L**
- LaserWriter 8.x and PPD files **Spooler-36**
- LaserWriters
  - building font lists **Commands-65**
  - monitoring status **Commands-68**
- line feed characters and print spoolers **Spooler-44**
- lw file described **Files-55**
- lwdq.x
  - command **Spooler-41**
  - described **Commands-66**
- lwpr
  - command

- described **Commands-67**
- lwstatus
  - command
    - described **Commands-68**
- M**
- Macintosh Chooser
  - logging in to file server **AFP-31**
  - mounting AFP volumes **AFP-32**
  - selecting print spoolers **Spooler-51**
- Macintosh desktop
  - .desktop file **Files-58**
  - and AFP volumes **AFP-25**
- Macintosh privileges
  - and Unix permissions **AFP-26**
- Macintosh resource forks
  - deleting **AFP-25**
- mail file
  - described **Files-57**
- mounting
  - AFP volumes **AFP-32**
- N**
- NBP
  - defined **Appendix A-vii**
- nbplkup
  - command **Spooler-41**
  - described **Commands-69**
- network numbers
  - default **Appendix B-v**
- defined **Appendix A-iv**
- notation of **Appendix B-vii**
- node numbers
  - defined **Appendix A-iv**
- nodes
  - defined **Appendix A-iv**
- P**
- packages file
  - described **Files-55**
- packets
  - defined **Appendix A-iv**
- permissions
  - and privileges **AFP-26**
- pinging
  - AppleTalk devices **Commands-61**
- PostScript
  - controlling conversions **Spooler-44**
  - line length
    - and print spoolers **Spooler-44**
  - printers
    - printing to **Commands-66, Commands-67**
    - printing to a file **Spooler-50**
    - text to PostScript converters **Commands-71**
- PostScript dictionary
  - and print spoolers **Spooler-44**

PPD files **Spooler-36**  
 preparing to use **AFP-30**  
 print spoolers  
     and binary encoded images  
         **Spooler-44**  
     and PostScript conversions  
         **Spooler-44**  
     and PostScript dictionary  
         **Spooler-44**  
     AppleTalk to Unix  
         defined **Spooler-39**  
     AppleTalk to Unix to Apple-  
         Talk  
         defined **Spooler-41**  
     preparing to use **Spooler-35**  
     selecting with Chooser  
         **Spooler-51**  
     Unix to AppleTalk  
         customizing **Spooler-44**  
 privileges  
     and permissions **AFP-26**  
 processes  
     autostarting **Processes-10**  
     controlling **Processes-8**,  
         **Processes-10**  
     described **Processes-9**  
     description **Processes-9**  
     starting **Processes-8**  
     stopping **Processes-8**  
 pso  
     command **Spooler-41**  
         described  
             **Commands-71**  
 publish if already published  
     in version 4.0 **AFP-15**

## Q

qcap file  
     described **Files-55**  
     parameters for entries  
         **Spooler-43**

## R

resource forks  
     deleting **AFP-25**  
 routers  
     and routing tables **Appendix B-v**  
     defined **Appendix A-v**  
     identifying **Appendix B-ii**,  
         **Appendix B-iii**  
 routing table  
     defined **Appendix A-v**

## S

sorting value  
     for AFP volumes **AFP-20**  
 starting processes **Processes-8**  
 stopping processes **Processes-8**

## T

text files  
     converting to PostScript  
         **Spooler-41**,  
         **Commands-71**

## U

UNIX  
     minimum knowledge pre-  
         requisites **Appendix A-ii**  
 Unix permissions  
     and Macintosh privileges  
         **AFP-26**

- uShare
  - defined **Preface-3**
- usrestart
  - command **Processes-11**
  - described **Commands-75**
- usrm
  - command **AFP-26**
  - described **Commands-76**
- ussetkey
  - command described **Commands-77**
- usstart
  - command **Processes-8, Processes-10**
  - described **Commands-78**
- usstop
  - command **Processes-8**
  - described **Commands-79**
- V**
- volumes
  - attributes **AFP-19**
  - setting **AFP-20**
  - default private **AFP-24**
  - removing **AFP-24**
  - deleting Macintosh files **AFP-25**
  - mounting with Chooser **AFP-32**
  - naming **AFP-20**
  - permissions and privileges on **AFP-26**
  - read only **AFP-25**
  - setting password **AFP-25, Commands-80**
  - sorting value **AFP-20**
  - working with **AFP-33**
- vpaswd
  - command **AFP-25**
  - described **Commands-80**