Reference Manual for the OmniShare Application

(The Sharer)

Version 3.63

OmniShare Application (The Sharer)

OmniBus Systems Ltd
Stanford House
Stanford-On-Soar
Loughborough
Leicestershire
LE12 5PY
UK

Tel. UK: 08705 004 **300 – Fax:** 08705 004 **333**
Tel. International: +44 8705 004 **300 - Fax:** +44 8705 004 **333**

Hardware/Software support
Tel. UK: 08705 004 **350 – Fax:** 08705 004 **353**
Tel. International: +44 8705 004 **350 – Fax:** +44 8705 004 **353**
e-mail: support@omnibus.co.uk

All lines are open from 10 AM to 5 PM (UK time) Monday to Friday.

This document relates to the software version number stated and to no other application with the same name but different version number. As OmniBus Systems has a policy of continuous improvement to its products, the company reserves the right to change specifications without notice.

Contents

General Warning and Disclaimer

Some of the information in this manual is aimed more at OmniBus technical staff than users. Many of the Sharer's settings cannot be altered without seriously affecting the performance of the OmniBus system as a whole. Generally, the Sharer's settings should not be altered, except under direct instruction from OmniBus technical staff.

It is also expected that users will become familiar enough with the hardware and OS to care for routine maintenance, such as backup.

Version Numbers for the Sharer and System Manager

The OmniBus Sharer and System Manager are interdependent and it is important that the correct versions of the two applications are used together.

From V 3.60 onwards the following will apply to version numbers for the Sharer and System Manager:

Version X.AB.bf

X – Main version number (i.e. '3' from '3.60')

AB – Minor version number, 2-3 digits (i.e. the '60' from '3.60')

bf – bug fix number, always 2 digits (i.e. the first patch for V 3.60 would be V 3.60.01)

Each version of the Sharer has a preferred version of System Manager that should be used with it. The preferred version is the one that is supplied (with the Sharer) in the InstallShield package that contains the Sharer and System Manager.

All versions of the Sharer are designed to work with any previous version of System Manager. But not all versions of the Sharer will work with newer versions of System Manager. As an example, Sharer 3.61.03 is supplied with System Manager 3.70.02: Sharer 3.61.03 may be used with System Manager 3.11.04, or System Manager 1.40, but it cannot be used with System Manager 3.99.99.

What the Sharer does

The Sharer is a server application. It maintains the OmniBus file indexing system (also known as the OmniBus database) and provides an access point to the data stored there for other applications around the network.

The Sharer provides the following basic services to the OmniBus system:

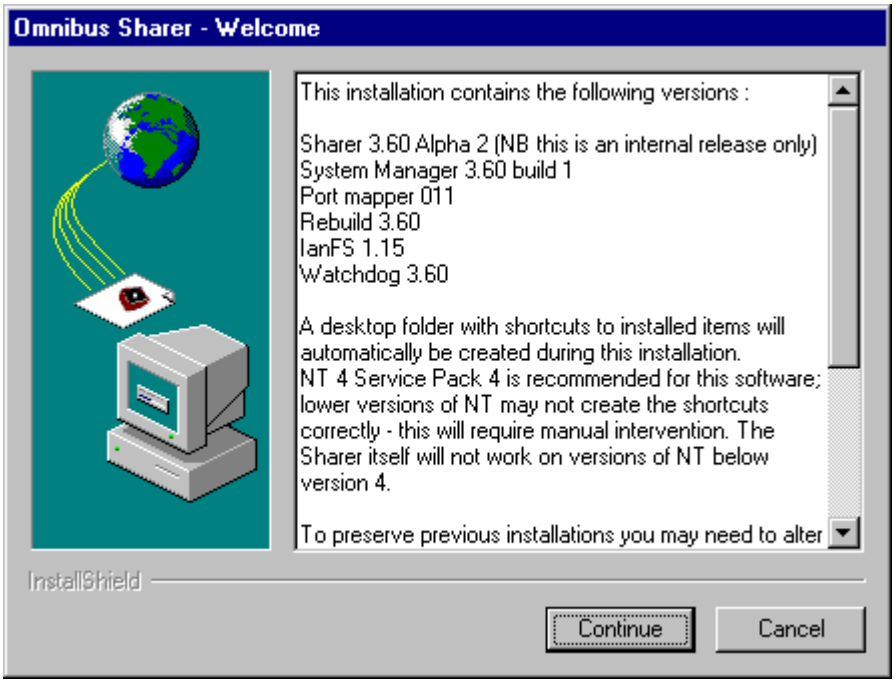
- When someone clicks on the logon button on an OUI it is the Sharer that grants access to the system.
- The Sharer provides handshaking between the OUI and applications on the system.
- The Sharer provides applications with access to the data stored in the OmniBus database – when an application loads a clip or a configuration file, the Sharer provides the data.
- The Sharer supplies the OUI with the file information that is displayed in the Filer-Fax.

What the Sharer does not do

- The Sharer is not a media control application. It registers files in OmniBus categories and creates folders for the data on the server. It is then up to applications to store whatever they want to in that folder.
- The Sharer is not aware of clips, schedules, or other media data. If an application accesses a clip through a category and loads it only to find that the clip is corrupt, then the Sharer cannot help.

Installation and initialisation of a new system

The Sharer requires an Intel™-based machine running Windows™ NT 4.0 with Service Pack 6a. Alpha-based machines running Windows™ NT 4.0 with Service Pack 6a are also supported. TCP/IP must be configured during the NT installation. A standard InstallShield program is provided for installing the Sharer, configuration files and related software. **Make sure you use the correct installation program for your target platform**. The InstallShield will fail to run on a machine of the wrong processor type. A windows error will be produced, saying "The image file <filename> is valid but is for a machine type other than the current machine."



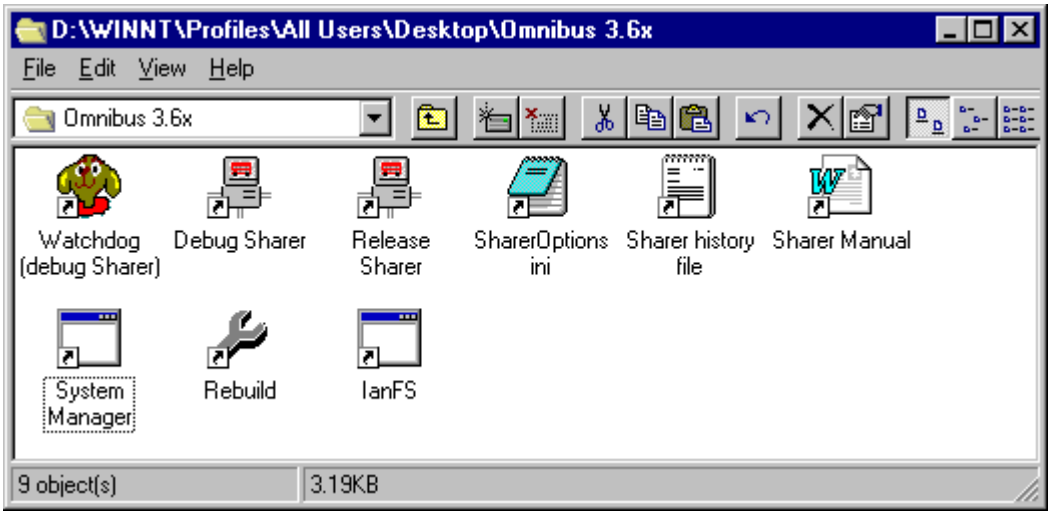
Some maintenance may be required when upgrading from previous versions particularly if an older version of the System Manager was in use. For further details see Appendix B: Upgrading.

The installation tool installs the software to the <current drive>\OmniBus\Sharer\…

The program alters Windows™ NT's Services file to configure NT's TCP/IP services for running the OmniBus software.

The program creates the <current drive>\OMN directory with SharerOptions.ini and Config – the Sharer needs these two files to initialise the OmniBus database.

The Program copies the Sharer.exe files and creates shortcuts in a folder on the desktop.



Straight after installation, the first time the Sharer is run it will automatically create a basic OMN database with basic default entries. The User SYS is created the first time the Sharer is run.

Config files and OS config files are edited and added to as necessary. If a new SharerOptions.ini is copied, no user settings will be lost from \omn\SharerOptions.ini: they will be copied to the new file.

How the Sharer interacts with the rest of the OmniBus network

The Sharer uses TCP/IP to communicate.

Registering Engines and OUIs

All OUI Workstations must be registered with System Manager to allow users to log on to them. If not, the Sharer will respond with a message like "`Your OUI (at 1.4.4.25) has not been registered in the OmniBus database`", and will log an error with code 700030.

All engines running OmniBus-enabled software (including third party software that communicates with OmniBus) should be registered with System Manager.

Multi-domain

The Sharer also validates against the Name Resolution Tables (NRTs), so that multi-domain filing will work. To allow multi-domain filing to work for OUIs and engines from another domain, these must be made 'public' using System Manager. This puts the required engines and OUIs into the domain's published NRT, which is forwarded to all connected domains.

Communications with the networked applications

Below are some examples of how the Sharer 'talks' to the rest of the OmniBus network.

"Conversation" with OUI during logon:

At logon, an OUI broadcasts a message looking for the Sharer.

The Sharer responds, from which message the OUI works out Sharer's TCP/IP address.

The OUI then responds and requests a log on.

The Sharer tells the OUI to get the user name and password (if there is a password)

The OUI responds with this information.

The Sharer then tells the OUI either: "password's wrong / user name is unknown" or "logon accepted" and the details for which local applications the OUI should supply to the user.

Logging onto an application:

An OUI asks the Sharer for an application list.

The Sharer sends a list of available applications to that user and marks applications known to be alive and available to that user.
The user sends a selection back to Sharer.

The Sharer informs the OUI where the application is, and tells it to logon to that engine.

Filing services:

Applications that want to save or retrieve data connect to the Sharer.

Application asks the Sharer for the actual path to let's say "Fred" in category "SOMEDATA."

The application then asks the Sharer for file "ConfigInfo" at that path.

The Sharer sends it back down over the TCP link.

Since the Sharer is a multi-threaded application several, or theoretically hundreds, of these conversations could be going at the same time – if the machine running the Sharer were powerful enough.

Security

File creation

Files are created with the user's default file creation flags. These are set for each user from the System Manager.

Filing rights enforcement

Delete protection:

Requests to dustbin files are checked against:

- the user being SYS (overrides all protection)
- the user's own right to delete files
- the file's delete protection flags.

Users are prevented from tagging files directly into the dustbin category by the Filer-Fax.

There is no safeguard for applications automatically tagging categories into DUSTBIN, although all tag and untag operations are logged in the Sharer's event logs. For example, a Cue Service Provider named DUSTBIN would tag all of its data into the DUSTBIN.

Write protection:

Write protection is not implemented in the Sharer. Individual applications are currently responsible for enforcing write protection on files.

Read protection:

A user who has no read access cannot log on to the system. Other read protection is not implemented.

Filing IP Address Checking

This can be controlled from SharerOptions.ini. If enabled, then any filing requests from unknown network nodes will receive an error response (code 790001). An ‘Unknown’ node is one that is not registered in System Manager 3, or marked as ‘Public’ in System Manager 3 in a connected domain.

Overview of the OMN directory structure, including key system files

- The Sharer stores all data in \Omn of the current drive.
- If you wish to install the Sharer to a different drive than the one that the data is on, you will need to create a shortcut to the Sharer and alter the Properties\Shortcut\Start in setting to the drive of the OMN directory.

Key files \ directories

Config	See below, "Details on \Omn\Config".
SharerOptions.ini	See below, "SharerOptions.ini".
Welcome	The message the OUI displays when you log on (it may be edited using System Manager).
Nxtfile	Used to decide the path to allocate to newly created files in the system.
Cat*.*	All user categories are stored here.
Domains*.*	Information for multi domain systems is stored here.
Index*.*	The category of all user categories.
Links*.*	The internal data used on multi-domain systems.
Logs*.*	All Sharer logs get stored here. (It is safe to delete any and all files in this directory at any time, even when the Sharer is running.)
Names*.*	Keeps track of how many instances of each file name there are (i.e. when to add '#1', '#2' to newly created files).
SYS_INDEX*.*	Top level category. This can be used to find locations of all other system categories.
Syscat*.*	System categories of applications, areas, OUIs, engines, groups, users.
Sysdat*.*	Data on applications, OUIs, engines, and users gets stored in here.

Unless otherwise stated, never modify, move or delete any of these files; otherwise your system will at best behave erratically, at worst stop working entirely – possibly with no hope of recovering the lost data.
The Sharer will not start if the contents of the OMN directory are not correct.

Exceptions to the above rule:

Welcome – The welcome message can be altered freely; this is easily done using System Manager.

SharerOptions.ini – These options can be altered under the supervision of OmniBus technical staff, or by other personnel if they are sure they know what they are doing.

Logs*.* – as already stated, these files may be deleted.

Details on \Omn\Config

The correct version of Config is automatically installed during the installation process. There are five types of command in this file:

SYS_ID

Text after the '0' is stored in the meta-data with all files registered on the system. It appears in the File Information Box when this is selected through the Filer-Fax.

DATA_TYPE

Files created by the OmniBus System are given a unique type – actually a number 'hard-wired' into the code. For example, a clip is type 50.

DATA_TYPE lines instruct the Sharer on the category in which files of each type are to be created. Files of type 50 go into category CLIP. If the line in the configuration file that reads "DATA_TYPE 50 CLIP" was altered to "DATA_TYPE 50 MY_CLIPS", then all clips would be created in category "MY_CLIPS" when the Sharer was restarted.

FILE_TYPE

Directories to use for file types.

The FILE_TYPE instructions inform the Sharer where to store data in the OMN directory.

For example: FILE_TYPE 30 \Omn\UCONF #SYSTEM DATA FILES

Data for files of type 30 is stored in \Omn\UCONF\…

PLATFORM_TYPE

Hardware information for the System Manager. Used by Site Management to control Software uploads.

OUI_BASED_APP

Information on local applications available to OUI for the System Manager for use when configuring an OUI through the System Manager.

ShararOptions.ini

The Sharer options are contained in \Omn\SharerOptions.ini. This is a standard Windows initiation file.

The file is divided into main sections: MultiDomain, Logging, Network, Debug, Filing, ODBC.

More detail on each section follows. Note that some of this information is aimed more at OmniBus technical staff than users.

MultiDomain

These settings are related to multi-domain systems. There are no user configurable elements in this section.

DomainsNRT	Default is OFF NRT = Name Resolution Table – used to inform engines in the local domain about engines available in other OmniBus domains, and to inform other domains about engines available in this domain. ON – Domain's NRT functionality enabled. The Sharer will maintain Local NRT and broadcast published NRT to other Sharers configured in \OMN\DOMAINS (configured through System Manager.) The Sharer will keep all interested engines informed of updates to NRT. OFF - Domains NRT functionality disabled. The Sharer will destroy the NRT file if it exists.
DomainsNRT FromFile	Default is OFF ON – If domain's NRT functionality is enabled: The Sharer will not maintain Local NRT - the contents of the NRT file remain untouched. It will still inform other local engines \ Sharers in other domains about NRT file's existence. (This is used as a debugging tool when comms are suspect: it allows manual entry of an NRT.) OFF – If domains NRT functionality is enabled: Sharer will maintain Local NRT as normal.
OmniBus DomainName	This is used to provide a domain name for the Sharer if none is configured in System Manager. It is common between machines in an NT cluster so long as this ini file is common. It is only applied if DomainNamePrefix is ON. This entry should be disabled by default, allowing the Domains information set by System Manager to have priority. Clashes between this entry and the entry from System Manager will stop the Sharer from running.

DomainName Prefix	Default is OFF Adds the domain name to the paths returned when locating files in the OmniBus database. Allows differentiation between files with the same path in 2 different domains. Switch ON for multi domain systems. Switching this ON enables all other prefix switches by default; they can be switched off individually.
Prefix Register	Leave commented out (*for internal de-bugging use only*)
Prefix CreateCat	Leave commented out (*for internal de-bugging use only*)
Prefix ResolveRef	Leave commented out (*for internal de-bugging use only*)
Prefix RenameFile	Leave commented out (*for internal de-bugging use only*)
Prefix FindFile ByRef	Leave commented out (*for internal de-bugging use only*)
OldPrefix FindFile ByRef	Comment out for non single-domain systems. Set to OFF for multi-domain systems.
Prefix FileFind	Leave commented out (*for internal de-bugging use only*)

Logging

This relates to details in the event and error logs. Users can adjust settings in this section without ill effect although, if certain logging options are switched off, diagnosing certain types of problems may not be possible.

Record NetMessages	Default is OFF for release version, ON for debug version Records all network messages coming into and going out of the Sharer in a file called TxRx??[hh].LOG where ?? is the day of the month and hh is the hour in 6 hour blocks starting at 00 for midnight.
Record RawCBNP	Default is OFF for release versions, ON for debug. CBNP is a text-based message type that the Sharer uses for some communications with applications. If this switch is ON those messages are dumped in full to the network log. Set to OFF – if there is a need to troubleshoot at this low level, it will have to be done under OmniBus supervision anyway.
Record Streamed InCBNP	Default is OFF for release versions, ON for debug. The Sharer must stream in – or interpret – CBNP messages. Setting this to ON dumps the results of streaming in a CBNP message to the network log. Set to OFF unless asked to set ON by OmniBus technical support.
RecordFNs	Default is OFF FNs are filing notifications – some applications can ask the Sharer to inform them of file changes within parameters given to the Sharer. If this is ON all Filing Notifications sent out are dumped to the TxRx log. Use of this switch could produce even larger TxRx logs if there are many active FN Requests.
Record FNReqs	Default is ON. An FN request is a request from a client application to send it an FN on given events. If this is ON then every FN request and cancellation is dumped to the network logs. Typically, applications will register their FN requests on startup only. This option can be left ON without fear of very large network logs building up.

Record AppErrors	Default is OFF Switches recording of all application errors in TxRx logs and in DAYnn logs. Switching this on can significantly increase the size of log files.
Record FileNotFound	Default is OFF Switches recording of error messages relating to error message 700003 (File not found). These messages are consistently generated as part of the normal operation of an OmniBus System. This flag allows them to be recorded when debugging specific problems.
Record ZeroByteReads	Default is OFF Switches recording of error messages relating to error message 700079 (Zero byte file reads). These messages are consistently generated as part of the normal operation of an OmniBus System. This flag allows them to be recorded when debugging specific problems.
Record XREFNoCats	Default is OFF Switches recording of error messages relating to error message 700147 (XREF contains no categories). These messages are consistently generated as part of the normal operation of an OmniBus System. This flag allows them to be recorded when debugging specific problems.

Network

This governs the Sharer's communications over TCP/IP.

Preferred HostAddress	Defaults to "" The IP address to choose, given a choice on a machine with multiple IP addresses. "" Causes the Sharer to pick the default IP address of the machine.
Backlog	Defaults to 60 Number of TCP connections that can be backlogged Range is 1 to 1024. Do not edit.
Broadcast	Default is ON ON - On starting, Sharer broadcasts its whereabouts to all OUIs and engines. OFF - Sharer does not perform this broadcast, leaving it to engines & OUIs to find the Sharer. However, the continuous poll of engines is still conducted. Force manual poll option from menu still works. This should never be OFF for customer sites.
Engine RollingPoll	ON – The Sharer conducts rolling poll of all engines in database This should never be OFF for customer sites.
OUIRolling Poll	ON - Sharer conducts rolling poll of all logged-on OUIs This should never be OFF for customer sites.

Debug

This governs what happens if the Sharer experiences a run-time programming error, or receives seriously faulty input from a client application.

Crash MessageBox	Default is OFF ON – If the Sharer crashes it will put up a message box informing the user. Using this option means the Sharer cannot automatically be restarted after a crash because user intervention is required (to OK the message box) before the Sharer terminates. OFF – The Sharer will just terminate on a crash. Use this setting if the Sharer is being run under Watchdog or Microsoft Cluster Server.
CrashDebug	Default is OFF ON – Sharer will allow a debugger (such as Dr Watson, or MSDEV) to attach and handle a crash. This option should not be ON where automatic restart in the event of a crash is wanted.
DbgAssertion MsgBox	Default is OFF Debug assertion failures are usually caused by seriously faulty input over the network from client applications. They can also be caused by failure of internal data. ON – Sharer puts up debug assertion failure message box if a debug assertion failure occurs. This allows a developer to attach a debugger on a development machine. Site versions should not have this option switched on. OFF – Debug assertion failures are logged in the error log file. No message box is put up; no user intervention required.

Filing

Some options related to the Sharer's management of the OmniBus database. On larger systems, some of the settings in this section may need fine tuning.

FilingFind IgnoreBinned Files	Default is ON Sharer returns 'not found' to applications for files in the DUSTBIN; although these files are still visible from the Filer-Fax.
BaseUntag	Default is OFF (i.e. an attempt to untag a clip from category CLIP will return an error.) ON = Sharer can untag files from their base types
EmptyBin Filechanged Delay	Min is 0; default is 250. For every file purged from the dustbin the Sharer sends a broadcast out to all applications in its domain. Some applications need extra time in between such messages to respond. This tells the Sharer how much delay – in milliseconds – to introduce between each file purged from the DUSTBIN. Max is currently around 19000ms. (System Manager 3 gives feedback to the user when the DUSTBIN is being purged. It times out if it receives no feedback for 20 seconds.) For some systems, this value may need increasing.
EmptyBin Folder DestroyDelay	Default is 300000 = 300 seconds = 5 minutes. Delay (in milliseconds) between emptying of a file from dustbin and actual destruction of its folder. Setting this delay can give applications a chance to load data on the file before it is destroyed. NB: setting this to a high value is unwise if large numbers of files are to be emptied from the dustbin. This is because the Sharer keeps track of folders to be destroyed in memory.
EmptyBin TrackingTime	Default is 1 = 1 day. Time (in days) for which the Sharer will track files being removed from the filing system. This allows the Filing Notification Synchronisation mechanism to retrospectively give client applications information about changes to the OmniBus database in the past. This allows for down time of the client application for this number of days. Setting this to a high value will impact the amount of time needed to maintain this list, which may affect Sharer performance.

PadInstance Count	Default is OFF OFF = Sharer generates files called File#1 when duplicate files are registered. ON = Sharer generates files called File#001 when duplicate files are registered. Enabling this could cause problems in very old versions of OmniBus client applications, which will expect the file to be called File#1 and may not recognise the new file, or may display the name incorrectly.

ODBC

This tells the Sharer which ODBC data source to use when supporting SQL queries of an OmniBus SQL database from the Filer-Fax.

DataSource	Defines ODBC source (as set up in Control Panel, ODBC) No default; this must be explicitly defined.
UserId	Default is "sa" User id for logon to SQL server data source
Password	Default is no password Password for logon to SQL server data source
TableName	Default is “vuOmniBus_SharerSQLSearch” for compatibility with the Media Data Gateway application. Defines table name to use in the search.

NB: For SQL searches you will need to install Microsoft Data Access Components on the machine the Sharer runs on. For SQL Server 7 this is a file called MDAC_TYP.EXE. It is available on Microsoft’s web page, and is available on the SQL server 7 CD ROM. These software components provide the ODBC drivers the Sharer needs to contact the SQL server database.

Security

This covers the Sharer's implementation of various security-based issues.

FilingIPAddressCheck	Default is "OFF" If ON then the Sharer will respond with an error to all filing requests from network nodes which are not registered in System Manager 3 or marked as 'Public' in System Manager 3 in a connected domain.

NB: If FilingIPAddressCheck is set to "ON" all machines which will access the Sharer's files must be registered using System Manager 3 or marked as 'Public' in System Manager 3 in a connected domain. This includes third party software and OmniBus software running directly on third party hardware. Use System Manager 3.60 "Poll for Engines" (or in System Manager 3.70, "Poll for new apps") and "Poll for Stations" to make sure that all currently running applications are found. If OUIs in a remote domain are to be used to look at your domain's Filer-Fax, then they must be marked as 'public' in that domain. If Columbus or Cue Service Providers are required for multi-domain, they must also be marked as 'public'.

Configuring the Sharer for multi-domain systems

Ordinarily, applications are only aware of other engines within the same network address space. Each domain has its own Sharer and OMN database. An engine in domain A cannot usually see an engine in domain B.

Multi-domain systems overcome this limitation. The Sharer is configured, through the System Manager, to see Sharers in other domains.

On starting, the Sharer contacts the other Sharers it knows of and exchanges information with them about available engines. Each Sharer uses this information to form its own NRT (Name Resolution Table) which applications can use to resolve an engine's name to its IP address.

For issues relating to the publication of engines' availability across multiple domains, see the appropriate section of the System Manager 3 Reference Guide.

For information about the functioning of the multi-domain protocol as a whole, please refer to the OmniBus Multi-Domain Specification document. (For details of this document, contact OmniBus Systems.)

There follows a brief description of how the system is configured as a whole.

Configuring the Sharer

The [MultiDomain] settings in SharerOptions.ini must be set correctly for proper use by a multi-domains system.

DomainsNRT = ON;

DomainsNRTFromFile = OFF;

Either DomainNamePrefix = ON; and OmniBusDomainName as required in System Manager 3 set-up for this domain (there is a 6-character limit).

Or DomainNamePrefix = OFF; and OmniBusDomainName disabled.

PrefixXXX = ON or disabled.

Once it has been configured, restart the Sharer.

If OmniBusDomainName is set to a value different to that in System Manager 3 the Sharer will fail to run until the conflict is resolved. Comment out the setting in SharerOptions.ini.

Configuring System Manager 3

Run System Manager 3. Set up the Domains screen in the System menu. Make sure in particular that 'This Domain' has been set up correctly.

Configuring the Sharer revisited

If DomainNamePrefix was OFF, set it to ON in SharerOptions.ini. OmniBusDomainName should be disabled. Restart the Sharer.

Remember that if the Domain name of 'This Domain' in System Manager does not match the setting in OmniBusDomainName, the Sharer will fail to run.

Setting up engines and applications

Set up System Manager 3 as in a normal system. (See the System Manager 3 Reference Guide.) Select as Public those engines and OUIs that are to be involved in multi-domain interactions. Remember that although any OUI may be 'Public', only dedicated engines (those with one and only one application) may be published in this way. (If this restriction prevents an engine from being configured correctly, contact the Sharer team via OmniBus Support).

Ensure all applications running on the system have been cleared for multi-domain compliance by OmniBus Customer Support. Enabling multi domain support introduces global changes into the syntax of the communications between the Sharer and applications.

Tests should be carried out to ensure that all applications requiring multi-domain interactions carry out their required tasks correctly.

NRT_EXTRA

This file can be created or edited to allow the configuration of devices for inclusion in the Name Resolution Table, which could not normally be added using System Manager. It should be placed in the OMN folder at "\omn\domains\nrt_extra".

The file is CSV-based, with linefeed or carriage return/linefeed separating entries.

An example of the use of this file would be to add entries allowing different VTR groups to be controlled using a single VSVC engine. For each IP address, there can only be one engine configured in System Manager. To allow NRT entries to reflect the group names of various VTR groups controlled by one engine, NRT entries must exist for each VTR group, keyed on the same IP address.

If we wanted to have groups VTR A; VTR B and VTR C, then Engine in System Manager would be configured normally as VTR A, and an additional entry in NRT_EXTRA would be added as follows:

```
VTR B,P,E,10.0.0.95,!APP_VSVC
VTR C,L,E,10.0.0.95,!APP_VSVC
```

The format for each line is

```
[NAME],[P/L],[O/E],[IP ADDRESS],[!App_name]
```

P means that the device is 'publicly available' for use by remote domains.

L means that the device is not 'publicly available' for use by remote domains.

O means that the device is an "OUI".

E means that the device is an "Engine".

The definition of a 'duplicate' entry is one which matches [NAME], [O/E] and [IP ADDRESS].

The contents of this file will overwrite any duplicate entries from the System Manager configuration.

Duplicate entries within this file will be ignored – only the first will be added.

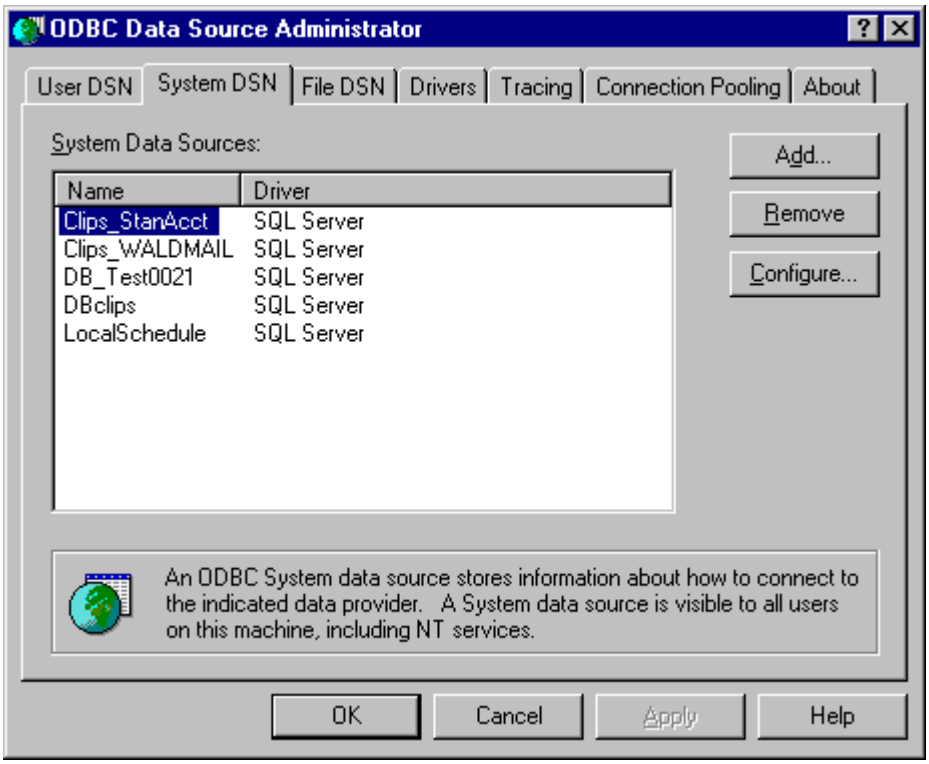
Configuration of SQL data sources

The Media Data Gateway Application publishes details of the OmniBus database to an SQL database. The Filer-Fax can search this SQL database using the Sharer as a go-between. The OUI sends in a request, which the Sharer converts into an SQL request. The Sharer fires this request at the SQL database and receives the response, adds some details to the results and passes them back to the OUI.

To allow this, the Sharer must be able to find the SQL database. This requires different configurations for different database communication protocols.

Configuring the Sharer to use ODBC

Using (in Windows NT 4) Control Panel \ ODBC32, set up a 'System DSN' for your database. This should be configured to point to the OmniBus SQL database. In the SharerOptions.ini file, go to the [ODBC] section and specify this alias as the 'DataSource' to use (see the ODBC section of the discussion on Sharer Options). Fill in the appropriate UserName, PassWord and TableName used by your SQL database.



Sharer Logs

All Sharer logs are stored in \omn\logs.

Day*nn*.txt – Error log for that day. *nn* = date. This wraps around at the end of the month, so DAY01.txt from the previous month is overwritten.

Event*nn*.txt – Sharer's event file. Key system events get recorded in here. Deletions, purging files from the dustbin, tag and untag operations.

(The DAY and EVENT logs are comma-separated files so they can easily be read into a table, such as in Excel or Word.)

TxRx*dd*[*hh*].log – Record of all network traffic coming into and going out of the Sharer. *dd* = Day of the month. *hh* – hour in 6 hour blocks. For 1st day of month following files get generated:

TxRx01[00].log

TxRx01[06].log

TxRx01[12].log

TxRx01[18].log

Only 4 days worth of network logs are kept because the files become very large. The Sharer automatically deletes any files older than that.

TraceFile.txt – Diagnostic information generated in the case of a serious error. This helps development staff to track down where the problem occurred.

The TxRx logs and TraceFile.txt logs generate information that can be used only by developers.

Interpreting the event log.

Event*nn*.txt

An ‘event’ is any major adjustment to the OMN database, for example:

- Additions to a system category (when a new engine or user, etc. is created through System Manager).
- Tags or untags.
- Deletions.
- File creation.
- Purging the dustbin.
- Destruction of a file’s data after it has been purged from the dustbin.

In its event logs the Sharer gives the following information about the cause of the event:

- The user and OUI, if known.
- The engine and application details, if known.
- Internal information on the program state of the Sharer at the time of the event.

An event can be caused:

- From an OUI by a logged on user. In this case, the user and OUI details are known but there are no engine details.
- By an application in use by a logged on user. In this case, the Sharer knows OUI details, application and engine details.
- By an application acting on its own with no logged on user directing it in which case the Sharer can supply the application details, but obviously no OUI or user details

The event log starts with a header that looks something like this:

```
29-06-1999  Sharer version : Jun 24 1999 16:25:54 Debug V 3.52
beta 1 (Patch 3)
```

This gives:

- The date stamp for the event file (29-06-1999)
- The Sharer version and time stamp for the exe build

The next line consists of column headers for the information that follows:

```
Time,Thread id,Thread handle,Thread name,User id,OUI name,OUI
IP,App name,Engine name,Engine IP,App UDP,File,Line number,event
code,message
```

Generally an event is caused either by a user through the OUI, or by a user performing some operation through an application – the Sharer endeavours to record as much as it can deduce about the source of the event. For example an application may create a file, and from its internal records the Sharer can find which OUI is logged on to that application and hence the user too.

Time	Time stamp for the event to the millisecond
Thread id	NT thread id for the thread in which the event occurred
Thread handle	NT thread handle for thread recording the event
Thread name	Internal thread name for thread recording the event
User id	If known, user id causing the event. Applications that perform operations without being logged on may be shown as 'SYS'
OUI name	If known, name of OUI user was logged on at
OUI IP	ditto for the OUI IP address
App name	If applicable, name of application causing the event
Engine name	If applicable the name of the engine causing the event
Engine IP	If applicable the IP address of the engine causing the event
App UDP	UDP port the application was running on (if the event was caused by an application)
File	Source file where the event occurred
Line number	Line number where the event was logged
event code	A code for the event
message	Plain text description of the event

The most common events and their meanings are listed below with examples of the plain-text messages.

Additions to a system category

When a new engine or user, etc. is created through System Manager.

```
Entry 'fred' (type 2) added to category '/USER'
('\OMN\SYSCAT\USERS.') User 'SYS.' Original request :
'/USER,fred,SYS.' Path allocated : '\Omn\SYSDAT\A\A\03\68.'
```

A new entry has been added to the user category (i.e. a new user has been created through System Manager).

The user is called 'fred'.

The data on the user is physically stored at \omn\sysdat\a\a\03\68.

Logons and Logoffs

The following words mark the point in the event log where a user logged on and logged off respectively.

```
LOGON
```

and

```
LOGOFF
```

The other fields in the event log entry give the user and OUI involved.

Tags or Untags

When a user tags or untags a file every category the file exists in has to be adjusted.

That is, “Fred” exists in categories “C1” and “C2” and is then tagged into cat “C3” The entries for Fred in “C1” and “C2” have to be adjusted to reflect the fact that “Fred” now also lives in “C3” and “C3” has a slot for “Fred” added to it.

The Sharer adds an entry in the event log for each affected category. The word ‘TAG’ or ‘UNTAG’ at the start of the message differentiates between a category being altered because of a tag or untag. In the example below, file “Deadlock Test X” is being tagged into category “cat1”.

The entries in the event log show these categories being adjusted: “SYS”, “Deadlock Test Cat 1”, “WEEK28” and “1999”. Category “cat1” has the entry added.

```
TAG.  Entry 'Deadlock Test X' (path '\Omn\A\A\03\67') to be
altered in category 'SYS' (cat ref 3, path '\Omn\CAT\A\A\00\25')
(category is affected by (un)tag operation on this file)

TAG.  File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be added
to category 'cat1' (cat ref 5, path '\Omn\CAT\A\A\03\28')

TAG.  Entry 'Deadlock Test X' (path '\Omn\A\A\03\67') to be
altered in category 'Deadlock Test Cat 1' (cat ref 9, path
'\Omn\CAT\A\A\03\31') (category is affected by (un)tag operation
on this file)

TAG.  Entry 'Deadlock Test X' (path '\Omn\A\A\03\67') to be
altered in category 'WEEK28' (cat ref 10, path
'\Omn\CAT\A\A\03\32') (category is affected by (un)tag operation
on this file)

TAG.  Entry 'Deadlock Test X' (path '\Omn\A\A\03\67') to be
altered in category '1999' (cat ref 16, path '\Omn\CAT\A\A\00\69')
(category is affected by (un)tag operation on this file)
```

Note that if a file is tagged into a category it already exists in, the Sharer will still go ahead and perform the operation, also generating entries in the event log.

Deletions

When a file is deleted into the dustbin you will see an entry similar to this:

```
File 'Deadlock Test X' path '\Omn\A\A\03\67' (cat ref 9, file ref
2) succesfully DUSTBINNED by user SYS
```

If the Sharer places an entry like this in its event log, the file has been dustbinned with no errors. The Sharer *never* dustbins a file without specific instructions to do so from a user or an application.

File creation

When a file is registered with the OmniBus system the Sharer takes the following steps:

- Allocates a path for the file's data (e.g. \omn\a\a\23\43)
- Adds entries to required categories.
- Adds the '*#nnn*' suffix if the file name has already been used.
- Informs the application registering the file of the file's name and path.

An entry like this is generated in the event log file:

```
File 'CARLTON TRAIL - All You Need' (registered as 'CARLTON TRAIL
- All You N#10') created at path '\omn\A\H\11\48' in categories :
CLIP PhilG WEEK26 1999
```

Following information is encapsulated in this entry:

- Client asked to register a file called "`CARLTON TRAIL - All You Need`"
- Sharer actually registered '`CARLTON TRAIL - All You N#10`' because the name had been used before and '#10' was the next available suffix for this name.
- File was created in categories 'CLIP', 'PhilG', 'WEEK26' and '1999'
- File was allocated path '`\Omn\A\H\11\48`' which is where the data for this item will be stored.

If problems arise with clips not being successfully created, it is important to remember that if the Sharer creates an entry like this in its event log as far as it is concerned the file has been successfully registered with the OmniBus system. I.e. the file has been added to the necessary categories and a folder has been created for it on the server. It is then up to media control applications to save the clip data in the folder assigned to the clip.

If an application asks the Sharer to register a file in a non-existent category, the Sharer creates the category.

File rename

Entries like these indicate a file rename. Remember that the Sharer may add '#nnn' to the new name if it has already been used.

```
File 'Satellite-12.45.03-30\06' (path '\Omn\A\H\12\14') in
category 'CLIP' (ref 13, path '\Omn\CAT\A\B\31\25') renamed to
'Dave's big one', client supplied 'Dave's big one' as new name.
```

Purging the dustbin

When a file is purged from the dustbin you will see an entry like this:

```
File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be removed from
category 'DUSTBIN' (cat ref 1, path '\Omn\CAT\A\A\00\16')

File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be removed from
category 'SYS' (cat ref 3, path '\Omn\CAT\A\A\00\25')

File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be removed from
category 'cat1' (cat ref 5, path '\Omn\CAT\A\A\03\28')

File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be removed from
category 'Deadlock Test Cat 1' (cat ref 9, path
'\Omn\CAT\A\A\03\31')

File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be removed from
category 'WEEK28' (cat ref 10, path '\Omn\CAT\A\A\03\32')

File 'Deadlock Test X' (path '\Omn\A\A\03\67') to be removed from
category '1999' (cat ref 16, path '\Omn\CAT\A\A\00\69')

Deleted entry 'Deadlock Test X' (\Omn\A\A\03\67) from DUSTBIN

Dustbin emptying completed. Attempted removal of 1 items.
```

- First entries are a record of the file's removal from the categories it existed in (file and category paths given).
- An entry recording the file's purging from the dustbin in the DUSTBIN_LOG database.
- Then an entry to say the file (file path given) has been totally removed from the filing system.
- After the time (measured in days) specified by SharerOptions.ini there is an entry saying the item has been removed from the DUSTBIN_LOG database.

When the dustbin has been completely purged, an entry is added to the log to say how many entries the Sharer tried to remove. It says, "attempted" in case any files failed to be removed from the bin: if this were the case, a note would be made in the error log.

The Sharer will *never* empty the dustbin without specific instructions from an application: usually System Manager, or the Filing Assistant.

Destruction of a file's data after it has been purged from the dustbin

Some time after a file has been purged from the dustbin the physical data for that file is destroyed. The time delay depends on one of the following:

- EmptyBinFolderDestroyDelay in SharerOptions.ini.
- Soft exit causes pending destructions to happen immediately as the Sharer closes down.

```
Removed directory '\Omn\A\A\03\67' following purge of file
'Deadlock Test X' from DUSTBIN (delay of 300 seconds)
```

Destruction of 'dangling directories'

The delayed destruction of files noted above occurs because the Sharer keeps an in-memory list of folders due for destruction. If the Sharer crashes, or there is a power outage, or a user issues a hard exit then the Sharer loses its chance to perform a delayed destruction of the folder. However, when a file is purged from the DUSTBIN the Sharer writes a file called !DESTROY_AT into its folder. This file gives the time of deletion for the folder. A low priority background task constantly scans the OMN directory for !DESTROY_AT files. If any are found and indicate a time that has already passed the Sharer destroys the folder.

```
Directory '\OMN\A\A\03\35' destroyed by background housekeeping
(was due for deletion at 16-07-1999 13:39:28.353)
```

The DUSTBIN_LOG database

The Sharer records deletions from the dustbin for the duration specified in SharerOptions.ini. This allows applications which use the filing notification mechanism to trace activity during system downtime.

Tidying a category

When entries are removed from a category (by untagging, or purging a file from the dustbin) dead space is left. New entries are not slotted into these spaces – they are just appended to the category.

Tidying a category means removing all dead space from it.

When the Sharer tidies a category it exclusively locks access to it, tidies it, and then unlocks it.

In this example, the numbers in brackets are how many entries in total and how many live entries. In this example, the category had nine dead entries to remove. On a busy site, a category will typically have dozens of entries to remove, perhaps thousands if tidying is not done frequently enough.

```
Locked cat at '\Omn\CAT\A\A\00\69' (1999) for tidying (33 entries
- 24 in index).
TidyCatByPath unlocking cat at '\Omn\CAT\A\A\00\69' (24 entries -
24 in index) 10 ms since startup.
```

Creation of a category

```
Category 'hhhhhh' created at path '\Omn\CAT\A\A\03\75'
```

Path is where the category is stored. It is made up of two files called DATA and INDEX. These two files record all the entries for the category.

Deleting a category

The Sharer automatically tidies a category before deleting it.

It then destroys the data that is used to store that category, then removes the category from its category of user categories.

```
Locked cat at '\Omn\CAT\A\A\03\72' (zzzzzzzz) for tidying (0
entries - 0 in index).
TidyCatByPath unlocking cat at '\Omn\CAT\A\A\03\72' (0 entries - 0
in index) 0 ms since startup.

Removed directory '\Omn\CAT\A\A\03\72' when removing category
'zzzzzzzz'
Low level removal of cat 'zzzzzzzz' Original request
'zzzzzzzz,SYS'
```

Categories can only be deleted when they are empty, and then only through System Manager. The category may be recreated if a file registration requires it.

The Sharer will never remove a category unless specifically told to by a client.

Common Error Codes

A discussion of all the errors that may be generated by the Sharer is beyond the scope of this document, but main error codes and sample messages follow.

Only plain text is given.

The logging of errors marked with an asterisk is controlled through SharerOptions.ini.

Startup : 'error' code 799001

```
OmniBus Sharer started
Jul 13 1999 13:42:43 Debug V 3.60 alpha 1
Machine name : 'ADAM'   NT domain name : 'OMNIBUS'
OmniBus domain : ''
Windows NT 4.0 Build 1381 Service Pack 4
```

Shut down : 'error' code 799000

Soft and hard exit have same code. The text of messages differentiates them.

Soft exit looks like this:

```
Soft exit
Sharer shut down by user request.
```

Hard exit looks like this:

```
Hard exit
Sharer shut down by user request.
```

Sharer crash : 799002

The Sharer has suffered an unrecoverable software error (sometimes caused by events beyond the Sharer's control such as hardware failure).

The Sharer will terminate.

Entries in the error log look like this:

```
********** SHARER CRASH ********** ********** SHARER CRASH
**********
Sharer exception error c0000005 at instruction 4673d7
```

Debug assertion failure: 799003

Only debug versions of the Sharer can generate this error.

This indicates that an internal safety check (for example, the integrity of data sent to the Sharer by a client) has failed. Usually, the Sharer can handle debug assertion failures without a crash resulting.

An assertion failure will look something like this:

```
********** SHARER DEBUG ASSERTION type 2 :
H:\SharerVOB\Sharer3\cpp\Main.cpp(1824) : Assertion failed!
```

This information may be of use only to a developer, but it may indicate the source of a problem – the entry may help to point out that a machine at an address is sending invalid data to the Sharer.

Sharer security handler triggered : 799012

The Sharer has suffered an unrecoverable software error known as a 'buffer overwrite'.

The Sharer will terminate.

If a Debug version of the Sharer is running, then the stack trace (tracefile.txt) will contain valid information about the location of the crash in code. This should be sent to Omnibus Support.

Entries in the error log look like this:

```
********** SHARER SECURITY ERROR type 1 : A fatal 'security error'
(buffer overrun) has occured - the sharer must stop.
```

700002 : Disk is full

Self-explanatory. You may need to delete large logfiles, or other large files on the disk to create space to keep the Sharer running. Contact OmniBus Support.

*700003 : File not found

Client application asked the Sharer to open a file (an NT file, not an OmniBus file) but the file did not exist. In this example, a client application asked the Sharer to send the first 1300 bytes of \omn\a\h\10\31\info but the file was not there.

```
Failed to open file '\Omn\A\H\10\31\INFO' to fetch from in
update_file_fetch.
Attempt to read 1300 bytes, file '\Omn\A\H\10\31\INFO', pos 0
failed.
```

This error can be caused innocuous actions like popping up the info box in the Filer-Fax. This is because the OUI asks the Sharer for the notes on the file whether they exist or not. This error is not logged by the settings in the default SharerOptions.ini

700004 : Directory not found

Client asked Sharer to open a file but the directory did not exist.

In this example, the OUI has asked for the access flags on a file, but somebody has manually deleted the directory from underneath the Sharer.

```
Open \Omn\A\A\03\59\!ACCESS failed.
Error reading access flag file (!ACCESS) in \Omn\A\A\03\59 for
SYS.
Error getting access flags from path \Omn\A\A\03\59.
```

700005 : Bad path

NT has certain requirements for folder names in its filing system. Asking for data from a folder with illegal characters in its name produces this error. The result is the same as 700004.

700009 : Bad parameters

Usually generated by an application sending the Sharer unexpected values for parameters.

700024 : Array bounds overflow

Most commonly seen when trying to tag a file to over 16 categories. Also seen when performing the data_dump_category function as part of the normal operation of that filing call.

700030 : OUI details not found in OMN Dbase

User logged on from an OUI that is not registered in the database.

Use System Manager to register the OUI.

```
Note : Your OUI (at 1.4.4.25) has not been registered in the
OmniBus database.
```

700040 : Engine hasn't responded to Sharer's poll during log on from OUI

A user selected an application from the OUI. The Sharer polled the application to see if it was still alive. The application failed to respond.

This can be caused by an engine dying or being turned off just after the user has selected an application. The error can also be caused when two machines had the same IP address.

```
No response to gen_runleaf and report_listapps from engine
Lavdrus' PC at 1.4.24.20 when running app Logo Miles.
```

*700079 : Zero bytes file read

Requests to read zero bytes from a data file can be seen as a waste of network bandwidth. Applications now frequently ask for an entire file from the Sharer, in which case this error will be generated spuriously whenever the 'entirety' of a zero length file is requested. This error is not logged by the settings in the default SharerOptions.ini

700082 : File not found in category

An application asked the Sharer to find information on a file in a category, but the file doesn't exist in the category. Sometimes applications already know the location of a category on the server (i.e. that CLIP lives at \omn\cat\a\a\32\45) so these errors often refer to the category by path, not name.

In this example an application asks for details on file 'Restore Virtuals' in the category at path '\omn\cat\a\b\31\62' :

```
Category,file : '\\OMN\Omn\CAT\A\B\31\62,Restore Virtuals'. Error
finding file.
```

700089 : File not found in index

Same error as above. This error is usually returned when a client has searched in a category by its name, not its path.

In this example, a client has asked the Sharer to tag ‘z-kill me now’ from CLIP into another category, but the file does not exist in ‘CLIP’:

```
Error loading file 'z-kill me now' (ref 0) from cat 'CLIP' (ref 0)
to start tagging operation
Error tagging file 'z-kill me now' ref 0 into 1 categories
Error performing tag / untag operation.
```

700094 : Can’t untag files from all its cats

A file must exist in at least one category. In this example a user has "ORed" all categories into his filter and then tried to untag a file from them all.

```
File 'ST4' left with no categories if 3 categories were untagged.
Operation abandoned.
Error untagging file 'ST4' from 3 categories
Error performing tag / untag operation.
data_tag_files_by_ref : Function failed
```

700095 : User tried to delete a file but didn’t have the rights

In this example GUEST has tried to bin a file and can’t:

```
User GUEST not allowed to bin file ST2 at path '\Omn\A\A\03\21'
Error binning file ST2 in category cat2 for user GUEST
update_bin_single_file failed. Cat ref 6; file ref 2; search
string : GUEST,,
```

700115 : Binned files are not accessible

If FilingFindIgnoreBinnedFiles in SharerOptions.ini is switched to YES (the default) then searching in any category for a file that has been binned causes an error. In this example, an application has searched (by path) the category at \omn\cat\a\b\31\25 for 'MUSVID - Barbarus. Hold O#1' – the Sharer returns an error because the file is in the dustbin.

```
Category,file : '\\OMN\Omn\CAT\A\B\31\25,MUSVID - Barbarus. Hold
O#1'. Error finding file.
Search resulted in a binned file : (ref 1130) in category
'\\OMN\Omn\CAT\A\B\31\25'
```

700127 : Can't untag from base type

If BaseUntag in SharerOptions.ini is OFF (the default) then attempting to untag a file from its base category is an error. In this example a user has tried to untag 'Adrian Clip One#5' from category 'CLIP' but the file is of type CLIP so the Sharer doesn't allow it:

```
Cannot untag file 'Adrian Clip One#5' at path '\Omn\A\G\73\37' of
type 50 from its base category 'CLIP'
Error untagging file 'Adrian Clip One#5' from 2 categories
```

700142 : Filing Notification requests cancelled due to engine time out.

Applications can register interest in the Sharer's filing operations. They can request the Sharer to inform them of most types of changes to the database and can specify tight ranges on what they are interested in. Applications like the Media Data Gateway use this facility to stay synchronised with the OMN database.

If an engine fails to respond to the Sharer's polling and it has open FN requests an error like this one is caused:

```
1 FN Requests cancelled due to engine 'Media Data Gateway' (IP
1.0.1.10, UDP port 48648) dying / timing out.
```

*700147 : Invalid cross-reference file found

Generated by the Sharer when applications are reading files after the file has been dustbinned. This is frequently seen when the EmptyBinFolderDestroyDelay is set and so is not logged by the settings in the default SharerOptions.ini

790001: Filing request from unknown IP address

This error is generated when a filing request is made of the Sharer from an unregistered OUI or engine. No engine or OUI that is not registered in the database can request information from the Sharer.

```
Node is not registered with the system. File request refused.
```

Allocation of file paths

When a user saves a file (clip, config file, etc) the filing system must allocate a new folder for the data. The Sharer allocates a path of the format: \omn\a\a\32\34.

A type id is passed to the Sharer. This corresponds to file types defined in the CONFIG file. Each type corresponds to a path defined within the \OMN directory.

Example defs from CONFIG file:

```
FILE_TYPE 0 \Omn    # Default directory
FILE_TYPE 1 \Omn\CAT  # Index files
FILE_TYPE 2 \Omn\SYSDAT # SYSTEM DATA FILES
```

If no type ID is passed the directory path begins with \Omn.

Files of type 2 get stored in \omn\sysdat; a few other FILE_TYPE commands in the CONFIG file govern other types.

A unique directory path is then tacked on to the end of this. NXTFILE is used to determine the next path name to allocate.

To take a typical path of the form \omn\A\B\01\02 the ranges for each part of the path are as follows:

\OMN	\ A .. Z	\ A .. Z	\ 00 .. 76	\ 01 .. 76

The maximum path, therefore is "Z\Z\76\76". This gives a maximum potential of 26*26*77*76 file names = 3,955,952 paths.

When the Sharer creates a folder for a user file it also creates several other files which make up the file's meta data:

!XREF details on which categories the file exists in. A rebuild tool is available which can recreate every category by using information in these files. (Category corruption can be caused by power outages. There are no known software problems that would damage a category.)

!ACCESS access flags for the file

!DESTROY_AT as noted above, a temporary file indicating when the directory should be destroyed, following a purge of the dustbin.

No user should ever touch any of the above.

Common questions about filenames

Why do my file names get altered?

The OmniBus Filing System allows file names to be up to 29 characters long. File names that exceed that are trimmed. This will be shortened for foreign language file names.

Why does '#1', '#2' and so on get added to my file names?

The OmniBus Filing System allows multiple copies of the same file to be registered. This can be very useful if you wish to save many files without having to invent a new name for every save.

For example, if you save a file called "My clip" and then save it again, the system will automatically append '#1' to the second file (renaming to 'My clip#1'), '#2' to the next and so on up to '#999' at which point the system will not let you save any more files with that name.

File name padding

SharerOptions.ini allows configuration of PadInstanceCount. If set to ON, #001, #002, etc. are used instead of #1, #2, etc. This allows alpha-numeric sorting to list 'hashed' clips in ascending numerical order.

Why are files with different names altered?

The side effect of the above is that the system has to consider file names which match up to the 25th character to be identical, even if the last few characters are different.

The following file names are all considered to be "AMV/BTPB300/030 - BT RECO" by the system:

File name	Length
"AMV/BTPB300/030 - BT RECO"	25
"AMV/BTPB300/030 - BT RECON"	26
"AMV/BTPB300/030 - BT RECONN"	27
"AMV/BTPB300/030 - BT RECONNE"	28
"AMV/BTPB300/030 - BT RECONNEC"	29
"AMV/BTPB300/030 - BT RECONNECT"	30

So if you registered these files this would be the result:

File name entered by user	Registered as
"AMV/BTPB300/030 - BT RECO"	"AMV/BTPB300/030 - BT RECO"
"AMV/BTPB300/030 - BT RECON"	"AMV/BTPB300/030 - BT RECO#1"
"AMV/BTPB300/030 - BT RECONN"	"AMV/BTPB300/030 - BT RECO#2"
"AMV/BTPB300/030 - BT RECONNE"	"AMV/BTPB300/030 - BT RECO#3"

If the above rule was not enforced this example shows what could happen:

File name entered by user	Registered as
"This is test file number 1"	"This is test file number 1"
"This is test file number 1"	"This is test file number #1"
"This is test file number 11"	"This is test file number 11"
"This is test file number 11"	"This is test file number #1"

The system would consider "This is test file number 1" and "This is test file number 11" to be different.

When the '#1' suffix was stamped on to the end of the second copies of those files duplicate file names would result.

How do foreign languages fit into this scheme?

Foreign languages may be used in file names and category names. They are stored using a compressed form of unicode that we have christened *iuchars*.

Even if a file name is wholly in a foreign character set the '#nnn' at the end will be in Latin.

How can I reset the counter at the end of my file names?

Place *every* instance of the file in the dustbin and then empty the dustbin using the System Manager application. If this doesn't seem to work, logically OR every category except DUSTBIN into the Filer-Fax and check there are no other instances of the file name left in other categories.

Why do attempts to register a file fail sometimes?

Converting back to Latin at the end of a string and adding the '#nnn' suffix adds extra characters to a filename. There is a system-wide limitation of 29 characters per file. If the extra characters needed to register multiple instances of a file cause it to overflow this limit the registration will be rejected.

Why can't I use the '#' character?

The '#' character is a reserved character. If a file with '#' in its name is sent to the Sharer it will be zero-terminated at the point the '#' occurs.

Why can't I use the ',' character?

The ',' character is a reserved character. If a file with ',' in its name is sent to the Sharer it will be rejected.

What characters will the System Manager allow?

The System Manager has slightly tighter rules for which characters it will allow in user names, engine names etc.

The System Manager history file details this.

How many characters may be used in a file name?

Some of the points covered in this section have been dealt with in "Why do my file names get altered?" However, this section provides a little more detail on what the current limitations of the filing system are.

English file names

Up to 29 characters may be used in a file name.

Files that match up to the 25th characters are considered identical as noted above. Therefore, long file names may have '#1' etc added to them, even if the last few characters differ.

Foreign language file names

Storing foreign language file names takes significantly more space than English file names. There is an overhead of two characters for every language change, and since the default language is English, every foreign language name loses at least two characters before it is typed.

i.e. "<Cyrillic>abcd" uses 6 characters.

Spaces exist only in the English character set, so every space in a foreign language string costs five characters: 2 to switch to English, the " ", and 2 characters to switch back.

i.e. "<Cyrillic>ab cd" uses 11 characters.

The Sharer matches file names up to the 25th character, including "hidden" characters used for switching between languages, and has to allow for extra characters to switch back to English when adding '*#n'* on to a file name. Therefore, foreign language names will be trimmed and have '*#n*' added at shorter lengths than English names. Usually this will occur at the 23rd character.

So (allowing 5 characters for a space) foreign language file names can be up to 27 characters long; and names that match up to the 23rd character will have '*#n*' added to prevent duplicates.

Day-to-day Operation

Once it is set up and running, the Sharer does not require user intervention. If something goes wrong, Appendix A gives some tips on how to resolve the situation.

Usually when all is well, users don't notice the Sharer's operation. However, the Sharer will forcibly log people off applications, or off the system, under some circumstances:

- If a logged on user is deleted, the user will be logged off forcibly from all applications, and their OUI.

This avoids the situation arising where a user is logged on, but has no ability to use the system properly. In previous versions of the system, the user would remain logged on, but be unable to log on to new applications. They would also be unable to perform any actions on the system, which would require knowledge of the user's access rights (e.g. loading, creating, or deleting clips).

- If an engine is deleted whilst in use the user will be forcibly logged off the application (not off the system.)
- If a logged on user's OUI is deleted, he will be forcibly logged off the system.
- If an application is deleted anyone logged on to it is forcibly logged off.

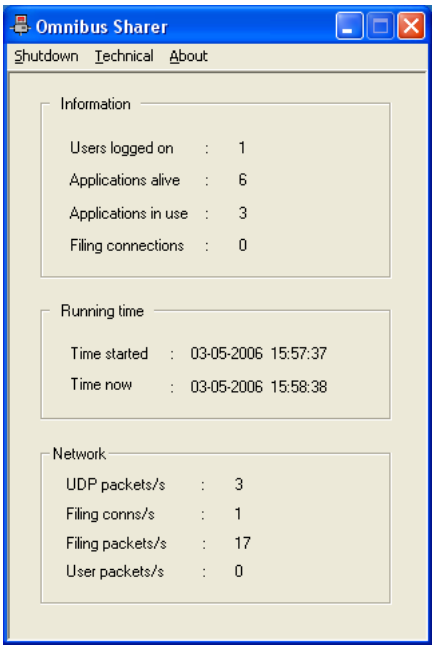
Under other circumstances, it is advisable for a user to log off then log on again:

- If a user's set-up is altered to make an application available to him, he should log off the system and then log on again.

In other cases, no action is required after a system change:

- If a new engine is added to the database, or an existing engine is altered via the System Manager, this will be picked up immediately.

Front end



The Sharer is not a user application like Word, or Excel. It is a message processor acting on behalf of the rest of the OmniBus System. The front end, therefore, is minimal.

On the front end you will find information in three sections:

Information

- **Users logged on**. The number of users currently logged on to the system.
- **Applications alive**. The number of live applications around the network; this includes 3rd party applications, and the local applications on the OUI task bar.
- **Applications in use**. The number of applications in use is the number of applications that are logged on to.
- **Filing connections**. When a client application requires filing services from the Sharer, it connects to the Sharer over the network. Filing connections gives the number of such open network links.
  This number will typically be quite low (0 – 10) but can briefly climb higher if many applications around the network are making demands of the Sharer.
  If the number of filing connections surpasses the number of applications alive and stays that way, client applications are opening connections to the Sharer and leaving them open. This can happen if engines are shut down in the middle of operations involving the Sharer. The Sharer should eventually time these connections out, but you should consider restarting the Sharer at a quiet moment to terminate these excess connections. The need to do this will be rare.

Running time

The start time and current time.

Network

The section tracks how many messages per second are being streamed through the Sharer's network event handlers. There are four counters, reflecting the four different event handlers. Certain actions in the system (starting the Sharer, broadcasting the Sharer's location, emptying the dustbin etc) trigger higher levels of network traffic than normal.

Because systems vary from one site to another, it is not possible to know what an 'unusual' threshold for each of these values is.

- **UDP packets/s**. The number of UDP messages directed at sharerport each second. This is expected to get relatively large when the Sharer is broadcasting its availability.
- **Filing conns/s**. The number of incoming TCP filing connections directed at filerport each second. Each connection made will lead to a count in the "Filing connections" GUI count. This is expected to get relatively large when the Sharer is broadcasting its availability, or when files are modified or deleted.
- **Filing packets/s**. The total number of incoming TCP filing requests directed at all TCP filing connections each second. This is expected to get relatively large when the Sharer is broadcasting its availability, or when files are modified or deleted.
- **User packets/s**. The total number of incoming TCP packets pertaining to logged on Users. This is expected to get relatively large when the Sharer is broadcasting its availability.

Menu options

Shutdown

Soft Exit

Attempts to cleanly shut down the Sharer: an attempt is made to ensure no lost user data in filing operations by gracefully terminating all connections.

You will see the Users logged on decrease, then the filing connections counter will decrease. After this, the Sharer will terminate.

A dialogue box is put on screen whilst the Sharer is closing down. This informs you that the Sharer is attempting to close down cleanly. You have the option to allow the Sharer to do so, or instruct it to abandon the attempt to close down cleanly and just die.

OK = closes informational dialogue box; Sharer continues to shut down gracefully.

Die now = hard exit

Pressing the close box in the top right corner of the Sharer Dialog Box also causes a soft exit:

Hard Exit

Terminates the Sharer immediately. No attempt is made to break off connections gracefully, or to preserve user data. There is a very small chance that OMN database corruption could occur if this option were used while someone was using the Sharer to save a file.

Technical

Record Net Messages

Toggles on and off.

Is initially set according to the value of RecordNetMessages in SharerOptions.ini

Records every network message processed by the Sharer to \omn\logs\TxRxdd[hh].log where dd is the date and hh is the hour.

Only the last four days network log files are kept; earlier files are automatically deleted by the Sharer to prevent too much disk space being used by these files.

Four network log files are created per day, every six hours. So during a day TxRxdd[00], TxRxdd[06], TxRxdd[12] and TxRxdd[18] are created.

Log Application Errors

Toggles on and off.

It is initially set according to the value of RecordAppErrors in SharerOptions.ini

Records any application errors in the network logs (if switched on, see above) and in the DAYnn error log. Since the Sharer can receive many application errors from around the network, the DAYnn can become very large. Thus, some users may feel the need to switch this logging off. The down side of this is that diagnostic information is lost, which may have proved useful.

Force Manual Poll

Causes the Sharer to broadcast a request around the network for engines to reply with their status. The Sharer does conduct a rolling poll of every engine on the network; this option is to be used if you have cause to believe that the Sharer does not have an accurate picture of the state of the network. (I.e. difficulty logging on to an application that you know is running and is free).

Broadcast Sharer Location

The Sharer broadcasts its location to all engines and OUIs around the network. Use this if it appears that engines / OUIs do not know which IP address the Sharer is on.

Cause Access Violation

Causes the Sharer to perform an illegal operation, resulting in Windows NT shutting it down. This option is used to simulate the software crashing and test the Sharer's generation of diagnostic information.

The option should not be used by users, unless they are running the Sharer under the Watchdog and wish to ensure that the Watchdog restarts the Sharer if it crashes.

Cause Debug Assertion Failure

Causes the Sharer to perform a non-fatal, illegal instruction. This is used to test the Sharer's generation of diagnostic information.

This option should not be used by users.

Cause Buffer Overrun

Causes the Sharer to perform a fatal buffer overrun. This option is used to simulate the software crashing and test the Sharer's generation of diagnostic information.

This option should not be used by users.

Debug

Causes the Sharer to trigger the system debugger.
This option should never be used by users and is not available in the release version.

About

Displays Sharer version number and information on the operating system.

Sharer Maintenance Guidelines

Tidy All Categories:

Do this about once a week (although the exact frequency should be determined with OmniBus depending on the turnover of clips in the system). As mentioned above, categories keep growing; and dead space created by deleting an entry from a category, either by untagging it or by purging it from the DUSTBIN, is not reused. It must be reclaimed by tidying the category.

To do this, go into the SYSTEM menu of the System Manager application and select the FILING menu. The *Tidy Category* and *Tidy All Categories* buttons can then be found. Select *Tidy All Categories.*

You may wish to tidy very busy categories (i.e. CLIP, DUSTBIN, *year*) more regularly. It is best to tidy categories at times when the system is slack.

De-fragment the Hard-Drive:

See the note below on defragmentation.

Check the Omn\Logs\ folder for excessive files taking up drive space:

Nothing in this folder is essential and all these files can be deleted even whilst the Sharer is running. The Sharer can record every network packet it receives and transmits. This type of information can be very useful when trouble-shooting site problems, but can lead to huge log files.

The debug version of the Sharer has this option switched on by default. The release version has it switched off by default.

To turn net message recording on or off use this menu option from the Sharer's front end: Technical / Record net messages, or the [Logging]RecordNetMessages switch in SharerOptions.ini.

Logging of application errors can be controlled using the Sharer's front-end menu option: Technical / Log application errors. Its default value at start-up is controlled by the [Logging]RecordAppErrors setting in SharerOptions.ini.

Back up regularly

Omnibus database backup procedure (every 24 hours)

As a backup will become out of date the instant that the files are updated we recommend you perform regular **full** backups. The backup interval is at the discretion of the user.

What to back up

The files that need backing up are stored on the RAID drive.

On a Level 1 system this is the C:\ drive; on a Level 2 system, the X:\ drive

The directories are:

<drive:>\OMN*.*

<drive:>\OmniBus*.*

Ensure that the <drive:>\OMN\Log*.* is de-selected as the files stored here are very large and do not need backing up.

Backing up with the Sharer running

You can back up the Sharer while it and the System Manager are running. However, there are a few things to note:

1. Locked files (such as the Sharer and System Manager EXE files, NEXT, INDEX and CAT files and any data files that are in use) will **not** be backed up.
2. The data will be stale as this type of backup can only take a snapshot of the files at the time that it backs them up.
3. The OmniBus rebuild tool will take care of any inconsistencies in the OmniBus database that are created due to the omission of locked files.
4. To restore any applications that are not backed up, the simplest method is to re-install them from the original InstallShield programs and then copy the backed up '.ini' files over the top.
5. You should perform backups when the Sharer is least active; i.e. when dub processing and clip requests are at a minimum.

Backing up with the Sharer not running

This is the ideal situation as you can back up the complete database without loss of data and it will be more up-to-date.

Backup location

You can copy the OMN directory to a network hard drive with ArcServe (Advanced or Enterprise editions).

You could also use a standard IT back-up solution – for advice on this, contact OmniBus Support.

Using Open File Agents

Open File Agents are not necessary as the OmniBus rebuild tool has been enhanced to be capable of re-constructing the OmniBus database based on the actual files that were backed up.

Those database files that are not captured will be picked up on subsequent backup operations.

For an example of how to perform a back-up, see Appendix D: Using CA ArcServe for Backup/Restore.

Improving access time for the OmniBus Database

The OmniBus database is typically stored in an NTFS partition. NTFS volumes suffer from fragmentation and this can have a very heavy impact on the Sharer's performance.

The answer is a disk defragmenter. We have used Executive Software's Diskeeper to good effect. One operator, after using this software, reported a ten-fold decrease in the time taken to display the CLIP category in the Filer-Fax.

Executive Software themselves can be found at http://www.execsoft.com/

UK web site is http://www.execsoft.co.uk/

Ensure you order the correct version – NT workstation, or server. Be especially careful to check the version if the defragmenter is to be applied to a RAID volume.

Disabling DOS file names

If possible, disable automatic "8.3" (short MSDOS) file name creation. In the registry, change:

"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem\

NtfsDisable8dot3NameCreation" to 1.

Testing has shown "8.3" names slow large directory access considerably.

However, this will disable access to these directories for Win3.1 and DOS clients if the file names are long. For most users this will not be anything to worry about.

What if the Sharer won’t run?

There follows a brief list of errors that can stop the Sharer from running.

"Failed to get Windows Version information. NT Error code N.

Note down this error message and inform OmniBus Systems."

The Sharer failed to find out which version of Windows NT was running. Ensure the operating system is Windows NT 4 Service Pack 3, or above. If it is, contact Support.

If N=2138, you may have put the Sharer or Watchdog in the Startup menu. This will not work on most systems. Contact OmniBus Support for information about having the Sharer start automatically at logon.

"This program can only run on a 32-bit Windows platform.

NT is the recommended operating system. Sorry."

or

"This program was designed for Windows NT Version X.xx

This computer is running Windows 95 Version Y.y

Performance is, therefore, NOT guaranteed to be stable."

or

"This program was designed for Windows NT Version X.xx

This computer is running Windows NT Version Y.yy

Performance is, therefore, NOT guaranteed to be stable."

The Sharer will not run on anything under NT 4 + SP 3

"Failed to get startup network information.

NT Error code N.

Note down this error message and inform OmniBus Systems."

The NT machine does not have the Workstation service enabled in its network settings. From the control panel, choose the Network applet. Under the Services tab, ensure the Workstation service is installed. Under the Bindings tab, ensure the Workstation service is enabled. If it is, contact OmniBus Support.

" Error reading in \omn\SharerOptions.ini"

or

"Filing system cannot initialize.

The Sharer CANNOT RUN."

The Sharer can't find enough of the OmniBus System database to allow it to run. Check that the \omn directory contains SharerOptions.ini and CONFIG. Ensure the CONFIG has its read only attribute DISABLED. Check the contents of the two files are plain text using Notepad.

"Error *n* when trying to read setting '*setting*' of section '*section*' in config file:

'\omn\SharerOptions.ini'

Setting was invalid: *value*

Sharer CANNOT RUN."

A field in SharerOptions.ini is invalid. Check the setting.

"Invalid CONFIG file. Replace '\OMN\CONFIG' with correct version.

The Sharer CANNOT RUN."

An older version of \OMN\CONFIG is present. This should not be a problem if the Sharer has been correctly installed since the correct version of CONFIG will be copied over then.

"OmniBus domain name conflict in setup: new 'a'; previous 'b'

Check .ini file setting, referring to system documentation. Sharer CANNOT

RUN."

The Multi-Domain options in SharerOptions.ini clash with those set with System Manager 3. Comment out the "OmniBusDomainName" option in SharerOptions.ini.

"Error verifying config files:

DomainNamePrefix was set ON but OmniBusDomainName was not set.

No valid OmniBusDomainName was found in \OMN\DOMAINS\DOMAINS.

Sharer CANNOT RUN."

The Multi-Domain options in SharerOptions.ini clash with those set with System Manager 3. Set a valid domain to be 'This domain' in System Manager 3, or include a valid OmniBusDomainName setting in SharerOptions.ini.

"Sharerport UDP socket failed on startup. Error *code* …"

The Sharer cannot acquire the network resources it needs to run.

Caused either by:

- Another instance of the Sharer being frozen in memory (thus holding net resources)
  Check task manager for Sharer3.exe; if there is a copy of the application you may need to "End process" it if it won't shut down properly.
- TCP/IP services not set up correctly on machine.
  Is the TCP/IP protocol actually installed?
- Is the <winnt>\system32\drivers\etc\SERVICES file configured for use by OmniBus?

This is usually done at install time. But if you remove and re-install TCP/IP for any reason this file will be overwritten by a default.

NB: After successful installation, the most common problem when the Sharer won't run up is another copy still running in memory.

What do I do if the Sharer does something wrong?

A full description of the problem is necessary, especially if the problem is repeatable. Supplying error logs (and preferably network message logs as well) will allow us to trace the problem quickly.

Get the following information, and then contact OmniBus Support.

- A detailed description of the problem (including error information from the OUI or application if possible).
- Time that the problem occurred, name of clips, categories involved, as well as any paths you may be aware of (i.e. from the information button in Filer-Fax you may know the path of a file)
- Applications involved (including version information).
- OUIs and Engines used (particularly the IP addresses).
- Information on the files and categories involved (including any category details of the files).
- As many of these as possible : DAY*nn*.txt, Event*nn*.txt, TraceFile.txt, TxRx*dd*[*hh*].log.

Crashes

Although testing on the Sharer is rigorous, as with any software system there are bound to be problems. Sometimes these problems cause the software to crash.

The word ‘crash’ does not refer to hardware problems such as a failed disk, faulty SCSI cabling that takes a RAID system off line, etc. If you experience problems and use the word ‘crash’, we assume a software problem. This can confuse the issue if the problem is really hardware. A RAID system, for example, does not crash: it goes offline.

Terminology:

When software is still apparently running but not responding it has *frozen.*

When software has behaved illegally and has been terminated by the OS, it has *crashed*.

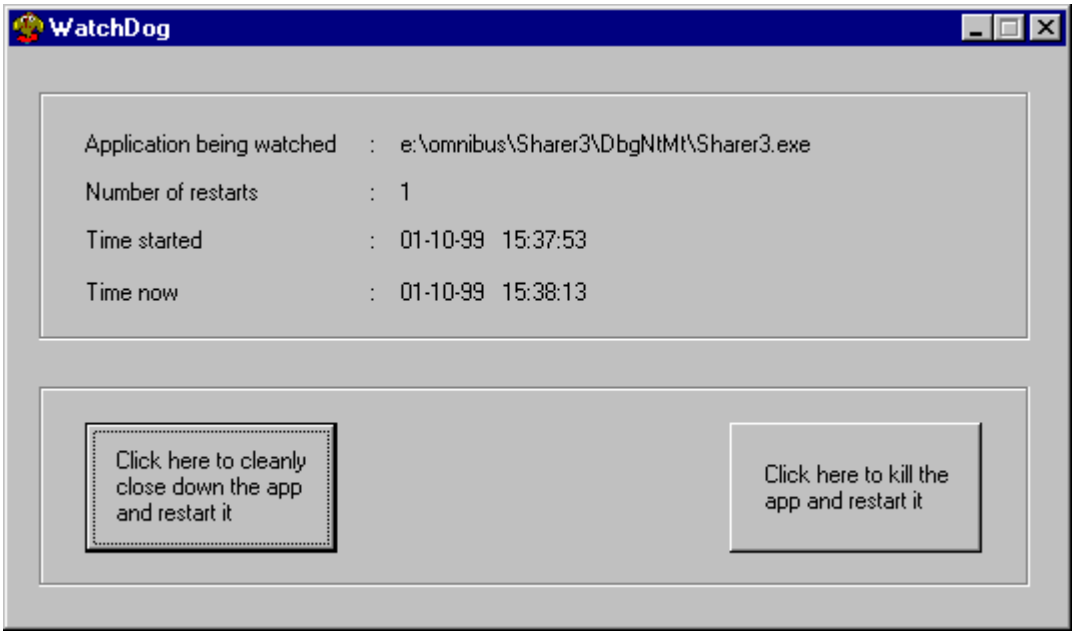
It is much more common for the Sharer to crash than freeze. Sometimes it looks as though the Sharer has frozen because a client has made an unusually heavy demand of it: this situation rights itself as the workload reduces. Whether to terminate and restart the Sharer under such circumstance must be decided at the time and no rules can be made.

Note that a hard termination of the Sharer may cause corruption of the OmniBus database, although the risk of this is very small.

If the Sharer is simply started up from the desktop, then when it crashes it will just be terminated by the OS. It can take a few minutes to realise that this has happened, go to the machine and manually restart the Sharer.

In the event of a crash the Sharer generates a file in \omn\logs called TraceFile.txt. This file contains diagnostic information about the crash and should be sent, along with as many other logs as practical, to OmniBus.

The Watchdog



The Watchdog application is designed to minimise problems due to crashes. It is a simple application that runs up the Sharer and watches it. If the application terminates for any reason, the watchdog restarts it. It can be used on any application, but has one Sharer specific feature.

Using the Watchdog to restart the Sharer in the event of a crash can significantly increase reliability by making most abnormal terminations invisible. The Watchdog is not a part of the Sharer in any way and shares no source code with it. It is a very simple application that is stored completely separately from the Sharer in the computer's memory.

The Watchdog cannot detect a Sharer freeze. The user must make the decision to restart the Sharer if it appears to have frozen, with a hard termination if nothing else works.

Setting up Watchdog

To manually set the Watchdog up to run and watch the Sharer:

- create a shortcut to the file watchdog.exe
- call up the properties box for the shortcut (right click, choose 'Properties' from the menu)
- Choose the shortcut tab
- The target field should have the path of the application to watch added after Watchdog's exe path. That is, if Watchdog is installed at x:\omnibus\watchdog\rel\watchdog.exe and the Sharer is at x:\omnibus\Sharer3\dbgntmt\Sharer.exe then the target field would read: x:\omnibus\watchdog\rel\watchdog.exex:\omnibus\Sharer3\dbgntmt\Sharer.exe
- Start in should point to the omn directory i.e. "x:\omn"

When you run the watchdog up it will automatically start up the target application.

A shortcut for the Watchdog to run the Sharer is created at install time, so the above steps should not be necessary

Once the shortcut has been set up, double click on it to start the Watchdog, and the Sharer. The Watchdog starts the Sharer itself, you do not need to start the Sharer. The Watchdog does not attach itself to an already-running instance of the Sharer: it tries to run up a new instance. So, ensure the Sharer is not running before starting up the Watchdog.

If the Sharer does crash, the Watchdog will automatically restart it.

The Watchdog will restart the Sharer if it stops for any reason, however. Some users have become confused after the following trail of events:

- Start the Watchdog (which starts the Sharer).
- Shut down the Sharer for some reason.
- Watchdog restarts the Sharer: minimised so it doesn't appear on the desktop.
- User tries to start the Sharer, not realising the Watchdog automatically did this as soon as they shut the Sharer down.
- 2nd instance of Sharer refuses to run and user believes the software is broken.

To shut down the Sharer under the Watchdog you need to:

- Shut the Watchdog down.
- Shut the Sharer down.

The Sharer will keep running if you shut the Watchdog down, but it will not restart if it crashes. The Watchdog will not reattach itself to the Sharer if you run it up, it will try to run up a new instance. You will need to shut the Sharer down, and then run up the Watchdog.

There are two buttons on the Watchdog:

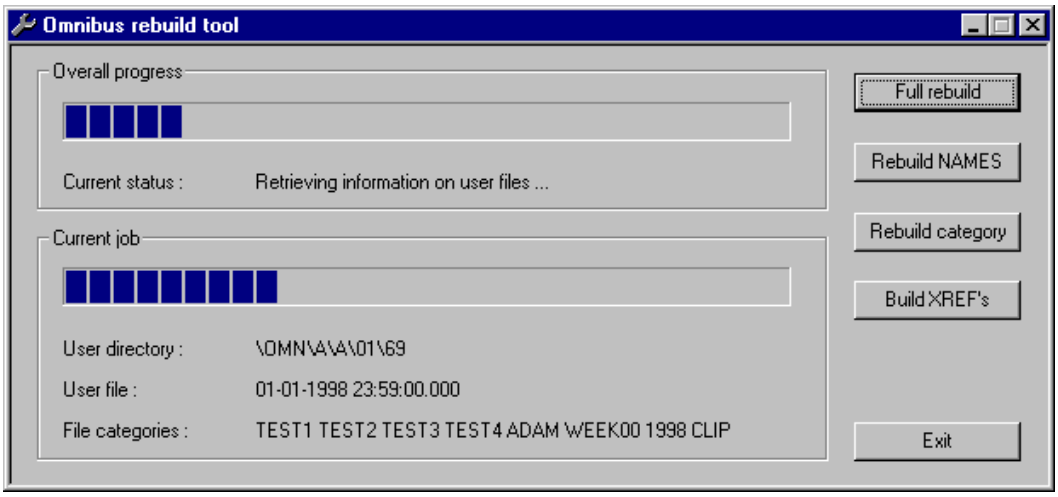
The button marked "Click here to cleanly close down the application and restart it" works only for the Sharer; its effect on other applications is undefined. This button performs a soft exit on the Sharer, giving it the chance to cleanly close down.

The button marked "Click here to kill the application and restart it" will work with any application. It performs a hard exit on the Sharer.

Other information on the front end of the watchdog is:

- Full path to target application being watched
- Time watchdog was started
- Current time
- Number of times target application has been restarted

Appendix A: The Rebuild Tool



Occasionally the OMN database can become corrupt. This is manifested by:

- Applications failing to load files because the underlying category structure has been compromised.
- The Filer-Fax fails to display the contents of a category.
- '*#nnn*' continues to be added to a file name, even if all instances of the file have been placed in the dustbin and the dustbin purged.

This can be caused by one of:

- Versions of the Sharer lower than 3.40 could not cope with a full disk; database corruption resulted.
- Power outages.
- Old applications not registering files correctly. Such applications should be upgraded.

The OmniBus rebuild tool can usually cure even severe problems with database corruption. It is designed to work with V 3.6*x* of the Sharer.

It should run on the same drive as the OmniBus database, or should have its start-up directory pointed at the OmniBus database (from the properties of its shortcut).

Users should only use this tool if they know what they are doing, and it should never be used while the Sharer is running.

A full backup of the data in the OMN directory is recommended before using this tool. Although the chances of causing any damage to the database using this tool are remote.

There are five buttons on the Rebuild Tool front end:

- Full rebuild
- Rebuild NAMES
- Rebuild category
- Build XREFs
- Exit

Full rebuild

Assumes all user categories are damaged. The whole category system is rebuilt using information from !XREF files found in \omn*.*

If you have trouble with several categories, use this.

Rebuild NAMES

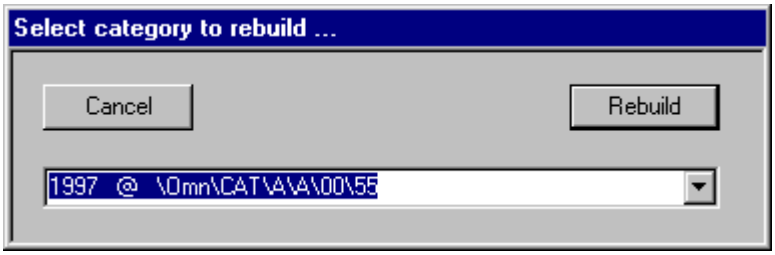
Rebuilds only the NAMES category.

User categories (i.e. categories like 'CLIP', '*year*') must be sound for this option to work.

If you experience difficulties with the '*#nnn*' suffix being added on to files when it shouldn't be, then a problem with the NAMES index is the cause.

This scans all user categories and makes a note of all file names in the database. From this information, NAMES is rebuilt.

Rebuild category



Rebuilds a single category.

Presents you with a list of all categories in the system and the paths they live at on the disk. Select the category to rebuild and hit Rebuild.

Scans all user categories (except the category being rebuilt) to determine which files in the system belong in the rebuild category.

If you know that a damaged category contained files which existed only in that category this method will fail (because references to the files will not be found in other categories) and you will need to use Full Rebuild instead.

Build XREFs

Used when upgrading from a version of the Sharer below V 3.36

!XREF files contain rebuild information that can be used in case of severe damage to the database. Versions of the Sharer below V 3.36 did not record this information. This button scans through the whole database, creating an !XREF file for each file in the OmniBus database.

After using this option you should select Full Rebuild to complete the process.

Exit

Exits the tool.

Appendix B: Upgrading

Upgrading Sharer versions

Use the install program to install the Sharer and related software: necessary shortcuts will be created in Program Files \ OmniBus.

The database will need some conversion after the Sharer has been upgraded. Using the Rebuild tool perform one of the following actions, depending on which version of the Sharer was running:

Below V 3.36	Build XREFs then Rebuild All
Sharer V 3.36 – 3.40	Rebuild All
Sharer V 3.41 and above	no action needed

Converting an Acorn OmniBus System Database

A system that had been running on an Acorn under OmniShare will need some conversion work to be usable under Sharer 3.

Acorn to NT (Sharer 1 to Sharer 3)

The conversion from Sharer 1 running on a RISC PC to Sharer 3 running under Windows NT is quite complicated, and involves a considerable transfer of data.

How long does it take and what effects will it have on end users?

The timescale for this operation depends almost entirely on the amount of data in the database. We suggest allowing a full day for copying large databases. The Sharer will be off line for the entire time, and will start on a new physical machine, with all of the data being stored on a new physical storage device.

What is the procedure?

- Tidy all categories on the Acorn and delete $.OMN.LOGFILE.
- Put the !WinServ application onto the Acorn. In !WinServ.Options ensure that the section [TypeMaps] does not contain an entry for 'xxx = Data'.
- Map a drive to Root on the Acorn by IP address
  i.e. \\1.2.3.4\ROOT in the "map network drive" dialog box of Explorer.
- Copy the contents of \\<acorn>\Root\OMN to \OMN on the new server. This can take a very long time. When this is complete, a copy of the OmniBus Database is on the new server. Unfortunately, none of the OS specific cross-references within the database are valid for Windows NT yet.
- On the new server, run up IanFS. The data has to go through several conversion steps to be usable on the new platform.
- Many files in the database may have paths starting with <omn$>: this will need altering to $.omn. Use the ALL_PATH_PREFIXES command to alter any paths beginning with "<omn$>" to "$.omn".
- The database is now full of category entries that refer to Acorn directories for their folders. Use the CONVERT command to alter the database to NT's format.
- Put the latest version of the configuration file for NT into the OMN directory on the DEC.
- Alter \OMN\NXTFILE :

  ```
  A.A.00.21□21
  ```

  needs to be :

  ```
  A\A\00\21
  21
  ```

- Install and run the appropriate version of the Sharer.

Profile Video Disk Recorder Issues

GVG Profile™ may give you problems because it stores its clips locally using the path allocated by the Sharer. Under NT, all registered data will have the same path structure but will be different because of using NT path descriptions instead of ADFS paths.

I.e. "$.OMN.A.A.34.45" is the name of a clip on Profile, but is now known as "\OMN\A\A\34\45" and will not be recognised by Profile.

There is a conversion utility within Profile to address this problem. It relies on the tail end of the clip path being directly off \omn – i.e. \omn\a\a\00\01 not \omn\clip\a\a\00\01 and so the DATA_TYPE statement within the Config file should never be used to alter where profile media gets stored. It must always be directly off the OMN directory. (If no changes are made to the configuration file this will normally be the case anyway.)

How will the system be affected (Sharer 3 vs Sharer 1)?

Differences in the way the dustbin works

The old Sharer would untag a file from every category it existed in when a file was placed in the dustbin. The file would then be left existing only in the dustbin and, if it was untagged from there, would be inaccessible.

The new Sharer leaves a file in all its categories and tags it into the dustbin as well. Simply untagging a file from the dustbin leaves it as it was.

Emptying the dustbin will cause Sharer 3 to remove such a file from all its categories and then destroy it, as well as broadcasting to the network that this file is now dead.

Why can't I log on?

If you have trouble logging on under the new Sharer, log on directly to the System Manager and ensure that the user is configured as having a valid user group and home area.

Resave the user data anyway using the Modify button in the user setup screen. Older versions of the System Manager used to save slightly corrupt user data, which the new Sharer rejects.

What other applications will be affected by the change to Sharer 3?

Apart from the issues mentioned above:

- Sharer 3 does not support the Filer-Fax on OUI 1.
- Sharer 3 does not support the PortfolioTx Archive Unit.

System Manager 1 to System Manager 3

How long will it take and what are the effects on end users?

The procedure will take a few minutes to complete. Modifying the database after completion may take an hour or so, depending on how many engines are registered in the database.

- The Sharer should be restarted after conversion.
- The way in which users are allowed to log on to applications is different under the new system (see the section on making applications available). Each engine has a list of all applications that may run on it.
- Any errors in the Filing System found during conversion should be dealt with immediately after conversion, using the low-level filing application IanFS if necessary. System Manager 3 may be unable to delete some entries until this is done.

What is the procedure?

The application “System Data Upgrade (1 to 3)” should be used to convert System Manager 1 systems to System Manager 3 compatible systems.

- Install the Sharer. Make sure you copy across the new CONFIG file and copy the relevant options from the old SharerOptions.ini to the new SharerOptions.ini.
- BACK UP \OMN\SYSDAT*.
- Using System Manager 1, add Application “System Manager 3” (!app_man3).
- Make sure all Users, Engines OUIs and Applications have valid areas and groups.
- Run up System Manager 3.
- Log on (you haven't converted the database yet, so the Sharer lets you log on under the old rules) and add System Manager 3 to the System Manager engine's available applications, so that after the conversion, you won't have to log on directly. Save the data using the Modify button. The engine is saved as new version.
- Log off System Manager 3. If you try to log on normally, the Applications list will not show System Manager 3 as available - the Application is old version (created by old System Manager) the Engine is a new version (just modified by System Manager 3).
- Run the conversion application - see the instructions enclosed in the ZIP. It needs an Intel PC connected to the network. Everything is now new version.
- Log on to System Manager 3 again - to make sure you can under the new rules.
- Check SYS
- Make sure SYS has the four OUI applications in/out of the User's Available Applications list (as appropriate, given the Allow/Disallow flags). Select Allow and Disallow to change SYS’s data twice (back to its original state).
- Log on to your OUI again as SYS, to make sure the 4 local applications come up.
- Add all the correct Available Applications to each Engine in turn.
- Remove application System Manager (!app_man) from the system database.

How will the system be affected?

What is the main difference between the two programs?

System Manager 3 maintains tighter control of the OmniBus system database. The potential for users to enter incorrect or potentially damaging information is reduced.

Extra functionality is being introduced into the OmniBus system. Extra data on machines, applications and users is required for these improvements. This extra data is only accessible through the new System Manager 3.

System Manager 3 does not rely on direct access to the storage media of the OmniBus database to function. The programme may run on different hardware to the Sharer if necessary, so the potential for any failure in System Manager to cause the Sharer to fail can be reduced. The Sharer's multithreaded file access ensures that no errors are introduced to the filing system by System Manager.

Other differences

The availability of applications is governed by slightly different rules (see the section on making applications available).

Emptying the dustbin and tidying categories are also under the control of the Sharer. When emptying the dustbin while using System Manager 3, the user may terminate the operation part way through. When tidying categories using System Manager 3, already tidy categories are left unchanged (a very quick operation). When tidying all categories, the operation may be terminated part way through. The operation will cease after the tidying of the current category is complete.

Records of System Manager 3 actions are now held in the Sharer's logs. The logfile features of System Manager have been removed.

The "install fonts" option has been replaced by a generic "install" procedure in System Manager 3.10. System Manager 1 "install fonts" should be used for systems with System Manager 3.0x.

The availability of OUI-based applications is governed by applications in the user's list of available applications in System Manager 3.1x. This provides a generic interface for the availability of any applications.

Multidomain and Object Configuration are only available in System Manager 3.

Appendix C: Miscellaneous Questions

My 'CLIP' category (or any other category) doesn't work. Can I recover it?

See rebuild tool (Appendix A)

Why can't I put a file in the DUSTBIN?

Check the file's access rights. The delete flag may not be enabled. If it is not, you will not be able to delete the file. If any of the delete flags are not enabled, turn them all on.

If you are not the file's owner and are not logged on as "SYS" you may not be allowed to alter the file's access rights.

If the above doesn't work try un-tagging the file from some of the categories it exists in. When a file is dustbinned it is tagged into the DUSTBIN category and not untagged from any categories it already exists in. If the file already exists in 16 categories (the maximum permissible in the system) the dustbinning attempt will fail.

I'm having problems with low level operations using IanFS

IanFS is a low level utility for manipulating the OmniBus database. Great damage can be caused to the database with improper use of this tool. It should only be used under the direct supervision of OmniBus technical support staff. These notes are mainly aimed at on site support staff rather than users.

Most problems can be solved using the Rebuild tool (see Appendix A) rather than this utility.

How do I slow down the display so that I can read it?

To pause the display on an NT-based system press the "Pause" key (top right of the keyboard), and any other key to continue.

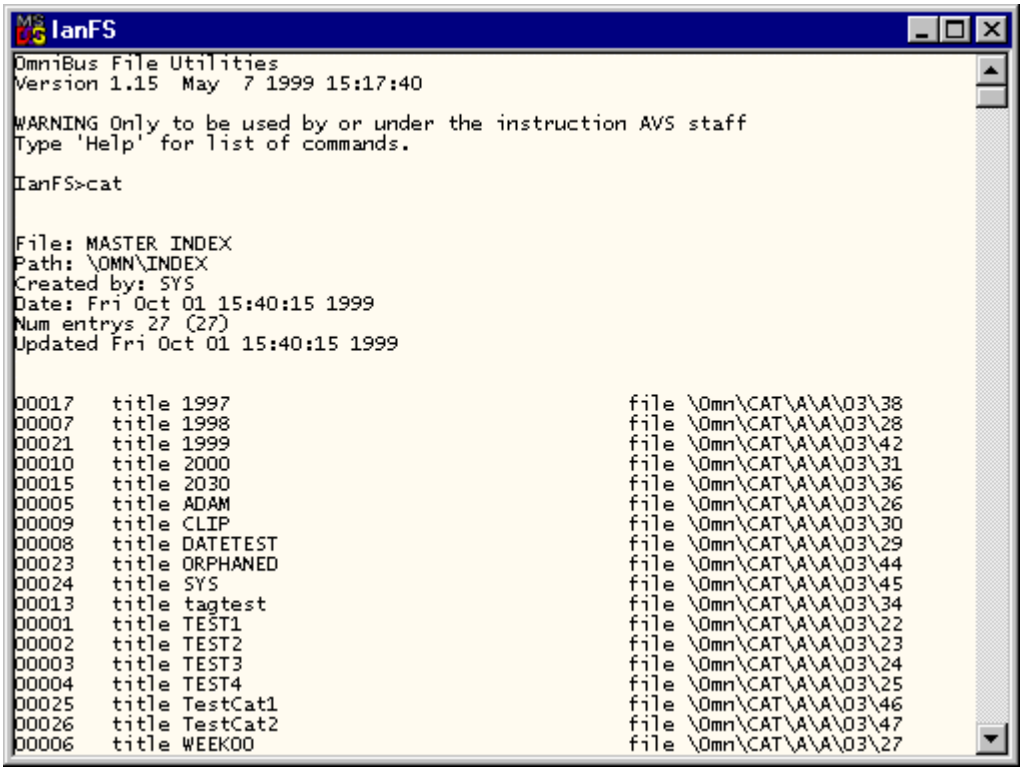
Alternatively, alter the Command Shell's display properties to give a larger screen buffer. This allows you to scroll up and down the output of your Command Shell. Use the 'Console' Control Panel Applet to configure all Command Shells to behave like this, or right click on the Command Shell's title bar to get at the properties of that Command Shell. From the 'Layout' tab, alter the 'Screen Buffer Size' 'Height' to be (say) 1000 to get 1000 lines of output to scroll through.

How do I remove an invalid category from the OmniBus database?

NB. This method is to be used ***only*** when the usual method of removing a category fails.

Only use this method when the system is not live.

- Run up the IanFS utility
- Type "cat" and hit enter.
- On an NT based system the display will look something like this :



- To pause the display on an NT-based system press the “Pause” key (top right of the keyboard), and any other key to continue.
- The numbers on the left are reference numbers for the categories in the system. Note down the reference number for the category you wish to delete.
- When IanFS has listed all the categories type “remove_item” and hit ENTER.
- IanFS responds with “Enter Directory Path : “
- On an NT system, type “\OMN\INDEX” and hit ENTER.
- IanFS responds with “Enter index ref : “
- Type in the reference number you noted down earlier for the category to be deleted and hit ENTER.

On an NT system the display will look like this:

```
00001  title WEEK46      file \Omn\CAT\A\B\32\51
00016  title WEEK47      file \Omn\CAT\A\B\32\52
00009  title WEEK48      file \Omn\CAT\A\B\32\53
00018  title WEEK49      file \Omn\CAT\A\B\32\54
00268  title WEEK50      file \Omn\CAT\A\B\32\55
00011  title WEEK51      file \Omn\CAT\A\B\32\56
ok list done

IanFS>remove_item

Enter Directory Path :\omn\index

Enter index ref :11
```

```
Ok

IanFS>
```

The dead category has now been removed.
Remember, only use this when the system is not live.

How do I remove an invalid file from a category?

NB. This method is to be used ***only*** when the usual method of removing a file fails.
Only use this method when the system is not live.

- Run up the IanFS utility
- Type "cat" and hit enter.

On an NT based system the display will look something like this :

```
IanFS>cat


File: MASTER INDEX
Path: \OMN\INDEX
Created by: SYS
Date: Thu Jan 04 15:07:18 1996
Num entrys 145 (303)
Updated Mon Apr 07 16:52:44 1997


00002  title 1995          file \Omn\CAT\A\B\31\07
00032  title 1996          file \Omn\CAT\A\B\31\08
00273  title 1997          file \Omn\CAT\A\B\31\09
00302  title a new cat        file \Omn\CAT\A\B\44\39
00191  title Adam          file \Omn\CAT\A\B\31\10
00043  title Adam Test        file \Omn\CAT\A\B\31\11
00046  title AdamsTest        file \Omn\CAT\A\B\31\12
00280  title AlanB          file \Omn\CAT\A\B\31\13
00286  title ALFRED         file \Omn\CAT\A\B\34\14
```

The numbers on the left are reference numbers for the categories in the system.
Note down the path of the category in which the bad file exists.

- Type "list"
- IanFS responds with "Enter category name :"
- Type in the name of the category containing the bad file and hit ENTER.
- IanFS lists out all live files in the category.
  The entry to be deleted will be included in this list.

The numbers on the left of the listing are the reference numbers for the files within the category; and the system paths for the files are displayed to the right.

On an NT system the listing looks like this:

```
IanFS>list

Enter category name : 1997

File    : 1997
Path    : \OMN\CAT\A\B\31\09
Created by : SYS
Date    : Mon Jun 16 11:06:44 1997

Updated   : Fri Jun 20 11:04:09 1997

Num entrys : 2863 (2879)

23    Name : '00003'          Path : \OMN\A\B\32\66
26    Name : '00007'          Path : \OMN\A\B\32\71
67    Name : '00018'          Path : \OMN\A\B\32\76
13    Name : '00021'          Path : \OMN\A\B\32\73
5    Name : '00037'          Path : \OMN\A\B\27\68
.
.
.
7    Name : '00040'          Path : \OMN\A\B\33\63
2    Name : '00041'          Path : \OMN\A\B\33\64
No errors

IanFS >
```

- Note down the reference number for the file to be deleted.
- Type “remove_item” and hit ENTER.
- IanFS responds with “Enter Directory Path : “
- Enter the category path you noted down previously.
  I.e. for “1997” on it would be “\OMN\CAT\A\B\31\09”
- IanFS responds with “Enter index ref : “
- Type in the file reference number you noted down earlier and hit ENTER.

On an NT system the display will look like this :

```
IanFS>remove_item
```

```
Enter Directory Path : \OMN\CAT\A\B\31\09

Enter index ref :11

Ok

IanFS>
```

The dead file has now been removed.

Remember, only use this when the system is not live.

Appendix D: Using CA ArcServe for Backup/Restore

For DEC Alpha-based system server only

See the document "ARCserve v6.5 for Windows NT User Guide" (available in electronic format as ASTN65.PDF) for reference on how to use the ARCServe Software.

Note: This version can be found on the ARCserve 6.5 installation CD.

For Intel-based system server only

See the documents "ARCserve 2000, Administrator Guide" and "ARCserve 2000, Getting Started" (available in electronic format as ADMGuide.PDF and Getstart.pdf) for reference on how to use the ARCserve software.

Note: This version can be found on the ARCserve 2000 installation CD.

Backing Up

1. Run the ArcServer Manager software
2. Click on the backup wizard, which starts the backup manager.
3. Select the source tab and highlight the following directories for drive X:
   OMN – this directory contains the OmniBus database and system information. (If you want to save space on the storage destination, deselect the "Log" directory.)
   OmniBus – this directory contains the OmniBus applications; i.e. Sharer3, System Manager and the Sharer configuration file.
4. Select the destination tab and then select device group and media from the device tree.
5. Enter a description for your backup job in the description tab in the summary window.
6. Schedule your job to execute now if required.
7. Monitor progress in the Job Status Manager.
8. View the job log entry for the job when the job has completed (Job Log tab). Check to see that all files have been backed up (exceptions will be the INDEX file, NEXT file and all executable files, which will be open at the time of backup).

Note: We recommend you run regular full backup jobs, making sure the "Full, keep archive bit" is always set.

Restoring

Note: Before starting the restore, rename the OMN directory on the destination to "OMN-old" if it exists. This will ensure that no data is corrupted in the event of a failed restore.

ARCserve provides four choices of restore; OmniBus prefers "restore by tree".

Find the quick access wizard window to find the ARCserve restore wizard. The Wizard allows you to submit a job without running ARCserve manager.

1. Run the ArcServer Manager software
2. Run the Restore Wizard

3. Select “restore by tree”.
4. Highlight the OMN and OmniBus directories.
5. Leave the box ticked to restore the directories to their original location.
6. Select the destination location where the files are to be restored. This will be the RAID disk subsystem and the root of drive X.
7. Execute the job immediately.
8. Enter a description for the job.
9. Monitor progress in the Job Status Manager.
10. View the job log entry for the job when the job has completed (Job Log tab). Check to see that all files have been restored.

Rebuild The Database Integrity

As the database may not be consistent (due to locked files being missed, etc.) you must use the rebuild tool to clear up any partial entries, etc.

For details on how to do this, see Appendix A: The Rebuild Tool .

Note: If the OmniBus directory needs restoring:

1. Rename <drive:>\Omnibus\shareroptions.ini to shareroptions.ini.old
2. Locate the original InstallShield file and install this with the relevant options.
3. Delete the installed <drive:>\Omnibus\shareroptions.ini file
4. Rename the <drive:>\Omnibus\shareroptions.ini.old to shareroptions.ini