

RemoteFS

User Guide

Introduction

RemoteFS is a software & hardware package for the RISC OS operating system which allows files, directories or indeed whole filing systems on one Acorn RISC computer to be accessed from another Acorn RISC computer connected in one of several ways.

A computer running RemoteFS can be a "Client" (ie it imports/receives files), a "Server" (ie it exports/sends files), or both. Client machines can access those filing systems which have been made available from a Server machine. Multiple Clients can access the same Server, and many Servers can be running RemoteFS at once within a single site if you use the "Network RemoteFS" package.

At present, RemoteFS supports connections made using the following media:

- RS232/RS423 serial cable - (Supplied with 'Cabled RemoteFS*')
- Bi-directional parallel port cable - (Supplied with *Cabled RemoteFS*)
- Ethernet (TCP/IP or AUN protocols) - (Supplied with *Network RemoteFS*)
- Econet - (Supplied with 'NetWork RemoteFS*')

These connections are known as "Transports" and are made available to RemoteFS by loading "transport modules", each of which supports a particular type of connection. Before RemoteFS can be used it must be configured to be made aware of the types of connection available.

Only certain combinations of connection may be made between different kinds of computer. The list below summarises the possible combinations:

- Any pair of computers may be connected using the Serial transport, assuming that serial interfaces are present in both computers (remember that the A3000 does not have one as standard).
- Those computers having "Bi-directional" parallel ports can be connected to each other, viz: A5000, A3010, A3020, A4000, A4.
- Any group of computers may be connected using *Network RemoteFS' with either Econet or Ethernet transports, assuming that the relevant interfaces are present in both computers.

Software license

Your attention is drawn to the license conditions associated with RemoteFS. These are:

The "Cabled RemoteFS" package allows up to two machines to run both Client and Server software simultaneously.

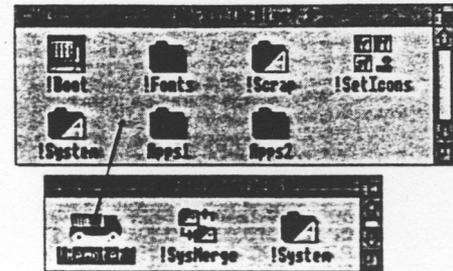
The "Network RemoteFS" package is a site license and allows any number of machines at the same location to run both Client and Server software.

Note that the RemoteFS software is uniquely serially numbered to allow tracing of illegal copies, but not copy protected. Feel free to make backup disks for your own use.

Installing RemoteFS

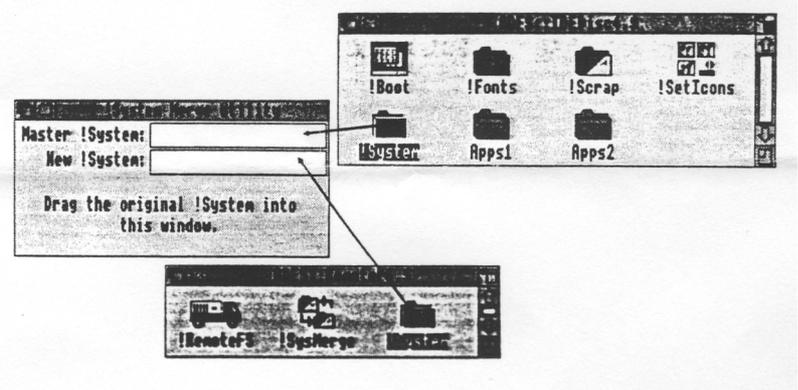
Before you begin to use RemoteFS you must install it on at least two computers, configure the software according to the kind of connection (serial cable or parallel cable) you are going to make between them, and make the connection. You will then be able to transfer your files.

To install any version of the RemoteFS package, simply copy the RemoteFS application from its floppy disc onto the target machines by dragging it from the floppy disc window to your hard disc window from within the RISC OS Desktop. The disc is not copy protected and you may make as many copies as you like of the software for backup purposes.



The RemoteFS release disc also includes copies of the latest release of Acorn's !System folder (containing important operating system modules) and a utility called !SysMerge to simplify the process of upgrading.

To use !SysMerge, double-click on its application icon within the floppy disc filer window and drag first your existing !System folder and then the new !System folder supplied into the !SysMerge window as instructed. Your !System will be updated automatically.



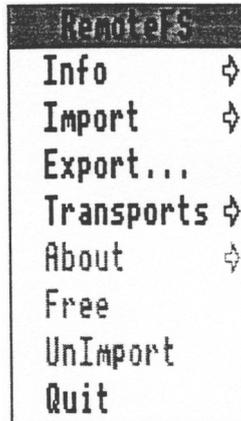
Setting up the Server

Before RemoteFS can be used from a Client machine, a Server machine must have been set up and be running. The Server needs to be configured to define which files, directories or filing systems are to be "Exported", ie made available to Clients.

To start up RemoteFS as a Server, simply double-click on the !RemoteFS application icon in the filer window where it is stored. After a few moments you will see the following icon appear on the icon bar:



Press the middle mouse button whilst pointing at this icon to display RemoteFS' main menu. It looks like this:



Here is a brief summary of the purpose of each of the menu choices:

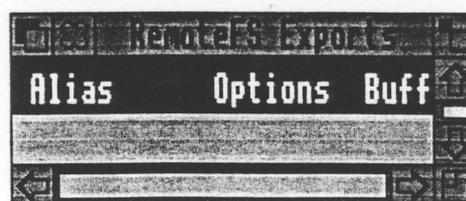
- `Info` displays information about the version of RemoteFS you are using.
- `Import` selects the Client functions; it allows remote filing systems made available by another machine to be used locally.
- `Export...` selects the Server functions; it allows local filing systems to be made available to users of Client machine(s).
- `Transports` allows the various different transport modules to be selected.
- `About` will be 'greyed out' when RemoteFS is first started and can only be selected when an import has been chosen from the `Import` menu; they provide information about the location and transport in use, as well as allowing the `Import` to be disabled.
- `Free`
`UnImport`
- `Quit` removes RemoteFS and disables all imports and exports. A warning box will appear before RemoteFS quits to protect against accidental selection.

Setting up Exports

To configure the RemoteFS Server, it is necessary to select the filing system path to be exported and a transport over which to export it.

First, bring up the RemoteFS menu by pressing the middle mouse button whilst pointing at the RemoteFS icon and click on `Export` The first time you do this you will see an empty window labelled "RemoteFS Exports" appear.

The RemoteFS Exports window looks like this:



Clicking the Menu button over the RemoteFS Exports window produces the following menu:



Here is a brief note about the purpose of each of the options.

- `Alter ' '` allows the name and access controls of the export to be altered
- `Unexport` allows the export to be cancelled, disconnecting any current clients
- `New Export` allows a new export pathname to be entered
- `Save Exports` to save the current exports configuration so that it will be available next time RemoteFS is started

Exporting an object

There are three types of object than can be exported, and three methods by which they can be exported.

Method 1

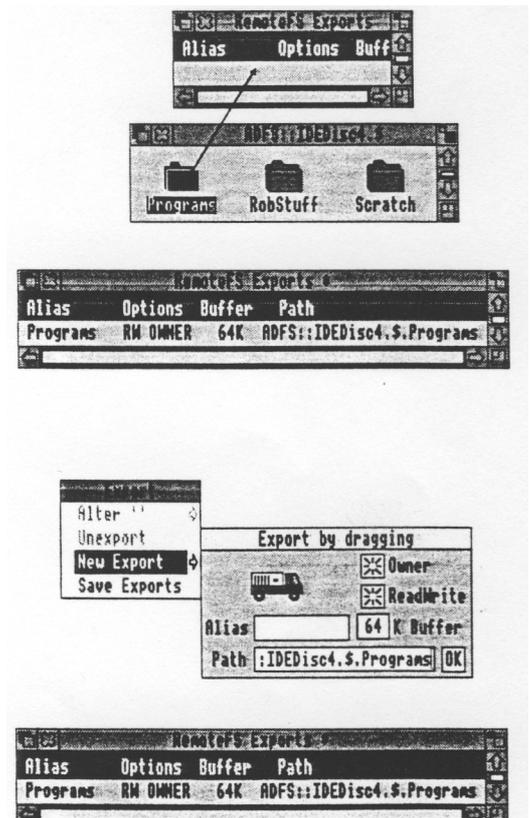
Any file or directory (except the root directory) can be exported by simply dragging its icon from the relevant filer window, into the RemoteFS Exports window.

When a file, or directory has been dragged across, its details will be displayed in the RemoteFS Export window, including its path name and access status.

Method 2

Any file, directory or root directory, can be exported by using the 'New Export' sub menu. This is accessed by moving the pointer off to the right of the 'New Export' option on the Export menu (see top of page).

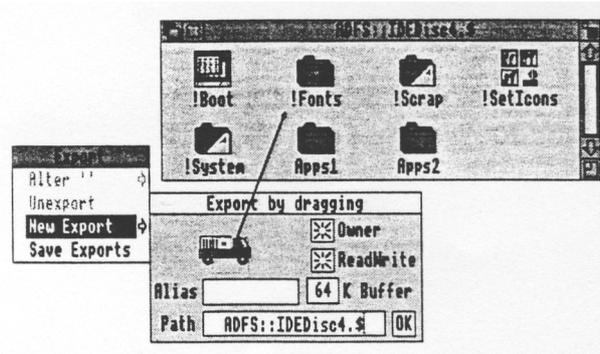
If you now type the (full) path name of the object you wish to export into the box labelled 'Path', and click on OK, the object will be exported and its details will appear in the Exports window.



Method 3

Finally, whole filing systems (ie. root directories) or directories can be exported using the 'Truck' icon on the 'New Export' sub menu window. Make sure you have the directory (root or otherwise) you wish to export, open on the desktop and then drag the 'Truck' icon across to it.

The object will be exported and its details will appear in the Exports window.



If all this sounds a bit complicated, just experiment a few times to get the hang of each one. Each method has its own advantages and you should soon get used to selecting the appropriate one.

Altering an Export's configuration

Once at least one export has been set up, it is possible to alter its configuration. To do this, select the 'Alter' option from the Export menu, making sure that the pointer is over the export to be changed when you do so. Each of the entries in the Alter menu are considered below.

Alias

This is the name by which is export is known by Client users. You can alter the name by deleting the current one and typing something new.

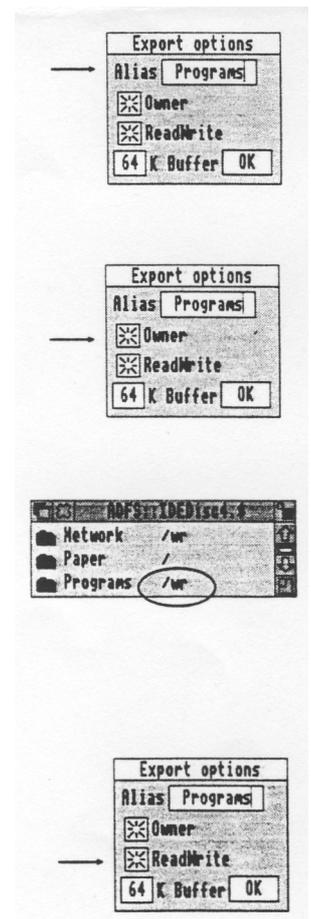
Owner

As far as a Server is concerned, there are two sorts of person: OWNER and OTHER. OWNER has access exactly as someone using the Server machine would have. OTHER is restricted by the public/private status of the file as controlled by the wr attributes after the '/'.

The 'wr attributes' refer to those shown in a normal filer window (see right). Further details of file attributes are given in the RISC OS 2 & RISC OS 3 User Guides, one of which will have been supplied with your computer.

Read/Write

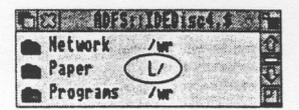
You can also export a RemoteFS path to give Client access of either READONLY or READWRITE. READONLY ensures that Clients cannot alter any files; READWRITE allows them the same access to data as a user of the Server has.



Unlike FileCore filing systems like ADFS or SCSIFS, RemoteFS pays attention to access attributes on directories as follows:

- R - needed to scan directory
- W - needed for CREATE/DELETE/Set-access on members of directory
- L - disallows CREATE/DELETE/Set-access by OTHER on members without the W bit, and prevents directories from being deleted

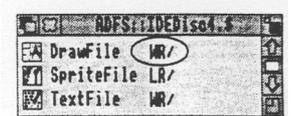
(Where 'R', 'W' & 'L' refer to access attributes displayed in a normal filer window. Further details of access attributes are given in the RISC OS 2 & RISC OS 3 User Guides, one of which will have been supplied with your computer.)



Access attributes on files act as before:

- R - needed to read object
- W - needed to write object/set load, exec or date, type.
- L - If set prevents object being deleted.

(Where 'R', 'W' & 'L' refer to access attributes displayed in a normal filer window. Further details of access attributes are given in the RISC OS 2 & RISC OS 3 User Guides, one of which will have been supplied with your computer.)

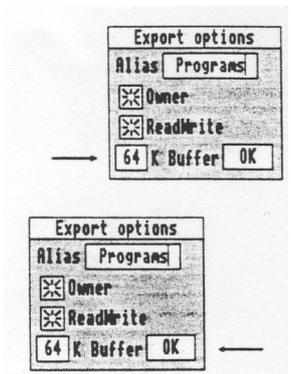


FileCore ignores all access permissions on directories. To prevent this causing total confusion, RemoteFS behaves as if the owner WR bits are always set on a directory. FileCore also doesn't allow changing the access of the root of a disc, so this is always presented as LWR/r if exported. This means that you cannot let someone not OWNER import the root of a disc in a way that allows them to create files or sub-directories in it - you have to use a directory.

Note that the control of changing the attributes of an object depends on the access to its containing directory, even if the object is the export point. You can export individual files, but the directory containing them will still need appropriate access if they are to be updated. Note also that filecore checks RemoteFS access as if it were OWNER, so clearing owner bits will prevent all accesses.

Buffer

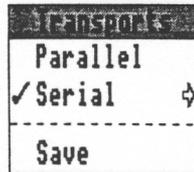
The buffer size used by RemoteFS can be altered using this menu entry. Larger buffers are likely to give faster transfers but use more memory.



After updating any export control, you MUST click on the OK button within this window or press RETURN to confirm your alterations.

Selecting a transport

Before an Export can become available to Clients, you must select a transport over which the data will be carried. To do this, choose the "Transports" option from RemoteFS' main menu. You will see a sub-menu which looks similar to that illustrated below. The exact list will of course depend on what transport modules are present in the version of RemoteFS you are running.

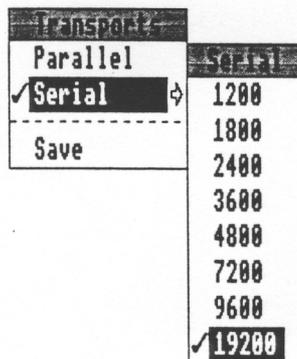


To select a transport, simply click on it with the left-hand mouse button. You may select more than one transport if you wish. Selected transports will show a tick to the left of their menu entry.

RemoteFS will warn you if it thinks that the transport you have selected is not supported by your hardware configuration. If this happens, check the interface concerned is working correctly before trying again.

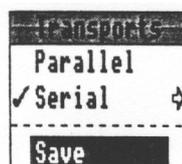
Transport sub-menus

Some transports, notably the Serial transport, have a sub-menu for controlling particular features of that transport. The Serial transport allows a 'baud rate' (speed) to be selected; the currently selected speed is ticked. To alter the selection simply click with the left hand mouse button on the new speed. It is important to ensure that the SAME speed is selected at both ends.



Saving your transport selection

To save the transports you have selected so that they will be available automatically next time you use RemoteFS, click on the "Save" option at the bottom of the Transports menu.



Setting up the Client

To use RemoteFS at a client machine, you must first ensure that the transport you are planning to use is available on both the Server and the Client; ie that not only is the relevant hardware (if any) present, but also that the cable is plugged in. Once you are satisfied that the chosen transport is available, you are ready to configure the Client. Follow these steps to do this:

Start up !RemoteFS by double-clicking in the usual way and call up the main menu. Click with the left-hand button on the "Transports" option and select the transport you wish to use as described above. Finally, bring the main menu up again and slide off the "Import" entry to the right in order to reveal its sub-menu. Select the transport required by clicking with the left-hand mouse button.



If the transport is available and has a RemoteFS Server running at the distant end with at least one export, after a few moments a window similar to that shown will appear. In this example, the directories 'Paper' & 'Programs', plus the file 'Memo' have been imported.



Double clicking on an individual file will load that file, whilst double clicking on a directory will open a directory filer window and also change the RemoteFS icon to show that an import has taken place. If several directories are exported by the Server over this transport, then they will each display an icon on the icon bar.



The RemoteFS filing system icons behave almost exactly like "real" filing systems; you can perform any kind of operation on them you like. Details of using RISC OS filing systems are given in your computer's Welcome Guide.

Further reading

You should now be familiar with RemoteFS and how to get it up and running. If you have any problems, try going through the relevant section again and don't be afraid to experiment. Do, of course, make sure you backup your RemoteFS disc, and then use the backup disc in future. Further information regarding filing systems and their general use can be found in the Welcome and RISC OS guides, supplied with your computer.

Technical Information

The rest of the manual contains information regarding more advanced use of RemoteFS.

Using RemoteFS from the command line

It is necessary to instruct the Server machine to allow access to exported filing systems. Do this using the *Export command as follows:

```
*Export <alias> <path>
```

<alias> is the name that will be used to access these files on the remote machine. It should be short and descriptive. - something like "IDEDisc4" is good. <path> is the actual path to a directory that will be accessed, such as "ADFS::IDEDisc4.\$".

Once you have done this and assuming that a transport module is running on both machines and the RemoteFS module is loaded on the Client, you should now be able to access remote files by name using a path along these lines:

```
Remote#serial::IDEDisc4.$myfile
```

Use *D1R to select a directory of this form and then try cataloguing with *CAT, for example

You can use the *Export command (see below) at the Supervisor "*" prompt (press F12) to tell the RemoteFS Server to allow client access to files, just as if you had set up an export from within the Desktop.

How RemoteFS works

When you use RemoteFS on your machine (called the Client) to access files on another machine (the Server), there are three separate software modules involved.

RemoteFS, running on the Client, provides a full RISC OS filing system interface, to let you access files just as you would from any other filing system. It converts the calls from the RISC OS FileSwitch module into "request blocks", that are sent across to the Server machine.

This sending is done by a "transport" module, which understands the particular method you have used to connect your machines together. This module must be running on both machines involved. There are transports available for serial & new (bi-directional) parallel ports (supplied in 'Cabled RemoteFS'), plus TCP/IP (Internet) & AUN/Econet (supplied with Network RemoteFS).

"RemoteDaemon" is the Server module. This lives on the Server machine, and when requested to do so by a transport, takes the request block and performs the file access as needed.

There is one extra module, "RemoteFSFiler" that supplies the filer icons and menus, to let you open Filer windows on RemoteFS inside the desktop.

RemoteFS transports

The following section provides information peculiar to each RemoteFS transport.

RemoteDaemon - "local"

This is not actually a separate transport. It is merely the Server pretending to be a transport, so that you can find out exactly what you are exporting easily from the desktop.

RemoteSerial - "serial"

You can change the baud rate by running it as `"/remserial 9600"`. Make sure both ends are matched. Notice that this module is using the serial port all the time that it is loaded, and will conflict with anything else that tries to use it.

RemoteParallel - "parallel"

You need the bi-directional parallel port that is fitted to newer machines to use this, and a special cable available from Atomwide. You can not run RemoteFS over the parallel port if you have a dongle attached to either parallel port, eg the Impression & Artworks dongles.

RemoteInternet - "internet;machine"

The machine name can be either a numeric Internet address or a symbolic name assigned in your hosts file (see TCP/IP documentation.) A menu of some possible machines is created dynamically by broadcast messages from machines set up as Servers, or you can enter a machine name in a writable menu item.

RemoteEconet - "econet;machine"

The machine name that a Server uses to identify itself is the name of its first export. (In the same way as Level 4.) These names are broadcast so that a menu is built up in each Client machine, or you can enter a specific econet address into a writable menu item.

Image filing systems

Image filing systems are interpreted on the Client - in fact the Server presents them as files, and FileSwitch takes over. This will mean that only one Client at a time will be able to access a given image, and it must have appropriate software available to do so.

RemoteDaemon

```
*Export <alias> <path> [READONLY|READWRITE] [OWNER|OTHER] [nK]
```

Allows object to be accessed remotely. The options modify the access that people importing it are allowed. nK is the amount of memory to use for buffering disc access. It should be smaller for

RemoteFS User Guide

slow media, such as floppy, to improve the disk/transport overlap. The Server will claim two buffers totalling to the size given whenever it needs them to perform a request. The system variable `Export$Options` is scanned before the command line to set up any default options.

```
*UnExport <alias>
```

Stop allowing remote

```
access. *Exports
```

Lists exported objects.

RemoteFS

```
*Import <alias> <transport-detail>
```

Sets up the association between a given import and a transport. This causes the filer icon to appear on the desktop, and allows you to use `remote::<alias>...` to access it. It normally happens automatically whenever you use a special field to specify the transport.

```
*UnImport <alias>
```

Removes the association. Note that if something is still accessing it using an explicit transport, it will only be re-established again.

```
*Imports
```

Lists the currently know imports.

```
*RemoteTimeOut
```

Allows you to set the timeout period in seconds for which RemoteFS caches directories. After this length of time, filer windows will be checked in case files on the Server have changed. The directory contents are only thrown away completely after five times this period, so normally the check back with the Server is very quick. However, if you wish to disable it, set it to zero.

RemoteFiler

This presents the bar icons for imports under the desktop. It should be noted that the 'Free*' option will only produce an interactive display on RISC OS 3.1, due to deficiencies in RISC OS 3.0.

Compressed files and RemoteFS

The commands that are sent between your RemoteFS client and your RemoteFS server act directly on the data in the files. It would be possible for the transport layer (serial, parallel etc.) to perform compression on your data and doing so would be perfectly within the specification. However, at present, none of the transport layers do this. There are a number of good reasons for not compressing the users data.

Data compression is based on a branch of computer science called information theory. Information theory tells us that there is a certain amount of information in a file and the quantity of information in the file limits by how much that file can be compressed. The calculations that tell you exactly how much information there is are based on probability. No compression algorithm works every time and all compression algorithms are capable of making your data larger rather than smaller; all you can say is that they will probably make the data smaller.

The way an algorithm can say it will 'probably' compress a file is by making some assumptions about the contents of that file; assumptions based on knowledge about what you might be compressing. A great deal is known about the way letters are put together to form words and this information can help the compression system. The algorithm will break down however if the assumptions are wrong. If the data is made up of random bytes then the compression system will almost always (i.e. very probably) make the data larger not smaller.

Another, more important case where the compression system will fail is when the data is already compressed. Since information theory tells us that there is a lower limit to the size of a file, if we have already squashed the file to that size then further efforts to squash it will be in vain. The vast majority of executable programs on the Acorn Archimedes are already compressed using a very efficient compression system called 'squeeze'. If a squeezed program is fed to any of the common compression systems it will end up bigger.

'Why does this matter to RemoteFS?' you may ask. Well, all users would like RemoteFS to go as fast as possible. Since RemoteFS passes data over connections with a limit to the rate at which data can be passed, it might seem a good idea to make sure you pass as little data as possible. Compressing the data is all well and good as long as the data compression works, but in the case where the data does not compress you still have to spend the time trying to compress the data even though it fails - you do not know until you have tried.

On a very slow transport it may still be possible to win since the time take to try the compression may be comparable to the time take to transmit the data. On the faster transports such as the parallel port and the ethernet, the fact that many files can not be compressed will slow the transport down so much that it is not worth trying.

So how can compression help the performance of RemoteFS? As was stated earlier, compression is all about knowing about your data in advance. The more you can make correct assumptions, the more you can compress the data. By using programs that explicitly allow you to compress their data you can make sure you move the minimum of data over the RemoteFS link -and thus get the best performance. Squeezed applications will load faster than ones that have not been squeezed. Bit-mapped pictures stored as GIF files will be moved faster than the same image stored as a sprite.

The bottom line is that you should know rather more about what is in a file than RemoteFS does, so you are better placed to suggest which sort of compression to try!