Disc Image Manager

User Guide

Written by Gerald J Holdsworth

Version 1.32

Licence

Disc Image Manager is distributed under the GNU GPL v3 licence:

This source is free software; you can redistribute it and/or modify it under the terms of the GNU General Public Licence as published by the Free Software Foundation; either version 3 of the Licence, or (at your option) any later version.

This code is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public Licence for more details.

A copy of the GNU General Public Licence is available on the World Wide Web at http://www.gnu.org/copyleft/gpl.html. You can also obtain it by writing to the Free Software Foundation, Inc., 51 Franklin Street - Fifth Floor, Boston, MA 02110-1335, USA.

Introduction

Disc Image Manager originally started out as a Delphi class, TDiscImage, written for inclusion into Repton Map Display so that it could read Repton data files direct from disc images. To test that the class worked OK, and as a demo application as I released the source code into the public domain, I wrote a GUI front end for it called Disc Image Reader.

This project was still in Delphi so, therefore, Windows only. It was not until November 2020 that someone on the Stardot forum asked about accessing Acorn DFS images from MacOS. Owning a Mac myself, and with it being my main computer (relegating the Windows laptop to a secondary PC), I also noticed this hole in the 'market'. So, I decided it was time to port this across.

However, to do so, I needed to port the project into Lazarus (basically the free, multi-platform, version of Delphi). Previous attempts at using this product had failed, but this time I was determined. So, with Lazarus newly installed on my Mac, I then imported Disc Image Reader and, to my surprise, it worked with only a few minor changes. Finally, we had Disc Image Reader running on macOS. That became version 1.05, and further development commenced.

The first thing to add in was make the application write back to the images. I had already written the code to write back to a DFS image, but never tested it. And thus, Disc Image Reader became Disc Image Manager. It also found a new home on the Internet, at GitHub, in addition to my own website.

Roll onto the latest version (1.32 at the time of writing) and it has come on beyond what I expected it to be at the start. With the support, bug reports, and suggestions from the Stardot community, development has come on in leaps and bounds. So, I thank you all who use this for, well, using it, and I hope you find it useful.

What Is It and What Can It Do?

Disc Image Manager is an application for managing retro-computer disc images. Basically, you can read and write to them using a modern-day operating system. So, why would you want to do that? If you are asking that question, then you probably should not be here!

The formats it can deal with, as of version 1.32 are:

- Acorn Disc Filing System (DFS), including the Watford DFS variant, both single and double sided. Disc Image Manager can read and write to a DFS image, and create new, blank, images.
- Acorn Advanced Disc Filing System (ADFS). Disc Image Manager can read and write to all known formats of Acorn ADFS floppy and hard drive images – old map (8-bit ADFS and the original Archimedes format); new map (RISC OS 2 onwards); old directory (8-bit ADFS); new directory (from Arthur OS onwards); and big directory (RISC OS 4.39 onwards). As with DFS, it can also create new, blank, images both floppy and hard drive (up to 1GB).
- Commodore 64/128. The formats for Commodore 1541, 1571 and 1581 are all catered for, and can be read and written to. Again, new, blank, images can be created.
- Commodore AmigaDOS. Currently, these can only be read from, but both floppy and hard drive images.
- MMFS. This part is still in beta and only a basic read from this format can be achieved.

- Acorn File Store (AFS). This is still in beta and can only read Level 2 and 3 File Server images. For Level 3, this is both as stand-alone images or as a separate partition in ADFS. As of version 1.32, Disc Image Manager can now create blank Level 2 and Level 3 images, although this is untested on real or emulated systems.
- !Spark and !PackDir. Currently only readable.

There are plans to add, in the future:

- Writing back to Commodore AmigaDOS, along with the creation of blank images, including hard drives.
- Sinclair Spectrum+3.

Downloads

Disc Image Manager can be downloaded from GitHub:

https://github.com/geraldholdsworth/DiscImageManager

Or from my own website:

http://www.geraldholdsworth.co.uk/index.php?link=DiscImageReader

You can also contact me via the Stardot forum, or through email: gerald@geraldholdsworth.co.uk.

Compatibility

Although not extensively tested on all systems, the provided binaries should work on the following systems:

Windows: XP, 7, 8, 10 (both 32- and 64-bit versions are provided). However, it has only been tested on Windows 10 64 bit.

Linux: This has been tested on Linux Mint Cinnamon 64 bit, and Raspbian OS 32 bit.

macOS: Leopard to Big Sur, but I have only tested on High Sierra and Catalina. Others have reported it working OK on Mojave.

It has also been lightly tested with ‘High DPI’ setups – i.e., under Windows or Linux you can magnify the screen beyond 100%. I have not found a way of doing this under macOS yet.

Installation and Basic Usage

All the pre-built binary downloads are provided in either a ZIP archive (Windows and Linux) or a DMG file (macOS). To install them, just open the archive/package and drag the application to wherever you wish it to be. It is just a single file (well, directory in the case of macOS) and can be run anywhere. There are no special installers, or anything special you need to do. It will just run. OK, you might get an annoying little message from the OS saying it is not from a registered developer, but once past that, it will just run.

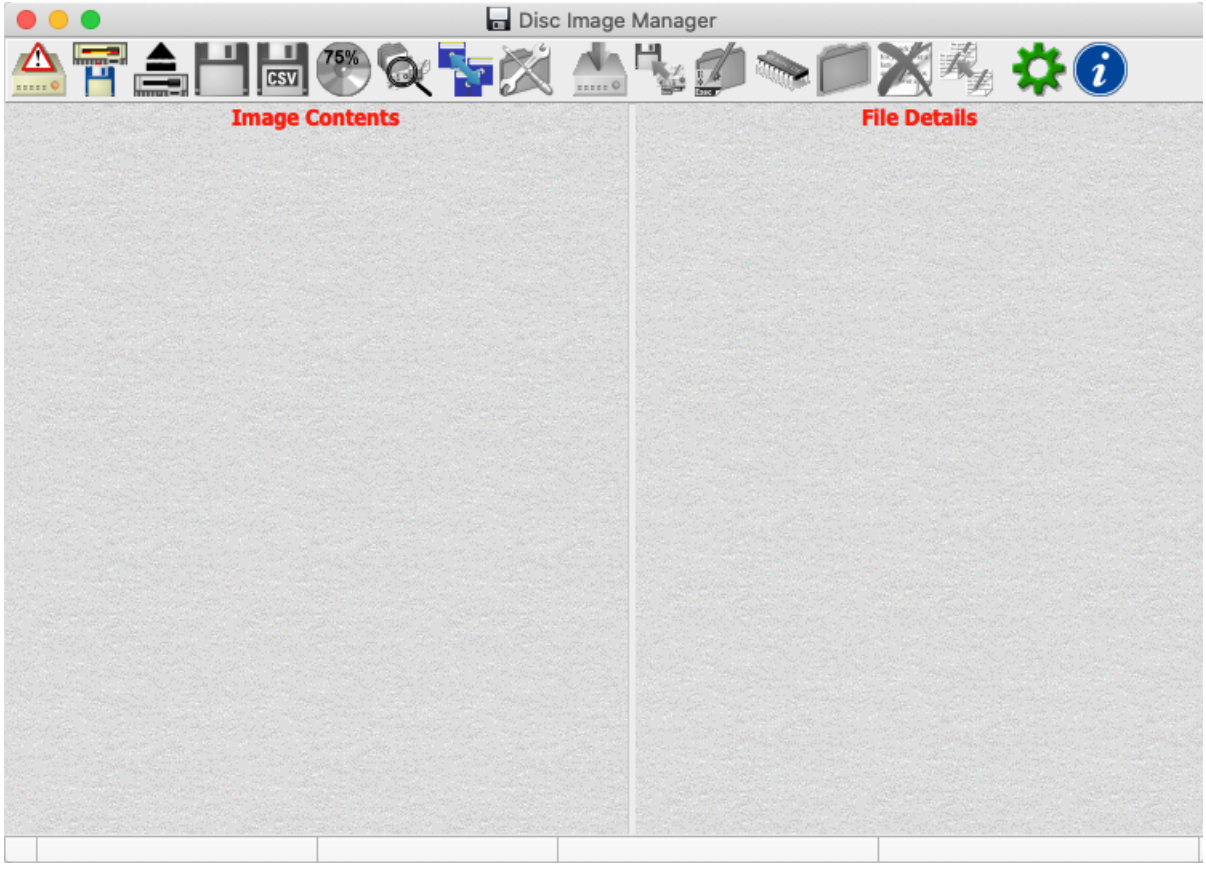
Using Disc Image Manager

Once you have it running you will be presented with the main window (note that all screen shots are taken from the macOS version). Along the top, in Windows and Linux, will be the main menu bar (this is located on the macOS menu bar at the top of the screen, in macOS). Below this will be the tool bar. All the buttons in the toolbar are replicated through the main menus and, for some, also in the context menu (more on this later).

Below this will be two panes – Image Contents and File Details. The former will display the contents of a disc image, while the latter will display the file (or directory) details on the selected file (or directory).

Finally, along the bottom is the status bar which gives various information about the opened image.

Layout



The toolbar buttons are, from left to right:

- New Image File: creates a new image.

- Open Image File: opens an image file.
- Close Image File: closes the currently open image.
- Save Image As: Saves the currently open image.
- Save File Details As CSV: Saves the file details, of all the files on the image, as a CSV file.
- Display the Free Space Map: Displays the free space map, along with some other details about the image. It will also allow you to change some of these details.
- Open File Search Window: Allows you to search the image for one or more files or directories by filename or filetype.
- Split/Combine DFS Images: This little tool will either split a double sided DFS image into two single sided DFS images, or combine two single sided DFS images into one double sided image.
- Repair Broken ADFS Directories: This tool will find and attempt to repair any broken directories on an ADFS disc.
- Download File(s): Allows you to extract (download) files from the image onto the host filing system.
- Add File(s): Does the opposite of the above.
- Rename File: You can rename a file or directory here.
- Show File Viewer: This will show you the contents of the selected file. This will include a hex dump but may also include other views.
- New Directory: Creates a new directory on the image, depending on the format.
- Delete File: Removes a file (or directory) from the image.
- Duplicate File: Clones a file within the same directory.
- Preferences: Customise various aspects of the application.
- About Disc Image Manager: What it is, who wrote it, what version and some other information.

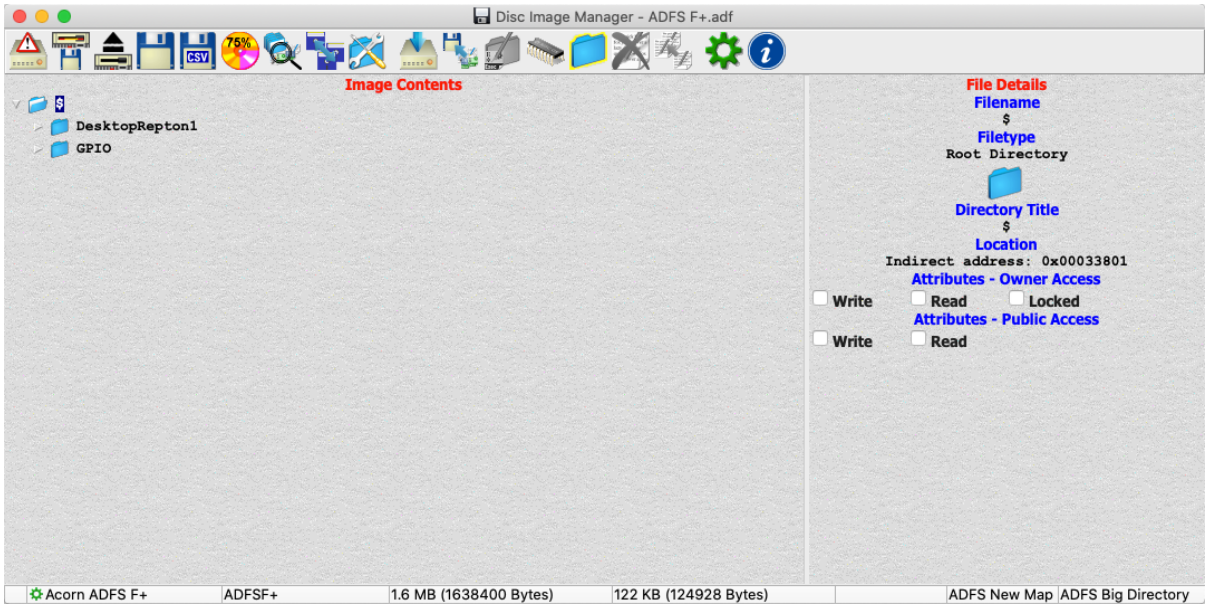
The status bar, along the bottom, will give this information (when an image is opened), from left to right:

- Modified: Displays a small icon  if the image has been modified since the last open/save.
- Image format: What system the image is for, along with what ‘shape’ it is.
- Disc title(s): If available, the title of the disc. For DFS double sided images, it gives the titles of both sides.
- Capacity: The maximum capacity of the image.
- Free Space: The amount of free space left on the image.
- Single/Double Sided: Indicates whether this is a single sided or double sided DFS image. Blank if not DFS.
- ADFS Map Type/Amiga FS type: Indicates whether the ADFS map is old or new, or the AmigaDOS is OFS or FFS.
- ADFS or Amiga Directory Type: Indicates whether this is an ADFS old, new or big directory, or if it is an Amiga directory or directory cache.

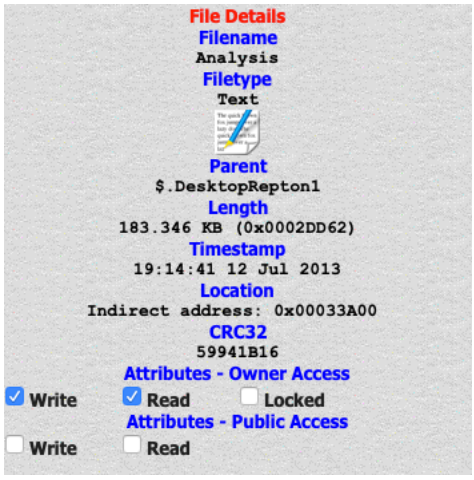
Working With an Image

OK, so let's open an image. There are three ways of doing this – either click on the toolbar button, through the main menu or, simply, just by dragging the file over the main window and dropping it. At this point, I will point out that you can do most things by dragging and dropping, or by not using the menus or toolbar.

Assuming it is a file that Disc Image Manager recognises and supports, you should get a populated display:



Now we have one open we can look at the File Details pane in greater detail. Incidentally, you can resize the main window by clicking and dragging one of the edges or corners, and the two panes using the vertical sizer between them.



Some of the fields displayed will be editable, while the others you can copy to the clipboard just by clicking on them.

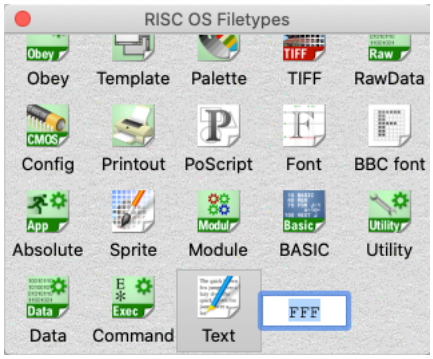
The Filename, Parent, Length, Location and CRC32 can all be copied to the clipboard (renaming a file or changing its parent will be covered later).

The Filetype, Timestamp, Directory Title, Load address and Execution address can be changed by clicking on them. The Attributes can be changed just by ticking, or unticking, the appropriate box.

Not all fields will be available for all files on all systems. As you can see here, the Load and Execution addresses are not available as they are encoded for the filetype and timestamp. And the Directory Title is not available because we are looking at a file. The format of the location will also change depending on the file system.

Changing The Details

So, let's start by changing the filetype. Click on the filetype text or icon to open a small menu window:



This will list all the filetypes known to the application (in fact, it lists all those that are known and there is a filetype icon available). Right at the end is an editable field for entering a filetype other than that listed. Just click on the desired filetype, or enter your own and press Enter, to change.

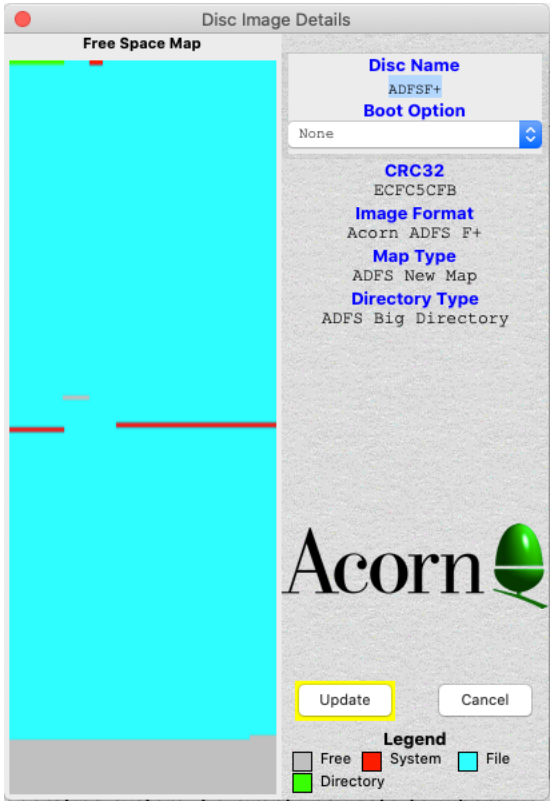
To change the Directory Title, again just click on it to make the field editable (unless, like here, it is an ADFS '+' format that does not have directory titles). Press Enter once you are happy with your change. Changing the Load & Execution addresses, and timestamp, are done in the same way.

Renaming a file is achieved by slow double clicking on the file – this means, click to select a file, then click again to turn the filename editable. If you are too fast, it will open the file viewer (more on this later). Press Enter to finalise the change.

If, with any of these changes, they are invalid to the file system, then they will revert. So be aware of the filing system restrictions (e.g., DFS filenames are 7 characters in length, maximum). You will also find that you will not be able to do any edits when you have multiple files/directories selected.

Free Space Map

There are some more changes you can make, but these are achieved through the free space map window:



On the left-hand side, you will find a nice graphical display representing the disc. It will show free space, space used by files, space used by directories and space used by the system.

On the right-hand side the disc name and boot option will be displayed, along with a repeat of some of the image's information.

The Disc Name and Boot Option are editable, and you can simply click on them to change. Once you are happy with your changes, click on Update to make the change. Clicking on Cancel will forget all changes in this window.

If you have a double sided DFS image open, you will get two free space maps, and two boxes for the Disc Name and Boot Option.

Not all formats will give this display and, therefore, are not editable.

Downloading (Extracting) Files

One thing that most people will want to do is to get files off the image and onto the host filing system. At the time of writing, dragging a file out of the application is not possible. So

therefore, we are left with the other option of using the toolbar button, or the main menu. You can also right click to open a context menu which will have this option listed (along with Rename).

Select the file, or files, you wish to download and use your preferred method (toolbar/main menu/context menu) to open the Save As dialogue box. This is an OS derived dialogue so will behave as others do on your operating system. So simply navigate to where you want to download these files to. When they are downloaded, they will have an 'inf' file alongside (unless you have selected not to – see Preferences). This gives other information about the file (such as load and execution addresses) which would otherwise be lost. Full details about this file can be found at the end of this guide.

Adding Files

Another operation that people will want to do is to add files, from the host operating system to an image. This can be done by dragging and dropping, just like opening an image. Disc Image Manager will also take account of any 'inf' files and skip over such files if they are part of the selection – I have seen similar utilities to Disc Image Manager that attempts to import the 'inf' files if they are selected.

You may find that the toolbar button, and menu item, for adding a file is greyed out. This is because a file can only be added if a directory is selected, where it will be added to that directory. Dragging and dropping a file will automatically select the root if none are selected.

Import of Another Image Contents

Something to watch out for – if you have an image open and you drag another image in, Disc Image Manager will recognise this and ask you if you wish to import the contents into this image. You might have wanted to open the new image or import it as a file. Well, you are given these options too. If you choose to import, then the following dialogue will open giving you the option of which files to import and which to leave:

Just simply untick those which you don't require and tick those that you do. You will find that as you tick, or untick, items the parent directory will change from ticked (☑), or unticked (☐), to an intermediate state (−).

Also, ticking, or unticking, a directory will cause all the contents, including any sub-directories and their contents, to be ticked or unticked. It can therefore be seen that if you tick the root ($) then the entire contents will be selected, and conversely, unticking the root will cause the entire contents to be deselected.

Once you are happy with your selection, just click on OK to commence the import, or Cancel to cancel the entire operation.

Adding the Contents of a !Spark or !PackDir archive

If Disc Image Manager recognises the file as a !Spark or !PackDir archive it will inflate and add the contents (if you wish). But if the uncompressed contents do not fit on the disc, you will not get the option and the file will be added as is instead.

In future these archives will be treated as a filing system, just like DFS or ADFS, so these can then be opened in the same way as image files and imported in the same way.

Currently, for archives, Disc Image Manager will check to make sure that the current open image is of a format suitable to receive the contents and a warning is issued if not. During the import any errors are ignored and skipped but logged and reported to the user once the operation has completed.

Deleting Files and Directories

To delete a file or directory, just select it and click on Delete File (either from the toolbar, main menu, or context menu). A word of warning – if you have multiple files or directories selected, then they will all be deleted, and if you have a directory selected then the entire contents, including sub-directories, will be deleted. But you will be presented with a confirmation box before this happens.

Creating a New Directory

This is very simple – just click on Create Directory toolbar button, or use the appropriate menu item, and a new directory is created (if the filing system supports directories). It will be given a default name which you can then change using the rename procedure above.

Copying and Moving Files

Files can be copied from one directory to another or, in the case of DFS, from one side to another. They can also be moved. This is done a similar fashion to the host operating system. However, at the time of writing, copying to the clipboard then pasting back is not available.

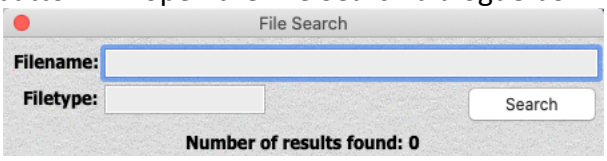
In the directory listing pane (Image Details), select a file, directory. Then you can just drag it to the desired directory (side) on the same image. While in the drag/drop operation, pressing CTRL (Windows and Linux) or ALT (macOS) will result in the file being copied (indicated by the plus icon ⊕), while pressing SHIFT will result in the file being moved. The destination directory is indicated by it being selected, and opened if not already done so, as you hover over it. Once happy with the destination, just drop the file.

Copying files into the same parent directory (you cannot move files into the same parent as this would be pointless) has the effect of cloning the file. Currently, this cannot be done with directories. This cloning operation can also be achieved by using the toolbar button, main menu item or context menu item.

At the time of writing, only single files, or directories, can be operated on in this fashion.

Searching for Files

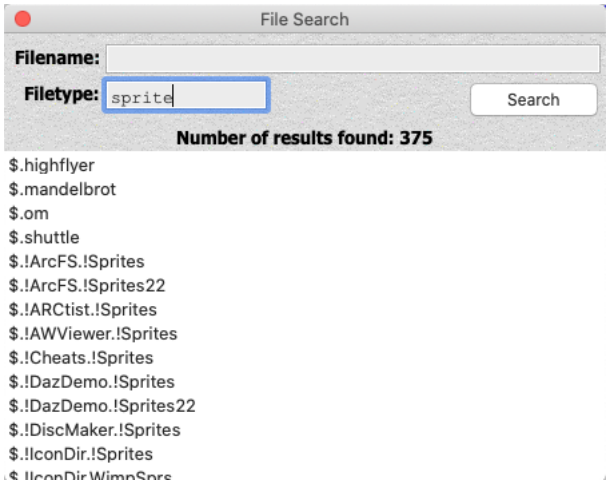
Clicking on the Search button will open the File Search dialogue box:



Just enter your search criteria and press Enter or click on Search. Both fields can be filled out if you so wish to search on both simultaneously. You can use wildcards:

- '#' will match any single character: 'a#c' will find 'abc', 'acc', 'adc', etc.
- '*' will match zero or more characters: 'ab*' will find 'ab', 'abc', 'abcd', etc.

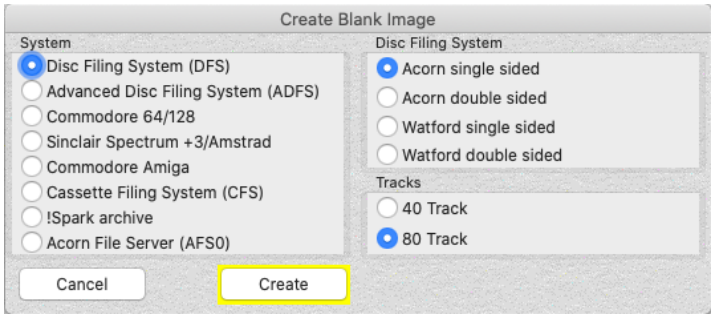
The search is case-insensitive, so 'abc' is the same as 'ABC' and 'Abc'.
The results will be listed below:



Double clicking on one of the results will select it in the main window.

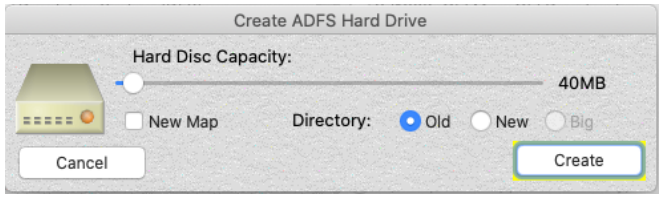
Creating a New Image

Disc Image Manager can create new, blank, images. These can then either be used to add files to, or for use in an emulator. To do this, click on the New Image button to open the New Image Dialogue box:



Simply just select the System on the left-hand side, then the format on the right-hand side. Some formats are unavailable and will cause the Create button to be greyed out.

Selecting to create an ADFS Hard Drive, and clicking on Create, will result in a further dialogue box:



Here you will need to specify the size (minimum of 20MB, maximum of 1GB), map type (New map or not) and Directory type (Old, New or Big). Some ADFS restrictions:

- You can only have an old or new directory with an old map.
- You can only have a new or big directory with a new map.
- Old map only supports up to, and including, 512MB. Above this it will be new map.

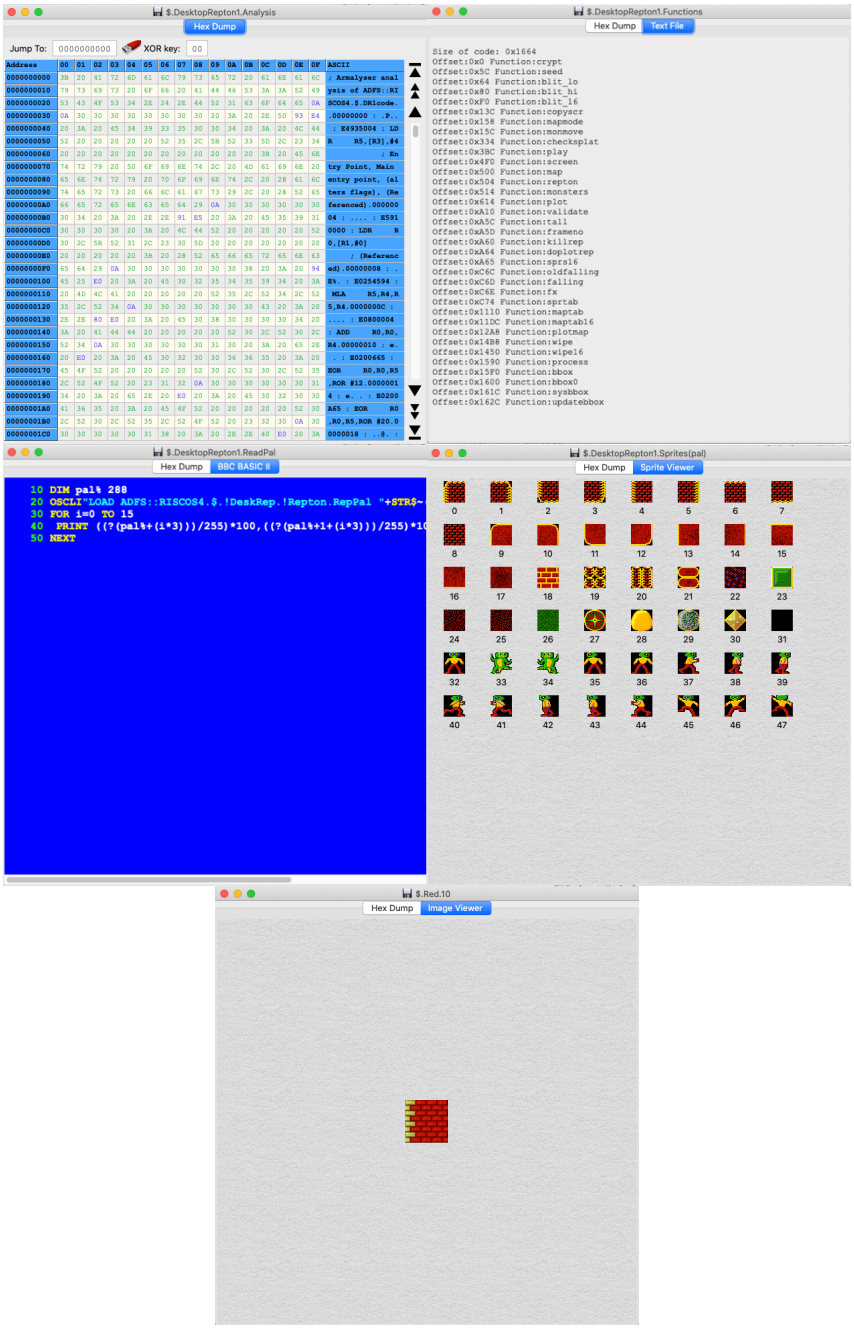
The maximum size of 1GB is purely down to memory restrictions. At the time of writing, Disc Image Manager loads the entire image into memory. Once the code is changed so that it opens and accesses the image directly from the disc, the 1GB upper limit can be removed.

Clicking on Create will then create the hard drive image.

Once the new image has been created, it will be displayed in the main display window and the root will automatically be selected.

The File Viewer

Double clicking on a file will open it in a separate window as a hex dump, allowing you to see the contents. But, if Disc Image Manager recognises this file as something it can display, it will:



Above we have, from left to right and top to bottom:

- Hex dump view: Every file will have this. Accessible via the ‘Hex Dump’ tab at the top.
- Text view: If Disc Image Manager determines that the file is pure text (only contains characters in the ASCII range 32-126 or 9, 10 or 13) then it will display it as text.
- BBC BASIC view: If Disc Image Manager determines that the file is a BBC BASIC file, it will detokenise and prettify the BASIC file.
- RISC OS Sprite view: If Disc Image Manager determines that the file is a RISC OS sprite file, the contents will be displayed as such.

- Image view: Other image formats (Windows Bitmap, PNG, or JPEG) will be displayed here.

Some Tools

There are a couple of tools supplied with, and part of, Disc Image Manager. The first is the ability to repair ADFS broken directories.

Repair ADFS Broken Directories

Disc Image Manager will attempt to open and display ADFS directories if they are broken or not. Directories will be displayed, on the right-hand pane, as broken along with an error number. At the time of writing, the following will attempt to be repaired:

- Broken directory due to the start and end sequence number not matching: When ADFS writes to a directory, it updates the sequence number (the number in brackets at the top of a directory display in 8-bit ADFS) at the head of the directory, makes the write, then updates the sequence number in the tail. This means that if anything happens during the write, it will result in a broken directory. Bit 0 of the error code is set in Disc Image Manager.
- Broken directory due to the start and end identity name not matching or incorrect (old and new directory): In ADFS, directories are identified by the string 'Hugo' (after Hugo Tyson) or 'Nick' (after Nick Reeves). There is an id in the header and the tail. Bit 1 of the error code is set in Disc Image Manager.
- Broken directory due to the start and end identity being incorrect (big directory): As above, but with big directories the id is 'SBPr' in the header and 'oven' in the tail (after Simon Proven). Bit 2 of the error code is set in Disc Image Manager.
- Broken directory due to incorrect cyclic redundancy check: To ensure the integrity of the directory a CRC is calculated based on the contents and stored in the tail. Old directories this can be zero. Bit 3 of the error code is set in Disc Image Manager.

There are some other cases that will cause a broken directory. The main one is the use of interleaved images (mainly with ADFS 'L' shape discs). If a disc is read in and assumed interleaved where it is not, then the result will be broken directories. Disc Image Manager will detect these, as the directory will not be where it should and therefore is unlikely to be able to be repaired. We will cover interleaved images later.

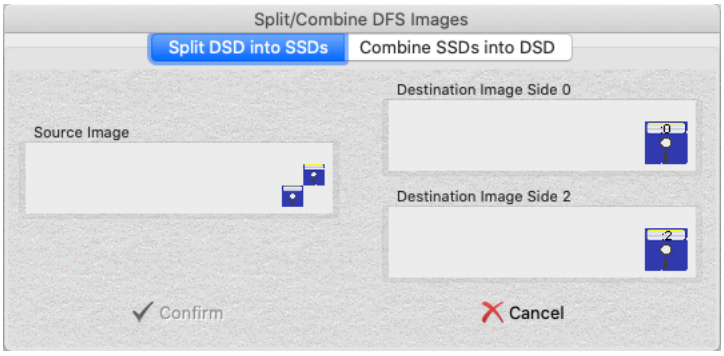
Another more obscure cause is where a directory is not sector aligned, either the start or the length (in the case of big directories, whose length can change). Disc Image Manager does not (at the time of writing) detect or report these.

To repair an image, it must be loaded into Disc Image Manager.

Split or Combine DFS Images

There are cases where you may have a double sided DFS image and need it as two separate single sided images; or you have two single sided images and need them combined as a double-sided image. This may sound like a simple operation, but as DFS double sided images are interleaved (the tracks are stored as Side 0, Track 0 followed by Side 1, Track 0, etc.), this becomes more complicated.

For this, nothing needs to be loaded into Disc Image Manager. Just click on the toolbar button or the appropriate menu item to open the tool:

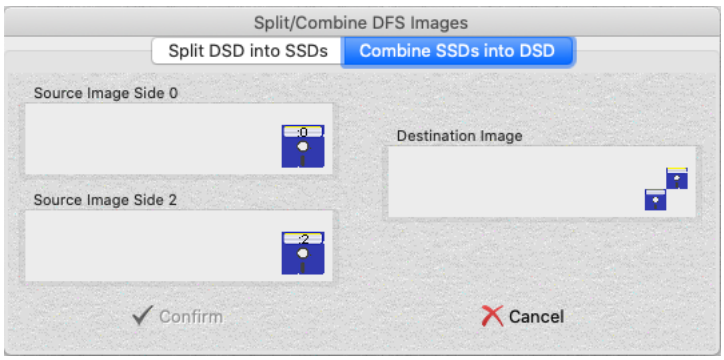


The window is split into two pages, selectable using the tabs at the top.

Splitting

For this, click on the double disc image under Source Image to show an open file dialogue box, navigate to your file and open it. The Destination Image boxes will then automatically be populated with pre-prepared filenames based on the source image filename. These can be changed if you wish by clicking on the single disc image for each side. Once you are happy, click on Confirm.

Combining

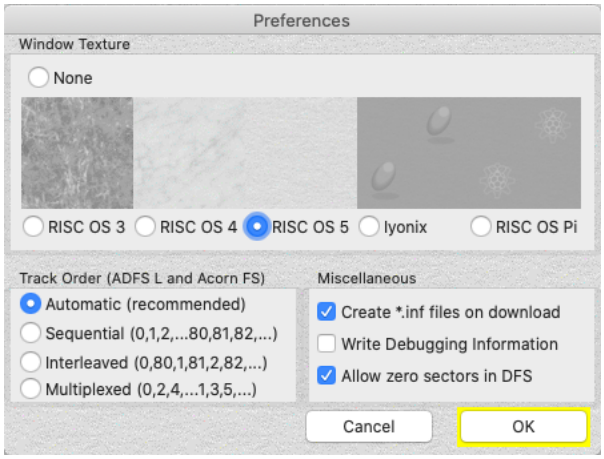


Like above, you will need to select the two single sided images, and the destination image filename will be automatically generated. Again, this can be changed.

In both cases the resultant image(s) is(are) not opened in Disc Image Manager.

Preferences

Clicking on the Preferences button (or using the main menu) will open the dialogue box so that Disc Image Manager can be customised.



The options available here (as of version 1.32) are:

- **Window Texture**: There are six options for the background texture of each window within Disc Image Manager:
  - None: all windows will have a plan background;
  - RISC OS 3 style;
  - RISC OS 4 style;
  - RISC OS 5 style;
  - RISC OS on Iyonix style; and
  - RISC OS on Raspberry Pi style.
- **Track Order**: We touched briefly on interleaving earlier on. This is done on all DFS Double Sided images, most ADFS 'L' images, and some Acorn FS images. As pointed out earlier on, if an ADFS L image is loaded as interleaved but is not interleaved, then it may well result in some broken directories. This is how the Automatic option works – it tries loading the image as interleaved and counts the number of broken directories. If there are any, it reloads it as multiplexed. Then, if there are still some broken directories, it tries sequential, before going back to interleaved. The other three options force the loading using a specific method.
  - **Sequential**: This is where the track order is one after the other. E.g., 0,1,2,etc all the way up to track 159 (for a double sided 80 track image).
  - **Interleaved**: This is where the track order swaps between sides. E.g. track 0, side 0; track 0, side 1; etc.
  - **Multiplexed**: This is where the drive double steps to the next track. So, you get the even tracks first, then the odd tracks.
- **Miscellaneous**:
  - **Create *.inf file on download**: Specify whether to produce an inf file or not when downloading (extracting) files from images to the host filing system.
  - **Write Debugging Information**: Create and update a file in the temporary area of the operating system (called 'DIM_LogFile.txt') to write any debugging information from the application.
  - **Allow zero sectors on DFS**: This will allow DFS images that specify a zero number of sectors (and, hence, an image length of zero). Because this means

that Disc Image Manager will accept a 512-byte file full of zeros as a legitimate DFS image. This also may cause issues reading in other formats, which is why this option has been provided.

Clicking on OK will save these settings to the registry and apply them instantly or, in the case of the Track Order option, the next time one is loaded.

Command Line Options

Disc Image Manager can be used by passing commands on the command line, and so therefore not utilising the GUI.
The command line options are:

Commands take the form <command>:<parameter1>|<parameter2>|...
Parameters in square brackets [] indicate optional parameters.
Any parameters containing spaces should be contained within double quotes (").
Any unrecognised commands, or insufficient number of parameters, will be ignored.

Add files to the image

`--add -a`

Parameters: <OS filename>[|<destination>][|<FS filename>][|<load address>][|<execution address>][|<attributes>]

Adds files specified in <OS filename>, which can be wildcarded, to the open image in <destination> directory. Assumes drive 0 and root if nothing selected or not found. The filename to be used on the image can also be specified, along with the load address, execution address and attributes. Addresses should be in hex, and the attributes can either be in hex or as text as per access command below. If the optional parameters are specified, then this overrides anything contained within the related 'inf' file, if this exists.

Set attributes for specified file

`--access -ac`

Parameters: <FS filename>|<attributes>

Sets the attributes (access rights) to the file or directory specified. <attributes> can be a string containing any, or all, of the following (in any order):

Bit	Attribute	Meaning
0	R	Owner Read
1	W	Owner Write
2	L	Owner Locked
3	E	Owner Execute only
4	r	Public Read
5	w	Public Write
6	e	Public Execute only
7	P	Private

Not all attributes are available on all file systems, so only those relevant to the system will be used. Instead of a string, a hex value can also be passed (using the bit value in the first column).
This will be expanded to take account of Commodore 64 and Amiga access rights.

Read an execute commands from a text file

`--cmdfile -f`

Parameters:<filename>

Reads and executes commands held in a text file <filename>. Commands will be those listed here and can include this command.

Create new directory

`--create -c`

Parameters: [<newdir>]|[<parentdir>]

Creates a new directory, <newdir>, in the currently open image as a child of <parentdir>. If <parentdir> is not specified, then the root will be used. If <newdir> is not specified, then a default name will be given (usually 'NewDir' plus a number).

Delete file or directory

`--delete -d`

Parameters:<filename>

Deletes a file/directory from inside the currently open image.

Write Debug Information

`--debug -db`

Parameters:ON or OFF

Turns the logging of debugging information on or off. This does not update the registry setting and is only for this session.

Set directory title

`--dirtitle -dt`

Parameters:<dirname>|<title>

Sets the directory title for the specified directory.

Extract files or directories

`--extract -e`

Parameters:<filename>[|<destination>]

Extracts file(s) specified in <filename> to the local OS path, or to the destination folder if specified. <filename> can contain wildcards:
* : one or many characters
# : any character
e.g. to extract everything from an ADFS image, both `--extract:$` and `--extract:$.*` will work. DFS will require the drive specifier, i.e. `--extract":0.$"` or `--extract":0.$.*"`.

Open image file

`--insert -i`

Parameters:<filename>

Opens image file <filename> from the host operating system. If it is not a valid image, nothing will be opened. <filename> should contain a full or relative path to the file.

Keep application open

`--keepopen -k`

Parameters:none

Keeps the application open after parsing the command line options.

Create a new image

`--new -n`

Parameters:<format>

Creates a blank image file for format <format>:
<format> can be:

DFSS	: Acorn DFS single sided 80 track
DFSS40	: Acorn DFS single sided 40 track
DFSD	: Acorn DFS double sided 80 track
DFSD40	: Acorn DFS double sided 40 track
WDFSS	: Watford DFS single sided 80 track
WDFSS40	: Watford DFS single sided 40 track
WDFSD	: Watford DFS double sided 80 track
WDFSD40	: Watford DFS double sided 40 track
ADFSS	: Acorn ADFS S
ADFSM	: Acorn ADFS M
ADFSL	: Acorn ADFS L
ADFSD	: Acorn ADFS D
ADFSE	: Acorn ADFS E
ADFSE+	: Acorn ADFS E+
ADFSF	: Acorn ADFS F
ADFSF+	: Acorn ADFS F+
ADFSHDD	: Acorn ADFS hard drive image
CFS	: Acorn Cassette Filing System
C1541	: Commodore 1541
C1571	: Commodore 1571
C1581	: Commodore 1581
AMIGADD	: Commodore Amiga DD (not currently supported)
AMIGAHD	: Commodore Amiga HD (not currently supported)

When creating a hard drive image, an additional parameter is required. However, this is optional and leaving it off will result in an old map, old directory 20MB image being created.

The second parameter takes the form:
<O|N><O|N|B><capacity>[M] where
<O|N> is the map
O: Old or N: New;
<O|N|B> is the directory type
O: Old, N: New, B: Big; and
<capacity>[M] is the capacity of the image in bytes (or MB if the M is included).

Notes: The minimum capacity is 20MB, while the largest is 1000MB (512MB for old map). The combinations of map and directory are:
OO: Old map old directory (similar to S, M and L shape floppy);
ON: Old map new directory (similar to D shape floppy);
NN: New map new directory (similar to E and F shape floppy); and
NB: New map big directory (similar to E+ and F+ shape floppy).

Set boot option

`--opt -o`
Parameters:<opt>

Sets the boot option for the disc. <opt> can be:
0 or none: *OPT4,0
1 or load: *OPT4,1
2 or run: *OPT4,2
3 or exec: *OPT4,3

`--opt1 -o1`
Parameters:<opt>

Sets the boot option for the disc, side 1 (DFS double sided only). See above for options.

Rename file or directory

`--rename -r`
Parameters:<oldfilename>|<newfilename>

Renames <oldfilename> to <newfilename> within the currently open image. <oldfilename> should be the complete path, while <newfilename> should just be the file's new name.

Save image file

`--save -s`
Parameters:[<filename>][|<uncompress>]

Saves the currently open image file as <filename>. If <uncompress> is set to TRUE (for UEF) then the file will be saved uncompressed. If <uncompress> is specified for any other image type, then this is ignored. If <filename> is omitted, then the current file and path will be used. If this is a newly created image then the filename used will be Untitled, with the appropriate extension, and will be saved to the local folder.
This command is not entirely necessary as the image will be saved after all other commands have been run with the defaults for <filename> and <uncompress>.

Set disc title

`--title -t`
Parameters:<disc title>

Sets the title for the disc

`--title1 -t1`

Parameters:<disc title>

Sets the title for the disc, side 1 (DFS double sided only).

Inf Files

Most applications dealing with disc images, including emulators, will utilise a file known as an 'inf' file. These, unsurprisingly, have an extension of '.inf'. They are named the same as the file to which they belong, and which is to be imported into (or has been exported from) one of these applications. The idea is that files residing on a FAT32/NTFS/etc. file system will lose the information that the BBC MOS, and RISC OS, requires.

There has not been any hard and fast format regarding these, so a discussion was held on the Stardot forums to bash out an agreed format, which now follows. Please also note that this also applies to directories, as well as files.

This currently does not extend to non-Acorn formats, yet.

*.inf format

So, the format agreed is this - a single line, in a text file, containing:

`<filename> <load> <exec> <length> <access> <extra info>`

Where:

Each field separated by at least one space, but could be more.

`<filename>` is the original BBC filename. Quotes are optional, but mandatory if the filename contains spaces. This could be different to the way the file is named on the host system (and hence the inf file).

`<load>` is the file's load address in hex.

`<exec>` is the file's execution address in hex.

`<length>` is the file's length in hex.

`<access>` can be either the access letters (L for DFS, LWRElwre for ADFS), or hex number according to the OSFILE API:

Bit	Meaning
0	'R': Readable by you
1	'W': Writable by you
2	'E': Executable by you
3	'L': Not deletable by you (locked on DFS)
4	'r': Readable by others (NFS, not 8-bit ADFS)
5	'w': Writable by others (NFS, not 8-bit ADFS)
6	'e': Executable by others (NFS, not 8-bit ADFS)
7	'l': Not deletable by others (NFS, not 8-bit ADFS)

For DFS, this will be 0x08 for locked, or 0x00 for not locked.

`<extra info>` is tag value pairs, using quotes where applicable (i.e., contains spaces) for any extra information.

Filename Translation

The filename for the host filing system (e.g., Windows) should be valid for that system, with the .inf file matching.

BBC	<->	DOS/Windows/macOS
#	<->	?
.	<->	/
$	<->	<
^	<->	>
&	<->	+
@	<->	=
%	<->	;

Applicable to both files and directories.

Credits

Disc Image Manager designed and written by

Gerald J Holdsworth

RISC OS graphics designed by

Richard Hallas

Help and advice from

Jasper Renow-Clarke (ADFS new map)

Robert Sprowson (ADFS new map hard drive parameters)

David Pilling (!SparkFS format)

Jonathan Harston (ADFS and Acorn FS interleaving)

Format sources

BBC Micro Advanced User Guide

BBC Master Reference Manual Part 1

RISC OS 2 Programmer's Reference Manual

RISC OS 3 Programmer's Reference Manual

MDFS.net

RISC OS Open Website

This guide written by

Gerald J Holdsworth