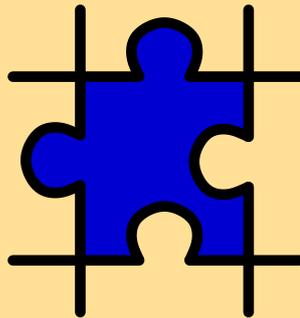# The dvdisaster User's Manual

Written by the dvdisaster development team
support@dvdisaster.org

for dvdisaster version 0.79.10

Abstract

This is the *orange manual*, describing the usage of *dvdisaster*, a tool for creating error correction data ("ecc data") for optical media such as CD, DVD and BD discs. Use cases for creating ecc data, recovering defective media using ecc data and for general maintenanance of optical media are given. See `https://dvdisaster.jcea.es/` for additional resources on the dvdisaster project, e.g. for the *blue* manual (codecs.pdf) containing a formal specification of the error correction data format.

Contents

Preface

After the publishing of version 0.79.5, the project has been dormant for another half decade. As time has passed by, it is foreseeable that optical media will become extinct just like floppy discs did long ago. Still, it is important to preserve the contents of already existing optical media. Towards this end, we will maintain dvdisaster to keep it usable on current hardware and OS versions.

We do not plan for a rapid release cycle. Neither will many new features be introduduced like it was at the beginning of the project. Instead, we will strive to fill in the missing gaps left over from the still open RS03 release cycle. Afterwards, dvdisaster shall be kept sound and complete as long as optical media are still in use.

– – The dvdisaster development team, Spring 2021

Preface for version 0.79.5

Since the release of dvdisaster 0.79.3[1], nearly five years have passed. This was partly due to changed circumstances in its primary developer's life, but there was also a lot of coding going on behind the scenes. In comparison with its predecessor, dvdisaster 0.79.5 comes with lots of its internals being significantly reworked.

The most visible improvement of dvdisaster 0.79.5 is, of course, its multithreaded RS03 codec. While it takes about 62 minutes for protecting a 36 GiB image with RS02 on a mid range PC, the same task is done with RS03 in less than 7 minutes using 6 processor cores on the same machine. On a high end server with at least 16 cores and very good I/O, this can be done in under a minute. That's quite an improvement.

RS03 is ready for production use in the current release. Some non-essential features, especially reworking the adaptive reading for use with RS03 and multi-threaded RS03 decoding (media fixing) will be delivered with the following dvdisaster releases.

Other parts of the project had to be changed or even discontinued. A software project lives on development and continuous releases; else the project will eventually die. In this respect, dvdisaster was very endangered in the last few years. To prevent this from happening again, most effort is now directed into source code development; everything else is delegated or discontinued. Source code development basically means making the GNU/Linux version, which provides the code base for all other versions, and the FreeBSD and NetBSD ports, which are very easily derived from the GNU/Linux code. This is not the case for the Mac OS and Windows ports, which are, unfortunately, discontinued as of now.

Another feature which has to go are the separate stable and development releases. Starting with this version, all dvdisaster releases are considered production quality, so there is no need for different branches anymore.

Maintaining the multi-lingual online documentation, which also served as the project home page, did also prove to be too time consuming. The project home page has been changed into a simple download platform for the project sources. It is now directed at package maintainers who will create and pass on binaries for the GNU/Linux, FreeBSD and NetBSD distributions.

The program documentation, which you are reading right now, is provided in PDF format which is much easier to author than the HTML version. The only language available is English. Most parts of this manual have been adapted from the old online documentation, so it still feels more like a website than a book. While hyperlinks are not as usable in PDF as in HTML, they have been kept in this document to stress that it is intended to be used as an online reference. So please do our environment a favour and do not print this manual. It is not meant to be read from front cover to back cover, anyways.

Okay, enough ranting already. May dvdisaster be helpful in protecting and recovering your valuable data, and thanks for using it!

– – cg, August 2015

---

[1]Version 0.79.4 was never finished and released.

1 Overview

The challenge: Optical media eventually fail.   Optical media (CD,DVD,BD) keep their data only for a finite time (typically for many years). After that time, data loss develops slowly with read errors growing from the outer media region towards the inside.

The dvdisaster solution: Archival with data loss protection.   dvdisaster complements optical media (↪ supported media) with error correction data in a way that they are fully recoverable even after some read errors have developed. This enables you to rescue the complete data to a new medium.

Error correction data, in short "ecc data", is either added to the medium or kept in separate error correction files. dvdisaster works at the image level so that the recovery does not depend on the file system of the medium. The maximum error correction capacity is user-selectable.

1.1 Common misunderstandings about dvdisaster

Before we describe in detail what dvdisaster can do, let's first clarify what it can't:

dvdisaster can not make defective media readable again.   Like a conventional backup, error correction data must be created from a fully functional optical medium - *you can not backup data which has already been lost*. When the optical medium develops defective sectors at a later time, those defective sectors are restored by re-calculating them from the ecc data. This won't make the defective medium working again, but will produce a new iso image which can be written to a new medium.

As said before, ecc data can not be created from already defective media. Although unreadable sectors can not be recovered in that case, dvdisaster might still be helpful in extracting the remaining readable portions of the medium.

It's not a ripping tool.   If you want a tool for copying protected media, you're looking at the wrong place. Such functions are outside the scope of dvdisaster's internal design and goals. Contrary to some myths saying otherwise: dvdisaster contains no hidden program fragments or switches for reading protected discs. Check the source code for yourself if you don't trust us.

1.2 How to use this manual

Many users just want to see some examples of solving typical tasks. Flip over to the Typical applications section in that case.

The remainder of this section gives an example of recovering a defective medium including screen shots, relates using ecc data to performing quality scans and full backups, and summarizes the pro and con of dvdisaster.

The downloads section provides a link to the download site, summarizes the system requirements, and clarifies that you can get and use dvdisaster as free software, at no cost and while keeping your full privacy.

There is also a chapter containing general questions and answers, technical questions and answers, and explanations of error messages.

The background information section provides details on the properties of the error correction, the difference between image level and file level data recovery, the RS01, RS02 and RS03 error correction methods, the linear and adaptive reading strategies, some remarks on how media read errors come into existance, and finally a few hints for storing error correction files.

As not all optical disc burning software may be compatible with dvdisaster, you might want to the perform compatibility tests before using it .

If you encounter a defect (programming error) or incompatibility with a certain (drive) hardware and software setup, please see the reporting defects section.

1.3   Example of the error correction
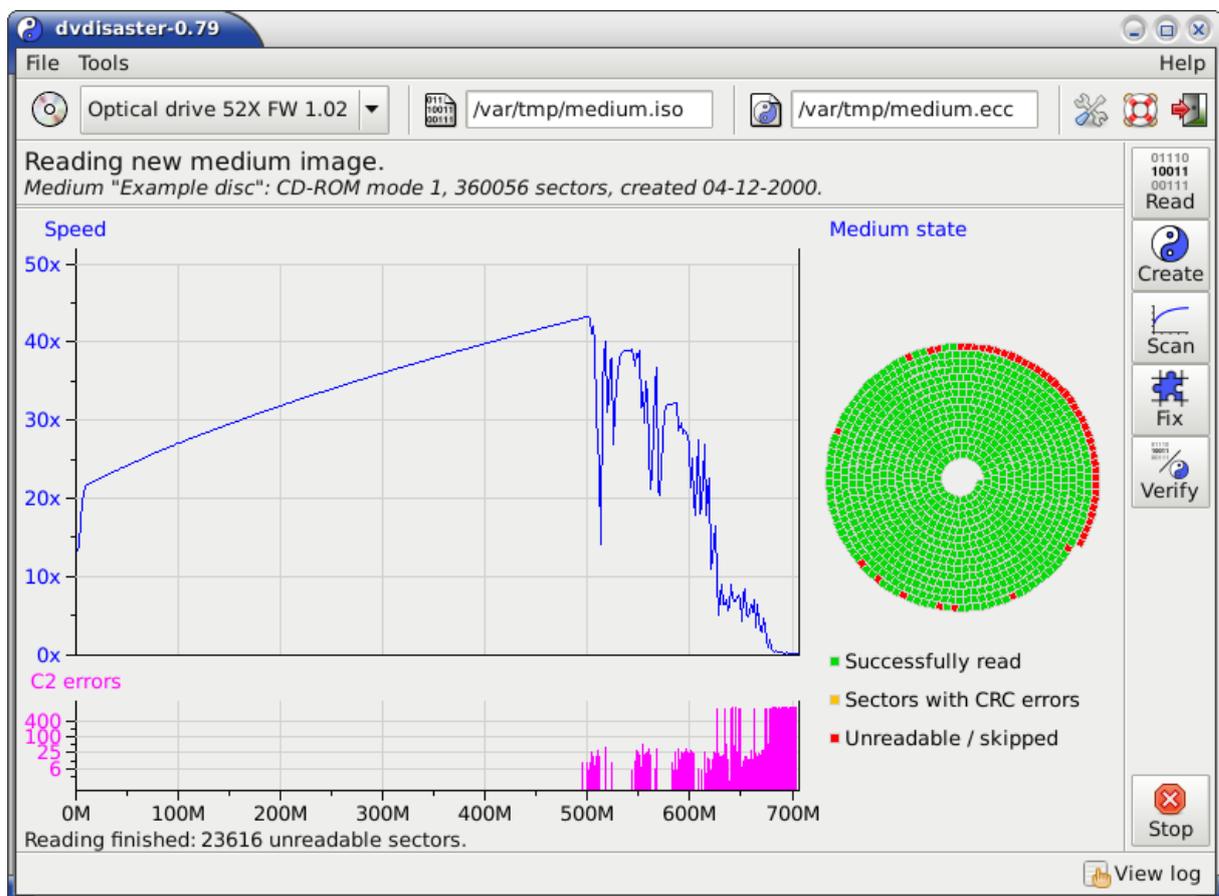
Figure 1: Reading a defective medium.

Recovery of aged media.   The medium processed here has become discolored and partly unreadable in its outer region. A reading attempt yields about 23.600 unreadable sectors of 360056 sectors total; resulting in about 6,6% defective sectors. Figure 1 shows the dvdisaster window after the reading attempt. The distribution of reading speed and read errors over the medium is graphically shown. The still readable sectors are stored in an ISO image called *medium.iso*.
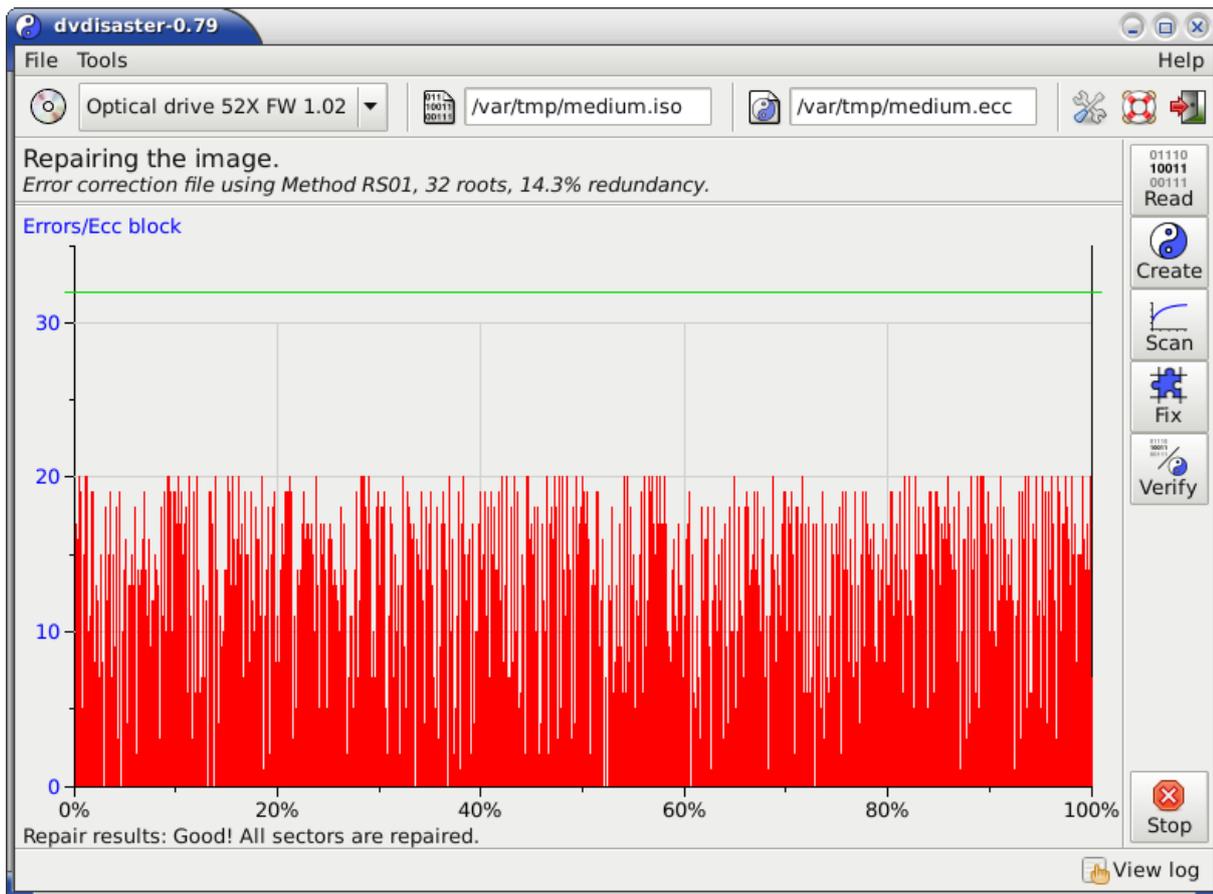
Figure 2: Repairing the defective image.

Repairing the defective image. The image which has been just read is incomplete since about 23.600 sectors could not be read. These sectors are now reconstructed using the error correction data created with dvdisaster. During the recovery a maximum of 20 errors per error correction block is encountered (see figure 2). This results in a peak error correction load of 63%, meaning that this degree of damage is handled well by error correction data created with default settings. The recovered image can now be written to a new medium.

Recovery needs error correction data: The recovery process described above uses error correction ("ecc") data. Think of this data as a special form of backup data (it needs less space than a normal backup, though). Like an ordinary backup, the ecc data needs to be created before the medium goes defective.

So if you have a defective medium but never created ecc data for it, you will not be able to recover the defective sectors (23.600 in the above example). The data located at the end of the medium will be lost, while you will probably be able to extract some files which are located at the beginning of the medium.

1.4   dvdisaster as a complement to quality scans

Quality scans, e.g. C2 error or PI/PO scans are a valuable tool for testing the results of the media writing process.

But quality scans are **not** a reliable means of **predicting the lifetime** of optical media. Consider we are looking for the right time to copy a worn-out medium onto a new one:

- Too early: Copying media because of a bad quality scan is cost-ineffective. Sometimes such media remain readable much longer than expected.

- Too late: When the quality scan reveals unreadable sectors some data has already been lost.

- Right before the medium fails: The ideal case, but how to tell?

However, we could do it the dvdisaster way:

- Create error correction data for the medium.

- Scan the medium regularly. Use it until the first read errors occur.

- Recover the read errors using the error correction data. Write the recovered image to a new medium.

1.5   Error correction data vs. full backup

A conventional backup strategy would be making one or more copies of the optical medium. This has a few advantages: Copying a medium is fast, and having two (or more) working copies available can be convenient, especially when working at different locations.

The disadvantage of this approach is that it guards only against incidental damage, but not against general aging. It is not helpful to have ten copies which all decay in a similar manner. If all ten copies are unreadable in the outermost region after a few years, data loss has occurred even though we were spending 900% of the original storage capacity for the backup.

Ecc data behaves differently since it is not a verbatim copy of the original data. It is a mathematical scheme working like this: Give me any 80% of the original data and I will be able to reconstruct the missing 20%, regardless of *where* the 20% are missing (whether at the beginning or at the end, maybe in between - doesn't matter). Incidentally, there is a strong relationship between being able to reconstruct a missing percentage of the original data and the size of the ecc data: If the ecc data is 20% of the size of the original data, it can roughly recover up to 20% of missing data; with ecc data being 30% of the original size up to 30% can be recovered and so on. But this relationship isn't even the greatest advantage of the ecc data; the "regardless of where the defects are" is the big deal.

Let's assume we want to have a 100% protection of a specific 4 GiByte DVD. Then we create another DVD containing 4 GiBytes of ecc data. At a later date, both DVDs decay and the last 30% of both become unreadable. Since we have still 70% of the original data and of the ecc data, everything is fine! We can still reconstruct the original data from them; using the second

DVD for ecc data is much more efficient than creating a second copy on it. In fact putting another copy on the second DVD would not have saved us from a 30% data loss.

We can even make some assumptions about our media. Maybe we expect that even a defective medium will not lose more than 15% of its data (don't take my word on it). And we make sure that ecc data will be saved on a different type of medium which is considered to have a longer life than optical media. Then creating ecc data with a recovery rate of 20% (always leave a safety margin) should suffice our needs. This would yield a reasonable data protection while spending only an additional 20% of storage for it.

This is not to say that ecc data is the final answer to all archiving means, but when used well, it can be much more efficient and secure than a simple backup strategy. See also the "Big Picture" section for a continued ecc data vs. full backup discussion.

1.6 Pro and con of dvdisaster

To summarize from the previous sub sections:

Advantages of using dvdisaster:

- **Protects** against aging and accidental medium damage (within certain limits).

- Read error tests run **faster** than quality scans; up to full reading speed depending on the drive.

- **Cost-effective:** Media must be replaced with a new copy only when they are really defective.

- **Space-efficient:** Ecc data requires less space than a full backup under most scenarios.

Limitations of using dvdisaster:

You need a backup and testing strategy and at least 15% of additional storage.

- Error correction data **must be created before the medium fails**, preferably at the same time the medium is written.

- Error correction data requires **additional storage space** either on the protected medium or by using additional media. Using the standard settings the additional storage space amounts to 15% of the original data size (approx. 700MiB for a full 4.7GiB DVD).

- No guaranteed protection against data loss as limits and statistical properties of the error correction may be exceeded with extremely bad luck.

See also the collection of background information to learn more about the functioning of dvdisaster.

2   Typical applications - HowTos

dvdisaster is a complex tool which would require a book of a few hundred pages to cover all of its features. Since we are currently lacking the resources for doing such a book (and you might be short on reading time also) we will take a different approach in this section. First we demonstrate how the different functions of dvdisaster work together. Then we describe common tasks and provide step by step instructions for solving them. In most cases following these steps will be all you need to do. At the end of each instruction set a discussion of further configuration options is included for advanced users.

2.1   Symbols used in this document

Working with dvdisaster requires certain combinations of optical media, media images and error correction data. Check out the following symbols to find out what you will need for the respective tasks:

Medium   (a CD for example):

These symbols indicate whether processing a medium is part of the respective task, and if the medium needs to be completely error free or may already be damaged.

good medium   bad medium
(**no** read errors)   (**with** read errors)

Medium image   (ISO image of a medium stored on the hard disk):

Some functions do not work directly with the medium, but with an ISO image on hard disk instead. Depending on the condition of the respective medium the image may be complete or incomplete.

complete image   incomplete image
(made from   (made from
good medium)   bad medium)

Error correction data

Recovering media images by using error correction data is the key feature of dvdisaster. These symbols show whether error correction data is required.

Medium containing   Separate error
error correction data   correction file

2.2 The big picture - understanding dvdisaster

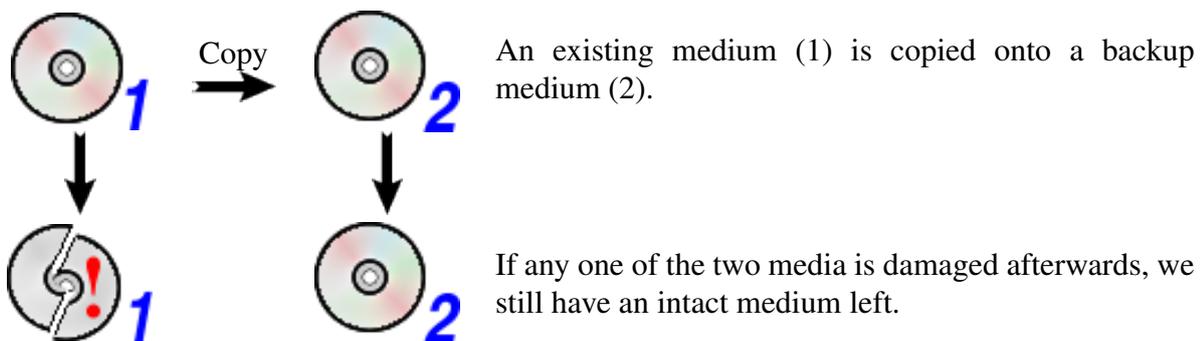In this sub section we are getting a basic understanding of dvdisaster:

- It is important to understand that dvdisaster works similar to a conventional backup in some regards, but that there are also important differences.

- The general idea of the error correction is explained.

- Jane demonstrates the proper usage of dvdisaster. She will create error correction data in advance and is therefore able to recover all data when her media become defective.

- However you should not follow the way of Joe. He does not use error correction data and finds out that his defective media are not recoverable even after multiple reading passes. As a consequence he loses data from a defective medium.

Of course these stories are purely fictional and any similarities with existing persons or situations are purely conincidental.

2.2.1 A comparison of dvdisaster with conventional backup

dvdisaster stores data on optical discs in a way that the data is fully recoverable even after the medium has developed some read errors. The method employed in dvdisaster uses less storage space (or additional media) than a full backup would do. Before using dvdisaster it is important to understand the similarities and differences between dvdisaster and a conventional (full) backup.

Let's first consider how a conventional backup scheme works:

An existing medium (1) is copied onto a backup medium (2).

If any one of the two media is damaged afterwards, we still have an intact medium left.

There are actually some cases where it is important to keep a second copy of an optical disc: One medium might get lost, burst while spinning in the drive, or it may be destroyed due to mishandling. However such cases of complete data loss are rare as long as optical media are handled properly.

It is more likely that the medium starts to gradually lose data after a few years - a nearly unavoidable aging process. When the medium is regularly used (or scanned for defects) the data loss will typically be noticed after 5% to 10% of the medium have already become unreadable. At this point the medium is unusable as a whole, but maybe 90% of it is still readable. *On the other hand a full backup copy of the medium is not required; we simply need a method for recovering the missing 10% of data.*

This is where dvdisaster comes into play. Consider this:
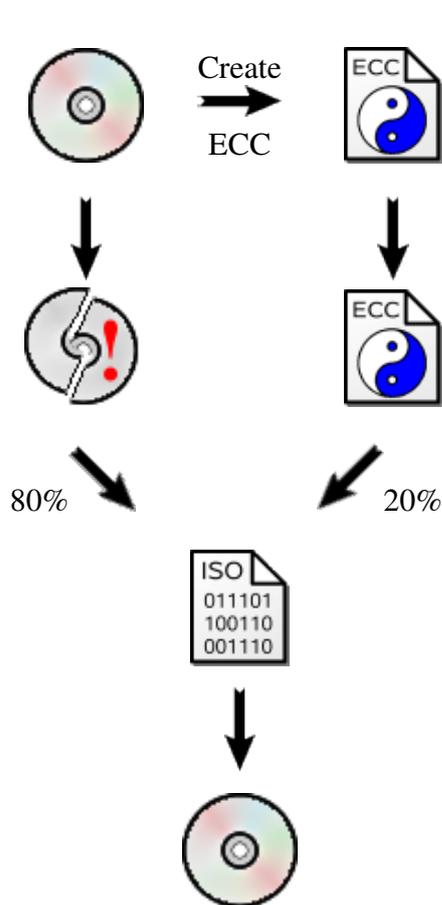
This time we do not make a full backup. dvdisaster is used to create error correction data ("ECC") which can recover up to 20% of a degraded medium. The value of 20% was chosen to have a safety margin over the expected data loss of 5-10%.

Wenn the medium fails at a later time, its contents are recovered from its still readable parts and from the error correction data.

For a successful recovery at least 80% of the data must still be readable from the medium, and the remaining 20% are recalculated from the error correction data.

The completely recovered data is now available as an ISO image on the hard drive (the medium remains defective as physical data loss is irrevocable).

Write the image to a blank medium using your favourite optical disc authoring software.

You now have a new error-free medium.

As you have seen the data recovery took more steps then doing a conventional backup and restore. So let's summarize the pros and cons of dvdisaster compared with conventional backup:

Advantages

- dvdisaster uses less storage. When using error correction data with a 20% recovery capability, protecting 5 media requires only one additional medium for the ECC data.

- Since all media will eventually age and start losing data in similar places (typically in the outermost region), doing a 1:1 copy might not help at all. Both copies may turn out defective in the same places after a few years.

Similarities

- Both backup copies and error correction data must be created before the master disc fails. You can't create them from an already defective medium.

Disadvantages

- If the recovery capability of the error correction data is exceeded (or the medium gets lost), no data can be recovered! Especially take note that error correction data with a repair rate of 20% together with a 75% readable medium does not result in 95% recovery. In that case, nothing beyond the 75% readable data from the medium can be recovered.

Some of these points are also discussed in Error correction data vs. full backup in the "Overview" section, from a slightly different view point.

2.2.2 The idea behind the error correction
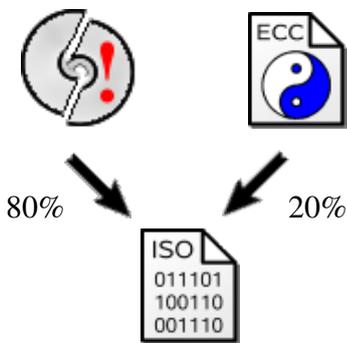
The example from the previous page told us how dvdisaster reconstructs data by using the still readable parts of the medium together with the error correction data.

In order to get the most out of dvdisaster a basic understanding of the error correction method is helpful. And while we are at it we can refute a misunderstanding we sometimes hear - the error correction data is not simply a copy of the last 20% data sectors. That'd really be a cheap shot ;-)

Example: Anna's desk drawer PIN
Anna has got a desk whose drawers can only be opened after entering the numbers "8 6 2 3" into a code lock. Since the drawers do not contain any sensitive information she decides to note down the numbers directly on the desktop:

$$8 \quad 6 \quad 2 \quad 3$$

Anna is cautious and expects one of the numbers to become unreadable by accidentally pouring ink over it. Therefore she also notes down the sum of the four numbers (the "+" and "=" signs have only be added for clarity):

$$8 + 6 + 2 + 3 = 19$$

After a while one of the numbers indeed gets covered by an ink spot:

$$8 + \quad + 2 + 3 = 19$$

But this is not a problem as Anna can re-calculate the missing number x by rearranging the still readable parts of the equation:

```
8 + x + 2 + 3 = 19,   hence
```
```
x = 19 - 8 - 2 - 3,   and therefore x = 6.
```

It is easily seen that any one of the original five numbers can be recovered from the remaining four. The example also demonstrates some important properties of the error correction:
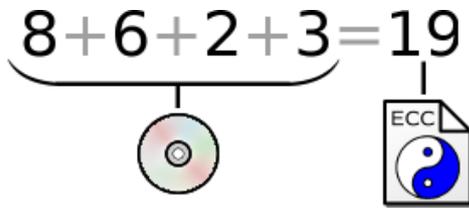
For a given set of data (e.g. the numbers "8 6 2 3") additional error correction data (e.g. the sum "19") can be created so that a lost datum can be re-calculated from the remaining data.

The same principle is used in dvdisaster; the protected sequence of numbers is nothing else than the ISO image of an optical disc.

The concept of **redundancy** can be explained as follows:

- One "error correction number" is calculated for four input numbers. 1 of 4 (or 1/4) relates to a redundancy of 25%.

- From one error correction number we can re-calculate exactly one missing number, or at most 25% of data. The redundancy is equivalent to the maximum capacity of the error correction.

- Additional storage required for the error correction data is also determined by the redundancy (again, 25% in the example).

dvdisaster uses the term of redundancy accordingly. In addition please observe that

- no data can be recovered when the data loss exceeds the redundancy (the equation in the example can not be solved for two or more unknowns).

- the error correction data must be calculated at a point in time where all data is still present / readable.

The above shown example does not generalize into an error correction scheme for recovering more than one missing data value. To do so a more powerful equation system is needed which can be solved for more than one missing value. dvdisaster uses a Reed-Solomon code which does have such properties; however the required math is usually not taught in school. Interested readers are therefore referred to the respective books in coding theory.

2.2.3   Using dvdisaster the right way

Let's demonstrate how Jane uses dvdisaster.

10. Feb. 2009		Jane creates a new CD with important data.
		To protect the CD from data loss she creates error correction data with dvdisaster. She keeps both kinds of data for later usage.
14. May 2010		Jane knows that during daily use not all data of her CD might be accessed. So after a year has passed she scans the CD for read errors to make sure that it has not developed any defects in seldom used data regions. However after one year the CD is still perfectly readable.
19. Aug 2012		Two more years have passed and Jane notices that some data on the CD is no longer readable. A scan for read errors confirms that the CD has become defective due to aging.

read

Jane uses dvdisaster to read as much sectors as possible from the defective CD into an ISO image.

reconstruct

By using the error correction data Jane recovers the missing parts in the ISO image.

Write new CD

Jane writes a new CD from the recovered ISO image. She keeps the error correction data for the new CD as it may also become defective in the future.

2.2.4 Using dvdisaster the wrong way

Joe bets on his media keeping their content without additional protection.

10. Feb. 2009		Joe creates two CDs containing important data. But he does not make any provisions against data loss on those media.
14. May. 2010		Joe uses his CDs regularly. After one year they are still perfectly readable.
19. Aug. 2012		After two more years Joe notices that some data on one CD is no longer readable.

scan

20. Aug. 2012

Joe downloads dvdisaster and performs a read error scan. He finds out that the CD contains 25000 unreadable sectors. A scan of the second CD reveals that it has developed 1500 unreadable sectors gone unnoticed so far.

reading

21. Aug. 2012

Joe uses dvdisaster to read as much sectors as possible from the defective media. But since he does not have error correction data there is no way of recalculating the unreadable sectors.

many
reading
attempts

05. Sep. 2012

Joe takes advantage of dvdisaster's feature to complete images through multiple reading passes. He moves the defective images to several computers to perform reading attempts with different drives. After two weeks of trying at least all missing sectors from the second CD have been read. However on the first CD still 21000 sectors remain unreadable by any drive he tried.

only one CD
recovered

06. Sep. 2012

Joe dismisses the first CD as unrecoverable and considers himself lucky to have a complete image from the second CD again. However if he had created error correction data in time, he'd probably[2] saved two weeks of reading attempts and recovered the contents from both CDs.

[2]The error correction assumes a typical aging process. If the CD gets severely damaged it becomes unrecoverable even with error correction data. Do not rely on dvdisaster alone for protecting important data; instead employ additional measures like creating additional copies on different types of media.

2.3 Scanning media for errors

Task   The medium is scanned for unreadable sectors.

Required:
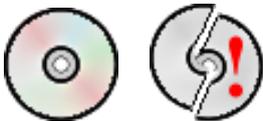
A medium in any state (good or containing read errors)

If error correction data is available additional tests are carried out. However scanning will also work without error correction data.

What to do: 1. Configure basic settings

2. Scan the medium

3. Interpret the results

Related functions: Reading of damaged media and

recovering images

2.3.1 Basic settings

Figure 3: Opening the configuration dialog.

The relevant tabs are described on the next pages. They are found in the configuration dialog. Open the dialog by selecting the symbol marked green in the screen shot (, see figure 3). The symbol may look different due to the symbol theme you are using.
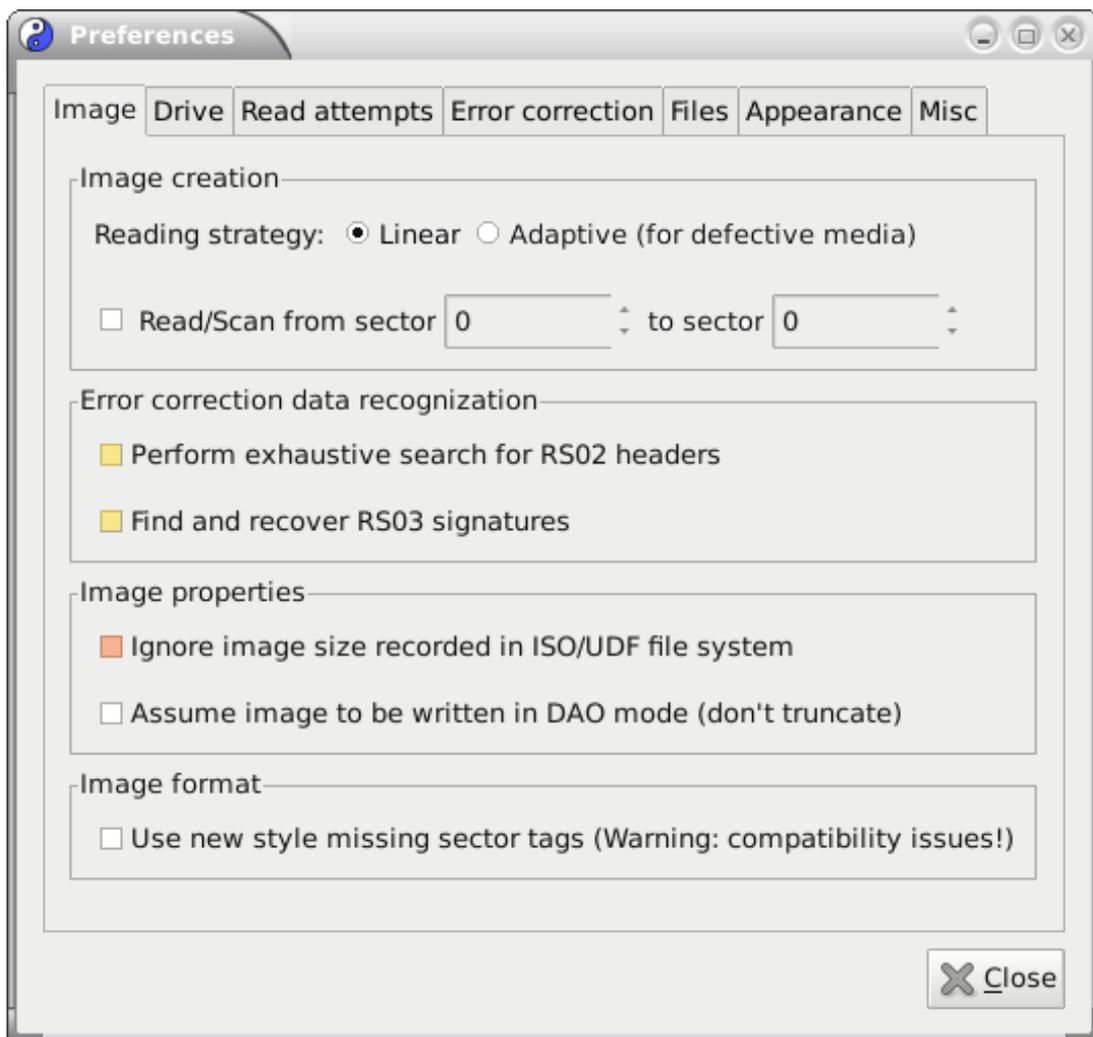
Figure 4: The "Image" tab.

Error correction data recognition. If you are sure that your medium contains embedded RS02 or RS03 error correction data, activate the respective items (marked yellow). Using the error correction (ecc) data will improve scanning results, but searching for non existing ecc data will cost several minutes.

Image properties. Selecting the proper method for determining the image size is important. Make sure that "Ignore image size recorded in ISO/UDF file system" (marked red) is switched off. This option should only be used in special cases during image recovery.

Adjust the remaining settings as shown in the screen shot.
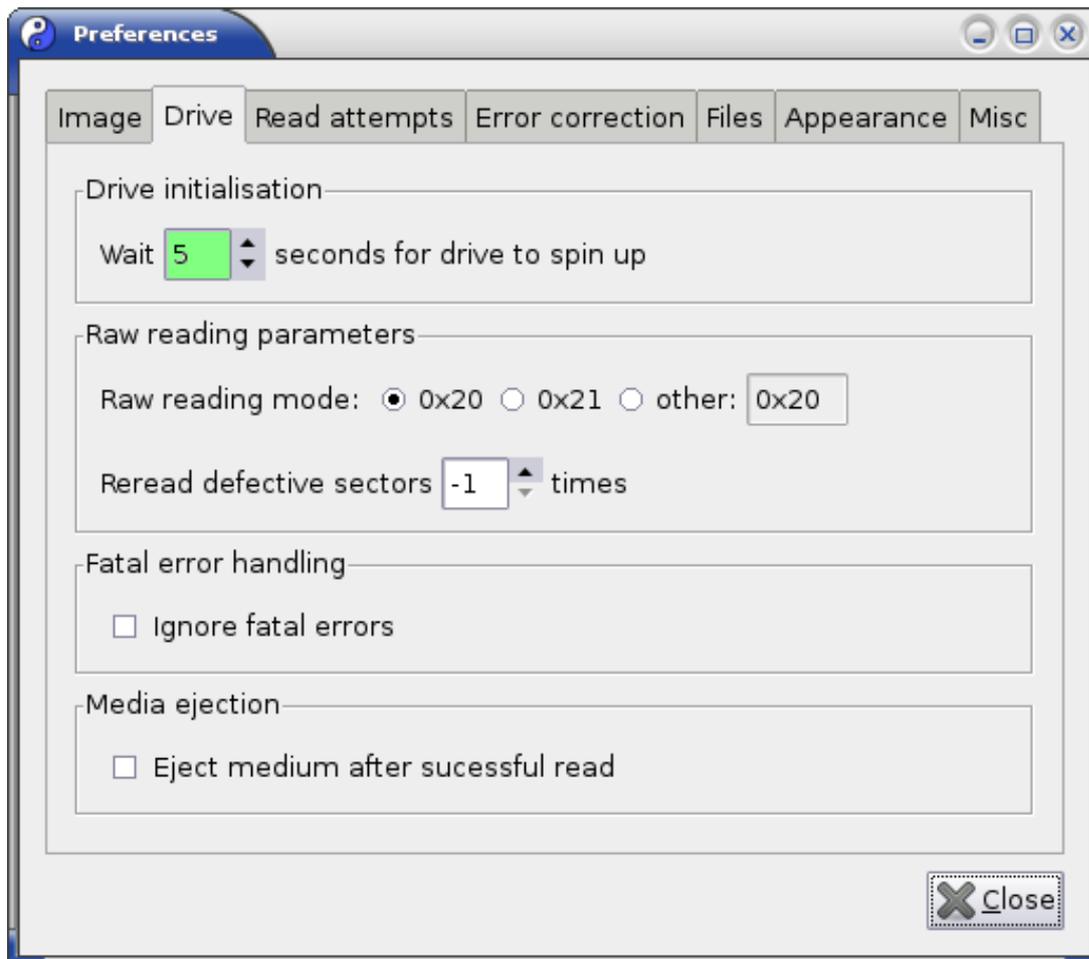
Figure 5: The "Drive" tab.

Drive initialization.   Reading data from the drive while it is spinning up can generate spurious error reports. Adjust the spin up time for your drive (typically 5-10 seconds) in the field marked green to make dvdisaster wait for the appropriate time.

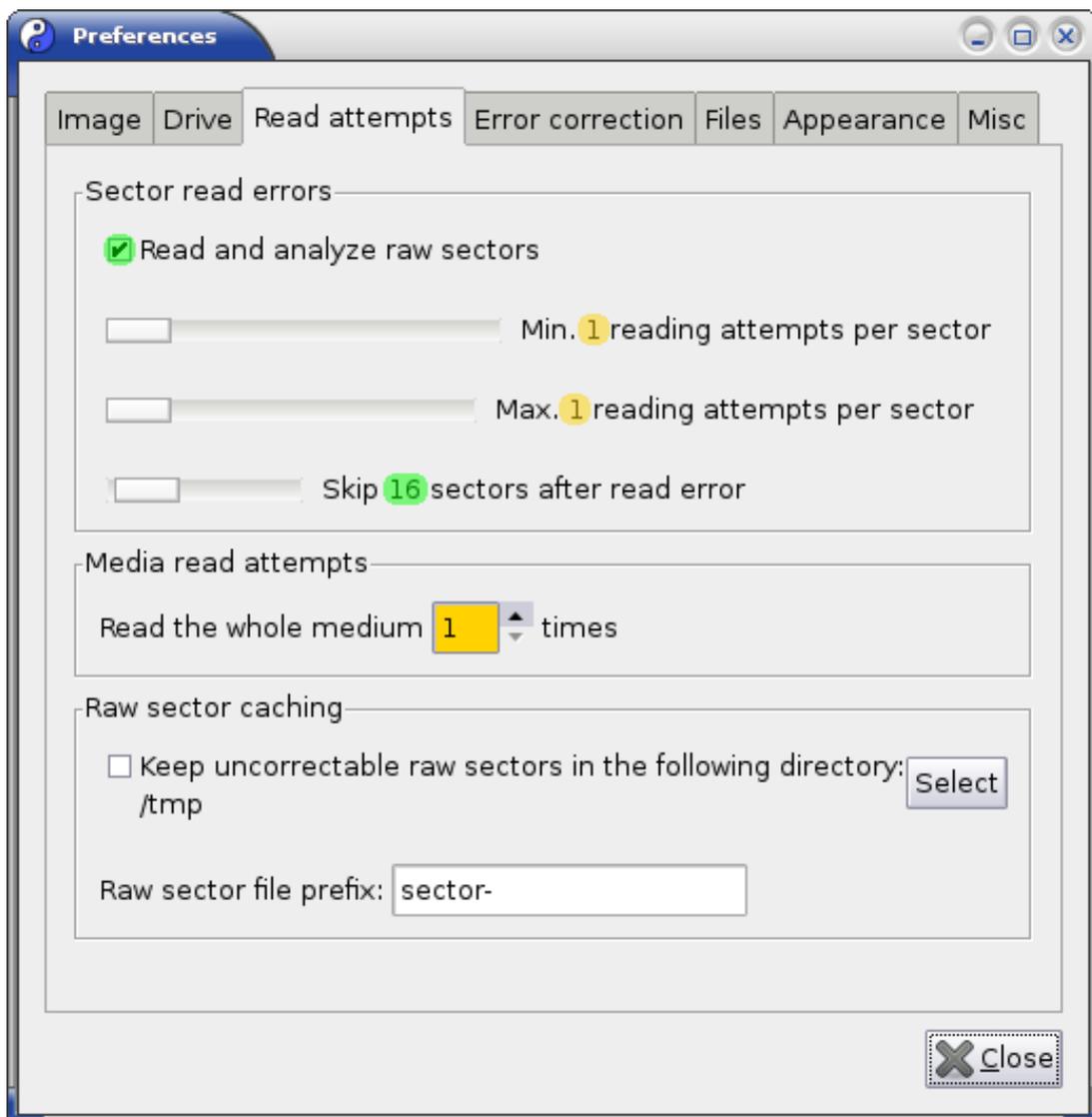Leave the other settings at the values shown; you can optimize them later.

Figure 6: The "Read attempts" tab.

Sector read errors.   The option "Read and analyse raw sectors" (first green marking) uses C2 analysis and possibly more raw data reported by the drive for a better assessment of CD media quality. This setting does nothing for DVD and BD media, but it is safe to remain activated unless it causes problems with your drive reading CDs.

Adjust the reading attempts settings as shown here. Using larger values causes unnecessary reading activity but will not improve the scan. After a read error no less than 16 sectors should be skipped (second green marking); when scanning badly damaged media this setting can be optimized using larger values.

Media read attempts.   Performing multiple read attempts is not recommended during a scan; set the number of retries to 1 in the three places marked in orange. Collecting raw sectors should also be off during the scan.
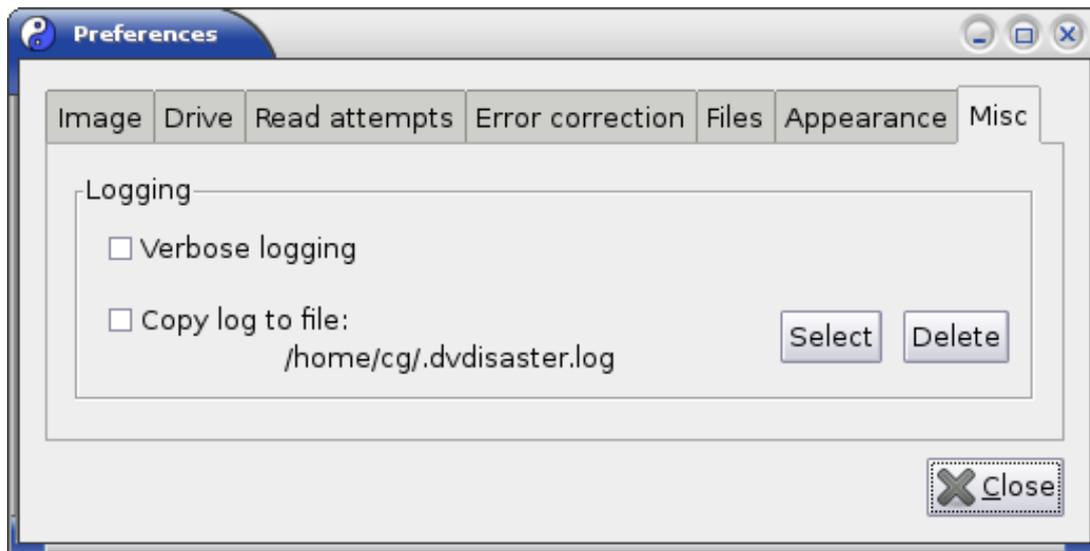
Figure 7: The "Misc" tab.

The "Misc Tab".   Currently this tab only has functions for creating log files. This is helpful for reporting defects but should be left off during normal operation.

Not used tabs.   The "Error correction" and "Files" tabs have no influence on scanning media. The "Appearance" tab allows you to adapt the output colors to your taste, but these have no further effects on the scanning process.

2.3.2 Scanning for errors - Walkthrough

Please make sure that dvdisaster has been configured as described in the basic settings section as some settings might negatively affect the scanning results. Then perform the following steps:
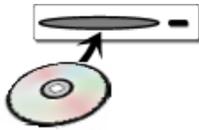
 Insert the medium you want to scan into a drive which is directly connected to your computer. You can not use network drives, software drives and drives inside virtual machines.

 Close any windows which may be opened by your operating system for viewing or performing the medium contents. Wait until the drive has recognized the medium and the medium has spun down.
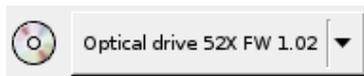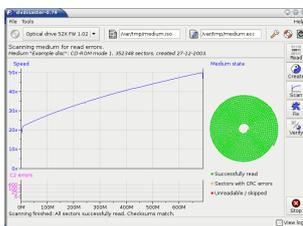
 Select the drive containing the medium in dvdisasters drop down menu.

 Select the error correction file for this medium if you have one available. Ecc data from RS02 or RS03 augmented media is used automatically.
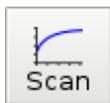
 Start the scan by **clicking the "Scan" button.**

 Watch the scanning progress. Do not perform any other actions on your computer while the scan is running. Opening or working with other programs as well as moving other windows around might affect the scanning results.
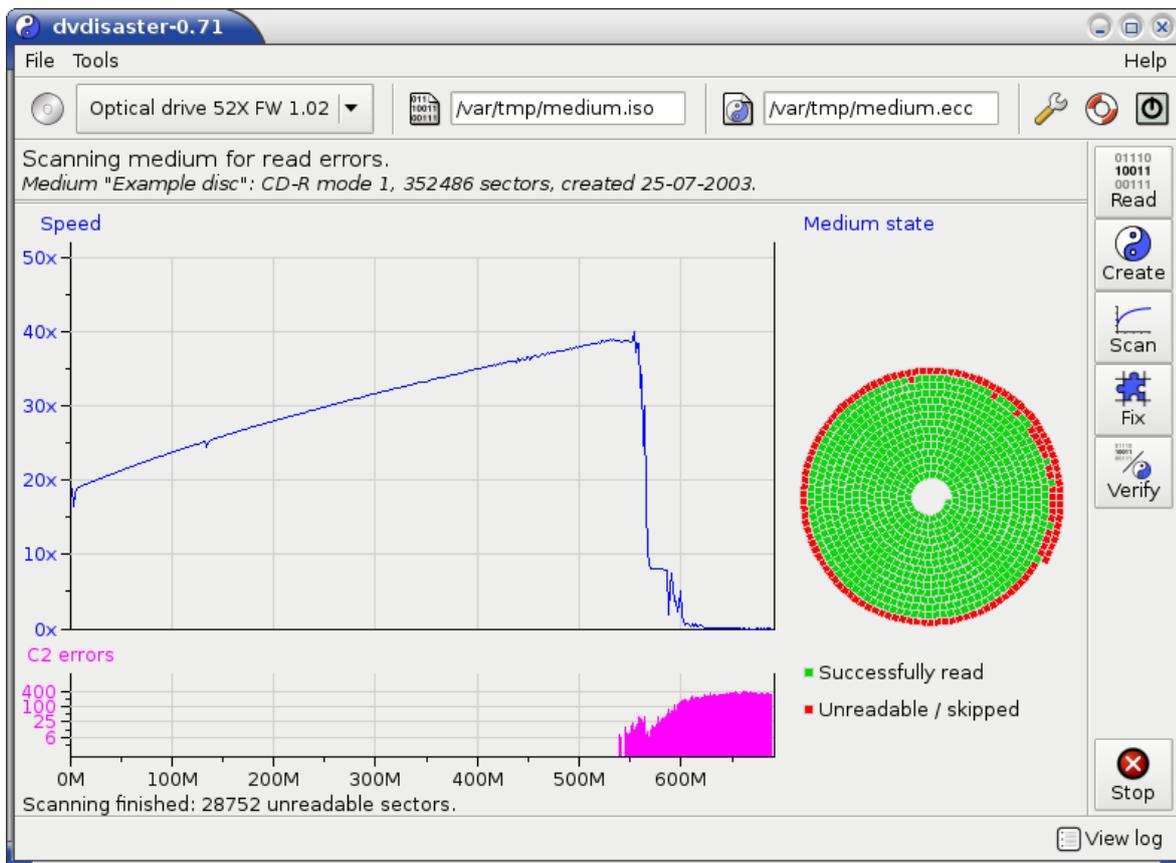
2.3.3   Interpreting the results

Figure 8: Scanning the medium.

Overview.   dvdisaster provides several information about the scanning results (see fig. 8):

- The spiral under **"Medium State"** (to the right).

  The spiral provides information about the medium readability. The medium is fully readable when all segments of the spiral are colored green. Yellow or red blocks mark places where data could not be correctly read from the medium. The total number of unreadable sectors is printed in the "Scanning finished:" message at the window bottom.

- **"Speed"** - The reading speed curve (upper left).

  The reading speed is not an absolute gauge of the medium health, but it is usable as a rule of thumb: The more regular the curve, the better the medium. You will find examples of good and bad reading speed curves on the following pages.

- **"C2 errors"** - A medium state gauge provided by the drive (down left).

  This kind of analysis is currently only available for CD media. CD drives have a built-in error correction which can eliminate small data losses caused by minor defects on the medium. The number of C2 errors is a measurement of how often the drive needed to employ its internal error correction during the read - this value should be zero on good media.
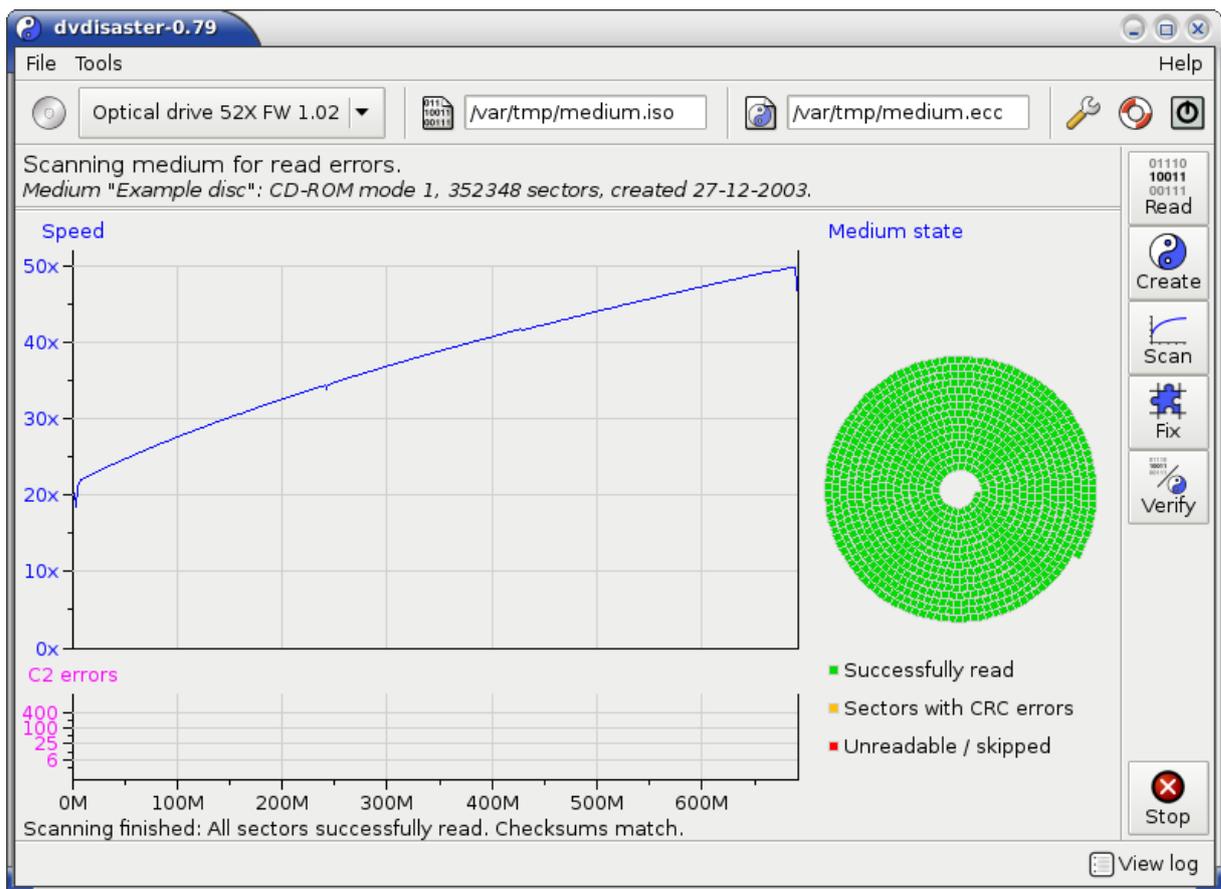
Figure 9: Good CD.

Example of a good medium: This screen shot shows a perfect CD: All blocks under "Medium state" are green, no C2 errors have been reported and the reading curve runs smoothly. A rising reading speed is normal for most media (see the next screen shot for a counter example). The small spikes at the beginning and at the end of the curve are normal; minor glitches like the one shown at 250M are also harmless.
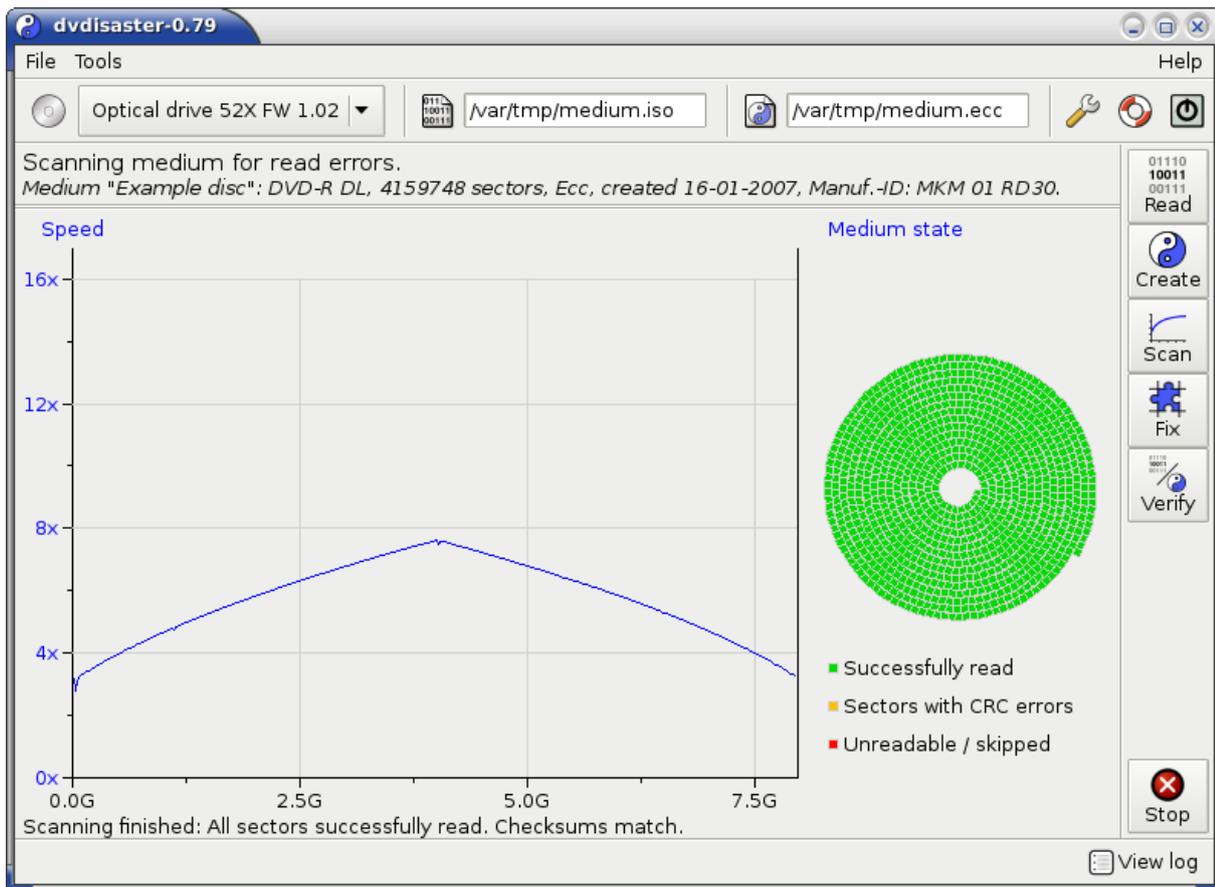
Figure 10: Good two-layered DVD.

Sometimes the reading curve won't rise steadily: Multi-layered media might yield reading curves which are rising and dropping in a symmetric pattern. Not shown but also possible are flat curves without any change in reading speed (most typically seen with DVD-RAM).
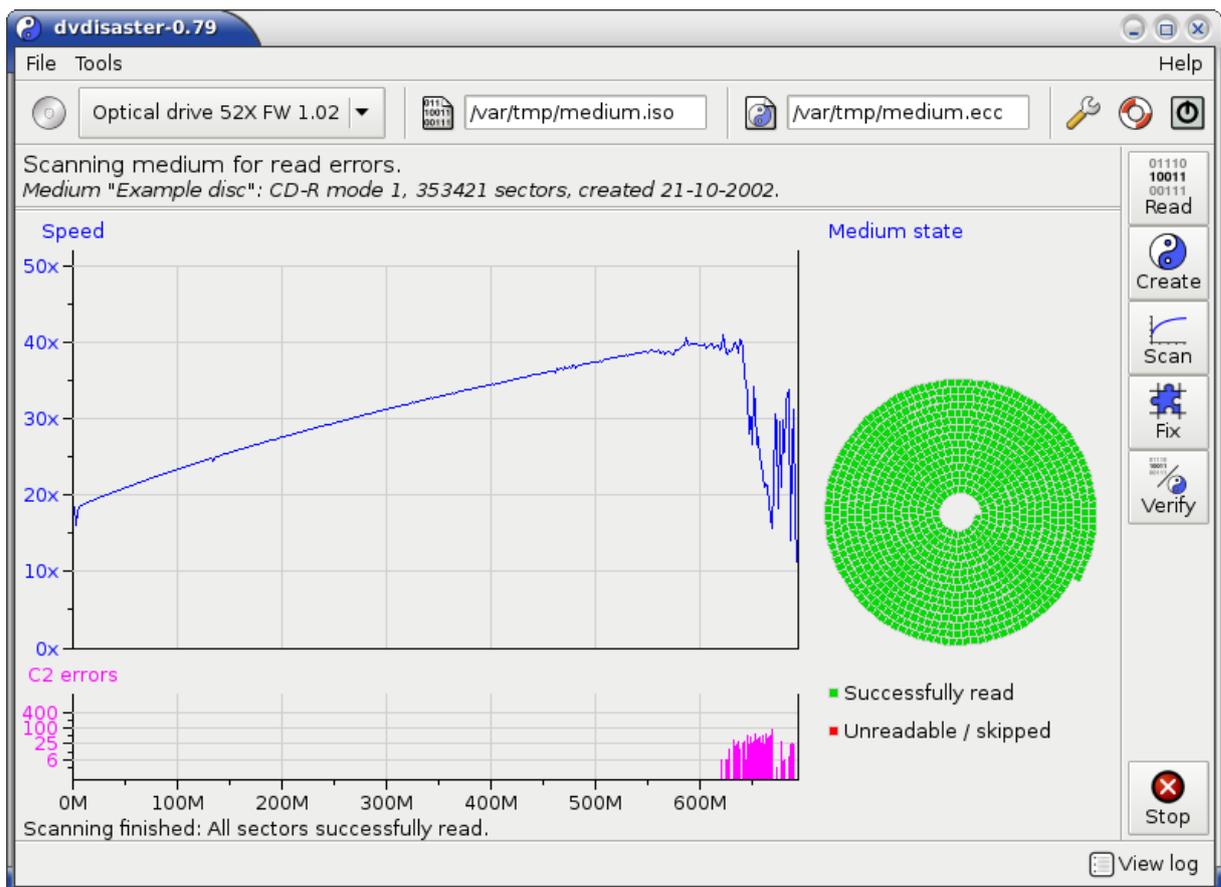
Figure 11: Weak CD.

An example of a weak medium. This medium is still readable as indicated by the green spiral shown under "Medium state". However there are clear signs of serious trouble ahead: The drive must slow down significantly towards the end of the medium in order to read from it. Note the steep fall of reading speed after the 600M mark. This comes along with C2 error rates rising to the 100 mark; this is another warning that the medium is decaying in the outer region. If you have not created error correction data this is probably the last opportunity to do so as the medium will develop the first read errors soon.
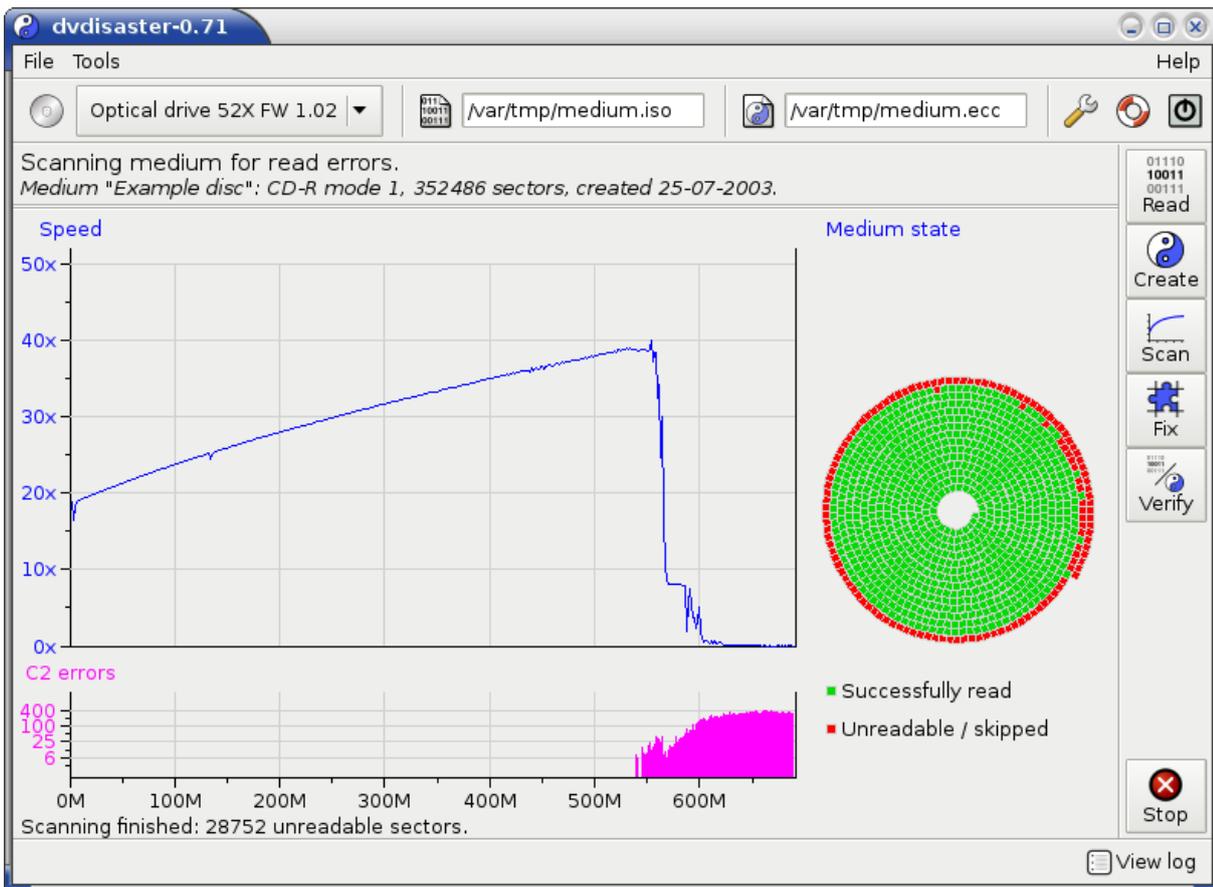
Figure 12: Defective CD.

Example of a defective CD. The red sectors in the spiral visualize large unreadable sections in the outer region of the medium. At the bottom of the window you will find the information that the medium contains 28752 unreadable sectors. This sums up to about 8.2% defective sectors (of 352486 sectors total) and is well within the recovery bounds by error correction (ecc) data made with default settings - if you have made the ecc data in time! Otherwise the contents of the red sectors are lost since ecc data cannot be created from already defective media.
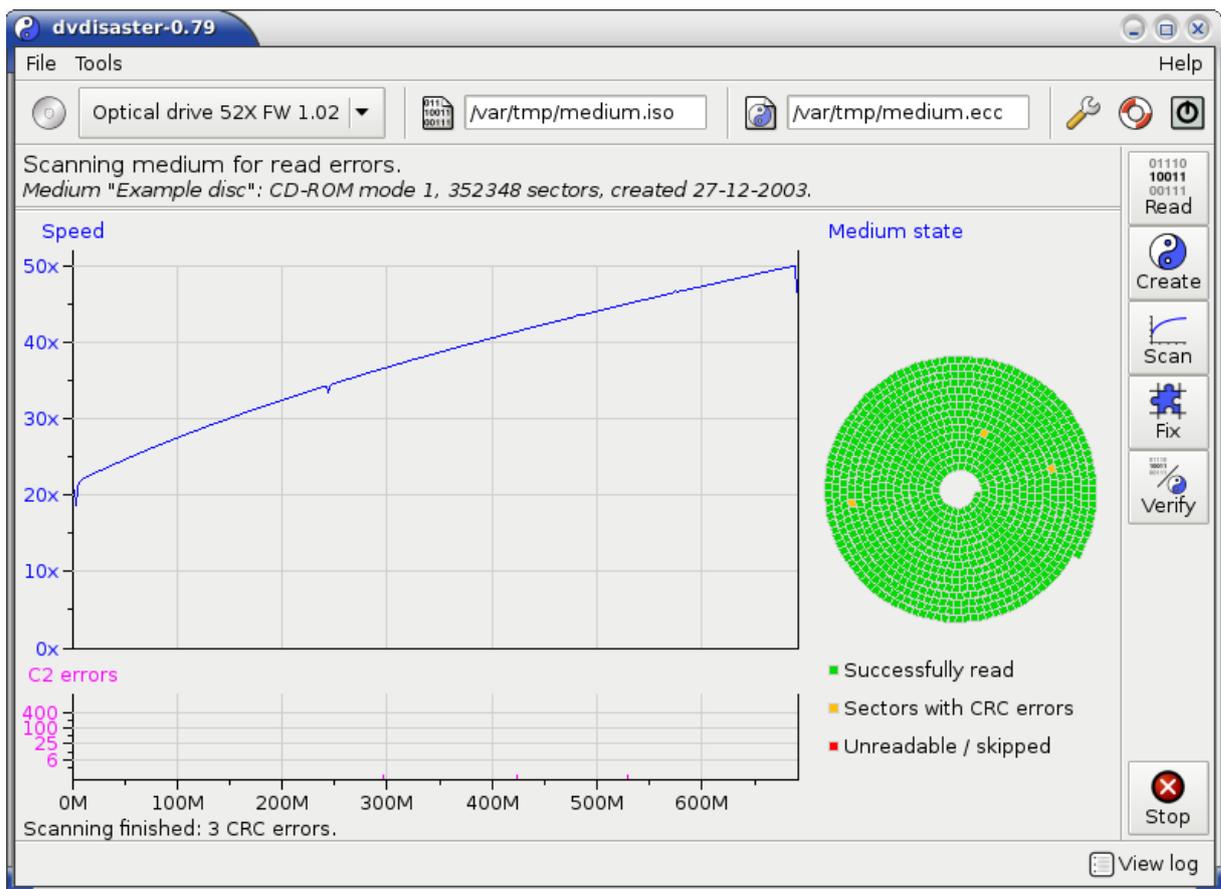
Figure 13: Checksum errors.

Checksum errors. Yellow spots in the spiral depict places where the medium was fully readable, but the data read did not match checksums in the error correction data. There are two main causes:

- The image has been manipulated after the creation of error correction data and before writing it to the medium. This can happen when the image is mounted with write access after ecc data has been created. Typical signs are CRC errors in sector 64 and in sectors 200 to 400 as the system updates the file access times there. Performing a data recovery using dvdisaster is typically harmless in this situation.

  However if you have modified files in the image after creating the ecc data, the error correction data will be both worthless and dangerous. Applying a recovery to the medium will restore the image state at the time the ecc data has been created, and this will obviously not represent the most recent contents of the medium.

- There are technical problems with the computer system, especially in mass storage communication. Perform the scan again and observe the CRC error locations. If CRC errors disappear or surface at different locations your system might have defective RAM, bad drive cabling/controllers or incorrect clock speeds.
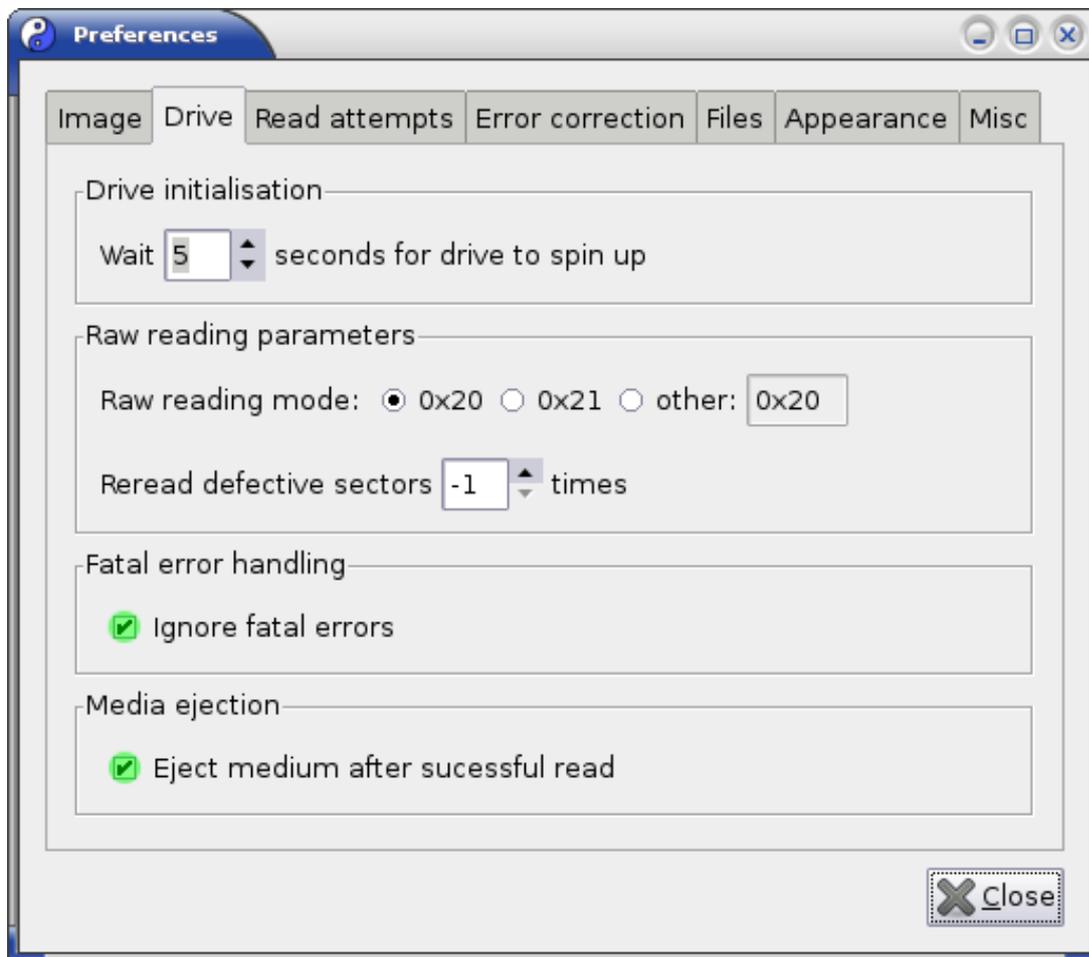
2.3.4   Advanced settings

Figure 14: Advanced settings in the "Drive" tab.

Fatal error handling. dvdisaster will normally abort the scan when the drive reports a fatal internal error like mechanical problems. The intention is to avoid damaging the drive. However some drives will erroneously report such problems when they get confused by damaged media. If you have such a drive you can use this option to force the scan to continue.

Media ejection. dvdisaster tries to eject the medium after a successful scan if this option is activated. However ejecting the medium might be prohibited by the operating system so this is not guaranteed to work. For example if upon media insertion a window is opened for performing the contents it may not be possible to automatically eject the medium.
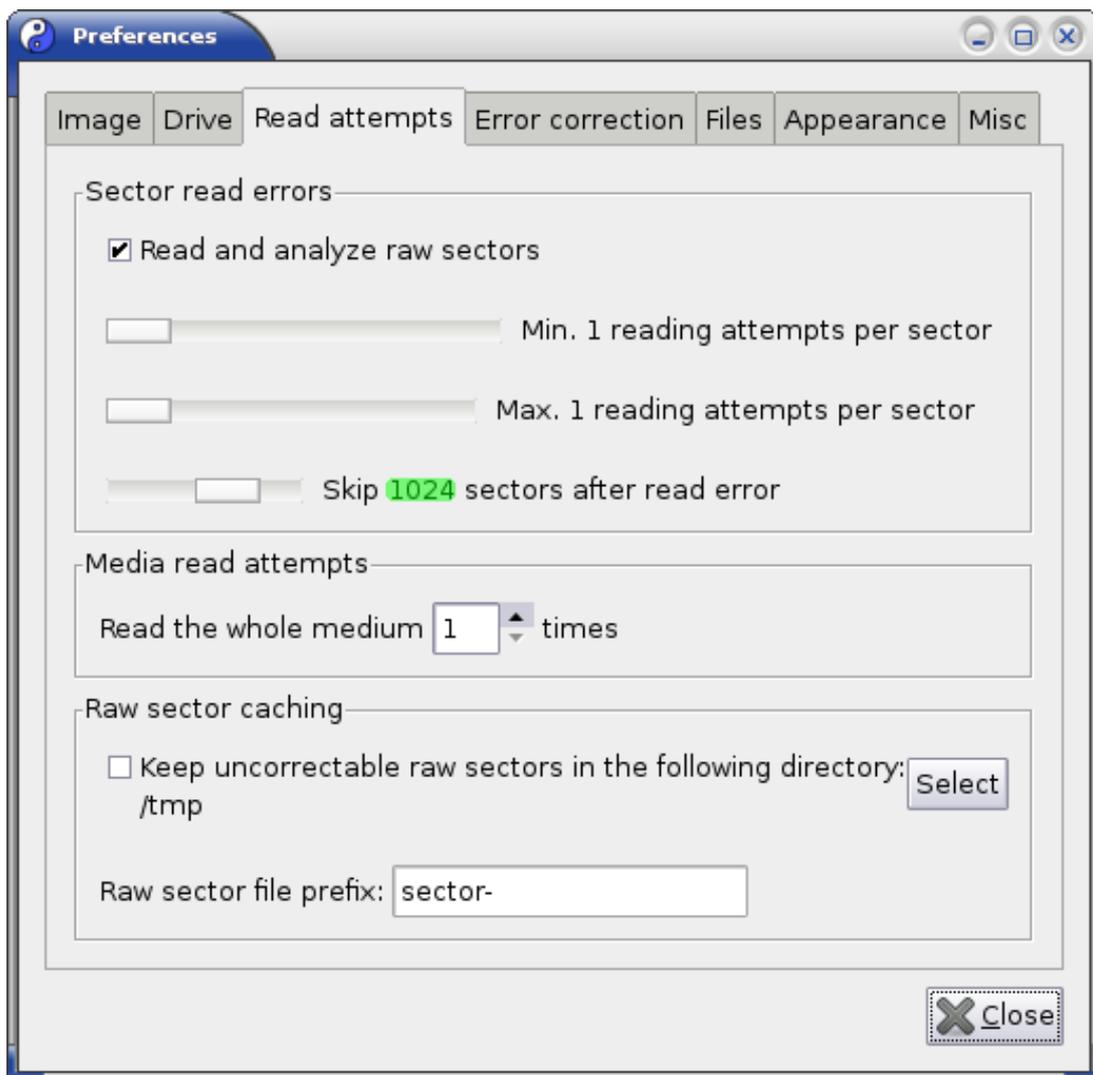
Figure 15: Advanced settings in the "Read attempts" tab.

Sector read errors. Attempts for reading defective sectors cost a lot of time. Since it is likely to encounter another defective sector after hitting a read error, skipping a few sectors after a read error saves time and reduces wear on the drive. If you only want a quick overview of a damaged medium setting this value to 1024 might help. But keep in mind that all skipped sectors are counted as being defective so the number of reported errors becomes higher and less accurate.

2.4   Selecting the right type of ecc data

Error correction data can either be created in form of a separate error correction file or it can be placed directly onto the medium.

Answering the following questions can quickly guide you to the right ecc data format:

- *Do you need error correction data for a medium which has already been written?*

  In that case, you need to create an error correction file because an already existing medium can not be augmented with error correction data.

- *Are you planning to write the medium right now?*

  If the medium is full or nearly full (less than 20% free), not enough space might be available for storing the error correction data. It is strongly recommended to create an error correction file in that case. Otherwise, you can put the error correction data directly onto the medium. To do so you must create an ISO image first and then augment it with error correction data before you write it to the medium.

More information on keeping error correction data.
dvdisaster helps protecting your media from data loss by forehanded[3] creation of error correction data. Error correction data must be treated like normal backup data, e.g. you need to keep it available during the whole lifetime of the respective medium.

The easiest way is storing the error correction data on the medium you want to protect. But this is only possible if the medium has not yet been written: To employ this method you need to create an ISO image first, then augment this image with error correction data, and finally write the augmented image to the medium.

If the medium has already been written, or insufficient space is left for augmenting the image, you still can create error correction data in form of a free-standing error correction file. This file must then be stored somewhere else, e.g. you need to take additional provisions to archive your error correction files.

More information about the pro and con of these methods can be found in the background information section.

---

[3]Let's repeat again for clarity: Error correction data must be created before the medium becomes defective. It is not possible to create error correction data from defective media; in that case unreadable sectors can not be recovered.

2.5   Creating ecc data as a separate file

Task   An error correction file is created for an optical medium.

Note: This page describes how error correction data is created and placed into a separate file. There is also a method for placing the error correction data directly onto the medium.
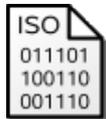
Would you like help on deciding between these two methods?

Required:

A good, error free[4] medium,

or

an already existing and complete[4] ISO image of the medium (e.g. the image used for writing the medium).

What to do: 1. Configure basic settings

2. Create the error correction file

3. Archive the error correction file

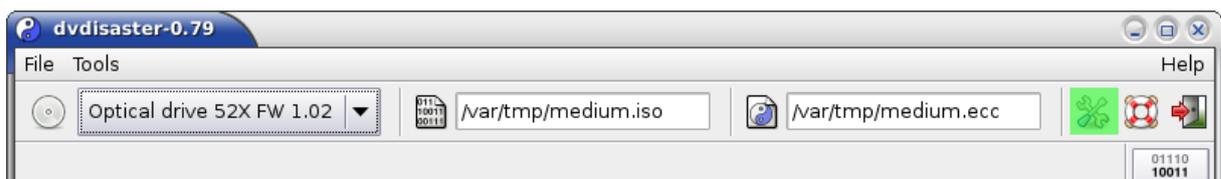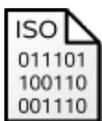### 2.5.1   Basic settings for reading the image from the medium

Figure 16: Opening the configuration dialog.

The relevant tabs are described on the next pages. They are found in the configuration dialog. Open the dialog by selecting the symbol marked green in the screen shot (, see figure 16). The symbol may look different due to the symbol theme you are using.

If you already have an ISO image available you can skip the next three tabs and proceed with the error correction settings. But make sure that you really have an ISO type image; other formats like ".nrg" do not produce usable error correction data.

---

[4]Error correction data must be created before any data loss occurs: It is not possible to create error correction files from an already defective medium.
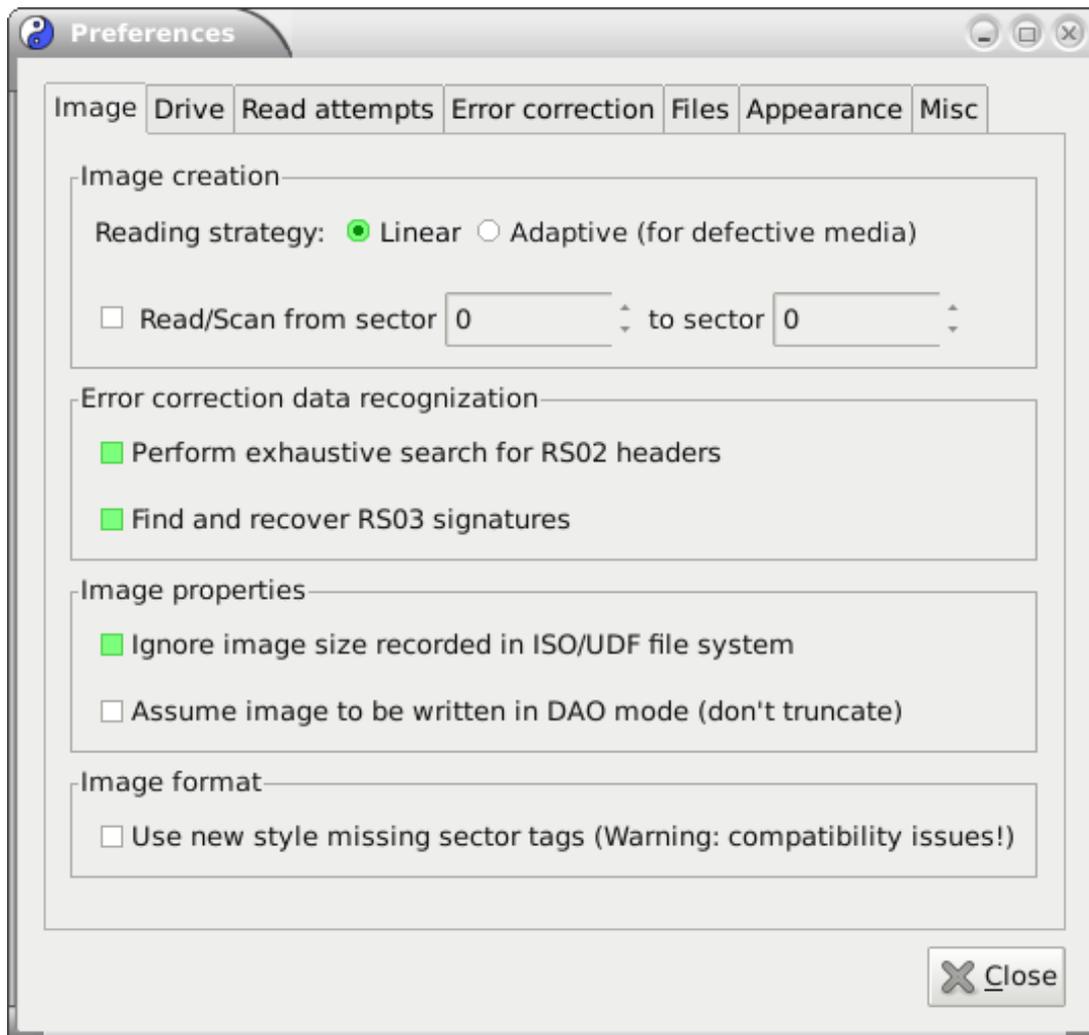
Figure 17: The "Image" tab.

Reading strategy.   Make sure that the linear reading strategy is selected (marked green).

All other options should be turned off; especially those marked green as they might have adverse effects on the image reading process.

Figure 18: The "Drive" tab.

Drive initialization.   Reading data from the drive while it is spinning up can generate spurious error reports. Adjust the spin up time for your drive (typically 5-10 seconds) in the field marked green to make dvdisaster wait for the appropriate time.

Leave the other settings at the shown values.
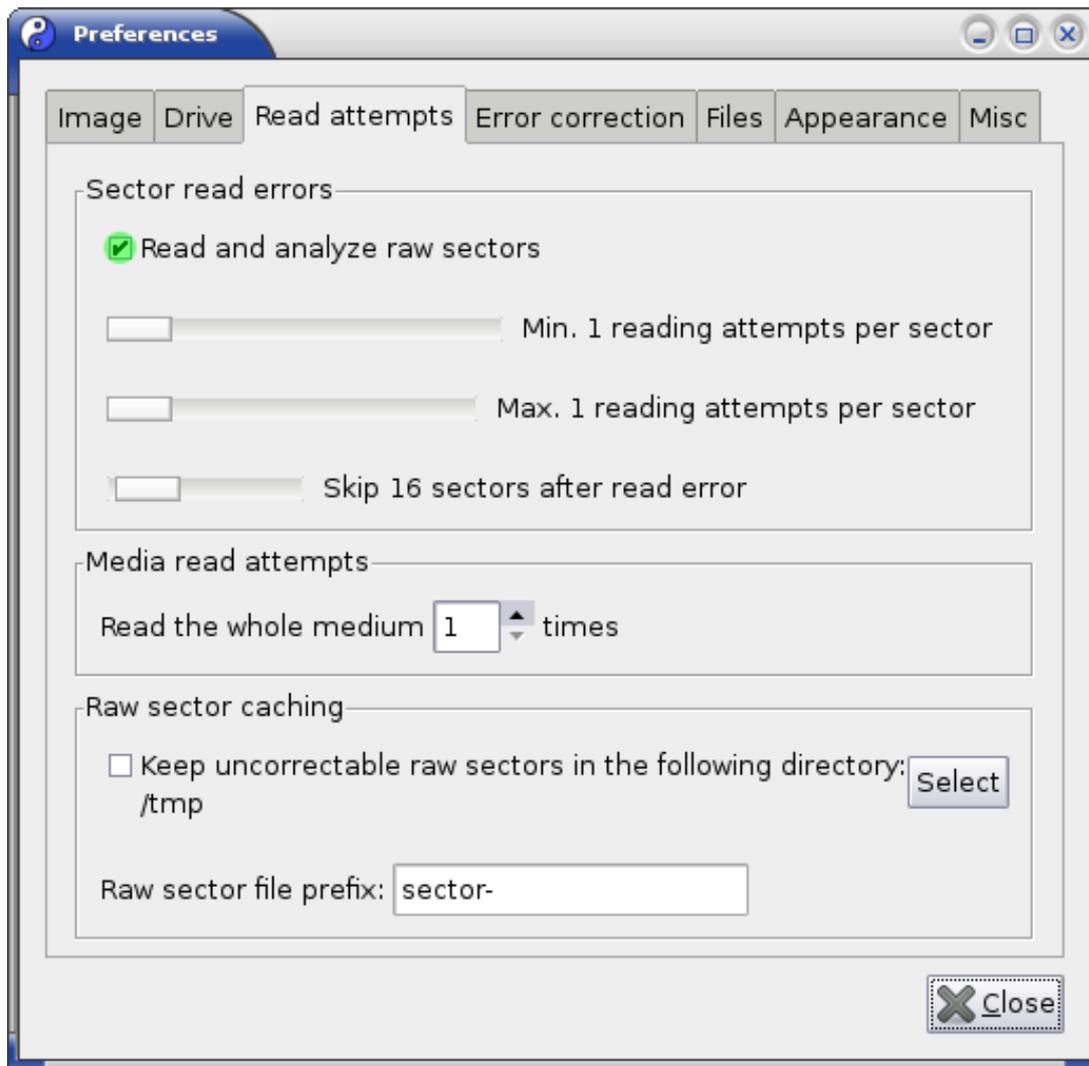
Figure 19: The "Read attempts" tab.

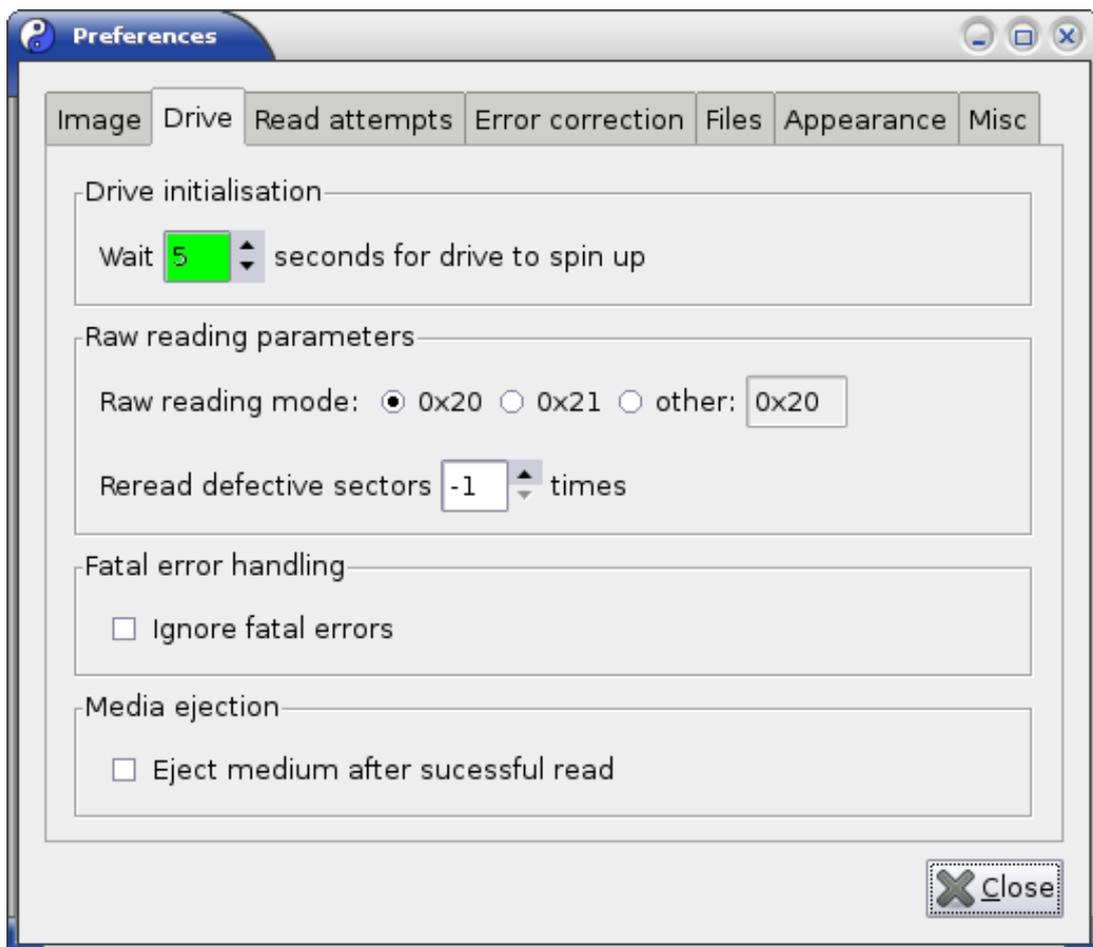**Sector read errors.**   The option "Read and analyze raw sectors" (marked green) uses additional information provided by the drive to check the integrity of read data. This is recommended as we are interested in creating error correction data from a properly read image.

On the other hand since error correction data can only be created from fully readable media we do not need multiple reading attempts and caching of raw sectors as shown in the screen shot.

2.5.2   Basic settings for the error correction data
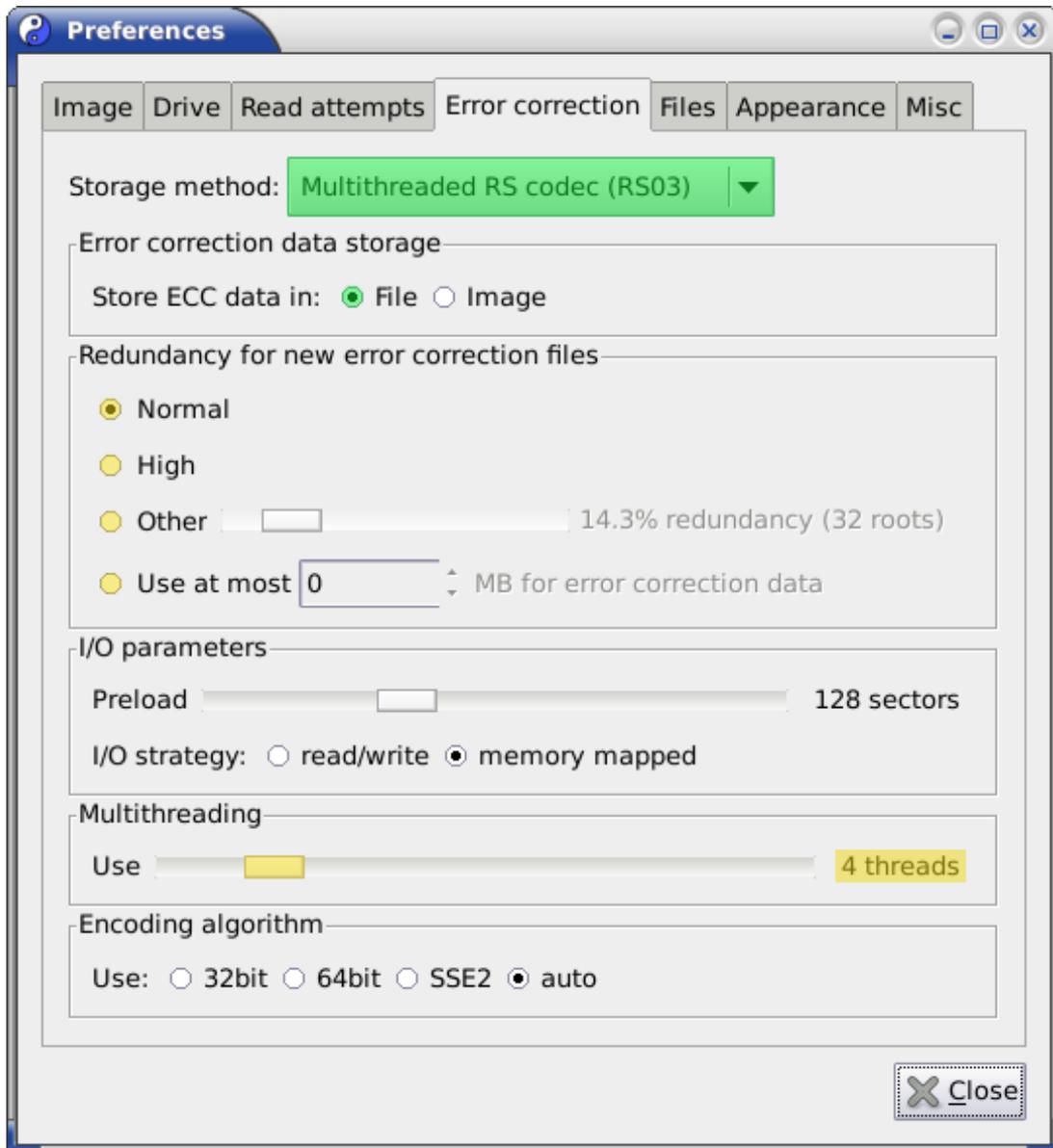
Figure 20: The "Error correction" tab for the RS03 codec.

Error correction files can be created using either the RS03 or the RS01 method (see this comparison for details). It is recommended to use RS03 as it contains all features present in RS01, but encodes much faster due to certain optimizations.

Settings for encoding with the RS03 method.   First select the entry "Multithreaded RS codec (RS03)" in the drop down menu (marked green). After this selection the contents of the tab will display the choices given for the RS03 method.

Error correction storage.   Select "File" (also marked green). This option causes error correction data to be stored in a separate file and will also make the redundancy choice available which is located below.

Redundancy for new error correction files.   Select the redundancy which suits your needs. The redundancy will determine the maximum error correction capability: An error correction file with x% redundancy can correct up to x% of read errors under optimal circumstances. Since the best case is usually not encountered you should add some safety margin to the redundancy by picking one of the following choices (see yellow markings):

- The "normal" and "high" presets provide a redundancy of 14.3% and 33.5% respectively.

- You can freely choose the redundancy by activating the "other" item and dragging the slider.

- By activating the "Use at most" button you can specify the error correction file size in MiB. dvdisaster will choose a suitable redundancy so that the error correction file will be close to but not larger than the specified size.

The redundancy will also determine the size of the error correction file; using x% redundancy will create an error correction file of about x% the size of the image. Using redundancies lower than the "normal" setting (14.3%) is not recommended as the error correction might be overloaded too quickly.

Multithreading.   The RS03 encoder can distribute its workload onto multiple processor cores by using multithreading. On machines with up to four cores, set the number of threads equal to the number of cores (e.g. use 4 threads on a 4 core machine). On machines with more than 4 cores, use one thread less than the number of available cores; e.g. use 7 threads on an 8 core machine - this leaves one core free for housekeeping tasks).

Leave the other settings as shown in the screenshot; you can optimize them later.
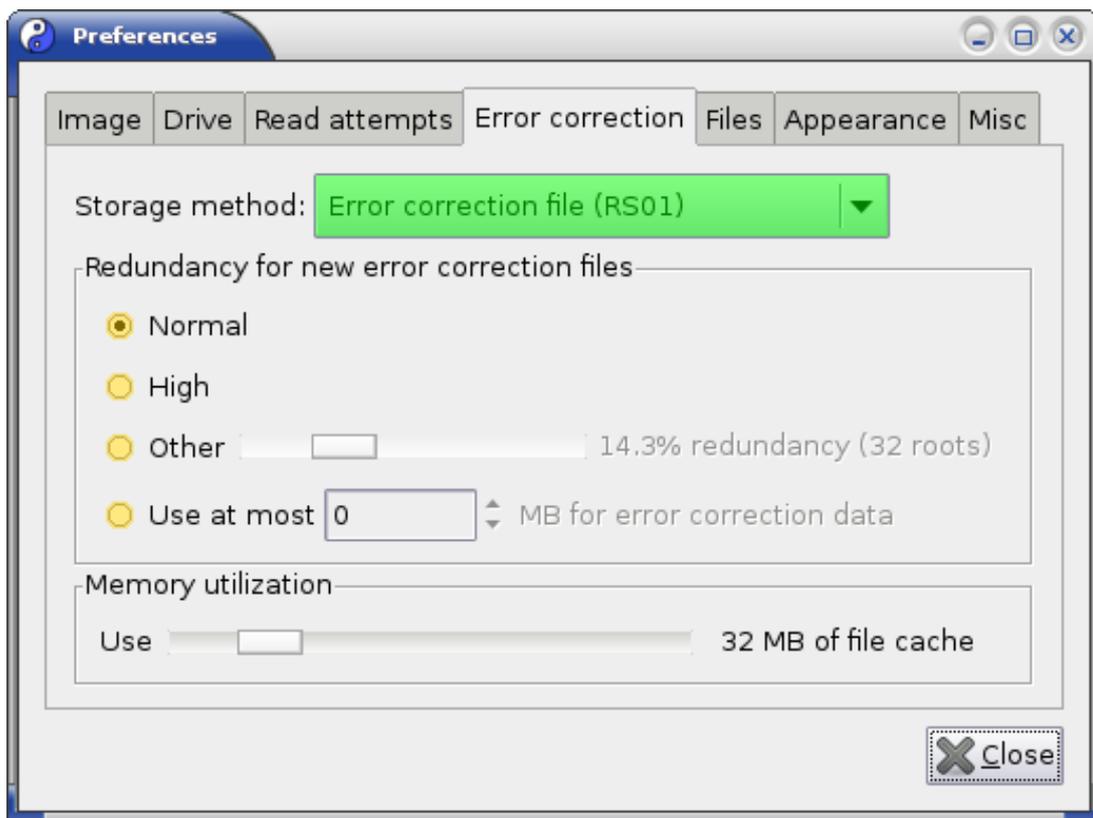
Figure 21: The "Error correction" tab for the old RS01 codec.

Settings for encoding with the RS01 method.   If you want to use the old RS01 method, choose "Error correction file (RS01)" in the "Storage method" drop down menu (green marking; see fig. 21).

Then select a redundancy setting which suits your needs; see the explanations on redundancy for the RS03 method above for more information. For RS01, the "normal" and "high" settings are somehow optimized for speed, but still much slower than their RS03 counterparts.
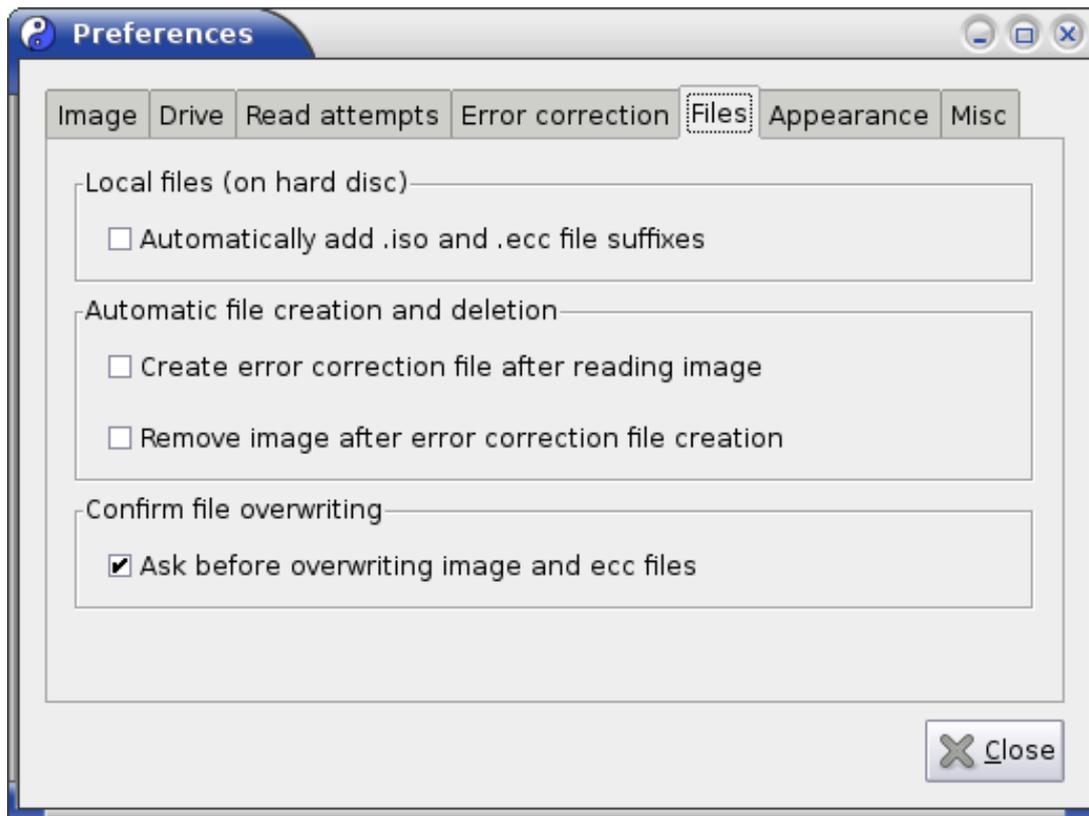
Figure 22: The "Files" tab.

The "Files" tab. In this tab, only activate the option for confirming file overwriting. Leave the other options off for the moment; suggestions for further optimization follow later.

Not used tabs. The "Misc" tab currently has only functions for creating log files. This is helpful for reporting defects but should be left off during normal operation. The "Appearance" tab allows you to adapt the output colors to your taste, but these have no further effects on the error correction data creation.

2.5.3 Creating the error correction file

The error correction file can only be created from an ISO image on hard disk, not directly from the medium. If you have the ISO image already/still available from making the medium, turn over to the next page. Otherwise, follow these instructions for extracting the ISO image from the medium.

Insert the medium you want to read into a drive which is directly connected to your computer. You can not use network drives, software drives and drives inside virtual machines.

Close any windows which may be opened by your operating system for viewing or performing the medium contents. Wait until the drive has recognized the medium and the medium has spun down.

Select the drive containing the medium in dvdisasters drop down menu.
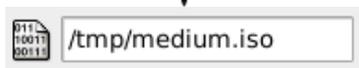
Select a directory and file name for storing the ISO image.

Create the ISO image by clicking the "Read" button.
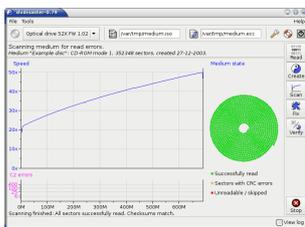
Watch the reading progress. Wait until the medium has been completely read. If the medium turns out to contain defective sectors it will not be possible to create error correction data.

Now continue with the next page to create an error correction file from the ISO image.

Now that we have an ISO image of the medium, we can create the error correction file from it. It you do not have the ISO image yet, please follow the instructions on the previous page for extracting it from the medium.

Select the directory and name of the ISO image for which you want to create the error correction data. It is assumed that the ISO image has been created by some other means, e.g. by using your optical disc authoring software. If you extracted the ISO image from a medium using dvdisaster as described one page before, this field is already filled in correctly.

Select a directory and name for storing the error correction file.

Create the error correction file by clicking the "Create" button.
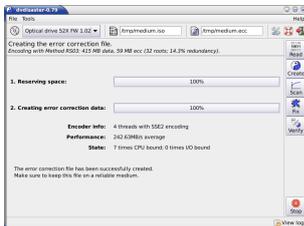
Wait until the creation process finishes. This may take a while depending on the image size, selected redundancy and used encoding method.
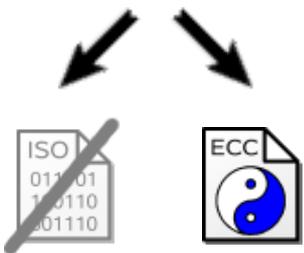
Wrapping up. You can delete the image file now. However you must keep the error correction file and, even more important, protect it from being damaged. Refer to the next page for some suggestions about error correction file archival.

2.5.4 Tips for archival of error correction files

Optical discs are currently among the most cost-effective exchangeable mass storage media. Therefore you are probably considering them for storing error correction files.

Nothing is wrong with doing so, but be aware that your data and protective error correction files are kept on media with equal reliability. When you encounter read errors on a data medium it is likely that the medium containing the respective error correction files has also gone defective. After all both media have been written at the same time, and they have the same aging characteristics.

This might come at a surprise, but it can not be guaranteed that an error correction file remains usable when it is stored on a defective medium - here is a explanation of the technical background.

Therefore it is important to protect error correction files just as if they were normal data. To be more specific, the medium containing error correction files must be protected with error correction data as well. Here are two ways of doing this:

1. Storing error correction files on separate media:

   Use additional media just for keeping the error correction files. If you use no more than 80% per medium for error correction files it can be augmented with error correction data. This allows you to recover the medium if you run into problems reading the error correction files at a later time.

2. Storing error correction files on the next medium in sequence:

   Maybe you are using media for an incremental backup strategy. In that case you could collect files until the first medium can be filled. Write that medium as usual and create an error correction file for it. Include that error correction file into the backup set which will go onto the second medium. When the second medium has been written, write the error correction file for it onto the third medium and so on. This way all media in the chain are protected with error correction files (with the ecc file for the last medium residing on hard disk until another medium is written).

Of course Murphys Law may strike and result in all media of the chain becoming defective. In that case you need to recover all media, starting with the most recent one ;-)

2.5.5   Advanced settings for the error correction data
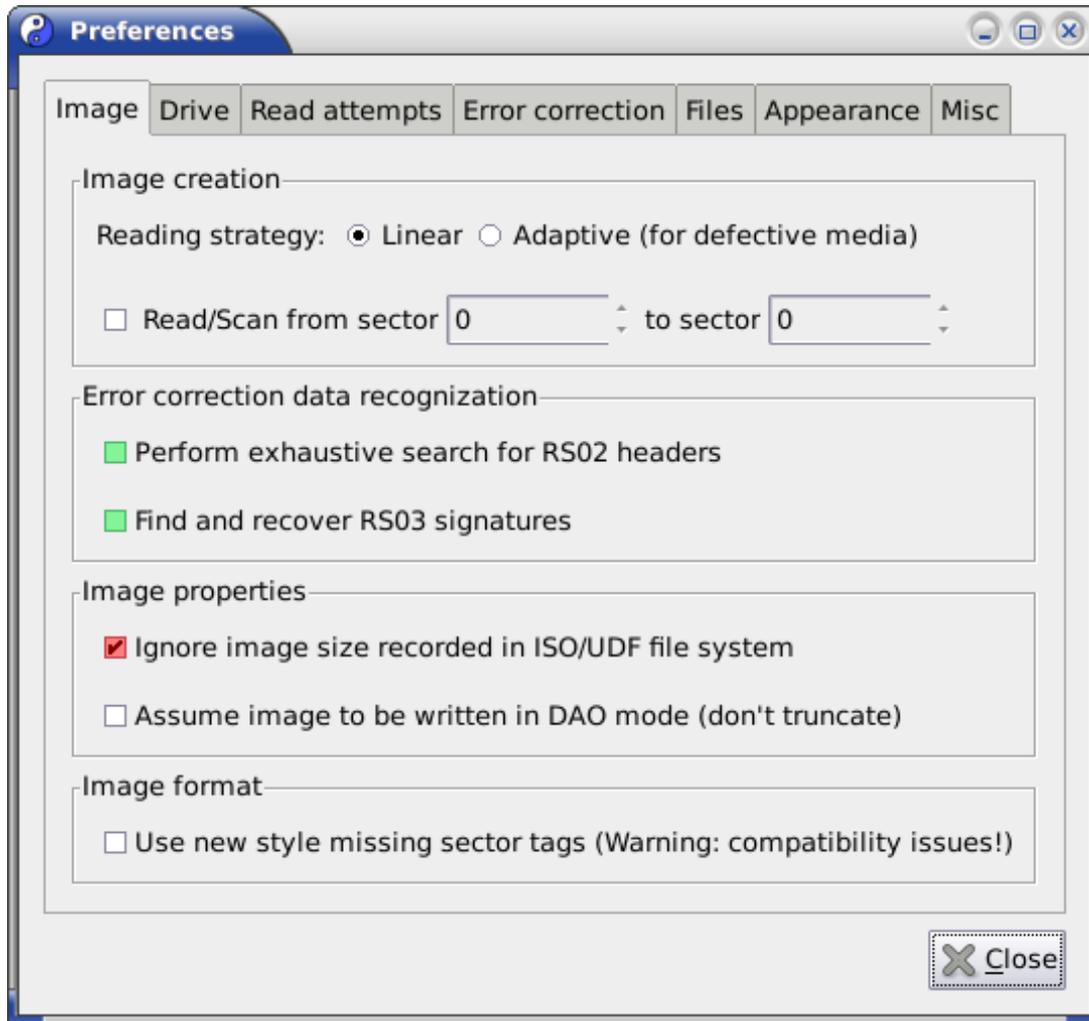
Figure 23: The "Image" tab.

Error correction data recognition.   Since we are reading the image for the purpose of creating error correction data, it makes no sense to look for error correction data in the image itself and the options marked green should remain turned off.

Image properties.   In some cases it might be necessary to change to way dvdisaster determines the size of the ISO image. The normal strategy is to query the ISO image from the ISO/UDF file system, and only if this fails, to query the information directly from the optical drive. Using this order makes sense as image sizes reported by most drives are unreliable in many cases while ISO/UDF file system information is usually correct. However in some rare cases the image size recorded in the ISO/UDF filesystem is wrong. Some GNU/Linux live CDs may have this problem. If you read back the ISO image from such CDs and its md5sum does not match the advertised one from the download site, try re-reading the image with the "Ignore image size recorded in ISO/UDF file system" option (marked red) turned on. But do not blindly turn this option on as it could create sub optimal or corrupted ISO images, especially if you plan to use the image for error correction data generation.
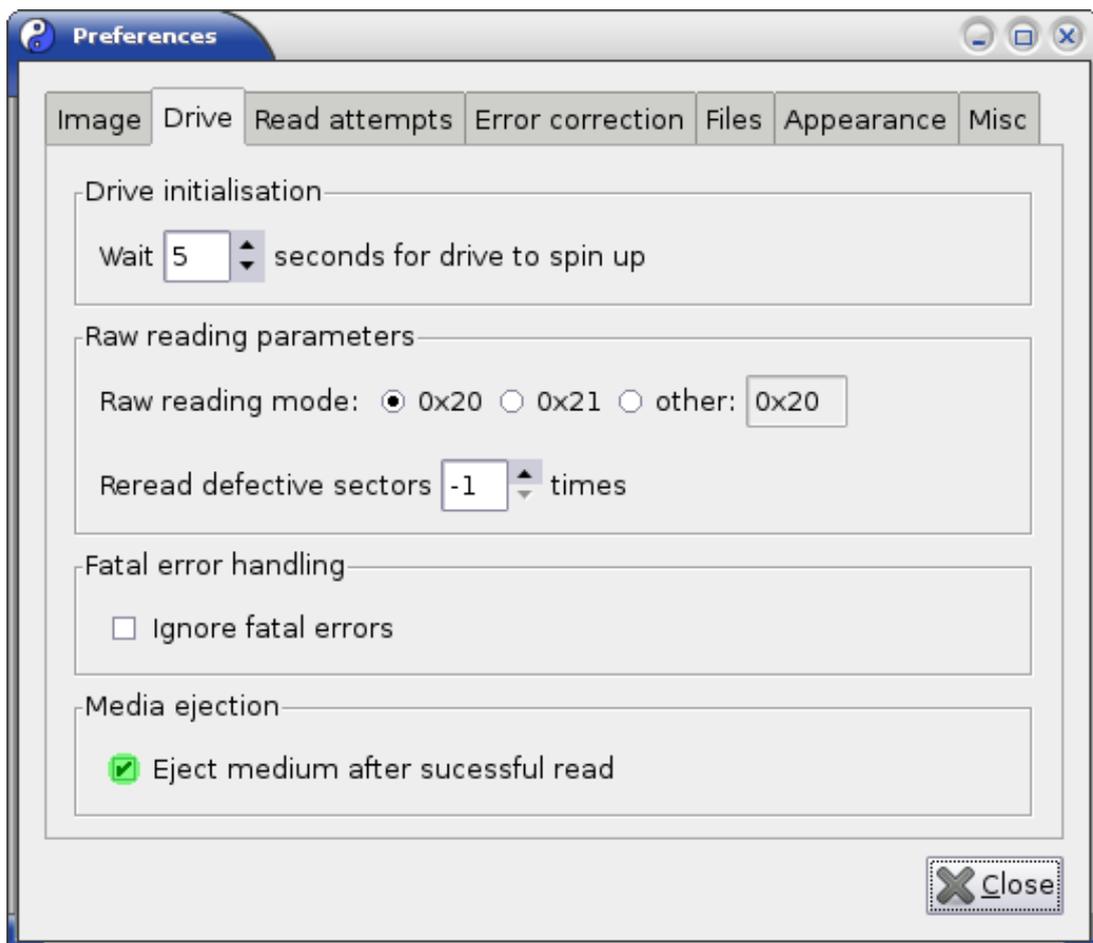
Figure 24: The "Drive" tab.

Media ejection. This feature is helpful when you are processing a batch of media. Use it together with the options shown in the "Files tab" at the next page.

dvdisaster will try to eject the medium after the image has been read. However ejecting the medium might be prohibited by the operating system so this is not guaranteed to work. For example if upon media insertion a window is opened for performing the contents it may not be possible to automatically eject the medium.
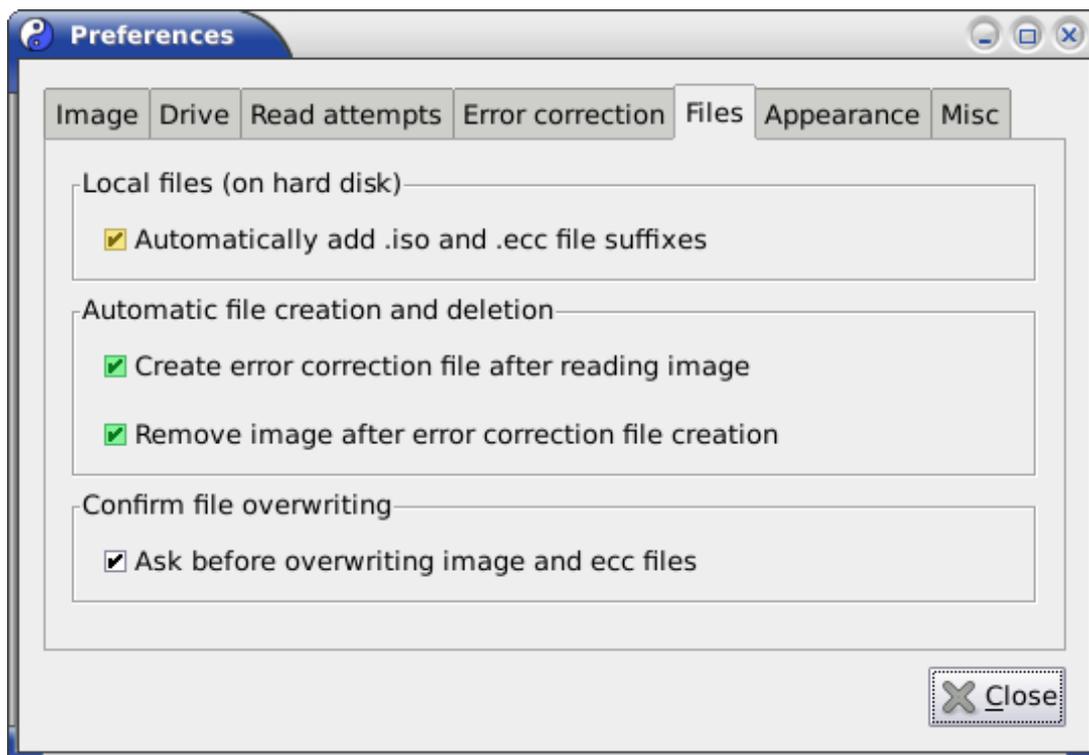
Figure 25: The "Files" tab.

Local files (on hard disk).   To ease entering new file names, activate the option marked yellow. This will append the ".iso" and ".ecc" file name extensions automatically.

Automatic file creation and deletion.   You can automate the process of creating error correction files using these options. The first option marked green lets dvdisaster create the error correction file immediately after the medium has been (completely) read. The second option marked green deletes the image when the error correction file has been successfully created.

Please note: Remember to choose a different name for the error correction file after inserting a new medium. Otherwise the previous error correction file will be overwritten.
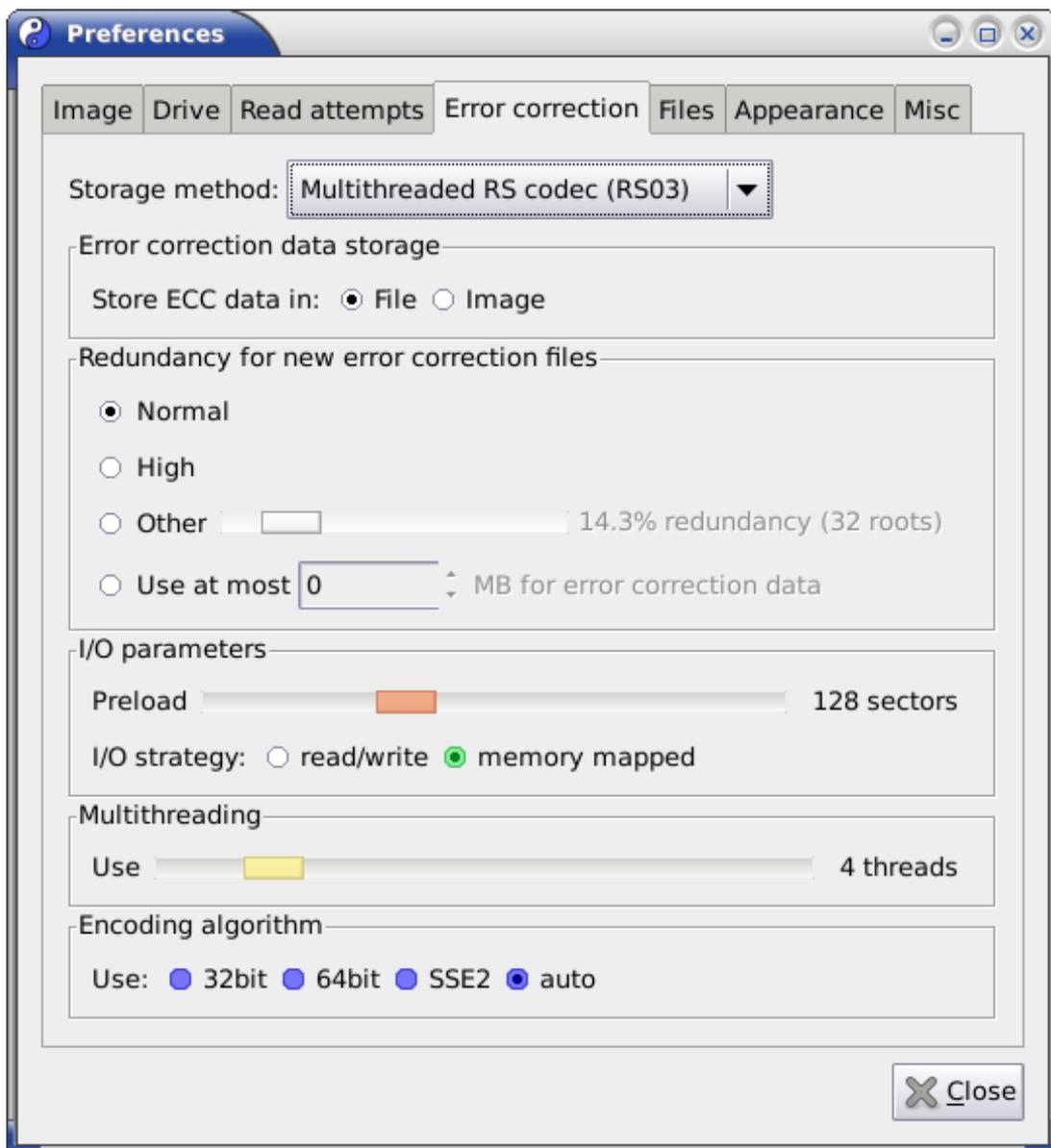
Figure 26: The "Error correction" tab for the RS03 codec.

Optimizing the error correction data creation.
Creating error correction data requires both huge processing power and high random access disk I/O capabilities. When encoding on conventional hard disks with spinning platters, disk I/O will be the limiting factor over all other hardware. Therefore it is recommended to encode on fast storage media like SSDs, or even better, on RAM-based filesystems such as /dev/shm on GNU/Linux.

I/O parameters.   dvdisaster optimizes access to the image and error correction data by preloading and caching parts of them. The optimal preload value (marked red) depends on the storage system used for the image and error correction files. Use small preload values for systems with

low latency and seek time, e.g. SSDs and the RAM file system. For magnetic hard disks performance may be better using larger preload values. Make sure that you do not use more than half of your physical RAM for preloading. A preload value of n will use approx. n MiB of RAM.

The I/O strategy option (marked green) controls how dvdisaster performs its disk I/O while creating error correction data. Try both options and see which performs best on your hardware. The read/write option activates dvdisaster's own I/O scheduler which reads and writes image data using normal file I/O. The advantage of this scheme is that dvdisaster knows exactly which data needs to be cached and preloaded; the disadvantage is that all data needs to be copied between the kernel and dvdisaster's own buffers. Usually, this I/O scheme works best on slow storage with high latency and seek times; e.g. on all storage involving spinning platters.
The memory mapped option uses the kernel's memory mapping scheme for direct access to the image file. This has the advantage of minimal overhead, but may be adversely affected by poor caching and preloading decisions made by the kernel (since the kernel does not know what dvdisaster is going to do with the data). This scheme performs well when encoding in a RAM-based file system (such as /dev/shm on GNU/Linux) and on very fast media with low latency such as SSDs.

Multithreading.   RS03 can use multiple threads - and therefore CPU cores - for encoding (yellow slider). For systems with 4 cores or less, start with setting the number of threads to the number of cores. If you have more cores, leave one core for doing I/O and graphics updates, e.g. use 7 threads on an 8 core system. On systems with hyper-threading capabilities, increasing the number of threads until double the number of physical cores might improve performance.

Do not expect performance to scale linearly with the number of used CPU cores (although it should do so for at least the first four to eight cores). Hard disk performance is more limiting than raw CPU power. When using more than 8 cores, memory bandwidth may eventually limit performance.

Encoding algorithm.   This option affects the speed of generating RS03 error correction data per CPU core (buttons marked blue). dvdisaster can either use a generic encoding algorithm using 32bit or 64bit wide operations running on the integer unit of the processor, or use processor specific extensions. Available extensions are SSE2 for x86 based processors and AltiVec on PowerPC processors. These extensions encode with 128bit wide operations and will usually provide the fastest encoding variant. If "auto" is specified, the SSE2/AltiVec algorithms will be selected if the processor supports them.

2.6   Putting error correction data directly onto the medium

Task   Error correction data is stored along with the user data on the same medium.

Note: This page describes how an ISO image is augmented with error correction data prior to writing it onto a medium. There is also a method for creating and placing error correction data into a separate file.
Would you like help on deciding between these two methods?

Required: • an authoring ("burning") software capable of creating ISO images

• the medium which is to be augmented with error correction data has not yet been written (already written media can not be augmented)

• at least 20% of free space on the medium which is to be created

What to do: 1. Configure basic settings

2a. Create an ISO image,

2b. augment it with error correction data,

2c. and write it to a medium.

2.6.1   Basic settings for augmenting images with ecc data

Figure 27: Opening the configuration dialog.

The relevant tabs are described on the next pages. They are found in the configuration dialog. Open the dialog by selecting the symbol marked green in the screen shot (, see figure 27). The symbol may look different due to the symbol theme you are using.

The image can be augmented with error correction data using either the RS03 or RS02 methods. RS03 is a further development of RS02 and should be preferred for most applications. Especially, RS03 can use multiple threads and will encode much faster on multicore systems than RS02. RS02 is a bit more space efficient than RS03, so it can squeeze out slightly more redundancy of the remaining space than RS03 (typically one additional root for the error correction code). This effect is most pronounced on small media such as CD. RS03 will always fill the medium to the maximum possible redundancy while RS02 allows for user selected redundancies. For media filled with less than 30% of data, RS03 will create a three-fold redundancy using 170 roots which is quite compute intensive. With RS02, a lower redundancy can be selected which is faster to compute. However due to RS03s multithreading capabilities it might keep its performance advantage over RS02 even when encoding with full redundancy.
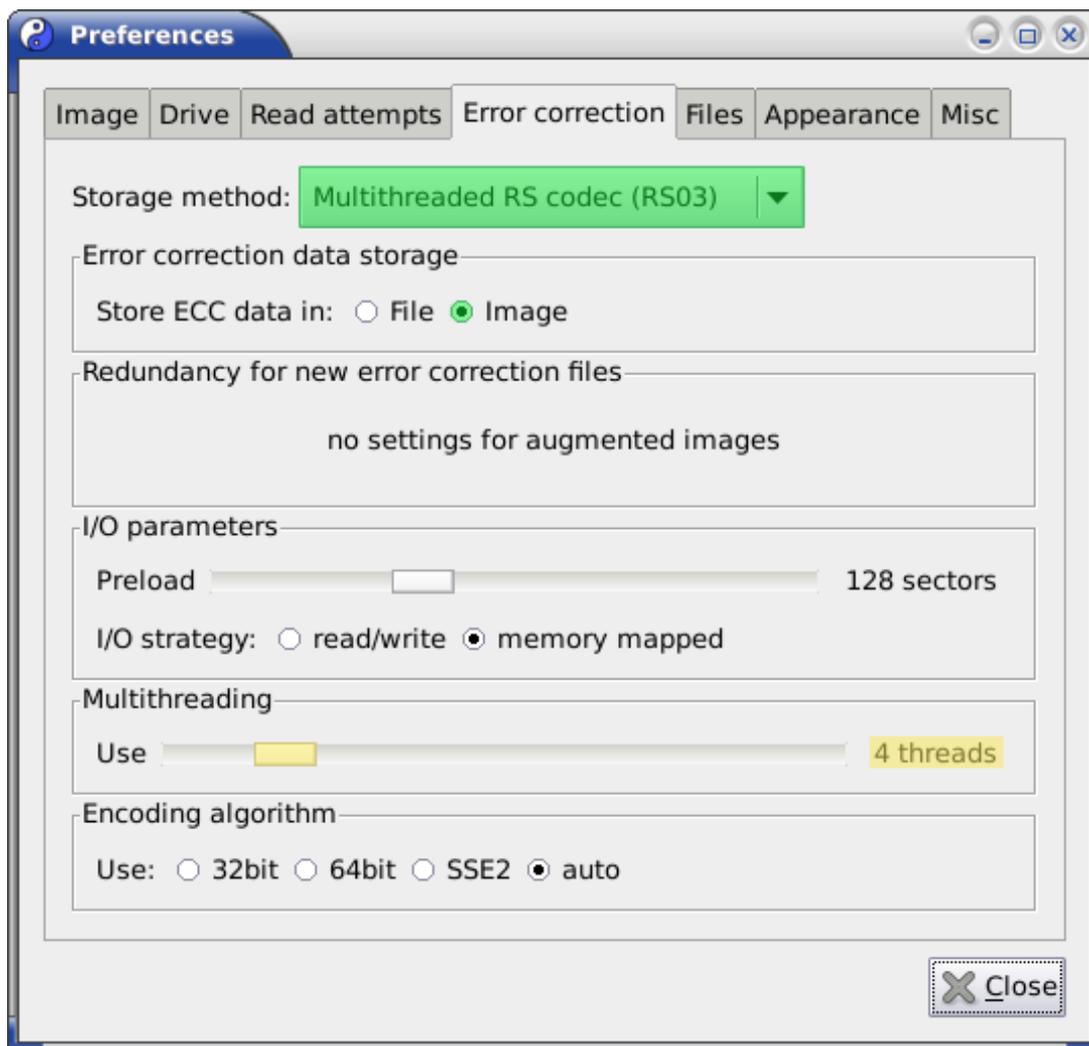
Figure 28: The "Error correction" tab for the RS03 codec.

Settings for encoding with the RS03 method.
Select the entry "Multithreaded RS codec (RS03)" in the drop down menu (marked green). After this selection the contents of the tab will display the choices given with the RS03 method.

Error correction data storage.   Select "Image" (also marked green) so that error correction data will be appended to a given image. The image will always be filled to the maximum possible redundancy so there will be no choices in the "Redundancy" field.

Multithreading.   The RS03 encoder can distribute its workload onto multiple processor cores by using multithreading. On machines with up to four cores, set the number of threads (marked yellow) equal to the number of cores (e.g. use 4 threads on a 4 core machine). On machines with more than 4 cores, use one thread less than the number of available cores; e.g. use 7 threads on an 8 core machine - this leaves one core free for housekeeping tasks.

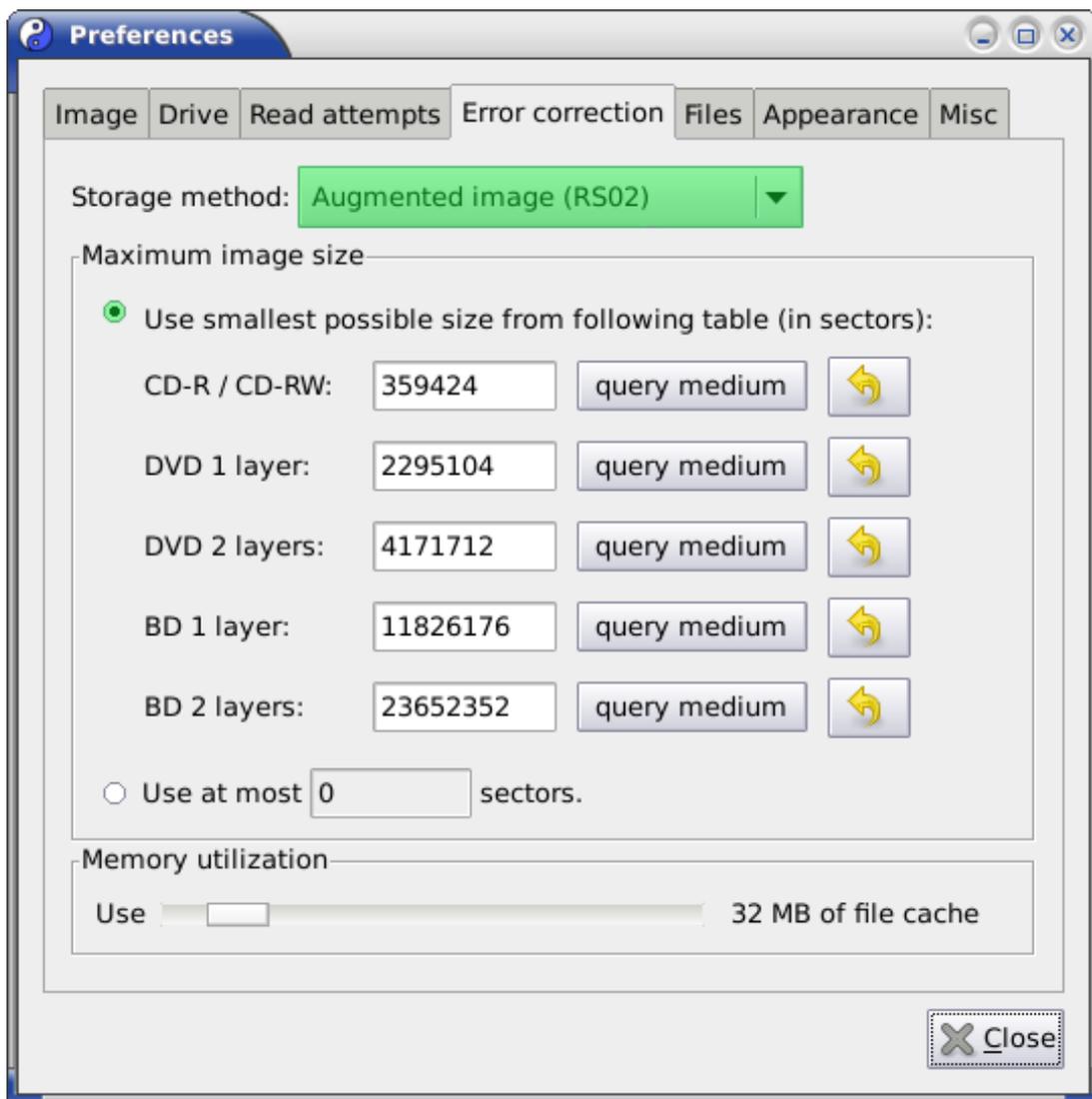Leave the other settings as shown in the screenshot; you can optimize them later.

Figure 29: The "Error correction" tab for the RS02 codec.

Settings for encoding with the RS02 method.
Select the entry "Augmented Image (RS02)" in the drop down menu (marked green). After this selection the contents of the tab will display the choices given with the RS02 method.

Maximum image size.   Select "Use smallest possible size from following table" if you are working with standard media sizes. dvdisaster will then choose the smallest possible medium type which can be used for storing the image. The remaining free space on the medium will be used for error correction data and the image will be prepared accordingly.
See the optimization section for working with non standard media and the remaining settings.

Not used tabs.   The "Misc" tab currently has only functions for creating log files. This is helpful for reporting defects but should be left off during normal operation. The "Appearance" tab allows you to adapt the output colors to your taste, but these have no further effects on the error correction data creation. All other tabs are not relevant for augmenting the image.

2.6.2   Overview: Creating the augmented image and writing it to a medium

dvdisaster is specialized in working with error correction data and reading of defective media. Creating ISO or UDF images and writing them to a medium is a totally different business, and also bears high complexity. We do not want to re-invent medium writing in dvdisaster, as a lot of useful programs have already been written for this task. You should however pick a writing application which supports SAO/DAO (session at once / disc at once) writing on CD media and does not modify ISO images supplied by third-party software (like dvdisaster). Not all burning programs are compatible with dvdisaster, so new programs should be checked before using. For a start we recommend the K3B burning program.

The following overview shows the steps for creating, augmenting and writing the ISO image. More detailed information on these steps is given in the next pages.

First create an ISO image using your optical disc writing software. Select the files you want to write to the medium, but do not start the writing process yet. Instead, create an ISO image on your hard disk. See the walkthrough for creating an ISO image with K3B for more information.

When you have prepared the image **switch over to dvdisaster**. Make sure that it has been configured as described in the basic settings.
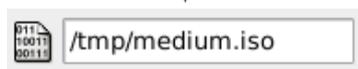
Select the directory and name of the ISO image which you have just created. You can either enter the image file name directly into the text field or bring up a file dialog by selecting the icon to the left of the text field.

Augment the image with error correction data by clicking on the "Create" button.
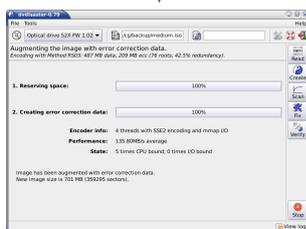
Please wait while the error correction data is being created. This may take a while depending on the image size and the available free space on the medium. Processing a DVD image with about 20-30% free space should take a few seconds on recent hardware.

Please note: dvdisaster does not create a new image, but will rather augment the existing one. If you look at the image in the file manager before and after processing it with dvdisaster you will note how the image size increases.

(continued: next page)

(continued from previous page)
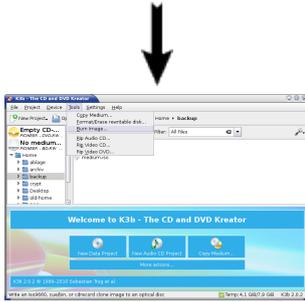
Write the augmented ISO image on the medium. Select the augmented image in your writing software and start the writing process. See the walkthrough for writing an ISO image with K3B for more information.
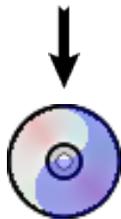
Finished! You have now created an optical disc which is protected by error correction code.

Related information

- Check whether the writing process has affected the error correction data.

  It is recommended to perform this test once every time you change to a new version (or vendor) of your media writing software to make sure that it interoperates well with dvdisaster.

2.6.3   Detailed example: Creating an ISO image on hard disk.

Since there are many different media writing programs available we are demonstrating the required steps by using the free optical disc writing application **K3B** as an example. If you are using a different software you should be able to figure out the required actions from the descriptions below.

Figure 30: Starting a new project.

Begin a new project.   First open your media writing application. Many programs expect you to start a new project. Within the project you will then make the selections for the new medium.

Using K3B: *Begin a new project by clicking the "New Data Project" button located in the lower left area of the main window.*
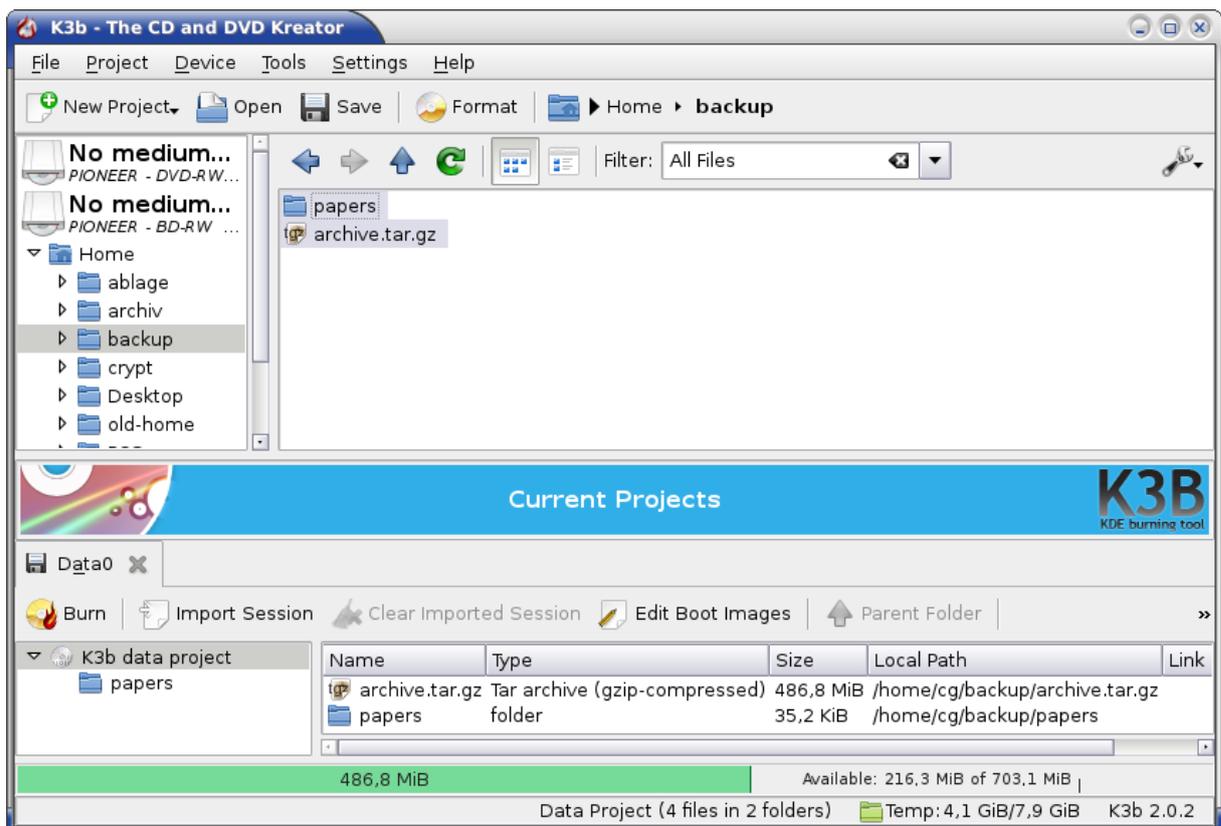
Figure 31: File selection.

Select the files to be written on the medium.   Typically there is a file selection dialog from which you can select files or drag them into the project.

Using K3B: *Use the upper half of the window to navigate your filesystem. Drag the files and folders you want to write to the medium into the "Current projects" area in the bottom half of the window. In the example the folder* papers *and the file* archive.tar.gz *have been selected for writing onto CD.*

Important: Do not completely fill the medium. Make sure to keep at least 20% of the medium space for the error correction data.

Using K3B: *The currently used medium space is shown in the scale bar at the bottom of the window (486,8 MiB).*
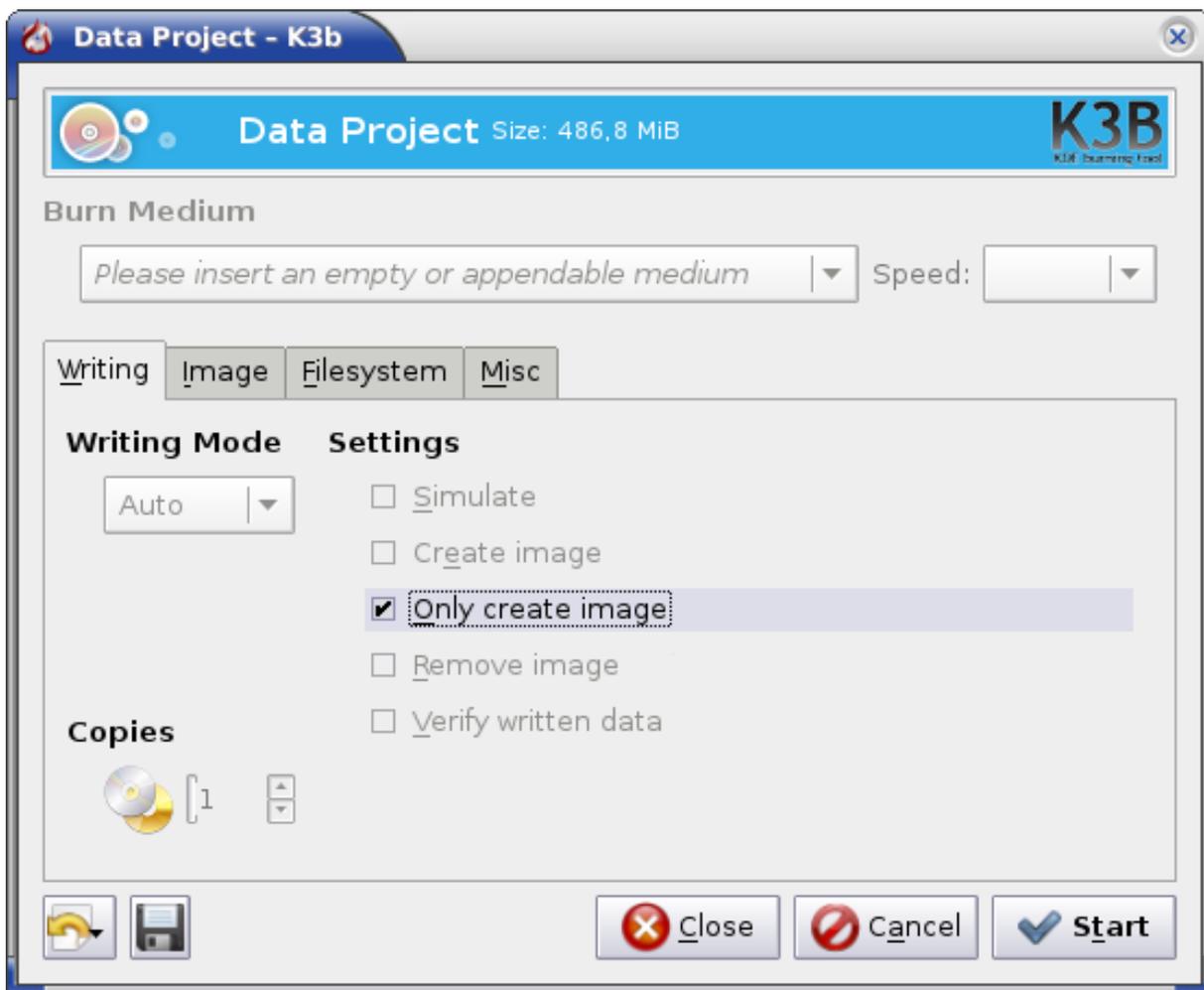
Figure 32: Configuring the writing software.

Configure the writing software.   The software will let you choose the writing target before the actual writing process is invoked. Do **not** select the optical drive or medium here, but configure the creation of an ISO/UDF image on hard disk as described below. If your program seems to be missing an image writing option you might have to select an "image recorder" instead of the actual optical drive.

Hint: Remove all media from the drives before proceeding to make sure that you do not inadvertently start the writing process.

Using K3B: *Click the "Burn" button in the "Current projects" window (see previous screen shot) to open the medium burning dialog. Select "Only create image" from the choices listed under "Settings".*
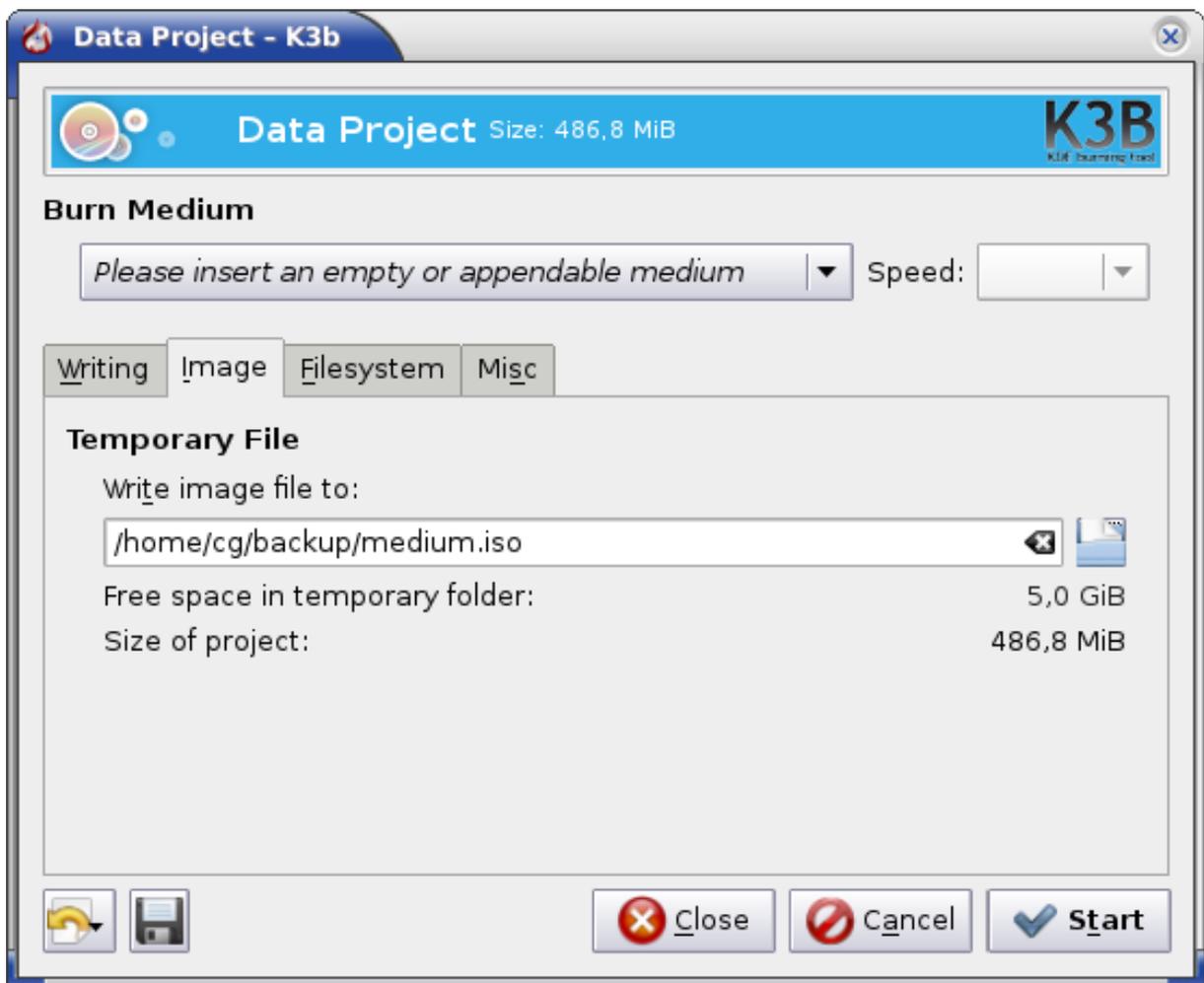
Figure 33: Select image file and type.

Select the image file and type.   Select the target directory, name and type for the image file. Use image files of type ".iso" or ".udf" only! Other image formats like ".nrg" are not supported by dvdisaster; processing such image files with dvdisaster will render them unusable without further notice or error messages.

Using K3B: *Switch to the "Image" tab in the dialog you opened in the previous step. Pick the destination directory and file name in the text field below "Write image file to:", or click the button right to the text field to bring up the file chooser. In the example, the image file will be named "medium.iso" and placed in the "backup" directory in the user cg's home folder. K3B only allows you to create ".iso" type images, so there is nothing to do with respect to the image file type.*
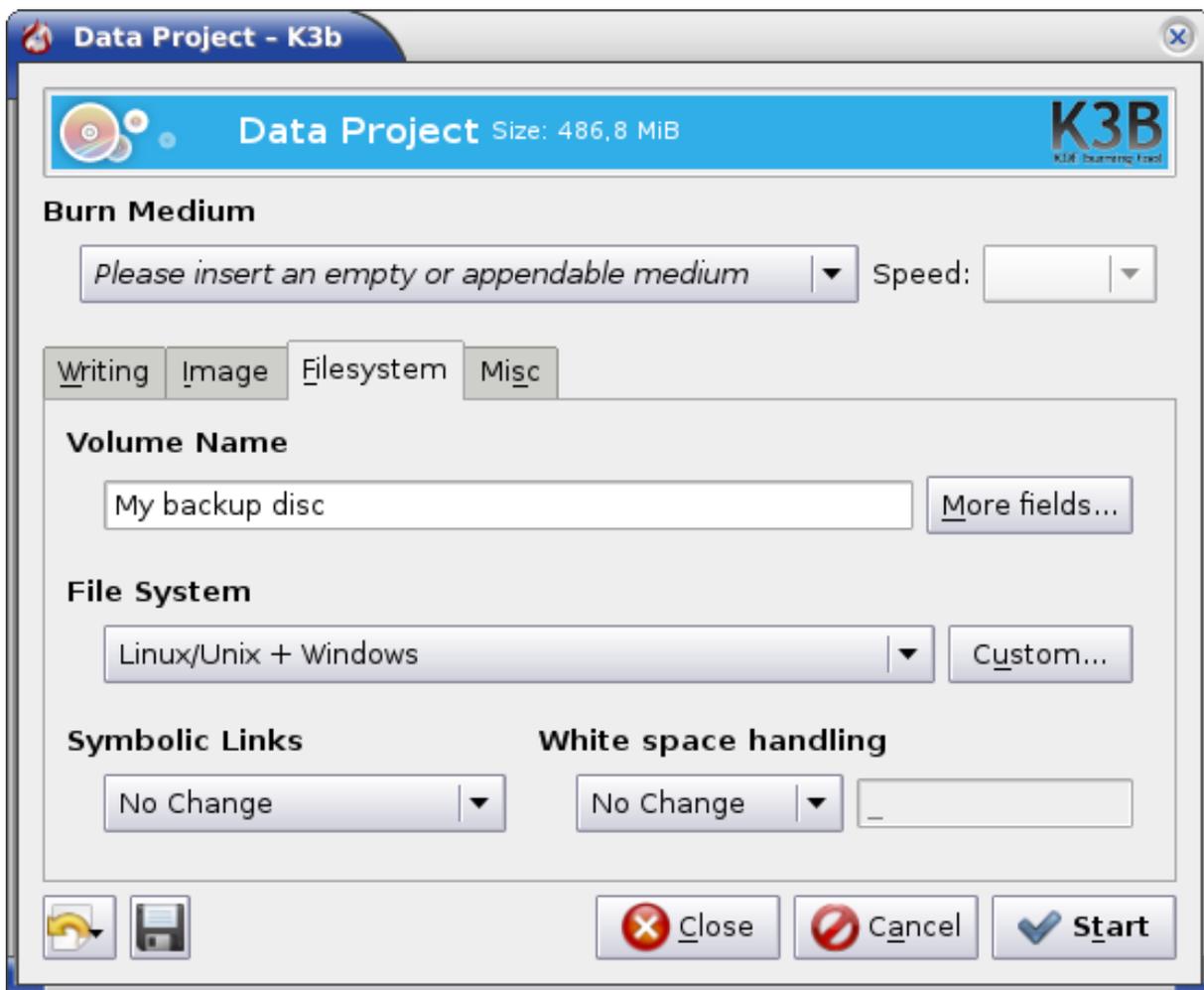
Figure 34: Select the medium name/label.

Select the medium name. Optionally, provide the disc with a descriptive volume name.

Using K3B: *Switch to the "Filesystem" tab in the dialog. Enter the desired name under "Volume Name". In the example, the volume is named "My backup disc".*

Finally, click the "Start" button to create the ISO image.
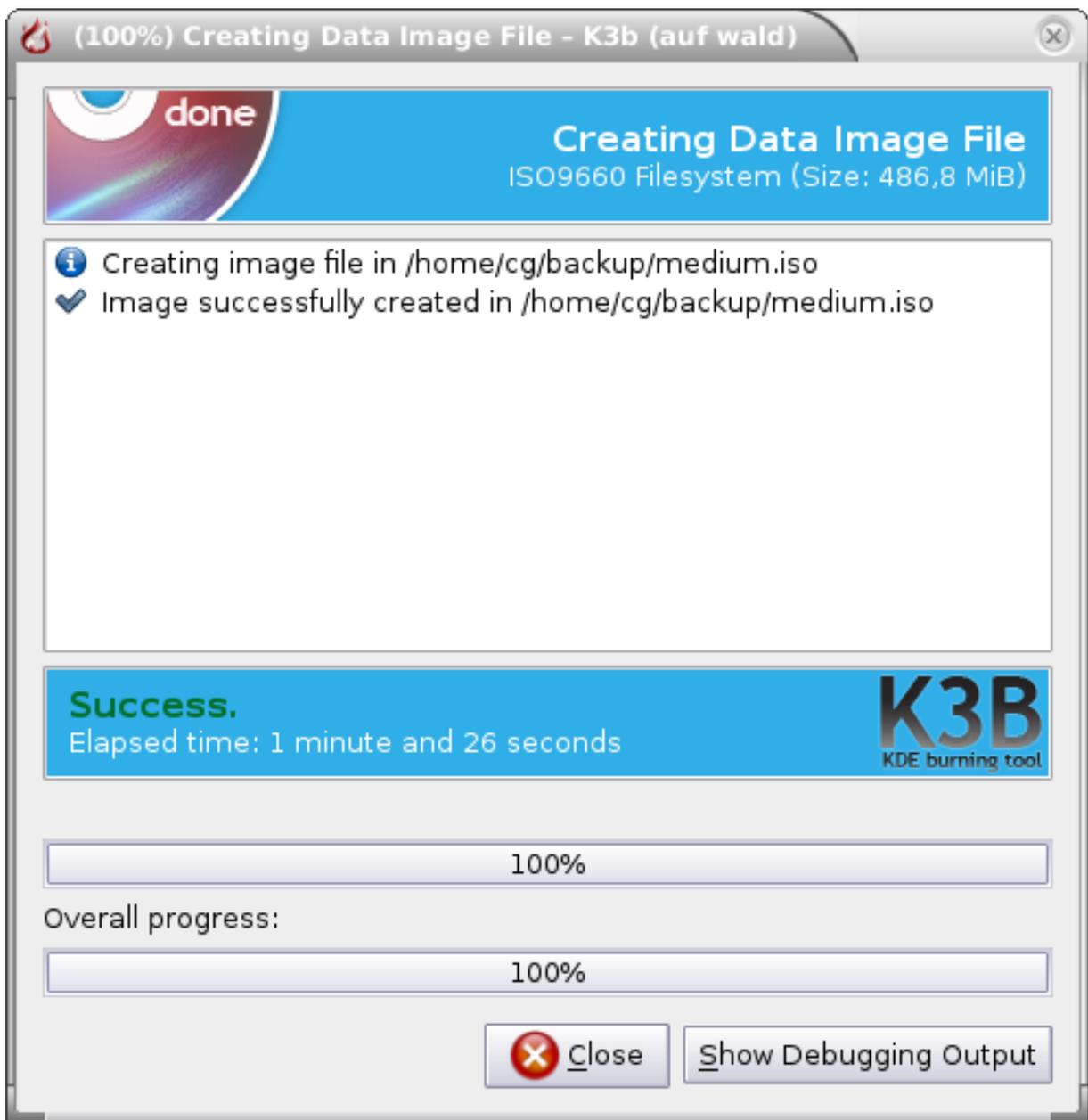
Figure 35: Wait for the image creation to finish.

Wait for the image creation to finish.   It might take a while to produce the image.

When the ISO image has been created, please return to the overview of creating augmented images and continue with the second step ("switch over to dvdisaster").

2.6.4   Detailed example: Writing the augmented ISO image on a medium.

As we have mentioned in the overview section, writing the augmented image onto the medium is not dvdisaster's task. Use your favourite optical disk writing software for performing this step; in the following example we are using **K3B** again.

Figure 36: Switching the application to image writing.

Select writing of the image.   Open your media writing software again. Invoke the mode for writing pre-existing ISO images on the medium.

Using K3B: *From the "Tools" menu, select the "Burn image" entry.*
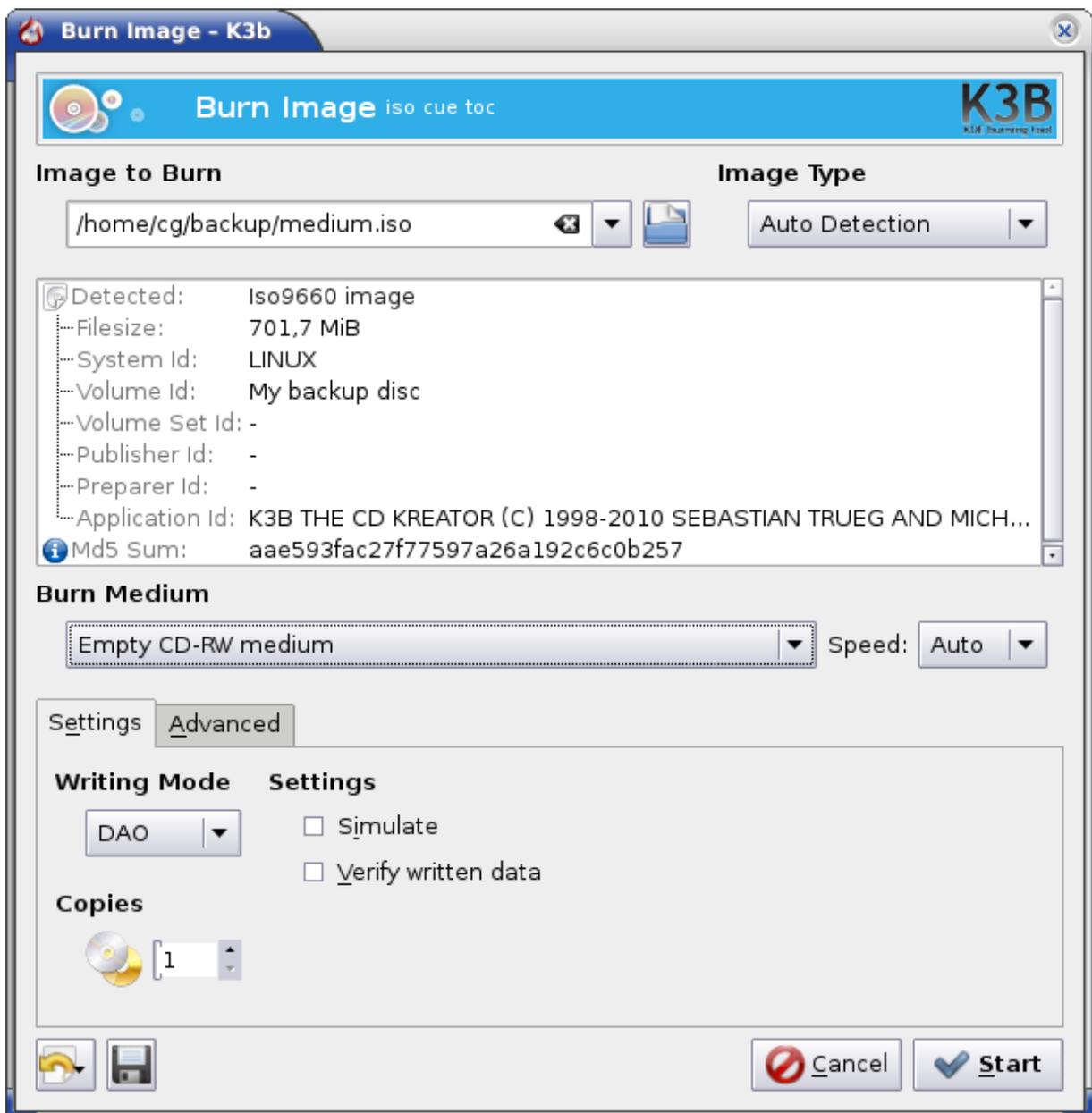
Figure 37: Image selection.

Image selection.   Select the image you have just created and augmented with dvdisaster.

Using K3B: *Fill in the path to the image file in the text field under "Image to burn", or invoke the file chooser by pushing the button to the right of the text field. Select the image file you have just created with dvdisaster.*
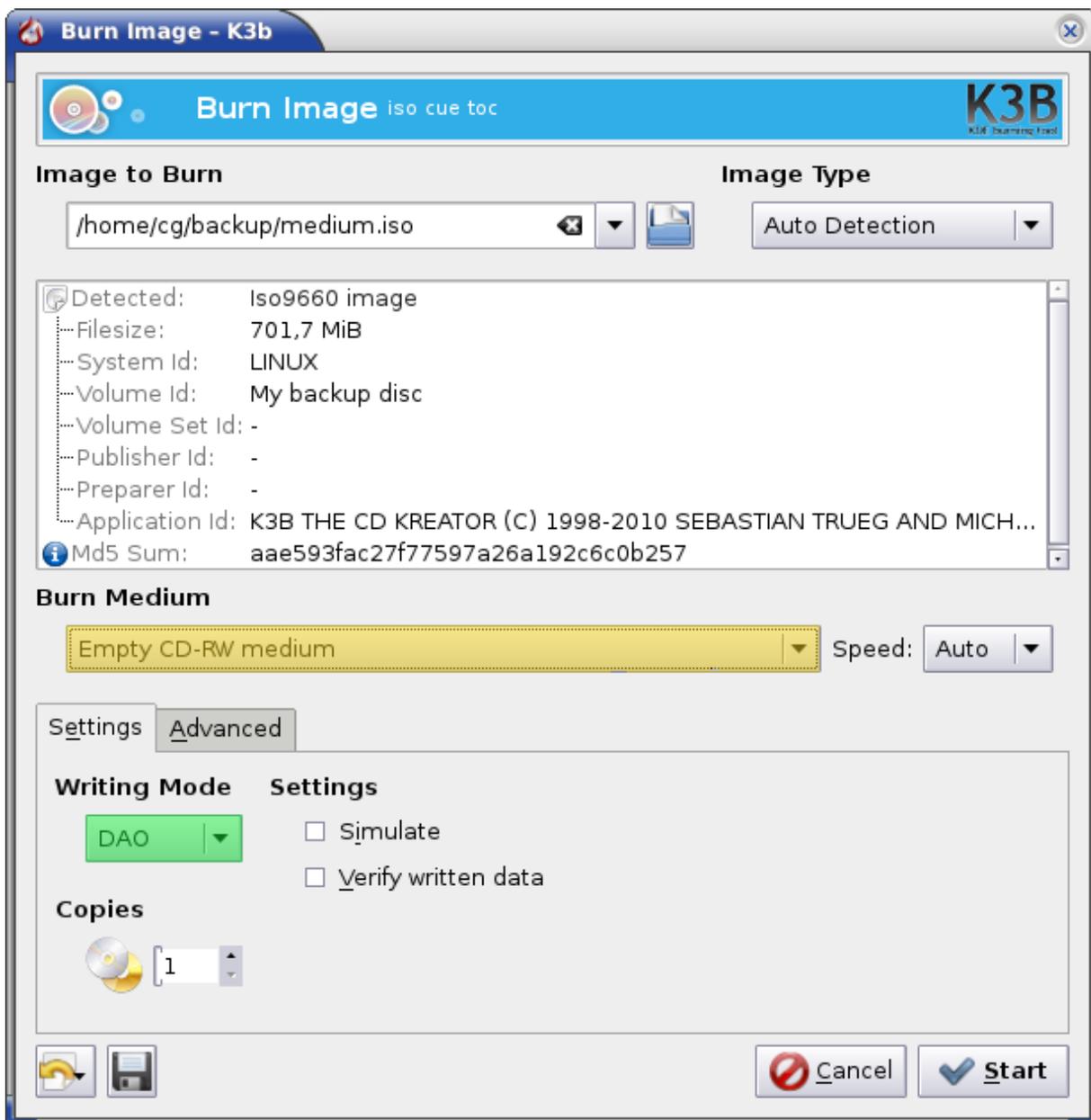
Figure 38: More settings.

More settings.   Select the drive which contains the blank medium you are going to write. When writing CD media, select the "SAO" ("session at once") writing mode if supported by your drive. Sometimes this mode is also called "DAO" ("disc at once"). This improves the compatibility between the medium and the error correction. In addition this prevents you from accidentally adding more sessions to the disc which would destroy the error correction data.

Using K3B: *Select the medium you wish to write in the drop down menu under "Burn Medium" (marked yellow). If you have more than one optical drive, it is recommended to leave all drives empty except for the one you wish to use for burning. Choose "DAO" in the drop down menu under "Writing mode" (marked green) when writing CD media.*
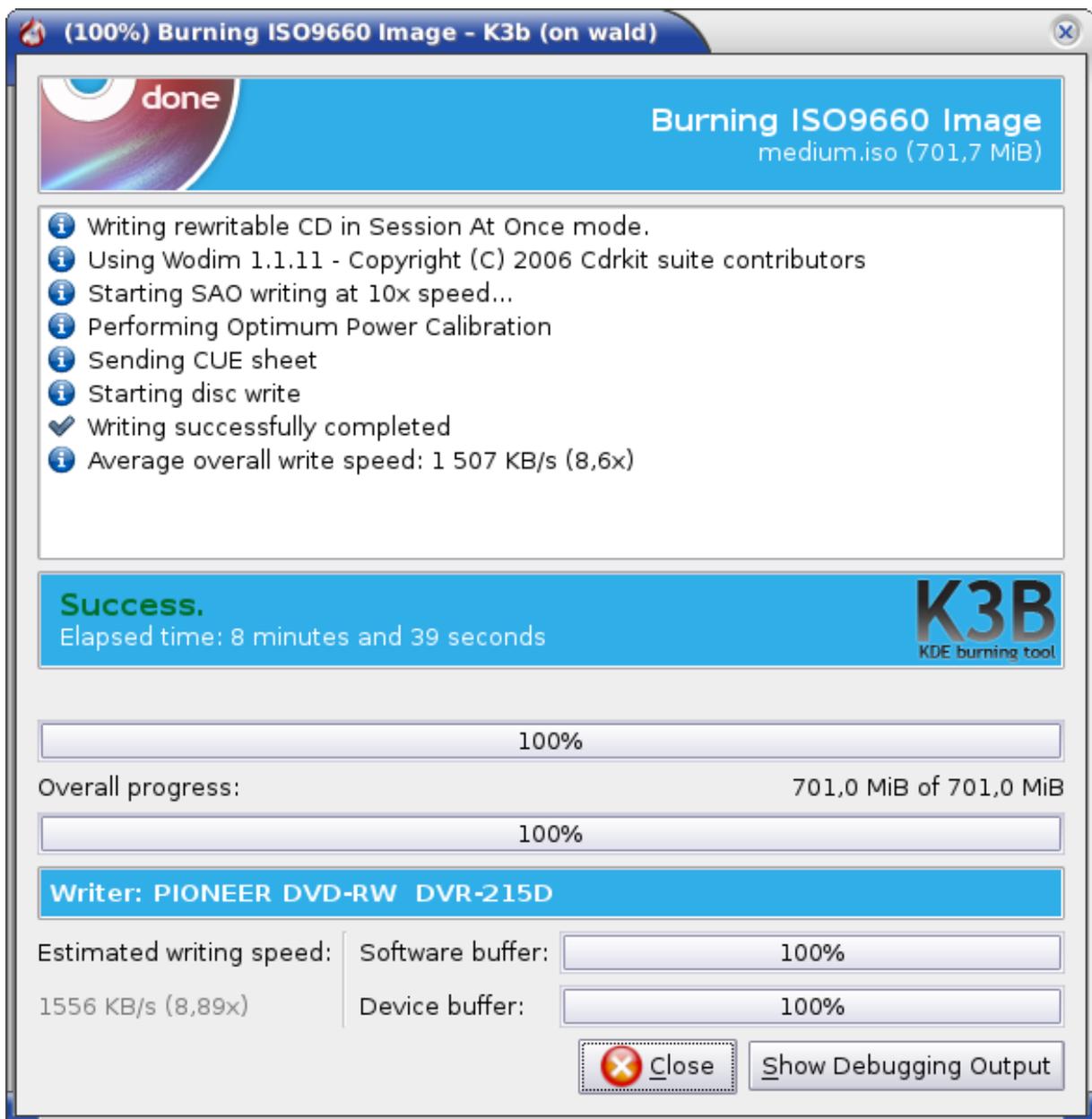
Figure 39: Writing the medium.

Writing the medium. Now start the writing process and wait until it finishes.

Using K3B: *Click on the "Start" button in the window from the previous screen shot.*

After this step, you have finished creating an optical disc which is protected by error correction code.

2.6.5   Advanced settings for augmenting images with ecc data
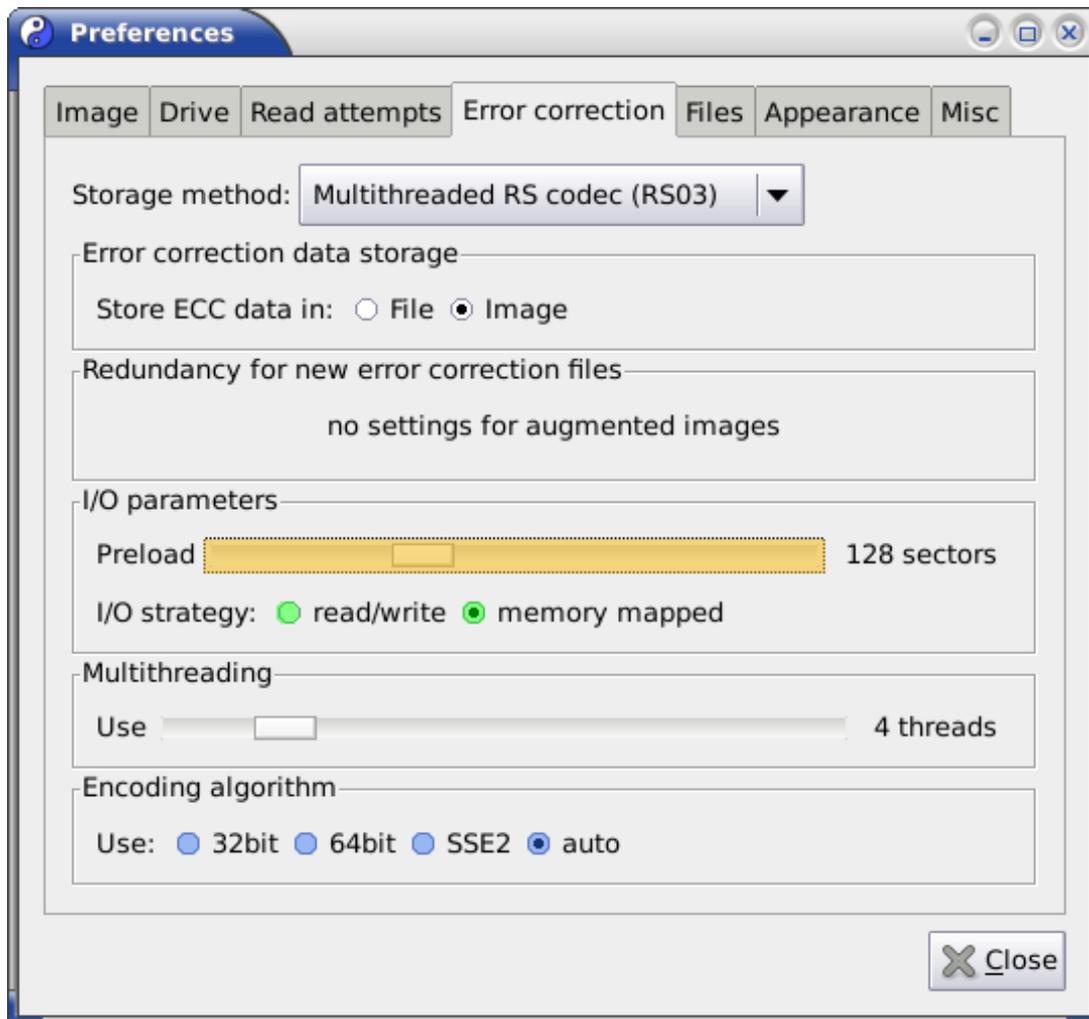
Figure 40: Settings for the RS03 codec.

Settings for encoding with the RS03 method.

I/O parameters.   dvdisaster optimizes access to the image and error correction data by preloading and caching parts of them. The optimal preload value (see options marked yellow) depends on the storage system used for the image and error correction files. Use small preload values for systems with low latency and seek time, e.g. SSDs. For magnetic hard disks performance may be better using larger preload values. A preload value of n will used approx. n MiB of RAM. Do not preload more data than you have physical RAM available.

The *I/O strategy* setting controls how dvdisaster performs its disk I/O while creating error correction data. Try both options and see which performs best on your hardware setting. The read-/write option (marked green) activates dvdisaster's own I/O scheduler which reads and writes image data using normal file I/O. The advantage of this scheme is that dvdisaster knows exactly which data needs to be cached and preloaded; the disadvantage is that all data needs to be

copied between the kernel and dvdisaster's own buffers. Usually, this I/O scheme works best on slow storage with high latency and seek times; e.g. on all storage involving spinning platters. It may also be helpful when the system is under high load; e.g. when a lot of I/O intensive processes are competing for the global I/O buffers (but it is not recommended to use dvdisaster concurrently with other I/O intensive processes anyways). The memory mapped option uses the kernel's memory mapping scheme for direct access to the image file. This has the advantage of minimal overhead, but may be adversely affected by poor caching and preloading decisions made by the kernel (since the kernel does not know what dvdisaster is going to do with the data). This scheme usually performs well when encoding in a RAM-based file system (such as /dev/shm on GNU/Linux) and on very fast media with low latency such as SSDs.

Encoding algorithm.   This option (marked blue) affects the speed of generating RS03 error correction data. dvdisaster can either use a generic encoding algorithm using 32bit or 64bit wide operations running on the integer unit of the processor, or use processor specific extensions. Available extensions are SSE2 for x86 based processors and AltiVec on PowerPC processors. These extensions encode with 128bit wide operations and will usually provide the fastest encoding variant. If "auto" is selected, the SSE2/AltiVec algorithms will be selected if the processor supports them; otherwise the 64bit algorithm will be used. Therefore, the "auto" setting is a good choice unless you are running very uncommon or legacy hardware.
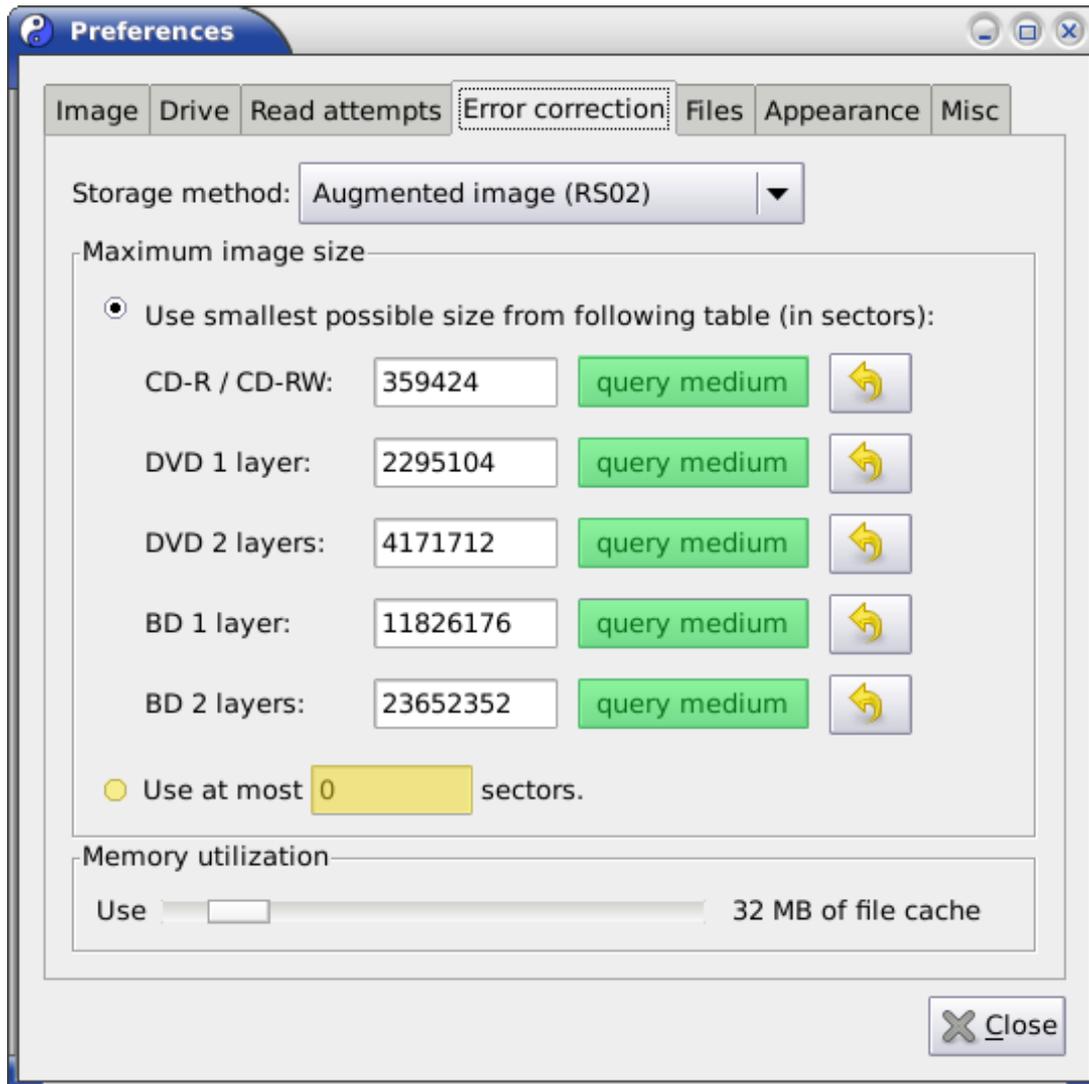
Figure 41: Settings for the RS02 codec.

Settings for encoding with the RS02 method.
Please note that this section is only relevant to RS02. RS03 does not have the respective features.

Choosing the image size.   dvdisaster has a table of standard sizes for CD, DVD and BD media. Any media should meet these size requirements. Some vendors produce slightly higher capacity media. If you have such media, insert a blank one into the currently selected drive and click the "query medium" button (marked green) to the right of the proper medium type. dvdisaster will determine the medium size and update the table accordingly. Alternatively, you can also enter the medium size directly into the respective numerical fields.
Note: The medium size can only be determined in drives which are capable of writing the respective media type.

Arbitrary image sizes.   You can set a specific image size which will not be exceeded after augmenting it with RS02 error correction data. To do so activate the button beneath "Use at most ... sectors" and enter the maximum image size in units of sectors (1 sector = 2KiB).

2.7   Recovering media images

Task   Recover the contents of a defective medium.

Required:

A defective medium containing error correction data,

or

a defective medium with an appropriate error correction file[5].

What to do: 1. Configure basic settings for reading,

2a. create an ISO image from the defective medium,

2b. recover the image and write it to a new medium.

2.7.1   Basic settings for recovering media images

Figure 42: Opening the configuration dialog.

The relevant tabs are described on the next pages. They are found in the configuration dialog. Open the dialog by selecting the symbol marked green in the screen shot (, see figure 42). The symbol may look different due to the symbol theme you are using.

The settings shown here configure dvdisaster for reading the defective medium. There are no dedicated settings for reconstructing the image from the error correction data.

---

[5]The error correction file must have been created at a time the medium was still intact: It is not possible to create error correction data from an already defective medium.
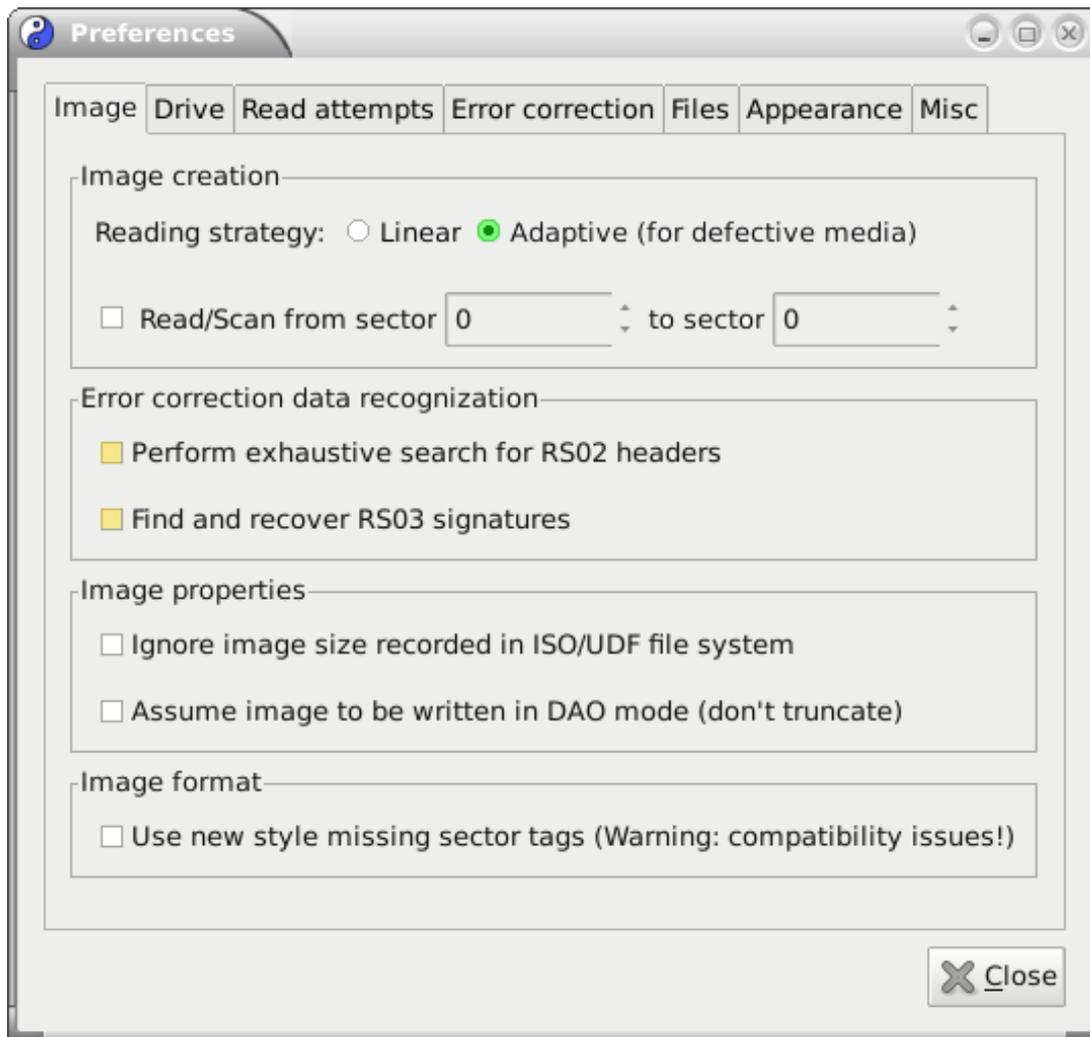
Figure 43: The "Image" tab.

First switch the reading strategy to "Adaptive" (marked green) so that dvdisaster uses information from the error correction data to make the reading process as efficient as possible.

When the defective medium has been augmented with RS02 or RS03 error correction data, activate the respective options for error correction data recognition (marked yellow). Do not activate these options when working with error correction files. Otherwise dvdisaster will search the image for error correction data, which takes a lot of time.

Leave the remaining settings at the values shown in the screen shot.
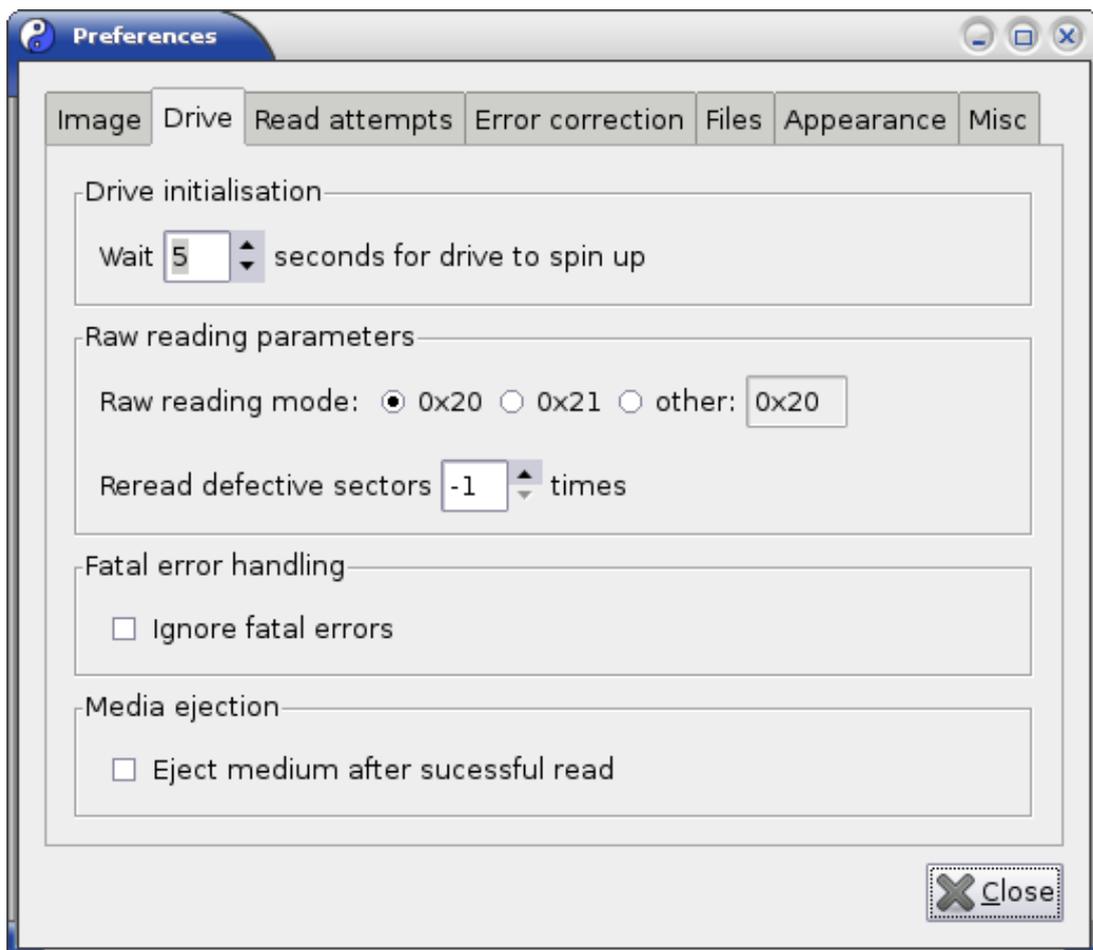
Figure 44: The "Drive" tab.

Leave this tab at the shown default settings for the moment. Some drives might read CD media better using the raw reading mode "21h" (this mode is ignored for DVD and BD media). See the advanced settings for more information.
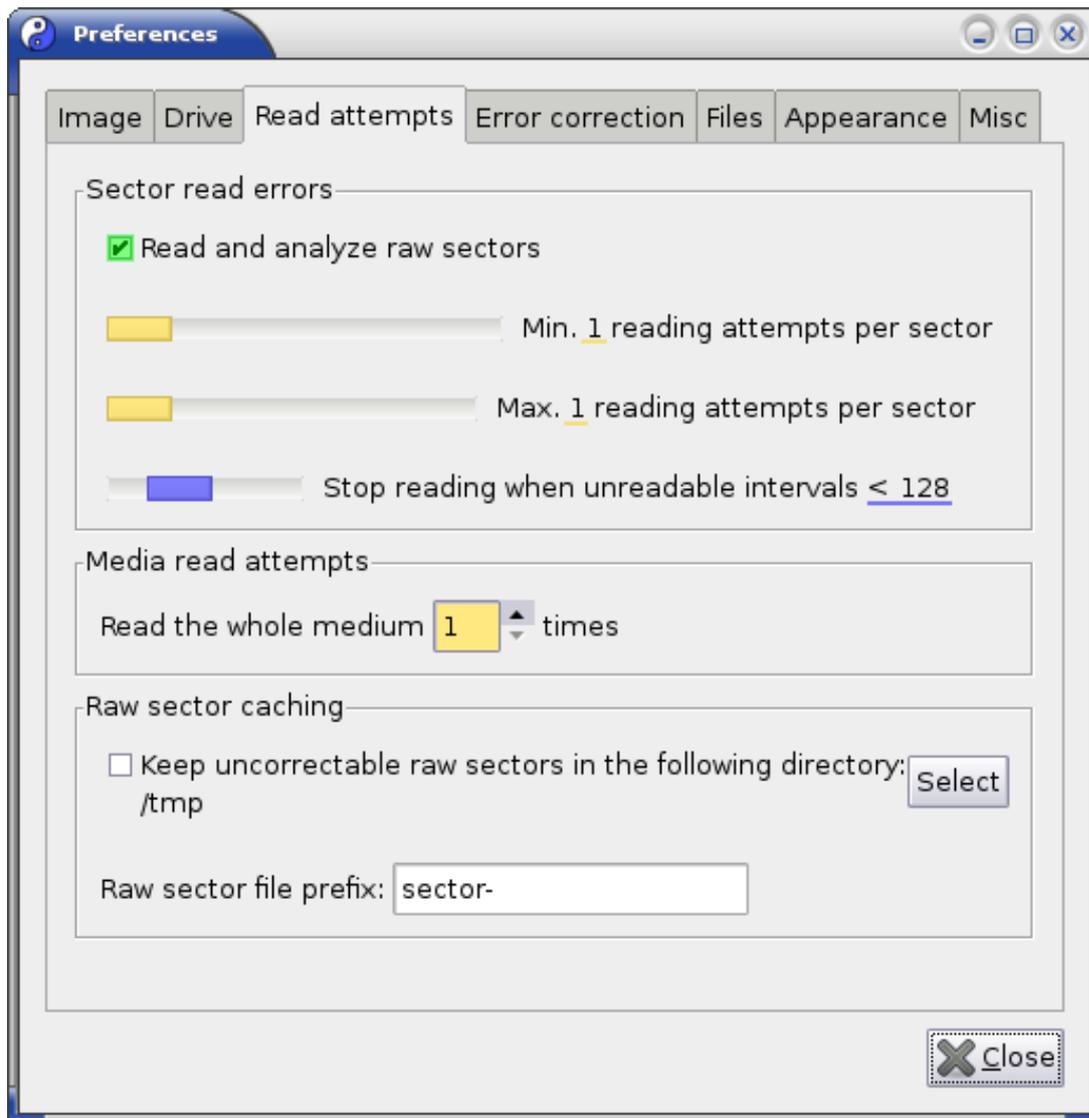
Figure 45: The "Reading attempts" tab.

The strength of the adaptive reading strategy lies in finding the still readable sectors and avoiding the lengthy process of trying to read defective sectors. Therefore select "raw" reading (marked green) as it will not cost additional processing time, but reduce the number of reading attempts to the minimum values (marked yellow). Use a moderate termination criterium of 128 unreadable sectors (marked blue) for the first reading attempt. Do not activate raw sector caching yet. If it turns out that these settings do not provide enough data for a successful recovery they can be optimized later.

Not used tabs. The "Error correction" and "Files" tabs have no influence on the reading process. The "Misc" tab currently has only functions for creating log files. This is helpful for reporting defects but should be left off during normal operation. The "Appearance" tab allows you to adapt the output colors to your taste, but these have no further effects on the reading process.

2.7.2 Recovering media images - Walkthrough

Please make sure that dvdisaster has been configured as described in the basic settings section. Then perform the following steps:

Insert the defective medium into a drive which is directly connected to your computer. You can not use network drives, software drives and drives inside virtual machines.

Close any windows which may be opened by your operating system for viewing or performing the medium contents. Wait until the drive has recognized the medium and the medium has spun down.

Select the drive containing the defective medium in dvdisasters drop down menu.

If you are using error correction files enter the file name in the shown field. Leave this entry blank when the medium has been augmented with error correction data.

Click the "Read" button to start the reading process.

Watch the reading process progress. The adaptive reading strategy performs a systematic search for readable sectors. You will observe temporary gaps which will be closed in later stages. Usually this effect is less pronounced as shown in the screen shot. If all defective sectors are located at the end of the medium the reading process may even stop before touching the first defective sector.

(continued at the next page)

The next actions depend on the outcome of the reading process.   The reading process terminates automatically when enough data for a successful recovery has been gathered (compare the output marked in green). In that case continue the recovery by clicking on the "Fix" button as described two pages later.
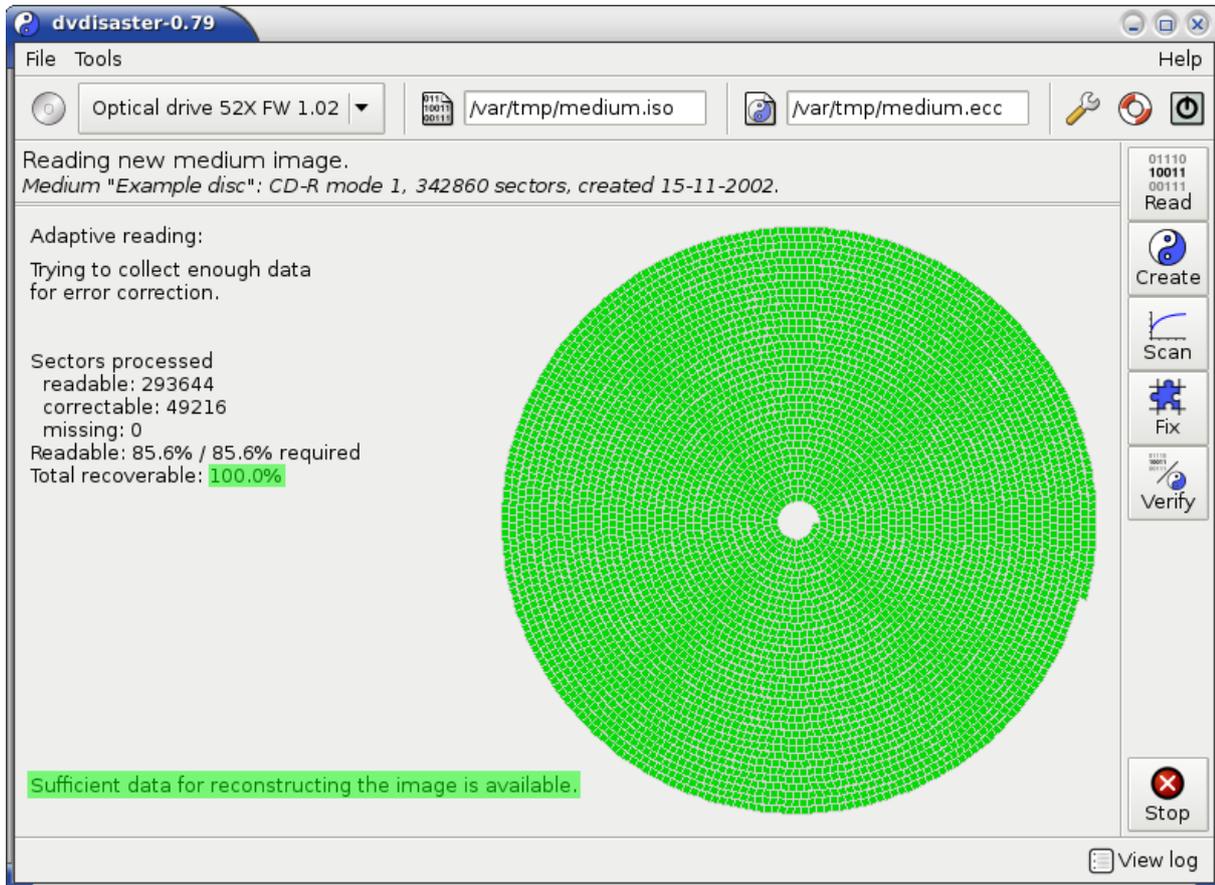
Figure 46: A successful reading attempt.

The reading process will also abort if it could not find enough readable sectors (see the output marked in red). The image is not yet recoverable in this incomplete state. Please try to gather additional data following the tips shown in advanced settings.
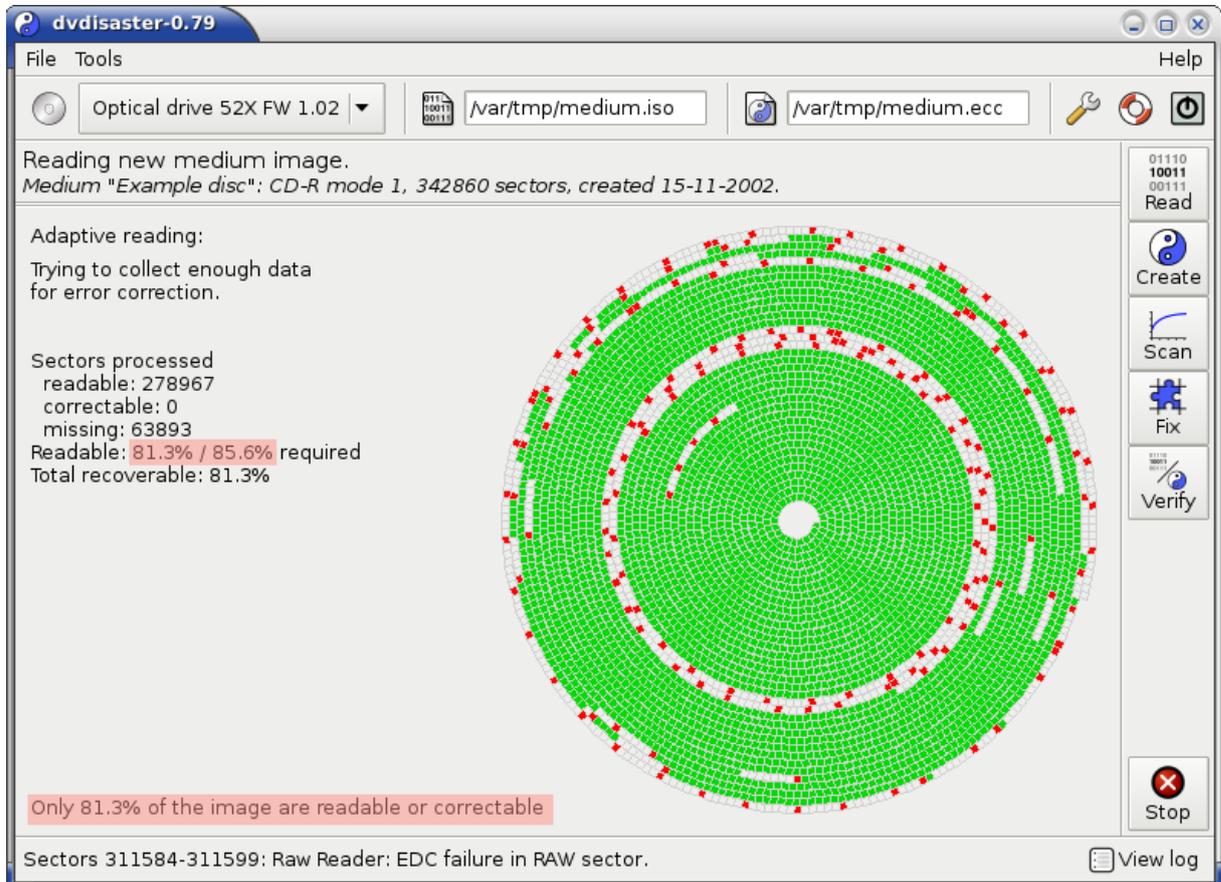
Figure 47: An incomplete reading attempt.

Recovering the image of the defective medium

(continued from previous pages)

Click the "Fix" button to begin the **image recovery**. The recovery will succeed only if the reading process stated success!
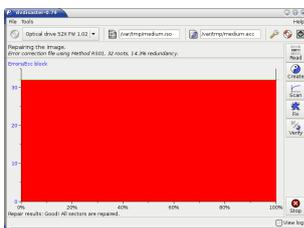
Watch the progress of the recovery. The adaptive reading will stop as soon as enough data has been collected for a successful recovery; therefore the error correction will always be loaded to the max. This causes the display of the massive red area in the "Errors/Ecc block" graph and is no cause for worry. Depending on the medium size and your system speed the recovery may take several minutes to hours.

When the recovery finishes all data in the ISO image will be complete again.

Write the recovered ISO image to a new medium using your favourite media writing software. This step is the same as writing a newly created augmented image to a medium.
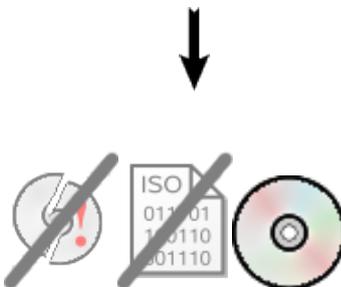
Now you have created a new medium containing the fully recovered data. Make sure to check it for read errors. Then you can discard the defective medium and delete the ISO image. However if you have created an error correction file for the old medium then you can keep it to protect the newly created medium.

2.7.3   Advanced settings for recovering media images

The first attempt of reading the defective medium will usually provide enough data for the error correction. If it did not, try the following.

Estimating the chance of recovery
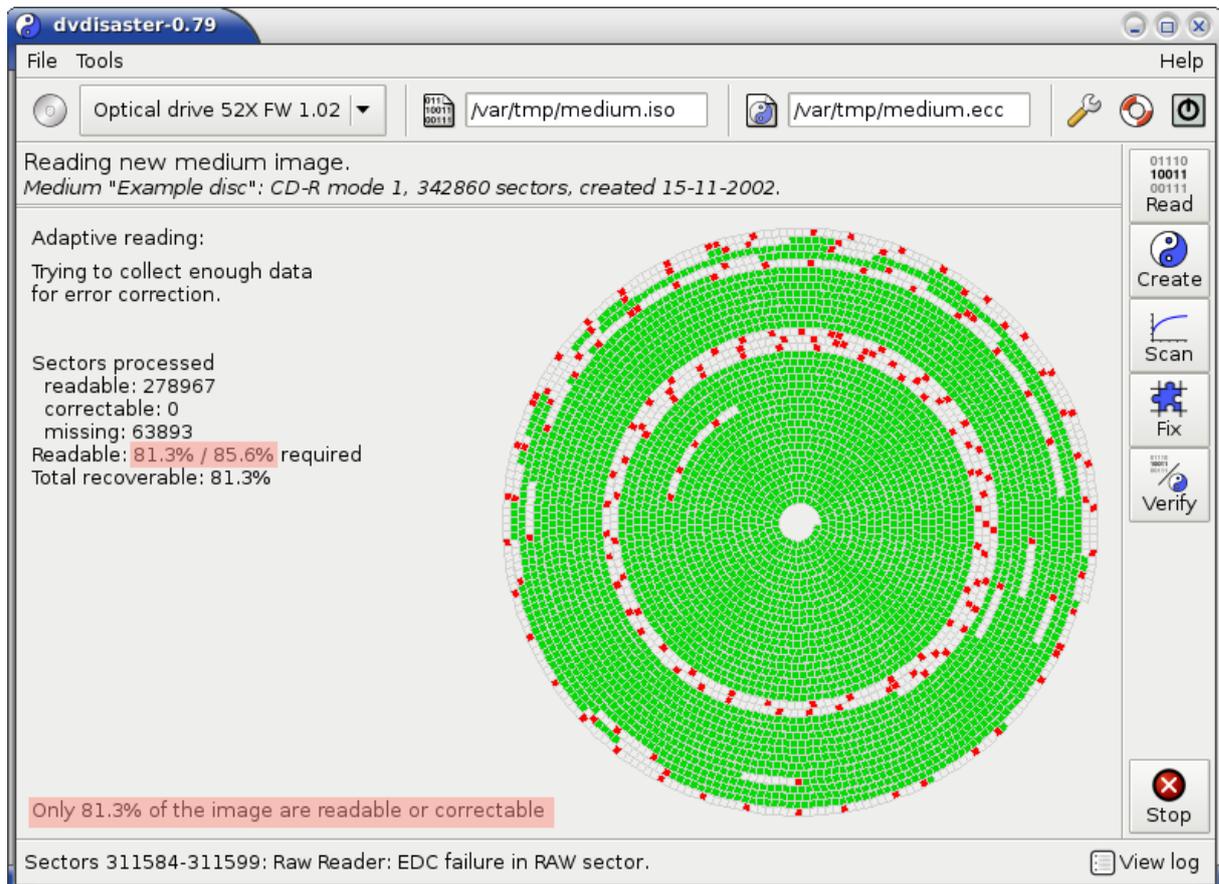
Examine the output of the reading process. Under the "Sectors processed" section you will find the actual percentage of readable sectors and how many percent will be needed for a full recovery. Using the difference between the two values (85.6% - 81.3% = 4.3% in the example) you can estimate the likelyhood of being able to collect enough sectors for a successful recovery. The following table relates the missing percentages to the likelyhood of a successful recovery.

< 5%	Chances are good that you will get enough data using more reading attempts.
5% - 10%	If you have several drives with different reading characteristics you may get the required data by being persistent and patient.
10% - 20%	You are in trouble. If the missing sectors do not drop significantly below 10% during the next 2-3 reading attempts the medium is probably unrecoverable.
10% - 20%	Too much data loss; you can write this medium off as unrecoverable. To prevent this from happening again, use error correction data with higher redundancies and shorten the intervals for defect scanning.

Try the following settings one by one in further read attempts. Please perform a complete reading pass for each setting so that you learn how it affects the outcome (sometimes the results also differ depending on the drive used for reading). When you have gone through the list you may combine them into more powerful configurations.
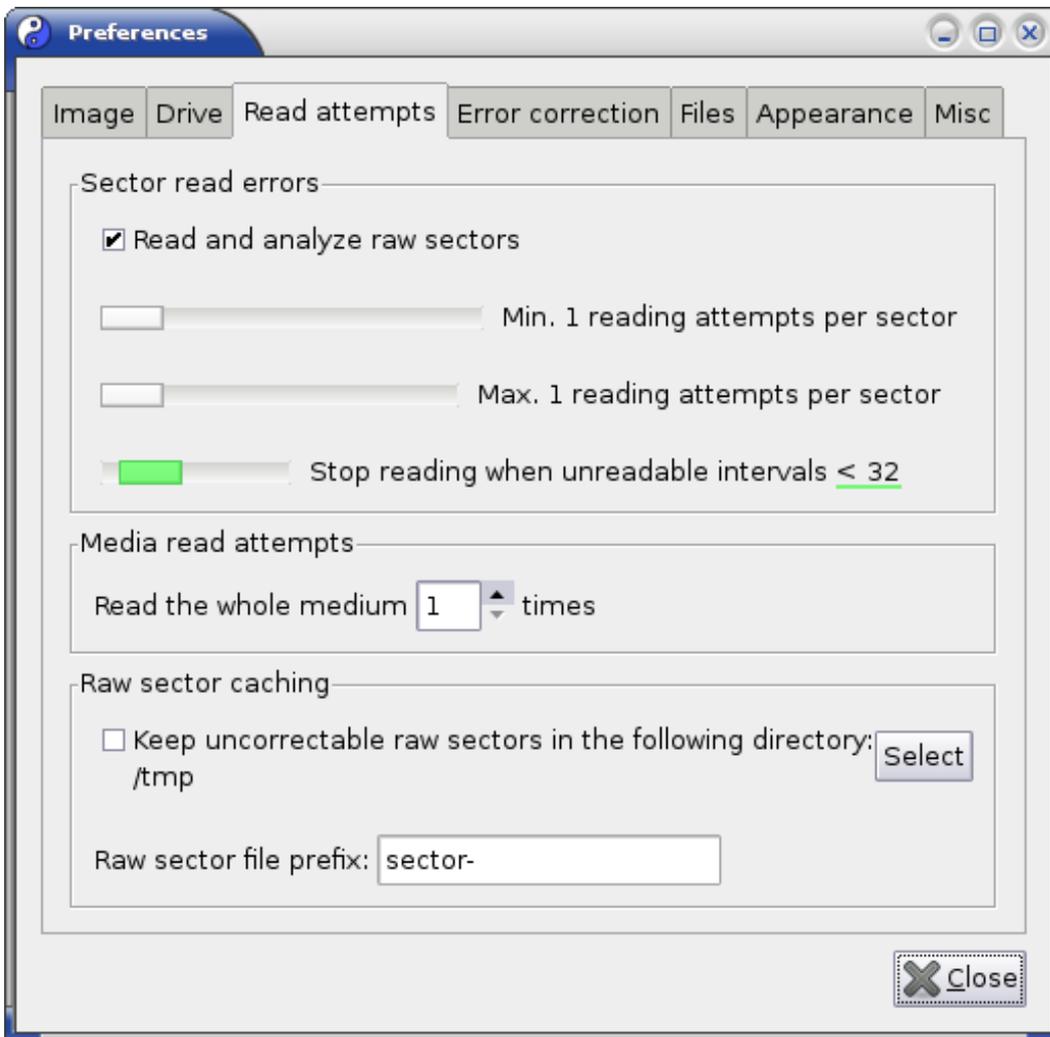
Figure 48: Another reading pass with finer granularity.

Perform another reading pass with a finer granularity
Do not alter any values except for setting a smaller value for terminating the reading process. Recommended values are: 32 for BD, 16 for DVD and 0 for CD (use the slider marked green). Perform another reading attempt using this setting. You can repeatedly read the medium as long as any pass provides a significant number of new sectors.

Hint: Let the drive cool down between the reading passes. Eject and load the medium before each pass; sometimes the medium comes to rest in a better position and the number of readable sectors improves.

Complete the image using different drives
Perform additional reading attempts using different drives. Transfer the image to other computers to see if their drives can contribute more readable sectors.
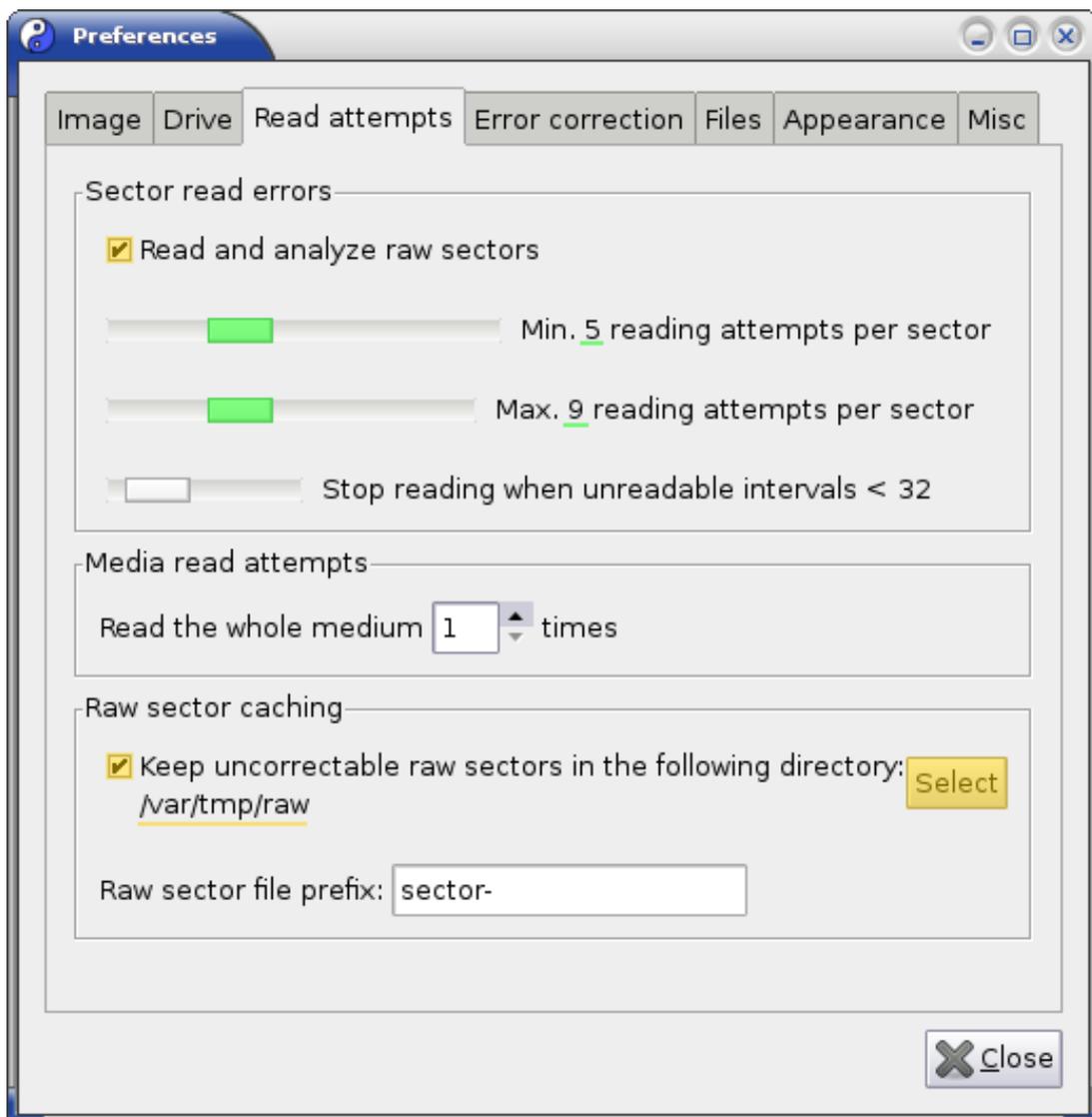
Figure 49: Increasing the number of reading attempts per sector.

Increase the number of reading attempts per defective sector.

For all media types (CD, DVD, BD):

Set the number of reading attempts per sector to a minimum of 5 and a maximum of 9 (green markings).

Only for CD media:

Some drives are capable of partially reading defective sectors on CD media. Activate the "Raw sector caching" option and specify a directory where fragments of defective sectors should be stored (yellow markings). If enough fragments of a defective sector have been collected it may be possible to fully reconstruct it from that information.

See the hints on the next page to check that these settings really have an effect.

Making sure that the new settings work:
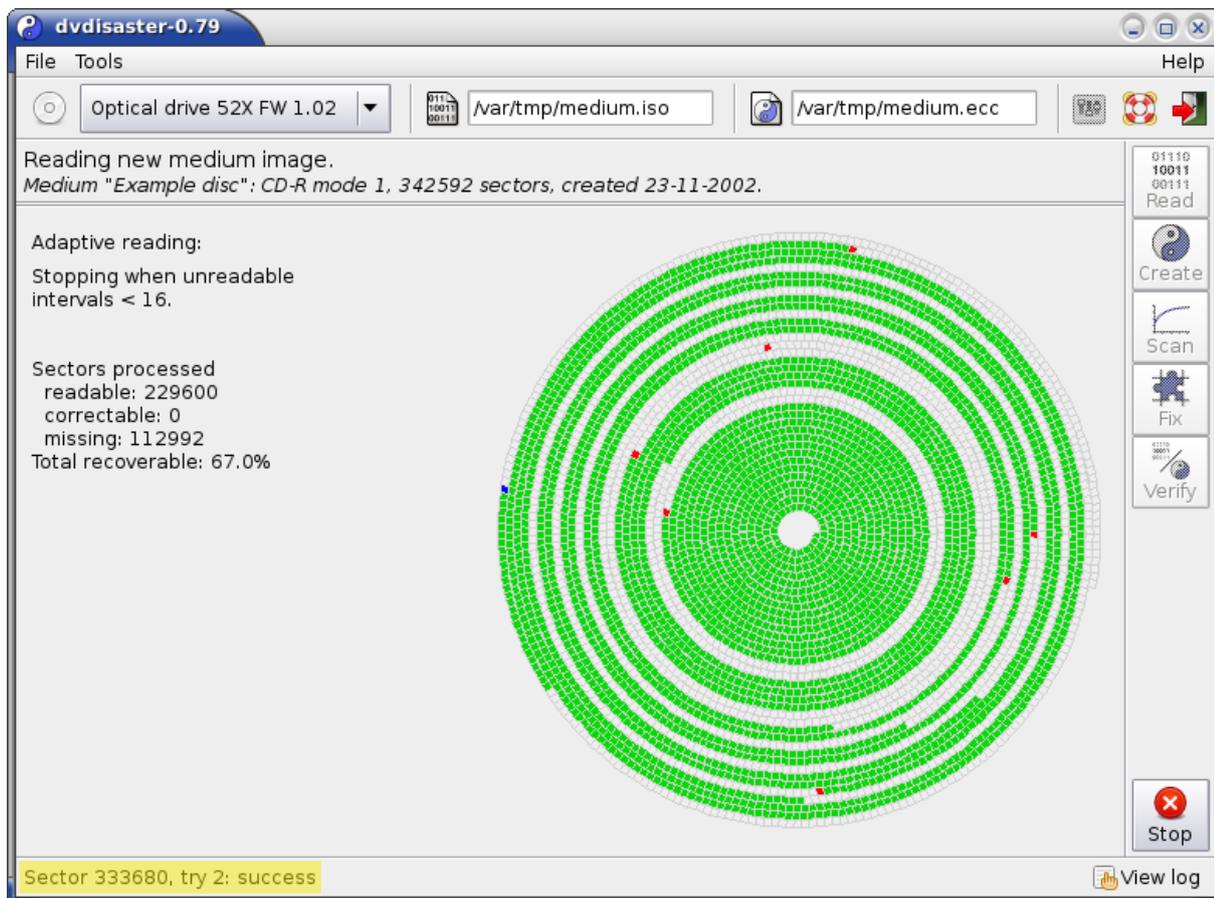
Figure 50: Multiple reading attempts.

Examining results of multiple reading attempts (CD, DVD, BD):
Not all drives show an improvement after increasing the number of reading attempts. Watch for messages of the form "Sector ..., try x: success" (highlighted in yellow). These indicate that the drive could read a sector after several reading attempts. If you never see such messages, increasing the number of reading attempts does not pay off for the respective drive.

Examining partial reading of defective CD sectors:
When the whole medium has been processed, look into the directory you entered above (/var/tmp/raw in the example). If no raw files have been created the drive may not support the required reading mode. However if you have several drives which do create raw files, then let them all work in the same raw file directory. Collecting raw sector fragments from different drives hightens the chance of reconstructing the defective sectors.
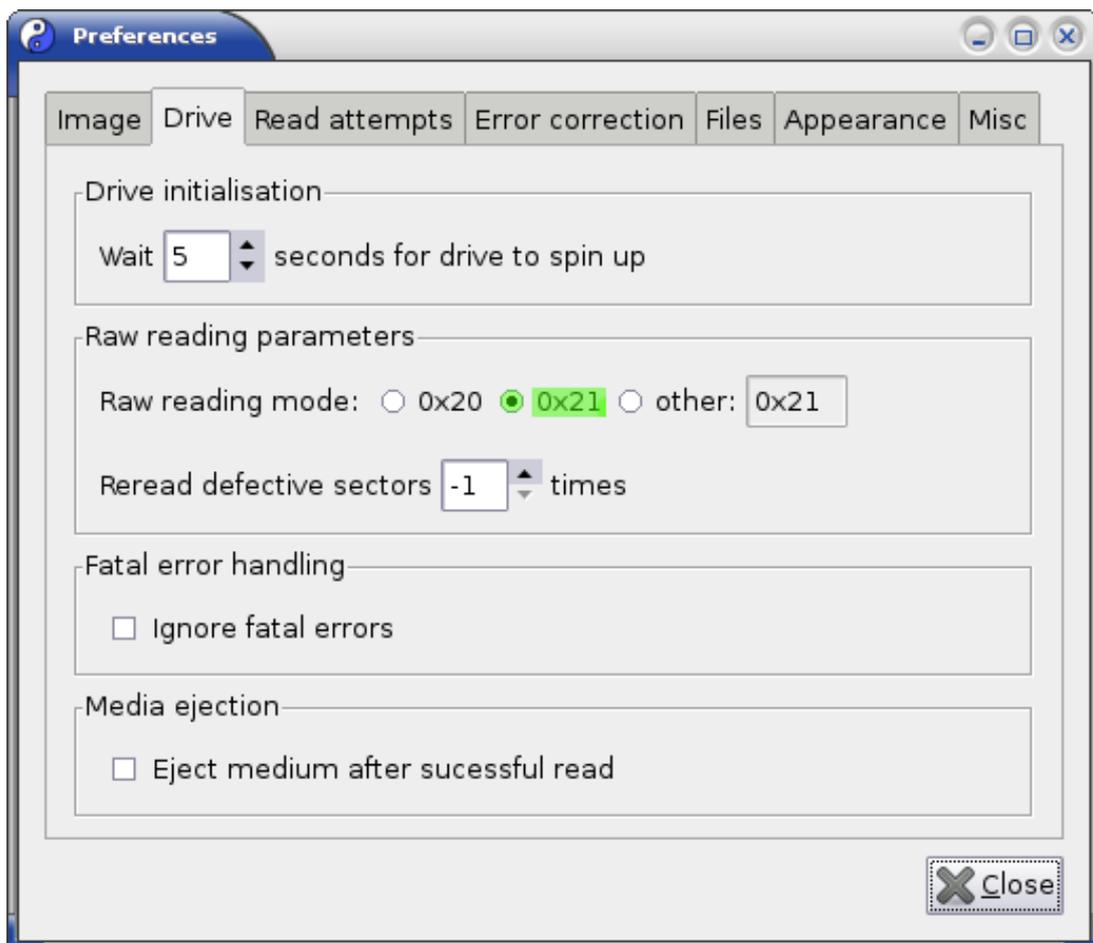
Figure 51: Use a different raw reading mode.

Use a different raw reading mode for CD media:
Using the preset "20h" raw reading mode might not work on some drives. Perform another reading attempt using raw reading mode "21h" (see the screenshot). Check again whether some raw files have been created.

2.8   Getting information on images and error correction data

Task   Shows information on types and states of images and error correction files.

Required:

An image file (either fully working or defective) and optionally the error correction file for it.

What to do: 1. Show the information

2. Interpret the results for RS01, RS02, or RS03.

2.8.1   Showing the information

The are no settings for this function; however you need an image file and optionally the error correction file belonging to it.
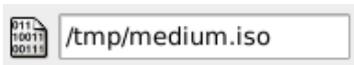
Enter the file name of the ISO image for which you want to get information. The image must already be present on hard disk; otherwise use the "Read" function to get it from a medium.

Enter the name of the error correction file which belongs to this medium. Leave this entry blank when the image has been augmented with error correction data.

Start the evaluation by clicking on the "Verify" button.
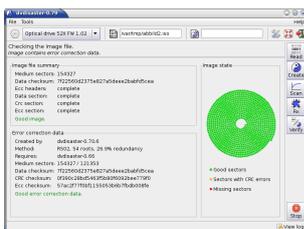
Watch the verification progress. In order to display all information the image and error correction files must be fully read.

The verification process produces slightly different output depending on the the type of error correction data used:

- whether error correction files or augmented images have been used; and

- which error correction method (e.g. RS01, RS02 or RS03) has been used.

See the following subsections for interpreting the specific outputs for:

- error correction files created by the RS01 method; or

- images augmented with error correction data by the RS02 method; or

- any error correction data created by the RS03 method.

2.8.2   Results for images with RS01 error correction files
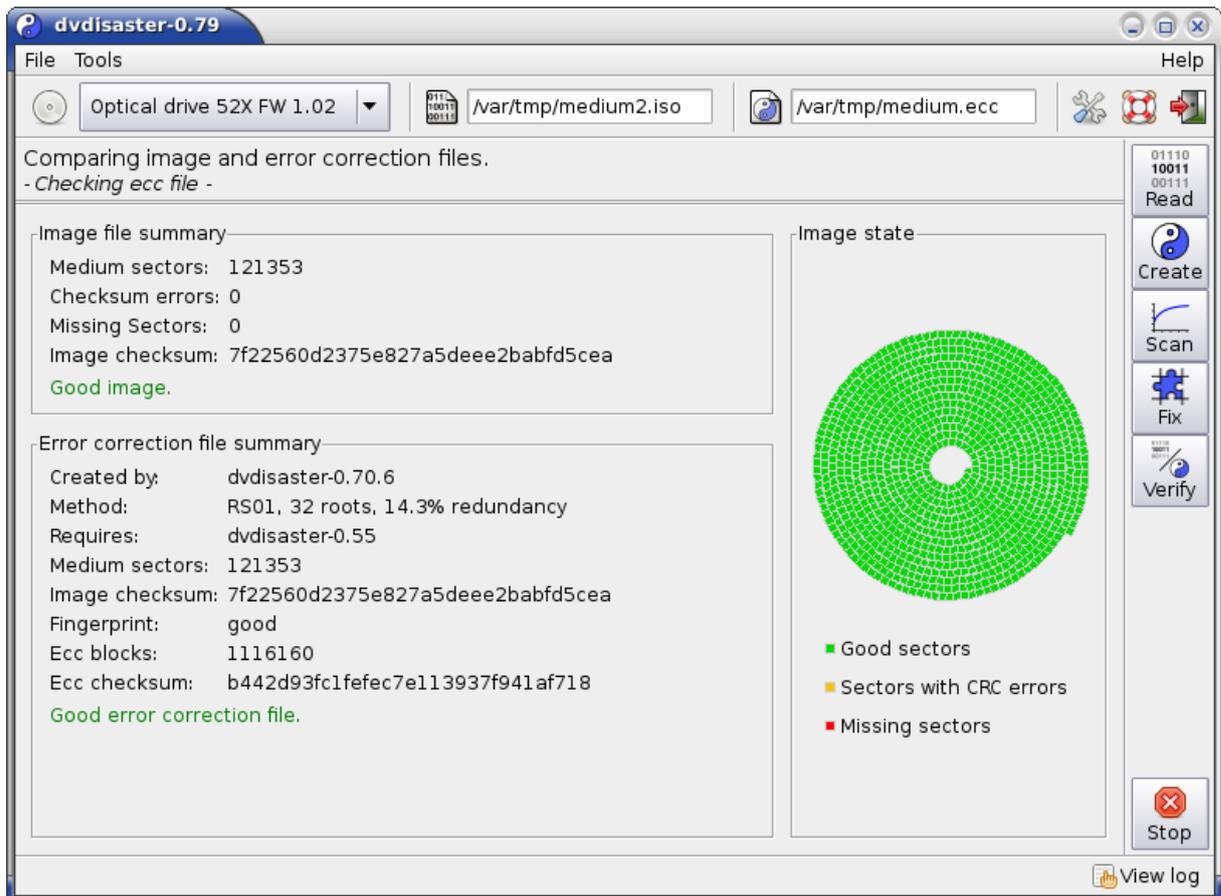
Figure 52: Info for an image with RS01 error correction file.

Output field **"Image file summary"**:

Medium sectors	The number of sectors in the ISO image (one sector = 2KiB).
Checksum errors	The error correction file contains CRC32 checksums for each image sector. If this value is greater than zero some sectors were readable but their contents do not match the checksum. The error correction will try to recalculate the contents of these sectors.
Missing sectors	This is the number of sectors which could not be read from the medium. The error correction will try to recover the contents of these sectors.
Image checksum	A MD5 checksum is calculated for the complete ISO image. You can reproduce this value using the command line of GNU/Linux:

```
md5sum medium2.iso
```

If all values in this output field are okay the message "Good image." appears. Otherwise the most important error will be explained there.

Output field **"Error correction file summary"**:

Created by:	Prints the dvdisaster version which was used for creating the error correction data. Alpha/developer versions are highlighted in red.
Method:	The method and redundancy used for creating the error correction file.
Requires:	Processing the error correction data may require at least the shown version of dvdisaster.
Medium sectors:	The expected number of sectors in the image file (as recorded in the ecc data).
Image checksum:	The expected MD5 sum of the image file.
Fingerprint:	dvdisaster uses the checksum of a special sector to determine whether the error correction file was made for a given image.
Ecc blocks:	The error correction divides the image into small blocks which can be processed independently. This information is not of interest as long as the number of ecc blocks is correct ;-)
Ecc checksum:	A MD5 checkum is calculated over the error correction file, not taking into account the first 4KiB. You can reproduce this value using the command line of GNU/Linux:

```
tail -c +4097 medium.ecc | md5sum
```

If all values in this output field are okay the message "Good error correction file." appears. Otherwise the most important error will be explained there.

2.8.3   Results for images augmented with RS02 error correction data
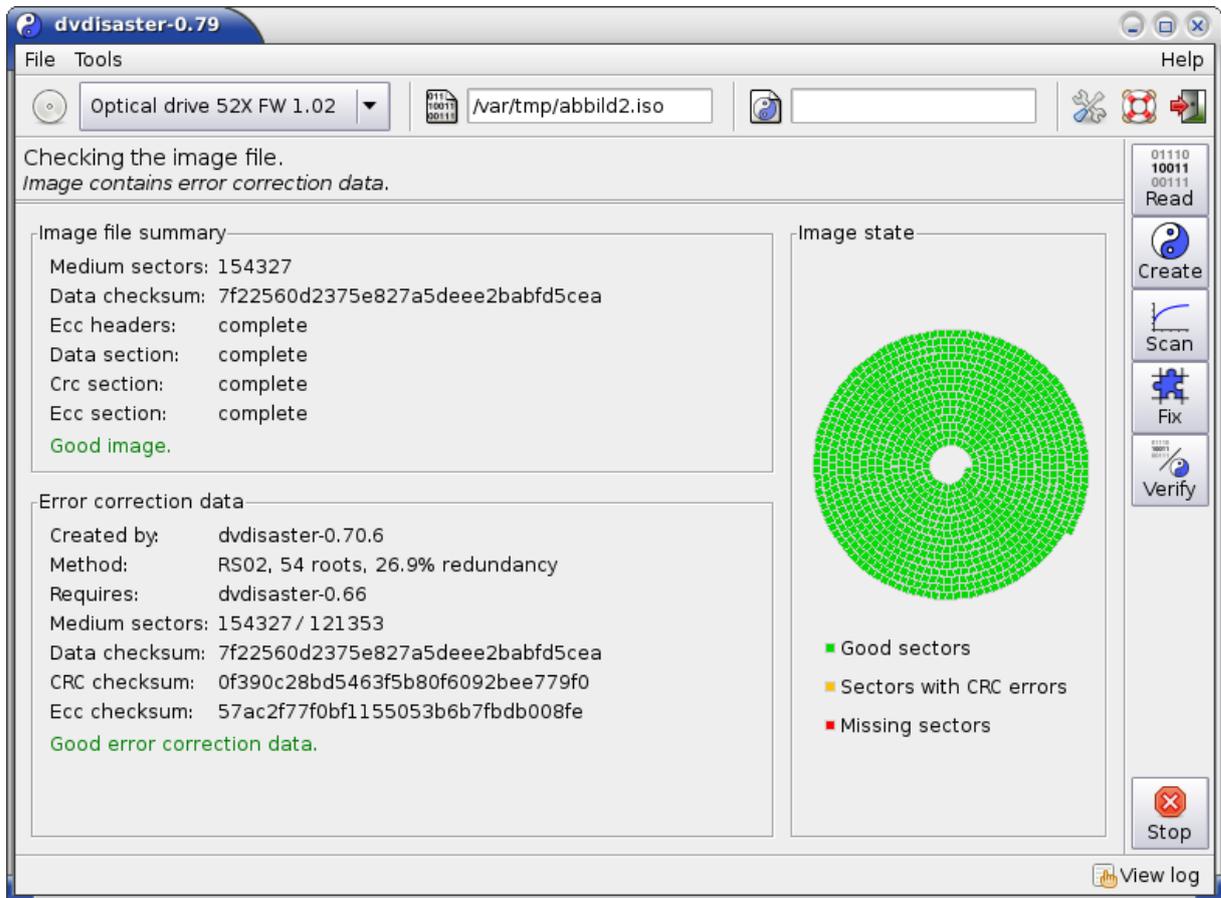
Figure 53: Info for an image augmented with RS02 error correction data.

When verifying an image against its embedded error correction data the information will be given with respect to:

- the whole (augmented) image ("Image file summary"), and

- the error correction data part ("Error correction data").

Output field **"Image file summary"**:

Medium sectors	The number of sectors in the augmented image (including the sectors added by dvdisaster; one sector = 2KiB).
Data checksum:	The MD5 checksum of the original image (prior to augmenting it with error correction data).
Ecc headers, Data section, Crc section, Ecc section	The augmented image consists of three sections plus some ecc header sectors embedded into them. These values describe how many sectors are unreadable in the respective sections.

If all values in this output field are okay the message "Good image." appears. Otherwise the most important error will be explained there.

Output field **"Error correction file summary"**:

Created by:	Prints the dvdisaster version which was used for creating the error correction data. Alpha/developer versions are highlighted in red.
Method:	The method and redundancy used for creating the error correction data.
Requires:	Processing the error correction data may require at least the shown version of dvdisaster.
Medium sectors:	The first value is the number of sectors in the augmented image; the second one describes the number of sectors the image had before it was processed with dvdisaster. Since the error correction data is placed behind the user data, the checksum of the original image can be obtained as follows (using the command line of GNU/Linux):

```
head -c $((2048*121353)) medium.iso | md5sum
```

The first parameter for head is the sector size (2048) multiplied with the original image length (121353). This property of augmented images can also be used to cut off the error correction data:

```
head -c $((2048*121353)) medium.iso
>stripped_image.iso
```

Data checksum:	The MD5 checksum of the original image (see previous explanations).
CRC checksum:	MD5 checksums of the CRC and ECC sections of the augmented image.
ECC checksum:	These two can not be easily reproduced outside of dvdisaster.

If all values in this output field are okay the message "Good error correction data." appears. Otherwise the most important error will be explained there.

2.8.4 Results for RS03 error correction data (ecc files and augmented images)

The RS03 error correction method can produce both error correction files and augmented images. The information screen is very similar for both ways of storing the error correction data (see fig. 54 and 55).

Output field **"Error correction properties"**:

Type:	Indicates whether the error correction data is stored in a separate file ("Error correction file") or contained within the image ("Augmented image").
Method:	The method and redundancy used for creating the error correction data.
Created by:	Prints the dvdisaster version which was used for creating the error correction data.
Requires:	Processing the error correction data may require at least the shown version of dvdisaster.
Data checksum:	The MD5 checksum of the original image used for creating the error correction data, if available. Calculating this checksum requires reading the image twice, so it is turned off by default for performance reasons. Future versions of dvdisaster will provide a switch in the preferences dialogue (currently it is not possible to switch this feature on).
Fingerprint:	dvdisaster uses the checksum of a special sector to determine whether the error correction file was made for a given image. Not used for augmented images.
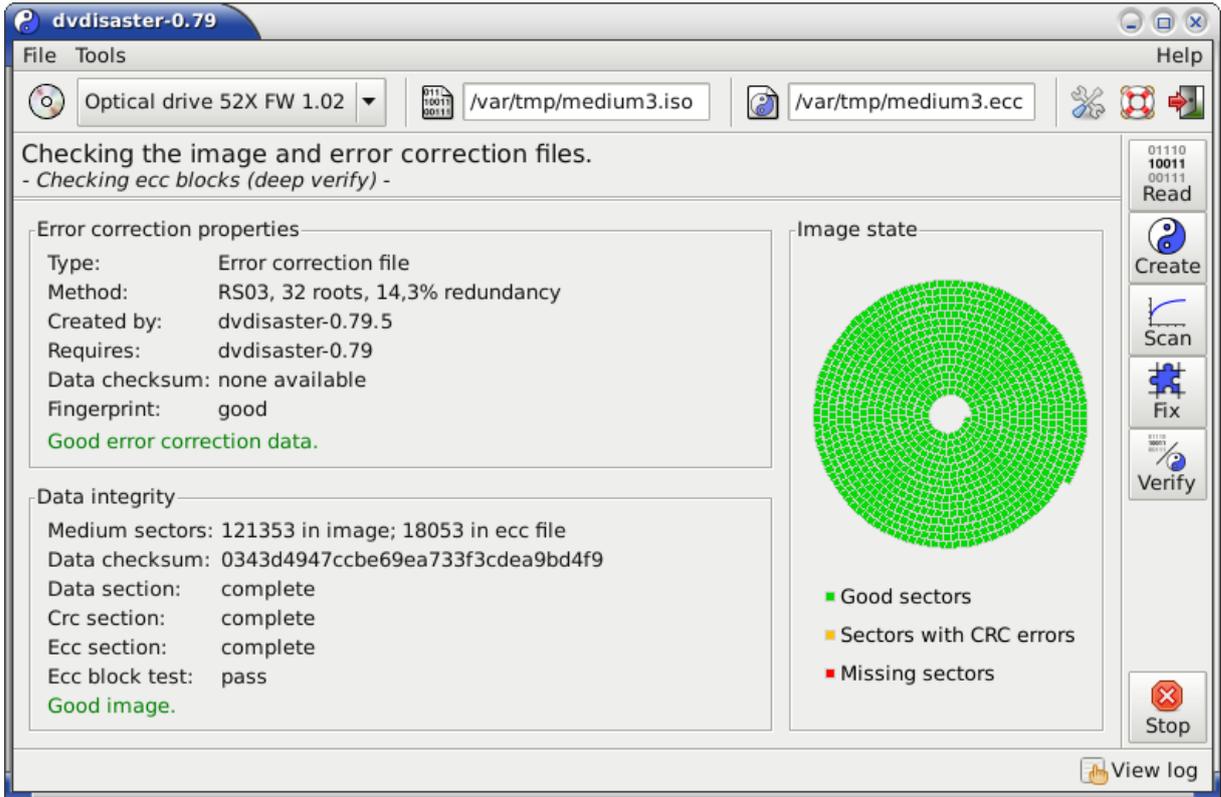
Figure 54: Info for an image with RS03 error correction file.
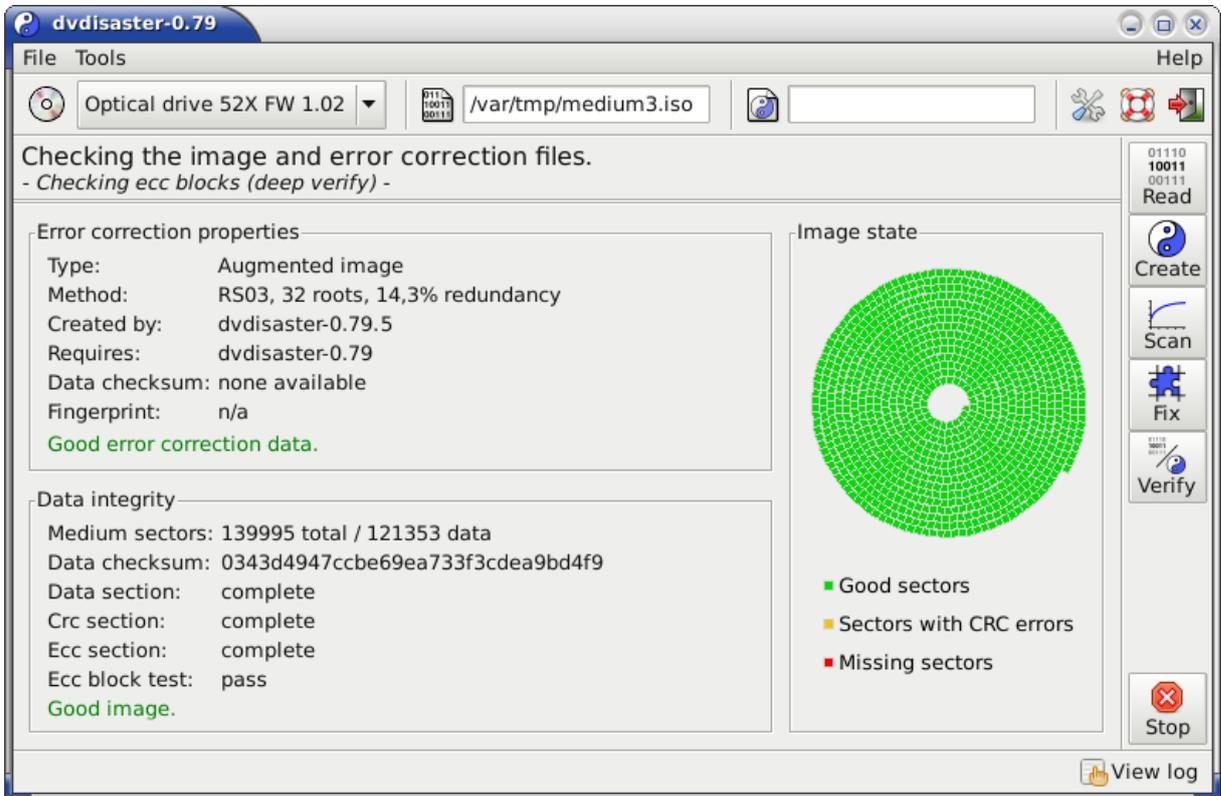
Figure 55: Info for an image augmented with RS03 error correction data.

If no problems are identified with the error correction data, the message "Good error correction data." appears. Otherwise the most important error will be explained there.

Output field **"Data integrity"**:

Medium sectors:	For error correction files, the first value is the number of sectors in the image, and the second value is the size of the error correction file measured in 2048 KiByte sectors. For augmented images, the first value is the total number of sectors in the augmented image, and the second one describes the number of sectors the image had before it was processed with dvdisaster.
	Please note that given the same image and redundancy, the size of the error correction file will not match the size of the ecc data in the augmented image. This is normal behaviour due to padding in the augmented image.
Data checksum:	The MD5 checksum of the image, calculated during the verification process.
	For RS03 augmented images, the checksum can be calculated using GNU/Linux tools just as for RS02. The checksum of the original image can be obtained as follows (using the command line of GNU/Linux):

```
head -c $((2048*121353)) medium.iso | md5sum
```

The first parameter for head is the sector size (2048) multiplied with the original image length (121353). This property of augmented images can also be used to cut off the error correction data:

```
head -c $((2048*121353)) medium.iso
>stripped_image.iso
```

Data section, CRC section, ECC section	If any inconsistencies or missing sectors are found in the image (data section) or the error correction data (CRC and ECC section), they are reported here.
ECC block test:	In addition to testing the internal checksums, the RS03 verify process uses the error correction data to prove that all data in the image (and optionally the ecc file) is correct. This process may take several minutes depending on the image size and can be interrupted using the "Stop" button.

If all values in this output field are okay the message "Good image." appears. Otherwise the most important error will be explained there.

2.8.5   Examples

You have already seen examples of good images and error correction files and good images augmented with error correction data on the previous pages. In the following some typical examples of error situations are presented:
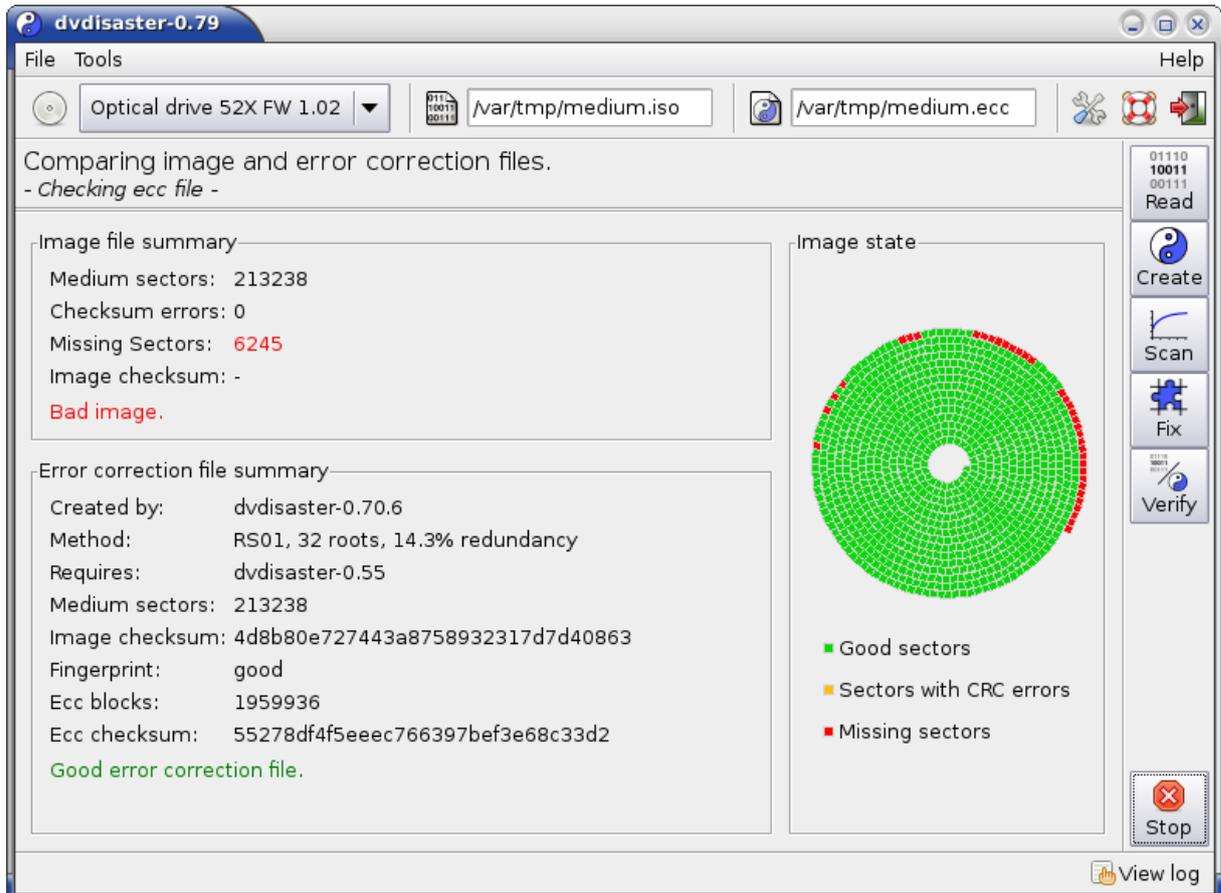
Figure 56: Image with unreadable sectors, RS01.

Image with unreadable sectors and error correction file. This is a typical case of an image from a defective medium, before the error correction has been applied. The image shown here contains 6245 unreadable sectors; error correction data is present by means of an error correction file.
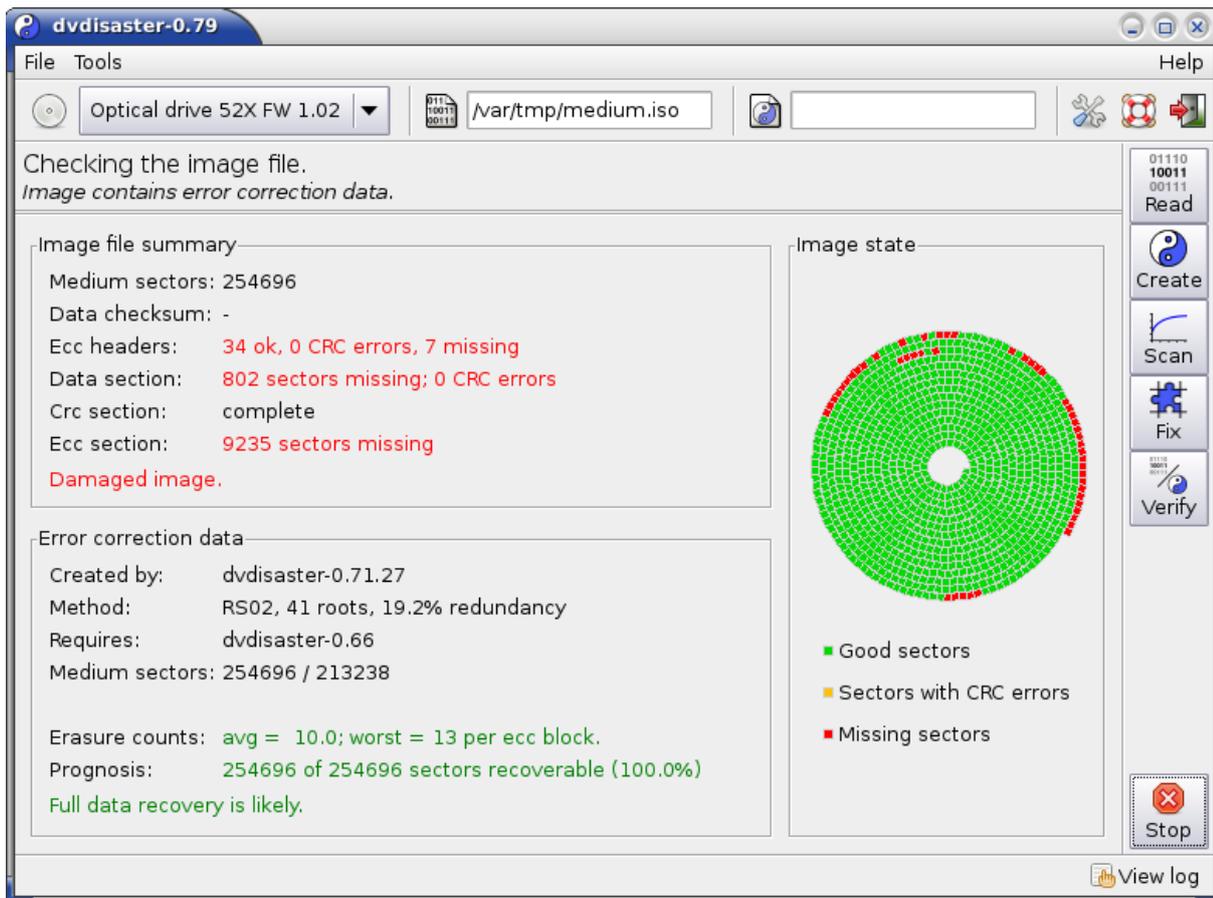
Figure 57: Image with unreadable sectors, RS02.

Image augmented with error correction data, containing unreadable sectors. This is another example of an image from a defective medium before recovery. This image contains unreadable sectors towards its end. Especially the ECC section is affected since the error correction data is located at the end of the image. Please note that this does not weaken the error correction since its corrective power is independent from the error location: 10000 errors at the beginning of the medium are just as easy to correct as 10000 errors towards its end. See also the introduction into the error correction and QA item 1.4 on this topic.

The RS02 encoder which was used for creating the error correction data is capable of predicting the odds of a successful image recovery. These are shown at the end of the error correction data output area.
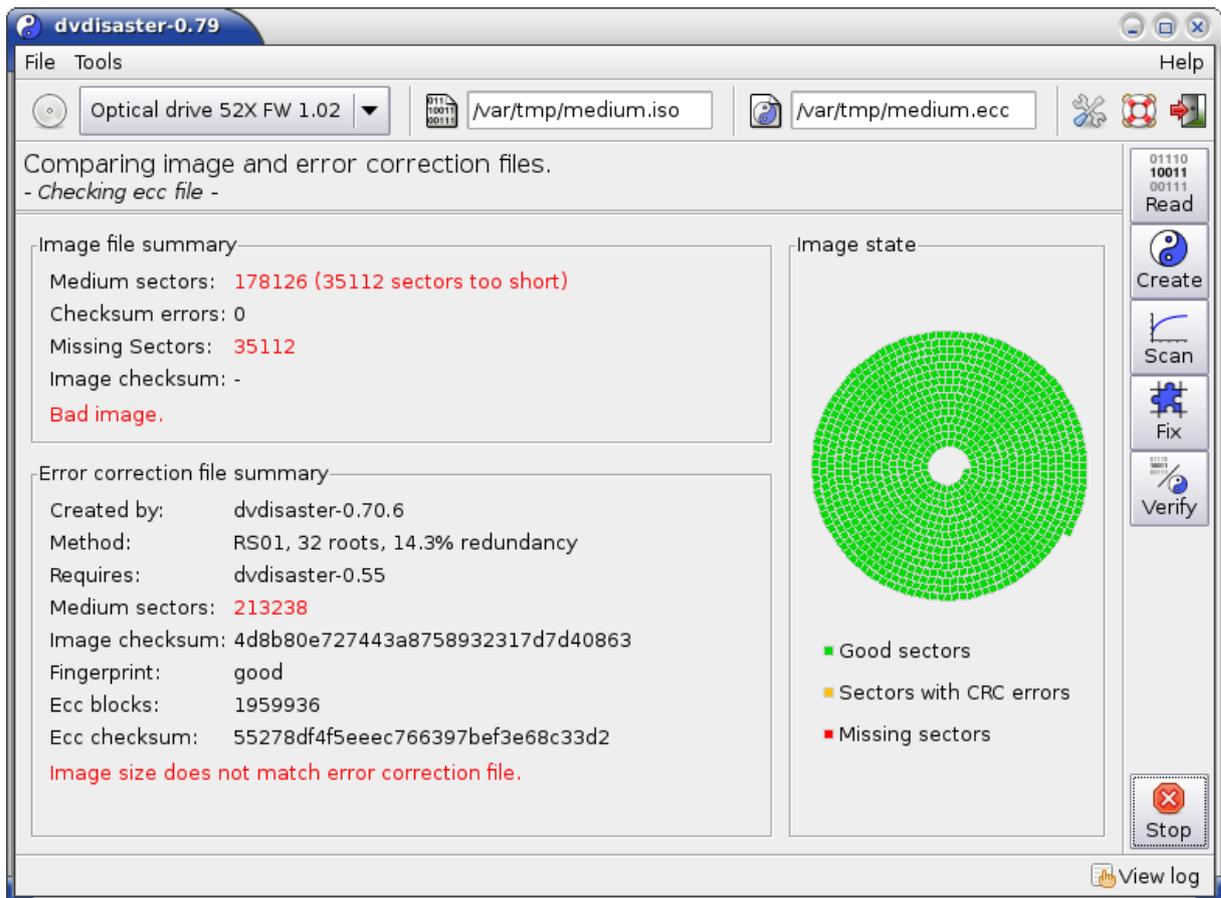
Figure 58: Truncated image due to an aborted read, RS01.

Image from aborted reading process.   This image is shorter than expected; this usually happens when the reading process is stopped prematurely.
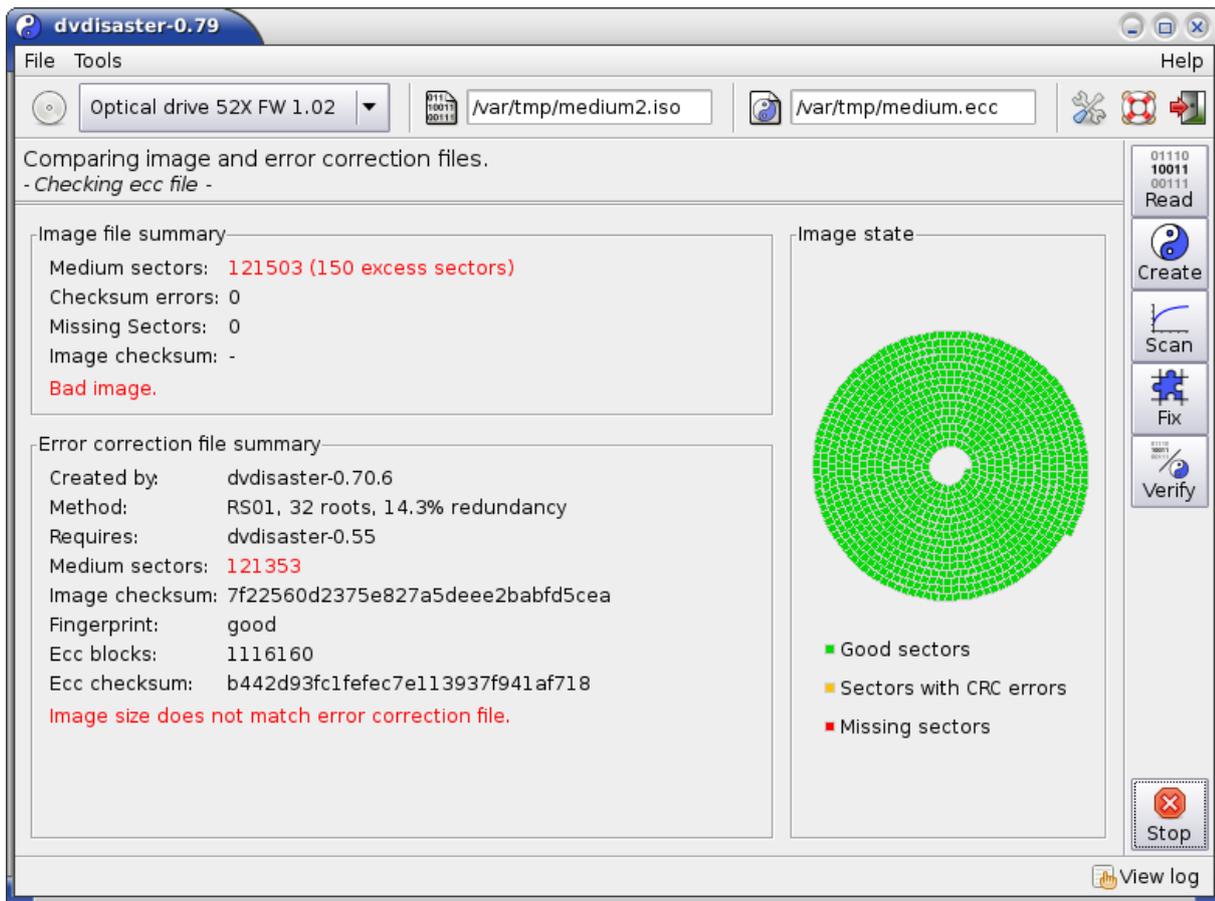
Figure 59: Image too large because of padding sectors, RS01.

Image is larger than expected. This image is larger than expected; possible causes are discussed in the section about image compatibility. It may be possible to recover from this problem; see hints related to using error correction files and using augmented images.
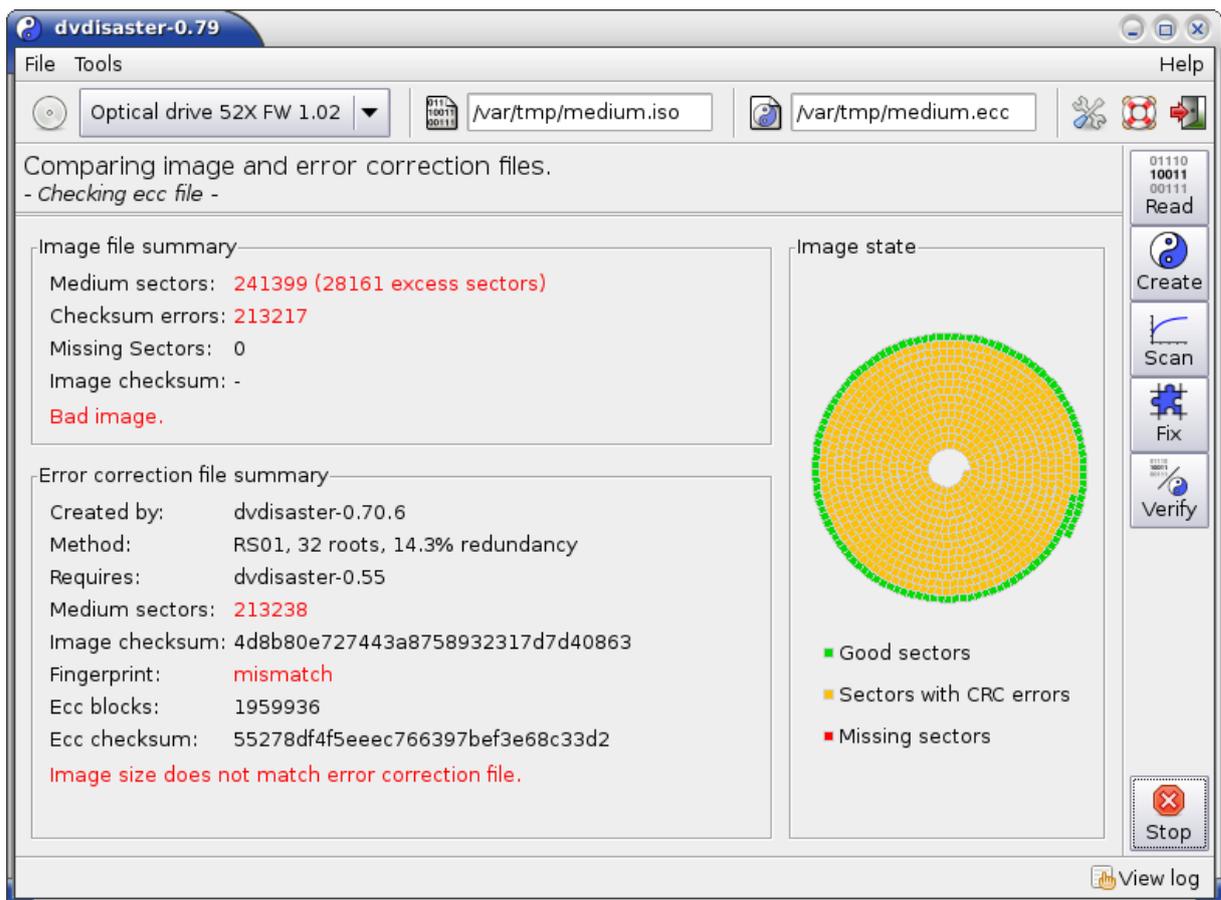
Figure 60: Wrong error correction file, RS01.

Wrong error correction file.   The error correction file was created for a different image. This causes lots of CRC errors since the data sectors have different contents. However the most important hint is:

Fingerprint: mismatch

This tells you that the error correction file does not belong to the image.

2.9   Testing image compatibility

Since dvdisaster depends on your favourite media writing software for writing media and creating ISO images, it is important to make sure that both softwares agree on how to create a proper ISO image.

Why dvdisaster uses ISO images.   Some dvdisaster functions work on image files stored on hard disk. Optical drives are too slow for carrying out the required access patterns, and they would wear out quickly during this process. However hard disks are designed for this type of access, and they do them quickly and without wear.

Testing image compatibility is important.   During the work with dvdisaster you can (and sometimes must) use ISO images which have been created by third-party software. Most products create usable images when advised to use the ".iso" file format, but some alter the image while writing it to the medium. This renders ecc data unusable. Also, processing non-iso images with dvdisaster will result in unusable error correction data. Especially, **formats like .nrg are not suitable** for processing with dvdisaster.

Possible scenarios.   The following situations require exchanging ISO images between dvdisaster and a third party software:

a) Creating error correction files from ISO images made by a CD authoring software

  An optical disc authoring software is used to create an ISO image. This image is used for writing the medium and for creating the error correction file. When using the authoring software for the first time with dvdisaster, make sure that the image was written to the medium without modifications.

b) Augmenting ISO images with error correction data

  dvdisaster adds "invisible" error correction data to the ISO image in order to minimize interference with other applications reading from the medium. But this invisibility might prevent some optical disc writing software from properly writing the error correction data to the medium. Make sure that your writing software does correctly transfer the error correction data when using it with augmented images for the first time.

2.9.1   Testing compatibility with CD/DVD/BD writing software for error correction files

Motivation: You want to write data to a medium and create an error correction file for it. In order to save time you do the following:

1. You create an ISO image using your optical disc writing software.

2. You write the image to a medium.

3. You create the error correction file from the same image.

Possible incompatibility: The writing software creates a medium which does not exactly match the image. This might prevent the error correction from recovering the medium contents when it becomes defective.

How to test compatibility: Please note that some steps are only sketched out here; follow the links to the respective sections to find detailed instructions and examples.
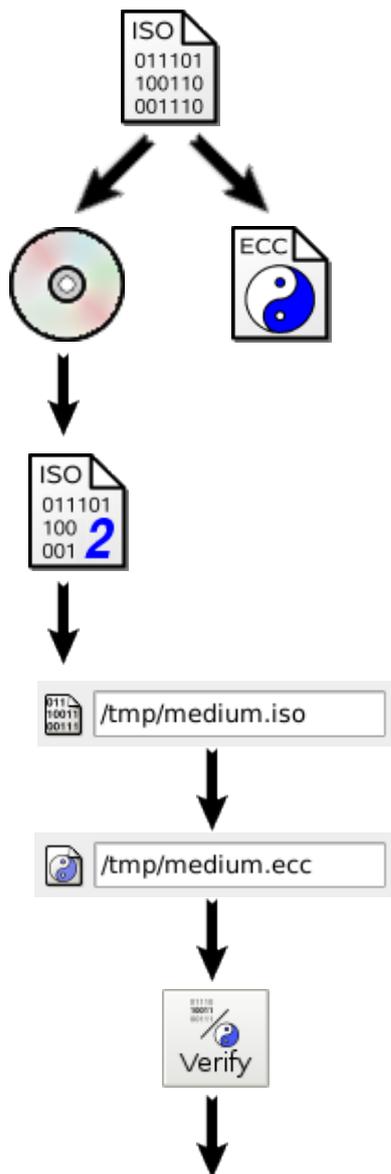
Create an ISO image of the data you want to write on the medium. If you need help on creating ISO images please refer to the example of creating ISO images.

Write the medium and create the error correction file. Use the just created image to write the medium. Then perform these basic settings and create an error correction file from the image you still have on your hard disk (e.g. do not yet read it back from the medium you have just created).

Create a *second* image from the *written* medium. Use these settings and read the medium as described in creating an image for making an error correction file. However you can stop the walk-through when the reading is finished as we do not need to create the error correction file again.

Enter the file name of the *second* ISO image which you have just read from the medium. Please note that the following test is useless when working with the image which was initially created using the optical disc authoring software.

Enter the name of the error correction file in case it is not already present from the previous actions.

Start the evaluation by clicking on the "Verify" button.

Look at the verification results. If you get the green messages "Good image." and "Good error correction file." your authoring software and dvdisaster are compatible. You can continue creating the error correction files directly from the ISO images produced by the authoring software.
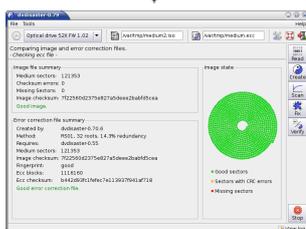
Possible error causes and remedy:
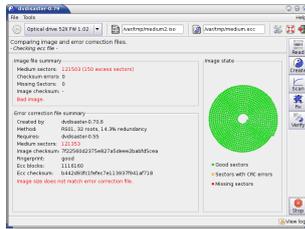
Typical problem: wrong image size. The verification process may find out that the image is larger than expected. Typically the difference is 150 or 300 sectors for CD media and 1-15 sectors for DVD/BD media. These might simply be zero padding sectors appended to the image by the writing software. To find out if this really is the case do the following:

Start a recovery process.

Confirm the dialog. A dialog will appear asking you if it is okay to remove the superflous sectors from the image. Answer "OK".

Stop the recovery process, as after truncating the image there is nothing more to do.

Start the verification again by clicking on the "Verify" button.
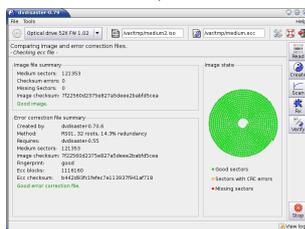
Consider the new results. If you now get the green messages "Good image." and "Good error correction file." your problem is purely cosmetic: The writing software has indeed added zero padding sectors while writing the medium which can always be removed by the process described here.

If the problem persists after carrying out the above steps do not assume that dvdisaster and the writing software are compatible. The created error correction files will probably be unusable.

Use the following method for creating the error correction files instead:

Alternative method avoiding incompatibilities:

1. First write the data to the medium.

2. Use dvdisaster to read an ISO image from the written medium.

   Do *not use* the original ISO image.

3. Use the image read by dvdisaster to create the error correction file.

This method takes more time due to the additional reading process, but it also has the advantage of testing the newly created medium for readability.

2.9.2   Testing compatibility with CD/DVD/BD writing software for images augmented with error correction data

Motivation: dvdisaster can put error correction data together with the user data on the medium. The error correction data is appended to the ISO image in a way invisible to most applications in order to not interfere with them.

Possible incompatibility: The optical media writing software may also be unable to see the error correction data. While being unlikely it is possible that the writing software will truncate or damage the error correction data while creating the medium. In this case the error correction will not work.

How to test compatibility: Please note that some steps are only sketched out here; follow the links to the respective sections to find detailed instructions and examples.
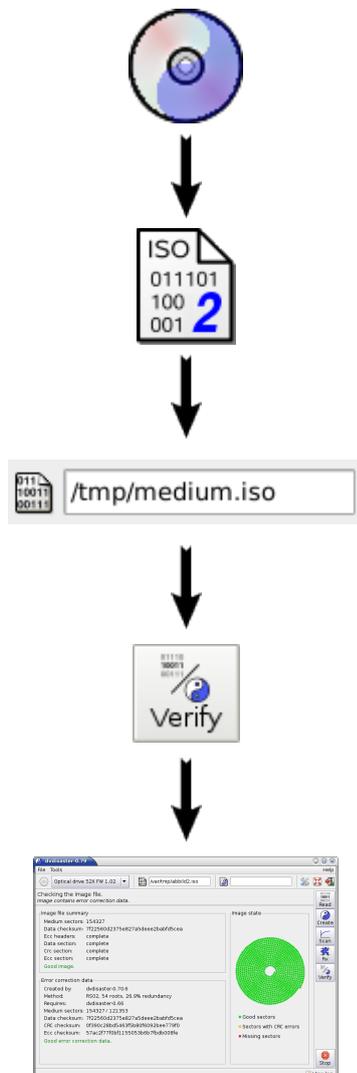
First create a medium which was augmented with error correction data. Do not forget to use the proper settings and follow the step by step instructions. Do not use rewriteable DVD or BD media as they may influence the test under some circumstances (see item 3.4 in the questions and answers).

Create a second image from the written medium. Use the same settings and steps as in reading a medium for creating an error correction file; however you can stop after the reading has finished as we do not need the error correction file.

Enter the name of the *second* ISO image which you have just read from the medium. Please note that the following test is useless when working with the image which was initially created using the optical disc authoring software and augmented with dvdisaster.

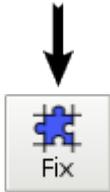**Start the verification again** by clicking on the "Verify" button.

Look at the verification results. If you get the green messages "Good image." and "Good error correction data." your authoring software and dvdisaster are compatible with respect to the augmented images.
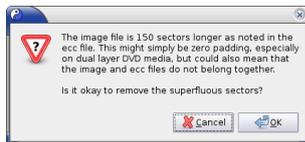
Possible error causes and remedy:
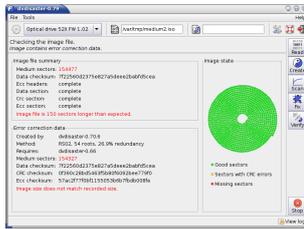
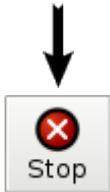**Typical problem: wrong image size.** The verification may find out that the image is larger as expected. Typically the difference is 150 or 300 sectors for CD media and 1-15 sectors for DVD/BD media. These might simply be zero padding sectors appended by the writing software. To find out if this really is the case do the following:
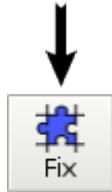
**Start a recovery process.**

Confirm the dialog. A dialog will appear asking you if it is okay to remove the superflous sectors from the image. Answer "OK".

Stop the recovery process, as after truncating the image there is nothing more to do.

Start the verification again by clicking on the "Verify" button.
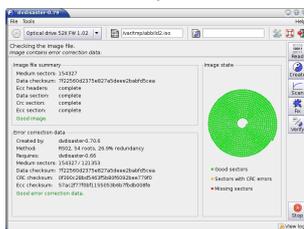
Consider the new results. If you now get the green messages "Good image." and "Good error correction file." your problem is purely cosmetic: The writing software has indeed added zero padding sectors while writing the medium which can always be removed by the process described here.

If the problem persists after carrying out the above steps you can not use the optical disc writing software for creating media from augmented images. Perform the test again using a software from a different vendor.

2.10   Dialogs and buttons

This section explains commonly used dialogs and buttons in the dvdisaster user interface:

The drive selection menu.
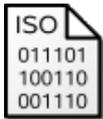
The image file chooser window.

The error correction file chooser window.
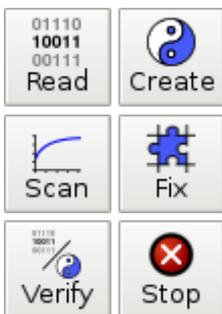
The buttons for starting actions.

2.10.1   Drive selection

Figure 61: Location of the drive selection in the main window.

The drive selection menu is located in the upper left corner of the tool bar (see green marking). Click into the field to the right of the CD symbol to drop down the drive selection. Then select the drive which contains the medium you want to process with dvdisaster.

To simplify identification of the drives the following information is given in the menu entries:

- The device identification which is typically comprised of the vendor name and the drive model number. These values have been programmed into the drive by the vendor. Since dvdisaster is displaying them without further processing you will see here whatever the drive vendor deemed appropriate. Sometimes this identification is not very meaningful.

- The handle under which the drive is managed by the operating system (e.g. /dev/sr0 using GNU/Linux)

Example (for GNU/Linux):

Figure 62: Unfolded drive selection.

2.10.2   Image file selection
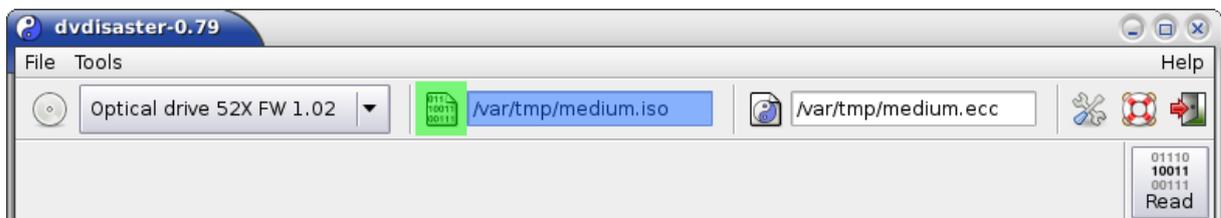
Figure 63: Location of the image file selection in the main window.

The image file contains the data from all medium sectors, including the information whether a sector was readable. dvdisaster works on image files because they are stored on hard disk which makes certain random access patterns much faster. Applying this kind of random access to CD/DVD/BD drives would slow them down significantly and eventually wear them out (the image files are read/created using sequential access which unlike random access is handled efficiently by the drives). The default file suffix for images is ".iso".

There are two ways of choosing the image file:

- using a file chooser dialog (button marked green), or

- by directly entering the file location (text entry field marked blue).

The direct entry is helpful when you are processing several files in the same directory. In that case simply change the file name in the text field.

2.10.3 Error correction file selection

Figure 64: Location of the error correction file selection in the main window.

The error correction file contains information for reconstructing unreadable sectors from a defective medium. It can also be used to check a medium for damaged or altered sectors. The default file extension is ".ecc".

There are two ways of choosing the error correction file:

- using a file chooser dialog (button marked green), or

- by directly entering the error correction file location (text entry field marked blue).

The direct entry is helpful when you are going to create several error correction files in the same directory. In that case simply change the file name in the text field.

If you are processing images which are augmented with error correction data it is recommended to leave this field blank; e.g. to delete any text within it.

2.10.4   Starting actions

To start an action in dvdisaster, click on one of the buttons
marked green in the thumbnail image to the right.

Reading medium contents into an image file to:

- read in a defective medium for a subsequent recovery.

- read in an error-free medium for creating an error correction file.

Creating error correction data
(only possible from defect-free media!)

Scanning a medium for read errors

Recover the image of a defective medium
provided that error correction data is available.

Display information on images and error correction data.

Other buttons related to the above actions:

View log file of running action (marked yellow).
See also: Log file creation.

Aborting the running action (marked red).
Some actions may take some time to abort; especially when this button
is hit while reading a defective sector.

3   Downloading dvdisaster

dvdisaster is available for recent versions of the FreeBSD, GNU/Linux and NetBSD operating systems. It is provided as free Software under the GNU General Public License v3.

The dvdisaster developer site (`https://dvdisaster.jcea.es/`) contains source code releases for the FreeBSD, GNU/Linux and NetBSD operating systems. These are mostly aimed at maintainers of binary packages for the beforementioned platforms. As an end user you might find it more convenient to install dvdisaster from the package system of your operating system bundle or distribution. But if you prefer to download and compile the source package on your own, you're welcome to visit the site.

The source code archives contain a file `INSTALL` with further instructions for building dvdisaster. The build process follows the common `./configure; make; make install` scheme. The installation step is optional; you can use dvdisaster directly from the build tree.

3.1   System requirements

Hardware requirements

- x86, ARM or PowerPC processor;

- an up-to-date CD/DVD/BD drive with ATAPI, SATA or SCSI interface;

- enough hard disk space for creating .iso images from processed media.

Supported operating systems

The following table gives an overview of the supported operating systems. The specified releases have been used for developing and testing the dvdisaster version made available as of this writing. Typically, slightly older and newer OS versions will also work.

Operating System	Release
GNU/Linux	Debian Jessie 8.0, Kernel 3.16
FreeBSD	10.1
NetBSD	6.1.5

Not supported operating systems

Support for Windows and Mac OS X has been ended and is not planned to be resumed in the future (see QA item 2.4 for an explanation).

3.2   Make sure you're not getting ripped off: The small print (and other things).

The dvdisaster project provides this software as free software to you using the GNU General Public License v3. The dvdisaster project also wants to make sure that you know you can download the software at no cost and keeping your full privacy. To make it clear how we distribute dvdisaster, what we do and what we won't do, we have compiled the following list:

Internet and download sites

The dvdisaster project uses the following web site for supplying software downloads:

<div align="center">

`https://dvdisaster.jcea.es/`

</div>

No other internet or download sites are run by the dvdisaster project.

No money or personal data required.

There is **no registration process** for using this software. The dvdisaster project **never** asks you to enter personal data, to pay a fee or to donate money for:

- using its internet and download sites,
- downloading the software, and
- running the software.

In fact, please do not offer any donations to the project. We cannot accept them for various reasons.

Cryptographic signature and checksums

dvdisaster releases are always published with cryptographic signatures and md5 checksums. See the download site for examples. Be very cautious if signatures and checksums are missing, invalid or not matching those published at the sites mentioned above.

4 Questions and answers

This section answers some general questions on dvdisaster. There is also a sub section about technical questions, and one discussing error messages.

Q 1.1: How is "dvdisaster" pronounced?

Since the word stems from the english language, simply spell "dv" before saying "disaster". Perhaps "dee-vee-disaster" is a suitable phonetic circumscription.

Q 1.2: What are quality scans and why don't you support more?

Optical media have a built-in error correction which is similar to the method used in dvdisaster. Some drives can report the number of errors corrected while reading a medium. This provides a rough estimate of the writing and media qualities.

Since dvdisaster is free software, it can only include code and information which can be redistributed freely. This is currently true for C2 scanning of CD media, which is officially standardized and has free documentation available.

On the other hand, DVD quality scans ("PI/PO scans") and BD disc scans are not standardized. Those drive vendors who support it are using proprietary programming interfaces. The respective specifications seem not to be available for use in free software. So we must patiently wait until manufacturers understand that having free software available for a drive will sell more drives.

Q 1.3: Is dvdisaster compatible with future releases?

Yes, dvdisaster files are intended for an archival time of many years. When upgrading to a newer version of dvdisaster you can continue using images and error correction data created from previous versions. There is no need to recreate them again.

Q 1.4: Augmented images have the error correction data appended at the end of the medium. Isn't that a bad choice?

No. First a bit of terminology: If we augment 80 bytes of user data with 20 bytes of error correction (ecc) data, we get an "ecc block" comprised of 100 bytes. Now take the following into consideration about the ecc block:

1. The position of the error correction data within the ecc block does not matter.

   The Reed-Solomon (RS) decoder does not differentiate between user data and error correction data. In the view of the RS decoder our ecc block is a sequence of 100 bytes from which an arbitrary subset of 20 bytes can be recovered. It can recover the first 20 bytes, the last 20 bytes, or any combination from within as long as the remaining 80 bytes are still intact. From this it follows that the position of the ecc data within the ecc block does not matter; whether it is appended at the end of the user data or is interleaved with it has no influence on the error correcting capability.

2. Properly distributing the ecc block offsets influence of bad media spots.

   Optical media have a higher probability of failing in the outer area; but for technical reasons this is the only place where the error correction data can be stored. However

this effect is offset by distributing the ecc block content over the medium. Let's assume that our medium is filled 80% with user data, leaving the remaining 20% free for error correction data. Now consider the 100 byte ecc block again. We need to pick 80 bytes from the user data for it and require 20 additional byte positions in the error correction data area. Even under these constraints it is possible to evenly distribute the 100 bytes over the medium, from the inside to the outside, each having a maximum distance to its neighbors. This means that under the above scenario, 80% of each ecc block comes from the first 80% of the medium, and 20% from the rest (the outer area). Together with point (1), this negates the influence of bad spots on the medium. Symmetry implies that for each error correction byte stored in the (bad) outer region there will be a user data byte located in the (good) inner medium region.

(If you do not already see the point, imagine putting the ecc data into the inner medium region and the user data in the outer region. Consider point (1) again to see that nothing changes with respect to the error correction.)

Q 1.5: What's the difference between image based and file based data recovery?

Optical media are comprised of 2048 byte-wide sectors. Most of those sectors are used to store file data, but some of them hold so-called "meta data", e.g. information on directory folders. In figure 65 (below) there is a directory "Pics" holding three files "forest.jpg", "rock.jpg" and "protect.par"[6]. Note how these files are mapped onto physical sectors (green/blue squares) on the medium, and that an additional meta data sector (red square) is needed for storing the "Pics" directory structure.
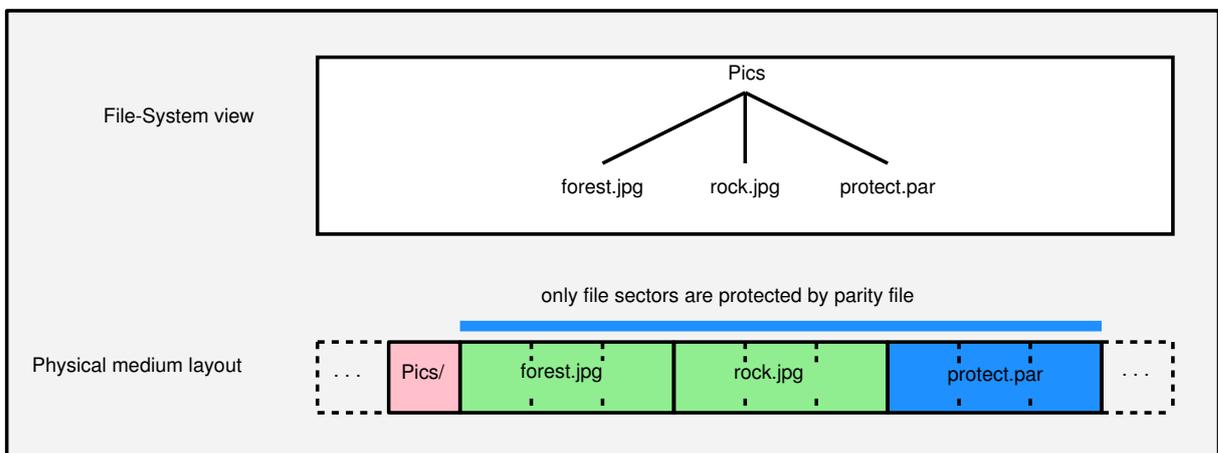
Figure 65: Relation between file system and sectors on the medium.

Shortcoming of file based recovery on optical media. Now let's assume that we are working with file based error correction. The file "protect.par" holds error correction information which can be used to recover unreadable sectors within the files "forest.jpg" and "rock.jpg". This will

---

[6]No offense intended against the PAR/PAR2 project. Carsten is just confident that file based protection does not work on optical media :-)

only work as long as we need to recover sectors which are part of a file. But if meta data sectors become unreadable, the file based protection will collapse. Consider figure 66. When the red directory sector becomes unreadable, not only the directory "Pics" but also all files under "Pics" become inaccessible. This is due to the logical structure of the ISO/UDF file system, as there is no way to tell how the green and blue sectors relate to files anymore when the directory is lost. So we have a complete data loss although all sectors comprising the files are still physically readable.
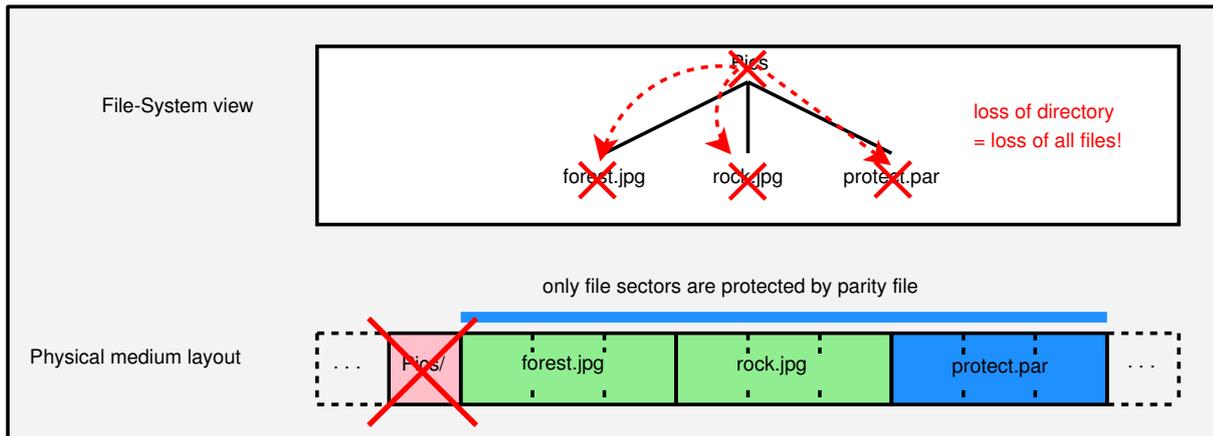
Figure 66: Losing the metadata sector produces complete data loss.

Please note that moving "protect.par" to a separate medium does not rectify the problem - the directory block is still not recoverable as it is not protected by the error correction data in "protect.par".

Advantages of image level recovery on optical media.   dvdisaster applies an image level approach to error recovery. The medium is read and processed as an ISO image. The ISO image contains a sequence of all sectors found on the medium, including those which are meta data for the file system. Since the dvdisaster error correction data protects all sectors in the ISO image, file contents as well as meta data sectors (e.g. directories) can be restored. See fig. 67 for the different range of protection.

In addition, neither reading the damaged ISO image nor applying the error correction requires any information from the file system contained on the medium. As long as the drive is still able to recognize the medium, dvdisaster will be able to recover the still readable sectors from it. Therefore there are no "single sectors of failure" as in the file based approach.
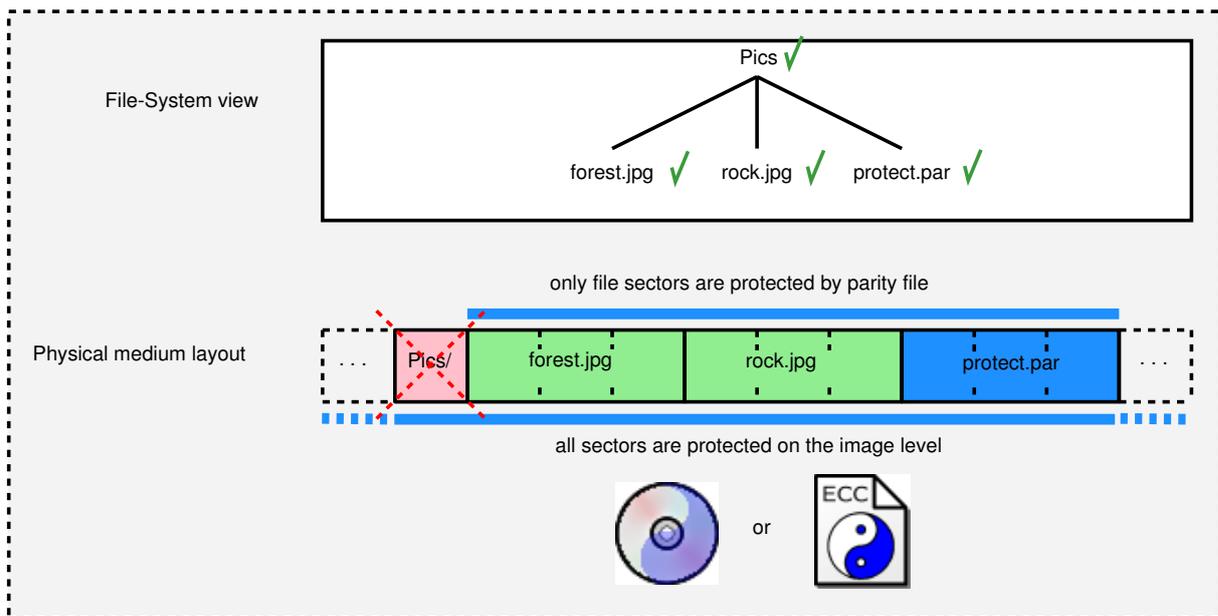
Figure 67: Image level protection

4.1 Technical questions

Q 2.1: Which on-screen translations of the program are available?

The current version of dvdisaster contains screen texts in the following languages:

Czech	–	unmaintained and outdated
English	–	complete
German	–	complete
Italian	–	unmaintained and outdated
Portuguese	–	unmaintained and outdated
Russian	–	unmaintained and outdated
Swedish	–	unmaintained and outdated

Translators for other languages are welcome!

dvdisaster will automatically obtain language settings from the operating system. If the local language is not yet supported, english text will be used. A different language can be selected using environment variables.

Example for the bash shell and german language:

```
export LANG=de_DE.utf8
```

Please note that we are currently not providing translations of this user manual and the internet site. See topic Q 2.5 for more details.

Q 2.2: Which media types are supported?

dvdisaster supports (re-)writeable CD, DVD and BD media. Media containing multiple sessions or copy protection can *not* be used.

Usable media by type:

CD-R, CD-RW

- only Data CD are supported.

DVD-R, DVD+R

- No further limitations are known.

DVD-R DL, DVD+R DL (two layers)

- The drive must be able to identify (Q3.5) the medium type. Typically this is only the case for drives which can also write two layered media.

DVD-RW, DVD+RW

- Some drives report wrong image sizes (Q3.4).
  Remedy: Determine the image size from the ISO/UDF file system or the ECC data.

DVD-RAM

- Usable only when written with ISO/UDF images like DVD-R/-RW;

- Not supported when used as removeable medium for packet writing.

- Similar issues with image size recognition (Q3.4) as noted above.

BD-R, BD-RW   (one or two layers)

- No limitations are known.

BDXL-R

- Currently only the three-layered (100GB) version is tested and supported.

- Note: BDXL is not backwards compatible with standard BD drives.

Not usable types (image can not be extracted):

- BD-ROM (pressed BDs), DVD-ROM (pressed DVDs), CD-Audio and CD-Video.

Q 2.3: Which file systems are supported?

dvdisaster works exclusively on the image level which is accessed sector-wise. That means it does not matter with which file system the medium has been formatted. See also Q1.5.

Since dvdisaster neither knows nor uses the file system structure, it can not repair logical errors at the file system level. It can not recover lost or deleted files.

Q 2.4: What happened to the Windows and Mac OS ports?

As you may have noticed, the project has progressed very slowly in the last years. The main developer is currently very short of time, and some co-developers are also busy with other tasks. So the question was either to stop development at all, or to continue developing using as few resources as possible.

The main problem is that we have just one version of dvdisaster which needs to live up to the different standards of GNU/Linux, Windows and Mac OS. The path of least resistance is to continue developing dvdisaster for GNU/Linux, since dvdisaster was originally started on GNU/Linux and is still primarily developed on it. dvdisaster is a "native" GNU/Linux application, meaning that it only uses tools and interfaces which come bundled with GNU/Linux, such as the GNU compiler suite and the GTK+ graphical user interface toolkit.

Producing the Windows version requires a huge effort, as dvdisaster can not be built with native Windows tools like Visual C++. It requires installing an Unix-like development environment containing the GNU compiler suite (e.g. from MingW) and a port of the GTK+ toolkit. These tools need to be obtained directly from the upstream source (e.g. from www.gtk.org), as no responsible developer would use pre-made binaries from third-party sites. That guarantees for huge extra work every time the tool chain needs to be updated.

When the project was started in the year 2004, the Windows 2000 port of GTK+ provided a sufficient compatibility layer to make a decent looking and usable Windows version. Today, since Windows 8 the operating system has diverged so significantly from GNU/Linux that it makes GTK+ based applications look poor in terms of usage and visual appearance. Also, Windows is taking different ways in dealing with 64-bitness and multithreading over several CPU cores. This makes it very difficult to write an implementation of the RS03 codec which works well on both GNU/Linux and Windows. To create a version which lives up to the current standards of both GNU/Linux and Windows, dvdisaster would have to be written from scratch with native bindings to the Windows GUI and process interfaces. The same is true for Mac OS; necessitating another complete rewrite of the source code. Resources for that are not available, so the project will just continue developing the GNU/Linux version.

Please note that this does not make your error correction data immediately worthless under Windows, as you can continue using the current 0.72.3 binary for a long time. Also, ecc data created under Windows can be used to recover media using GNU/Linux, even when booting Linux as a live system from a memory stick.

Q 2.5: How can I help providing translations?

There are several ways of providing translations to the project.

Translating the on-screen texts.   See topic Q 2.1 for a list of available translations. You are welcome to take over an unmaintained translation or to provide one for a new language. Send an email to *carsten@dvdisaster.org* for further instructions.

Translating the user manual and/or the internet site.   The project is *currently not accepting contributions for translations of the internet site and the user manual.* Coordinating a translation effort is a very time consuming process which is currently exceeding our resources.

You are welcome, of course, to provide an unoffical translation of the project documentation on your own webspace and at your own discretion.

Q 2.6: There are problems opening this manual from dvdisaster

dvdisaster uses `xdg-open` for bringing up the PDF viewer. `xdg-open` keeps a record of file types and applications capable of displaying them. However on some systems, `xdg-open` is either not configured properly for viewing PDF, or misconfigured to use inappropriate programs like *Gimp*.

Remedy.   Configure `xdg-open` properly. Usually the configuration is found in the "settings" menu of your desktop. As an alternative, use the command line to configure it for *xpdf*:

```
xdg-mime default xpdf.desktop application/pdf
```

4.2   Error messages

Q 3.1: "Warning: 2 sectors missing at the end of the disc"

This warning appears with CD media written in "TAO" (track at once) mode. Some drives report an image size which is 2 sectors too large for such media, producing 2 pseudo read errors at the end of the medium which do not mean data loss in this case.

Since the writing mode can not be determined from the medium, dvdisaster assumes a "TAO" CD if exactly the last two sectors are unreadable, and trims the image accordingly. It is up to you to decide whether this is okay. You can advise dvdisaster to treat these sectors as real read errors by using the `--dao` option or the preferences tab for reading/scanning.

To avoid these problems, consider using the "DAO / Disc at once" (sometimes also called "SAO / Session at once") mode for writing single session media.

Q 3.2: Program blocks right after invocation

Using old Linux kernel versions (kernel 2.4.x) the program occasionally blocks right after the start and before any actions are carried out. It can not be terminated using Ctrl-C or "kill -9".

Eject the medium to make the program terminate. Insert the medium again and wait until the drive recognizes the medium and spins down. Calling dvdisaster again should work now.

Q 3.3: What does "CRC error, sector: n" mean?

The respective sector could be read, but the checksum of its contents does not match the value noted in the error correction file. Some possible causes are:

- The image has been mounted with write permission and was therefore altered (typical evidence: CRC errors in sector 64 and in sectors 200 to 400).

- The computer has some hardware problems, especially when communicating with its mass storage devices.

If you suspect technical problems, try creating another version of the image and error correction files and verify them again. When errors disappear or surface at a different locations, your computer may be suffering from defective memory, broken drive cabling, or wrong CPU/system frequency settings.

Q 3.4: Read errors or wrong image size with -RW/+RW/-RAM media

Some drives report incorrect image sizes when -RW/+RW/-RAM media are used.
Two common cases are:

Problem 1)   The drive reports the size of the largest image ever written to the medium, not that of the actual image.

Symptoms: After erasing a medium it is written with a file sized about 100MiB. But the image read back is several GiB long and contains the remainings of older images.

Problem 2)   The drive reports the maximum possible medium capacity (typically 2295104 sectors) instead of the number of actually used sectors.

Symptoms: When reading beyond a certain point of the medium, only read errors occur, but all files on the medium are still readable and complete.

Possible remedy: Activate the option for determining the image size from the ISO/UDF file system or from the ECC (RS02/RS03) data.

If the required ISO/UDF sectors are unreadable and you are using error correction files to recover damaged media there are two possible workarounds:

- Execute the "Verify" function with only the error correction file being selected/given. Note down the correct image size from the output and restrict the reading range accordingly.

- Simply read in the image with the incorrect (larger) size. When invoking the "Fix" function, answer "OK" when you are asked whether the image should be truncated.

Q 3.5: My self-written media is recognized as "DVD-ROM" and rejected.

The medium book type has probably been set to "DVD-ROM". Typically, a drive capable of writing the same media format is required for processing the medium with dvdisaster.

For example, a two-layered DVD+R with a wrong book type may only be accepted on a writer which can write to such media. Try another drive for reading images in these cases.

Q 3.6: No drives appear under FreeBSD or NetBSD.

You need read *and* write permission for the respective device (like /dev/pass1 on FreeBSD or /dev/rcd0d on NetBSD).
Please see the respective hints in the INSTALL file which is included in the source code archive.

Q 3.7: "Ecc file has been created with version 0.40.7."

Some developer versions of dvdisaster mark their ecc files with a special bit. This causes dvdisaster versions up to 0.65 to falsely display the above error message. Please use the developer versions only together with dvdisaster 0.66 or newer versions.

5   Background information

The information in this sub section is not required for operating dvdisaster. However it is helpful for understanding how dvdisaster works and may help you getting the most out of the program according to your needs.

5.1   Properties of the error correction

This sub section outlines the basic ideas behind dvdisaster so that you can see for yourself whether it meets your demands on data safety. If in doubt, you should not use dvdisaster as a single means of protecting your data and deploy additional data backup strategies.

Method of error correction.   dvdisaster uses a Reed-Solomon code together with an error correction algorithm optimized for the treatment of erasures. The implementation draws a lot of inspiration and program code from the excellent Reed-Solomon library written by Phil Karn. Using the standard settings of an error correction file 223 medium sectors are combined into one error correction code ("ECC") block. Medium read errors are regarded as "erasures"; therefore a maximum of 32 bad medium sectors[7] are correctable per ECC block.
The 223 sectors are selected so that they are evenly distributed over the medium surface. Therefore large contigous areas of defective sectors can be corrected before the threshold of 32 defects per ECC block is reached. This kind of error pattern is especially common for an aging medium where the outer area is starting to degenerate, and for scratches along the data spiral.
On the other hand, radial or diagonal scratches are expected to be correctable in the optical disc drive itself. If not, the employed error correction strategy performs rather neutral in these cases (neither especially good nor extraordinary bad).

Limits of error correction.   In the wost case, 33 defective sectors (for the standard setting) are sufficient to prevent a full data restoration. However to achieve this effect, the errors have to be distributed over the medium in such a way that they occur in the same ECC block - such patterns are very unlikely. Empirical tests have shown that on aging media about 10% of the overall sector count may become defective before the threshold of 33 defects per ECC block is reached. Scratches will cause the threshold to be reached earlier, therefore it is recommended to visually check the media in regular intervals. Media with read errors caused by scratches should be replaced immediately.

Hardware limits.   Most drives will not recognize media when the lead-in area before the first sector (near the center hole) is damaged. In such cases, dvdisaster will not be able to recover any content from the media.
It is not feasible to enhance the reliability of poor quality media by using dvdisaster. Cheap media can decay within a few days to an extent which will exceed the capabilities of the error correction code.

---

[7]The given threshold of 32 correctable errors per ECC block is the standard setting. It is possible to select other values for higher or lower error correction capabilities. When error correction data is put directly on the medium, the number of correctable errors depends on the free space on the medium.

5.2   Image level data recovery

Media recovery with error correcting codes takes place in two steps:

1. First as much data as possible is read from the defective medium.

2. Then the still missing data is recovered using the error correction code.

The amount of readable data (step 1) does not only depend on the reading capabilities of the drive, but also on which logical level the reading process takes place. This page discusses the logical levels and explains why dvdisaster uses image level reading. Please see also Q1.5 for a visualization of this concept.

Logical levels of a medium.   Optical discs are organized into *data sectors* containing 2048 bytes each. Consecutively reading and storing these sectors produces a medium *image*.
But working with single data sectors is unwieldy from a users perspective. Therefore media contain *file systems* which combine data sectors into *files*. This requires accurate book-keeping about which data sectors each file is comprised of, and also further attributes like file names and access permissions. For this book-keeping some data sectors are reserved on the medium and filled with respective data structures.
As a consequence media contain different *logical levels*: Viewing the medium as a sequence of data sectors means working at the *image level*. However looking at the medium as a collection of files is the perspective of the *file(system) level*.
The two levels have different properties when it comes to data recovery:

Shortcomings of reading a medium at the file level.   Reading a defective medium at the file level means trying to read as much data as possible from each file.
But a problem arises when data sectors are damaged which have book-keeping functions in the file system. The list of files on the medium may be truncated. Or the mapping of data sectors to files is incomplete. Therefore files or parts from files may be lost even though the respective data sectors would still be readable by the hardware. This is very bad since even small readable portions of damaged files are valuable for the error correcting code.
An extremely bad case occurs when the error correction data is also stored in files. Then the error correction data is required to repair the file system, but the defective file system prevents access to the error correction data. That means total data loss and rises some issues on the treatment of error correction files (more on that later in this sub section).
A similar problem arises when files are augmented with PAR2 error correction data and then all these files are stored on the same medium. If the internal bookkeeping of the file system becomes damaged, all files and therefore the error correction data become inaccessible and are therefore lost.
However the situation improves greatly when employing an image-based approach:

Advantages of reading at the image level.   Reading at the image level uses direct communication with the drive firmware to access the data sectors.
The number of readable sectors depends only on the reading capabilities of the drive, but not on the state of the file system. A read error in one sector does not block access to other data

sectors. Since *all* sectors are recovered which are still readable by the hardware, this method provides the best foundation for the error correction.

Another consideration is that the image contains all data sectors of the medium. When the image is completely recovered the file system stored within is also fully repaired. A protection at the image level is therefore more extensive than error correction at the file level.

dvdisaster is working exclusively at the image level to take advantage of these properties. The RS02 and RS03 methods even allow for storing the error correction data at the same medium. This is possible since reading the error correction information at the image level can not be blocked by errors at other medium locations (damaged sectors with error correction data will reduce the error recovering capacity, but not make recovery impossible).

The RS01 method protects media at the image level, too, but stores the error correction data in files. The next section hints at some pitfalls arising from that.

Consequences for error correction file storage.   The error correction data created by dvdisaster protects media at the image level. But how are the error correction files protected?

Since error correction files are read at the file level they are subject to the problems discussed above. If the medium containing the error correction files becomes damaged it may not be possible to access or read them completely.

Therefore it is important to protect error correction files at the image level as well: Media containing error correction files must be protected with dvdisaster, too.

Since image level protection is assumed there is *no further damage protection* contained in the RS01 error correction files! This would not help much, anyways: Error correction files could be created in a way that allows them to provide a reduced data recovering capacity even when being damaged[8]. But however such internal protection would be designed, the error correction file would still be only protected at the file system level with all the disadvantages discussed above!

In addition, the computing time and redundancy used for internal protection is better spent at the image level: The Reed-Solomon error correction works best when error correction information is spread over huge amounts of data. It is better to protect the image as a whole than individually protecting each file within.

5.3   The RS01, RS02 and RS03 methods

dvdisaster contains three error correction methods. RS01 and RS02 have been available right from the start of the project. RS03 has been added recently to take advantage of recent hardware, and to implement improvements learned from evaluating its predecessors.

Comparison of the methods.   All three methods build on the same Reed-Solomon error correction. They calculate error correction information for ISO images which is used to recover unreadable sectors if the disc becomes damaged afterwards.

---

[8]Error correction files created with the new RS03 codec have exactly these properties, e.g. they are robust against damage as long as there is enough error correction capacity left. This is possible as there is sufficient computing power and fast hard drive access available now to create a suitable file format. At the time RS01 was developed these resources were not yet available. But it must be stressed that even for RS03 error correction files the disadvantages of reading at the file level hold!

The methods differ in the way the error correction information is stored:

- RS01 creates **error correction files** which are stored separately from the image they belong to. Since data protection at the file level is difficult, error correction files must be stored on media which are protected against data loss by dvdisaster, too.

- To apply the RS02 method an image is first created on hard disk using a optical disc authoring (or writing) software. Before the image is written on the medium, dvdisaster is used to **augment the image** with error correction data. Therefore the data to be protected and the error correction information are located at the same medium. Damaged sectors in the error correction information reduce the data recovering capacity, but do not make recovery impossible - a second medium for keeping or protecting the error correction information is not required.

RS03 is a further development of RS01 and RS02. It can create both error correction files and augmented images:

- RS03 can distribute work over multiple processor cores and is therefore much faster than RS01/RS02 on modern hardware.

- RS03 **error correction files** are - contrary to RS01 - robust against damage. This should not delude you into careless handling of your error correction files though - the disadvantages of reading at the filesystem level are still valid.

- RS03 **augmented images** do not require so-called master blocks holding important information. This makes RS03 a bit more robust, but also more restrictive: The augmented image must completely fill the medium now while the size of augmented images can be freely chosen in RS02.

The changes for parallel computation and higher robustness make RS03 a bit less space efficient, e.g. RS03 error correction data has slighly less error correction capacity than its RS01/RS02 counterparts on images with equal size.

Comparison of error correction storage.   The following table summarizes the differences between error correction files (RS01, RS03) and augmented images (RS02, RS03):

Error correction files	*Image augmented with error correction data*
any possible redundancy can be chosen	redundancy is limited by free space on the medium (= medium capacity - size of data image)
already effective at 15% redundancy since error correction files are required to be free of damage	requires more redundancy (recommended: 20-30%) to compensate defects in the error correction data
medium can be completely filled with data	usable medium capacity is reduced by amount of error correction data
can be created for already existing media	only applicable before writing the new medium since the image must be augmented with error correction information in advance
separately storing the error correction file from user data strengthens data protection	common storage of user data and error correction data may reduce error correction capacity
mapping between error correction files and media must be kept; error correction files must be protected against damage	easy one-medium solution; error correction information needs not to be cataloged or explicitly protected
no compatibility issues with play-back units	media with augmented images may not play correctly on all units

Further reading.   The specification for the dvdisaster codecs is named *codecs.pdf* (dubbed "the blue manual"). It is included in the source distribution. Good knowledge in coding theory and programming is required.

5.4   The linear reading strategy

dvdisaster contains two different reading strategies.

The linear reading strategy is recommended for:

- Creating images from undamaged media, e.g. to generate the error correction file; and

- Scanning the medium for reading speed and read errors.

The adaptive reading strategy is recommended for:

- Extracting data from damaged media.

Properties of the linear reading strategy.

Optical media are organized into sectors which are continously numbered beginning with zero. Each sector contains 2048 bytes of data.
The linear reading strategy reads the medium from the start (sector 0) until the end (last sector). The reading speed is shown graphically to provide information about the medium quality (see figure 68 for an example and the explanations in the following paragraphs).
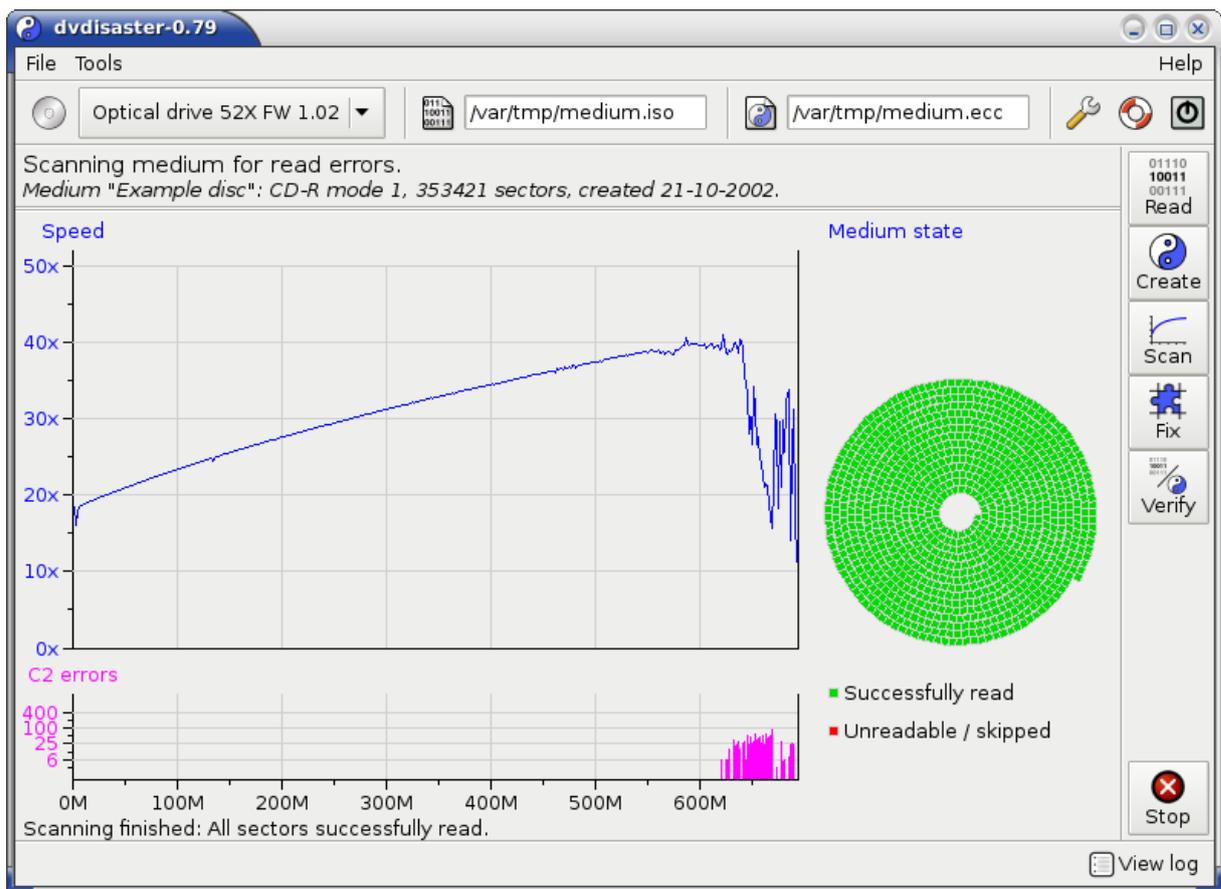
Figure 68: Linear reading of a weak but fully readable CD.

Configuration.

Number of sectors to skip after a read error.   Reading attempts for defective sectors are slow and may wear out the drive mechanics under unfavourable conditions. A series of read errors, spread over a continous sector range, is more common than single spot defects. Therefore a configuration option exists so that a certain number of sectors will be skipped after the occurance of a read error. The skipped sectors are assumed to be defective without further reading attempts. Some guide lines for selecting the number of skipped sectors are:

- Skipping a large number of sectors (e.g. **1024**) gives a quick overview of the degree of damage, but will usually not collect enough data for repairing the medium image.

- Smaller values like **16, 32 or 64** are a good trade-off: The processing time will be considerably shortened, but still enough data for repairing the image is usually collected.

- On DVD and BD media read errors do usually extend over at least 16 sectors for technical reasons. Therefore a sector skip less than 16 is not recommended for such media.

Limiting the reading range.   Reading can be limited to a given range of sectors in the "Image" preferences tab. This is helpful during multiple read attempts of damaged media.

Estimating media quality.

The speed curve.   Most drives will slow down while reading medium areas which are in bad condition:

- Drops in the reading speed can be a warning for an imminent medium failure.

- However some drives will read with full speed until the bitter end. With such drives media deterioration may not show up in the reading curve until actual read errors occur.

The reading curve is most accurate when using the "Scan" function. During the "Read" operation the read data will be written to the hard drive at the same time, which may cause irregularities in the reading curve depending on the operating system and hardware used.

Read errors.   Read errors cause red markings in the spiral or respective messages at the command line. This means that the medium could not be read at these places during the current reading pass:

- The medium is most likely defective.

- The image should be repaired as soon as possible and then be transferred to a new medium.

5.5   The adaptive reading strategy

Please note: Adaptive reading is unavailable in the current version of dvdisaster as it requires rework for RS03. It will be re-introuced soon. It is documented here for completeness:

The adaptive reading strategy is recommended for:

- Extracting data from damaged media,

The linear reading strategy is recommended for:

- Creating images from undamaged media, e.g. to generate the error correction file.

- Scanning the medium for reading speed and read errors.

Properties of the adaptive reading strategy.

The adaptive reading strategy uses a "divide and conquer" approach for locating still readable portions of a damaged medium. The strategy is based upon an article published by Harald Bögeholz in c't-Magazin 16/2005 where it was published together with the program *h2cdimage*:

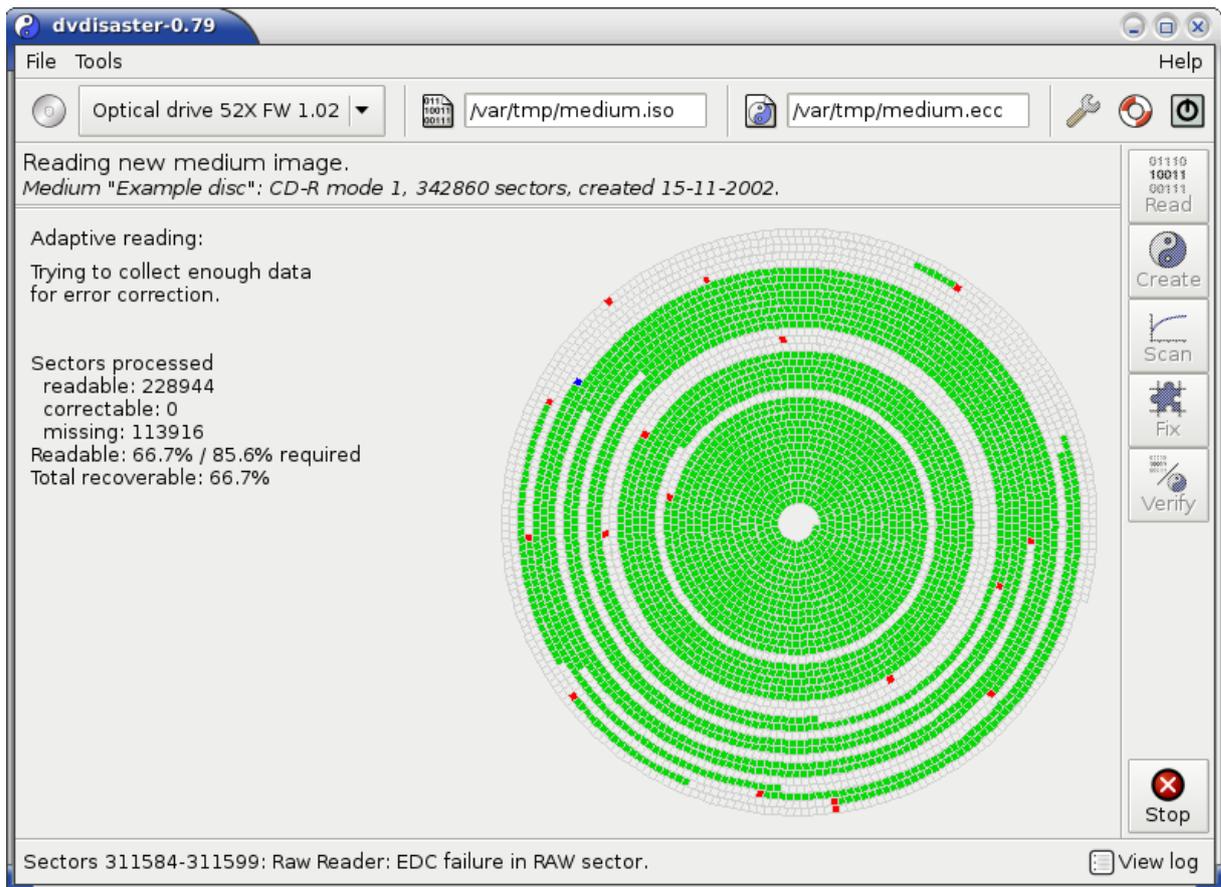1. At the beginning the medium is considered as a single unread interval. Reading begins with sector zero.

Figure 69: Progress of th adaptive reading.

2. The reading process continues sequentially unless either the end of the current interval or a read error is encountered.

3. The reading process is terminated if either (3a) sufficient sectors for a successful error correction have been read or (3b) no unreadable intervals exceeding a given size remain.

4. Otherwise the largest remaining unread interval will be determined. Reading continues in the middle (e.g. second half) of this interval; the first half of this interval is kept for a later reading pass.

The termination criterium (3a) is especially efficient: Reading will stop as soon as enough sectors have been collected for a successful image recovery using the error correction file. This can reduce the reading time by as much as 90 percent compared with a full read attempt, but does of course only work when error correction data is available.

Configuration

Error correction file. Adaptive reading works best when error correction data is available. Obviously the error correction data must have been created at a time where the medium was

still fully readable. To use a error correction file during adaptive reading, enter its name before starting the reading process.

Limiting the adaptive reading range.   Reading can be limited using the "Image" preferences to a part of the medium. This is not recommended when error correction data is used since the limit might prevent sectors from being read which are required for a succesful error correction. If no error correction data is available, limiting the reading range might be helpful during multiple reading attempts.

Early reading termination.   If no error correction data is available, adaptive reading will stop when no unread intervals larger than a selectable size remain.
The termination value should not be smaller than 128. Otherwise the laser head will have to carry out lots of repositioning moves during the final phase of the reading process. This will negatively affect both the life expectancy of the drive and its reading capability. A better approach is to stop adaptive reading earlier and then try reading the remaining sectors with an additional linear reading pass.

5.6   Remarks on read errors

Optical media have their own error correction code which protects the data against small manufacturing errors and inaccuracies during writing. If the writer and medium are compatible and of high quality, the error correction built into the medium will at first be mainly idle. This leaves enough reserves to compensate normal wear and aging effects during many years of the medium usage.

When the capacity of the built-in error correction is finally exhausted, read errors will start to appear on the medium. These will be reported by the "Scan"-operation of dvdisaster. Depending on the time of first occurrence, two types of read errors are of particular interest:

Read errors appearing right after writing the medium.   This is a sign of:

- media from a faulty production run, or

- media which are not compatible with the writer.

A prudential choice is to dispose of the faulty media and to write the data on error-free media, possibly switching to a different producer.

Please withstand the temptation of trying to preserve the faulty media by means of an error correction file - this will most likely end with data loss.

Read errors after a few months/years.   The built-in error correction of the medium will be increasingly loaded during its life time until it finally fails and read errors show up. This happens for mechanical reasons (scratches, warping of the plastic material) as well as for chemical causes (decaying dye and/or reflective layer).

These effects typically occur while the medium is stored away for a few months, and it may not be possible to read in all sectors afterwards.

Therefore it is crucial to create the error correction data in time. The ecc data contains information for recalculating the contents of missing sectors (within certain limits). Therefore with the help of the ecc data dvdisaster can recover images even if not all sectors could actually be read by the drive.

Since the error correction can reconstruct missing sectors up to a certain number, it is not necessary to squeeze out a defective medium for every readable sector. The adaptive reading strategy checks during reading whether enough data for error correction has been collected. As soon as this is the case, reading stops and still unread sectors will be recovered using the ecc data.

Some hints for effectively reading damaged media

The outcome from reading damaged media depends on several factors:

- **Not all drives are built the same.**

  Different drives have different reading capabilities. Take advantage of dvdisaster's function for completing an image with several reading passes and use different drives for each pass. Transfer the image file between computers using a network or removable media in order to use drives installed in different machines.

- **Eject and insert the medium again.**

  Sometimes it makes a difference to eject the medium, turn it about a quarter, and then load it again for another reading pass.

- **Some drives read better while being cold.**

  Turn off the computer over night and perform another reading attempt in the next morning.

  But note: "Cold" refers to normal living room conditions - putting hardware or media into the fridge can be very unhealthy for them.

5.7   Hints for storing the error correction files

Currently there are few (if any) exchangeable media technologies which can be a cost-effective alternative to the various optical disc formats. So you will probably not only use optical discs for data archival, but store the respective error correction files on CD, DVD or BD as well.

There is nothing wrong with that, but bear in mind that your archived data and the error correction files are stored on media with the same degree of reliability. When read errors occur on the archived data, be prepared that the disc with the respective error correction file might have aged beyond full readability, too.

Therefore it is important to protect your error correction files with the same care as your other data[9]. This is best achieved by integrating the error correction files into your normal data backup scheme. Here are two ideas:

---

[9]You might also choose an augmented image using RS02 or RS03 instead of creating an error correction file.

1. Storing the error correction files on dedicated media:

If you decide to store error correction files on separate media, it is important to protect those media with dvdisaster as well. To avoid a never-ending chain (error correction files for media of error correction files for ...), try the following:

Lets assume that five error correction files can be stored at each medium. Write the first five error correction files to the first medium and create another error correction file for that medium. Now save that error correction file together with four other error correction files on the second medium. If you continue that way, all error correction files except for those from the last medium (which may still be kept on hard disk) are protected by dvdisaster.

2. Putting the error correction file on the next medium of a series:

If you do not fill your media to the max (e.g. with less than 4GiB for a single layered DVD), you can store the error correction file of one medium on the succeeding medium within a series.

6 Reporting defects (aka bugs)

Like all complex software, dvdisaster may contain defects (programming errors) and incompatibilities with certain (drive) hardware and software setups. You are invited to tell us about any difficulties you encounter with the program or the documentation so that we can improve things in future releases.

To make sure that we are getting the right information, we have provided the following checklist for defect reporting.

Please check first that you are really experiencing a defect:

- Make sure that you are using the latest genuine version from our download site. dvdisaster versions provided by third parties may contain functions and defects which are not present in the original version (and we can't fix problems introduced by third parties).

  Please do also note that the dvdisaster project does no longer make and publish versions for the Windows and Mac OS operating systems. Reported defects for those platforms will not be processed.

- Double check that the issue you have encountered is not already covered in the Questions and Answers section.

- Please note that dvdisaster will only work with the (re-)writeable varieties of media, so seeing it **reject DVD-ROM and BD-ROM** is not an error. Also, CD-Audio, VCD, SVCD and multisession CD are not supported as well as all HD-DVD formats (complete list of supported media formats).

- dvdisaster works only with real optical drives. Not supported are network drives, software drives (e.g. alcohol) and drives in virtual machines.

How to report issues with the program:

Please report your findings by sending an email to *carsten@dvdisaster.org*. Your report should contain:

- Information about the operating system and dvdisaster version you are using;

- the drive and media type(s) which exhibited the problem;

- a textual description of the issue you encountered;

- a screen shot of the error message and/or output which might provide further information about the problem;

- differences between working and non-working configurations if the issue is experienced only on certain drives/computers;

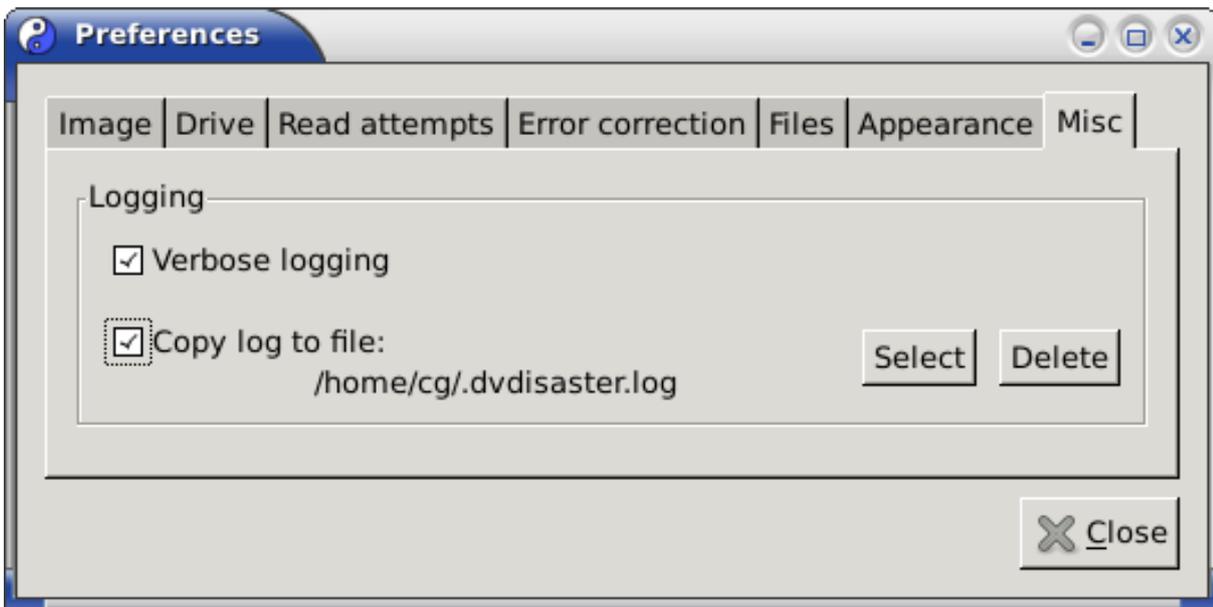- a log file if you suspect that the issue is related to a drive or medium incompatibility.

Figure 70: Creating a log file.

How to create a log file:

If you suspect incompatibilities with your drive and/or media as the cause of your issue, please activate the log file feature in the preferences dialog as shown in the screen shot. Then perform a scanning or reading action and attach the log file to your bug report.

Thanks for your feedback!