RDB-Salv

Version 1.00 Rev. 1 of 28-May-1996

Angela Schmidt, Translation by Bernd Felsche

Note

These conditions shall be interpreted according to the laws of Germany. The German text of these conditions shall take precedence over any translation thereof for the purposes of legal interpretation.

Copyright ©

Disclaimer

Angela Schmidt shall not be held responsible for any damages or losses, direct or consequential, resulting from the use, or inability to use the software. This applies even if Angela Schmidt has been made aware of the possibility of losses or damage.

Alterations

RDB-Salv shall *not* be altered (patched). Those who do this anyway, should not be surprised by extremely uncomfortable side-effects. You are *explicitly* warned against removal of the registration number requester.
Of course, RDB-Salv shall *not* be distributed if it has been altered — even when the altered software was based on the freely distributable version.
If a modification is thought to be useful, it's worth your while to contact the Author, who may after all have the desired function in the next version.

Trademarks

This documentation mentions various hardware and software by name. Such names are often protected Trademarks and their mention in this document shall in no way damage their legal status.

Licence Agreement

This Agreement is a legal contract between you, the end user, and Angela Schmidt. You agree to accept the conditions of this contract by use of the software.
The *registered* version with a valid registration number entitles you to use a *single* copy of the software on *one* computer (i.e. at only *one* location for *one* unit)
If you use the *unregistered* version of RDB-Salv, you can use as many copies as you like simultaneously, and distribute copies to as many people as you like at no charge. Copy fees are prohibited — exceptions must be obtained in writing from the author. Distribution via the following CD-ROMs is hereby expressly permitted: Meeting Pearls, Gateway!.
An *unregistered* version of RDB-Salv can be recognized by the startup requester for registration number and user address. You are dealing with an *unregistered* version when *all* fields are empty (no default values). It is safest to only ever pass on the original archive — nothing can go wrong that way.

Registration

Please read more about registration in the AmigaGuide document on RDB-Salv

Author

You can contact me at the following addresses:

EMail: Angela.Schmidt@stud.uni-karlsruhe.de

http: http://home.pages.de/~Angela/

Snailmail (for as long as I'm still studying, at least until end 1996)
Angela Schmidt
Klosterweg 28/I501
76131 Karlsruhe
Germany
Tel.: +49 721 695307 (only at a *civil* hour!)

Index

1 General

This chapter explains why you need RDB-Salv, how it works, its system requirements, and how to install it.

Please aquaint yourself with these points before using RDB-Salv.

Note that these instructions include an *extensive* glossary. If you are unsure of a particular term in the technical jargon, you should check the glossary first, to see if it's explained in there.

Terminology which is adequately explained in you system's manuals has not been included in the glossary. If you are new to, and/or not familiar with your AMIGA, you are urged to keep its manuals at hand.

1.1 Introduction

Ever had a checksum error or similar in one of your partitions? You've probably used something like Dave Haynie's *DiskSalv* to salvage the partition. Maybe you had a backup of the affected partition, which you could use to fix the problem.

Now imagine that you lose *all* partitions on a disk drive at once. If you're lucky, you have a current backup of all the data, but as Murphy would have it, that is seldom the case. If you don't have a tool which can recover your data, you may have destroyed days, weeks; even months of work.

This is by no means just a ficticious horror story — it's happened to many people in the past. Their *RigidDiskBlock* (abbreviated as *RDB* or *RDSK*) or similar *Management Block* (e.g. *Partition Blocks*) has been overwritten, erased or damaged, preventing access to all data on the disk, even though the data on the disk is otherwise undamaged.

Many have undoubtedly accepted defeat in such situations and repartitioned and initialised their hard disks, thus losing all data. Others have desparately sought to reconstruct the RDB (and the associated partition blocks) by guessing partition information. But even after many hours of reconstruction, the work has not usually resulted in success, mainly because there are simply too many combinations to try. The loss of the RDB (and/or other important Management Blocks) has in the past been associated with the total loss of data on the disk.

RDB-Salv puts an end to that as of right now. RDB-Salv is a professional tool which will help you to recover your data in such situations. It's easy to use and besides offering the ability to salvage partition data, it provides functions for backing up and restoring the RDB and associated Management Blocks. It can't be any easier or more secure.

1.2 Background/Functional Description

Most hard disks are divided into *Partitions* for example into a *System-Partition* and a *Work-Partition*. Data can be accessed whilst the system knows the size and location of the partitions on the hard disk. If this information is lost, then so is all the data on the disk. However, the data can be accessed again as soon as the system is informed of the size and location of partitions.

When the system is intact, this *Partition Data* is usually stored in the first few blocks of the hard disk. It would be very difficult to recover any data from the hard disk, without an appropriate tool, if these blocks are destroyed through an operational error, a buggy program or even a virus.

The first 16 blocks of a hard disk contain the *RDB* (aka *RDSK*), which contains a few global values, as well as pointers to *partition blocks* and other management blocks. The partition blocks are especially important as they contain all the data defining a partition.

RDB-Salv offers several options to prevent you suffering from an "RDB-Mishap" in future:

- RDB-Salv provides means to make a backup copy of the RDB and associated blocks and to perform a subsequent restore when necessary from the backup copy. This is preferable to salvage operations, even with the aid of RDB-Salv.
- Sometimes, only parts of the RDB and management blocks are over-written. RDB-Salv can scan your disk for management blocks which are no longer recognized by your system, and salvage the lost data.
- If you haven't rebooted since the loss of the RDB i.e. all partitions are still visible, then RDB-Salv can read the data from the *DeviceList* from RAM and thus recover the necessary information. A reboot or power off makes this method of recovery impossible.
- Even when both of the above methods fail, you needn't despair. RDB-Salv can scan your entire hard disk and calculate parameters for all your AmigaDOS partitions. Some values, which are fortunately not critical should be restored manually as their automatic recovery is simply not possible. Amongst other values, this includes the names of the salvaged partitions (e.g. '`DH0:`', '`HD0:`', '`wb_3.x:`', ...).

RDB-Salv will not change the contents of your hard disk — except when you confirm its actions in the requester which appears. You need not worry about RDB-Salv destroying something without it asking you to confirm its actions first.

When you wish to restore a partially or completely destroyed *RigidDiskBlock*, RDB-Salv will search for or calculate many parameters which will allow the afflicted partitions to once again be addressed. You can then either store these directly as RDB and partition block, or opt to enter them manually into a *Prep program* (e.g. *HDToolBox*). Additionally, you can store the values as a *MountList* or mount the devices.

1.3 Requirements

RDB-Salv requires OS 2.04 or later. RDB-Salv will speak to you in your own language as of OS 2.1, and OS 3.0 and later provides for enhancements relating to the GUI.

Of course you'll need a hard disk! The controller/host adaptor must not have too many errors. RDB-Salv goes to great lengths to work around *device driver* problems.

A significant *Limitation* must be stated now: RDB-Salv can only find AMIGA partitions. AFS, NetBSD and Mac partitions won't be found. The same goes for other unrecognizable partition types. However, RDB-Salv will in these cases, detect unused partitions and report them as such, so that it's possible for you to guess which of the unrecognizable partitions is which.

1.4 Installation

Please use the *Installer script* to install RDB-Salv. The script has been developed to accomodate various system configurations and to set corresponding *Tooltypes* for RDB-Salv's icon.

Click on the Installer icon and follow the directions. If you're unsure of anything, use the Help button to obtain helpful information.

Some manual adjustments may be necessary as it's not possible to know all possible configurations of *device drivers* and to probe for them in the Installer script. The Installer script will advise you when it can't recognize your *device driver*. You should then proceed as follows:

1. Click *once* on the newly-installed RDB-Salv icon.
2. Choose *Information* ... from the Workbench Icons menu.
3. Edit the following Tooltypes according to the description in Section 1.5 [Tooltypes], page 6. If you are unsure of how to do this, consult your Workbench manual.
   - `BUFMEMTYPE`
   - `NOSCSICMD`
   - `NOGETGEOMETRY`
   - `NONEWSTYLE`
4. Click on *Save*. All done!

Another few Tooltypes should be adjusted subsequently — as per the description in Section 1.5 [Tooltypes], page 6. The Installer script is unfortunately unable to determine these values automatically.

- `MAXTRANSFER`
- `MASK`

You can now start RDB-Salv. Simply double-click on the RDB-Salv icon. A window requesting your registration number and address will appear after a little while.

If you're registered, enter the requested data and click on *Save*. This personalizes RDB-Salv for you so that you don't always have to re-enter your registration number. This personalized version should *never* be passed on.

If you don't wish to modify RDB-Salv at this time, even though you have registered, click on *Use* after entering your registration number. All program functions will then be available, without actually changing RDB-Salv.

Without registration, you have no option but to click on *Cancel*. This limits the functionality of RDB-Salv but you are most welcome to explore and test to see if RDB-Salv meets your needs.

When you have completed the information in the requester, RDB-Salv starts and lets you get on with your work.

1.5 Tooltypes

A list of Tootypes recognized by RDB-Salv follows. Values appropriate to the computer should be set before starting RDB-Salv.

`DEVICE=<default-devicename>`
: The device driver normally used, e.g. '`scsi.device`'. This value can be easily changed later from within the GUI. Please refer to your hard disk system manual to determine a suitable value.

`UNIT=<default-unit>`
: *Device unit* normally used, e.g. '`0`' for Unit 0. This value can be easily changed later from within the GUI. Please refer to your hard disk system manual to determine a suitable value.

`BACKUPFILENAME=<default-backupfilename>`
: Default filename for the backup copy, e.g. '`DF0:RDB-Backup`'. This value can be easily changed later from within the GUI.

`BUFMEMTYPE=<bufmemtype>`
: Memory type from which to use for device access. This value only makes sense when `NOGETGEOMETRY` has been set or if the device driver does not support `TD_GETGEOMETRY`. `TD_GETGEOMETRY` if available defines the type of memory and this will *always* be used. Please refer to your hard disk system manual to determine a suitable value. If your manual doesn't help, then it's best to use `0x00000001` which corresponds to `MEMF_PUBLIC`.

`MAXTRANSFER`
: Smallest MaxTransfer value for all disks. This value is used by RDB-Salv and passed on to potential file systems. Please refer to your hard disk system manual to

determine a suitable value. One is usually safe to specify `0x0000FE00` — although this has an unfortunate side-effect of significantly reduced transfer rates over higher values. Values which are too high can lead to data transfer problems. If in doubt, try it out. If you are unsure, then sacrifice speed for reliability and specify the `0x0000FE00` value.

`MASK`
: Desired value for Mask. The value will be passed on to potential file systems. Please refer to your hard disk system manual to determine a suitable value. If that doesn't help, try it with `0xFFFFFFFE`. That value is now used by most systems. RDB-Salv only does data transfers to memory which has been obtained from `AllocMem()`, and doesn't validate that this agrees with `MASK`.

`NOGETGEOMETRY`
: Prevent despatch of `TD_GETGEOMETRY`. You *must* set this option if your device driver crashes when it gets a `TD_GETGEOMETRY`. If your device driver doesn't support `TD_GETGEOMETRY`, you should set this option to avoid unnerving Alerts. Consider upgrading your device driver anyway.

The following symptoms indicate that your device driver has trouble with `TD_GETGEOMETRY`:

- You get a warning requester when you first press *Start*, telling you that `TD_GETGEOMETRY` isn't working.
- Your machine crashes after your first click on *Start*.

`NONEWSTYLE`
: Don't use New-Style commands. You *must* set this if your device driver crashes when it gets a `NSCMD_DEVICEQUERY`. Do not set this option if your device driver simply doesn't implement this command (applies to almost all drivers as it is very new).

You can tell if your device driver has problems with `NSCMD_DEVICEQUERY` because your computer will crash when you first click on *Start*.

`NOSCSICMD`
: Prevent despatch of `HDSCSICMD`. If your device driver crashes when it gets a `HDSCSICMD`, then you *must* set this option. Should your device driver not support `HDSCSICMD`, you can set this option to avoid annoying alerts. You should consider a device driver update anyway.

The following symptoms indicate that your device driver has trouble with `HDSCSICMD`:

- You get a requester telling you that `HDSCSICMD` isn't working; probably preceded by another that says that `TD_GETGEOMETRY` isn't working either.
- Your computer crashes after clicking on *Start* for the first time, possibly preceded by a warning about `TD_GETGEOMETRY` not working.

`SCSIREADWRITE`
Use SCSI-Direct commands (`HDSCSICMD`) to read and write hard disk blocks. This option is not allowed if `NOSCSICMD` is set. This will become significant in future — when hard disks larger than 4 GB are the rule, rather than the exception — as it allows addressing of SCSI disks larger than 4 GB.

`BADDEVICE1`
This option should not normally be set. It is only to be set if the return code from attempting to read non-existent blocks is inconsistent. This is a hack to allow for bad device drivers.

`NORECOVER`
Switch off *Find lost blocks* when `NORECOVER` is set. This value can be easily changed later from within the GUI. As an alternative, a Tooltype can by the way be set to `RECOVER=TRUE|YES|1` to explicitly switch *on Find lost blocks.*

`READDEVLIST`
The *Read* function will default to *Read from DeviceList* when this is set — *Read from Media* otherwise. This value can be easily changed later from within the GUI.

`OPTIMIZE` Activate *Optimize* for the *Backup* function. Storage of the management blocks will be optimized to reduce space requirements. Unused blocks which may still contain lost data will not be backed up. This value can be easily changed later from within the GUI.

`DEBUG` Debug mode — should *not* normally be switched on.

If you're an expert and prefer to use the shell, then these Tooltypes can be specified as parameters to the program — unless otherwise specified.

2 Using RDB-Salv

Having successfully installed RDB-Salv and having gained some understanding of the underlying principles, you can being using RDB-Salv. This chapter outlines points to note while using this program.

2.1 Starting RDB-Salv

RDB-Salv is started from the Workbench by double-clicking on it's Icon. A window will open, presenting the main menu. This lets you choose between multiple pages of functions.

About The Information Page (*About*). Information about those who contributed to RDB-Salv is presented here.

Backup The *Backup* Page. Backups of the RDB and associated management block can be created with the aid of this page.

Restore The *Restore* Page. This page is needed to write the backed-up RDB and management blocks back to the hard disk.

Salvage The *Salvage* Page. Un-corrupted data, still on your hard disk is used to try to re-establish the RDB and associated management blocks using the functions on this page. Salvage attempts meet with a success rate of well over 90 %, though 100 % can never be guaranteed.

Read The *Read* Page. All available RDB and management blocks are read and displayed using this page. It's also possible to search for long-lost RDB and management block data. For example; any backup copies which have been saved to the hard disk *against* the recommendation to save them to floppies.

Click on one of the five gadgets in the main menu to select the page you wish to use.

2.2 Exiting from RDB-Salv

Sooner or later, you'll wish to quit from RDB-Salv. There are several possibilities:

1. The close Gadget.
2. The Quit menu option.
3. The `AMIGA-Q` keyboard shortcut.
4. The *Quit* button which appears on several of the pages.

Note that *no* confirmation will be required as this is, more often than not, an annoyance.

2.3 About

You may be curious as to who contributed to RDB-Salv, who owns the Copyright ©, etc. Then this page is at your service!

2.4 Backup

An ounce of prevention is better than a pound of cure. It's highly recommended that you make backup copies of your RDB and associated management blocks. A backup copy is always preferable to any attempt to recover the data after a *disaster*. It's imperative that the backup copy is not saved on the hard disk but on one or more floppies, as it doesn't help much to have a backup RDB on a hard disk struck by disaster.

The backup copy is written as a single file. This usually fits comfortably onto a single floppy disk.

Although RDB-Salv's Section 2.6 [Salvage], page 11 function does not support all file system types, the backup function is able to backup relevant information for all file system types, including NetBSD, Mac, AFS.

To create a backup copy:

1. Format an empty floppy disk. Refer to your system manual for an exact description of how to go about this if you are unsure.
2. Label the floppy disk and insert it in a drive.
3. Click on Backup. Provide the correct information in the three text gadgets:

   Device *Device* must contain the name of the device driver of the device for which you wish to create the backup. e.g. '`scsi.device`'. Consult your hard disk controller or host adapter manual for the name of the device driver noting that its specification is case sensitive. Click on *Choose...* to see all possible device drivers. Double-click on the correct name to accept it.

   Unit This number corresponds to the *Unit* number of the device. SCSI *Unit* numbers usually range from 0 to 6. If you don't know the number off-hand, then start up the corresponding prep program (e.g. *HDToolBox*) which will give you an idea of all unit numbers. Alternatively, check your hard disk manual.

   Backup-File
   Backup-File specifies the name of the file to which the backup of the RDB and associated management blocks is to be written. If you click on *Choose...*, a filename browser will appear so that you can choose a directory and enter a filename which should be on the formatted floppy. The backup copy should never be stored on the hard disk from which the RDB originates.

4. Set the check-mark *Optimize* appropriately. When selected, the backup will contain (just about) only those block which contain directly relevant data. This can significantly reduce the size of the backup copy. However, it won't backup those lost blocks which you may be able to find using the *Read* page. Laymen are advised against selecting this option.
5. Click on the *Start* gadget to start the backup. A requester will appear when the backup has been completed successfully.
6. Just to make sure, make a copy of the floppy. Floppy disks are much less reliable than hard disks and it would be extremely annoying to discover that your backup is unusable. If need be, refer to your system manual to find out how make a copy of a floppy disk.

2.5 Restore

This page is used to restore the hard disk RDB if you suffer from a disaster some time after making a backup of the RDB and associated management blocks.

Insert your backup floppy, click on *Restore* and complete the fields corresponding to *Device*, *Unit* and *Backup-File* as described in Section 2.4 [Backup], page 10. Data will of course be read from *Backup-File* and written to the specified hard disk, and not the reverse.

Click on *Start* when ready. The RDB data on the hard disk will be destroyed and over-written from the backup copy. If you are absolutely sure that you want to do this, then confirm the action in the requester.

2.6 Salvage

If you lose your RDB and don't have a backup copy, then this page can help. Specify the device and unit as per Section 2.4 [Backup], page 10 and click on *Start*.

Your entire hard disk will now be scanned, which depending on its size, gives you enough time for a coffee break. RDB-Salv searches for recognizable data from AmigaDOS partitions and thus arrives at previous *Partition Boundaries*. The *progress indicator* shows the percentage of the scan completed. A drive name will be listed in the left window panel whenever a partition has been found. Only AmigaDOS partitions will be found.

Items can be selected from the list after the successful scan and subsequent (short) compute interval. The right window panel indicates detailed information about the selected partition.

Check the list carefully. Does it list some partitions which don't really exist? Or perhaps partitions which have long been over-written? Maybe you've even saved a *floppy disk image* to the hard disk; as this would be shown as a separate partition. Check each list entry individually and verify that it actually exists. The panel to the right shows *additional information* about each partition so that you can tell if it overlaps one or more other partitions. If two or more partitions overlap then it indicates that all but *one* of these do *not* really exist; they are so-called pseudo-partitions.

Having identified all the incorrectly-indicated partitions, select them using the mouse and click on *Remove*. Don't panic; this does not alter or erase any data on the hard disk! You're only telling RDB-Salv which partitions really exist and help it to avoid subsequent errors. If a partition does not overlap others, you will be asked to confirm its removal by a requester as it makes little sense to do so in most cases.

You must remove *all* incorrectly-indicated partitions to avoid subsequent problems. Exceptions to this are partitions named *RDSK* ... and *Unused space*. These *pseudo-partitions* serve as "place holders" and cannot usually be removed.

Under exceptional conditions, it can take several seconds to remove a partition (depending on CPU), this time being required to recalculate several values. This is the exception, rather than the rule.

Perhaps you are asking yourself why *RDSK* ... and *Unused space* appear in the list, yet not in *HDToolBox* and other prep tools. The answer is simply that prep tools do not generally allow you to use the *whole* hard disk for partitions as they silently reserve small sections for themselves. RDB-Salv indicates these sections as follows:

RDSK ... This entry encompasses the section of disk in which the RDB and associated management blocks are stored. Some space must be reserved for this — which is done by reserving space at the start of the hard disk.

Unused space
This entry identifies a "hole" in your partition data, i.e. a space where no recognizable data can be found, and where an RDSK is not usually found. The end of a hard disk will often have one of these. In such cases, the prep program has simply given away a few blocks at the end of the disk, which have now been identified by RDB-Salv.

But there is another possibility: Perhaps you have an unformatted partition. Or AFS or NetBSD partitions, similar. These will also be indicated as *Unused space*.

You should note that several of these unrecognized partitions which are *adjacent*, will only be shown as *a single Unused space*. Unfortunately, RDB-Salv is not able to identify partition boundaries within the unused space in this case. Such data cannot be salvaged using RDB-Salv.

As mentioned previously, the end of the disk often contains a *pseudo-partition* called *Unused space*. This is the only pseudo-partition which can be removed using *Remove*. If you're sure that you have no more partitions at the end of your disk, then you should remove this pseudo-partition from the list using the *Remove* button. This can avoid subsequent problems with other prep programs.

If you'd like to test if RDB-Salv has established correct data, then you can use the *Salv/Read* menu to mount partitions or to generate a *MountList*. You can also save and print all results. Please read Section 3.3.12 [Salv/Read], page 20 for details of how to do this.

If all went well, you'll probably want to set up a new RDSK. Use the *Save as RDSK and partition blocks ...* menu option to be found under the *Salv/Read* menu. Further information about this can be found under Section 3.3.12 [Salv/Read], page 20.

After writing out the salvaged RDSK, you must edit it using *HDToolBox* or similar. The procedure to follow with *HDToolBox* is detailed herein. Refer to the Appendix entitled Appendiz B [RDB-Adjustments], page 23 for a tabular outline of the exact procedure.

Start *HDToolBox* and select the damaged drive. Click on the *Partition Drive* button and select *Advanced Options*. Check and alter the following values on *every* partition.

Partition Device Name
: Enter the required *device name* here, according to what it was called before the disaster. It's not possible to determine the previous name automatically. There are often no problems at all resulting from a partition getting a new name after the salvage operation.

Buffers
: If you've specified an unusual number of *Buffers*, then enter the corresponding value here. A larger number of buffers results in greater RAM usage, less buffers result in slower access to the partition.

HostID
: Enter the correct value if your host adapter is not set to Unit 7.

Bootable
: This option must be set if you need to be able to boot from the partition.

BootPri
: Enter the Boot-Priority on all Bootable Partitions. A value of 0 is normal. If you choose a value above 4 or 5, then it may not be possible to boot from the internal drive without resorting to the Bootmenu.

Note that partitions named as *UNUSED<n>* (where <n> is replaced by some number) will require special handling as they refer to an unrecognized file system or free blocks. RDB-Salv stores all *pseudo-partitions* with such a name. You'll usually know the correct values to enter as you probably set up the partition yourself some time ago.

A small unused partition at the end of the drive indicates that you (or your *prep program*) gave away some blocks when the drive was partitioned. This is no cause for worry. If you'd removed the *Unused space* as described above, then this confusion would have been avoided. Simply ignore the partition. If you *really* want those last few kilobytes, then they can be mounted and formatted. Under *no* circumstances should you attempt to merge it with the previous partition as that would make it unreadable.

Undefined regions between partitions usually indicate an unrecognizable partition. This partition cannot be completely salvaged by RDB-Salv. It requires manual intervention on your part. Note that such regions may actually be more than one partition. You must try to remember which partition(s) are located in the region (AFS, NetBSD, Mac, ...), and to configure them using *HDToolBox*. Due to technical constraints, RDB-Salv is only able to identify and salvage real AMIGA partitions.

If you wish to use non-resident file systems, click on *Add/Update* and create a new file system entry — as detailed in the system manual. When done, return by clicking on *Ok*.

Now click on *Change...* for *every* partition. Verify if the partition should be mounted automatically by setting the *Automount this partition* option accordingly.

RDB-Salv attempts to set values so that a *useful* configuration results on as many systems as possible. This configuration is however conservative and results in noticeable performance degradation on many systems. You can change this behaviour by verifying that optimum values have been set for *MaxTransfer* and *Mask* for every partition. These values are set in the same window where *Automount this partition* was just verified.

Check your hard disk system manual for appropriate values if you are unsure. If the manual indicates better values, then they should be changed accordingly. You can save yourself some time and effort later by saving them in RDB-Salv's ToolTypes as described in Section 1.4 [Installation], page 5.

When you have corrected all values leave this page using *Ok*.

Having completed adjustments for all partitions, leave the partitioning page using *Ok* and then click on *Save Changes to Drive*.

You may wish to change some global values which have *not* been changed by RDB-Salv:

- BadBlockList
- DriveInit-Code
- Host Adapter Address (HostID)
- Interleave value
- Park-Cylinder
- WritePreComp
- ReducedWrite
- StepRate
- AutoParkSeconds
- Flags: Reselection, LastDisk, LastLun, LastTID, Synchronous

Please check your system manual, and your hard disk manual to know how to set these values and what they mean!

If you change any of these values, please note that the values for *Cylinders*, *Heads*, *Blocks per Track* and *Blocks per Cylinder* established by RDB-Salv, will not be destroyed. Should this be the case anyway, as with *HDToolBox* when *Change Drive Type* creates an entry and *Read Configuration* is activated, then you *must* re-enter the values calculated by RDB-Salv manually! RDB-Salv indicates these values under *Drive Parameters* for every partition. The values are, by the way, identical for all partitions on the same drive.

It's recommended that you use *HDToolBox*'s *Read Configuration* function (click on *Change Drive Type* followed by *Define New...* and *Read Configuration*) to obtain the default parameters for your disk drive. You must *always* however restore the values destroyed by this procedure as described above, otherwise your data will *still* not be available because of a defective RDB!

When you've complete the above changes, click on *Save Changes to Drive* again.

You can now dare to reboot and keep your fingers crossed...

2.7 Read

This page primarily serves to read an existing *RDB* and its associated *partition blocks* as well as to locate lost, but not yet over-written partition blocks. If the system has not been rebooted, and partitions are still mounted after loss of the *RDB-Data*, then the *DeviceList* can also be scanned for data.

If you can still see data from apparently lost partitions, then it's preferred that you use *this* page (*Read*) as this page can locate more *accurate* information about your old partitions. However, it's not infrequent that *Read* does not locate some or any information. In that case, you can still resort to *Salvage*.

To use this page, you must specify — as described in *Backup* — the *Device* and *Unit*.

If you have not rebooted since losing the RDB, (i.e. if all the partitions are still mounted) you can set the *Method* to *Read from DeviceList*. Otherwise, set *Method* to *Read from Media*.

If you also wish to find lost blocks while reading from media, then set the *Find lost blocks* option. If you only wish to see the current RDB and the associated partition blocks, then set *Find lost blocks* to off.

If *Find lost blocks* is set, even the space behind the RDB and associated management blocks will be scanned for old *RDB-Data* — if need be, the entire disk. This can also locate backup copies of RDBs amongst the lost partitioning information; backup copies which have been stored on the hard disk against recommendations.

Click on *Start*. The available RDB and associated partition blocks will be found and displayed very quickly. If you have selected *Find lost blocks*, then a requester appears before the scan starts to remind you that you can cancel the operation after about a minute if intend to only search for lost RDB-Data, as that always resides at the *beginning* of the hard disk.

The left panel shows any data as ghosted text, as soon as it is located. If the scan is interrupted, all data found to that point will be indicated as solid text.

You may select any of the items found, upon completion of the scan to read details of the selected item in the right-hand panel. Lost blocks can now be added to the RDSK — as described in

Section 3.3.12 [Salv/Read], page 20 — using the *Add to RDSK* menu option in the *Salv/Read* menu.

Note: If the addition of partitions leads to overlap in the RDSK, then some prep programs will become terribly confused! Take care!

3 Description of the Main Menus

Until now, only specific menu options have been described. This chapter describes all menu options in detail.

3.1 The Project Menu

The obligatory Project menu of RDB-Salv offers only two options:

3.1.1 About ...

Selecting this option results in a requester which provides details of the program version and Copyright details. If you submit a *bug report*, then the *precise* version number as shown here, must be included. If you omit the version number, it won't be possible to identify the cause of the problem.

3.1.2 Quit

This menu option terminates RDB-Salv immediately. No confirmation is required or offered.

3.2 The Settings Menu

The Settings menu allows you to load, save and set RDB-Salv's defaults.

These operations will only work if RDB-Salv has an icon as the defaults are stored there. The advantage of this is that the defaults can be set externally — and they need to be in some cases (see Section 1.4 [Installation], page 5).

If you have deleted the icon, then defaults for RDB-Salv cannot be stored there. As you'll have to start RDB-Salv from the CLI anyway, you can include the options on the command line.

3.2.1 Save settings

This option saves GUI-setable values to RDB-Salv's icon. If there is no icon, then an error message to that effect will appear.

3.2.2 Last saved

Retrieves settings from the values in RDB-Salv's icon. If the icon doesn't exist, an error message to that effect will appear.

3.2.3 Default

This option sets all GUI-setable values to default. These values are reasonable defaults for RDB-Salv.

3.3 The Salv/Read Menu

This menu becomes selectable when you are on the *Salvage* or *Read* pages. It's also necessary for you to have successfully completed a data read, or that you have completed a salvage attempt.

3.3.1 Save Results

It's likely that you'll want to store the results shown. This option performs that function, storing the values from the right-hand panel for each entry in the list in the left-hand panel.

If you haven't used this option before, a file-requester will appear, allowing you to specify the name and location of the file.

The requester will not appear on subsequent selections of this option. Data will be written directly to the same file, *without* an intervening requester for confirmation.

3.3.2 Save Results as ...

This option performs the same function as *Save Results* but *always* presents a file-requester for specifying the target file.

3.3.3 Print results

This works like *Save Results* except that results are printed instead of being saved in a file.

3.3.4 Save all MountLists as ...

If partitions have been found or read, then the data which describes the size of the partition, etc can be written as a *MountList*. A partition which has a *MountList* entry can be mounted using the *Mount* command from the CLI.

This option writes a MountList entry for *all* partitions which have been found. If selected from the *Salvage* page, then entries will have names beginning with *SALV* or *UNUSED* as the original names can no longer be retrieved, having been stored in the partition blocks, not in the partitions themselves. If selected from the *Read* page, then the entries will be created with their *real* names.

3.3.5 Mount all Devices

All partitions which have been found are mounted. The names used for mounting will be as described in *Save all MountLists as ...*.

Any errors which occur while mounting a partition will be shown in an output window.

The *Mount* CLI command must reside in the C directory for this option to work.

3.3.6 Save MountList as ...

This option is used if you only wish to save the MountList for a particular entry, not *all* of them. Naming will be as described under *Save all MountLists as ...*.

3.3.7 Mount Device

Use this option to mount a particular device, which is a frequent requirement. Naming will be as described under *Save all MountLists as ...*.

Any errors which occur while mounting a partition will be shown in an output window.

The *Mount* CLI command must reside in the C directory for this option to work.

3.3.8 Write RDSK block ...

This option writes a new RDSK block — after you confirm the action in the safety requester. The old RDSK block and associated data will be destroyed. It means that you won't be able to get to any of the partitions on the drive! Most of this data can be recovered from the *Read* page with *Find lost blocks* selected and clicking on *Start*.

This option is useful to wipe out the contents of the RDBs, perhaps so that you can rebuild it step-by-step from selected, salvaged partitions.

If you only want to salvage selected partitions, then proceed as follows:

1. *Salvage* the required partitions so that they are shown in the left-hand list.
2. Select the *Write RDSK block* ... from the *Salv/Read* menu.
3. Then, for each of the partitions which you want to salvage:
   - Select the partition in the list on the left.
   - Select option *Add partition block to RDSK* ... from the *Salv/Read* menu.

3.3.9 Save as RDSK and partition blocks ...

This option will save all the read or salvaged data as RDSK and partition blocks. Data is written to the selected drive after you confirm the action in the safety requester which appears. This is similar to what happens in *HDToolBox*, when *Save Changes to Drive* is selected.

If you choose this option from the *Salvage* page, then partitions will be saved as *SALV*<*n*> or *UNUSED*<*n*> as appropriate, where <*n*> is a number. You will see an extended safety requester, which gives you a chance to cancel the operation; and the option to save the unused space as partitions. Partition blocks will be written for every regions of unused space if you choose to take that option. You should then immediately use *HDToolBox* to adjust values on the *pseudo-partition*.

In theory, having successfully written the RDSK, you can reboot. It is however *highly* recommended that you adjust certain values using *HDToolBox* — as described under Section 2.6 [Salvage], page 11. This is especially important if this menu option has been chosen from the *Salvage* page.

3.3.10 Remove partition block from RDSK ...

This option can only be selected if you are on the *Read* page and a valid partition has been selected in the left panel.

It removes the selected partition from the RDSK after you confirm the action in the safety requester. This means that the data required to mount the partition will be erased.

The partition will *not* appear in *HDToolBox* and will not be mounted when the computer is next booted. You'd better know what you're doing!

3.3.11 Add partition block to RDSK ...

This option is available if you select a partition, not already in the RDSK, from the list in the left-hand panel. This partition will then be added to the RDSK, following your confirmation in the safety requester. If you choose this option of the *Read* page, then the original name will be used; otherwise the name *SALV*<*n*> or *UNUSED*<*n*> will be used, where <*n*> is a number.

As all partition blocks in the RDSK should have the same number of *sectors* per *cylinder*, a warning will appear if the partition being added has has an incompatible number of sectors per cylinder. It's possible to cancel the operation at this point.

3.3.12 Add all partition blocks to RDSK ...

This option works like *Add partition block to RDSK ...*. However, it automatically adds *all* partitions which do not already exist in the RDSK.

Like so many other options, partitions found with *Salvage* will be named *SALV*<*n*> or *UNUSED*<*n*> where <*n*> is a number.

Appendiz A Questions

Q: I always get *unused space* showing up after my last partition after a *Salvage* completes. I'm sure that the last partition went to the last cylinder. What gives?

A: A few blocks are often lost when the drive is partitioned. These blocks reside after the last cylinder specified by you but don't make a complete cylinder and can therefore not be used. RDB-Salv recognizes that these blocks may be used to create a small partition, and indicates it as such. The existence of the partition has a few side effects (experts should read the next paragraph). It's recommended that you remove the partition from the list by selecting it and clicking on the *Remove* button.

Experts may value this more explicit explanation. You must reduce the value of the *BlocksPerCyl*, *BlocksPerTrack* and *Heads* parameters drastically to be able to mount the partition. This has the effect of dramatically increasing the number of *cylinders*. Which is not a problem in itself, but various prep tools (*HDToolBox* included) have problems with large values.

Q: I have a *floppy disk image* stored in an 880kB file on my hard disk using FMS. Salvage finds this image. What does it mean?

A: Read Section 2.6 [Salvage], page 11.

Q: Why are salvaged partitions always called *SALV*<*n*> or *UNUSED*<*n*>? Please have the next version choose the right name like *DH0*.

A: It's simply not technically possible. You've lost your RDB and partition blocks which is unfortunately the only place where the device names are stored. Therefore, restoring the names is no longer possible. If the device name was important, then you probably know it anyway. Don't you?

Q: *Salvage* returns the wrong parameters. I'm absolutely sure that the values were different before.

A: It's entirely possible that *Salvage* returns *different* values. But theses values are not necessarily wrong as there are numerous combinations which will allow the partitions to be mounted. *Salvage* calculates *one* of them. Just try them and you'll see them working!

Q: I keep getting a requester telling me that a so-called `TD_GETGEOMETRY` isn't working. What does it mean?

A: Read Section 1.4 [Installation], page 5.

Q: My computer crashes when I click on *Start*.

A: Read Section 1.4 [Installation], page 5. Take special note of the sections on `NOGETGEOMETRY`, `NOSCSICMD` and `NONEWSTYLE`.

Q: I've managed to salvage my hard disk but it's running much more slowly now. What can I do about it?

A: Check the values of *Mask* and *MaxTransfer* as described in Section 2.6 [Salvage], page 11 and correct the values if necessary. A larger number of *Buffers* can also increase speed.

Appendiz B RDB-Adjustments

The table below describes RDB adjustments which must be done using *HDToolBox* after you have written the *RDSK* and *partition blocks*. Adjust each of the values in the column corresponding to the function which you were using. Special notes apply to some entries in the table. These are indicated by numbers in round brackets.

Entries annotated with "–" stay, by and large, unaltered. Entries marked with "???" can/should/must be changed. This must be checked in each individual case.

The following numbers are for descriptions which are too long to fit directly in the table:

(1) *SCSI* or *XT* selected according to drive type.

(2) Click and confirm with *Continue*.

(3) Enter the number of *Cylinders* (from the right-hand panel under *Hard disk parameters*).

(4) Enter the number of *Heads* (from the right-hand panel under *Hard disk parameters*).

(5) Enter the number of *BlocksPerTrack* (from the right-hand panel under *Hard disk parameters*).

(6) Enter the number of *BlocksPerCylinder* (from the right-hand panel under *Hard disk parameters*).

(7) Click on this if a file system is to be stored in the RDSK.

Description of Entry	Salvage	Read/Media	Read/DevList
Change Drive Type	click	click	click
Drive Types	(1)	(1)	(1)
Define New...	click	click	click
Read Configuration	(2)	(2)	(2)
FileName	–	–	–
Manufacturers Name	–	–	–
Drive Name	–	–	–
Drive Revision	–	–	–
Cylinders	(3)	(3)	(3)
Heads	(4)	(4)	(4)
Blocks per Track	(5)	(5)	(5)
Blocks per Cylinder	(6)	(6)	(6)
Reduced Write Current Cylinder	–	–	–
Write Precomp Cylinder	–	–	–
Park head where (cylinder)	–	–	–
Supports reselection	–	–	–

Ok	click	click	click
Ok	click	click	click
Modify Bad Block List	–	–	–
Low-level Format Drive	–	–	–
Partition Drive	click	click	click
Advanced Options	click	click	click
Start Cyl	–	–	–
End Cyl	–	–	–
Total Cyl	–	–	–
Buffers	???	–	–
Partition Device Name	???	–	–
HostID	???	–	???
Bootable	???	–	???
Boot Priority	???	–	–
Change...	click	click	click
File System	–	–	–
Fast File System	–	–	–
International Mode	–	–	–
Directory Cache	–	–	–
Automount this partition	???	–	???
File system block size	–	–	–
Identifier	–	–	–
Mask	???	–	–
MaxTransfer	???	–	–
Beginning	–	–	–
End	–	–	–
Use custom boot code	–	–	–
Number of custom boot blocks	–	–	–
Ok	click	click	click
Add/Update...	(7)	(7)	(7)
Add New File System	(7)	(7)	(7)
Ok	(7)	(7)	(7)
Verify Data on Drive	–	–	–
Save Changes to Drive	click	click	click
Exit	click	click	click

Appendiz C Glossary

Bug Report
: Larger projects invariable contain some ugly errors which the user may encounter at some time. In such cases, we ask that the user submit a *Bug Report*. This is a description of the error which has been encountered. Check the circumstances under which the error occurs and write down everything! Note the *exact* version of RDB-Salv which you are using and the configuration of your computer. Most significantly, note the name and version of the *device driver* which you are using. If you are using '`scsi.device`', simply type "`version scsi.device full`" at a CLI prompt and note the output.

Coffee Break
: A coffee break is that amount of time required by a program to do a particular task in background. Most of the time, there's no precise indication of how long this is. Sometimes, it won't be long enough to get the kettle to boil, yet at other times, you'll have enough time to invite the neighbours around to share the latest gossip over some cake.

I'd like to take this opportunity to apologize to those who may have been misled that a *single* cup of coffee was sufficient, by my comments on *Meeting Pearls III*. Of course, a *bucket* of coffee was what I meant. :-)

DeviceList
: This is a system list maintained in RAM, The list contains all devices (e.g. '`DF0:`', '`DH0:`') as well as volumes and assigns known to the system. This list contains all the information required to mount devices such as '`DF0:`' and '`DH0:`'. If your RDB and associated partition blocks have been destroyed, but the devices are still mounted (if you haven't rebooted[1]), then you can salvage your *partitioning data* by scanning the device list for the required devices.

Device Name
: The *device name* is the name of a *partition* as visible from within a *prep program*. '`DH0:`' is a typical example, whereas '`Workbench:`' is *not* a device name, being a *volume name*.

Disaster
: A *disaster* is the greatest foreseeable accident. In relation to RDB-Salv, this means the destruction of the *RDB* and associated *partition blocks*. Such a disaster previously meant the loss of all data on the hard disk.

Device Unit
: See *Unit*.

Device Driver
: Every hard drive requires a *Device Driver* so that it can be used in the AMIGA in a system-conformant way. This driver is responsible for reading data from, and

[1] Note: of course this won't work if you've suppressed the actual mount using the NOMOUNT flag

writing data to the hard disk, amongst other tasks. Devices drivers have a name ending in `.device`. For example; `scsi.device`, `gvpscsi.device`, `omniscsi.device`, `z3scsi.device`, `dracoscsi.device`, etc.

Floppy Image

Floppy disks and even partitions can be written into files by means of "unrolling" them so that the file contains blocks starting from the lowest (0) through to the highest-numbered block at the end. Such a file contains a *Floppy Image*. At some time thereafter, the file can be written to a floppy of equal size to yield an exact copy of the original. The floppy has, in principle, been copied.

Unfortunately, RDB-Salv cannot always discriminate between data representing a real partition or just data from a *Floppy Image* previously stored in a file as it often looks exactly like a real partition. Section 2.6 [Salvage], page 11 describes how to get around problems caused by such a situation.

Hard Disk Parameters

The right-hand panel of the *Salvage*-Page show data including *hard disk parameters*:

Every hard disk has a fixed number of blocks on which data may be stored. These blocks are divided into *Cylinders*, *Tracks*, *Heads* and *Sectors*. A *cylinder* contains one or more tracks — dependent on the number of heads (if there are two heads, then every cylinder contains two tracks, etc.). A track, in turn, consists of a number of sectors.

The actual number of cylinders, heads and sectors is not important in this case. What is important, is that $Cylinders \times Heads \times Sectors$ results in a number which does not exceed the total number of available blocks, and which permits a division so that all partitions can be mounted, as *partition boundaries* can only be specified as cylinder numbers.

Most of the time, more than one combination of values gives *hard disk parameters*. So don't be surprised when RDB-Salv shows values which you are certain are different from the original ones. Just verify it for yourself.

Installer Script

A unified method of installation for all software packages on the AMIGA was developed to make this as simple as possible. Every package includes a plain-text file — the Installer script — to be interpreted by the Installer program at installation time to make the installation process as independent as possible of the prior knowledge of the user.

Management Block

These are blocks such as *partition blocks*, when referred to in connection with RDBs. A verbal shorthand referring to these as *RDB*s is, strictly speaking, incorrect. The RDB consists of a single block — management blocks are however, all those blocks somehow linked to the RDB.

Meeting Pearls

You don't know about Meeting Pearls? That's got to be fixed straight away! Meeting Pearls is Germany's most popular CD-ROM series for the AMIGA as well as being extremely good value for money as the creators don't demand any payment for their work[2], so only direct production and distribution costs need to be paid.. If you have a CD-ROM drive, it's well worth a look.

MountList In principle, the *MountList* contains the same data as *partition blocks*. The significant difference is that MountLists are *files* usually in the '`DEVS:`' directory and therefore readily accessible to the user. Partition blocks, in contrast, are very difficult to access for the normal user.

A MountList may contain entries specifying *several* partitions.

Partition Most hard disks are divided into regions (called partitions), named something like '`DH0:`', '`wb_2.x:`', '`work:`' etc. Sometimes, only a *single* such region exists. A partition can only be mounted if its precise location is known.

This information can be lost in the case of a disaster. Fortunately, RDB-Salv is able to recalculate this information in just about all cases, so that the data can be salvaged.

Partition Block

Such a block is located at the start of the hard disk for (almost) every partition. It contains all the information necessary to mount the partition, e.g. the precise location of the partition. When this information has been destroyed, RDB-Salv can be used to recompute the necessary data.

Partition Boundaries

Most hard disks are divided into *regions* as described under *Partition*. The boundary between partitions is called a *partition boundary*.

Partition Data

Almost every *partition* has a *partition block* which describes how the partition is to be mounted. The information contained in these partition blocks is called *partition data*.

MountLists by the way; also consist of *partition data*.

Prep Program

Prep programs are used to divide hard disks into *regions*. Well-known prep programs include *HDToolBox*, *FaaastPrep*, *ExpertPrep* and *RD-Prep*.

Program Icon

Many programs have an icon — a small symbol on which you can click when it appears on the Workbench. Icons belonging to programs are called *program icons*. Further information about this can be found in your Workbench manual.

[2] although donations are most welcome :-)

Progress Indicator
The *progress indicator* indicates what proportion of work has been done by the use of a bar graph. On occasions, this indicator will move at a very uneven rate. This will always occur if the it wasn't possible to pre-compute how long the particular operation would take.

Pseudo Partition
A partition which isn't normally shown when using *HDToolBox* or other *prep programs*.

RigidDiskBlock
RDB
RDSK The *RDB* is a single block which resides somewhere in the first 16 blocks of a hard disk and contains information about the hard disk. It contains, amongst other things, the location of partition blocks. Contrary to common usage, the *RDB*, *RDSK* or *RigidDiskBlock* is always only *one* block, and thus doesn't include partitioning and other management blocks.

RDB Data
RDB data includes not only the data of the *RDB*, but also that of associated blocks, like *partition blocks*.

Region See *Partition*.

System Partition
Almost every AMIGA which has a hard disk, has such a partition. It's the partition from which the machine is booted, and it's often called something like '`wb_2.x:`', '`SYS`' or '`System:`'.

UNUSED<n>
RDB-Salv chooses this *volume name* for all parts of the hard disk which are unused. This name is used not only when *partition blocks* or *MountLists* are written, but also when volumes are mounted.

Unit A particular hard disk drive is identified by its *unit* or *device unit*. Valid unit numbers are typically between 0 and 6 for SCSI drives. If you have an external drive, this may have a switch for setting a number. The number is usually the unit number of the drive. If in doubt about the unit number, consult the manual for your hard disk system.

Volume Name
The *formatted* name of the *partition* is the *volume name*. This name is also visible on the *Workbench* screen and can be changed at almost any time[3] using Workbench *Rename* ... menu function. '`Workbench`' is a typical volume name, whereas '`DH0:`' is *not*; that being the *device name*.

[3] as long as the unit isn't write-protected

Work-Partition

Most hard disk on the AMIGA are divided into at least two regions. One of these partitions — the *Work-Partition* — contains user-dependent data, like special programs, projects created by the user, etc. This partition is usually called '`work:`'.

Appendiz D Acknowledgements

Firstly; a *big* round of thank-yous to all the nice people who translated RDB-Salv to another language:

Bernd Felsche
: For the translation of the manual to English.

Eivind Olsen
: For the translation of the manual and catalog to Norwegian.

Francesco Dipietro
: For the translation of the catalog to Italian.

Fredrik Zetterlund
: For the translation of the manual to Swedish.

Johan Roennblom
: For the translation of the catalog and parts of the manual to Swedish.

Wessel Dankers
: For the translation of the manual and catalog to Dutch.

Yann-Erick Proy
: For the translation of the manual and catalog to French.

And all the others
: Maybe there will join a few translators *after* this manual has been released. These persons will be named in the next version.

And naturally thanks to all those who've contributed to making RDB-Salv a success. Special thanks go to Christian Krapp and Patrick Ohly who provided the scroller and registration routines respectively.

Andreas Horneffer
: He was a beta tester and helped to fix some bugs under V39.

Bernhard Möllemann
: Who received almost every beta version, reviewed all changes and gave his opinion. He was, by the way, a good beta tester and he has written the first device which supports new style commands, so I could test the new style interface of RDB-Salv.

Christian 'cr8y' Krapp
: Who programmed the pretty scroller routine as well as drawing the ambulance and the Amiga-Logo.

Klaus Melchior
: He too was a good beta tester.

Markus Illenseer
: Who also did some good testing as well as reviewing the German- and French-language manuals.

Michael-Wolfgang Hohmann
Who created the icons.

Olaf Barthel
Who also did some good beta testing.

Patrick Ohly
Who wrote the routine to check the registration number; tested RDB-Salv and wrote bug reports.

Reinhard Spisser, Sebastiano Vigna
Both contributed by providing makeguide for the AMIGA; the program used to format this text. Many thanks.

Stefan Schulz
Who did some beta testing and was responsible for ensuring that this project was *indeed* revived and did not wind up gathering dust on the shelf as it had threatened to do.

Keyword Index