MRBackup

COLLABORATORS			
	TITLE : MRBackup		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		October 20, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

MRBackup

1.1 Data Compression

MRBackup provides you with the ability to compress your files as they are
written to a save-set and to decompress them when they are restored. The
primary motivation for doing this is to save space on the backup media and
thus reduce the amount of media required to hold the save set. There is a
performance penalty exacted for this, however. You must determine if the
savings in space are worth the extra time required to perform the backup or
restore of compressed files. The use of data compression also places extra
demands on system memory which may be a consideration if you are running
other programs (multitasking) while MRBackup is running.

    Data Compression Method

    The Compressor

    Compression Estimating

    Go to Top Level

1.2 data compression method

MRBackup employs Lempel-Ziv compression. While this method does not yield
the highest compression ratios, it is one of the faster software
compression algorithms available. Its ability to be "tuned" through the
use of user-specified code size limits allows you to make certain
performance trade-offs. Larger code sizes will make greater temporary
demands on system memory but will result in higher compression ratios.

When a file is compressed, special codes are written at the beginning of
the compressed file to indicate that it is compressed and to record the
size of the codes used for compression. Thus, you need not remember what
code size was used to compress a particular file when you later decompress
it.

1.3 the compressor

MRBackup performs data compression with a separate program named
Compressor. Whenever you start a backup with compression enabled or a
restore with decompression enabled or if you compress/decompress individual
files using the Utilities, MRBackup will check to see if the Compressor is
running. If not, you will be asked for permission to start it. MRBackup
will then enter into a "conversation" with the Compressor, requesting data
to be processed and retrieving the results.

There are several advantages to this approach. Some of them are too
technical to be discussed here. Of most importance to the user is that as
a result, the MRBackup program is smaller. If data compression is not
being used, less memory is being used by MRBackup. The Compressor program
is designed to employ "overlays". When it is idle, it uses almost no Amiga
resources since it releases the data compression code and buffer memory and
waits for the next request to do something.

Starting and Stopping the Compressor Manually
You may elect to start and stop the Compressor "manually" if you so desire.
To start the Compressor, simply enter

    RUN MRBackup:Compressor <nil: >nil:

from the Shell command prompt or double-click the Compressor icon. To stop
the Compressor, enter

    MRBackup:Compressor quit

from the Shell command prompt.

1.4 compression estimating

Compression estimating allows MRBackup to better determine if a file will
fit on the current backup diskette when performing an AmigaDOS backup with
compression enabled (it is irrelevant to Fast Disk and SCSI Tape backups).
MRBackup does not know in advance what a file's size will be after it is
compressed. Therefore, when determining whether a file will fit on the
current backup diskette, the file's full size is used. This can result in
a significant amount of wasted space on each diskette. If you set the
compression estimate to a non-zero value, MRBackup will apply this estimate
when determining space available. A reasonable value to start with is 35
(%). This means that you expect most files to be 65% of their original
size (100%-35%) when compressed. Please note that this may lead to
occaisional "disk full" errors, depending on how aggressive your estimate
is. In this case, MRBackup will delete the partially copied file and force
a new diskette. This is a feature you'll have to develop a "feel" for. Of
course, you can always play "safe" and leave this value at zero.