

**NAME**

lbzip2 – parallel bzip2 utility

**SYNOPSIS**

**lbzip2|bzip2** [-n *WORKER-THREADS*] [-c|-t] [-d] [-1 .. -9] [-v] [-S] [ *FILE* ... ]

**lbunzip2|bunzip2** [-n *WORKER-THREADS*] [-c|-t] [-z] [-v] [-S] [ *FILE* ... ]

**lbzcat|bzcat** [-n *WORKER-THREADS*] [-z] [-v] [-S] [ *FILE* ... ]

**lbzip2|bzip2|lbunzip2|bunzip2|lbzcat|bzcat -h**

**DESCRIPTION**

Compress or decompress *FILE* operands or standard input to regular files or standard output, by calling Julian Seward's **libbz2** from multiple threads. The **lbzip2** utility employs multiple threads and an input-bound splitter even when decompressing **.bz2** files created by standard **bzip2** (but see **BUGS** below).

**INVOCATION**

The default mode of operation is compression. If the utility is invoked as **lbunzip2** or **bunzip2**, the MO is switched to decompression. Calling the utility as **lbzcat** or **bzcat** selects decompression, with the decompressed byte-stream written to standard output.

**OPTIONS**

**-n** *WORKER-THREADS*

Set the number of (de)compressor threads to *WORKER-THREADS*. If this option is not specified, **lbzip2** either queries the system for the number of online processors (if both the compilation environment and the execution environment support that), or it exits with an error.

**-c, --stdout**

Write output to standard output, even when *FILE* operands are present. Excludes **-t**.

**-t, --test**

Test decompression; discard output instead of writing it to files or standard output. Excludes **-c**.

**-d, --decompress**

Force decompression over the mode of operation selected by the invocation name.

**-z, --compress**

Force compression over the mode of operation selected by the invocation name.

**-1 .. -9**

Set the compression block size to 100K .. 900K, in 100K increments.

**--fast** Alias for **-1**.

**--best** Alias for **-9**. This is the default.

- v, --verbose**  
Print a short message to standard error whenever a (de)compression operation commences.
- S**  
Print condition variable statistics to standard error for each completed (de)compression operation.
- f, --force**  
Accepted for compatibility with **bzip2**, otherwise ignored; **lbzip2** never overwrites files.
- k, --keep**  
Accepted for compatibility with **bzip2**, otherwise ignored; **lbzip2** never removes input files.
- s, --small, -q, --quiet, --repetitive-fast, --repetitive-best**  
Accepted for compatibility with **bzip2**, otherwise ignored.
- h, --help, -L, --license, -V, --version**  
Display license and version information, plus help on command-line usage.

## ENVIRONMENT

### LBZIP2, BZIP2, BZIP

Before parsing the command line, **lbzip2** inserts the contents of these variables, in the order specified, between the invocation name and the rest of the command line. Tokens are separated by spaces and tabs, which cannot be escaped.

### LBZIP2\_TRACE\_ALLOC

If this variable is set to a non-empty value, **lbzip2** prints a memory allocation trace to standard error. The trace can be checked with the **malloc\_trace.pl** Perl script (part of the source distribution).

## OPERANDS

*FILE* Specify files to compress or decompress. If no *FILE* is given, **lbzip2** works as a filter. *FILE*s with **.bz2**, **.tbz**, **.tbz2** and **.tz2** name suffixes will be skipped when compressing. When decompressing, **.bz2** suffixes will be removed in output filenames; **.tbz**, **.tbz2** and **.tz2** suffixes will be replaced by **.tar**; other filenames will be suffixed with **.out**. If an **INT** or **TERM** signal is delivered to **lbzip2**, then it removes the regular output file currently open before exiting.

## EXIT STATUS

**0** if **lbzip2** finishes successfully. This presumes that whenever it tries, **lbzip2** never fails to write to standard error.

### EXIT\_FAILURE

if **lbzip2** encounters any fatal error. (**EXIT\_FAILURE** corresponds to a platform-dependent integer constant in [1..255], usually **1**.)

### SIGABRT

(via **assert()**) if a runtime assertion fails (ie. **lbzip2** detects a bug in itself). Hopefully whoever compiled your binary wasn't bold enough to **#define NDEBUG**.

### SIGINT, SIGTERM

**lbzip2** catches these signals so that it can remove an interrupted output file. In such cases, **lbzip2** exits by re-raising (one of) the received signal(s).

## BUGS

The compressed output is a sequence of single-block bzip2 streams, instead of being a single multi-block bzip2 stream. Thus no output-global combined CRC can be written, and the resulting **.bz2** file will return multiple **BZ\_STREAM\_ENDs** when decompressed with **libbz2**.

The multiple-workers decompressor (**lbzip2 -d -n WORKER-THREADS**, where **WORKER-THREADS** is greater than **1**, either by default or by manual selection) doesn't check stream-level (combined) CRCs, only block-level CRCs. It can also fail to decompress some rare valid **.bz2** files. (See the Bugs section of the README in the source distribution for more on this.) Since the author cannot rule out that such a failure may get reported as corrupt input instead of the dedicated error message, **lbzip2** doesn't reserve a separate exit status for corrupt input -- it could be misleading.

The bit-string search algorithm used in the multiple-workers decompressor is naive and slow.

Separate input files don't share worker threads; at most one input file is worked on at any moment.

## AUTHOR

Written by Laszlo Ersek <lacos@caesar.elte.hu>, <http://lacos.hu/>.

## COPYRIGHT

Copyright (C) 2008, 2009 Laszlo Ersek.

This is free software. You may redistribute copies of it under the terms of the GNU General Public License, version 2:

<http://www.gnu.org/licenses/gpl-2.0.html>

or -- at your opinion -- any later version, eg.

<http://www.gnu.org/licenses/gpl.html>

## THANKS

Adam Maulis at ELTE IIG; Julian Seward; Paul Sladen; Michael Thomas from Caltech HEP; Bryan Stillwell; Zsolt Bartos-Elekes; Imre Csatlos; Gabor Kovesdan; Paul Wise; Department of Electrical and Information Engineering at the University of Oulu.

## SEE ALSO

**bzip2(1)**

<http://www.bzip.org/>

**pbzip2(1)**

<http://compression.ca/pbzip2/>

**bzip2smp(1)**

<http://bzip2smp.sourceforge.net/>

**smpbzip2(1)**

<http://home.student.utwente.nl/n.werensteijn/smpbzip2/>

**dbzip2(1)**

<http://www.mediawiki.org/wiki/Dbzip2>

**p7zip(1)**

<http://p7zip.sourceforge.net/>