

bzip2

将文件压缩成bz2格式

补充说明

bzip2命令 用于创建和管理（包括解压缩“.bz2”格式的压缩包。

bzip2 采用 Burrows-Wheeler 块排序文本压缩算法和 Huffman 编码方式压缩文件。压缩率一般比基于 LZ77/LZ78 的压缩软件好得多，其性能接近 PPM 族统计类压缩软件。

命令行参数有意设计为非常接近 GNU gzip 的形式，但也不完全相同。

bzip2 从命令行读入文件名和参数。每个文件被名为“原始文件名.bz2”的压缩文件替换。每个压缩文件具有与原文件相同的修改时间、权限，如果可能的话，还具有相同的属主，因此在解压缩时这些特性将正确地恢复。在某些文件系统中，没有权限、属主或时间的概念，或者对文件名的长度有严格限制，例如 MSDOS在这种情况下bzip2 没有保持原文件名、属主、权限以及时间的机制，从这个意义上说bzip2 对文件名的处理是幼稚的。

bzip2 和 bunzip2 在缺省情况下不覆盖已有的文件。如果想覆盖已有的文件，要指定 -f 选项。

如果未指定文件名 bzip2 将压缩来自标准输入的数据并写往标准输出。在这种情况下 bzip2 会拒绝将压缩结果写往终端，因为这完全无法理解并且是没有意义的。

bunzip2 (以及 bzip2 -d) 对所有指定的文件进行解压缩处理。不是由 bzip2 产生的文件将被忽略，同时发出一个警告信息 bzip2 按下列方式由压缩文件名确定解压后的文件名：

filename.bz2	解压成	filename
filename.bz	解压成	filename
filename.tbz2	解压成	filename.tar
filename.tbz	解压成	filename.tar
anyothername	解压成	anyothername.out

如果文件名的后缀不是下列之一 .bz2, .bz, .tbz2 或 .tbz, .bzip2 将抱怨无法确定原始文件名，并采用原文件名加 .out 作为解压缩文件名。

在压缩时，如果不提供文件名bzip2 将从标准输入读取数据，压缩结果写往标准输出。

bzip2 采用 32 位 CRC 校验码作自我检查，以确认解压后的文件与原始文件相同。这可用于检测压缩文件是否损坏，并防止 bzip2 中未知的缺陷（运气好的话这种可能性非常小）。数据损坏而未检测到的几率非常之小，对于每个被处理的文件大约是四十亿分之一。检查是在解压缩时进行的，因此它只能说明某个地方出问题了。它能帮助恢复原始未压缩的数据。可以用 bzip2recover 来尝试从损坏的文件中恢复数据。

返回值：正常退出返回 0，出现环境问题返回 1（文件未找到，非法的选项I/O错误等），返回 2 表明压缩文件损坏，出现导致 bzip2 紧急退出的内部一致性错误（例如缺陷）时返回 3。

语法

```
bzip2 [ -cdfkqstvzVL123456789 ] [ filenames ... ]
```

选项

```
-c --stdout
# 将数据压缩或解压缩至标准输出。

-d --decompress
# 强制解压缩。 bzip2, bunzip2 以及 bzipcat 实际上是同一个程序，进行何种操作将根据程序名确定。指定该选项后将不考虑这一机制，强制 bzip2 进行解压缩。

-z --compress
# -d 选项的补充：强制进行压缩操作，而不管执行的是哪个程序。

-t --test
# 检查指定文件的完整性，但并不对其解压缩。实际上将对数据进行实验性的解压缩操作，而不输出结果。

-f --force
# 强制覆盖输出文件。通常 bzip2 不会覆盖已经存在的文件。该选项还强制 bzip2 打破文件的硬连接，缺省情况下 bzip2 不会这么做。

-k --keep
# 在压缩或解压缩时保留输入文件（不删除这些文件）。

-s --small
# 在压缩、解压缩及检查时减少内存用量。采用一种修正的算法进行压缩和测试，每个数据块仅需要 2.5 个字节。这意味着任何文件都可以在 2300k
# 的内存中进行解压缩，尽管速度只有通常情况下的一半。

# 在压缩时 -s 将选定 200k 的块长度，内存用量也限制在 200k 左右，代价是压缩率会降低。总之，如果机器的内存较少（8兆字节或更少），
# 可对所有操作都采用 -s 选项。参见下面的内存管理。

-q --quiet
# 压制不重要的警告信息。属于 I/O 错误及其它严重事件的信息将不会被压制。

-v --verbose
# 详尽模式 -- 显示每个被处理文件的压缩率。命令行中更多的 -v 选项将增加详细的程度，使 bzip2 显示出许多主要用于诊断目的的信息。

-L --license -V --version
# 显示软件版本，许可证条款及条件。

-l to -9
# 在压缩时将块长度设为 100 k 到 200 k .. 900 k 对解压缩没有影响。参见下面的内存管理。

-- # 将所有后面的命令行变量看作文件名，即使这些变量以减号 "-" 打头。可用这一选项处理以减号 "-" 打头的文件名，例如 bzip2 -- -myfilename.

--repetitive-fast --repetitive-best
# 这些选项在 0.9.5 及其以上版本中是多余的。在较早的版本中，这两个选项对排序算法的行为提供了一些粗糙的控制，有些情况下很有用。0.9.5
# 及其以上版本采用了改进的算法而与这些选项无关。
```

参数

文件：指定要压缩的文件。

实例

压缩指定文件**filename**:

```
bzip2 filename  
或  
bzip2 -z filename
```

这里，压缩的时候不会输出，会将原来的文件filename给删除，替换成filename.bz2.如果以前有filename.bz2则不会替换并提示错误（如果想要替换则指定-f选项，例如**bzip2 -f filename**；如果filename是目录则也提醒错误不做任何操作；如果filename已经是压过的了有bz2后缀就提醒一下，不再压缩，没有bz2后缀会再次压缩。

解压指定的文件**filename.bz2**:

```
bzip2 -d filename.bz2  
或  
bunzip2 filename.bz2
```

这里，解压的时候没标准输出，会将原来的文件filename.bz2给替换成filename[]如果以前有filename则不会替换并提示错误（如果想要替换则指定-f选项，例如**bzip2 -df filename.bz2**）

压缩解压的时候将结果也输出：

```
$bzip2 -v filename
```

输入之后，输出如下：

```
filename:  0.119:1, 67.200 bits/byte, -740.00% saved, 5 in, 42 out.
```

这里，加上-v选项就会输出了,只用压缩举例了，解压的时候同理**bzip2 -dv filename.bz2**不再举例了。

模拟解压实际并不解压：

```
bzip2 -tv filename.bz2
```

输入之后，输出如下：

```
filename.bz2: ok
```

这里，-t指定要进行模拟解压，不实际生成结果，也就是说类似检查文件,当然就算目录下面有filename也不会有什么错误输出了，因为它根本不会真的解压文件。为了在屏幕上输出，这里加上-v选项了,如果是真的解压**bzip2 -dv filename.bz2**则输出的是把"ok"替换成了"done"[]

压缩解压的时候，除了生成结果文件，将原来的文件也保存:

```
bzip2 -k filename
```

这里，加上 -k 就保存原始的文件了，否则原始文件会被结果文件替代。只用压缩举例了，解压的时候同理 `bzip2 -dk filename.bz2` 不再举例了。

解压到标准输出：

```
bzip2 -dc filename.bz2
```

输入之后，输出如下：

```
hahahhaahahaha
```

这里，使用 -c 指定到标准输出，输出的是文件 filename 的内容，不会将 filename.bz2 删除。

压缩到标准输出：

```
bzip2 -c filename
bzip2: I won't write compressed data to a terminal.
bzip2: For help, type: `bzip2 --help'.
```

这里，使用 -c 指定压缩到标准输出不删除原有文件，不同的是，压缩后的文件无法输出到标准输出。

使用 **bzip2** 的时候将所有后面的看作文件(即使文件名以 '-' 开头)：

```
bzip2 -- -myfilename
```

这里主要是为了防止文件名中 - 产生以为是选项的歧义。

From:

<https://rd.irust.top/> - 学习笔记

Permanent link:

<https://rd.irust.top/doku.php?id=command:bzip2>

Last update: **2021/10/15 14:58**

