

**NAME**

xdelta – Invoke Xdelta

**SYNOPSIS**

**xdelta** *subcommand* [ *option...* ] [ *operand...* ]

**DESCRIPTION**

Xdelta provides the ability to generate deltas between a pair of files and later apply those deltas. It operates similar to the **diff** and **patch** commands, but works on binary files and does not produce a human readable output.

Xdelta has three subcommands, delta, patch, and info. Delta accepts two file versions and produces a delta, while patch accepts the original file version and delta and produces the second version. The info command prints useful information about a delta. Each subcommand will be detailed separately.

**Gzip processing**

Attempting to compute a delta between compressed input files usually results in poor compression. This is because small differences between the original contents causes changes in the compression of whole blocks of data. To simplify things, Xdelta implements a special case for *gzip*(1) compressed files. If any version input to the delta command is recognized as having gzip compression, it will be automatically decompressed into a temporary location prior to comparison. This temporary location is either the value of the *TMPDIR* environment variable, if set, otherwise *"/tmp"*.

The Xdelta patch header contains a flag indicating that the reconstructed version should be recompressed after applying the patch. In general, this allows Xdelta to operate transparently on gzip compressed inputs.

There is one potential problem when automatically processing gzip compressed files, which is that the recompressed content does not always match byte-for-byte with the original compressed content. The uncompressed content still matches, but if there is an external integrity check such as cryptographic signature verification, it may fail. To prevent this from happening, the *--pristine* option disables automatic gzip processing.

**MD5 integrity check**

By default, Xdelta always verifies the MD5 checksum of the files it reconstructs. This prevents you from supplying an incorrect input during patch, which would result in corrupt output. Because of this feature, you can feel confident that patch has produced valid results. The *--noverify* option disables MD5 verification, but this is only recommended for performance testing.

**Compressed patch format**

Xdelta uses a fairly simple encoding for its delta, then applies zlib compression to the result. You should not have to post-compress an Xdelta delta.

**Delta**

The delta subcommand has the following synopsis:

**xdelta** *delta* [ *option...* ] *fromfile tofile patchout*

Computes a delta from *fromfile* to *tofile* and writes it to *patchout*

**Patch**

The patch subcommand has the following synopsis:

**xdelta** *patch* [ *option...* ] *patchin* [ *fromfile* [ *tofile* ] ]

Applies *patchin* to *fromfile* and produces a reconstructed version of *tofile*.

If *fromfile* was omitted, Xdelta attempts to use the original *fromfile* name, which is stored in the delta. The *fromfile* must be identical to the one used to create the delta. If its length or MD5 checksum differs, patch will abort with an error message.

If *tofile* was omitted, Xdelta attempts to use the original *tofile* name, which is also stored in the delta. If the original *tofile* name already exists, a unique *filename* extension will be added to avoid destroying any existing data.

### Info

The info subcommand has the following synopsis:

**xdelta** *info patchinfo*

Prints information about *patchinfo* and the version it reconstructs, including *filename* names, lengths, and MD5 checksums.

### Options

-0..9 Set the zlib compression level. Zero indicates no compression. Nine indicates maximum compression.

-h, --help  
Print a short help message and exit.

-q, --quiet  
Quiet. Suppresses several warning messages.

-v, --version  
Print the Xdelta version number and exit.

-V, --verbose  
Verbose. Prints a bit of extra information.

-n, --noverify  
No verify. Turns off MD5 checksum verification of the input and output files.

-m=SIZE, --maxmem=SIZE  
Set an upper bound on the size of an in-memory page cache. For example, --maxmem=32M will use a 32 megabyte page cache.

-s=BLOCK\_SIZE  
Set the block size, unless it was hard coded (20% speed improvement). Should be a power of 2.

-p, --pristine  
Disable the automatic decompression of gzipped inputs, to prevent unexpected differences in the re-compressed content.

**IDENTIFICATION**

Author: Joshua P. MacDonald, [jmacd@cs.berkeley.edu](mailto:jmacd@cs.berkeley.edu)

Manual Page Revision: 1.5; Release Date: Fri, 29 Jun 2001 06:01:08 -0700.

Copyright © 1997, 1998, 1999, 2000, 2001