gsar

NAME

gsar – General Search And Replace utility

SYNOPSIS

gsar [*options*] [*infile(s)*] [*outfile*]

DESCRIPTION

gsar (General Search And Replace) is a utility for searching for and --- optionally --- replacing strings in both text and binary files. The search and replace strings can contain all kinds of characters (0--255), i.e. Ctrl characters and extended ASCII as well.

The algorithm used is a variation of the Boyer-Moore search algorithm, modified to search binary files. As a result of this, **gsar** is blindingly fast.

Opposed to line oriented search programs (like **grep(1)**), gsar will find all matches on a line. Actually, **gsar** doesn't know anything about lines at all, all files and strings are treated as binary.

Gsar can search one or several files for a string and report the occurrences. **Gsar** can read one file, search for a string, replace it with some other string, and create a new file containing the changes. **Gsar** can perform a search and replace in multiple files, overwriting the originals. Finally, gsar can work as a filter, reading from standard input and writing to standard output.

OPTIONS

All options can be concatenated into one single option i.e the command: **gsar -i -b -l** is the same as **gsar -ibl**

An option which takes an argument must be the last one in the concatenated option, since the rest of the option is taken as a possible argument.

Fields enclosed in [] are optional, but mandatory when enclosed in <>. Options are case sensitive i.e **-b** is not the same as **-B**.

If no options are given, **gsar** just gives a brief help message.

[infile(s)]

Name(s) of input file(s) (wildcards allowed on most Unix shells and most DOS compilers). If the **-F** option is used input is taken from *stdin.*

[outfile]

Name of output file that is to contain the replacements. If the **-F** option is used, transformed output is sent to *stdout.*

-s<*string***>**

String to search for in file. Ctrl characters can be entered by using a ':' in the string followed by the ASCII value of the character. The value is entered using a ':' followed by three decimal digits or ':x' followed by two hex numbers. To enter a colon (:) in the search pattern use '::'. The *string* must follow directly after **s**.

Example: To search for the string *:foo* ('o' is 111 decimal, 6F in hex) use the search options:
-s::foo or **-s::fo:111** or **-s::fo:x6F**

If you want to search for a string with spaces in it, under MSDOS surround the expression with quotes. Under Unix, use the mechanisms your shell provides (commonly quotes) to include space or other special characters.

Example: search for *gsar is fast* use:

```
gsar "-sgsar is fast" foobar.txt
```

If response files are needed, most Unix shells will allow

```
gsar ‘cat foobar.txt‘ poppins.txt
```

-r[*string***]**

String which is to replace search string in file. Use **-r** to delete the search string from the file i.e. replace with nothing. Ctrl characters can be entered in the same way as in the search string. If this option is left out, **gsar** only performs a search. The *string* must follow directly after **r**.

-i

Ignore case difference when comparing strings. I.e. *foobar* matches *fooBAR*.

-B

Just display the search & replace buffers, for test purposes.

-f

If the output file already exists this switch can be used to force an overwrite of the existing output file.

-o

Search and replace of multiple files, overwrite the input file(s). For each input file, gsar creates a tempfile which contains the replacements and copies the tempfile to the original input file name. If no matches were found, the input file stays the same. The tempfile is removed.

Example:

```
gsar -s__ZTC__ -r__TURBOC__ -o foo.c bar.c bat.c
```

The files *foo.c*, *bar.c* & *bat.c* are all changed.

-c[*n*]

Display the context around a match in a textual manner. Undisplayable characters are displayed as a dot (‘.’). *n* is optional number of bytes in context. *n* must follow directly after **c**.

-x[*n*]

Display the context around a match as a hexadecimal dump. Undisplayable characters are displayed as a dot (‘.’). *n* is optional number of bytes in context. *n* must follow directly after **x**.

-b

Display the byte offset of the match in hex.

-l

Only list filename and number of matches if any (default).

-h

Suppress display of filename when displaying context or byte offsets.

-du

Convert a DOS ASCII file to UNIX (strips carriage return).

-ud

Convert a UNIX ASCII file to DOS (adds carriage return).

-F

Filter mode, gsar takes it’s input from *stdin* and redirects eventual output to *stdout*. All error messages are sent *stderr*.

-G

Display the GNU General Public Licence.

Examples

Search for two spaces at the end of a line (DOS text) and replace with just a carriage return overwriting the original files:

```
gsar -s:x20:x20:x0d -r:x0d -o foobar.txt *.c
```

Convert a UNIX text file to DOS format overwriting the original file:

```
gsar -ud -o unix.txt
```

Search for the string *WATCOM* and replace with *__ZTC__* using gsar as a filter. Output is redirected to a new file:

```
gsar -sWATCOM -r__ZTC__ -F < foo_w.c > foo_z.c
```

Display textual context of the string *error* in the file *gsar.exe* disregarding case. With 40 bytes in the context:

```
gsar -serror -i -c40 gsar.exe
```

Search for the string *gnu* in the file *fsf* and replace it with *wildebeest*, creating a new output file *africa*:

```
gsar -sgnu -rwildebeest fsf africa
```

(if the file *africa* exists, you have to use the **-f** option to overwrite it.)

Search for the string *error* in the file *command.com* and display the byte offset of each match:

```
gsar -serror -b command.com
```

Limitations

No wildcards or regular expressions allowed in search string.

On MSDOS platforms *stdin* from a tty is not allowed because *stdin* has been turned into binary. MSDOS will not catch the Ctrl-Z signifying EOF.

Authors

Tormod Tjaberg (coding, design (all bugs are his))
Hans Peter Verne (ideas, demands, testing, UNIX platforms)

If you have any comments, bug reports or whatever, we can be reached through email at:

tjaberg@online.no
hpv@kjemi.uio.no