GNU/Linux - Sed

Prasad H. L.

June 27, 2008

This document is distributed under GNU FDL. To obtain a full copy of GNU FDL [[click here]].

1 Summary

Summary

Introduction to

- Regular Expressions
- VIM Editor
- GREP

2 Introduction

Contents

Introduction

What is sed?
sed stands for **S**tream **Ed**itor **sed** is most commonly used with *streams* **sed** is a vary powerful utility for manipulating text data

Invocation

Invocation of sed is through

```
sed [OPTION]... {script-only-if-no-other-script} [input-file]...
```

Most common options:

OPTION	USE
-n,–**quiet**,–**silent**	suppress automatic printing
-e script,–**expression**=script	script
-f script-file,–**file**=script-file	add script file

3 Commands

3.1 s

The all important 's' command

's'

- '*s*' stands for *substitute*
- format: *s/REGEXP/REPLACEMENT/FLAGS*[1]
- substitutes first occurance of *REGEXP* with *REPLACEMENT*

Some useful utilities and flags

& represents the matched pattern

\[n] represents *n*th matched pattern

flag *g* replaces all occurances

ending */n* replaces only the *n*th matched pattern

/p prints only modified line (useful with -n option)

/w writes to a file

Examples of *s* command

Example

1. *sed 's/day/night/g' < old > new*
   replaces *day* with *night* in file *old* and outputs to file *new*

[1] */* is delimiter, replacable with any character

2. Example

   `sed 's/[a-z]*/(&)/' <old >new`

   brackets all groups of small characters in *old* and writes out to *new*

3. Example

   `sed 's/\([a-z]*\) \([a-z]*\)/\2 \1/'`

   switches first two small lettered words

4. Example

   `sed 's/[a-zA-Z]* //2`

   deletes the second word of a line

Ranges

Restrictions

Any operation can be restricted to be performed withing a range specified by

- Specifying a line by its number.
- Specifying a range of lines by number.
- All lines containing a pattern.
- All lines from the beginning of a file to a regular expression
- All lines from a regular expression to the end of the file.
- All lines between two regular expressions.

Restriction by examples

1. Example

   `sed ' /^#/ s/[0-9][0-9]*//'`

   deletes the first number on all lines starting with *#*

2. Example

   `sed '1,100 s/A/a/'`

   replaces the first occurance of *A* with *a* in the first 100 lines

3. Example

   `sed '/start/,/stop/ s/#.*//'`

   removes all lines starting with *#* and between the lines having *start* and *stop*

3.2 d

The d command

'd'

- '*d*' stands for *delete*
- format: */RESTRICTION/ d*
- deletes every line matching the restriction

For example

Example

sed '11,$ d' <file

deletes the 11th line onwards in *file*

Compare the above example with *head*

The 'p' command

'p'

- '*p*' stands for *print*
- format: */RESTRICTION/ p*
- prints every line matching the restriction (twice if sed not invoked with ***-n***)

For example

Example

sed -n '/match/ p'

Prints all lines with the word match

Compare the above with *grep*

3.3 q

The 'q' command

'q'

- '*q*' stands for *quit*
- format: */RESTRICTION/ q*
- quits sed on occutance of the restriction

For example

Example

sed '11 q'

prints the first 10 lines

Compare the above with *head*

3.4 Grouping

Grouping

Many actions on the lines satisfying the same restriction can be grouped with {,} For example

Example

```
sed -n ’/begin/,/end/
{
s/#.*//
/^$/ d
p
}’
```

Does the following between lines containing *begin* and *end*:

- remove lines starting with #
- Deletes empty lines
- prints all other lines

3.5 More Features

More Features

Other commands

r	Read from a file
w	Write to a file
a	Append a line
i	Insert a line
c	Change a line

Examples

’r’: Read a file

```
sed ’/INCLUDE/ r file’
```

Will insert *file* at the occurance of *INCLUDE*

’w’: Write a File

```
sed ’s/[0-9]*[02468]/&/w even’
```

Writes lines having an even number to file *even*

’a’: Append a line

```
sed ’/WORD/ a\ Some text’
```

Appends a line after every occurance of *WORD*

’i’: Insert a line

```
sed ’/WORD/ i\ Some text’
```

Inserts a line before every occurance of *WORD*

'c': Change a line
sed '/WORD/ c\ Some text' Replaces the current line at every occurance of *WORD*

References

References

[1] *info sed, info:/sed*.

[2] *man sed, man:/sed*.