

sed

A language for processing text streams

Introduction

- It is a programming language
- `sed` is an abbreviation for `s`stream `e`ditor
- It is a part of POSIX
- `sed` precedes `awk`

Execution model

- Input stream is a set of lines
- Each line is a sequence of characters
- Two data buffers are maintained: active *pattern* space and auxiliary *hold* space
- For each line of input, an *execution cycle* is performed loading the line into the pattern space
- During each cycle, all the statements in the script are executed in the sequence for matching *address pattern* for *actions* specified with the *options* provided

usage

- Single line at the command line

```
sed -e 's/hello/world/g' input.txt
```

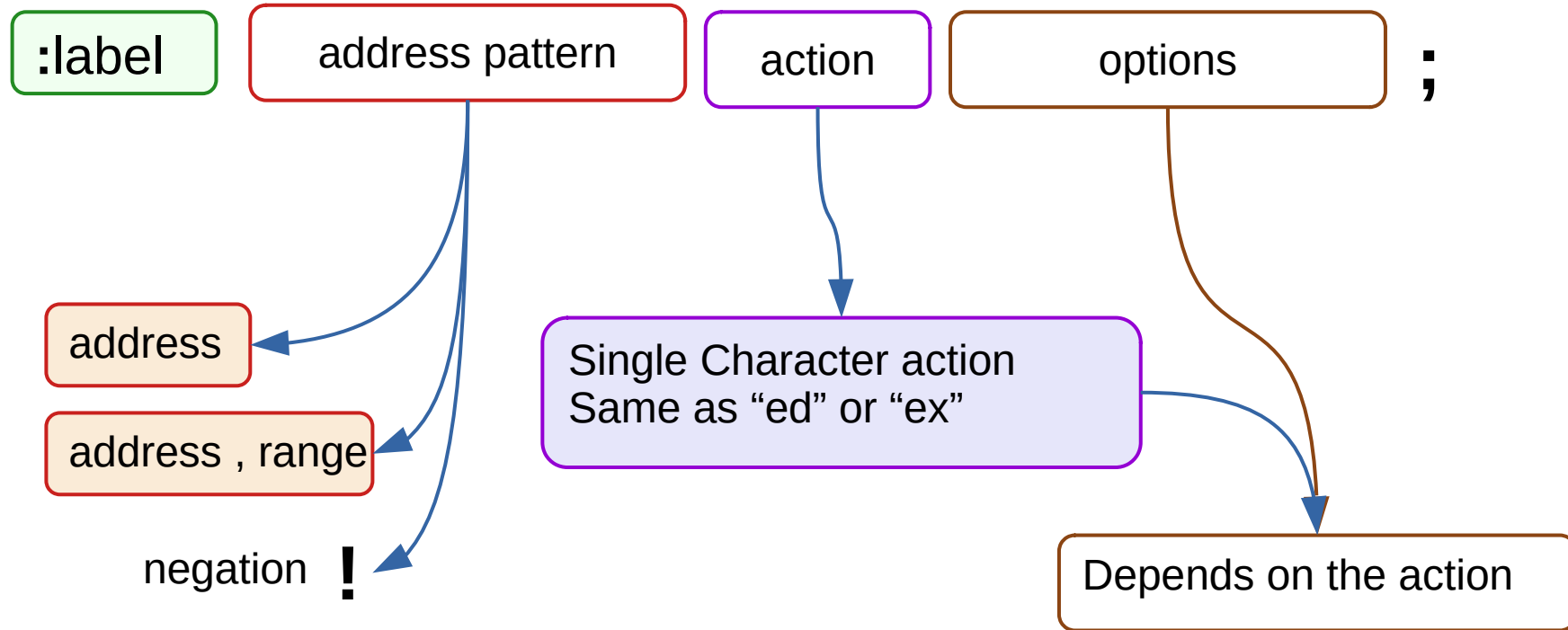
- Script interpreted by sed

```
sed -f ./myscript.sed input.txt
```

myscript.sed

```
#!/usr/bin/sed -f  
2,8s/hello/world/g
```

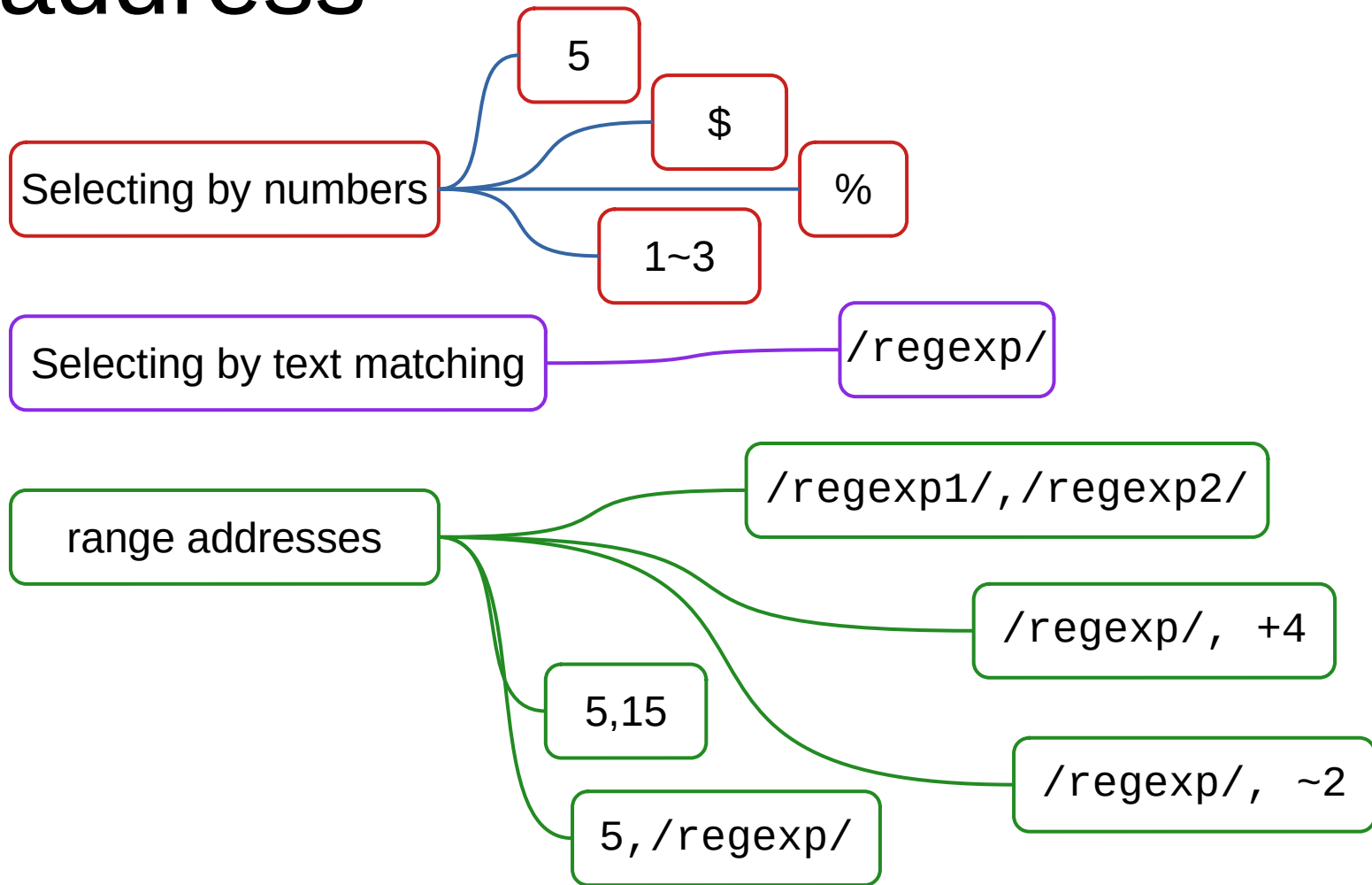
sed statements



Grouping commands

```
{ cmd; cmd; }
```

address



actions

p	Print the pattern space
d	Delete the pattern space
s	Substitute using regex match s/pattern/replacement/g
=	Print current input line number, \n
#	comment
i	Insert above current line
a	Append below current line
c	Change current line

programming

<i>b label</i>	Branch unconditionally to <i>label</i>
<i>:label</i>	Specify location of <i>label</i> for branch command
N	Add a new line to the pattern space and append next line of input into it.
q	Exit sed without processing any more commands or input lines
<i>t label</i>	Branch to <i>label</i> only if there was a successful substitution was made
<i>T label</i>	Branch to <i>label</i> only if there was no successful substitution was made
<i>w filename</i>	Write pattern space to <i>filename</i>
x	Exchange the contents of hold and pattern spaces

bash + sed

- Including sed inside shell script
- heredoc feature
- Use with other shell scripts on command line using pipe

sed is available everywhere !

sed is a meant for text processing, fast in execution

use sed to pre-process input for further processing