

Bash Workshop

Aline Abler



What will we do today?

- ▶ Theory
 - ▶ What is bash?
 - ▶ The basics
- ▶ Exercises

Bash is a not a programming language.

- ▶ It is a **shell**
- ▶ It is an interface between you and your computer
- ▶ It allows you to access the operating system's services (i.e. run programs)
- ▶ **It is not designed as a programming language**, but can be used as such

Bash is not the only shell there is.

- ▶ sh, ksh, zsh, fish, dash. . .
- ▶ Most commands work in any shell, but **some don't**
- ▶ Usually this doesn't matter too much

Bash scripts are not programming

- ▶ Glue existing programs together to create new ones
- ▶ It's possible to write entire programs, but please don't

Look, a bash script

```
#!/bin/bash

echo 'Hello World'

echo Bash is awesome

# Now some fun stuff:
sudo zypper update
notify-send 'Update complete'
feh --bg-fill 'pictures/fancy_wallpaper.jpg'
youtube-dl -o 'Video.%(ext)s' 'https://www.youtube.com/watch?v=lAIGb1lfpBw'
```

What bash can do

- ▶ Automate anything you can do from a console
- ▶ Let several separate programs work together

All about commands

Everything is a command

Kinds of commands

- ▶ **Functions**
are a sequence of bash commands that have been given a name
- ▶ **Builtins**
are small utilities that are part of bash
- ▶ **Programs**
are external programs that are installed on your system
- ▶ **Aliases**
are specific keywords that are replaced by an entire string (interactive mode only)
- ▶ **Keywords**
are like builtins, but with special parsing rules

Why is this so important?

- ▶ `if` and `while` are builtins
- ▶ `[[` is a keyword

All about strings

Everything is a string

Strings and word splitting

- ▶ A string is a sequence of characters that is **treated as a unit**
- ▶ Commands are strings, too
- ▶ Strings are split at every **space, tab, or newline**

Meaning of strings

```
echo Hello World
```

- ▶ `echo`, `Hello` and `World` are single strings
- ▶ The first string becomes the command, all following become **arguments**

```
echo 'Hello World'
```

- ▶ Here, `Hello World` is just one string

Return values

- ▶ What you run

```
echo 'Hello World'
```

- ▶ What you see

```
Hello World
```

- ▶ What bash sees

```
0
```

Why?

- ▶ Every command returns a number, its **return value**
- ▶ 0 means success
Everything else means there was an error
- ▶ Some commands also print to **stdout**

Why is that important?

- ▶ `&&`, `||`, `if` and `while` all act based on the return value of something
- ▶ They disregard that command's actual output

Bash doesn't only run commands

- ▶ Tilde expansion

`~/files` becomes `/home/alinea/files`

- ▶ Variable expansion

`$BROWSER` becomes `Firefox`

- ▶ Arithmetic expansion

`$((1 + 4))` becomes `5`

- ▶ Command substitution

`$(pwd)` becomes `/home/alinea/scripts`

- ▶ Pathname expansion (or globbing)

`files/qui*` becomes `files/quicknotes files/quiz`

Bash doesn't only run commands

- ▶ Expansion happens before any command is run
- ▶ It also happens **before word splitting**

The issue with word splitting

```
var='Unimportant File.odt'  
rm $var
```

becomes

```
rm Unimportant File.odt
```

Repeat after me:

If there's a dollar, you probably
need quotes!

Why is bash awesome?

- ▶ Bash can easily invoke any programs and connect them
- ▶ Bash is like glue

How to write a bash script

- ▶ I want a script that searches youtube and downloads the first video it finds

Splitting it up

- ▶ Search youtube
- ▶ Download video

Splitting it up even further

- ▶ Search youtube
 - ▶ Download youtube results website
 - ▶ Search website text
 - ▶ Find first youtube video link
- ▶ Download video

Google for a program that already does what you need

- ▶ `youtube-dl` can download a video from youtube

```
youtube-dl -o 'Video.%(ext)s' 'https://www.youtube.com/watch?v=1AIGb11fpBw'
```

- ▶ Now I only need to use bash to **glue** everything together

Hands on!

- ▶ Self study using a guide
- ▶ Try some of our exercises

Course material

- ▶ These slides, exercise sheet and bash guide:
<http://thealternative.ch/index.php?view=knowhow>
- ▶ Theme by Christian Horea, CC BY