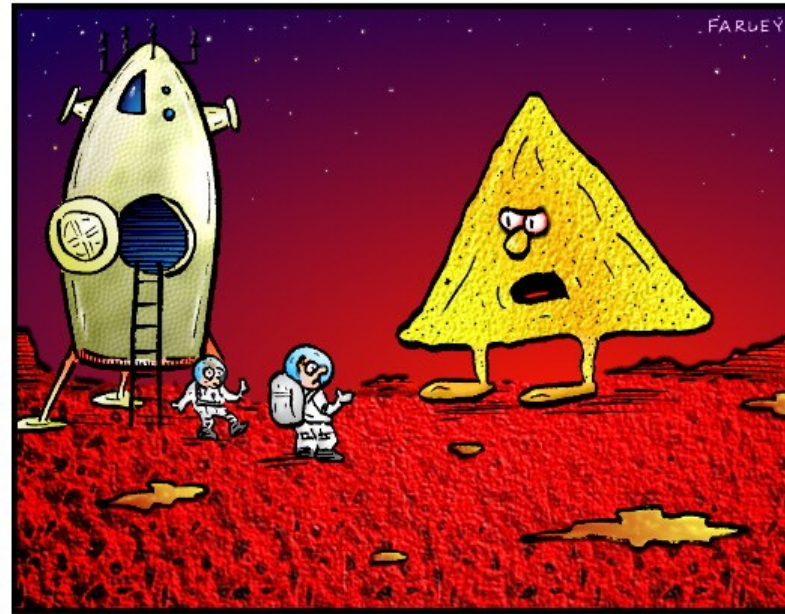


DOCTOR FUN

6 Dec 94



© Copyright 1994 David Farley. World rights reserved.
This cartoon is made available on the Internet for personal viewing only.
dffl@midway.uchicago.edu
Opinions expressed herein are not those of the University of Chicago
or the University of North Carolina.

"This is the planet where nachos rule."

System programming project: NachOS

Vania.Marangozova@imag.fr

Vincent.Danjean@imag.fr

The conclusion after the project

- ▶ *"We got it"*
- ▶ *"it"* being what you have learned during the OS lectures

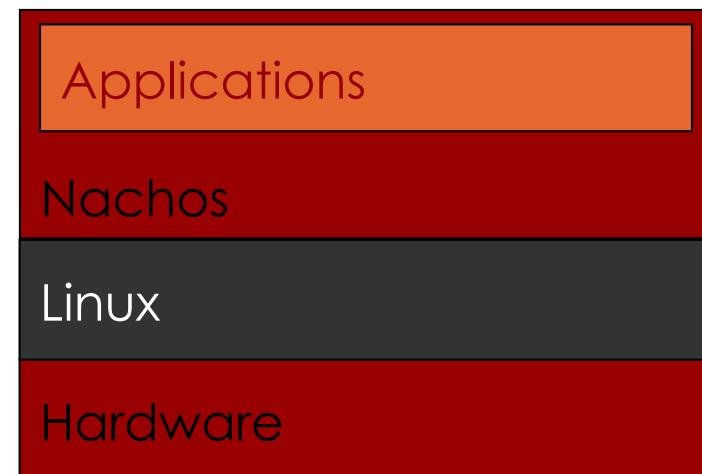
Goals

- ▶ Understand the internals of an OS
- ▶ Work with a big software
- ▶ Implement the major aspects of an OS

- ▶ Group work
 - ▶ project management
 - ▶ one month
 - ▶ groups of 4
 - ▶ planning, specs, docs, version management, ...

Working environment

- ▶ Nachos : a teaching OS
 - ▶ It is simple
 - ▶ It is simulated and therefore deterministic
 - ◆ easy to debug :)



Project steps

- ▶ Step 1 : Getting started
- ▶ Step 2 : I/O
- ▶ Step 3 : Multi-threading
- ▶ Step 4 : Memory management
- ▶ Step 5 : File system
- ▶ Step 6 : Network

Step 1 : Getting started

- ▶ Installation
- ▶ Compilation
- ▶ Running, commands, debugging

```
> userprog/nachos -x test/halt
```

```
Ticks: total 42, idle 0, system 30, user 12
```

```
Disk I/O: reads 0, writes 0
```

```
Console I/O: reads 0, writes 132
```

```
Paging: faults 0
```

```
Network I/O: packets received 0, sent 0
```

```
Cleaning up...
```

Step 2 : I/O

- ▶ What you will work towards to
 - ▶ no libc

```
> userprog/nachos -x test/putstring
We made it! Print a string in NachOS!

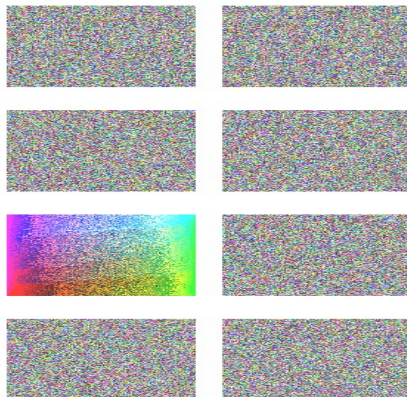
Ticks: total 42, idle 0, system 30, user 12
Disk I/O: reads 0, writes 0
Console I/O: reads 0, writes 132
Paging: faults 0
Network I/O: packets received 0, sent 0

Cleaning up...
```

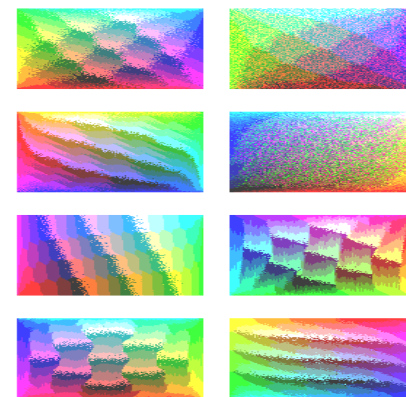
Step 3 : Multi-threading

- ▶ The provided NachOS system can only run one sequential process
- ▶ Goals: propose multi-threading to developers

▶ Sequential



▶ Parallel



Step 4 : Virtual Memory

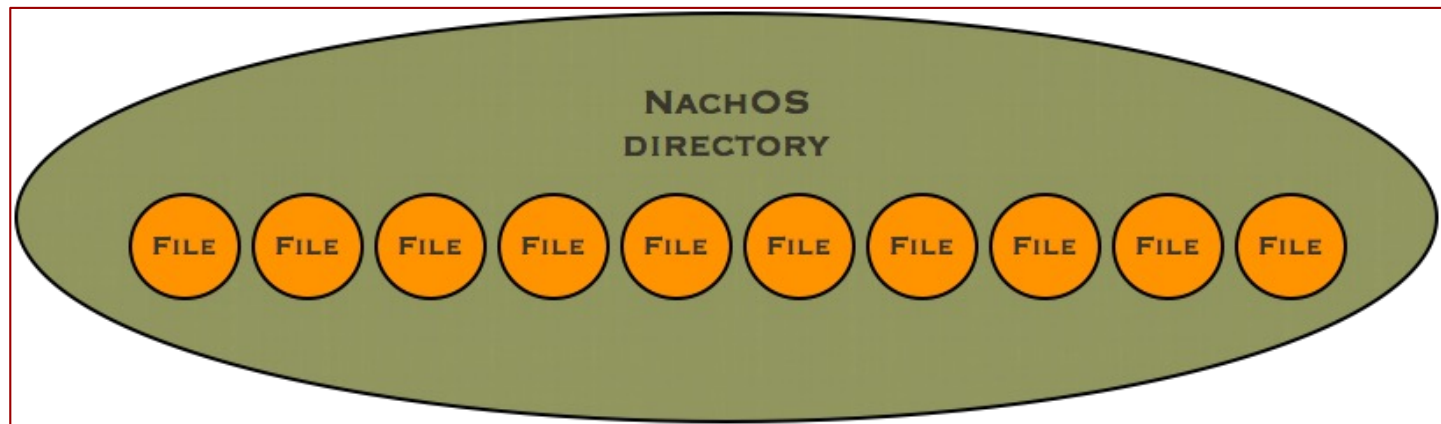
- ▶ Goals : implement multiprogramming
 - ▶ Launch multiple processes
 - ▶ Paging

- ▶ Application : shell

```
> userprog/nachos -x test/myshell
my_shell> help
Available commands:
help
run <pg>
quit
my_shell> run putstring
We made it! Print a string in NachOS!
```

Step 5 : File system

- ▶ The initial NachOS filesystem has
 - ▶ one directory
 - ▶ max 10 files in it
 - ▶ with a limited size



Step 5 : Goals

```
> userprog/nachos -x test/myshell
my_shell> pwd
nachos_root
my_shell> ls
test      divers      video
my_shell> cd test
my_shell> pwd
nachos_root/test
my_shell> cd ..
my_shell>
```

Step 6 : Network

- ▶ It is possible to launch multiple NachOS machines and connect them in a network

- ▶ Goals
 - ▶ Implement a reliable communication protocol (TCP inspired)

- ▶ Applications
 - ▶ client-server
 - ▶ FTP
 - ▶ Process migration
 - ▶ ...

To conclude

- ▶ A challenging project that makes you go through (understand, design, implement, test) the major parts of an OS
- ▶ Compulsory parts and bonuses (there is no limit)
- ▶ The more you play with it, the more expert you become, the more interesting it gets!
- ▶ The number of lines of code is quite limited
- ▶ Time is always short

- ▶ Evaluation : weekly progress meetings + git follow ups + defenses (code, documentation, oral presentation, demo)