EECS 2031

Software Tools

Module 3 – Unix under the hood

YORK UNIVERSITÉ UNIVERSITY

Processes

- Each running program on a UNIX system is called a process.
- Processes are identified by a number (process id or PID).
- Each process has a unique PID.
- There are usually several processes running **concurrently** in a UNIX system.

YORK UNIVERSITÉ UNIVERSITY

ps command

```
% ps a
```

list all processes

```
  PID TTY      STAT   TIME COMMAND
 2763 pts/11   S+     0:10 pine
14468 pts/19   R+     0:00 ps
14780 pts/21   S      0:00 xterm
26772 pts/2    S+     0:01 emacs
  ...
```


YORK UNIVERSITÉ UNIVERSITY

Background processes

- A process may be in the foreground, in the background, or be suspended
- To see all background processes: **`jobs`**
- To bring a process to the foreground: **`fg`**
- To suspend the foreground process: **`CTRL-Z`**
- Put all suspended processes to the background: **`bg`**

YORK UNIVERSITÉ UNIVERSITY

kill

```
% kill -KILL PID
```

to terminate a process

```
% kill -STOP PID
```

to suspend a process

YORK UNIVERSITÉ UNIVERSITY

Process-related Terminal Keystrokes

- Kill the foreground process: **`CTRL-C`**
- Suspend the foreground process: **`CTRL-Z`**

YORK UNIVERSITÉ UNIVERSITY

Multiple process example

- What processes will be created if I run this script?

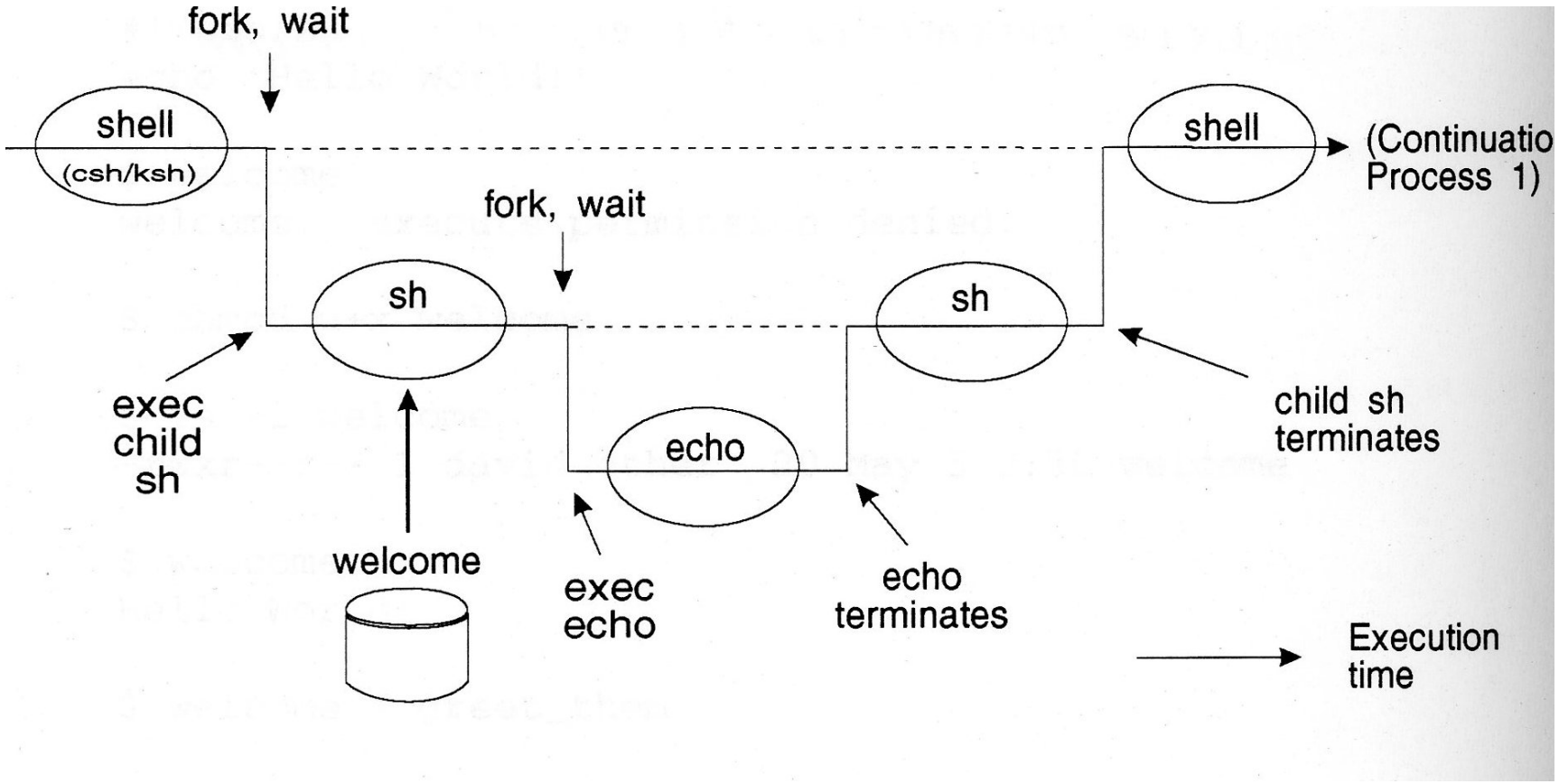
```
% cat welcome

#!/bin/sh

echo Hello World!
```


YORK UNIVERSITÉ UNIVERSITY

Processes in the last example



YORK UNIVERSITÉ UNIVERSITY

Processes: Explanation

- Every process is a “child” of some other process
- Your login shell fires up a child shell to execute the script
- The child shell fires up a new (grand)child process for each command.
- The parent shell sleeps while child executes.

YORK UNIVERSITÉ UNIVERSITY

Processes: Explanation

- Every process has a unique PID.

- Parent does not sleep while running background processes.

YORK UNIVERSITÉ UNIVERSITY

Process-Related Variables

- Variable **$$** is PID of the shell.

```
% shpid
  PID  TTY          TIME CMD
 5658 pts/75    00:00:00 shpid
 5659 pts/75    00:00:00 ps
11231 pts/75    00:00:00 tcsh
PID of shell is = 5658
```


YORK UNIVERSITÉ UNIVERSITY

Process Exit Status

- As we saw already, `$?` is a process-related variable that returns the exit status of the last process to terminate
- Good practice: Specify your own exit status in a shell script using the **`exit`** command.
  - If no exit code is given, 0 is returned

YORK UNIVERSITÉ UNIVERSITY

Environment and Shell Variables

- Shell variables: apply only to the current instance of the shell; used to set short-term working conditions.
  - displayed using **`set`** command.
- Environment variables: set at login and are available to all shells
  - displayed using **`printenv`** command.

YORK UNIVERSITÉ UNIVERSITY

Environment and Shell Variables

- By convention, environment variables have UPPER CASE and shell variables have lower case names.
  - Examples: **`HOME`** is an environment variable, **`home`** is a shell variable
- Most of the time, one deals with environment variables, unless you want distinct behaviour only in the current shell

YORK UNIVERSITÉ UNIVERSITY

PATH

- PATH is an environment variable that specifies directories to search for commands and programs
- Try **`echo $PATH`** in your account
- Add to the value of **`PATH`** with something like

```
setenv PATH ${PATH}:/cs/fac/bin
```


YORK UNIVERSITÉ UNIVERSITY

`.cshrc`

- To add a path permanently, add the last line of the previous slide to the end of the `.cshrc` file in your home directory

YORK UNIVERSITÉ UNIVERSITY