

# Administration de clients diskless : un cas d'école

*Le réseau premier cycle de l'université Paris  
XI Orsay  
Aspect général et pédagogique*

BURELLE Marwan LETOUZEY Pierre

burelle@lri.fr letouzey@lri.fr

# Introduction

# La situation

- $\simeq$  140 clients diskless
- 3 systèmes d'exploitation différents
- 2 couches réseaux (*IP* et *NetWare/IPX*)
- plusieurs configurations matérielles différentes...
- 5 ou 6 formations différentes...
- Utilisation continue de septembre à juillet...
- ...

# Motivations et problématique

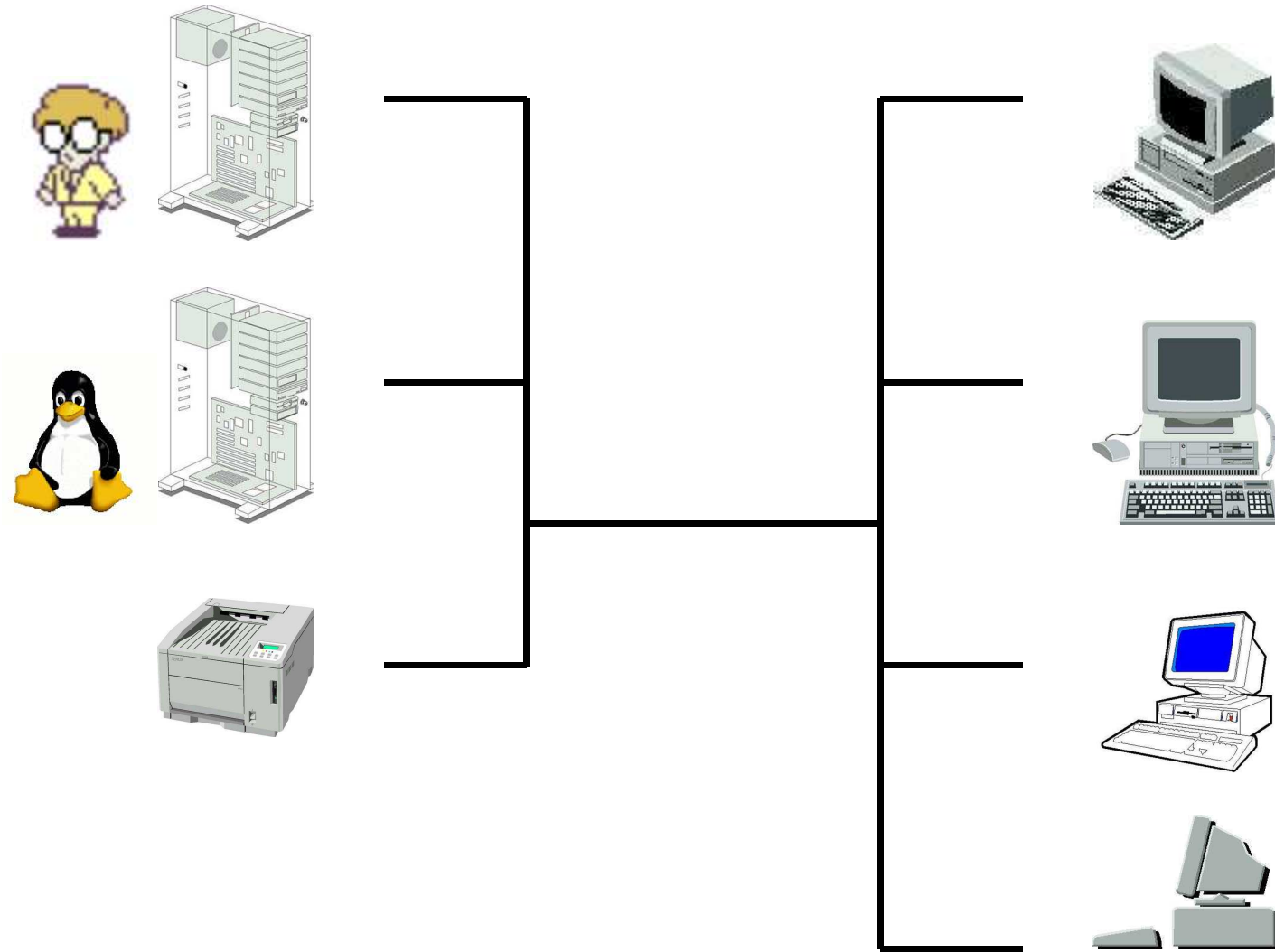
## Motivation

- Familiarisation avec les systèmes UNIX
- TP de programmation dans un environnement UNIX (*Emacs, OCaml...*)

## Problématique

- Machine diskless (sans disque dur)
- Réseau conçu à l'origine pour NetWare et Windows
- Matériel hétérogène

# Architecture en place



# Historique

## Début : 1998/1999

- Précurseurs : *Judicaël Courant, Claude Marché, Benjamin Monate et Ralf Treinen*
- Expérimentation en *Étude de cas* (langage *Caml Light*)
- TP du cours *LGL* (deuxième semestre)

## Le vrai démarrage : 1999/2000

- Initiation à Linux pour le premier semestre de DEUG
- TP de second semestre de DEUG (*OCaml*)

# Historique (2)

## Contraintes fortes du réseau NetWare :

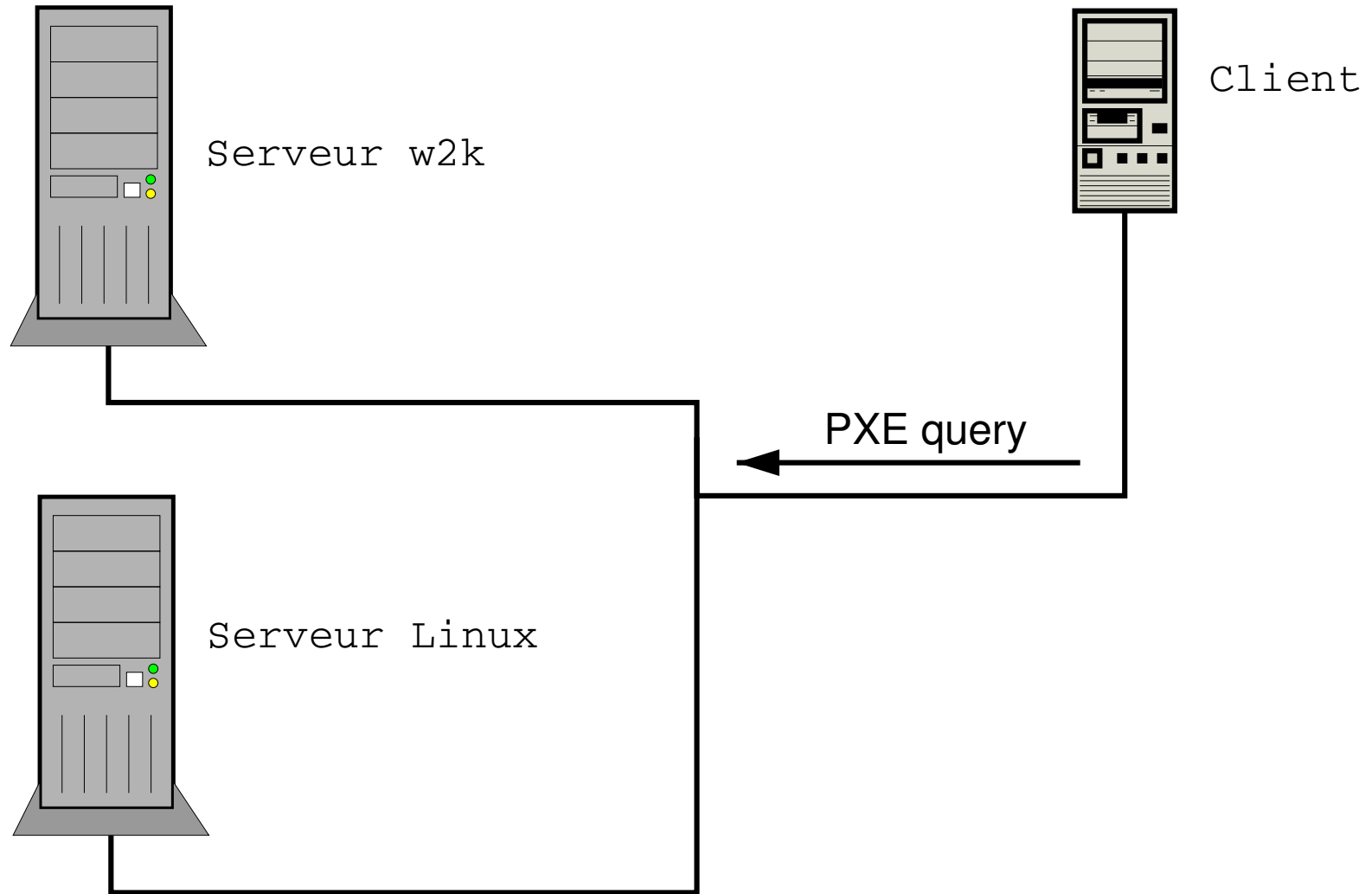
- Démarrage via disquette
- Interaction avec le réseau NetWare
- Passage de la distribution RedHat à la distribution Debian

# Solutions actuelles

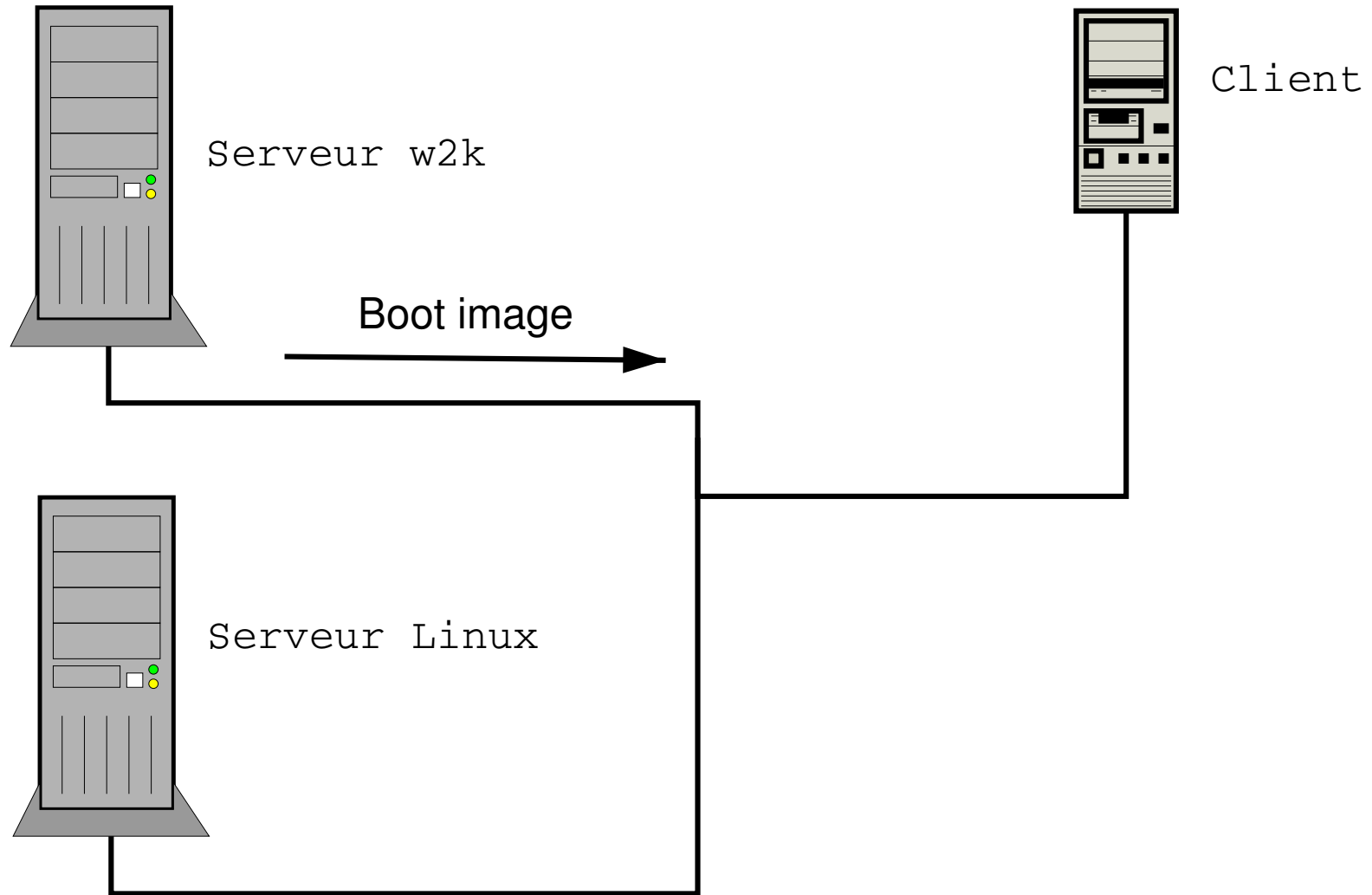
# Le système

- Distribution *Debian* relativement standard
- Boot réseau (*PXE*) → choix du système d'exploitation
- Boot kernel via *Grub*
- Système de fichier en *NFS*, *tmpfs* et *devfs*
- Impression via *CUPS*

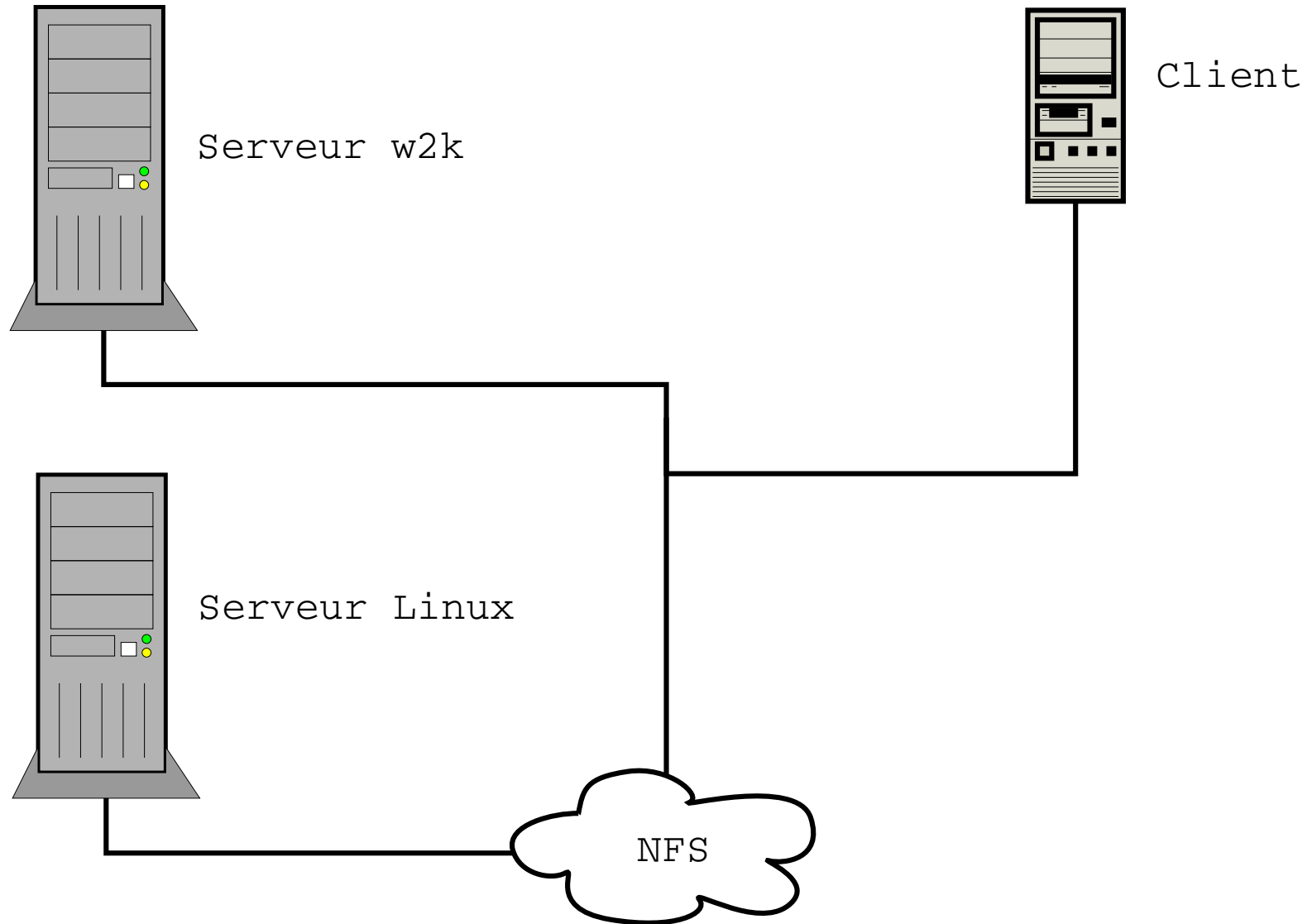
# Le boot



# Le boot



# Le boot



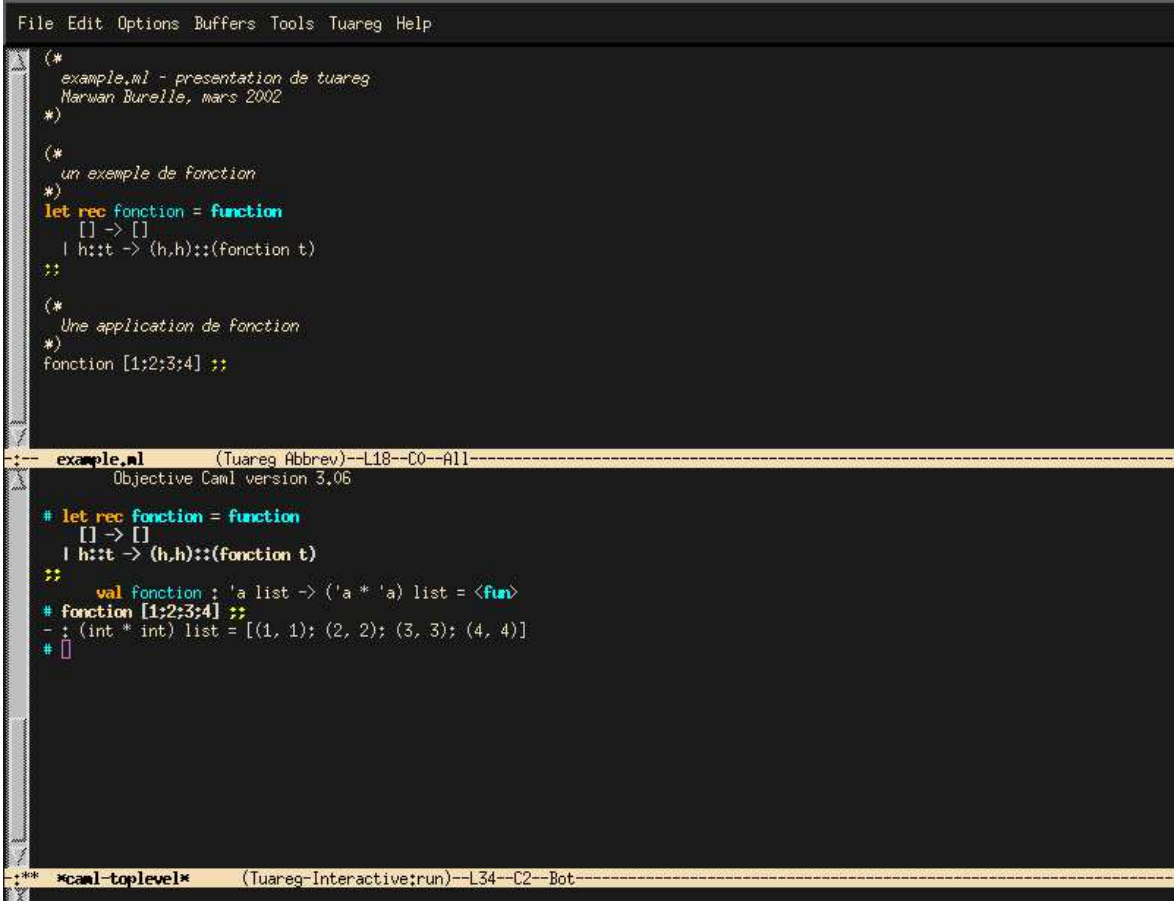
# Enseignement

# Initiation à Linux

- Sensibilisation aux contraintes des environnements multi-utilisateurs (*login, mots de passe...*)
- Présentation du système de fichiers
- Manipulation des fichiers et ligne de commandes
- Éditeur *Emacs*
- ...

# OCaml

- Langage développé a l'INRIA  
<http://caml.inria.fr>
- Édition via Emacs et le mode tuareg



```
File Edit Options Buffers Tools Tuareg Help
(*
  example.ml - presentation de tuareg
  Marwan Burelle, mars 2002
*)
(*
  un exemple de fonction
*)
let rec fonction = function
  [] -> []
  | h::t -> (h,h)::(fonction t)
;;
(*
  Une application de fonction
*)
fonction [1;2;3;4] ;;

-- example.ml (Tuareg Abbrev)--L18--C0--A11--
Objective Caml version 3.06

# let rec fonction = function
  [] -> []
  | h::t -> (h,h)::(fonction t)
;;
  val fonction : 'a list -> ('a * 'a) list = <fun>
# fonction [1;2;3;4] ;;
- : (int * int) list = [(1, 1); (2, 2); (3, 3); (4, 4)]
#

--** *caml-toplevel* (Tuareg-Interactive;run)--L34--C2--Bot--
```

# PixArchi

- Outil pédagogique développé par des étudiants de maîtrises d'Orsay (1999/2000).
- Simulateur d'architecture *RISC*
- Visualisation de la circulation de l'information dans les bus d'un processeur
- Programme écrit en *Java*

# Projets

# Interaction avec Windows

- Intégration des disques partagés par Windows pour les clients Linux (client *samba*)
- Partage de fichier Linux vers Windows (serveur *samba*)
- Accès aux données utilisateurs avec authentification
- Authentification commune Windows/Linux ?

# *XtremWeb*



- Projet de calcul distribué
- Pourquoi ne pas profiter des machines inutilisées la nuit ?
  - ⇒ Mise en place du client *XtremWeb*
  - ⇒ NAT pour la communication vers l'extérieur
  - ⇒ Réveil automatique des machines la nuit (*Wake On LAN*)