

# Noyau Linux et multi- processeur sur plateforme embarquée

## ST50

Vendredi 18 Mars 2005 – Thomas Petazzoni

## Mitsubishi Electric

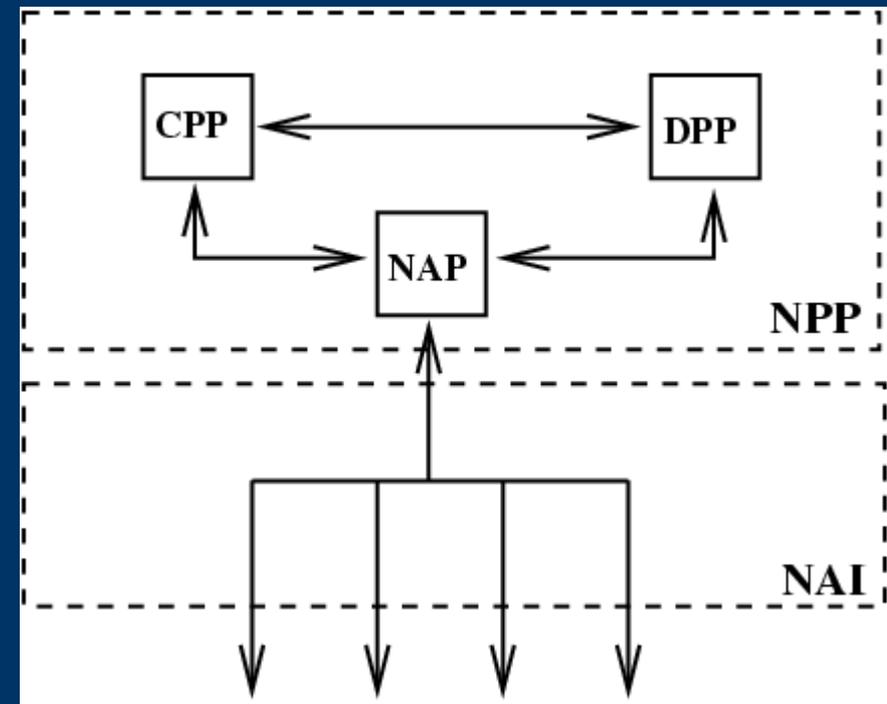
- 99.000 employés
- 28 milliards d'Euros de CA
- banque, assurance, automobile, pétrole, chimie, transport, immobilier, électronique ...

## Mitsubishi Electric ITE TCL

- Laboratoire de recherches en télécommunication
- Fondé en 1995, compte 33 salariés, 22 ingénieurs
- Trois équipes
  - télécoms et systèmes
  - **protocoles**
  - communications numériques

## Plateforme FlexNP

- **prototyper** des fonctionnalités réseau et surveiller un réseau : filtrage, QoS, détection d'intrusions ... à haut débit (Gigabit/s)
- **plateforme matérielle**
  - Carte NPP
    - Control Plane Processor : processeur MIPS PMC-Sierra bi-processeur
    - Data Plane Processor : processeur ARM/XScale
    - Network Access Processor : FPGA Xilinx
  - Carte NAI : interfaces réseau
- **logiciel** : noyaux Linux 2.4.X



# Carte NPP



Travail sur la partie CPP de la carte, avec le processeur MIPS.

Déroulement en trois étapes :

1. Chaînes de compilation croisée et *ramdisk*
2. Portage du noyau Linux 2.6
3. Mise en place du fonctionnement multi-processeur

# Cross-compilation et ramdisk

- **Génération de chaînes de compilation croisée**: ensemble d'outils permettant sur une machine hôte (architecture x86) de compiler du code pour une machine cible (architecture MIPS)
- **Génération de *ramdisk*** : système de fichiers minimal, chargé depuis le réseau ou une mémoire Flash, et permettant le fonctionnement de la plateforme

## Étapes

- recherche de l'outil adapté : Buildroot
- personnalisation
- mise en place pour l'équipe

# Portage du noyau Linux

- Au début du stage : noyau 2.4.19 personnalisé pour la partie MIPS
- Noyau 2.6
  - Plus de pilotes de périphériques
  - Plus adapté au multi-processeur
- Étapes du portage
  - Code d'initialisation
  - Pilote série
  - Pilote Ethernet
- Développement difficile
  - Proche du matériel
  - Matériel « à la pointe » et peu utilisé
  - Documentations difficiles à exploiter

# Multi-processeur 1/2

- MIPS bi-processeur utilisé jusqu'à maintenant en uni-processeur
- Choix d'un fonctionnement asymétrique plutôt que du SMP classique
  - Le premier processeur exécute le noyau Linux de manière conventionnelle
  - Le second processeur exécute un code personnalisé, sans système d'exploitation

## – Développement

- du code d'initialisation du deuxième processeur dans le bootloader de la plateforme en assembleur MIPS
- d'un module noyau
  - démarrage et arrêt du second processeur
  - chargement du code à exécuter
  - affichage de messages depuis le second processeur
  - gestion des exceptions du second processeur
- d'une application utilisateur s'interfaçant avec le module noyau

## – Développement et débogage difficile

## Sur le plan technique

- découverte d'une architecture matérielle spécifique
- fonctionnement interne du noyau Linux et multi-processeur
- intégration dans une équipe d'un laboratoire de recherches en télécommunication
- difficultés du développement bas niveau

## Sur le plan humain

- bonne intégration dans l'équipe
- travail en collaboration forte avec l'équipe

# Questions ?