

---

# Systemes Embarqués

---



ROB 2A, UE 4.1, ENSTA

---

# Introduction

---

SYSTÈMES EMBARQUÉS

# Plan

---

1. Logiciel Libre et Logiciel Open-Source
2. Licences de distribution
3. Définitions et propriétés
4. Comment choisir un OS embarqué ?
5. Linux embarqué
6. Lab 1 - Setting-up Raspberry Pi OS Image

# Logiciel libre et logiciel open-source (1)

---

► Logiciel Libre : code source ouvert et pouvant être modifié. Fond social => liberté des utilisateurs (Free as Freedom)!

► Logiciel Open Source : code source ouvert et pouvant être modifié... Fond pragmatique => efficacité, praticité!

<http://www.gnu.org/philosophy/free-software-for-freedom.fr.html>

# Logiciel libre et logiciel open-source (2)

---



Organisation dont le but est de :

- promouvoir les logiciels libres
- protéger les utilisateurs de la privatisation des logiciels
- élaborer des licences de distribution

# Licences de distribution (1)

---

## **Copyleft**

L'utilisateur refuse qu'une évolution quelconque de son travail soit accompagnée d'une restriction



# Licences de distribution (2)

---

## **Licences libres Copyleft :**

- GPL : GNU General Public Licence. Édition de liens possible qu'avec du code GPL  
<http://www.gnu.org/licenses/gpl-3.0.fr.html>
- LGPL : Lesser GPL. Édition de liens moins restrictive  
<http://www.gnu.org/licenses/lgpl-3.0.fr.html>

**Licences libres non-Copyleft :** les versions modifiées ne sont elles mêmes pas nécessairement libres

- BSD
- MIT License
- ...

# Licences de distribution (3)

---

## **Exemple de logiciels sous licences :**

- nmap: GPL /usr/share/doc/nmap/copyright
- GNU C library : LGPL /usr/share/doc/libc6/copyright
- roplib : MIT License



# Définitions et propriétés (1)

---

## **Kernel :**

- noyau d'un système d'exploitation
- gère les ressources matérielles
- permet la communication entre composants logiciels/matériels
- existe plusieurs architectures

## **Système d'exploitation :**

- Kernel + logiciels comme compilateur, shell, debugger, ...
- couche d'abstraction par rapport au matériel
- interface générique de programmation

# Définitions et propriétés (2)

---

## **Système embarqué :**

- composition d'une partie électronique et logicielle
- souvent très limitée d'un point de vue ressources matérielles (CPU, mémoire, ...)
- Autonome
- doit respecter des contraintes d'environnement (vibration, chaleur, ...)

## **Système d'exploitation embarqué :**

- OS sur lequel un logiciel embarqué va être exécuté.
- Contrainte forte par rapport à la consommation matérielle / énergétique.
- OS classique souvent inenvisageable

# Définitions et propriétés (3)

---

## Linux embarqué :

- Système d'exploitation embarqué basé sur un Kernel Linux
- Kernel Linux + composants open-source
- construit sur mesure par rapport aux besoins

## Remarque :

Initialement le nom **Linux** ne désignait que le Kernel Linux, aujourd'hui il est autant utilisé pour désigné le kernel que le système d'exploitation GNU/Linux



# Comment choisir un OS embarqué (1)

---

## **OS embarqués existants :**

- sans base Linux : VxWorks, QNX, micro-C OS, Windows Phone, ...
- à base Linux : Debian, Raspberry Pi OS, Ubuntu Core, Google Android, Tizen, Wind River Linux, ELDK, ...

## **Comment choisir ?**

# Comment choisir un OS embarqué (2)

---

*Choisir entre système propriétaire et système libre ou open-source*

## **Système propriétaire :**

Les -

- Cher
- Personnes maîtrisant les outils rares
- Dépendance à l'entreprise propriétaire

Les +

- Bon support et réactivité
- Garantie en cas de problème  
--> Responsabilité de l'entreprise

## **Système libre ou open-source :**

Les -

- Logiciel fourni "As is"  
--> Pas de garantie de bon fonctionnement  
--> Problème de responsabilité
- les licences trop ouvertes (comme la GPL) peuvent poser problèmes pour les entreprises ne souhaitant pas reverser leurs travaux

Les +

- Code source ouvert et donc modifiable à volonté
- Code souvent de meilleure qualité grâce à de nombreux reviewers

# Linux embarqué

---

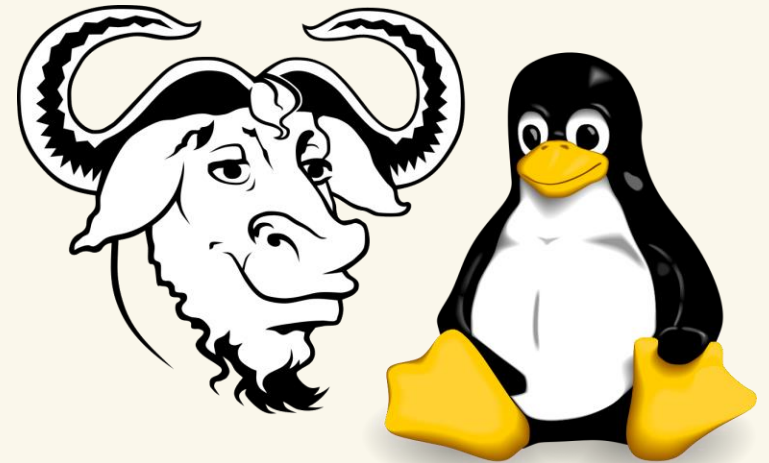
GNU/Linux est reconnu pour sa très grande fiabilité

Avantages :

- Peu cher : pas de royalties, outils de développement libres
- Portabilité : x86, arm, ppc, amd, sparc
- Open Source

Inconvénients :

- Beaucoup de licences différentes
- Beaucoup de solutions pour faire la même chose contrairement à une solution propriétaire sur étagère



# Lab 1 – Setting up Raspberry Pi OS image

---

Raspberry Pi OS (anciennement nommé Raspbian) est un OS Linux basé sur Debian, lui-même une distribution de GNU/Linux, optimisé pour les Raspberry Pi

L'objectif de ce premier Lab est la création et modification d'une image Raspberry Pi OS directement sur votre machine.

