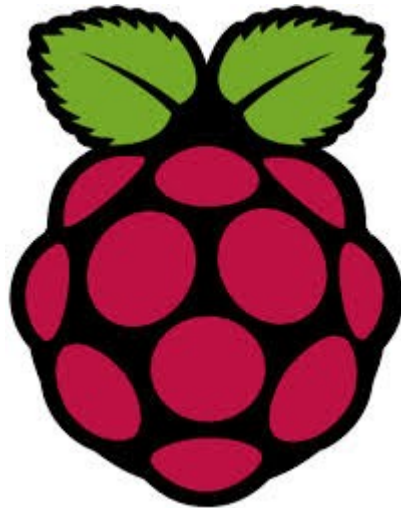


*Raspberry pi :*  
*Développer une petite application web*  
*sur Raspberry*



## Introduction

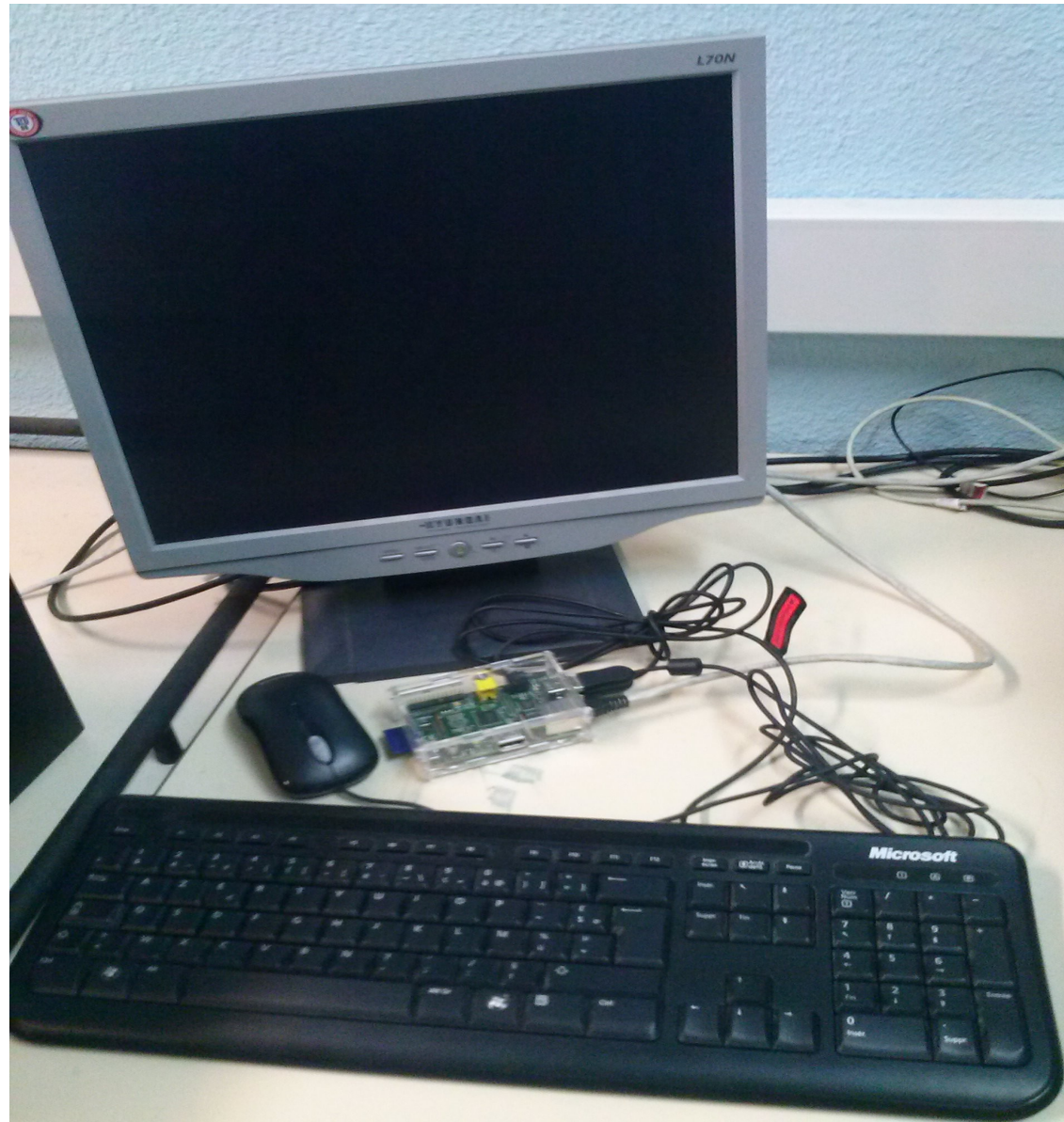
*Le **Raspberry Pi** est un nano-ordinateur basé sur une architecture ARM (conçu par David Braden) qui permet l'exécution de plusieurs variantes du système d'exploitation libre Linux et des logiciels compatibles. Destiné à encourager l'apprentissage de la programmation informatique, il est fourni nu c'est à dire la carte mère seule (dans le but de diminuer les coûts et de permettre l'utilisation de matériel de récupération).*

*Cette machine, de la taille d'une carte de crédit, est idéale pour fonctionner en serveur.*

*On appelle **serveur Web** aussi bien le matériel informatique que le logiciel, qui joue le rôle de serveur informatique sur un réseau local. Nous allons donc l'utiliser pour installer un serveur Web sous Raspbian «wheezy» sur le réseau local.*

# *I) Préparation du Raspberry (de base)*

## *a) Le branchement*



## *b) Installer Rasbian « wheezy »*

Il s'agit du système d'exploitation associé recommandé par le fabricant du Raspberry Pi.

Il faut le télécharger sur le site suivant :

<http://www.raspberrypi.org/downloads>.

Et copier ensuite l'image du système [2013-05-25-wheezy-raspbian.img](#) sur une carte SD.

## *c) Préparer la carte SD*

Sous Linux, il faut d'abord déterminer le chemin du périphérique correspondant à la carte SD avec la commande **df** (disk free : espace libre et utilisé par les partitions).

```
$ df -h.../dev/sdb1 299G 70G 229G 36% /media/DONNEES  
/dev/sdc1 3,9G 0 3,9G 0% /media/ADFB-008C
```

Chercher la ligne correspondant à la carte mémoire : il s'agit de la ligne avec la valeur qui se rapproche le plus de la carte SD, ici 3,9Go pour une carte de 4Go.

Puis, démonter la partition :

```
$ sudo umount /dev/sdc
```

Maintenant, vous pouvez copier l'image sur le carte SD :

```
$ sudo dd bs=1M if=~/.2013-05-25-wheezy-raspbian.img of=/dev/sdc
```

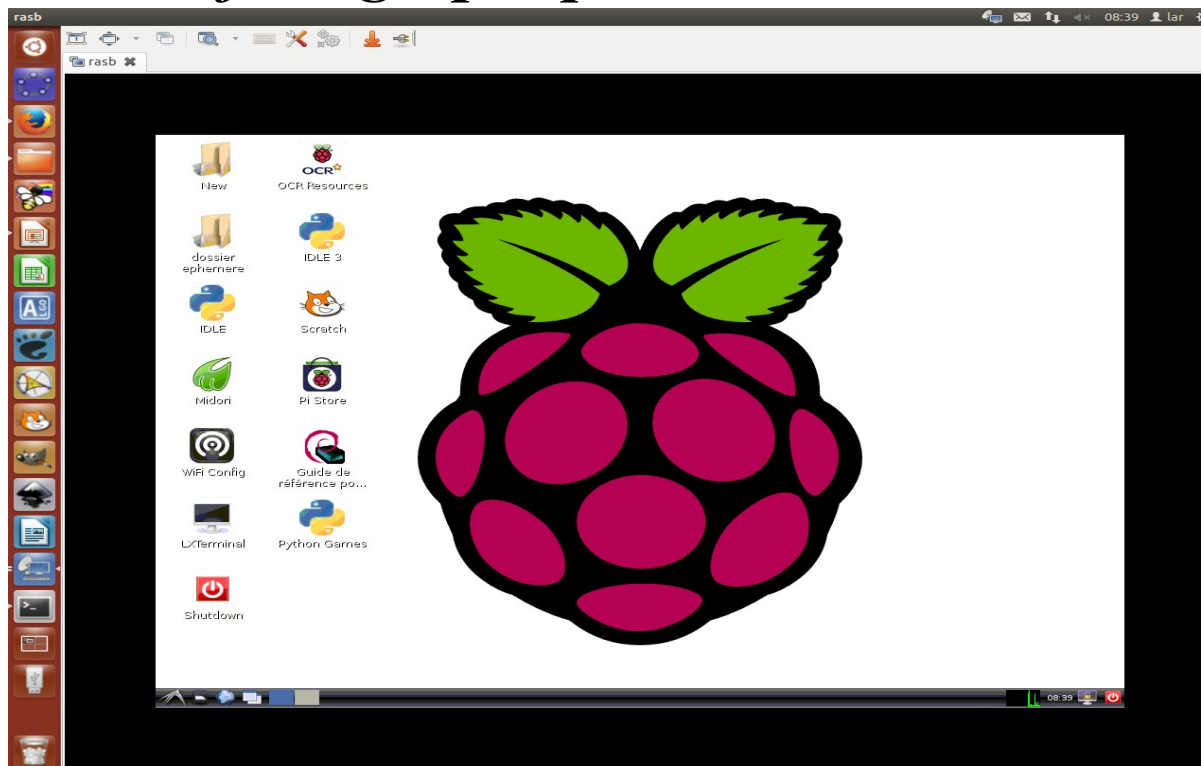
## *d) Le premier démarrage*

*Insérer la carte SD*

*login : **pi***

*mot de passe : **rasberry***

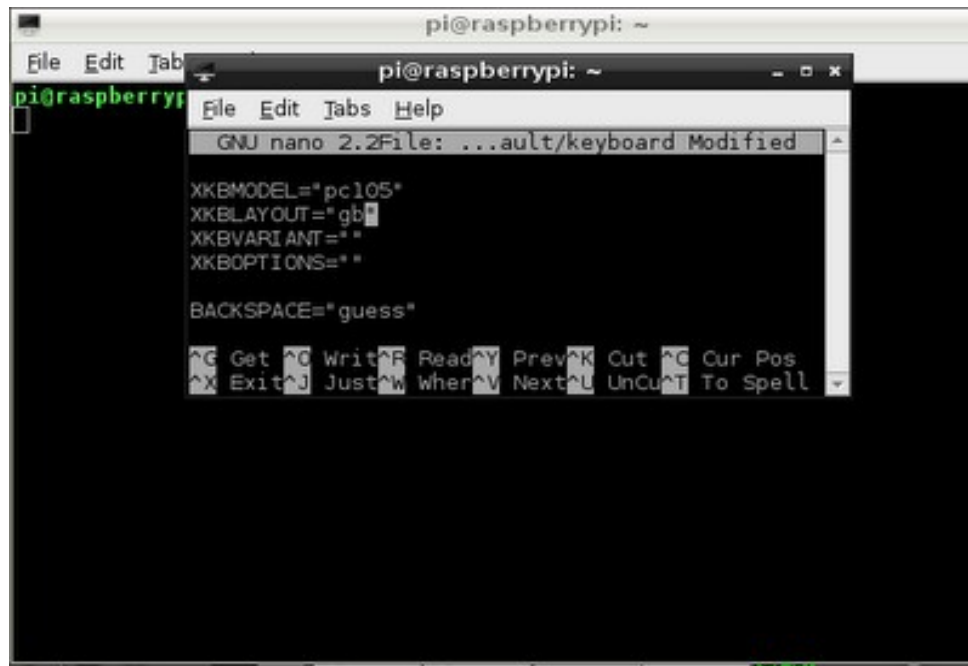
*Pour afficher l'interface graphique: **startx***



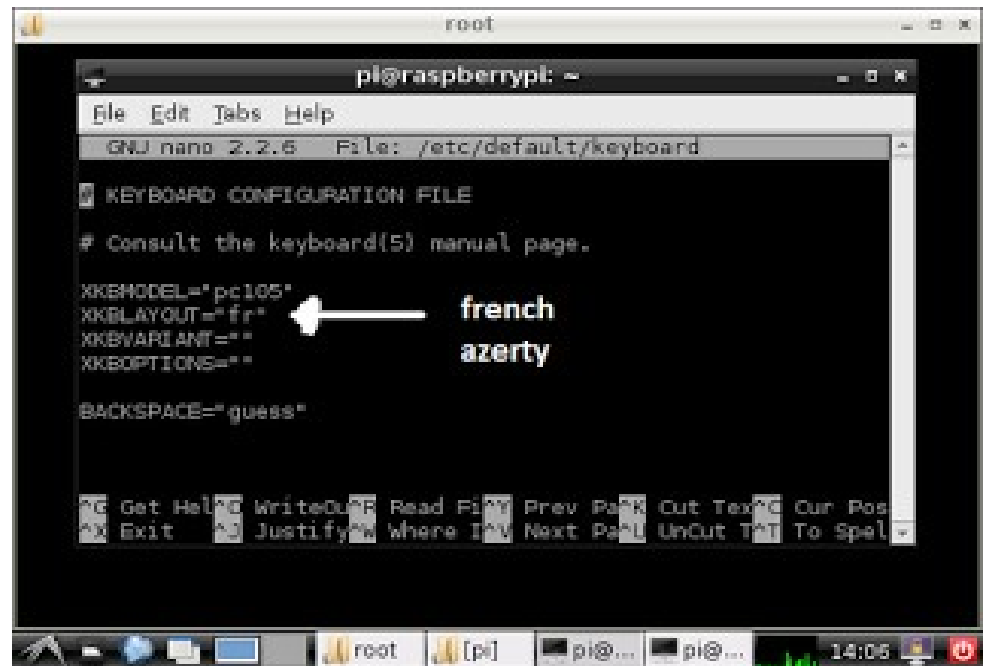
## e) Configurer le raspberry

### Mettre le clavier en AZERTY

*sudo nano /etc/default/Keyboard*



```
pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.2.6 File: ...ault/keyboard Modified
XKBMODEL='pc105'
XKBLAYOUT='gb'
XKBVARIANT=''
XKBOPTIONS=''
BACKSPACE='guess'
^G Get ^O Write ^R Read ^Y Prev ^K Cut ^C Cur Pos
^X Exit ^J Justif ^W Where ^V Next ^L UnCut ^T To Spell
```



```
root
pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.2.6 File: /etc/default/keyboard
KEYBOARD CONFIGURATION FILE
# Consult the keyboard(5) manual page.
XKBMODEL='pc105'
XKBLAYOUT='fr'
XKBVARIANT=''
XKBOPTIONS=''
BACKSPACE='guess'
^G Get Help ^O WriteOut ^R Read Fil ^Y Prev Pa ^K Cut Tex ^C Cur Pos
^X Exit ^J Justify ^W Where In ^V Next Pa ^L UnCut ^T To Spell
```



## Mettre le système en français

*sudo raspi-config*

### Raspi-config

info	Information about this tool
expand_rootfs	Expand root partition to fill SD card
overscan	Change overscan
configure_keyboard	Set keyboard layout
change_pass	Change password for 'pi' user
change_locale	Set locale
change_timezone	Set timezone
memory_split	Change memory split
overclock	Configure overclocking
ssh	Enable or disable ssh server
boot_behaviour	Start desktop on boot?
update	Try to upgrade raspi-config

<Select>

<Finish>

## *Faire les mises à jour*

*sudo apt-get update*

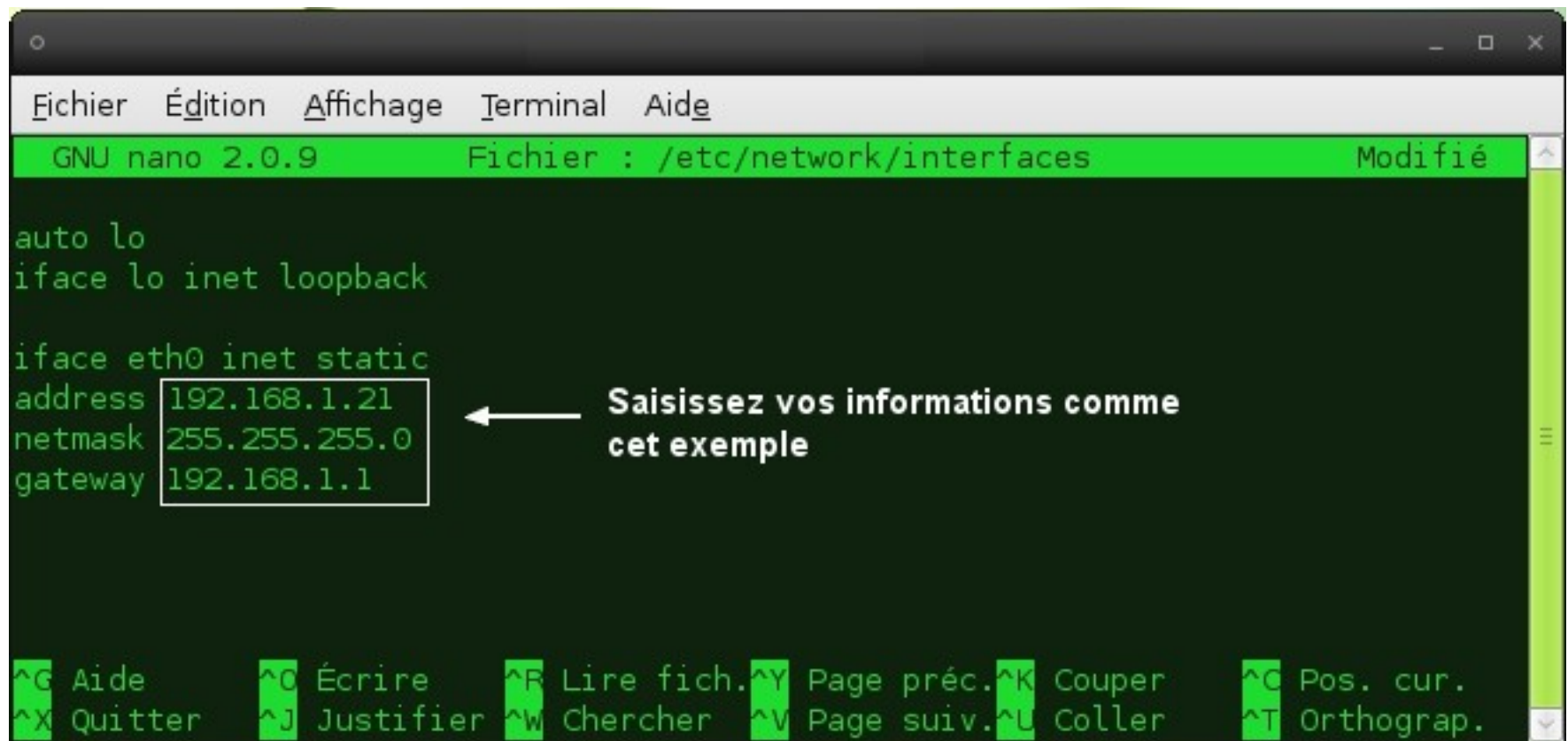
*sudo apt-get upgrade\_*

## *II) Préparation plus approfondie*

## *a) Créer une adresse IP fixe*

*Nous avons créé une adresse IP fixe, c'est la pratique de base pour un serveur web. Cette adresse IP reste la même ce qui facilite son identification et ses paramétrages .*

*sudo etc/network/interfaces*



The screenshot shows a terminal window with the nano text editor open. The title bar indicates the file being edited is `/etc/network/interfaces`. The editor shows the following configuration:

```
auto lo
iface lo inet loopback

iface eth0 inet static
address 192.168.1.21
netmask 255.255.255.0
gateway 192.168.1.1
```

A white box highlights the IP configuration lines for `eth0`. A white arrow points from the text "Saisissez vos informations comme cet exemple" to the IP address `192.168.1.21`. The bottom of the terminal shows the nano editor's command palette with various shortcuts like `^G Aide`, `^O Écrire`, etc.

## *b) Installer SSh (Secure Shell)*

*Le serveur SSH permet d'établir un contrôle sécurisé à distance avec le Raspberry pi (accéder au système grâce au terminal).*

*sudo apt-get install openssh-server*

*pour l'activer : sudo service ssh start*

*pour arrêter : sudo service ssh stop*

*pour relancer : sudo service ssh restart*

## *c) Installer VNC (Virtual Network Computing)*

*Le serveur VNC permet lui d'accéder au système via un autre ordinateur, celui-ci est utilisé comme écran à distance.*

*sudo apt-get install tightvncserver*

*pour activer vnc sur le Raspberry pi: vncserver :1*

*Il faut l'utiliser en parallèle avec le logiciel libre Remmina =>*

*172.18.40.26.1*

## *d) Installer le serveur FTP (Pure-FTPd)*

*Le serveur FTP (File Transfer Protocol) est un protocole utilisé sur Internet pour l'échange de fichiers, il permet de récupérer et d'envoyer des fichiers sur un serveur .*

*sudo apt-get install pure-ftpd*

*=>FileZilla est un client polyvalent et compatible ainsi que simple d'utilisation. (organisation)*

### *III) Le serveur LAMP*

*Un serveur LAMP est serveur Web. Il fait référence aux quatre composantes d'un tel serveur Web :*

*Linux : le système d'exploitation constituant la base du système.*

*Apache : le serveur HTTP qui gère la communication avec le client.*

*MySQL : le système de gestion de base de données.*

*PHP : le langage de script utilisé pour générer les pages dynamiques.*



## *a) Le serveur HTTP (Apache)*

*Apache est le plus populaire des serveurs HTTP. Il est produit par la « Apache Software Foundation » (c'est un logiciel libre fourni sous la licence spécifique Apache) .*

***sudo apt-get install apache2***

*Le contenu de cette page provient du fichier **/var/www/index.html**. Les sites Web seront gérés par Apache et devront se situer à l'emplacement **/var/www**.*

*Nous avons configuré Apache, c'est à dire : tirer root en tant qu'utilisateur pour avoir mains mise sur les droits, ceci afin d'accéder librement et de pouvoir faire les modifications désirés (exemple : renommer les dossiers)*

***sudo leafpad /etc/apache2.conf/***

## *b) Le système MySQL et le langage PHP*

*sudo apt-get install php5*

*sudo apt-get install mysql-server*

*sudo apt-get install libapache2-mod-php5*

*sudo apt-get install php5-mysql*

*sudo apt-get install phpmyadmin*

*(mot de passe sql: qwa123ty)*

## Conclusion

*J'ai pu réaliser mon serveur web malgré mon manque de connaissances en informatique.*

*Grâce à ce projet, j'ai pu acquérir certaines bases en informatique. Créer un serveur web a été pour moi, un projet très captivant et a sollicité toutes mes capacités et beaucoup de mon temps.*

*J'ai pu découvrir ce qu'était un Raspberry pi, le manipuler mais aussi découvrir différents langages, systèmes et méthodes dans le cadre de la création de serveur. De plus, ce projet m'a permis de mettre en avant mon site Web réalisé au cours de l'année (« CILAOS »).*

# Sources

*<http://fr.wikipedia.org/wiki/>*

*<http://www.ubuntu-fr.org/>*

*<http://blog.univ-reunion.fr/academieisn/tag/raspberry-pi/>*

*<http://www.epingle.info/?p=1873>*