

Truc LF: #defines pour écrire du code C portable



par Guido Socher ([homepage](#))

L'auteur:

Guido adore vraiment lorsqu'un ordinateur est taillé à ses besoins et ressemble à ce qu'il veut. C'est pourquoi il utilise Linux.

Traduit en Français par:

Jean-Etienne Poirrier
([homepage](#))



Résumé:

Ceci est une petite astuce. A partir de maintenant, LinuxFocus aura au-moins une nouvelle astuce tous les mois. Si vous avez des idées de nouvelles astuces, envoyez-les à guido@linuxfocus.org

Introduction

Java est connu pour son « écrit une fois, jamais fonctionnel » et, en C, c'est plutôt « écrit une fois, compilé et toujours fonctionnel ». En d'autres mots, du code C est très portable et la beauté est aussi que vous pouvez avoir des déclarations pré-processeur #ifdef pour faire la distinction entre les différentes plate-formes. De cette manière, vous avez juste une pièce de code qui peut être recompilé sur différents systèmes, même sans avoir à lancer un script d'auto-configuration avant la compilation.

Voici un exemple :

```
#ifdef sun
    // solaris
static char *port="/dev/ttya";
#endif

#ifdef linux
    // linux
static char *port="/dev/ttyS0";
#endif
```

Quel #define est disponible ?

La question est simplement : comment savez-vous qu'il y a un #define linux ou __linux__ ? ... et quels autres #defines avons-nous ?

C'est assez simple. Lancez juste cette commande et vous obtiendrez une liste des déclarations #define prédéfinies :

```
touch /tmp/foo.h ; cpp -dM /tmp/foo.h
```

C'est tout.

Site Web maintenu par l'équipe d'édition LinuxFocus
© Guido Socher
"some rights reserved" see linuxfocus.org/license/
<http://www.LinuxFocus.org>

Translation information:

en --> -- : Guido Socher ([homepage](#))

en --> fr: Jean-Etienne Poirrier ([homepage](#))

2005-08-26, generated by lfparsr_pdf version 2.51