

Nome: Jorge Pereira <jpereiran@gmail.com>
Blog: <http://blog.jorgepereira.com.br/>
Arquivo: howto-brincando-com-mamona.pdf
Versão: 0.1 - 19/10/2008

“Conhecendo, Instalando e brincando com a plataforma MAMONA”



Índice

1	Apresentação.....	2
2	Autor.....	2
3	Considerações sobre o texto.....	2
4	Considerações Prévias.....	2
5	Instalação.....	3
5.1	Instalando o Mamona.....	3
5.2	Instalando a plataforma.....	3
5.3	Escolhendo qual dispositivo a ser utilizado para o hacking.....	4
5.4	Instalando o Mamona SDK.....	4
6	Executando o Mamona.....	4
6.1	Dando partida!.....	5
6.2	Instalando o VIM no seu Mamona (Opcional).....	5
6.3	Instalando pacotes nativos para facilitar o desenvolvimento.....	5
6.4	Verificando os pacotes disponíveis.....	5
6.5	Mamona e seu ambiente gráfico.....	5
7	Iniciando o ambiente gráfico.....	6
8	Referências.....	8

1 Apresentação

Creio que algumas pessoas já tenham ouvido falar sobre a plataforma Linux para dispositivos moveis desenvolvido pelo INDT (Instituto Nokia de Tecnologia), que tem por sua principal finalidade ser uma alternativa OpenSource, assim como possuir todo seu core composto de ferramentas de código aberto, podendo ser baixado por completo a partir de seu [site oficial](#).

Aos meus olhos, vejo isto como um grande diferencial em relação a grande maioria dos telefones e *tablets* existentes no mercado, que quase que sempre quando se precisa desenvolver algo e necessário recorrer ao uso do JAVA (J2ME) ou Symbian/C++ (Symbian OS), sem falar da vasta incompatibilidade entre as JSR (*Java Specification Requests*), limitações, dificuldades na portabilidade da aplicação e obviamente sem esquecer dos sistemas operacionais “obscuros” (*acho que peguei pesado!*) que alguns dispositivos executam.

2 Autor

Eu me chamo Jorge Pereira, e qualquer pessoa que me conhece sabe que tenho muito interesse em assuntos envolvendo tecnologias do gênero, pois ao decorrer da minha vida profissional tive contato com muitas coisas que costumo classificar como “Coisas divertidas”, como maquinas de cartões de créditos (*Conhecidos como POS “Point of Sale”*), máquinas de moedas, cédulas de dinheiro (*máquinas estilo coke-machine*) e outros dispositivos, além de aplicações com uso de PIC e estando trabalhando muito ultimamente com desenvolvimento utilizando ARM + uClinux, ente outra coisas legais.

3 Considerações sobre o texto

Eu quando estive na edição 2008 do evento “*BossaConference*” dentre várias palestras e assuntos abordados, fiquei muito interessado na apresentação do [Matthew Garrett](#) sobre o “Ubuntu Mobile” (*leia um breve texto sobre a palestra aqui*), que em seguida realizei algumas pesquisas e fui lendo um pouco mais sobre o [MAEMO](#). Navegando cheguei em um link sobre o projeto [MAMONA](#). (*de cara achei estranho, pois mamona e um tipo de semente que quando criança lembro ter visto várias das arvores próximo a minha casa, se não conhece a semente veja mais aqui*)

“O Mamona e uma plataforma Linux para sistemas embarcados ARM EABI (Que utilizem processadores ARM compostos de uma interface binária de aplicações para sistemas embarcados), que tem como seu principal objetivo oferecer uma alternativa completamente OpenSource tendo 100% de seu core composto de ferramentas de código aberto. Alem de ser baseado no debian, desenvolvido e mantido pelo INDT. Tem como principal idéia que seja utilizado em toda plataforma onde você tenha liberdade de instalar e testar os componentes. Se você achar algum celular não fechado (NEO é um exemplo), você pode colocar o mamona lá sem problemas!”

Confesso que quando li pela primeira vez sobre o Mamona, fiquei muito interessado em conhecer um pouco mais sobre o ambiente, e quem sabe desenvolver algo que automaticamente iria ver uma forma de contribuir com o projeto. Então comentei com alguns amigos mais próximos sobre a plataforma, e alguns já tinham ouvido falar porém não tinha tido tanto interesse em se aprofundar devido a não ter algum aparelho compatível para rodar o sistema e tais aplicações, respondi em seguida dizendo que havia instalado todo o ambiente em casa e estava desenvolvendo pequenas aplicações e rodando em seu próprio emulador com auxilio do QEMU. Percebi que gerou bastante interesse e foi a partir disto que surgiu a idéia de escrever este texto explicando talvez um pouco mais detalhado do que o material já existente no [WIKI](#) do projeto MAMONA.

Lembrando que não tenho quaisquer ligação direta ou indireta com [INDT](#), empresa responsável pelo projeto MAMONA, assim como tal artigo visa apenas falar um pouco sobre a plataforma, instalação e uso do ambiente.

4 Considerações Prévias

Conforme o próprio site do projeto menciona, existem três formas de dar inicio a brincadeira com o mamona, uma é utilizando a plataforma no dispositivo (não irei abordar nada sobre o assunto neste artigo devido a não ter um dispositivo para tais fins), segunda e utilizar o SDK para desenvolver aplicativos direcionados ao MAMONA e a terceira e você realizar *hacks* em sua plataforma e realizar testes em um dispositivo com a plataforma ou através do emulador QEMU. Lembrando que este texto e todos os procedimentos foram feitos e uma estação *Linux Ubuntu 8.04- amd64*, assim como você deverá ter previamente instalado em seu Linux as ferramentas necessárias para compilação e sincronização com repositórios de controle de versão GIT, e especificamente o **GCC-3.x** como sendo necessário para a compilação do “mamona-installer”. (compilador gcc, headers, libc6-dev, etc)

```
[jpereira@jiraya ~]$ sudo apt-get update
[jpereira@jiraya ~]$ sudo apt-get install build-essential gcc-3.4 gcc-3.4-base git-core
```

5 Instalação

5.1 Instalando o Mamona

Conforme descrito logo abaixo, segue os passos para o download e instalação.

```
[jpereira@jiraya ~]$ wget -c http://dev.openbossa.org/mamona/0.2/installer/mamona-installer_0.2.3.tar.gz
[jpereira@jiraya ~]$ tar -xf mamona-installer_0.2.3.tar.gz
[jpereira@jiraya ~]$ cd mamona-installer
[jpereira@jiraya mamona-installer]$ ./autogen.sh
[jpereira@jiraya mamona-installer]$ ./configure --prefix=/usr
[jpereira@jiraya mamona-installer]$ make
[jpereira@jiraya mamona-installer]$ sudo make install
```

OBS: Será instalado os seguintes artefatos abaixo.

```
/usr/bin/mamona-platform-install
/usr/bin/mamona-qemu-arm
/usr/bin/mamona-chroot
/usr/bin/mamona-gen-utils
/usr/bin/mamona-sdk-install
/usr/bin/mamona-platform-config
/usr/bin/mamona-cputransp
/usr/share/doc/mamona-installer/copyright
/usr/share/doc/mamona-installer/README.Debian
/usr/share/doc/mamona-installer/changelog.gz
/usr/share/doc/mamona-installer/AUTHORS
/usr/share/mamona/distributions
/usr/share/mamona/distributions-noemu
/usr/lib/mamona-installer/mamona-stdcheck
/usr/lib/mamona-installer/debootstrap-mamona
/usr/lib/mamona-installer/mamona-mamonarc
```

Breve descrição de alguns dos artefatos.

- **mamona-platform-install** – Esse *script* é utilizado para instalar o ambiente necessário para construir e hackear a sua própria plataforma.
- **mamona-sdk-install** – Instalador do MAMONA SDK
- **mamona-cputransp** – Ativa/Desativa compatibilidades dos binários juntamente com o mamona-qemu-arm.
- **mamona-chroot** – Script auxiliar utilizado para acessar o Mamona SDK (ambiente em chroot)
- **debootstrap-mamona** – Utilizado para criar imagem a ser utilizada no dispositivo com todo o ambiente.
- **mamona-qemu-arm** – QEMU customizado com as devidas alterações necessárias para uso do MAMONA

5.2 Instalando a plataforma

Para instalarmos a plataforma, é necessário executar o script “mamona-plataform-install” conforme abaixo. Neste caso eu fiz a minha instalação no diretório /usr/local/mamona, caso não seja informado nada ao parâmetro “-d”, será instalado no seu diretório corrente.

```
[jpereira@jiraya ~]$ sudo mamona-platform-install -d /usr/local/mamona
[jpereira@jiraya ~]$ sudo chown jpereira.jpereira -R /usr/local/mamona
```

OBS: Lembrando que este procedimento poderá levar alguns minutos, minha conexão de 1Mb levou em média 30 minutos, abaixo segue lista do que será instalado no diretório de instalação.

```
[jpereira@jiraya ~]$ ls -l /usr/local/mamona
bitbake
conf
mamona.git
mamonarc
sources
tmp
tools.git
[jpereira@jiraya ~]$
```

5.3 Escolhendo qual dispositivo a ser utilizado para o hacking

Você pode encontrar lista completa dos dispositivos disponíveis em `/usr/local/mamona/mamona.git/conf/machine/` no meu caso escolhi utilizar o *nokia810* conforme abaixo.

```
[jpereira@jiraya ~]$ cd /usr/local/mamona
[jpereira@jiraya mamona]$ source mamonarc -m nokia810 -s
```

Values were written to mamonarc file!

```
#
## Exported MAMONA variables:
#
# PATH = /usr/local/mamona/bitbake/bin:/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin
# BBPATH = /usr/local/mamona/mamona.git:/usr/local/mamona
# MAMONA_HOME = /usr/local/mamona
# MAMONA_REPO_PATH = /usr/local/mamona/mamona.git
# MAMONA_BUILD_NAME = default
# MAMONA_MACHINE = nokia810
# MAMONA_BUILD_METHOD = singlecore
```

```
*INFO* Your local.conf is in /usr/local/mamona/conf
(m:nokia810) [jpereira@jiraya mamona]$
```

OBS: Repare que o seu prompt (PS1) será alterado contendo iniciais referente ao dispositivo escolhido, assim como adicionado alguns diretórios no seu PATH corrente.

5.4 Instalando o Mamona SDK

Como a principio a idéia e realizar a emulação do ARM (EABI) em sua estação i386, precisamos iniciar um script que tem por finalidade registrar o tipo de binário utilizado para que o emulador interprete automaticamente associando como uma aplicação executada com auxilio do QEMU, procedimento chamado de "binfmt_misc", com isto basta executar o script abaixo.

O mamona-cputransp vai cuidar também pra evitar conflitos com o Scratchbox (ambiente de desenvolvimento do Maemo). Você não vai poder executar o scratchbox junto com o Mamona, mas após usar o mamona, um simples mamona-cputransp stop vai deixar o seu sistema como antes! ;)

```
[jpereira@jiraya ~]$ sudo /usr/local/bin/mamona-cputransp start
Mamona cpu transparency started.
[jpereira@jiraya ~]$
```

A Principio fiz a instalação com os valores padrões que consistem na instalação do dispositivo "nokia 810", tendo o diretóri o `"/usr/local/mamona-root"` como base para instalação do SDK, arquitetura ARM (única suportada ate o momento) e tendo o repositório origem padrão como sendo o da versão 0.2 (<http://dev.openbossa.org/mamona/0.2>)

```
[jpereira@jiraya ~]$ sudo /usr/local/bin/mamona-sdk-install -d
```

OBS: Relaxe, este procedimento pode demorar alguns minutos. O tamanho médio do download fica por volta de 220Mb, a partir disto faça os cálculos do tempo médio em sua conexão!

6 Executando o Mamona

Até este ponto já temos instalado todos os arquivos necessários para utilizar nosso SDK juntamente com o Mamona, abaixo segue os passos para acessar o ambiente. Lembrando que para a execução do mamona e necessário estar com o script "mamona-cpustransp" inicializado, no meu caso achei interessante para que o script seja executado sempre na inicialização do meu Linux, bastando acrescentar a linha abaixo no arquivo `/etc/rc.local` utilizando seu editor de texto predileto.

```
if [ -x "/usr/local/bin/mamona-cputransp" ];then
  /usr/local/bin/mamona-cputransp start
fi
```

6.1 Dando partida!

Abaixo passos para montar o dispositivo “*pseudos terminais*”, necessário para execução do terminal “*xterm*” no seu Mamona.

```
[jpereira@jiraya jorge]$ sudo mount -t devpts none /usr/local/mamona-root/dev/pts
[jpereira@jiraya jorge]$ sudo mamona-chroot /usr/local/mamona-root -d
Mounting /proc...
Chroot /usr/local/mamona-root ...
sh-3.2# uname --machine
armv5tel
sh-3.2# apt-get -f install
sh-3.2#
```

OBS: Repare que quando executei o “*uname -m*” ele me retornou como sendo uma máquina com processador ARM, quando na verdade está sendo emulado com o auxílio do QEMU. Preferi executar o “*apt-get -f install*” para corrigir e resolver quaisquer tipo de inconsistência caso exista em relação aos pacotes *debs*.

6.2 Instalando o VIM no seu Mamona (Opcional)

```
sh-3.2# apt-get install vim vim-syntax
sh-3.2# mv /bin/vi /bin/vi.old
sh-3.2# ln -fs /usr/bin/vim /bin/vi
```

Caso queira, instale uma versão incrementada do meu VIMRC conforme abaixo.

```
sh-3.2# cd /home/root
sh-3.2# wget -O .vimrc http://blog.jorgepereira.com.br/wp-content/dist/scripts/dot.vimrc
sh-3.2# mkdir -p .vim/backup
```

6.3 Instalando pacotes nativos para facilitar o desenvolvimento

Primeiramente devemos adicionar mais um repositório em nosso mamona para efetuamos a instalação de nosso compilador conforme abaixo.

```
sh-3.2# echo 'deb http://dev.openbossa.org/mamona/0.2/noemu mamona main' >> /etc/apt/sources.list
sh-3.2# apt-get update
```

Agora basta instalarmos nosso SDK contendo compiladores para serem executados a partir de um host i386 sem a necessidade de ser emulado através do QEMU.

```
sh-3.2# apt-get install task-mamona-sdk-noemu
```

6.4 Verificando os pacotes disponíveis

Caso queira verificar quais pacotes existem disponíveis no repositório, basta seguir o comando abaixo.

```
sh-3.2# apt-cache dump | grep "^Package:" | sort -n
```

OBS: Dúvidas e detalhes sobre gerenciamento de pacotes “*deb*”? basta consultar o manual “*man apt-get*”

6.5 Mamona e seu ambiente gráfico

O mamona utiliza como seu gerenciador de janelas padrão o famoso “*Enlightenment*”, creio que seja pela sua simplicidade, e este será muito importante para caso seu interesse seja o desenvolvimento de aplicações gráficas para o seu Mamona. Abaixo segue os procedimentos necessários para à instalação.

```
sh-3.2# apt-get -fy --force-yes install task-mamona-wm
```

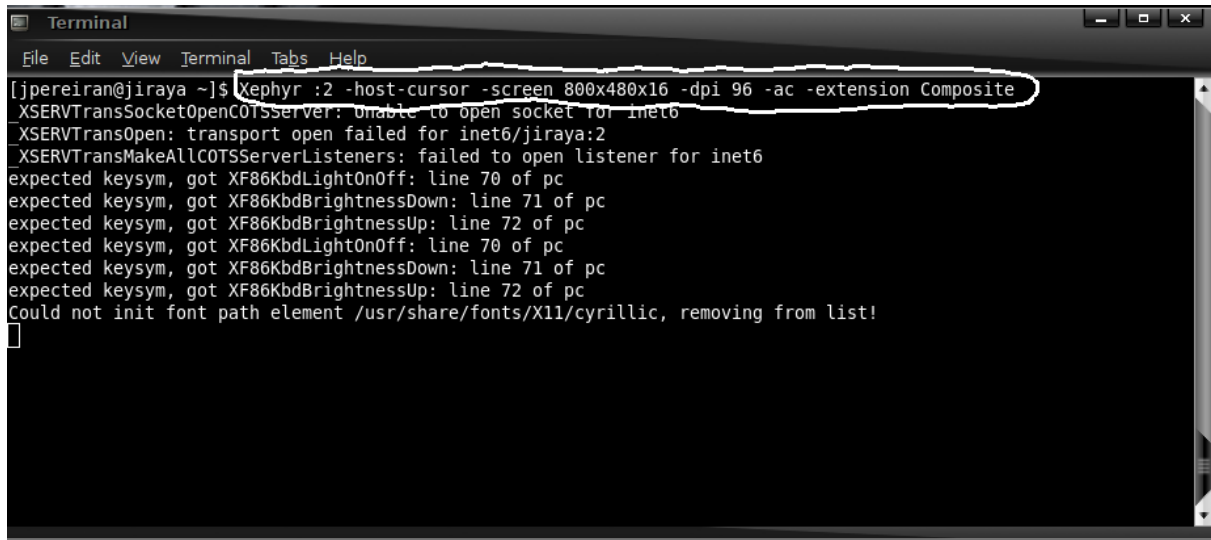
7 Iniciando o ambiente gráfico

A Idéia inicial e simples, a partir do seu ambiente de desenvolvimento do mamona irá se conectar no seu host em uma instancia do seu servidor X11 disponibilizada através do Xephyr, caso não tenha ele basta instalar através do seu apt-get no seu Ubuntu.

```
[jpereira@jiraya ~]$ apt-get install xserver-xephyr
```

O Procedimento agora e simples, no seu Linux a partir de um terminal crie uma instancia do seu X11 com o *xephyr* conforme abaixo.

```
[jpereira@jiraya ~]$ Xephyr :2 -host-cursor -screen 800x480x16 -dpi 96 -ac -extension Composite
```

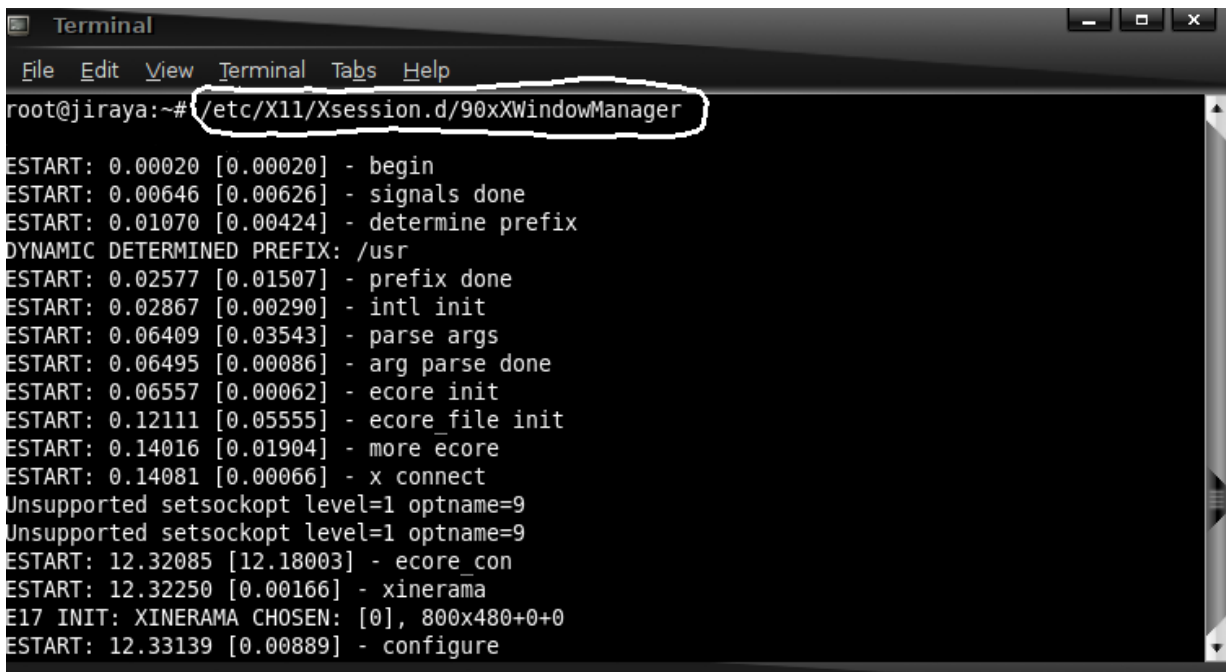


```
Terminal
File Edit View Terminal Tabs Help
[jpereira@jiraya ~]$ Xephyr :2 -host-cursor -screen 800x480x16 -dpi 96 -ac -extension Composite
XSERVTransSocketOpenCOTSServer: Unable to open socket for inet6
XSERVTransOpen: transport open failed for inet6/jiraya:2
XSERVTransMakeAllCOTSServerListeners: failed to open listener for inet6
expected keysym, got XF86KbdLightOnOff: line 70 of pc
expected keysym, got XF86KbdBrightnessDown: line 71 of pc
expected keysym, got XF86KbdBrightnessUp: line 72 of pc
expected keysym, got XF86KbdLightOnOff: line 70 of pc
expected keysym, got XF86KbdBrightnessDown: line 71 of pc
expected keysym, got XF86KbdBrightnessUp: line 72 of pc
Could not init font path element /usr/share/fonts/X11/cyrillic, removing from list!
```

No seu ambiente do mamona, execute o procedimento abaixo.

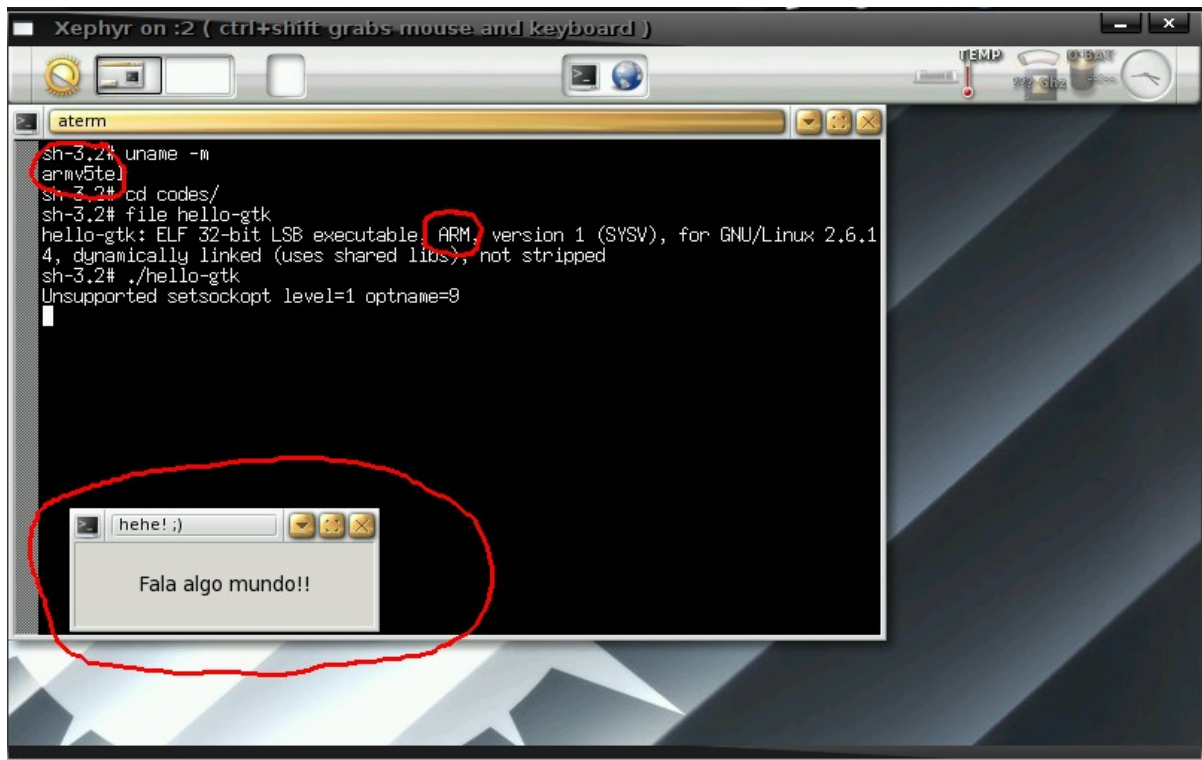
```
sh-3.2# export DISPLAY=:2
```

```
sh-3.2# /etc/X11/Xsession.d/90xXWindowManager
```



```
Terminal
File Edit View Terminal Tabs Help
root@jiraya:~# /etc/X11/Xsession.d/90xXWindowManager
ESTART: 0.00020 [0.00020] - begin
ESTART: 0.00646 [0.00626] - signals done
ESTART: 0.01070 [0.00424] - determine prefix
DYNAMIC DETERMINED PREFIX: /usr
ESTART: 0.02577 [0.01507] - prefix done
ESTART: 0.02867 [0.00290] - intl init
ESTART: 0.06409 [0.03543] - parse args
ESTART: 0.06495 [0.00086] - arg parse done
ESTART: 0.06557 [0.00062] - ecore init
ESTART: 0.12111 [0.05555] - ecore_file init
ESTART: 0.14016 [0.01904] - more ecore
ESTART: 0.14081 [0.00066] - x connect
Unsupported setsockopt level=1 optname=9
Unsupported setsockopt level=1 optname=9
ESTART: 12.32085 [12.18003] - ecore_con
ESTART: 12.32250 [0.00166] - xinerama
E17 INIT: XINERAMA CHOSEN: [0], 800x480+0+0
ESTART: 12.33139 [0.00889] - configure
```

OBS: Abaixo segue imagem contendo exemplo do comportamento esperado, execução do mamona juntamente com uma simples aplicação que fiz utilizando GTK e em seguida o código fonte.



Execução do ambiente, e simples aplicação em GTK!

```
Terminal
File Edit View Terminal Tabs Help
Terminal Terminal Terminal
1 /**
2  * Autor: Jorge Pereira <jpereiran@gmail.com>
3  * Arquivo: hello-gtk.c
4  * Last Change: Sun Oct 19 20:15:28 2008
5  * Desc: Only hello world! ;)
6  * Compilacao: gcc $(pkg-config --cflags --libs gtk+-2.0) -o hello-gtk hello-gtk.c
7  */
8 #include <gtk/gtk.h>
9 #include <glib.h>
10
11 static void
12 do_destroy (GtkWidget * widget, gpointer data)
13 {
14     g_print ("Saindo...\n");
15     gtk_main_quit ();
16 }
17
18 int
19 main (int argc, char *argv[])
20 {
21     GtkWidget *window;
22     GtkWidget *label;
23
24     gtk_init (&argc, &argv);
25     window = gtk_window_new (GTK_WINDOW_TOPLEVEL);
26     gtk_container_set_border_width (GTK_CONTAINER (window), 20);
27     gtk_window_set_title (GTK_WINDOW (window), "hehe! ;)");
28     gtk_window_set_default_size (GTK_WINDOW (window), 200, 50);
29     g_signal_connect (G_OBJECT (window), "destroy", G_CALLBACK (do_destroy), NULL);
30
31     label = gtk_label_new ("Fala algo mundo!!");
32     gtk_container_add (GTK_CONTAINER (window), label);
33     gtk_widget_show_all (window);
34
35     gtk_main ();
36
37     return 0;
38 }
~/hello-gtk.c [FORMATO=unix] [TIPO=C] [ASCII=042] [HEX=2A] [linha=0006,0002][15%] [LINHAS=38]
"hello-gtk.c" 38L, 998C written
```

Código fonte da aplicação "hello-gtk.c"

8 Referências

- Site oficial do projeto Mamona em <http://dev.openbossa.org/mamona>
- Dúvidas tirada com "Ricardo Salveti de Araujo" <ricardo.salveti@openbossa.org>

Qualquer comentário será bem vindo!
Jorge Pereira <jpereiran@gmail.com>
<http://blog.jorgepereira.com.br>