



# Lenguaje Clip

## Introducción y Conexión con RDMS

**Migrando Aplicaciones para GNU/Linux**

**Clip-Castellano** - <http://www.clip-castellano.com.ar>

**Clip-BR** - <http://www.clip-br.web.br.com>

Rosario, Santa Fe, Argentina

21/11/2005





# ¿ Qué es Clip?

---

- Un compilador xBase con licencia GPL.
- Totalmente compatible con el código generado para CA-Clipper 5.3
- Modelo OO muy rápido y eficiente.
- Soporte internacional.
- Soporte para acceder a bases de datos relacionales.
- Expresiones regulares.
- TCP/IP
- Clases para GTK+ y GTK2
- Administración de puertos serie
- Generación de interfaz gráfica mediante documentos XML



# Histórico

---



- Computer Associates, propietario de Clipper, deja de desarrollarlo en 1995.
- Miles de programas hechos en Clipper dejan de tener soporte.
- Es necesario generar un compilador GPL compatible con Clipper.
- La empresa rusa ITK comienza el desarrollo de un compilador para xBase basado en Clipper 5.3 en el año 1998.
- Primer programa compilado en octubre de 1999.
- Migración de sistemas hechos en Clipper en Rusia y países vecinos.
- A partir de ese punto Clip se desarrolla rápidamente hasta llegar a la versión actual: 1.1.14, la cual está estable y en producción.





# Cómo obtener Clip

---

## *Clip-Castellano*

<http://clip-castellano.sourceforge.net/descarga.php>

## *Clip-BR*

<http://www.clip-br.web.br.com>

## *Página Oficial do Clip – (Rússia)*

<http://www.itk.ru/english/download.shtml>

## *linux.ru.net*

<ftp://ftp.linux.ru.net/mirrors/clip>





# Instalación

## *Fuentes*

```
export CLIPROOT=/usr/local/cliproot
tar -zxvf clip-prg-1.1.14-1.tgz
cd clip-prg-1.1.14-1
make local
```

```
vim /etc/profile
export PATH=$PATH:/usr/local/cliproot/bin
```

## *RPM*

```
rpm -ivh clip-lib-1.1.14-1.i386.rpm
clip-prg-1.1.14-1.i386.rpm
```

## *Debian*

```
dpkg -i clip-prg-1.1.14-1.i386.rpm
dpkg -i clip-prg_1.1.14-1_i386.deb
```

## Dependências

```
gcc
flex
bison
libc6-dev
make
bin-utils
```





Algumas variáveis de ambiente devem ser inseridas :

**LANG=es\_ES**

*Juego de caracteres*

**CLIP\_CLIENTCS=cp437**

*Caso o sistema tenha sido desenvolvido originalmente para **DOS**.*

**CLIP\_CLIENTCS=cp866**

*Caso o sistema tenha sido desenvolvido em GNU/Linux.*

**CLIPROOT=/usr/local/cliproot**

*Caminho principal da instalação do **CLIP**.*

**PATH=\$PATH:/usr/local/cliproot/bin**

*Caminho do diretório **BIN**. Pode ser diferente de acordo com a instalação.*



# Alô Mundo

---



```
vim alomundo.prg

// ALOMUNDO.PRG
// Programa exemplo para compilação no CLIP.

clear

devpos(00,00) ; devout("Alô Mundo");

return(nil)
```



# Compilando



```
Clip -e -m alomundo.prg
```

```
./alomundo
```





# Diferencias

---



## Impressão

No GNU/Linux, o sistema de impressão é diferente. Há duas maneiras : Diretamente através da porta (*cat texto.txt > /dev/lp0*) ou usando um sistema de impressão (*lpr teste.txt*). Esta última alternativa é mais aconselhada, pois fornece ao desenvolvedor uma série de ferramentas para administração da fila de impressão.

## Bibliotecas

É comum que sistemas sejam desenvolvidos usando bibliotecas de terceiro. No **CLIP**, ao migrar um sistema, é necessário que todo o código fonte utilizado esteja disponível, inclusive o das bibliotecas. Bibliotecas de terceiros, quando fechadas, não trabalharão corretamente.

## Programas Gráficos

O sistema gráfico do **GNU/Linux** é diferente do **DOS**, de modo que sistemas com recursos gráficos não vão funcionar ao serem migrados. É necessário que recursos próprios do **CLIP** e do **GNU/Linux** sejam utilizados.



# Clip con MySQL



- Primero debemos instalar el soporte para MySQL, tanto en Linux como en Clip:

Para Debian:

```
# apt-get install libmysqlclient12-dev  
# dpkg -i clip-mysql-1.1.14.deb
```

Para Mandriva:

```
# urpmi mysql-devel  
# rpm -i clip-mysql-1.1.14.rpm
```

Si se compila Clip desde los fuentes se debe tener instalado el soporte para MySQL antes de la compilación





# Clip con MySQL

- Ejemplo de como conectarse a MySQL y actualizar datos.
- Suponemos que tenemos funcionando el servidor MySQL
- Nos conectamos al servidor con Clip:

```
* -----  
function conectargc( cUsuario, cPass ) // En gcfunc.prg  
* -----  
local oConn  
  
local host := "127.0.0.1" // IP o nombre del host  
local port := NIL // uso el puerto por omisión  
local db := "gc" // me conecto a la base de datos  
local socket := NIL // uso el socket por omisión  
local flag := NIL // no se usa en MySQL  
  
* Conexión a la base de datos *  
oConn := ConnectNew( "MS" , host , port , cUsuario , cPass , db ,  
socket , flag )  
return oConn
```



# Clip con MySQL



Algunos datos de prueba:

Invocamos el monitor:

```
mysql -h 127.0.0.1 -u root -p
```

Creamos la base de datos:

```
create gc;
```

```
use gc;
```

```
CREATE TABLE `provincia` (  
  `id_provincia` int(3) unsigned NOT NULL default '0',  
  `pcia_nombre` varchar(20) NOT NULL default '',  
  `pcia_abreviatura` char(3) NOT NULL default '',  
  `pcia_codigoib` int(3) unsigned NOT NULL default '0',  
  `pcia_letra` char(1) NOT NULL default '',  
  KEY `prov_nombre` (`pcia_nombre`,`pcia_codigoib`)  
) TYPE=InnoDB COMMENT='Tabla de provincias unidas del río de la plata';
```

```
INSERT INTO `provincia` VALUES (901,'CAPITAL FEDERAL','CAP',901,'C');
```

```
INSERT INTO `provincia` VALUES (902,'BUENOS AIRES','BAI',902,'B');
```

```
INSERT INTO `provincia` VALUES (903,'CATAMARCA','CAT',903,'K');
```



# Clip con MySQL



Ahora hacemos un simple Browse:

```
* -----  
* Programa para conectarse y mostrar la tabla provincia  
* Nombre del programa: browsepvca.prg  
* -----  
function main()  
oConn := conectargc( "gc" , "clip" ) // Usuario y clave  
oRs := oConn:CreateRowset( "select * from localidad" )  
oRs:browse()  
  
return NIL
```

Compilamos ahora nuestra pequeña aplicación:

```
clip -e browsepvca.prg gcfunc.prg -lclip-mysql
```

Probamos la aplicación:

```
./browsepvca
```



# Clip con MySQL



El resultado de nuestros esfuerzos:

id_provincia	pcia_nombre	pcia_abreviatura	pcia_codigoib	pcia_letra
918	SAN JUAN	SJU	918	J
919	SAN LUIS	SLU	919	D
920	SANTA CRUZ	SCR	920	Z
921	SANTA FE	SFE	921	S
922	SANTIAGO DE	STG	922	G
923	TIERRA DEL	TIE	923	V
924	TUCUMAN	TUC	924	T



# Conclusión

---



- . Sistemas antes depreciados, agora voltam a ter novas características inseridas.
- . Redução do tempo despendido em suporte oriundo de problemas de corrompimento de dados.
- . Estabilização de aplicações.
- . Tempo disponível para novos projetos.
- . Maior confiança em aplicações antigas.
- . Uma das soluções para sistemas legados.
- . Manutenção do investimento.
- . Alegria dos Clippeiros :-) !!!





# *Muchas Gracias !!!*



Gustavo Courault

gcourault@gmail.com

<http://www.clip-castellano.com.ar>



Eduardo Stefani

ecs@eduardostefani.eti.br

<http://www.eduardostefani.eti.br>

