

GNU build system (Autotools): Autoconf, Automake y Libtool

Borja Bergua Guerra

12 de Marzo de 2009

Copyright © 2009 Borja Bergua Guerra
<http://creativecommons.org/licenses/by-sa/2.0/>
This is a derived work of Alexandre Duret-Lutz
<http://www.lrde.epita.fr/~adl/autotools.html>

1 Introducción

- configure
- Autoconf
- Automake
- Libtool
- Esquema general

2 Ejemplo 1: Hola Mundo

- Código fuente
- Entrada a Autotools
- Ficheros generados
- Hola Mundo explicado

3 Ejemplo 2: Más Hola Mundo

- Código fuente
- Entrada a Autotools

4 Ejemplo 3: Hola Mundo usando Libtool

- Código fuente
- Entrada a Autotools

Formas de no portabilidad en C

Consideremos funciones en C...

- que no existen en todos los sitios (p.ej., `strtod()`)
- que tienen diferentes nombres (p.ej., `strchr()` vs. `index()`)
- que tienen diferentes prototipos
(p.ej., `int setpgrp(void);` vs. `int setpgrp(int, int);`)
- que se comportan de forma diferente
- que pueden estar definidas en diferentes ficheros de cabeceras
(*string.h* vs. *strings.h* vs. *memory.h*)

¿Cómo debería un paquete lidiar con todo esto?

Se hace necesaria una configuración automática

- Mantener a mano una colección de `#define` para cada sistema es una pesadilla.
- Pedir a los usuarios que añadan las opciones de compilación `-D`, `-I` y `-l` necesarias a los *Makefile* es inadmisibile.
- Las compilaciones complicadas dificultan la aceptación del software libre.

Se hace necesaria una configuración automática

- Mantener a mano una colección de `#define` para cada sistema es una pesadilla.
- Pedir a los usuarios que añadan las opciones de compilación `-D`, `-I` y `-l` necesarias a los *Makefile* es inadmisibile.
- Las compilaciones complicadas dificultan la aceptación del software libre.
- En 1991 un grupo de gente comenzó a escribir shell scripts que **adivinaran** estas configuraciones para algunos paquetes GNU.
- Desde entonces el script *configure* es obligatorio en cualquier paquete del proyecto GNU project.

Propósito de *configure*

- *configure* comprueba que el sistema tiene las funciones, bibliotecas y herramientas necesarias.

Propósito de *configure*

- *configure* comprueba que el sistema tiene las funciones, bibliotecas y herramientas necesarias.
- Genera un fichero *config.h* con todos los `#define`

Propósito de *configure*

- *configure* comprueba que el sistema tiene las funciones, bibliotecas y herramientas necesarias.
- Genera un fichero *config.h* con todos los `#define`
- Así como los *Makefile* para construir el programa.

Para que sirve Autoconf?

Para que sirve Autoconf?

- Es un procesador de macros.

Para que sirve Autoconf?

- Es un procesador de macros.
- A partir de un fichero *configure.ac* genera el potente shell script *configure*.

Para que sirve Autoconf?

- Es un procesador de macros.
- A partir de un fichero *configure.ac* genera el potente shell script *configure*.
- *configure* realiza comprobaciones en el sistema.

Para que sirve Autoconf?

- Es un procesador de macros.
- A partir de un fichero *configure.ac* genera el potente shell script *configure*.
- *configure* realiza comprobaciones en el sistema.
- *configure* recibe opciones que modifican la forma de construir el programa.

Para que sirve Autoconf?

- Es un procesador de macros.
- A partir de un fichero *configure.ac* genera el potente shell script *configure*.
- *configure* realiza comprobaciones en el sistema.
- *configure* recibe opciones que modifican la forma de construir el programa.
- Finalmente *configure* genera el *Makefile* que construirá el programa.

Para que sirve Automake?

Para que sirve Automake?

- Nos ayuda a escribir *Makefile* portables y que sigan los estándares de GNU (GNU Coding Standards).

Para que sirve Automake?

- Nos ayuda a escribir *Makefile* portables y que sigan los estándares de GNU (GNU Coding Standards).
- A partir de un *Makefile.am* muy sencillo genera un completo *Makefile.in*

Para que sirve Automake?

- Nos ayuda a escribir *Makefile* portables y que sigan los estándares de GNU (GNU Coding Standards).
- A partir de un *Makefile.am* muy sencillo genera un completo *Makefile.in*
- Un *Makefile.in* describe cómo construir el programa.

Bibliotecas compartidas: El infierno de la portabilidad

- Casi cada sistema tiene su propio formato de bibliotecas compartidas
 - *libhello.so*
 - *libhello.dll*
 - *libhello.sl*
 - *libhello.dylib*
 - ...

Bibliotecas compartidas: El infierno de la portabilidad

- Casi cada sistema tiene su propio formato de bibliotecas compartidas
 - *libhello.so*
 - *libhello.dll*
 - *libhello.sl*
 - *libhello.dylib*
 - ...
- Su construcción requiere diferentes flags
 - -fPIC, -shared
 - -KPIC, -G
 - -bM:SRE

Bibliotecas compartidas: El infierno de la portabilidad

- Casi cada sistema tiene su propio formato de bibliotecas compartidas
 - *libhello.so*
 - *libhello.dll*
 - *libhello.sl*
 - *libhello.dylib*
 - ...
- Su construcción requiere diferentes flags
 - -fPIC, -shared
 - -KPIC, -G
 - -bM:SRE
- Enlazar con la biblioteca también puede requerir flags específicos.

Bibliotecas compartidas: El infierno de la portabilidad

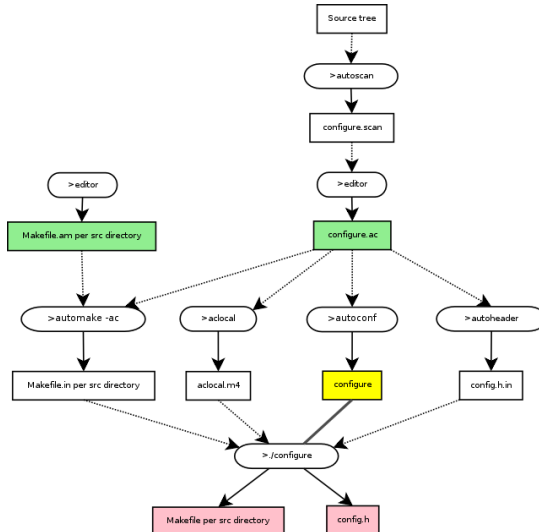
- Casi cada sistema tiene su propio formato de bibliotecas compartidas
 - *libhello.so*
 - *libhello.dll*
 - *libhello.sl*
 - *libhello.dylib*
 - ...
- Su construcción requiere diferentes flags
 - -fPIC, -shared
 - -KPIC, -G
 - -bM:SRE
- Enlazar con la biblioteca también puede requerir flags específicos.
- No es viable que un desarrollador siga el rastro de todos estos detalles.

Para que sirve Libtool?

- Un nuevo formato de biblioteca que abstrae a todos los demás.
 - libhello.la (libtool archive)

Para que sirve Libtool?

- Un nuevo formato de biblioteca que abstrae a todos los demás.
 - [libhello.la](#) (libtool archive)
- Un *wrapper script* para el compilador y el enlazador.
 - traduce las operaciones que impliquen a [libhello.la](#) a la operación correcta para el sistema actual usando la biblioteca real.



src/main.c

```
#include <config.h>
#include <stdio.h>

int
main (void)
{
    puts ("Hello _World!");
    puts ("This _is _" PACKAGE_STRING " ." );
    return 0;
}
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([--Wall --Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([--Wall --Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

Makefile.am

```
SUBDIRS = src
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([--Wall --Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

Makefile.am

```
SUBDIRS = src
```

src/Makefile.am

```
bin_PROGRAMS = hello
hello_SOURCES = main.c
```

Generación

```
autoreconf --install  
./configure  
make  
make install
```

Autoconf

Automake

configure

Autoconf

- *configure.ac* —> *configure*

Automake

configure

Autoconf

- *configure.ac* —> *configure*

Automake

- *Makefile.am* —> *Makefile.in*

configure

Autoconf

- *configure.ac* —> *configure*

Automake

- *Makefile.am* —> *Makefile.in*

configure

- *config.cache*: datos en cache de tests pasados para acelerar los siguientes tests.

Autoconf

- *configure.ac* —> *configure*

Automake

- *Makefile.am* —> *Makefile.in*

configure

- *config.cache*: datos en cache de tests pasados para acelerar los siguientes tests.
- *config.log*: salida producida por configure.

Autoconf

- *configure.ac* —> *configure*

Automake

- *Makefile.am* —> *Makefile.in*

configure

- *config.cache*: datos en cache de tests pasados para acelerar los siguientes tests.
- *config.log*: salida producida por configure.
- *config.status*: shell script que se puede usar para recrear la configuración actual.

Autoconf

- *configure.ac* —> *configure*

Automake

- *Makefile.am* —> *Makefile.in*

configure

- *config.cache*: datos en cache de tests pasados para acelerar los siguientes tests.
- *config.log*: salida producida por configure.
- *config.status*: shell script que se puede usar para recrear la configuración actual.
- *config.h*: fichero de cabeceras con directivas para el preprocesador de C/C++.

Autoconf

- *configure.ac* —> *configure*

Automake

- *Makefile.am* —> *Makefile.in*

configure

- *config.cache*: datos en cache de tests pasados para acelerar los siguientes tests.
- *config.log*: salida producida por configure.
- *config.status*: shell script que se puede usar para recrear la configuración actual.
- *config.h*: fichero de cabeceras con directivas para el preprocesador de C/C++.
- *Makefile*: reglas de construcción del programa.

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])  
AM_INIT_AUTOMAKE([-Wall -Werror foreign])  
AC_PROG_CC  
AC_CONFIG_HEADERS([config.h])  
AC_CONFIG_FILES([Makefile src/Makefile])  
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.
- Inicializa Automake. Activa todos los warnings de Automake y y los devuelve como errores. Es un programa **foreign** (externo).

configure.ac

```
...  
AM_INIT_AUTOMAKE([-Wall -Werror foreign])  
...
```

```
~$ autoreconf --install  
configure.ac:2: installing './install-sh'  
configure.ac:2: installing './missing'  
src/Makefile.am: installing './depcomp'
```

`configure.ac` sin la opción `foreign`

```
...  
AM_INIT_AUTOMAKE([-Wall -Werror])  
...
```

```
~$ autoreconf --install  
configure.ac:2: installing './install-sh'  
configure.ac:2: installing './missing'  
src/Makefile.am: installing './depcomp'  
Makefile.am: installing './INSTALL'  
Makefile.am: required file './NEWS' not found  
Makefile.am: required file './README' not found  
Makefile.am: required file './AUTHORS' not found  
Makefile.am: required file './ChangeLog' not found  
Makefile.am: installing './COPYING'  
autoreconf: automake failed with exit status: 1
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])  
AM_INIT_AUTOMAKE([-Wall -Werror foreign])  
AC_PROG_CC  
AC_CONFIG_HEADERS([config.h])  
AC_CONFIG_FILES([Makefile src/Makefile])  
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.
- Inicializa Automake. Activa todos los warnings de Automake y y los devuelve como errores. Es un programa **foreign** (externo).

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.
- Inicializa Automake. Activa todos los warnings de Automake y y los devuelve como errores. Es un programa foreign .
- Comprueba que haya un compilador de C.

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.
- Inicializa Automake. Activa todos los warnings de Automake y y los devuelve como errores. Es un programa `foreign`.
- Comprueba que haya un compilador de C.
- Declara *config.h* como el fichero de cabeceras de salida.

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.
- Inicializa Automake. Activa todos los warnings de Automake y y los devuelve como errores. Es un programa `foreign` .
- Comprueba que haya un compilador de C.
- Declara *config.h* como el fichero de cabeceras de salida.
- Declara *Makefile* y *src/Makefile* como ficheros de salida.

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

- Inicializa Autoconf. Especifica el nombre del paquete, el número de versión y la dirección a la que enviar fallos.
- Inicializa Automake. Activa todos los warnings de Automake y y los devuelve como errores. Es un programa `foreign`.
- Comprueba que haya un compilador de C.
- Declara *config.h* como el fichero de cabeceras de salida.
- Declara *Makefile* y *src/Makefile* como ficheros de salida.
- Genera todos los ficheros de salida declarados.

Makefile.am

```
SUBDIRS = src
```

- Entra recursivamente en *src/*.

src/Makefile.am

Makefile.am

```
SUBDIRS = src
```

- Entra recursivamente en *src/*.
- No se declara nada más para el directorio actual.
(El *Makefile.am* de primer nivel es normalmente pequeño.)

src/Makefile.am

Makefile.am

```
SUBDIRS = src
```

- Entra recursivamente en *src/*.
- No se declara nada más para el directorio actual.
(El *Makefile.am* de primer nivel es normalmente pequeño.)

src/Makefile.am

```
bin_PROGRAMS = hello  
hello_SOURCES = main.c
```

- Estamos construyendo programas.

Makefile.am

```
SUBDIRS = src
```

- Entra recursivamente en *src/*.
- No se declara nada más para el directorio actual.
(El *Makefile.am* de primer nivel es normalmente pequeño.)

src/Makefile.am

```
bin_PROGRAMS = hello  
hello_SOURCES = main.c
```

- Estamos construyendo programas.
- Estos programas se instalarán en *bindir*.

Makefile.am

```
SUBDIRS = src
```

- Entra recursivamente en *src/*.
- No se declara nada más para el directorio actual.
(El *Makefile.am* de primer nivel es normalmente pequeño.)

src/Makefile.am

```
bin_PROGRAMS = hello  
hello_SOURCES = main.c
```

- Estamos construyendo programas.
- Estos programas se instalarán en *bindir*.
- Sólo hay un programa que construir: *hello*.

Makefile.am

```
SUBDIRS = src
```

- Entra recursivamente en *src/*.
- No se declara nada más para el directorio actual.
(El *Makefile.am* de primer nivel es normalmente pequeño.)

src/Makefile.am

```
bin_PROGRAMS = hello  
hello_SOURCES = main.c
```

- Estamos construyendo programas.
- Estos programas se instalarán en *bindir*.
- Sólo hay un programa que construir: *hello*.
- Para crear *hello* simplemente compila *main.c*.

src/main.c

```
#include <config.h>
#include <stdio.h>

int
main (void)
{
    puts (" Hello _World! " );
    puts (" This _is _" PACKAGE_STRING " ." );
#ifdef ENABLE_WIFI
    puts (" Ahora _funciona _por _wifi" );
#endif
    return 0;
}
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_ARG_ENABLE(  [wifi],
               [[AS_HELP_STRING(
                           [--enable-wifi],
                           [Enable Wifi module.])]],
               [CPPFLAGS="$CPPFLAGS -DENABLE_WIFI"]
)
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_ARG_ENABLE(  [wifi],
               [[AS_HELP_STRING(
                           [--enable-wifi],
                           [Enable Wifi module.])],
               [CPPFLAGS="$CPPFLAGS -DENABLE_WIFI"
               )
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_ARG_ENABLE(  [wifi],
               [[AS_HELP_STRING(
                           [--enable-wifi],
                           [Enable Wifi module.])]],
               [CPPFLAGS="$CPPFLAGS -DENABLE_WIFI"]
)
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

configure.ac

```
AC_INIT([Mi_hola], [1.0], [bug-report@address])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_CC
AC_ARG_ENABLE(  [wifi],
               [[AS_HELP_STRING(
                           [--enable-wifi],
                           [Enable Wifi module.])]],
               [CPPFLAGS="$CPPFLAGS -DENABLE_WIFI"]
)
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile src/Makefile])
AC_OUTPUT
```

lib/say.c

```
#include <config.h>
#include <stdio.h>

void say_hello (void)
{
    puts ("Hello World!");
    puts ("This is " PACKAGE_STRING ".");
}
```

lib/say.h

```
void say_hello (void);
```

src/main.c

```
#include "say.h"

int main (void)
{
    say_hello ();
    return 0;
}
```

lib/Makefile.am

```
lib_LTLIBRARIES = libhello.la  
libhello_la_SOURCES = say.c say.h
```

src/Makefile.am

```
AM_CPPFLAGS = -I$(srcdir)/../lib  
bin_PROGRAMS = hello  
hello_SOURCES = main.c  
hello_LDADD = ../lib/libhello.la
```

Makefile.am

```
SUBDIRS = lib src
```

configure.ac

```
AC_INIT([Mi_hola], [2.0], [bug-report@address])
AC_CONFIG_AUX_DIR([build-aux])
AM_INIT_AUTOMAKE([-Wall -Werror foreign])
AC_PROG_LIBTOOL
AC_PROG_CC
AC_CONFIG_HEADERS([config.h])
AC_CONFIG_FILES([Makefile lib/Makefile src/Makefile])
AC_OUTPUT
```

Alternativas

- CMake. <http://www.cmake.org>
- SCons. <http://www.scons.org>
- Waf. <http://code.google.com/p/waf/>
- Apache Ant. <http://ant.apache.org>

Bibliografía básica

- Using GNU Autotools. Alexandre Duret-Lutz.
<http://www.lrde.epita.fr/~adl/autotools.html>
- Autotools Tutorial for Beginners. Mark K. Kim.
<http://markuskimius.wikidot.com/programming:tut:autotools>
- Autotools Tutorial. Sarah George. Developing Programmers.
<http://www.developingprogrammers.com/index.php/2006/01/05/autotools-tutorial/>
- autotut: Using GNU autoconf,make,header. Felipe Bergo.
<http://www.seul.org/docs/autotut/>
- Learning the GNU development tools. Eleftherios Gkioulekas.
<http://www.st-andrews.ac.uk/~iam/docs/tutorial.html>

Bibliografía básica

- **Using GNU Autotools. Alexandre Duret-Lutz.**
<http://www.lrde.epita.fr/~adl/autotools.html>
- Autotools Tutorial for Beginners. Mark K. Kim.
<http://markuskimius.wikidot.com/programming:tut:autotools>
- Autotools Tutorial. Sarah George. Developing Programmers.
<http://www.developingprogrammers.com/index.php/2006/01/05/autotools-tutorial/>
- autotut: Using GNU autoconf,make,header. Felipe Bergo.
<http://www.seul.org/docs/autotut/>
- Learning the GNU development tools. Eleftherios Gkioulekas.
<http://www.st-andrews.ac.uk/~iam/docs/tutorial.html>

Bibliografía específica

- GNU Autoconf, Automake and Libtool. Gary V. Vaughan, Ben Elliston, Tom Tromey and Ian Lance Taylor. New Riders. Octubre, 2000. <http://sources.redhat.com/autobook/>
- GNU Make.
<http://www.gnu.org/software/make/manual/>
- GNU Autoconf - Creating Automatic Configuration Scripts.
<http://www.gnu.org/software/autoconf/manual/>
- GNU Automake.
<http://www.gnu.org/software/automake/manual/>
- GNU Libtool - The GNU Portable Library Tool.
<http://www.gnu.org/software/libtool/manual/>

GNU build system (Autotools): Autoconf, Automake y Libtool

Borja Bergua Guerra

12 de Marzo de 2009

Copyright © 2009 Borja Bergua Guerra
<http://creativecommons.org/licenses/by-sa/2.0/>
This is a derived work of Alexandre Duret-Lutz
<http://www.lrde.epita.fr/~adl/autotools.html>