

# AdaIntl

Paquete de internacionalización y localización de aplicaciones en Ada95.

## Índice de contenido

1- ¿Que es AdaIntl ?.....	3
2- ¿Cómo funciona?.....	4
3- Descripción de los tipos y funciones de AdaIntl.....	5
3.1- Tipos.....	5
3.1.1 – Internationalization_Type.....	5
3.1.2 – Debug_Level_Type.....	6
3.1.3 – Language_Type.....	7
3.1.4 – Available_Languages_Array.....	8
3.2- Funciones.....	9
3.2.1 – Initialize_AdaIntl (función).....	9
3.2.2 – Initialize_AdaIntl (procedimiento).....	11
3.2.3 – Set_Default_Domain.....	12
3.2.4 – Set_Debug_Mode.....	13
3.2.5 – Set_Language.....	14
3.2.6 – Get_Default_Domain.....	15
3.2.7 – Get_Debug_Mode.....	16
3.2.8 – Get_Language.....	17
3.2.9 – “-” (un parámetro).....	18
3.2.10 – “-” (dos parámetros).....	19
3.2.11 – Get_Available_Languages.....	20
3.2.12 – Clean_AdaIntl.....	21
3.3 – Excepciones.....	22
3.3.1 – File_Not_Correct.....	22
3.3.2 – Conf_File_Not_Correct.....	23
3.3.3 – Language_Not_Valid.....	24
3.3.4 – Not_Initialized.....	25
4- Estructura y funcionamiento interno.....	26
4.1 – Directorios y ficheros de AdaIntl.....	26
4.1.1 - /src.....	27
4.1.2 - /example.....	28
4.1.3 - /doc.....	28
4.1.4 – Instalación.....	29
4.2 – Ficheros de traducción y configuración.....	30
4.2.1 – Dominios.....	30
4.2.2 – Ficheros de configuración.....	32
4.3 – Funcionamiento interno.....	33
5- Código de ejemplo.....	34
6- Lista de cosas por mejorar o arreglar.....	36
7- Version e historial.....	38
8- Créditos y licencia.....	39
8.1- LGPL.....	39
Preámbulo.....	40
Términos y condiciones para la copia, distribución y modificación.....	41
Cómo aplicar estos términos a sus nuevas librerías.....	46

## 1- ¿Que es AdaIntl ?

AdaIntl es una librería multiplataforma hecha totalmente en Ada95 y liberada bajo licencia LGPL que permite localizar software multilingüe de forma sencilla.

Para traducir un programa sin ninguna librería para ello, es necesario modificar todas las cadenas de texto del código fuente de forma manual, y volver a compilarlas y linkarlas. Se pueden crear estructuras complejas que devuelvan cadenas de texto en distintos idiomas, pero cualquier traducción nueva o corrección pasa necesariamente por la edición del código fuente, lo que lo dota de muy poca flexibilidad.

AdaIntl permite de una forma extremadamente sencilla crear archivos de texto donde se guarden todas las cadenas de texto que usa el programa. Para traducir la aplicación a otros idiomas, tan solo será necesario editar esos archivos de texto (manualmente, con otras aplicaciones, etc). De esa forma un mismo binario puede estar en tantos idiomas como se desee, y permite cambiar de uno a otro en tiempo de ejecución.

Funciona de forma muy similar a GNU/gettext, el cual si se quiere usar en Ada se debe importar desde C (como hace GTKAda.Intl). AdaIntl evita el tener que instalar gettext o importar y depender de código externo a Ada. AdaIntl es directo, sencillo y fácil de usar.

Es importante resaltar que AdaIntl **NO** traduce el programa, AdaIntl **NO** es un traductor.

### *Resumen de características:*

- Guarda las cadenas de texto en archivos para ser traducidas fácilmente.
- 100% hecho en Ada95.
- Multiplataforma.
- No usa variables de entorno.
- Posibilidad de cambiar de idioma en tiempo de ejecución.
- Guarda el idioma elegido en un archivo para cargar ese idioma en la próxima ejecución.
- 2 tipos de estructura de archivos: en carpetas o en el mismo directorio.
- Identificación de las cadenas de texto mediante el hash "Elf".
- 6 modos de Debug.
- Posibilidad de adaptar y convertir a archivos \*.po.
- Soporte para 175 idiomas.

## 2- ¿Cómo funciona?

AdaIntl guarda las cadenas de texto del código fuente en archivos de texto (llamados “dominios”) para poder ser traducidas y posteriormente leerlas.

Las cadenas de texto que se guardan son las que llevan “-” antes de cada string. Por ejemplo, una cadena que no guardaría sería:

```
Put_Line("Cadena de texto");
```

En cambio sí que se guardaría en el dominio la siguiente cadena:

```
Put_Line("-Cadena de texto");
```

Dado que las cadenas de texto van a ser traducidas, se calcula un hash para identificarlas en el archivo de localización. Por tanto las cadenas que tengan el mismo hash se consideran iguales.

Al ejecutar el programa, AdaIntl calcula el hash de "Cadena de texto" y mira si está en el archivo de localización. Si no es así, lo escribe en él. Si el hash está en el dominio, lee la cadena de texto del fichero (identificada por el hash, ya que la cadena puede variar al ser traducida) y la devuelve para ser visualizada por pantalla.

Para acelerar el proceso, las cadenas de texto de cada dominio están cargadas en memoria y almacenadas en un [árbol AA](#). Para más detalles sobre el funcionamiento interno, se puede consultar el punto 4.3.

Para organizar mejor la traducción, AdaIntl permite usar varios dominios. Por ejemplo, puede haber un dominio “Mensajes\_de\_error”, otro “Avisos”, otro “Acerca\_de”, etc. En cualquier momento se puede pasar de un dominio a otro, ya sea especificando el dominio por defecto, o especificando explícitamente que dominio usar con la función “-”.

Además permite elegir el idioma y cambiarlo en cualquier momento, guardar un archivo de configuración con el idioma elegido por el usuario (y así usar ese mismo idioma en la próxima ejecución), obtener todos los idiomas en los que está disponible la aplicación, etc.

## 3- Descripción de los tipos y funciones de AdaIntl

### 3.1- Tipos

#### 3.1.1 – Internationalization\_Type

Este tipo tan solo se usa para poder inicializar **AdaIntl** durante la etapa de declaración del programa. No vuelve a usarse más adelante (excepto para volver a inicializarlo si hace falta).

La función encargada de inicializar **AdaIntl** es “**Initialize\_AdaIntl**” y se explica en el apartado 3.2.1.

¿Porque puede ser necesario inicializar **AdaIntl** en la etapa de declaración?  
Por ejemplo si nuestro programa tiene estas líneas:

```
Mensaje: String:= -“Hola”;  
begin  
  Initialize_AdaIntl(...);
```

En este caso primero se ejecutaría “-” con Hola, pero al no estar **AdaIntl** inicializado, daría un error. En cambio podemos ejecutar antes “**Initialize\_AdaIntl**” y de esta forma asegurarnos una ejecución correcta:

```
IT: Internationalization_Type:=Initialize_AdaIntl(...);  
Mensaje: String:= -“Hola”;
```

Aparte de permitir la ejecución de “**Initialize\_AdaIntl**”, el tipo “*Internationalization\_Type*” no tiene otro propósito.

Si no se quiere inicializar **AdaIntl** durante la etapa de declaración, se puede usar el procedimiento “**Initialize\_AdaIntl**”, el cual es exactamente que la función del mismo nombre, y de esta forma no hay que declarar ni usar ninguna variable de tipo “*Internationalization\_Type*”.

### 3.1.2 – Debug\_Level\_Type

AdaIntl dispone de 6 modos de uso y debug:

- **Deactivated**
- **No\_Debug**
- **Only\_Errors\_Stop**
- **Only\_Errors\_No\_Stop**
- **Total\_Stop**
- **Total\_No\_Stop**

- El modo “*Deactivated*” desactiva el sistema de localización. No se hace nada, y la función “-” devuelve la misma cadena de entrada.

- El modo “*No\_Debug*” hace funcionar la librería de forma normal. Si hay algún fallo, no imprime ningún valor ni da ningún aviso, sino que “-” devuelve la misma cadena de entrada. Este modo “*No\_Debug*” es el modo por defecto y es el que debería ser usado cuando el programa esté listo en la fase de explotación.

- “*Only\_Errors*”: Este modo tan solo imprime mensajes de error cuando hay fallos (archivos de traducción incorrectos, etc). Hay dos variantes, “*Only\_Errors\_No\_Stop*” y “*Only\_Errors\_Stop*”. El primero (No\_Stop) no detiene la ejecución del programa, ante un error “-” devuelve la cadena de entrada. El modo Stop lanza una excepción que detiene la ejecución del programa. Estos modos se pueden usar durante la desarrollo y testeo del programa, aunque también se podría usar durante la fase de explotación.

- Por último, “Total” imprime todo tipo de mensajes diciendo que hace y deja de hacer AdaIntl. También dispone de dos submodos “Stop” y “No\_Stop” que continua la ejecución o la detiene con una excepción. Este modo se usa para comprobar el buen funcionamiento de AdaIntl.

### 3.1.3 – Language\_Type

**AdaIntl** tiene soporte para 175 idiomas, contenidos en el paquete **d\_idiomas**.

*Language\_Type* es un tipo creado a partir de *d\_idiomas.T\_Language* para ofrecer visibilidad cara al programador. Cualquier cambio que se desee realizar en los idiomas se han de hacer en ese paquete **d\_idiomas**.

Los idiomas son los enumerados en el [ISO 639-1](#).

Sin embargo, cuenta con 2 diferencias:

- Algunos idiomas tienen 3 letras, como Isl y Ori, porque su código ISO es una palabra reservada en Ada95 (en estos casos, **IS** y **OR**).
- Se añade un idioma “comodín”. Es el idioma **NULO** (nul – no se usa **null** por ser palabra reservada en Ada95). Este debe ser el *ULTIMO* en la lista enumerada. Si se añadiesen otros idiomas, se deben colocar antes que nul: deberá seguir siendo el último para el correcto funcionamiento de **AdaIntl**.  
Este idioma “nul” no puede usarse en ningún momento. Sirve para indicar que el estado no inicializado de **AdaIntl**. En la inicialización (obligatoria, ver punto 3.2.1) se debe escoger otro idioma.

### 3.1.4 – Available\_Languages\_Array

Esta estructura permite saber que idiomas hay disponibles.

Es un vector de *booleans*, que dado un idioma “*Language\_Type*” (vease 3.1.3) devuelve “*true*” si existe un fichero de traducción para ese idioma, y “*false*” si no existe.

Se usa con la función “*Get\_Available\_Languages*”. Para más información y códigos de ejemplos, véase el punto 3.2.11.



## 3.2- Funciones

### 3.2.1 – Initialize\_AdaIntl (función)

```
function Initialize_AdaIntl (
  Language           : Language_Type;
  Default_Domain     : String         := "Language";
  Debug_Mode         : Debug_Level_Type := No_Debug;
  Directory           : String         := "";
  Load_Configuration_File : String    := ""      )
return Internationalization_Type;
```

Lo primero que hay que hacer es inicializar **AdaIntl**. Para ello antes de nada se debe ejecutar la instrucción “**Initialize\_AdaIntl**”, con una serie de argumentos. Es obligatorio ejecutar “**Initialize\_AdaIntl**” antes que cualquier otra función o procedimiento de **AdaIntl**. “**Initialize\_AdaIntl**” es una función y no un procedimiento para poder ser ejecutada antes que cualquier cadena de texto (tal como se explica en el punto 3.1.1).

Los argumentos de **Initialize\_AdaIntl** son:

- **Idioma (Language):**  
Es necesario especificar un idioma, que será el usado a partir de ese momento. Se puede escoger cualquier excepto “*Nul*” (ver punto 3.1.3).
- **Dominio por defecto (Default Domain):**  
También se debe especificar en que archivo se guardarán las cadenas. Puede haber varios dominios, pero solo uno será el dominio “por defecto”. Si no se especifica nada, el archivo será “*Language*”.
- **Debug:**  
**AdaIntl** permite varios modos de uso y debug, tal como se especifica en el punto 3.1.2.
- **Directorio (Directory):**  
Se puede especificar que estructura de fichero usarán los dominios. Hay 2 tipos disponibles: todos los ficheros de localización en el mismo directorio que el programa, u organizados en carpetas por idiomas.

Si “*Directorio*” es “” (cadena nula), los ficheros se organizan de la primera manera (todos en el mismo directorio y diferenciados por extensión).

Por ejemplo, si tenemos el programa “Aplicacion” y el dominio “Language” en inglés y castellano, el directorio del programa tendrá los siguientes archivos:

- *Aplicación* (binario del programa)
- *Language.ES* (dominio en castellano)
- *Language.EN* (dominio en inglés).

Este modo es útil cuando la aplicación es pequeña y tiene pocos dominios.

Por otro lado, se puede organizar por directorios. Para ello “*Directorio*” ha de tomar un valor distinto a la cadena nula y será el directorio donde se guardarán los dominios. Es **muy importante** que el último carácter sea el de separación de directorios (/). Por ejemplo, “language/” con el caso anterior crearía los siguientes ficheros:

- *Aplicación* (binario del programa)
- *Language* (directorio)
  - *ES* (directorio de dominios en castellano)
    - *Language* (dominio en castellano)
  - *EN* (directorio de dominios en inglés)
    - *Language* (dominio en inglés).

Es importante notar que los ficheros en este caso NO tienen extensión (en el modo anterior tenían extensión para diferenciar el idioma, como ahora se diferencian por el directorio, pueden tener la extensión que el usuario quiera).

Este modo es útil cuando la aplicación es grande y tiene muchos dominios, ya que organiza mejor los ficheros de localización. Sin embargo presenta un **problema**: Ada95 **no** permite crear directorios, por lo que antes de ejecutar la aplicación se deberán crear manualmente. De lo contrario fallará y dependiendo del nivel de Debug detendrá la ejecución y “-” irá devolviendo las cadenas de entrada sin localizar el programa. Ada2005 sí permite manipulación de directorios.

- **Ruta del fichero de configuración (Load\_Configuration\_File):**

Por último, se puede cargar el idioma de un fichero con la ruta especificada.

Si el fichero ya existe, se leerá y se usará el idioma indicado en ese fichero y se ignorará el *Idioma* especificado en el primer parámetro. Si el fichero es incorrecto, y dependiendo del modo de debug especificado anteriormente, abortará la ejecución o usará el *Idioma* especificado en el primer parámetro.

Si el fichero no existe, usará el *Idioma* especificado en el primer parámetro y creará el fichero de configuración con ese idioma.

### 3.2.2 – Initialize\_AdaIntl (procedimiento)

```
procedure Initialize_AdaIntl (  
  Language           : Language_Type;  
  Default_Domain     : String      := "Language";  
  Debug_Mode         : Debug_Level_Type := No_Debug;  
  Directory           : String      := "";  
  Load_Configuration_File : String  := ""      );
```

El procedimiento “*Initialize\_AdaIntl*” tiene el mismo efecto que la función “*Initialize\_AdaIntl*”. La diferencia, tal como se explica en el punto 3.1.1, es que la función “*Initialize\_AdaIntl*” sirve para iniciar AdaIntl en la fase de declaración. Aparte de eso, los parámetros y su uso es similar. Para más información, véase el punto 3.2.1.

### 3.2.3 – Set\_Default\_Domain

```
procedure Set_Default_Domain (  
    Domain : String );
```

Una vez ejecutado “*Initialize\_AdaIntl*”, es posible cambiar el dominio por defecto en cualquier momento ejecutando este procedimiento. Tan solo se ha de indicar cual es el nombre del archivo de traducción (dominio), y a partir de ese momento se usará ese dominio cuando la instrucción “-” (vease 3.2.9) no especifique un dominio.

### 3.2.4 – Set\_Debug\_Mode

```
procedure Set_Debug_Mode (  
    Debug_Mode : Debug_Level_Type );
```

Una vez ejecutado “*Initialize\_AdaIntl*”, es posible cambiar el nivel de Debug en AdaIntl . A partir de ese momento se usará el nivel de Debug especificado por el parámetro “*Debug\_Mode*”.

Los distintos niveles se pueden ver en el punto 3.1.2.

### 3.2.5 – Set\_Language

```
procedure Set_Language (  
    Language : Language_Type );
```

Una vez ejecutado “*Initialize\_AdaIntl*”, es posible cambiar el idioma por defecto en tiempo de ejecución con esta instrucción. El parámetro “*Language*” puede tener cualquier valor especificado en el punto 3.1.3.

Ejecutando “*Set\_Language*”, **no** se actualizan las frases cargadas en memoria, sino que se actualizarán a medida que se vayan usando los dominios (al acceder a un dominio se compara el lenguaje de los dominios cargados en memoria con el lenguaje por defecto, si no coinciden, se actualiza el dominio en cuestión).

### 3.2.6 – Get\_Default\_Domain

```
function Get_Default_Domain return String;
```

Esta función permite obtener el dominio por defecto que se está usando actualmente.

### 3.2.7 – Get\_Debug\_Mode

```
function Get_Debug_Mode return Debug_Level_Type;
```

Esta función permite obtener el nivel de Debug que se está usando actualmente (vease el punto 3.1.2).



### 3.2.8 – Get\_Language

```
function Get_Language return Language_Type;
```

Esta función permite obtener el idioma por defecto que se está usando actualmente. Especialmente útil debido a que AdaIntl puede cargar cualquier idioma de un fichero de configuración (véase 3.2.1, último parámetro).

### 3.2.9 – “-” (un parámetro)

```
function "-" (  
    Right : String )  
return String;
```

Una de las funciones más importantes y que más se usarán de **AdaIntl**. Es el equivalente a las funciones de GNU/gettext “*gettext()*” y “*\_()*”.

Esta función, al ser ejecutada, obtiene la cadena a traducir de su único parámetro. A continuación obtiene su traducción según el idioma por defecto que esté activo en ese momento y devuelve la cadena traducida. El dominio usado es el dominio por defecto.

Para obtener la cadena traducida, mira en memoria en el dominio por defecto si está almacenada. Si no es así, abre el fichero de traducción (dominio) y carga las frases del dominio.

Si aún así sigue sin estar presenten, entonces *añade* la frase sin traducir al dominio y la devuelve tal cual.

Esto significa que el archivo de traducción se crea *en tiempo de ejecución*. A medida que se vaya ejecutando el programa, y se ejecute “-” con las cadenas de texto, estas se irán guardando en los ficheros de traducción (si estos no existen). Si una línea no se ejecuta, la cadena de texto contenida **no estará** presente en los archivos de traducción.

Se desprende que por tanto la única forma (actualmente) de crear los archivos de traducción por primera vez es ejecutando la aplicación y asegurándose de que se ejecuten todos los “-”.

Si no se ejecutan todos (porque la aplicación falle, o se añadan nuevas cadenas), las nuevas aparecerán al final del archivo de traducción. Por tanto si se dispone de una traducción parcial, tan solo habría que añadir las nuevas líneas al final de cada archivo traducido (y traducir las nuevas cadenas).

Se pueden realizar programas externos, que al igual que [xgettext](#), lean el código fuente y extraigan automáticamente las cadenas a traducir, creando los ficheros adecuados, y de esa forma no haga falta “ejecutar” completamente la aplicación. Sin embargo, por el momento no hay ninguna aplicación que lo haga.

Ejemplo de uso de “-”:

```
Put_Line("-Probando, probando");
```

### 3.2.10 - “-” (dos parámetros)

```
function "-" (  
  Left : String;  
  Right : String )  
return String;
```

Esta función es equivalente a “-” con un solo parámetro, pero con la particularidad de que se ha de especificar el dominio a usar. El parámetro “*left*” es el dominio a usar, y “*right*” es la cadena de texto.

En lugar de usar el dominio por defecto, usará el especificado en “*left*”.

Ejemplo de uso de “-” especificando el dominio:

Queremos mostrar por pantalla “Probando, Probando”, y guardar la cadena en el dominio “Lista\_de\_saludos”.

```
Put_Line("Lista_de_saludos"-"Probando, probando");
```

### 3.2.11 - Get\_Available\_Languages

```
function Get_Available_Languages return Availabe_Languages_Array;
```

Esta función permite conocer en que idiomas está disponible la aplicación. Para ello mira si existe el fichero del dominio por defecto para cada idioma, y devuelve una estructura “*Available\_Languages\_Array*” (3.1.4), donde “*true*” significa que existe el fichero del dominio por defecto y “*false*” que no existe.  
El idioma “*null*” **siempre** devuelve “*false*”.

Ejemplo de uso de “Get\_Available\_Languages”:

Queremos mostrar por pantalla los idiomas disponibles en los que está nuestra aplicación:

```
Idiomas_Disponibles : Availabe_Languages_Array;  
begin  
  Initialize_AdaIntl (Es);  
  Idiomas_Disponibles:=Get_Available_Languages;  
  Put_Line("Los idiomas disponibles son:");  
  for Iterador in Language_Type'range loop  
    if Idiomas_Disponibles(Iterador) then  
      Put_Line(Language_Type'Image(Iterador));  
    end if;  
  end loop;
```

### 3.2.12 - Clean\_AdaIntI

```
procedure Clean_AdaIntI;
```

La instrucción Clean\_AdaIntI se ocupa de borrar de la memoria las frases, destruyendo los árboles y su contenido.

Se **debe** ejecutar esta instrucción antes de finalizar el programa o bien si se desea “reiniciar” AdaIntI en tiempo de ejecución (justo antes de volver a hacer “Initialize\_AdaIntI”).

### 3.3 – Excepciones

Dependiendo del nivel de Debug elegido (Punto 3.1.2), en caso de fallo AdaIntl detendrá la ejecución lanzando excepciones que indicarán el fallo cometido. A continuación se describen las excepciones.

#### 3.3.1 – File\_Not\_Correct

Esta excepción se lanza si el fichero de traducción (dominio) es incorrecto. En el punto 4.2.1 se explica como es un fichero de traducción correcto.

Dado que para que se lance esta excepción es necesario estar en el modo *Only\_Errors* o *Total*, aparecerá un mensaje de error por consola diciendo cual es el fallo (si falta un “ , o si sobran caracteres, etc).

Ejemplo:

La excepción File\_Not\_Correct saltará si faltan las comillas que cierran una cadena:

```
# "Francia"  
227035569="France
```

También puede saltar si el hash no es correcto:

```
# "Francia"  
2270 35569="France"
```

### 3.3.2 – Conf\_File\_Not\_Correct

Esta excepción se lanza si el fichero de configuración es incorrecto. En el punto 4.2.2 se explica como es un fichero de configuración correcto.

Dado que el fichero de configuración se lee para obtener el idioma a usar en lugar del idioma especificado en “*Initialize\_AdaIntl*”, si no es posible leer el archivo de configuración (que es distinto a que no exista, en ese caso se creará si así se ha especificado) se lanzará esta excepción si el nivel de debug es “Stop” (*Only\_Error\_Stop* o *Total\_Stop*); en caso contrario se usará el idioma especificado en “*Initialize\_AdaIntl*”.

### 3.3.3 – Language\_Not\_Valid

Tal como señala el punto 3.1.3, se puede usar cualquier idioma excepto “*Nul*”, el cual sirve para marcar que AdaIntl no está inicializado.

Si salta esta excepción, es que se ha puesto “*Nul*” como primer parámetro de “*Initialize\_AdaIntl*” (lo cual es incorrecto, insistimos en que “*Nul*” no se debe usar)



### 3.3.4 – Not\_Initialized

Si salta esta excepción, es que se ha ejecutado en primer lugar otra instrucción que no es *Initialize\_AdaIntl*, lo cual es incorrecto. Antes de ejecutar cualquier otra instrucción, se ha de inicializar AdaIntl. Para asegurarse de esto, se puede ejecutar la función *Initialize\_AdaIntl* como primera instrucción en la parte declarativa (ver 3.1.1).

## 4- Estructura y funcionamiento interno

### 4.1 – Directorios y ficheros de AdaIntl

El archivo zip que forma AdaIntl está compuesto por los siguientes directorios:

- /src
  - Contiene los ficheros con el código fuente en Ada95 de AdaIntl.
- /example
  - Contiene ejemplos de programas que usan AdaIntl.
- /doc
  - Contiene la documentación de la librería.

#### 4.1.1 - /src

- **adaintl.ads** (MD5: 98e4e51b72fe1a305755965fdf32a41f)
  - Fichero donde se declaran todas las funciones y procedimientos accesibles al programador.
- **adaintl.adb** (MD5: 6b048b86717c273d0b24e7d608af87e2)
  - Cuerpo donde está la implementación del primer nivel de AdaIntl.
- **aa\_tree\_package.ads** (MD5: c324c81a800db22259bf9a93b2c663f5)
  - Fichero de declaración del paquete de arbol AA genérico.  
Hecho por [Mark Allen Weiss](#).
- **aa\_tree\_package.adb** (MD5: fd9a9818f31fa7bcb5749211f5b494f7)
  - Cuerpo con la implementación del arbol AA genérico.  
Hecho por [Mark Allen Weiss](#).
- **d\_arbol\_dominios.ads** (MD5: 93c714a469058a57fff101c4844adf42)
  - Fichero de declaración que especifica y controla los árboles de dominio.  
Los árboles de dominio son las estructuras donde se guardan los nombres de dominios usados hasta el momento. De cada nodo del arbol (que representa un dominio) cuelga un “arbol de frases”, con todas las cadenas de texto de ese dominio en el idioma especificado. Véase punto 4.3.
- **d\_arbol\_dominios.adb** (MD5: fb3c5553f40017767ff62bdaa61a3c33)
  - Cuerpo con la implementación de los árboles de dominio.
- **d\_arbol\_frases.ads** (MD5: 17c10fcd0c7c772a2b6673d48e9b2a4f)
  - Fichero de declaración que especifica y controla los árboles de frases. Los árboles de frase contienen las cadenas de texto leídas de los ficheros de traducción. Cada dominio tiene su propio arbol de frases. Véase punto 4.3.
- **d\_arbol\_frases.adb** (MD5: 238541af538f2007c2e6f90ecc3e1d8d)
  - Implementación de los árboles de frases.
- **d\_archivo\_conf.ads** (MD5: 391c0a4b13bf948c91b5a82d17cd4d46)
  - Fichero de declaración que controla el acceso a los ficheros de configuración  
Véase punto 4.2.2.
- **d\_archivo\_conf.adb** (MD5: 906b5cf06978a4f5c2811ae10e714a1b)
  - Implementación del paquete que controla el acceso a lo ficheros de configuración.
- **d\_frases\_io.ads** (MD5: 7d9be43045bbe1125d2f093f4c08ed36)
  - Fichero de declaración que controla la entrada y salida de cadenas de texto.  
Véase punto 4.2.1.
- **d\_frases\_io.adb** (MD5: fe38f799bf4ef9432c4847f9d20bbfc7)
  - Implementación del paquete que controla la entrada y salida de las cadenas de texto.

- **d\_hash.ads** (MD5: d3293a90c92736601a776b6dd240db67)
  - Declaración del tipo Hash y de la instrucción que permite calcular el Hash de una cadena de texto.
- **d\_hash.adb** (MD5: 03ee2112df7831f4b0495c972e9d31dd)
  - Implementación del cálculo del Hash.
- **d\_idiomas.ads** (MD5: f491f7d850be3d6ec9d5c723ded57706)
  - Listado de idiomas disponibles en AdaIntl (véase punto 3.1.3).
- **d\_idiomas.adb** (MD5: 0b36c16dd004ad7d8c17e13baa39401c)
  - Contiene la implementación de la instrucción “*Language\_Name*” que dadas las siglas de un idioma, devuelve su nombre completo en inglés.
- **d\_debug.ads** (MD5: f8e2e77c2e4c74a2914786eee6d65b41)
  - Listado de niveles de debug disponibles en AdaIntl (véase punto 3.1.2).
- **adaintl\_version.ads** (MD5: fa430b5dccc54f125498b42841eab4a4)
  - Contiene información sobre la versión de AdaIntl, autor, contacto, etc.

#### 4.1.2 - /example

Contiene el código de ejemplo del punto 5, junto con sus ficheros de traducción y configuración. Está listo para ser ejecutado y ver el funcionamiento de AdaIntl.

#### 4.1.3 - /doc

Este directorio contiene la documentación de AdaIntl en formato Open Document Text (ODT) y PDF.

#### 4.1.4 – Instalación

AdaIntl puede ser usado directamente moviendo el contenido del directorio /src al directorio donde esté el código fuente del programa. A continuación bastará con incluirlo haciendo

*with AdaIntl; use AdaIntl;*

en el código fuente de la aplicación.

Sin embargo es recomendable instalar el paquete para no tener que incluir el código completo de AdaIntl en el directorio de la aplicación.

Para ello se pueden seguir estos pasos:

- 1- Extraer el contenido del archivo comprimido en un directorio vacío, como por ejemplo *C:\AdaIntl* o */home/usuario/AdaIntl*
- 2- A continuación se han de compilar las fuentes (en el subdirectorio /src). Por ejemplo, si se usa GNAT se puede ir al subdirectorio /src y ejecutar  
*gnatmake adaintl.adb*

Esto generará una serie de archivos .ali y .o de cada paquete que forma AdaIntl.

- 3- A continuación hay que permitir al compilador acceder a los ficheros en el subdirectorio /src. Para ello se han de añadir las variables de entorno: “ADA\_INCLUDE\_PATH” y “ADA\_OBJECTS\_PATH” y darles el valor del subdirectorio /src. Por ejemplo, en Windows habría que añadir estas variables (suponiendo la ruta de ejemplo anterior):

```
ADA_INCLUDE_PATH=C:\AdaIntl\src
ADA_OBJECTS_PATH=C:\AdaIntl\src
```

Con estos pasos ya podrás usar AdaIntl en cualquier aplicación que crees.

## 4.2 – Ficheros de traducción y configuración

### 4.2.1 – Dominios

Los dominios son los ficheros de traducción donde se guardan las cadenas de texto que usa la aplicación.

Los ficheros de traducción de AdaIntl siguen el siguiente patrón:

#### – Comentarios

- Las líneas que comienzan por # son comentarios y se ignoran. Se puede usar para dar explicaciones, comentar detalles de la traducción, o dar información sobre el idioma y programa.

#### – Cadenas de texto

- Las cadenas de texto vienen acompañadas del hash de la frase original sin traducir (para su identificación) y van entrecomilladas. Por ejemplo:
  - `84574961="Italy"`  
Si bien no es necesario, por claridad antes de la línea con la cadena de texto traducida se coloca una línea comentada con la cadena original. En este caso:  
`# "Italia"`
  - `84574961="Italy"`
- De esta forma la traducción es más fácil y sencilla.

Cuando en el código fuente aparezca

`-"Italia"`

independientemente de donde esté (en un Put, como cadena por defecto, etc), AdaIntl generará el hash de la cadena pasada por parámetro (en este caso “Italia”), y buscará la cadena con hash 84574961 para a continuación devolverla, en nuestro ejemplo “Italy”.

Para leer los ficheros de traducción, AdaIntl sigue básicamente esta gramática: *EOF* se refiere a fin de fichero, *EOL* se refiere a salto de línea, *any* se refiere a cualquier carácter excepto salto de línea o fin de fichero, etc. Los símbolos terminales no están en negrita.

**Doc:** *EOF* | **Linea Doc**

**Linea:** *EOL* | **L**

**L:** # {*any*} | **Hash** = “{*any*}”

**Hash:** {0,1,2...9}

Sin embargo, dado que la frase a traducir debe estar entre comillas, si se desea usar comillas estas deberán estar precedidas de “\”. Igualmente para imprimir el símbolo “\” este deberá estar dos veces seguidas. Ejemplo:

```
2270355="He said: \"Hello\" "
```

Ejemplo de fichero de traducción:

```
# Internationalization File Created with AdaIntl v0.1alpha
# AdaIntl v0.1alpha is an internationalizing tool for Ada95 programs created by Andres_age
# More info at www.canalada.org
```

```
# File: 'Paises.EN'
# Language: 'EN'
```

```
# "Italia"
84574961="Italy"
```

```
# "España"
80375921="Spain"
```

```
# "Francia"
227035569="France"
```

¡¡Es **MUY IMPORTANTE** a la hora de traducir limitarse a cambiar el contenido entrecomillado, el hash NUNCA se debe modificar o de lo contrario no se podrá relacionar la cadena con su correspondiente traducción!!

Estos ficheros son creados en tiempo de ejecución si no existen, y se debe mantener “sincronizado” su contenido con las cadenas en memoria.

Para ello se contemplan los siguientes casos:

- Fichero no existe; memoria vacía:
  - Se crea tan pronto como se acceda por primera vez a él y se deja vacío.
- Fichero existe; memoria vacía:
  - Si se accede por primera vez a un dominio y el fichero existe, las frases que contiene se cargan en memoria en el árbol de frases correspondiente.
- Fichero existe; frase nueva que no está en memoria
  - Si se requiere una frase que no está cargada en memoria, significa que tampoco está contenida en el fichero de traducción (si lo estuviese en el primer acceso al dominio se hubiera cargado en memoria esa frase). Por tanto la frase se añade al árbol de frases del dominio y se escribe la nueva frase al final del fichero de traducción.

La compatibilidad de lectura de archivos de GNU/gettext [\\*.po](#) es una de las tareas pendientes (punto 6), o incluso el uso de la estructura “msgid/msgstr” en lugar de la estructura usada actualmente “hash=cadena”.

## 4.2.2 – Ficheros de configuración

El procedimiento/función “*Initialize\_AdaIntl*” permite cargar el idioma que usará la aplicación de un fichero de configuración, tal como se explica en el punto 3.2.1.

Su sintaxis es parecida a la de los ficheros de traducción (4.2.1): Las líneas que comienzan con # son comentarios, y el resto es código.

De momento los ficheros de configuración tienen dos campos: el código del idioma a usar y su nombre.

El nombre es tan solo orientativo (se puede poner el que se quiera), el código es lo que usa AdaIntl para elegir el idioma.

Ejemplo de fichero de configuración con el castellano como idioma a usar:

```
# Internationalization Configuration File Created with AdaIntl v0.1alpha
# AdaIntl v0.1alpha is an internationalizing tool for Ada95 programs created by Andres_age
# More info at www.canalada.org

# File: 'Language.txt'
Language = "ES"
Language_name = "Spanish"
```

Si el fichero de configuración es incorrecto, bien ignorará el contenido y se usará el idioma escogido en “*Initialize\_AdaIntl*” o bien lanzará una excepción (ver 3.3.2) según el nivel de debug elegido.



### 4.3 – *Funcionamiento interno*

Tal como se mencionaba en el punto 4.2.1, las cadenas traducidas se mantienen en memoria, y de esa forma no es necesario acceder al fichero de traducción en cada acceso a una cadena de texto.

Para identificar la cadena traducida, se usa un hash de la cadena original.

En lugar de usar una tabla de hashing, se usa un árbol para almacenar las frases y sus hashes, debido a que permite un acceso mucho más rápido y una flexibilidad de tamaño que una tabla de hashing no tiene.

Por ejemplo, una tabla de hashing de dispersión abierta consistente en un vector de 1000 posiciones y con listas enlazadas dinámicas tendría un tamaño mínimo considerable (1000 elementos+listas vacías) y en el mejor de los casos el acceso es constante (1) pero en el peor sería (n). En cambio con un árbol, el tiempo de acceso es (log n) en el peor de los casos, y el tamaño es proporcional al número de cadenas de texto que hay (véase el punto 6 sub-apartado “*Tratar colisiones*” para ver un problema que plantea esta solución).

De forma que AdaIntl usa árboles para mantener las cadenas en memoria y acceder a ellas de forma fácil y rápida cuando se necesitan.

Sin embargo, ¿como diferencia las cadenas de cada dominio?

Para ello tiene a su vez otro árbol donde se almacenan los dominios. Cada nodo del árbol representa un dominio, y de cada nodo del árbol cuelga su *propio* árbol de frases con todas las cadenas de texto contenidas en ese dominio.

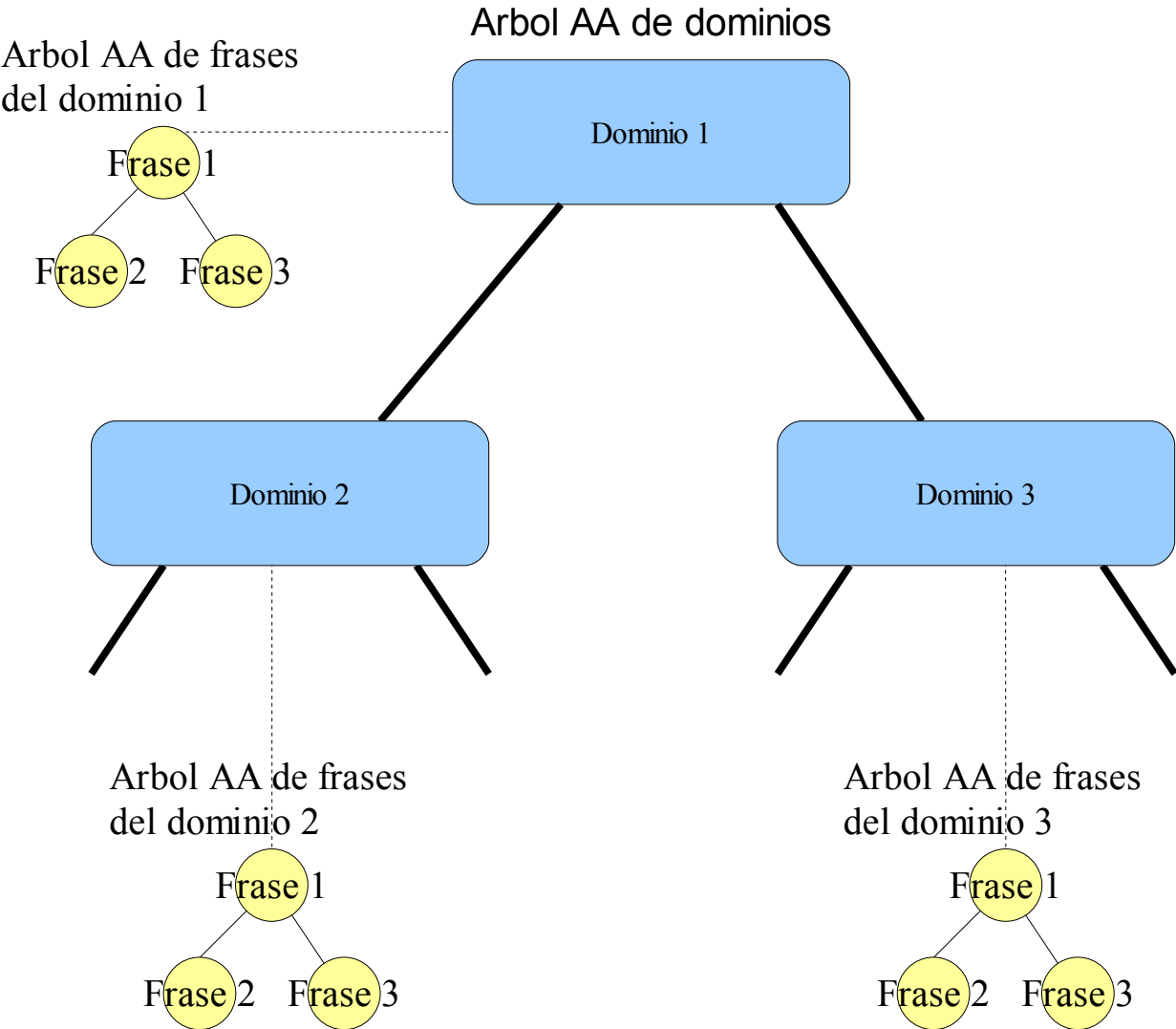
En la figura de la página siguiente se observa un esquema.

Por tanto, cuando AdaIntl quiere buscar la cadena de texto traducida de una cadena dada, calcula el hash de la original, busca el dominio usado y accede al árbol de frases del dominio correspondiente. Usando el hash, busca la cadena traducida y devuelve su valor. Si no encuentra el dominio o la frase, lo añade al árbol tal como se explica en el punto 4.2.1.

Del mantenimiento y acceso a cada árbol de frases se ocupa el paquete *d\_arbol\_frases*, mientras que del árbol de dominios se ocupa el paquete *d\_arbol\_dominios*.

Del cálculo del hash se ocupa el paquete *d\_hash*.

Cuando hay que escribir o leer frases del fichero de traducción, AdaIntl usa el paquete *d\_frases\_io*.



## 5- Código de ejemplo

A continuación propondremos un pequeño programa que ejemplifica el uso de **AdaIntl** . El programa hará lo siguiente:

- Inicializar **AdaIntl** .
- Imprimir frases por pantalla en 3 idiomas. Estas frases se guardarán en el dominio por defecto "*Frases*".
- Imprimir nombres de paises por pantalla en 3 idiomas. Estos nombres se guardarán en el dominio explícito (es decir, no será el dominio por defecto) "*Países*".
- Los dominios se guardarán en el mismo directorio que el programa.
- El idioma variará dinámicamente a lo largo de la ejecución.
- El idioma se cargará del archivo de configuración "Language.txt", por defecto será el castellano (ES).
- Por último, se limpiará la memoria.

En la primera ejecución se crearán los ficheros de traducción, de configuración, etc. Se imprimirá en pantalla el código por defecto (recordemos que **AdaIntl** **NO** traduce). Una vez creados estos ficheros, será necesario traducir manualmente los dominios, y en las siguientes ejecuciones se mostrarán por pantalla las frases en el idioma elegido (si estas cadenas de texto han sido correctamente traducidas).

```
with Adaintl;
use Adaintl;

with Ada.Text_IO;
use Ada.Text_IO;

procedure Ejemplo is

  C : Character;

begin

  Initialize_Adaintl (Es, "Frases", No_Debug, "", "Language.txt");

  Put_Line("A continuacion se ejecutaran algunas pruebas");
  Put_Line("para comprobar el correcto funcionamiento de Adaintl.");
  New_Line;
  Put_Line("Adaintl extrae las frases de los archivos de traduccion");
  Put_Line(
    "automaticamente. Estos ficheros se llaman 'dominios' (domain).");
  New_Line;
  Put_Line(
    "Las frases a traducir vienen precedidas por '-' en el codigo fuente.");
  Put_Line("Se puede usar el dominio por defecto, o especificar uno.");
  Put_Line("El idioma se puede cambiar en tiempo de ejecucion.");

  New_Line(2);
  Put_Line("Castellano (o el idioma indicado en Language.txt:");
  Put_Line("-" & " Frase numero 1 para probar");
  Put_Line("-" & " Frase numero 2 para probar");
  Put_Line("Países" & " España"); -- Dominio (explicito): "Países"
  Put_Line("Países" & " Italia"); -- Dominio (explicito): "Países"
```

```
New_Line(2);
Set_Language(En);
Put_Line("Ingles:");
Put_Line("-" Frase numero 1 para probar");
Put_Line("-" Frase numero 2 para probar");
Put_Line("Países"-" España"); -- Dominio (explicito): "Países"
Put_Line("Países"-" Italia"); -- Dominio (explicito): "Países"
```

```
New_Line(2);
Set_Language(Ca);
Put_Line("Catalan:");
Put_Line("-" Frase numero 1 para probar");
Put_Line("-" Frase numero 2 para probar");
Put_Line("Países"-" España"); -- Dominio (explicito): "Países"
Put_Line("Países"-" Italia"); -- Dominio (explicito): "Países"
```

```
Clean_Adaintl;
```

```
Get_Immediate(C);
```

```
end Ejemplo;
```

**Nota:** Este ejemplo está en la carpeta */example*.

## 6- Lista de cosas por mejorar o arreglar

### Probar a fondo AdaIntl

AdaIntl ha sido probada en sistemas Windows y Unix con GNAT.

Sin embargo sería conveniente probarlo con otros compiladores (teóricamente no debería dar ningún problema) y por supuesto usarlo en aplicaciones reales, etc. En otras palabras, “darle rodaje”.

### Compatibilidad con archivos \*.PO

Actualmente AdaIntl usa ficheros de traducción con formato propio, tipo  
*Hash = Cadena*

Sin embargo se podría aprovechar los programas hechos para los ficheros \*.PO que se generan con GNU/gettext (como *poEdit*), y así poder facilitar la traducción. El formato de los archivos PO es:

*msgid = Cadena original*

*msgstr = Cadena traducida*

La entrada y salida de los ficheros de traducción en AdaIntl está “encapsulada” en el paquete *d\_frases\_io*.

Modificando la gramática se podría admitir ambos formatos (formato antiguo para mantener compatibilidad a la hora de leer archivos de traducción antiguos, y usar el nuevo para leer/escribir).

### Tratar colisiones

Actualmente se usa un hash de la cadena original para identificar la traducida. El algoritmo usado es una implementación en Ada95 del [Elf Hash](#).

En lugar de usar una tabla de hashing, se usa un arbol para almacenar las frases y sus hashes, debido a que permite un acceso mucho más rápido ( $\log n$  en el peor de los casos) y una flexibilidad de tamaño que una tabla de hashing no tiene. Sin embargo, tal como se ha implementado, no se contempla la posibilidad de colisiones: si se produce una colisión en el hash (es decir dos cadenas distintas generan el mismo hash), para AdaIntl las dos cadenas son iguales, con las limitaciones que ello conlleva.

De momento, si se detecta una colisión (cosa que no debería ocurrir con mucha probabilidad, o directamente no ocurrir en muchos casos), se puede modificar una de las frases que “colisionan”, por ejemplo añadiendo un espacio al final, etc.

Es necesario mejorar este punto y tratar el caso en el que ocurran colisiones.

Otra posibilidad es no usar un hash, sino la cadena original. Por tanto si hay dos elementos “iguales”, ¡es que es el mismo string!

Si bien así se podría aprovechar para el punto anterior (mismo formato que archivos PO), el coste de recorrer un arbol de strings en lugar de usar un hash es muchísimo mayor.

## **Traducir la documentación**

Sería necesario traducir esta documentación al inglés como mínimo.

## **Concurrencia**

Habría que preparar AdaIntI para trabajar en aplicaciones concurrentes, donde distintos “tasks” acceden “a la vez” a frases, modificando los árboles de frases/dominio, etc.

## 7- Version e historial

La última versión de AdaIntl es la *0.1alpha*.

### Historial:

12 de julio de 2006 – 0.1alpha  
– ¡Primera versión!

## 8- Créditos y licencia

AdaIntl fue creada por [Andres\\_age](#) en julio de 2006.

Agradecimientos a:

- [Mark Allen Weiss](#) por permitirme usar y modificar una implementación suya de árboles AA.
- [Gneuromante](#) por su colaboración, ayuda y consejos.
- [Flash Noi](#) y a [Canal #Ada](#) por hospedar el proyecto.

### 8.1- LGPL

AdaIntl se ha liberado bajo la licencia LGPL. En la [página de la FSF](#) se puede leer la licencia en versión original. A continuación se incluye una traducción en castellano de la LGPL.

<http://gugs.sindominio.net/gnu-gpl/lgpl-es.html>

GNU LESSER GENERAL PUBLIC LICENCE (traducción)  
Versión 2.1, Febrero 1999

Sobre la traducción:

- Autor de la la traducción: Rafael Palomino <[rafaelus4@eresmas.net](mailto:rafaelus4@eresmas.net)>
- Fecha de la traducción: mayo de 2001

Notas sobre la traducción:

1. El término *library* ha sido traducido en este texto como "librería", en lugar de su traducción correcta "biblioteca" al estar esta primera denominación más de acuerdo con los términos informáticos utilizados en castellano.
2. El objetivo primordial de esta traducción es dar a explicar la licencia LGPL, sin embargo dicha traducción a debido ser muy "literal" en varias ocasiones para no caer en interpretaciones personales.

This is an unofficial translation of the GNU Lesser General Public License into spanish. It was not published by the Free Software Foundation, and does not legally state the distribution terms for software that uses the GNU LGPL — only the [original English text of the GNU LGPL](#) does that. However, we hope that this translation will help spanish speakers understand the GNU GPL better.

Esta es una traducción no oficial de la GNU Lesser General Public License. No fue publicada por la Free Software Foundation, y no establece legalmente los términos de distribución de software que usa la GNU LGPL; sólo el [texto inglés original de la GNU LGPL](#) lo hace. Sin embargo, esperamos que esta traducción ayudará a los hispanohablantes a entender la GNU LGPL mejor.

Copyright © 1991, 1999

Free Software Foundation, Inc. 59, Temple Place, Suite 330, Boston, MA 02111-1307 USA

. Se permite a cualquier persona la copia y distribución de copias literales de este documento de licencia, su modificación no está permitida.

(Esta es la primera versión emitida de la GPL Menor. Es también la sucesora de la *GNU Library Public Licence*, versión 2, de aquí la versión 2.1)



## Preámbulo

Las licencias, para la mayoría del software, están diseñadas para eliminar sus libertades a la hora de compartirlo o cambiarlo. En contraste, las Licencias Públicas Generales GNU pretenden garantizar su libertad para compartir y cambiar software libre; para asegurar que el software es libre para todos los usuarios.

Esta licencia, la Licencia Pública General Menor (*Lesser General Public Licence*), se aplica a algunos paquetes de software diseñados específicamente —típicamente librerías— de la Free Software Foundation y de otros autores que deciden usarla. Usted puede usarla también, pero le sugerimos que piense primero cuidadosamente si esta licencia o la General Public Licence ordinaria, es o no la estrategia mejor a usar en un caso particular, basándose en las explicaciones siguientes.

Cuando hablamos de software libre, nos referimos a libertad de uso, no a precio. Nuestras Licencias Públicas Generales están diseñadas para asegurar que usted tiene la libertad de distribuir copias de software libre (y cobrar por este servicio si lo desea); que recibe el código fuente o que puede obtenerlo si lo quiere; que puede modificar el software y usar partes de él en nuevos programas libres; y de que ha sido informado de que puede hacer estas cosas.

Para proteger sus derechos, necesitamos hacer restricciones que prohíban a los distribuidores el negarle estos derechos o a pedirle que renuncie a esos derechos. Estas restricciones se traducen en ciertas responsabilidades para usted si distribuye copias de la librería o si la modifica.

Por ejemplo, si distribuye copias de la librería, bien gratis o a cambio de un importe, debe dar a los receptores todos los derechos que nosotros le dimos a usted. Debe asegurarse que ellos, también, reciban o puedan obtener el código fuente. Si usted enlaza otro código con la librería, debe proporcionar a los receptores los ficheros objeto completos, de forma que ellos puedan reenlazarlos con la librería después de hacer cambios en la librería y recompilarla. Y debe mostrarles estos términos para que conozcan sus derechos.

Nosotros protegemos sus derechos con un método de dos pasos: (1) obtenemos los derechos de autor de la librería, y (2) le ofrecemos esta licencia que le da permiso legal para copiar, distribuir, y/o modificar la librería.

Para proteger a cada uno de los distribuidores, queremos dejar muy claro que no existe garantía para la librería libre. Además, si la librería es modificada por alguien y se transmite, los receptores deberían saber que lo que ellos tienen no es la versión original, de forma que la reputación del autor original no se vea afectada por problemas que podrían ser introducidos por otros.

Finalmente, las patentes de software plantean una amenaza constante para la existencia de cualquier programa libre. Nosotros deseamos asegurar que una empresa no pueda limitar eficazmente a los usuarios de un programa libre mediante la obtención de una licencia restrictiva de un titular de patente. Por lo tanto, insistimos en que cualquier licencia de patente obtenida para una versión de la librería debe ser consistente con todas las libertades de uso especificadas en esta licencia.

La mayoría del software GNU, incluyendo algunas librerías, está cubierto por la Licencia Pública General GNU ordinaria. Esta licencia, la Licencia Pública General Menor GNU, se aplica a ciertas librerías señaladas y es bastante diferente de la Licencia Pública General ordinaria. Nosotros usamos esta licencia para ciertas librerías con el objeto de permitir el enlace de esas librerías dentro de programas no libres.

Cuando un programa se encuentra enlazado con una librería, bien estáticamente o usando una librería compartida, la combinación de los dos es, legalmente hablando, un trabajo combinado, un derivado de la librería original. La Licencia Pública General ordinaria permitiría tal enlace sólo si la combinación completa ajusta sus criterios de libertad. La Licencia Pública General Menor permite un criterio más laxo para el enlazado de otro código con la librería.

Denominamos a esta licencia como Licencia Pública General "Menor" porque hace Menos que la Licencia Pública General ordinaria para proteger las libertades del usuario. También proporciona a los desarrolladores de programas libres Menos ventajas sobre los programas no libres competidores. Estas desventajas son la razón por la cual nosotros usamos la Licencia Pública General ordinaria para la mayoría de las librerías. Sin embargo, la licencia Menor proporciona ventajas en ciertas circunstancias especiales.

Por ejemplo, en raras ocasiones puede haber una necesidad especial de fomentar lo más ampliamente posible el uso de una determinada librería, de forma que esta se convierta en un estándar. Para conseguir esto, se debe permitir a los programas no libres el uso de estas librerías. Un caso más frecuente es aquel en el que una librería libre hace el mismo trabajo que el que realizan las librerías no libres más ampliamente usadas. En este caso, hay poco que ganar limitando la librería únicamente al software libre, de manera que usamos la Licencia Pública General Menor.

En otros casos, el permiso para usar una librería determinada en programas no libres posibilita a un mayor número de gente a usar una gran cantidad de software libre. Por ejemplo, el permiso para utilizar la librería GNU C en programas no libres posibilita a mucha más gente a usar al completo el sistema operativo GNU, así como su variante, el sistema operativo GNU/LINUX.

Aunque la Licencia Pública General Menor es menos protectora para las libertades del usuario, asegura que el usuario de un programa que está enlazado con la librería tiene la libertad y los medios para ejecutar ese programa usando una versión modificada de la librería.

Los términos y las condiciones exactas para la copia, distribución y modificación se indican a continuación. Preste

especial atención a la diferencia entre un "trabajo basado en la librería" y un "trabajo que utiliza la librería". El primero contiene código derivado de la librería, mientras que el último debe estar unido con la librería para ser ejecutado.

## Términos y condiciones para la copia, distribución y modificación

0.-

El acuerdo de esta licencia se aplica a cualquier librería de software u otro programa que contenga un aviso colocado por el titular de los derechos de autor u otras partes interesadas diciendo que puede ser distribuido bajo los términos de esta Licencia Pública General Menor (llamada también "esta Licencia"). A cada titular de permiso se le designa por "usted".

Una "librería" significa una colección de funciones y/o datos de software, preparados para ser enlazados de una forma cómoda con programas de aplicación (que usan algunas de estas funciones y datos) para formar ejecutables.

La "Librería" en lo que sigue, se refiere a cualquier trabajo o librería de software que haya sido distribuido bajo estos términos. Un "trabajo basado en la Librería" significa que, o la Librería o cualquier trabajo derivado, están bajo la ley de derechos de autor: es decir, un trabajo que contiene a la Librería o a una parte de ella, ya sea de forma literal o con modificaciones y/o traducida de forma clara a otro idioma (mas abajo se incluye la traducción sin restricción en el término "modificación").

El "código fuente" para un trabajo se refiere a la forma preferida del trabajo para hacer modificaciones en él. Para una Librería, el código fuente completo significa todos los códigos fuente para todos los módulos que contenga la librería, más cualquier fichero de definición de interfaz asociado, y los guiones (scripts) asociados para controlar la compilación y la instalación de la librería.

Otras actividades que no sean la copia, distribución y modificación no se encuentran cubiertas por esta Licencia; se encuentran fuera de su objetivo. La opción de ejecutar un programa utilizando la Librería no esta restringido, y el resultado de dicho programa esta cubierto únicamente si su contenido constituye un trabajo basado en la Librería (independientemente del uso de la Librería como herramienta para escribirlo). Que esto sea cierto va a depender de lo que haga la Librería y de lo que haga el programa que utiliza la Librería.

1.-

Puede copiar y distribuir copias literales del código fuente completo de la Librería tal y como la recibe, en cualquier medio, a condición de que usted publique de forma manifiesta y apropiada, en cada una de las copias, un aviso conveniente de derechos de autor y una renuncia de garantía; mantenga intactas todas las notificaciones que se refieran a esta Licencia y a la ausencia de cualquier garantía; y distribuya una copia de esta Licencia junto con la Librería.

Puede cobrar un importe por el acto físico de traspasar una copia y puede, a opción suya, ofrecer una protección de garantía a cambio de un importe.

2.-

Puede modificar su copia o copias de la Librería o de cualquier parte de ella, formando así un trabajo basado en la Librería, y copiar y distribuir tales modificaciones o trabajo bajo los términos de la sección 1 de arriba, siempre que usted también cumpla con todas estas condiciones:

a) El trabajo modificado debe ser por si mismo una librería.

b) Debe hacer que los ficheros modificados lleven avisos llamativos, declarando que usted cambió los ficheros y la fecha de cualquier cambio.

c) Debe hacer que a todo el trabajo le sea concedida una licencia, sin cargo a terceras partes, bajo los términos de esta Licencia.

d) Si una facilidad en la Librería modificada se refiere a una función o a una tabla de datos, que deba ser suministrada por un programa de aplicación que usa la facilidad de otra manera que como un argumento pasado cuando la facilidad es invocada, entonces debe hacer un esfuerzo de buena fe para asegurar que, en caso de

que una aplicación no suministre tal función o tabla, la facilidad aun funcione y haga que cualquier parte de su finalidad siga siendo significativa.

(Por ejemplo, una función en una librería para computar raíces cuadradas tiene un propósito que está bien definido completamente, independientemente de la aplicación. Por tanto, la Subsección 2d exige que cualquier función o tabla suministrada por la aplicación y usada por esa función debe ser opcional: si la aplicación no la suministra, la función raíz cuadrada debe seguir computando raíces cuadradas).

Estos requisitos se aplican al trabajo modificado como un todo. Si hay secciones identificables de ese trabajo que no derivan de la Librería, y pueden ser consideradas razonablemente independientes y trabajos separados, por ellas mismas, entonces esta Licencia y sus términos, no se aplicarán a aquellas secciones cuando usted los distribuye como trabajos separados. Pero cuando usted distribuye estas mismas secciones como parte de un todo, que es un trabajo basado en la Librería, la distribución del todo debe estar bajo los términos de esta Licencia cuyos permisos para otras licencias se extienden a todo el conjunto, y por tanto a todas y cada una de las partes, sin tener en cuenta quien lo escribió.

Así pues, la intención de esta sección no es exigir derechos o discutir los derechos de un trabajo escrito completamente por usted; más bien, la intención es ejercer el derecho a controlar la distribución de trabajos derivados o colectivos basados en la Librería.

Además, la mera agregación de otro trabajo no basado en la Librería con la Librería (o con un trabajo basado en la Librería) en un volumen de almacenaje o en un medio de distribución, no pone al otro trabajo bajo los objetivos de esta Licencia.

### 3.-

3.- Usted puede optar por aplicar a una determinada copia de la Librería, los términos de la Licencia Pública General GNU ordinaria en vez de los de esta Licencia. Para hacer esto, debe alterar todas las notificaciones que se refieren a esta Licencia, para que se refieran a la Licencia Pública General GNU ordinaria, versión 2, en vez de a esta Licencia. (Si ha aparecido una versión más reciente que la versión 2 de la Licencia Pública General GNU ordinaria, entonces, si lo desea, puede especificar esa nueva versión). No haga ningún otro cambio en estas notificaciones.

Una vez que se haya hecho este cambio en una copia dada, es irreversible para esa copia, de modo que la Licencia Pública General GNU ordinaria se aplica a todas las copias siguientes y a trabajos derivados realizados a partir de esa copia.

Esta opción es útil cuando usted desea copiar parte del código de la Librería dentro de un programa que no es una librería.

### 4.-

Puede copiar y distribuir la Librería (o una porción o derivado de ella, bajo la Sección 2) en código objeto o forma ejecutable bajo los términos de las Secciones 1 y 2 arriba indicadas, siempre que la acompañe con el correspondiente código fuente legible (a máquina) completo, que debe ser distribuido bajo los términos de las Secciones 1 y 2 de arriba, en un medio usado habitualmente para el intercambio de software.

Si la distribución del código objeto se hace ofreciendo el acceso a su copia desde un lugar designado, entonces ofreciendo un acceso equivalente a la copia del código fuente desde el mismo sitio se satisfacen los requisitos para la distribución del código fuente, aunque las terceras partes no estén obligadas a copiar el código fuente junto con el código objeto.

### 5.-

Un programa que no contiene derivado de ninguna porción de la Librería, pero está diseñado para trabajar con la Librería al ser compilado o enlazado con ella, se denomina un "trabajo que usa la Librería". Dicho trabajo, por separado, no es un trabajo derivado de la Librería, y por tanto cae fuera del ámbito de esta Licencia.

Sin embargo, enlazando un "trabajo que usa la Librería" con la Librería, crea un ejecutable que es un derivado de la Librería (porque contiene porciones de la Librería). El ejecutable está por tanto cubierto por esta Licencia. La sección 6 expone los términos para la distribución de tales ejecutables.

Cuando un "trabajo que usa la Librería" utiliza material de un fichero cabecera que forma parte de la Librería, el código objeto del trabajo puede ser un trabajo derivado de la Librería aunque el código fuente no lo sea. Que esto sea cierto es especialmente significativo si el trabajo puede ser enlazado sin la Librería, o si el trabajo es por sí mismo una librería. El límite para que esto sea cierto no está definido con precisión por la ley.

Si dicho fichero objeto utiliza solo parámetros numéricos, esquema de estructura de datos, y pequeñas macros y pequeñas funciones en línea (diez líneas o menos de longitud), entonces el uso del fichero objeto no está restringido, sin tener en cuenta si esto es legalmente un trabajo derivado. (Ejecutables que contengan este código objeto y porciones de la Librería estarán aun bajo la Sección 6).

En caso contrario, si el trabajo es un derivado de la Librería, usted puede distribuir el código objeto del trabajo bajo los términos de la Sección 6. Cualquier ejecutable que contenga a ese trabajo también cae bajo la Sección 6, esté o no enlazado con la Librería.

6.-

Como excepción a las secciones anteriores, puede también combinar o enlazar un "trabajo que usa la Librería" con la Librería para producir un trabajo que contenga porciones de la Librería, y distribuir ese trabajo bajo los términos de su elección, siempre que los términos permitan la modificación del trabajo por el uso propio del cliente y la ingeniería inversa para la depuración de tales modificaciones.

Debe incluir con cada copia del trabajo una notificación de que la Librería se utiliza en él, y de que la Librería y su uso están cubiertos por esta Licencia. Debe suministrar una copia de esta Licencia. Si el trabajo, durante su ejecución, muestra notas de derechos de autor, usted deberá incluir entre ellas las notas de derechos de autor de la Librería, así como una referencia que dirija al usuario a la copia de esta Licencia. Además, usted debe hacer una de estas cosas:

a) Acompañar el trabajo con el correspondiente código fuente legible (a máquina) completo de la Librería, incluyendo cualquier cambio que fuera utilizado en el trabajo (el cual debe ser distribuido bajo las Secciones 1 y 2 de arriba); y, si el trabajo es un ejecutable enlazado con la Librería, con el completo, legible (a máquina) "trabajo que usa la Librería", como código objeto y/o código fuente, de forma que el usuario pueda modificar la Librería y reenlazar entonces para producir un ejecutable modificado que contenga la Librería modificada. (Se entiende que el usuario que cambia los contenidos de los archivos de definiciones en la Librería no necesariamente será capaz de recompilar la aplicación para usar las definiciones modificadas).

b) Usar un mecanismo de librería compartida adecuado para enlazar con la Librería. Un mecanismo adecuado es uno que (1) utiliza en tiempo de ejecución una copia de la librería que está ya presente en el ordenador del usuario, en vez de copiar funciones de librería dentro del ejecutable, y (2) funcionará correctamente con una versión modificada de la librería, si el usuario instala una, mientras que la versión modificada sea de interfaz compatible con la versión con la que se hizo el trabajo.

c) Acompañar el trabajo con una oferta escrita, válida por tres años al menos, para proporcionar a dicho usuario los materiales especificados en la Subsección 6a de arriba, por un precio no superior al coste de realizar esta distribución.

d) Si la distribución del trabajo se hace ofreciendo el acceso a la copia desde un lugar determinado, ofrecer un acceso equivalente para la copia de los materiales especificados anteriormente desde el mismo lugar.

e) Verificar que el usuario ha recibido ya una copia de estos materiales o que usted ya le ha enviado una copia a este usuario.

Para un ejecutable, la forma requerida del "trabajo que usa la Librería" debe incluir todos los programas de datos y utilidades necesitados para reproducir el ejecutable desde él. Sin embargo, como una excepción especial, los materiales a distribuir no necesitan incluir nada de lo que es distribuido normalmente (ya sea en forma binaria o fuente) con los componentes principales (compilador, kernel, y demás) del sistema operativo en el cual funciona el ejecutable, a menos que el componente por él mismo acompañe al ejecutable.

Puede suceder que este requisito contradiga las restricciones de la licencia de otras librerías propietarias que no acompañan normalmente al sistema operativo. Dicha contradicción significa que no puede usar estas y la Librería juntas en un ejecutable que usted distribuya.

7.-

Puede colocar facilidades de librería, que son un trabajo basado en la Librería, juntas en una sola librería junto con otras facilidades de librería no cubiertas por esta Licencia, y distribuir dicha librería combinada, con tal que la distribución separada del trabajo basado en la Librería y de las otras facilidades de librería esté, por lo demás, permitida, y con tal que usted haga estas dos cosas:

a) Acompañar la librería combinada con una copia del mismo trabajo basado en la Librería, sin combinar con ningunas otras facilidades de librería. Esto debe ser distribuido bajo los términos de las Secciones anteriores.

b) Incluir una notificación destacada con la librería combinada del hecho de que parte de ella es un trabajo basado en la Librería, y explicando donde encontrar las formas sin combinar acompañantes del mismo trabajo.

8.-

No debe copiar, modificar, sublicenciar, enlazar con, o distribuir la Librería excepto como se estipula expresamente bajo esta Licencia. Cualquier otro intento para copiar, modificar, sublicenciar, enlazar con, o distribuir la Librería no es válido, y terminará automáticamente con sus derechos bajo esta Licencia. Sin embargo, aquellas partes que hayan recibido copias o derechos bajo esta Licencia por medio de usted no tendrán terminadas sus licencias mientras que dichas partes permanezcan en plena conformidad.

9.-

No se le exige que acepte esta Licencia, puesto que no la ha firmado. Sin embargo, ninguna otra cosa le concede a usted el permiso para modificar o distribuir la Librería o sus trabajos derivados. Estas acciones están prohibidas por ley si usted no acepta esta Licencia. Por tanto, al modificar o distribuir la Librería (o cualquier trabajo basado en la Librería), usted indica su aceptación de esta Licencia para hacerlo, y todos sus términos y condiciones para copiar, distribuir o modificar la Librería o los trabajos basados en ella.

10.-

Cada vez que usted distribuye la Librería (o cualquier trabajo basado en la Librería), el receptor recibe automáticamente una licencia del titular original de la Licencia para copiar, distribuir, enlazar con o modificar la Librería sujeto a estos términos y condiciones. Usted no debe imponer ninguna restricción posterior sobre el ejercicio de los receptores de los derechos otorgados aquí mencionados. Usted no es responsable de imponer conformidad con esta Licencia por terceras partes.

11.-

Si, como consecuencia de un juicio o infracción de patente o por cualquier otra razón (no limitada a asuntos de patente) a usted se le imponen condiciones (sea orden judicial, acuerdo u otras) que contradigan las condiciones de esta Licencia, eso no le dispensa de las condiciones de esta Licencia. Si usted no puede distribuirla de tal forma que satisfaga simultáneamente sus obligaciones con respecto a esta licencia y cualquier otras obligaciones pertinentes, entonces como consecuencia, no debe en absoluto distribuir la Librería. Por ejemplo, si una licencia de patente no permitiera la redistribución libre de derechos de autor de la Librería a todo aquellos que reciben copias directamente o indirectamente a través de usted, entonces la única forma en la que podría satisfacer tanto esto como esta Licencia sería abstenerse completamente de la distribución de la Librería.

Si cualquier parte de esta sección se considera inválida o inaplicable bajo cualquier circunstancia particular, se intentará aplicar el balance de la sección, y en otras circunstancias se intentará aplicar la sección como un todo.

No es el propósito de esta sección el inducirle a usted a infringir cualquier demanda de derechos de patente u otros derechos de propiedad o el impugnar la validez de tales demandas; esta sección tiene el único propósito de proteger la integridad del sistema de distribución de software libre, lo cual se lleva a cabo mediante prácticas de licencia pública. Mucha gente ha hecho generosas contribuciones al amplio rango de software distribuido a través de este sistema, confiando en la firme aplicación de este sistema; es decisión de autor/donante decidir si el o ella desea distribuir software a través de cualquier otro sistema y una licencia no puede imponer esa elección.

Esta sección tiene el propósito de esclarecer a fondo lo que se cree ser una consecuencia del resto de esta licencia.

12.-

Si la distribución y/o uso de la Librería está restringida en ciertos países mediante patentes o interfaces con derechos de autor, el propietario de los derechos de autor originales, el cual puso la Librería bajo esta Licencia, puede añadir una limitación a la distribución geográfica explícita excluyendo estos países, de forma que esta distribución se permita solamente en o entre países no excluidos. En tal caso, esta Licencia incorpora la limitación como si estuviera escrita en el cuerpo de esta Licencia.

13.-

La Free Software Foundation (Fundación de Software libre) puede publicar versiones nuevas y/o revisadas de la Licencia Pública General Menor de vez en cuando. Tales versiones nuevas serán similares en espíritu a la versión presente, pero puede diferir en detalles para abordar nuevos problemas o intereses.

A cada versión se le da un número de versión que la distingue. Si la librería especifica un número de versión de esta Licencia que se aplica a ella misma y "cualquier versión posterior", usted tiene la opción de cumplir los términos y condiciones tanto de esa versión como de cualquier versión posterior publicada por la Free Software Foundation. Si la Librería no especifica un número de versión de licencia, usted puede elegir cualquier versión publicada por la Free Software Foundation

14.-

Si desea incorporar partes de la Librería en otros programas libres cuyas condiciones de distribución son incompatibles con estos, escriba al autor para pedirle permiso. Para el software cuyos derechos de autor pertenecen a la Free Software Foundation, escriba a la Free Software Foundation; nosotros hacemos a veces excepciones a esto. Nuestra decisión se guiará generalmente por los dos objetivos de preservar el estatus libre de todo lo derivado de nuestro software libre y de promover la compartición y reutilización de software.

SIN GARANTIA

15.-

DEBIDO A QUE LA LIBRERÍA ESTÁ LICENCIADA LIBRE DE CARGO, NO HAY GARANTÍA para LA LIBRERÍA EN LA EXTENSIÓN PERMITIDA POR LA LEY APLICABLE. EXCEPTO CUANDO SE ESTABLEZCA DE OTRO MODO POR ESCRITO, LOS POSEEDORES DEL DERECHO DE AUTOR Y/O OTRAS PARTES SUMINISTRAN LA LIBRERÍA "TAL CUAL" SIN GARANTÍA DE NINGUNA CLASE, YA SEA DE FORMA EXPRESA O IMPLÍCITA, INCLUYENDO, PERO NO LIMITADA A, LAS GARANTIAS IMPLICITAS DE COMERCIALIZACIÓN Y CONVENIENCIA para UN PROPOSITO PARTICULAR. TODO EL RIESGO EN CUANTO A CALIDAD Y EJECUCIÓN DE LA LIBRERÍA ES SUYO. SI LA LIBRERÍA RESULTARA ESTAR DEFECTUOSA, USTED ASUME EL COSTE DE TODO EL MANTENIMIENTO, REPARACIÓN O CORRECCIÓN NECESARIA.

16.-

BAJO NINGÚN CONCEPTO, A MENOS QUE SEA REQUERIDO POR LA LEY APLICABLE O DE ACUERDO A UN ESCRITO, CUALQUIER POSEEDOR DE DERECHOS DE AUTOR O CUALQUIER OTRA PARTE QUE PUEDA MODIFICAR Y/O REDISTRIBUIR LA LIBRERÍA COMO SE PERMITE ARRIBA, SERÁ RESPONSABLE CON USTED POR DAÑOS, INCLUYENDO CUALQUIER DAÑO GENERAL, ESPECIAL, ACCIDENTAL O CONSECUENTE ORIGINADO POR EL USO O INCAPACIDAD DE USAR LA LIBRERÍA (INCLUYENDO PERO NO LIMITANDO, A LAS PÉRDIDAS DE DATOS O A LA PRODUCCIÓN DE DATOS INCORRECTOS, O PÉRDIDAS SUFRIDAS POR USTED O TERCERAS PARTES O UN FALLO DE LA LIBRERÍA para FUNCIONAR CON CUALQUIER OTRO SOFTWARE) INCLUSO SI TAL TITULAR U OTRA PARTE HA SIDO NOTIFICADA DE LA POSIBILIDAD DE TALES DAÑOS.

*FIN DE TÉRMINOS Y CONDICIONES.*

## Cómo aplicar estos términos a sus nuevas librerías.

Si usted desarrolla una librería nueva, y quiere que sea del mayor uso posible para el público, le recomendamos que la haga software libre para que todo el mundo pueda redistribuirla o combinarla. Usted puede hacer tal cosa permitiendo la redistribución bajo estos términos (o, alternativamente, bajo los términos de la Licencia Pública General ordinaria).

Para aplicar estos términos agregue la siguiente nota a la librería. Es más prudente agregar los avisos al comienzo de cada fichero fuente para transmitir de una forma más efectiva la exclusión de garantía; y cada fichero debería tener al menos la línea de derechos de autor y un puntero hacia donde se encuentre la notificación completa.

Copyright ©

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public Licence as published by the Free Software Foundation; either version 2 of the Licence, or (at your opinion) any later version.

This library is distributed in the hope that it will be usefull, but WITHOUT ANY WARRANTY; without even the implied warranty of merchantability or fitness for a particular purpose. See the GNU Lesser General Public Licence for more details.

You should have received a copy of the GNU Lesser General Public Licence along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, Ma 02111-1307 USA.

Añada también información sobre como contactar con usted mediante correo electrónico y postal.

Debería incluir también su empleo (si trabaja como programador) o sus estudios, si tiene alguno, para firmar una "renuncia a los derechos de autor" para la librería, si fuese necesario. Aquí tenemos un ejemplo; altere los nombres:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

firma de Ty Coon, 1 April 1990

Ty Coon, President of Vice

¡Eso es todo!