

Guía *básica* de BIB_TE_X*

Daniel Camarena

Versión 1, 23 de marzo de 2023

Para sugerencias y correcciones: vcamarenap@uni.pe

Resumen

BIB_TE_X es un programa auxiliar de L^AT_EX, diseñado para facilitar el manejo de la bibliografía. Pero es más que eso: es también una herramienta que permite a L^AT_EX extraer datos de una base de datos e insertarlos adecuadamente formateados en un documento.

Este texto pretende abordar todos los aspectos de BIB_TE_X, desde los más básicos a los más complejos, haciendo además hincapié en su *españolización*, pues BIB_TE_X, como tantas otras herramientas, presupone que será usada para trabajar con documentos en inglés.

Índice

Preliminar	2
I BIB_TE_X básico	5
1. Dinámica general del sistema	5
1.1. Preparar nuestro documento principal para usar BIB _T E _X	5
1.2. Cómo generar la lista de referencias bibliográficas	8
2. Las bases de datos bibliográficas	9
2.1. Reglas generales sobre la escritura y codificación de los ficheros «.bib» .	9
2.2. Registros y campos bibliográficos	10
2.3. Otras cuestiones relativas a los ficheros «.bib»	23
2.4. Herramientas para manejar ficheros «.bib»	26

* Copyright © 2023. Daniel Camarena.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A Copy of the license is included in the section entitled “GNU Free Documentation License”.

Se otorga permiso para copiar, distribuir o modificar este documento en los términos de la Licencia GNU para Documentación Libre, versión 1.3 o cualquier versión posterior publicada por la Free Software Foundation; sin secciones invariantes, sin textos de la cubierta frontal y sin textos de la cubierta posterior. Una copia completa de la licencia (en inglés) se incluye en el apéndice titulado “GNU Free Documentation License”.

3. Los estilos bibliográficos	28
3.1. Descripción de los estilos estándar de BIB _T E _X	28
3.2. Estilos adicionales	30
3.3. Cómo españolizar los estilos estándar	31
 II BIB_TE_X intermedio	 32
4. Generación automatizada de estilos bibliográficos	32
4.1. Makebst	32
4.2. Uso de los ficheros «.dbj»	32
5. Paquetes L^AT_EX relacionados con BIB_TE_X	32
5.1. Referencias bibliográficas completas en el lugar donde son citadas	33
5.2. Generación de varias listas bibliográficas	33
5.3. Otros paquetes	33
6. Otras cuestiones de interés	33
6.1. El comando newblock y la opción openbib	33
6.2. Elementos @preamble en los ficheros «.bib»	33
6.3. Comentarios y registros desconocidos en los ficheros «.bib»	33
 III BIB_TE_X avanzado	 34
7. Los ficheros de estilo como programas	34
7.1. Cuestiones generales	34
7.2. Objetos del lenguaje BST	35
8. Conclusión: Otros usos de BIB_TE_X	35
Apéndice A. GNU Free Documentation License	36

Preliminar

BIB_TE_X es un programa que amplía notablemente las capacidades de L^AT_EX con el manejo de la bibliografía. Las grandes ventajas que ofrece su uso¹ se incrementan cuando se maneja una lista bibliográfica amplia y cuando con la misma bibliografía hay que escribir varios documentos.

No hay mucha documentación sobre BIB_TE_X en español. Publicadas en papel sólo conozco dos obras, ambas de los mismos autores. La primera, [Cascales Salinas, Lucas Saorín, Mira Ros, Pallarés Ruiz, y Sánchez-Pedreño Guillén \(2000\)](#), sin duda constituye la más completa información sobre BIB_TE_X en nuestro idioma: mas de 40 páginas en las que se incluyen incluso los aspectos más avanzados.

¹BIB_TE_X nos ayuda a gestionar nuestra bibliografía de manera independiente a los documentos en los que debe aparecer, al tiempo que nos permite generar listas de referencias bibliográficas formateadas de manera consistente y cambiar fácilmente determinadas características de las mismas.

La segunda, [Cascales Salinas, Lucas Saorín, Mira Ros, Pallarés Ruiz, y Sánchez-Pedreño Guillén \(2003\)](#), no es tan extensa, pero constituye una buena introducción en la que se cubren todos los aspectos básicos y algunos aspectos avanzados.

En formato electrónico y en nuestro idioma conozco otros dos documentos. Uno de Luis Seidel, [Seidel \(1998\)](#) que es una buena introducción, aunque no resulta muy extenso. Otro de José Manuel Mira, [Mira Ros \(2004\)](#), que no está pensado para quienes se aproximen por primera vez a esta herramienta, pues en él se explica un *paquete* («flexbib») cuyo uso presupone que ya conocemos lo básico del trabajo con $\text{BIB}\text{T}_{\text{E}}\text{X}$.

Como es lógico, en inglés hay mucha más documentación. Los textos principales son el apéndice B del Libro sobre $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ de Lamport, [Lamport \(1986\)](#), y los dos documentos que sobre $\text{BIB}\text{T}_{\text{E}}\text{X}$ escribió su autor, Oren Patashnik: [Patashnik \(1988a\)](#) y [Patashnik \(1988b\)](#). El primero se ocupa de los aspectos más básicos, y está concebido como una *corrección* del mencionado apéndice B, y el segundo se ocupa de los aspectos más avanzados. El problema que tienen estos documentos es que el trabajo de Lamport es difícil de localizar para el público español.

Además de los documentos anteriores, en Internet pueden localizarse varios textos, la mayoría en inglés, algunos introductorios y otros no tanto. De entre ellos me parecen destacables [Young \(2002\)](#) (un documento bastante equilibrado) y [Markkey \(2005\)](#), dirigido a un público más avanzado que los otros textos mencionados.

El objetivo fundamental de este documento es el de, partiendo de la documentación que se acaba de citar, salvo [Lamport \(1986\)](#) al que no he tenido acceso, y alguna otra que no se ha mencionado todavía, como [Kopka y Daly \(2004\)](#), poner al alcance del público hispanoparlante un documento en el que se aborden todos los aspectos de $\text{BIB}\text{T}_{\text{E}}\text{X}$, desde los más básicos a los más avanzados. De ahí el título que he elegido, que tal vez a alguien le suene a pretencioso. Desde luego esta guía no es completa, porque hay aspectos sobre los que se pasa casi de puntillas². Tal vez ni siquiera sea *casi* completa. Pero sí es la más completa que conozco³.

La materia a tratar ha sido distribuida en tres partes, atendiendo a su nivel de complejidad y detalle:

1. **$\text{BIB}\text{T}_{\text{E}}\text{X}$ básico:** Cubre los aspectos básicos del funcionamiento de $\text{BIB}\text{T}_{\text{E}}\text{X}$ que afectan a las instrucciones que hay que incluir en el documento principal de $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, al diseño y mantenimiento de la base de datos, y a los distintos

²Como por ejemplo las herramientas para gestionar ficheros «.bib», numerosos estilos bastante extendidos, o los paquetes de ampliación para trabajar con $\text{BIB}\text{T}_{\text{E}}\text{X}$.

³Quizás con la salvedad de [Cascales Salinas y cols. \(2000\)](#). Aunque el problema de ese libro es que hoy no es ya fácil de localizar.

estilos estándar existentes, así como a los procedimientos para *españolizarlos*. Es la parte más extensa del documento. Se supone que sólo con leer esta parte estaremos totalmente preparados para trabajar con BIB_TE_X.

2. **BIB_TE_X intermedio:** En esta parte se describen aspectos que no son imprescindibles para trabajar con BIB_TE_X, pero cuyo conocimiento nos proporcionará un mayor control. En particular se aborda el uso de makebst para generar nuestro propio fichero de estilo.
3. **BIB_TE_X avanzado:** Esta última parte entra a fondo en los ficheros de estilo de BIB_TE_X. En ella se describe el lenguaje que estos ficheros usan internamente.

Para entender este documento no hay que ser un La_TE_Xperto, pero sí hay que saber lo suficiente de L_AT_EX, y del procedimiento estándar que en él se implementa para trabajar con referencias bibliográficas.

En general el contenido de este documento valdrá para cualquier distribución de L_AT_EX que incluya BIB_TE_X (o sea, todas las distribuciones). Pero hay algunos aspectos que dependen de la concreta instalación que se haya hecho del sistema como, por ejemplo, la localización de los directorios en los que BIB_TE_X lee las bases de datos o los ficheros de estilo. En tales casos la explicación que doy no vale para cualquier sistema, sino exclusivamente para el mío, que es una instalación estándar de L_AT_EX hecha en un sistema donde funciona Ubuntu versión 6.06. Aunque, en general, estos aspectos dependientes del concreto sistema utilizado son tratados en nota a pie de página.

Asimismo, a lo largo del presente documento se citan en ocasiones determinados paquetes para L_AT_EX o ficheros con estilos adicionales para BIB_TE_X. Salvo que se diga lo contrario, todos los paquetes y estilos citados están disponibles en la CTAN, («*Comprehensive TeX Archives Network*») y deben buscarse en:

- «<http://www.ctan.org/tex-archive/macros/latex/contrib/>»
- «<http://www.ctan.org/tex-archive/biblio/bibtex/contrib/>»

Parte I

BIBTEX básico

1. Dinámica general del sistema

1.1. Preparar nuestro documento principal para usar BIBTEX

Las órdenes `bibliography` y `bibliographystyle`:

Para usar BIBTEX necesitamos dos cosas:

1. Tener almacenadas en un fichero aparte, las referencias bibliográficas que pensemos usar.
2. En el lugar del documento principal en el que queramos que aparezca la lista con las referencias bibliográficas, debemos insertar las siguientes dos órdenes de L^AT_EX:

```
\bibliography{MiBiblio}  
\bibliographystyle{MiEstilo}
```

Estas órdenes provocan que tras las oportunas compilaciones (véase la sección 1.2, página 8), se genere una lista con las referencias bibliográficas usadas en el documento en la que, además, a cada una de ellas se le asignará una *etiqueta* identificativa⁴. Los datos correspondientes a tales referencias serán buscados en el fichero «MiBiblio.bib», y la lista se formateará de acuerdo con las indicaciones de estilo que se contengan en el fichero de estilo «MiEstilo.bst» y se insertará en el lugar del documento en el que se encuentre “\bibliography”. Las instrucciones de estilo controlarán asimismo el tipo de etiqueta identificativa que se asignará a las obras incluidas en la lista de referencias. Si queremos usar más de una base de datos podemos indicarnos, separadas por comas, como argumento de “\bibliography”. El

⁴**NOTA TERMINOLÓGICA:** En la documentación en castellano de L^AT_EX el término *etiqueta* se suele usar para hacer referencia a la cadena de texto que usan en las órdenes que sirven para tratar con referencias cruzadas tales como “\cite”, “\label”, “\index” o “\glossary”. Este tipo de *etiquetas* son de uso puramente interno, es decir: L^AT_EX las usa como marcadores para apuntar a distintas partes del documento, pero no se imprimen en él. Junto a estas *etiquetas* también se usa a veces el mismo término para hacer referencia al rótulo identificador de una entrada concreta de la lista de referencias bibliográficas, que se imprime en el documento. Para evitar confusiones, en este documento he reservado el término *etiqueta* para las que llegan a imprimirse en el documento final, mientras que para referirme a las usadas internamente como marcadores, he reservado el término *clave*.

efecto de esto último será que las distintas bases de datos indicadas se concatenarán en el orden en el que se indicaron⁵.

Nombres de ficheros y directorios:

El argumento de “\bibliography” y el de “\bibliographystyle” es el nombre de un fichero. Ambos nombres deben indicarse sin especificar la extensión, pues se asume que esta será la asignada al tipo de fichero de que se trate. De hecho BIB_{TEX} no funciona correctamente cuando el nombre de la base de datos no tiene la extensión «.bib», ni tampoco cuando en dicho nombre existe un espacio en blanco o algún carácter no anglosajón como ñes o vocales acentuadas⁶.

Los ficheros, el de bibliografía y el de estilo, deben encontrarse en el directorio de trabajo (el mismo en el que se encuentre el documento principal), o en alguno de los directorios en los que L_AT_EX busca por defecto, los cuales varían dependiendo de la distribución de L_AT_EX concreta de que se disponga⁷. Si los ficheros que queremos usar no están en alguno de estos directorios deberemos incluir, junto con el nombre del fichero, la ruta de acceso al mismo, en cuyo caso para dicha ruta se aplican las mismas reglas que para el nombre de los ficheros: no podrá contener espacios en blanco ni caracteres no anglosajones.

Contenido de la lista de referencias a generar:

Las características *formales* y de ordenación de la lista bibliográfica que se generará, dependen del estilo concreto usado, y se explican en la sección 3. Aquí me concentraré exclusivamente en los siguientes dos aspectos de la misma:

⁵Lo que tiene importancia, porque en los ficheros «.bib» hay utilidades que sólo funcionan si el orden en el que ciertos elementos se encuentran dentro del fichero es el correcto. Por ejemplo, las abreviaturas deben estar definidas antes de ser usadas, de modo que si tenemos un fichero que recoja una serie de abreviaturas para BIB_{TEX} y una base de datos que las use, siempre habrá que indicar el fichero de abreviaturas antes que la base de datos.

⁶Así ocurre en mi sistema. Es posible que en otros sistemas operativos no exista este problema, aunque creo que ocurrirá lo mismo, al menos en lo relativo a la extensión de los ficheros; porque “\thebibliography” se limita a copiar en el fichero «.aux» el texto que recibe como argumento, y BIB_{TEX} le añade a dicho texto la extensión «.bib», salvo en el caso de que ya termine así.

⁷En «web2c», que es la distribución de L_AT_EX estándar para sistemas Unix/Linux, el árbol de ficheros de T_EX se encuentra en «/usr/share/texmf». Allí hay un fichero denominado «texmf.cnf» en el que se almacenan las variables que controlan las rutas de búsqueda por defecto de L_AT_EX y sus programas auxiliares. Aunque a veces dicho fichero se encuentra en el directorio «web2c», o en «/etc/texmf/». En dicho fichero se encuentran las dos variables que afectan a los directorios relativos a BIB_{TEX}: BIBINPUTS (directorios en los que se buscarán los ficheros «.bib») y BSTINPUTS (directorios en los que se buscarán los ficheros de estilo).

Qué referencias se incluyen en la lista: En ella se contendrán todas las referencias bibliográficas que, a lo largo de nuestro documento, se hayan referenciado mediante los comandos “\cite” y “\nocite”:

- El comando “\cite[DatosAdicionales]{clave}” produce un doble efecto. En primer lugar, la referencia bibliográfica identificada en la base de datos mediante la clave recibida como parámetro, se incluirá en la lista bibliográfica. En segundo lugar, en el punto del documento donde se encontrara el comando, se imprimirá la etiqueta asignada a tal referencia en la lista de referencias junto con los datos adicionales que eventualmente hayamos incluido en el argumento opcional del comando.
- El comando “\nocite{clave}” produce el primero de los efectos indicados, pero no el segundo, es decir: la obra identificada por la clave será incluida en la lista bibliográfica final, y en ella se le asignará asimismo una etiqueta (como a todas las obras de la lista), pero en el lugar del documento en el que se encuentra el comando “\nocite” no se imprimirá nada.

Ambos comandos pueden recibir, como parámetro, las claves de varias referencias distintas, separadas mediante comas. Y en el caso concreto de “\nocite”, si se escribe “\nocite{*}” el efecto será incluir en la lista bibliográfica final todas las referencias bibliográficas incluidas en la base de datos.

Qué título tendrá la lista de referencias: Dependiendo del tipo de documento de que se trate, el título de la lista de referencias está controlado por dos variables de nombre distinto⁸. En documentos tipo «book» para cambiar el título de la lista debe modificarse el comando “\bibname”, mientras que en documentos tipo «article» el comando \LaTeX que hay que modificar es “\refname”. El paquete «babel», por otra parte asigna a cada uno de estos comandos un valor dependiente del idioma. En español los nombres son “Bibliografía” para los libros y “Referencias” para los artículos.

Si, por ejemplo, deseamos que la lista de referencias se titule “Bibliografía citada” y nuestro documento es de tipo «book» tendremos incluir en el cuerpo de nuestro documento⁹ la siguiente orden:

```
\renewcommand{\bibname}{Bibliograf\`ia citada}
```

⁸Lo que, en mi opinión, es una inconsistencia del sistema. El paquete «chbibref» incluye un comando que permite cambiar el título de la lista de referencias con independencia de que se trate de un documento tipo «book» o de un documento tipo «article».

⁹Y es que, según mi experiencia, esta orden no produce efectos si se incluye en el preámbulo. Posiblemente por la interacción que la carga del paquete «babel» produce.

Asimismo, si se desea que en el índice de contenido de nuestro documento aparezca una entrada para la bibliografía, hay que incluirla a mano mediante la orden “\addcontentsline” de \LaTeX ¹⁰, la cual debe insertarse *antes* de “\bibliography”, ya que si se inserta detrás, en el índice del documento aparecerá como página de la bibliografía la página en la que esta termina, y no la página en la que empieza¹¹.

1.2. Cómo generar la lista de referencias bibliográficas

Una vez que nuestro documento (llamémosle «MiDoc.tex») está preparado, en los términos que se acaban de exponer, debemos seguir los siguientes pasos:

- 1ž Compilar con \LaTeX nuestro documento «.tex». Ello hará que se genere un fichero de extensión «.aux», en el que se incluirá información sobre la base de datos a usar, el fichero de estilo a usar, y las referencias bibliográficas que hay que incluir en la lista de referencias.
- 2ž Ejecutar, desde la línea de comandos¹², la orden «bibtex MiDoc». Ello hará que $\text{BIB}\text{\TeX}$ lea el fichero «.aux»¹³, generado por la anterior compilación del documento, extrayendo de él la información que necesita para trabajar: qué base de datos debe usar, qué estilo, y qué referencias hay que buscar en la base. Y así, tras extraer de la base de datos los registros precisos, y formatearlos de acuerdo con el estilo indicado, $\text{BIB}\text{\TeX}$ genera un fichero de extensión «.bbl» en el que se contienen los comandos de \LaTeX necesarios para escribir la lista de referencias bibliográficas que hay que insertar en el documento principal. $\text{BIB}\text{\TeX}$ genera también un fichero adicional, de extensión «.blg» que es un fichero «.log»¹⁴.
- 3ž Compilar de nuevo el documento principal con \LaTeX . En esta segunda compilación, al leer la orden “\bibliography”, se insertará en su lugar el contenido

¹⁰El paquete «tocbibind» hace que los índices y la lista de referencias se incluyan automáticamente en la tabla de contenido.

¹¹Véase más adelante en la sección 6.2, a propósito de los elementos «@preamble» de los ficheros «.bib», cómo podemos incluir en un fichero «.bib» el código necesario para cambiar automáticamente el título a la lista de referencias y asegurarnos de que se incluye en la tabla de contenido.

¹²Numerosas herramientas para \LaTeX incorporan la posibilidad de ejecutar desde el mismo editor tanto \LaTeX como cualquiera de sus programas auxiliares, en cuyo caso no sería preciso ejecutar $\text{BIB}\text{\TeX}$ desde la línea de comando.

¹³De hecho $\text{BIB}\text{\TeX}$ se ejecuta sobre el fichero «.aux», y no sobre el fichero «.tex». Este detalle puede escapárseos porque $\text{BIB}\text{\TeX}$ no exige que se escriba la extensión. Pero si en el comando incluimos la extensión comprobaremos que «bibtex MiDoc.tex» generaría un error, mientras que «bibtex MiDoc.aux» funcionaría correctamente.

¹⁴Es decir: en él se almacenan todas las salidas generadas por $\text{BIB}\text{\TeX}$, incluyendo los mensajes de advertencia o error. En caso de que algo no haya funcionado como esperábamos es imprescindible la consulta de este fichero para ver qué es lo que ha podido fallar.

del fichero «.bbl» generado en el paso anterior. Asimismo la nueva compilación reescribe el fichero «.aux», añadiendo a la información que ya existía en él la generada ahora, que es más completa pues incluye los datos de la lista bibliográfica final que se acaba de insertar en el documento.

- 4ž Esta última información es usada en una nueva compilación con \LaTeX para escribir correctamente los rótulos que hay que colocar en lugar de los comandos “\cite”. Y eventualmente puede ser necesaria una nueva compilación: cuando alguno de los campos de la base de datos contenga algún comando de \LaTeX que implique el uso de referencias cruzadas. En particular, el comando “\cite”.

¿Parece complicado? No lo es. Lo que ocurre es que lo he explicado incluyendo detalles sobre cómo interactúan entre sí los distintos ficheros generados durante las compilaciones de \LaTeX . Simplificando y sin entrar en tales detalles, lo que hay que hacer es, tras haber indicado en el documento principal el nombre de la base de datos y del estilo, realizar una primera compilación con \LaTeX , ejecutar \BibTeX y luego compilar de nuevo con \LaTeX dos veces (o, en ciertos casos, tres). Esta doble (o triple) compilación final no es ninguna especialidad de \BibTeX , sino que es requerida por \LaTeX siempre que trabaja con referencias cruzadas. Asimismo habrá que repetir el proceso cada vez que incorporemos a nuestro documento principal un comando que suponga una modificación de lista bibliográfica final, bien por cambiarle el estilo, bien por incluir o eliminar referencias bibliográficas.

2. Las bases de datos bibliográficas

2.1. Reglas generales sobre la escritura y codificación de los ficheros «.bib»

Los ficheros «.bib» usados por \BibTeX son ficheros de texto sometidos a las siguientes reglas generales:

1. El tratamiento de los espacios en blanco, tabuladores y saltos de línea es similar al que estos caracteres reciben en \LaTeX ¹⁵. Son simples delimitadores de palabras y la regla es que da igual cuántos delimitadores haya entre dos palabras¹⁶.

¹⁵Hay, no obstante, una diferencia, y es que en \LaTeX una línea en blanco es relevante pues sirve para indicar un cambio de párrafo. En los ficheros «.bib», una línea en blanco es tan irrelevante como un salto de línea simple.

¹⁶Podríamos, en teoría, escribir todo el fichero «.bib» como una sola e inmensa línea en la que las palabras estuvieran separadas entre sí por un solo espacio en blanco. Pero tal fichero sería, para los seres humanos, más difícil de leer que otro en el que hayamos usado los tabuladores, espacios y saltos de línea para mejorar el aspecto visual (llamado *legibilidad*) del fichero.

2. $\text{BIB}\text{T}_{\text{E}}\text{X}$ no distingue entre mayúsculas y minúsculas¹⁷.

Uso de caracteres españoles en los ficheros «.bib»

La versión original de $\text{BIB}\text{T}_{\text{E}}\text{X}$ no era capaz de trabajar con caracteres no anglosajones (como ñes o vocales acentuadas). En la actualidad existe una versión de $\text{BIB}\text{T}_{\text{E}}\text{X}$ que sí puede hacerlo. No obstante se suele recomendar que para representar esos caracteres que requieren 8 bits se sigan las reglas generales de $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ y así, por ejemplo, en lugar de escribir «Pág» escribamos “P\’ag”¹⁸. Y aunque es cierto que escribir de ese modo nuestro idioma es incómodo, hay herramientas que lo facilitan¹⁹.

En todo caso, si decidimos introducir caracteres no anglosajones directamente en el fichero «.bib», debemos tener en cuenta que como el contenido de este fichero puede eventualmente incorporarse a un documento «.tex», si en ambos el tipo de codificación no coincidiera se generaría un error. Por ello, para usar estos caracteres deben cumplirse los siguientes requisitos:

- a) Debemos guardar el fichero «.bib» con la misma codificación con la que vayamos a guardar el fichero «.tex» en el que se hará uso de la base de datos.
- b) En el fichero «.tex» debe usarse la orden “\usepackage[cod]{inputenc}” donde “cod” significa una cadena representativa de la codificación usada en el fichero «.bib» (y en el «.tex»): «utf8», «latin1», etc.
- c) Debemos asegurarnos de que estamos usando la versión de $\text{BIB}\text{T}_{\text{E}}\text{X}$ de 8 bits.

2.2. Registros y campos bibliográficos

Un fichero «.bib» almacena una base de datos bibliográfica. En toda base de datos hay dos nociones fundamentales:

¹⁷La no distinción entre mayúsculas y minúsculas se aplica a los elementos de $\text{BIB}\text{T}_{\text{E}}\text{X}$ (nombres de registros o campos, abreviaturas, etc), pero no al contenido de los campos de los registros.

¹⁸El problema de representar caracteres no anglosajones directamente en un fichero de texto, es que estos caracteres no están incluidos en la tabla ASCII que todos los ordenadores reconocen, sino en otras tablas o *codificaciones*. Y, desgraciadamente para los hablantes de idiomas distintos al inglés, existen *muchas* codificaciones distintas para representar los mismos caracteres: “utf8”, “latin1”, “cp850”, etc.; y para trabajar con un documento que contenga este tipo de caracteres es preciso saber exactamente qué codificación se usó para almacenarlo pero, desgraciadamente, los ficheros de texto no guardan información sobre su propia codificación. Además, en el caso concreto de los ficheros «.bib», el uso de una codificación u otra puede afectar a la ordenación alfabética de los registros almacenados en la base de datos.

¹⁹Como, por ejemplo «pybliographer», una utilidad de gestión de ficheros «.bib» para sistemas Unix/Linux, en la que podemos teclear normalmente cualquier carácter, pero antes de guardar el fichero, los caracteres no anglosajones son convertidos a la secuencia $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ necesaria para representarlos. De esta herramienta se vuelve a hablar en la sección 2.4.1.

Campo: Un dato aislado y básico. Por ejemplo, el año de edición de un libro, o el nombre del autor de un artículo de revista, o la página del libro de las actas de un congreso en donde empieza una determinada ponencia...

Registro: Un conjunto de campos que permiten describir de forma completa un ejemplar del tipo de realidades a que se refiere la base de datos. En nuestro caso, tratándose de una base de datos bibliográfica, cada registro describe una referencia bibliográfica concreta (y completa).

Es decir: todos los campos que contienen datos de una misma referencia bibliográfica, constituirán un registro. Y la base de datos, en sí misma considerada, no será sino un conjunto de registros, cada uno de los cuales consta de varios campos.

2.2.1. Formato general de los registros y los campos

En los ficheros «.bib» se usa el carácter “@”, seguido de una palabra representativa del tipo de registro de que se trate, para indicar que empieza un registro. Asimismo se usan llaves para delimitar el contenido del registro. En consecuencia el formato de un registro es el siguiente:

```
@TipoRegistro{clave,  
  Campo1,  
  Campo2,  
  ...  
  CampoN,  
}
```

Donde *TipoRegistro* es el nombre que identifica a un concreto tipo de registro, y puede escribirse indistintamente en mayúsculas o minúsculas. En el contenido del registro, hay básicamente dos elementos:

La clave de identificación del registro es siempre su primer elemento. Se usa para distinguir a un registro concreto del resto de los registros de la base de datos, y por ello en una misma base de datos no debe haber dos registros que tengan claves iguales, y tampoco deben unirse o manejarse en un mismo documento dos o más bases de datos si ello implica que alguna clave vaya a repetirse²⁰. La clave puede consistir en cualquier combinación de letras, números y ciertos signos de puntuación. No se admiten ni los caracteres no

²⁰Digo que no *deben* usarse claves iguales; no que no pueda hacerse. Ello es porque BIBTEX no genera ningún error cuando en un fichero dos o más registros tienen la misma clave. Simplemente, en tal caso, toma en consideración exclusivamente la primera aparición de la clave, ignorando las restantes. Y, a tales efectos, téngase en cuenta que para el control de las claves BIBTEX no distingue entre mayúsculas y minúsculas.

anglosajones (vocales acentuadas y los caracteres “`añÑçÇ`”) ni ciertos signos de puntuación como la coma (que se usa para indicar que la clave ha terminado).

En las citas hechas en el documento «`.tex`» habrá que usar esta clave para identificar al registro al que se quiere hacer referencia.

Los campos: A `BIBTEX` le es indiferente en qué orden se escriban los campos de un determinado registro, así como si se usan cero, uno o más saltos de línea entre dos campos.

Para escribir un campo dentro de un registro podemos usar indistintamente cualquiera de los dos formatos siguientes:

```
NombreCampo = { Contenido },  
NombreCampo = " Contenido ",
```

En ambos casos intervienen los siguientes elementos:

Nombre del campo: Normalmente será uno de los campos, obligatorios u opcionales, previstos para el tipo de registro de que se trate. Si el nombre no coincide con ninguno de los que el estilo ha previsto para el tipo de registro que sea, su contenido será ignorado. Si el mismo campo se introduce más de una vez en el mismo registro, sólo será tomada en consideración su primera aparición.

Delimitadores de contenido: Se exige el uso de delimitadores para el contenido del campo siempre que este no tenga un valor puramente numérico o consista en una abreviatura previamente definida como tal mediante la función «`@String`» (véase la sección 2.3.1). Como delimitadores se pueden usar las llaves o las comillas dobles. La diferencia entre usar unas u otras es sólo una: cuando usamos comillas dobles como delimitadores, dentro de la cadena delimitada no podemos usar comillas dobles, salvo que las encerremos entre llaves. Por ejemplo:

```
title = {Comentarios a "El Buscón" de Quevedo},  
title = "Comentarios a {}El Buscón{} de Quevedo",
```

servirían para incluir un título que a su vez incluya dobles comillas.

El contenido de los campos: cualquier cosa que se incluya en un campo puede, eventualmente, incorporarse a un documento `LATEX`, y por lo tanto:

1. Es posible incluir, como contenido de un campo, comandos de `LATEX`, aunque ello sólo está recomendado en casos muy especiales, y nunca es una buena idea que esos comandos sean de *formato* de texto, pues la idea que está detrás de `BIBTEX` es que el formateo de las listas de referencias se haga depender del estilo usado.

2. Dentro del contenido del campo hay que respetar las reglas generales de \LaTeX , no usar sus caracteres reservados (salvo cuando se trate de un uso legal) y recordar que \LaTeX , a diferencia de \BibTeX , sí distingue entre mayúsculas y minúsculas.
3. Las llaves que pueda haber en un campo deben estar equilibradas. Es decir: toda llave abierta debe ser cerrada. Y ello se aplica incluso para llaves que se deben imprimir²¹.

Existen reglas especiales para el contenido de algunos campos concretos, pero estas se exponen más adelante, cuando se explican los campos reconocidos por los estilos estándar de \BibTeX .

La coma final: Indica que el contenido del campo ha terminado y *puede* empezar otro campo distinto. No es obligatoria en el último campo de un registro; pero no se produce ningún error por el hecho de introducirla también en él.

Desde el punto de vista de un concreto estilo bibliográfico, los campos presentes en un registro se incluirán en uno de los siguientes grupos:

Campos obligatorios: Son aquellos que el estilo en cuestión espera encontrar en un determinado tipo de registro, de manera que si en un registro concreto alguno de ellos no está presente, se generará una advertencia en la salida estándar y en el fichero «.blg» generado por \BibTeX . En algunas ocasiones la falta de uno de estos campos puede provocar un error, aunque eso no debería ocurrir en un estilo bien diseñado.

Campos opcionales: Son aquellos para los que el estilo en cuestión ha previsto algún tipo de acción, pero se ha previsto también su posible ausencia, y esta no tiene especial trascendencia.

Campos ignorados: Cualquier otro campo presente para el que el estilo en cuestión no ha previsto ningún tipo de acción.

2.2.2. Tipos de registros bibliográficos

Las referencias bibliográficas pueden ser de distinta naturaleza, y en consecuencia se distinguen diferentes tipos de registros. A continuación expondré los nombres de los distintos tipos de registro previstos en \BibTeX , indicando para cada uno de ellos sus campos obligatorios y opcionales específicos²². Para que

²¹Lo que se debe a que \BibTeX no interpreta los comandos de \LaTeX , por lo tanto si en un campo se lee el texto «\{», \BibTeX no ve un carácter imprimible, sino una llave abierta que debe ser cerrada dentro del campo. Por ello en el raro caso de que debamos incluir como contenido de un campo el carácter “{“ (o “)”) habría que usar el comando \LaTeX “\leftbrace” (o “\rightbrace”).

²²Con *específicos* quiero decir que, para evitar repeticiones, en la relación que sigue no se ha incluido en cada uno de los registros los nombres de dos campos opcionales que están presentes en todos los registros, sean del tipo que sean: «key» y «note».

BIB_TE_X pueda reconocer adecuadamente los registros y campos, los nombres de estos deben ser usados en inglés, y por ello los mencionaré en ese idioma. Entre paréntesis añadiré, para los tipos de registro, lo que significa su nombre. En la próxima sección se explica para qué sirve cada campo.

Article (artículo): Un artículo publicado en una revista. Campos obligatorios: `author`, `title`, `journal` y `year`. Opcionales: `volume`, `number`, `pages` y `month`.

Book (libro): Un libro *normal*. Campos obligatorios: `author` o `editor`, `title`, `publisher` y `year`. Opcionales: `volume` o `number`, `series`, `address`, `edition` y `month`.

Booklet (folleto): Un trabajo impreso y distribuido pero sin que conste la editorial o institución que lo patrocina. Campo obligatorio: `title`. Campos opcionales: `author`, `howpublished`, `address`, `month` y `year`.

Conference (conferencia): Idéntico a `InProceedings`. Se incluye exclusivamente para mantener la compatibilidad con el formato *Scribe*.

InBook (dentro de un libro): Una parte de un libro, que puede ser un capítulo (o sección o similar) o un rango de páginas, o ambas cosas. Campos obligatorios: `author` o `editor`, `title`, `chapter` y/o `pages`, `publisher` y `year`. Opcionales: `volume` o `number`, `series`, `type`, `address`, `edition` y `month`. El campo `title`, en estas referencias, se refiere al título del libro, no al título del capítulo o grupo de páginas a que se refiere el registro.

InCollection (en una colección): Una parte de un libro que tiene su propio título. Campos obligatorios: `author`, `title`, `booktitle`, `publisher` y `year`. Opcionales: `crossref`, `editor`, `volume` o `number`, `series`, `type`, `chapter`, `pages`, `address`, `edition` y `month`.

InProceedings (en las actas): Una conferencia, artículo o ponencia en las actas de un congreso o, o, en general, en un libro que agrupe varios trabajos de autores distintos y con títulos independientes. Campos obligatorios: `author`, `title`, `booktitle`, `year`. Campo opcionales: `crossref`, `editor`, `volume` o `number`, `series`, `pages`, `address`, `month`, `organization` y `publisher`.

Manual: Documentación técnica. Campo obligatorio: `title`. Campos opcionales: `author`, `organization`, `address`, `edition`, `month` y `year`.

MasterThesis (Proyecto fin de carrera): Lo que en el mundo académico norteamericano se denomina “*Master Thesis*” o “*Minor Thesis*” y que en España equivale a las llamadas “*Tesinas de licenciatura*” y a los *Proyectos de fin de carrera*; es decir: un trabajo de investigación menor, leído en una institución académica. Campos obligatorios: `author`, `title`, `school` y `year`. Opcionales: `type`, `address` y `month`.

Misc (miscelánea): Este tipo está previsto para ser usado cuando ninguno de los otros tipos encaje bien. Su peculiaridad es que carece de campos obligatorios. Campos opcionales: `author`, `title`, `howpublished`, `month` y `year`.

PhdThesis (tesis doctoral): Lo que en el mundo académico norteamericano se denomina “*PhD Thesis*”, “*Major Thesis*” y que equivale a las tesis doctorales españolas, es decir: un trabajo de investigación original y extenso que permite obtener el título de doctor que constituye la más alta graduación académica existente. Campos obligatorios: `author`, `title`, `school` y `year`. Opcionales: `type`, `address` y `month`.

Proceedings (libro de actas): El libro de actas donde se encuentra una conferencia o ponencia en un congreso, aunque yo lo uso con carácter general para todos aquellos libros que agrupan trabajos, cada uno de ellos con distintos autores. Campos obligatorios: `title` y `year`. Opcionales: `booktitle`, `editor`, `volume` o `number`, `series`, `address`, `month`, `organization` y `publisher`.

TechReport (informe técnico): Un informe publicado por un centro académico o institución similar, normalmente numerado dentro de una serie. Campos obligatorios: `author`, `title`, `institution` y `year`. Campos opcionales: `type`, `number`, `address` y `month`.

Unpublished (no publicado): Un documento que tiene un autor y un título pero que formalmente no ha sido publicado. Campos obligatorios: `author`, `title` y `note`. Opcionales: `month` y `year`.

A los campos mencionados hay que añadir, en todos los tipos de registro, un campo `key` y un campo `note`, ambos opcionales, salvo en las referencias tipo «Unpublished», donde `note` es obligatorio.

2.2.3. Significado de los distintos campos bibliográficos

Address: Este campo está pensado para almacenar la dirección de la editorial o institución responsable de una publicación. Es siempre opcional y conviene usarlo, sobre todo, cuando las editoriales sean pequeñas o poco conocidas. En la mayor parte de los casos, además, no se incluye aquí la dirección completa, sino exclusivamente la ciudad donde reside la editorial. Hay especialidades científicas donde es costumbre identificar las obras publicadas más bien por la ciudad que por la editorial, en cuyo caso puede ser buena idea indicar la ciudad no en el campo «`address`», sino en «`publisher`».

Author: Como este es el campo más complejo, dividiré su explicación en varios apartados:

La autoría de las referencias: Toda referencia bibliográfica ha de tener una *autoría* y en tal sentido, en todos los tipos de registros hay algún campo dirigido a recogerla. Ese campo es normalmente «author», aunque:

- a) En «Proceedings» la autoría viene recogida por el campo «editor», porque este tipo de registro recoge un libro sin autor propiamente dicho, que recopila o agrupa varios trabajos que sí tienen autor.
- b) En los tipos «Book» e «InBook», el campo «author» puede ser sustituido por «editor». Si se especifican ambos campos simultáneamente, el contenido del campo «editor» es ignorado.
- c) En las referencias de tipo «Manual», a falta de campo «author», la autoría se atribuirá al campo «organization».

En cualquiera de estos casos, la recomendación es que el nombre del autor se escriba tan completo como se conozca, es decir: que no se usen iniciales para sustituir al nombre propio, ya que ese efecto podemos obtenerlo mediante el estilo bibliográfico. Pero si en la base de datos no consta el nombre completo no hay estilo bibliográfico que sea capaz de adivinarlo.

Varios autores: Si hay más de un autor, para separar unos de otros hay que usar necesariamente la palabra “and” para que BIBTEX sepa donde empieza un autor y donde acaba otro. Si usamos la conjunción española equivalente, “y”, BIBTEX no formateará correctamente el contenido de este campo porque no sabrá separar a los distintos autores. El hecho de usar “and” dentro del fichero «.bib» no significa que en nuestros documentos se vaya a usar dicha palabra, pues eso depende del fichero de estilo.

Si *dentro* del nombre de uno de los autores aparece la palabra “and”, hay que encerrar entre llaves (distintas de las genéricas llaves para delimitar el contenido del campo) el nombre completo. Así, en los siguientes ejemplos:

```
author = "{McArthur And Inc.}",
author = {{McArthur And Inc.}},
author = {Thomas R. Delan And {McArthur And Inc.}},
```

El nombre que contiene, como parte de él, la palabra “And” va siempre entre llaves, con independencia de qué tipo de delimitador se haya escogido para el campo y de si hay o no mas nombres.

Distinción entre nombre y apellidos: El nombre de una persona tiene varias partes de las que las principales (en los nombres españoles) son el nombre propio (también llamado nombre de pila, o nombre a secas) y los apellidos. BIBTEX descompone los nombres para determinar qué parte de

los mismos hay que atribuir al nombre propio, cuál a los apellidos, y cual a las restantes partes que BIBTEX reconoce en un nombre²³. Para ello se siguen las siguientes reglas:

11. Si el texto introducido consta de una sola palabra, se asignará al apellido.
21. Si consta de dos o más palabras, se considerará que la última es el apellido y el resto es el nombre de pila. Esta regla se basa en que fuera de la Península Ibérica (y países de tradición cultural ibérica), es usual tener un solo apellido y varios nombres de pila.
31. En el caso de que el apellido conste de varias palabras y alguna de las palabras interiores estuviera escrita con minúsculas, se considerará que tal palabra es una partícula que separa el nombre del apellido por lo que se asignará al apellido todo lo que esté detrás de tal partícula, y al nombre lo que esté antes. Y así, por ejemplo, BIBTEX interpretará correctamente el nombre

Miguel de Cervantes Saavedra

siempre y cuando escribamos el “de” y lo hagamos en minúsculas. Pero si escribimos:

Miguel Cervantes Saavedra

Miguel De Cervantes Saavedra

miguel de cervantes saavedra

como BIBTEX no podrá identificar la partícula aplicará la regla 2, y considerará que “Saavedra” es el apellido y el resto nombre propio²⁴.

41. Si solo conocemos el apellido, y este consta de varias palabras, para evitar que las primeras se asignen al nombre de pila, hay que encerrar todo el nombre entre llaves (además de las delimitadoras del campo). Por ejemplo:

author = "{Ortega y Gasset}",

51. Si en un nombre se usa una coma, BIBTEX asignará al apellido todo lo que esté antes que la coma, y al nombre propio todo lo que esté detrás. Esta regla prevalece sobre todas las anteriores.

²³En realidad BIBTEX descompone los nombres en cuatro partes: Nombre de pila (en inglés, *First name*), Partícula de separación, Apellidos (en inglés *Last name* o *Surname* y partícula “Jr.”, muy habitual en los nombres anglosajones. En las líneas que siguen he omitido lo referente a esta última partícula, inexistente fuera del mundo anglosajón. No obstante si en nuestra base de datos debe introducirse algún nombre propio que incluya dicha partícula, normalmente esta será reconocida por BIBTEX y su presencia no alterará las reglas que a continuación se exponen.

²⁴Por ello si queremos que en nuestra lista bibliográfica las partículas se impriman en mayúsculas, pero que BIBTEX las reconozca correctamente, habría que escribir: “Miguel {\uppercase{d}e} Cervantes Saavedra”.

Para nombres españoles, con dos apellidos, lo más seguro es usar una coma entre el apellido y el nombre para impedir que `BIBTEX` realice malas interpretaciones. Porque aunque en ocasiones la partícula nos puede ayudar (como en “Miguel de Cervantes Saavedra”) la mayoría de los nombres carecen de partícula, y en muchas ocasiones la partícula produciría un error, como en “Félix Lope de Vega y Carpio”.

Booktitle: Es el título de un libro, parte del cual está siendo citado. Se usa, por lo tanto en las referencias que sirven para citar una parte de un libro que tenga un título distinto del título del libro, es decir: «InCollection» e «InProceedings». Este campo también está presente en las referencias de tipo «Proceedings», aunque en ellas se usa exclusivamente para las referencias cruzadas (véase la sección 2.3.2). A este campo le son de aplicación las reglas que más adelante se exponen sobre el campo «title».

Chapter: Contiene un número de capítulo, sección o unidad estructural de un libro. Se usa en las referencias de tipo InBook e InCollection para identificar una parte de un libro.

Crossref: Contiene una etiqueta en la base de datos para generar referencias cruzadas internas (véase la sección 2.3.2).

Edition: Se trata del número de edición de un libro. Es siempre un campo opcional; y de hecho sólo suele rellenarse cuando no se trata de la primera edición. Los estilos estándar presuponen que se escribirá con una palabra en mayúsculas (y en inglés). Por ejemplo “Second”, aunque si se escribe en español (“Segunda”), o con un ordinal expresado en forma numérica (“2ª”), no pasa nada²⁵. Lo que sí es importante, para generar listas bibliográficas consistentes, es que siempre lo escribamos del mismo modo.

Editor: Este campo ya ha sido mencionado a propósito del campo “author”. El editor de un libro es el que, sin ser su autor propiamente dicho, lo impulsa y hace nacer. Esta figura es importante hasta el punto de que sobre él recae la *autoría* (a efectos de la base de datos) en determinado tipo de obras como puede ser una antología, un estudio crítico de alguna obra antigua, etc. Por ejemplo: una edición crítica de la poesía barroca española, o el conocido “*Las mil mejores poesías de la lengua castellana*”. Las reglas relativas a este campo son similares a las del campo “author”.

Howpublished: Este campo se usa en las referencias de tipo “Booklet” y “Misc” que recogen libros que no han sido publicados por una editorial o institución “normal”. En este campo puede indicarse cómo se llegó a publicar la obra de que se trate. La primera palabra debe ir en mayúsculas.

²⁵En los estilos estándar detrás de este campo se añade la palabra “edition” y, si están españolizados, la palabra “edición”, por ello no conviene usar directamente un número que no sea un ordinal, ya que entonces la referencia contendría la expresión “2 edición”, que no es correcta.

Institution: Este campo existe exclusivamente en los informes técnicos (referencias «TechReport») y recoge el nombre de la institución que ha financiado el informe de que se trate.

Journal: Para artículos publicados en revistas este campo recoge el nombre de la revista. En determinados campos del conocimiento existen abreviaturas estándar para las revistas más conocidas, por lo que puede usarse la abreviatura en lugar del nombre completo de la revista. Sobre el uso de abreviaturas en los ficheros «.bib» véase la sección 2.3.1.

Key: Está disponible como campo opcional en todos los tipos de referencias. Se usa en relación con la generación de etiquetas, en aquellos estilos en los que estas se generan a partir del nombre del autor, como por ejemplo ocurre en el estilo «alpha». En tales casos, el valor del campo «key» será usado para la generación de la etiqueta si no consta ningún valor para el campo «author», incluso aunque si conste algún valor para alguno de los campos que en ciertos tipos de referencias pueden suplir a dicho campo (véase lo que, al explicar el campo «author» se dijo sobre la autoría, en pág. 16).

Month: En artículos de revista se usa para designar el mes en el que salió el número que contiene dicho artículo. En otro tipo de trabajos el mes en el que fueron publicados o terminados (para trabajos no publicados). La documentación oficial de BIB_TE_X recomienda introducir este campo mediante unas abreviaturas estándar en inglés²⁶, en cuyo caso usando la abreviatura, no habría que usar delimitadores para el campo (por las razones que se explican cuando se habla de las abreviaturas en la sección 2.3.1).

En teoría aquí ocurre igual que en el campo «author», respecto de la partícula “and” usada para separar entre sí a los distintos autores, y es que aunque hayamos introducido el mes en inglés (o mediante una abreviatura del nombre inglés), luego, usando el estilo bibliográfico correcto, en las referencias que se inserten en nuestros documentos aparecerá el nombre en español. Aunque, a diferencia de lo que ocurre con el campo «author», si aquí introducimos la información directamente en español, no se produce ningún error.

Note: Cualquier tipo de información adicional que no tenga cabida en el resto de los campos²⁷. En los estilos bibliográficos estándar el contenido de este

²⁶Se trata de: «jan», «feb», «mar», «apr», «may», «jun», «jul», «aug», «sep», «oct», «nov» y «dec».

²⁷Yo lo he usado, por ejemplo, para recoger el nombre de los traductores (en traducciones), o en ediciones que han sido actualizadas por personas distintas del autor original, para recoger el nombre de estos últimos, o en documentación electrónica, para recoger la dirección de Internet donde se encuentra. Aunque para esto último es preferible usar un campo adicional de nombre «url», pues existen varios estilos no estándar para BIB_TE_X que prevén la existencia de este campo y son capaces

campo se transcribe al final de la referencia y su primera palabra debería empezar en mayúsculas.

Number: Puede tener distintos significados. En los informes técnicos recoge el número de informe de que se trate. En artículos de revista se usa en revistas cuyos distintos ejemplares se identifican mediante un *número*, o mediante un número y un volumen. En libros sólo se usa si el libro forma parte de una serie o colección. En los registros de tipo «Book» y «Proceedings» (así como en sus derivados: «InBook», «InCollection» e «InProceedings») el contenido de este campo sólo es tomado en consideración si el campo «volume» se ha dejado en blanco.

Organization: La organización que financia una conferencia o que publica un manual. En los registros de tipo «Manual» este campo designa la *autoría* si «author» carece de contenido.

Pages: Se usa en artículos y en partes de libro para designar las páginas concretas en las que un trabajo se encuentra. Los estilos estándar de BIB_TE_X esperan que se pongan aquí las páginas inicial y final, separadas por uno o dos guiones. De hecho estos estilos convierten el guión sencillo en uno doble, que es el utilizado en L^AT_EX para designar intervalos de páginas. Si se quiere indicar sólo la primera página, se utiliza el carácter “+” que significa “desde esa página en adelante”. Como los estilos estándar suponen que se trata de más de una página, en la lista bibliográfica final, a veces al número aquí puesto se le añade la palabra “páginas” (*pages*), o la abreviatura “págs.” (pp) en plural.

Publisher: La editorial encargada de la publicación de un libro. Su contenido suele complementarse con el del campo «address».

School: Nombre de la Escuela, Facultad o Instituto en donde se ha confeccionado un trabajo académico (tesis doctoral o tesina de licenciatura).

Series: En libros que forman parte de una colección, se usa para recoger el nombre de la misma.

Title: Es un campo presente en todos los tipos de referencias bibliográficas, y es siempre obligatorio (salvo en las referencias de tipo «Misc»). Designa el título del trabajo al que nos estamos refiriendo, salvo en las referencias tipo «InBook», donde designa el título del libro del que forma parte el capítulo o rango de páginas al que nos estamos refiriendo.

En teoría los estilos bibliográficos estándar *normalizan* (o pueden normalizar) el uso de las mayúsculas y minúsculas en los títulos. Y por ello en

de formatearlo correctamente. Incluso el paquete makebst es capaz de generar automáticamente un estilo bibliográfico que reconozca este campo. Véase la sección 4.1.

las referencias generadas a partir de nuestra base de datos no siempre se escribirán los títulos tal y como aparecen en la base de datos. Para preservar una letra en mayúsculas hay que encerrarla entre llaves. Las llaves pueden ponerse también alrededor de una o varias palabras. Esto es especialmente importante si en el título se usan comandos de \LaTeX , ya que en ellos una alteración de las mayúsculas y minúsculas se traduciría en un error.

Type: Este campo existe en «InBook», «InCollection»²⁸, «MasterThesis», «PhdThesis» y «TechReport». En los tres últimos tipos mencionados los estilos estándar añaden a la referencia, como texto aclaratorio, el nombre del registro. Por ejemplo: en una tesis doctoral, se escribe: “*Tesis doctoral*”. Si en el campo «type» hay algún texto, se usará este texto en lugar del predeterminado. En los registros «InBook», aunque por defecto no se escribe ningún texto, sí se escribirá lo que se diga en este campo. Por lo tanto aquí puede escribirse una pequeña aclaración respecto del tipo de trabajo de que se trata, como: “Conferencia”, “Disertación”, etc.

Volume: En artículos de revista se usa para indicar en qué tomo de la revista se ha publicado dicho artículo. En libros que formen parte de una serie, para indicar el tomo o volumen, dentro de la serie en cuestión. En este último caso si se da algún valor al campo «volume» los estilos estándar ignorarán el contenido del campo «number».

Year: El año de la edición del libro o de publicación del artículo. Es otro de los datos presentes (de forma obligatoria u opcional) en todos los tipos de registros. Hay incluso estilos bibliográficos en donde las referencias se ordenan por el valor de este campo, y las citas en el texto lo incluyen también.

2.2.4. Algunos consejos sobre tipos de registros y campos

Cuando empezamos a manejar $\text{BIB}\TeX$ la gran cantidad de tipos de registros existentes (catorce en total) puede abrumar. A continuación expongo algunos consejos, dictados por la experiencia, sobre cómo debemos proceder.

1. Hay tres tipos de referencia básica: artículos de revista («article»), libros completos («book») y partes de libro. Para las partes de libro a su vez hay tres posibilidades: Si la parte se identifica por un número de sección o de páginas, la referencia correcta es «InBook», si se identifica por un título, yo distingo, a su vez, según cada parte del libro tenga un autor distinto, en cuyo caso utilizo «InProceedings», o todo el libro pertenezca al mismo autor, en cuyo caso prefiero usar «InCollection». Ello es porque no me tomo

²⁸La documentación oficial de $\text{BIB}\TeX$ y los textos que he leído coinciden en que en los registros «InCollection» el campo «type» es opcional. Las pruebas que yo he hecho me llevan a pensar que en estos registros este campo es ignorado.

demasiado en serio el nombre del tipo de referencia, y atiendo más bien al contenido de los campos²⁹.

2. Tampoco hay que tomarse demasiado en serio el nombre de los campos. En ocasiones puede ser conveniente, por ejemplo, incluir la ciudad de edición junto con el campo reservado a la editorial, y en otros casos puede ser preferible usar para ello el campo «address». Lo importante es que seamos consistentes en nuestras decisiones.
3. La distinción entre campos obligatorios y opcionales es importante sólo para confirmarnos que hemos elegido el tipo de registro adecuado. Pero tampoco es una tragedia que en un caso concreto, algún campo obligatorio se quede sin rellenar.
4. Para la información que consideremos relevante pero para la que no haya ningún campo que la haya previsto, podemos usar el campo «note». Pero también podemos usar campos específicos creados por nosotros. En tal caso los estilos estándar de `BIBTEX` la ignorarán, pero podremos crear un estilo bibliográfico propio que la tenga en cuenta. Yo, por ejemplo, utilizo muchas traducciones, y estimo que el nombre del traductor es un dato importante para la referencia. Empecé usando para ello el campo «note», pero al final diseñé un campo opcional llamado «traductor». Sobre la escritura de estilos bibliográficos propios, véase la última parte de este documento. Aunque cuando decidamos crear un campo adicional, si el tipo de datos que queremos incluir en él es relativamente corriente, puede merecer la pena que antes de bautizar el campo, nos demos un paseo por Internet; porque es muy posible que alguien haya ya sentido la necesidad de recoger tales datos, haya creado ese campo y, lo que es más importante, haya escrito un estilo bibliográfico que sepa qué hacer con él.
5. También es interesante la creación de campos adicionales para información que es importante para nosotros tener recopilada, pero que no importa a los lectores de nuestros documentos y no tiene por qué incluirse en las referencias bibliográficas generadas. Por ejemplo yo uso para los libros un campo llamado «tejuelo» en el que almaceno la “etiqueta” o “tejuelo” asignada a determinado libro en la biblioteca de mi centro de trabajo, lo que me ayuda a localizar físicamente el libro con rapidez.
6. Por último, aunque he señalado que no hay que tomarse demasiado en serio ni los nombres de los tipos de registro, ni los de los campos, ni su naturaleza obligatoria u opcional, y he añadido que puede ser interesante crear campos propios, todo ello no debe llevarnos a diseñar y escribir una base de datos que sólo nosotros seamos capaces de entender y utilizar. El sentido último

²⁹Y no hay que escandilarse por ello. El propio autor de `BIBTEX` recomienda en [Patashnik \(1988a\)](#) no tomarse excesivamente en serio los nombres de registros y campos.

de las herramientas de L^AT_EX (y en general del software libre) es el de *compartir* el trabajo: Que lo que ya ha sido hecho no tenga por qué repetirse (si está bien hecho y el autor autoriza su uso por otros). Cuando empezamos a escribir nuestra base de datos no hay forma de saber si terminará siendo tan importante que merezca la pena compartirla con otros y por ello es mejor empezar a escribirla con la mente puesta en que tal vez en el futuro debamos compartirla.

2.3. Otras cuestiones relativas a los ficheros «.bib»

2.3.1. Uso de abreviaturas y concatenación de cadenas

En un fichero «.bib» es posible definir abreviaturas que simplifiquen la escritura de los registros y eviten errores. Para usar una abreviatura, primero hay que definirla de acuerdo con el siguiente formato:

```
@String{Abreviatura = "Texto sin abreviar"}
```

El nombre de la abreviatura debe empezar por una letra y no puede contener caracteres en blanco ni ninguno de los siguientes caracteres:

```
" # % ' ( ) , = { }
```

El texto sin abreviar se debe especificar entre comillas. Las llaves para delimitar el contenido de “@String” son obligatorias.

La definición se puede hacer en cualquier lugar del fichero «.bib» que no sea el interior de un registro (o de algún otro elemento admitido en este tipo de ficheros). El único requisito en este punto es el de que la definición de una abreviatura debe hacerse *antes* de que ésta sea usada en algún campo.

Las abreviaturas no pueden usarse para los nombres de campo o de registro, sino exclusivamente en el contenido del campo. Y como las abreviaturas no pueden estar encerradas dentro de los delimitadores de campo, cuando todo el contenido del campo es una abreviatura, ese campo no requerirá delimitadores de contenido. Así en el siguiente ejemplo:

```
@STRING{TeX = "The {T}e{X} {B}ook"}
```

```
@Book{eijkhout,  
  author = {Victor Eijkhout},  
  title = {TeX by topic},  
  publisher = {Addison Wesley},  
  year = 1992,  
}
```

```
@Book{knuth,  
  author = {Donald E. Knuth},  
  title = TeX,  
  publisher = {Addison Wesley},
```

```

    year = 1986,
}

```

Hemos definido la abreviatura “TeX” y dos registros, en el título del primero de ellos aparece la cadena “TeX”, pero BIB_TEX no la toma como una abreviatura, porque tal cadena se encuentra dentro de unas llaves que delimitan el contenido de un campo. Por el contrario, en el segundo registro, el texto “TeX” usado como contenido del campo título, es tomado como un uso de la abreviatura, dado que no está dentro de ningún tipo de delimitadores de contenido de campo³⁰.

Si la abreviatura no cubre todo el contenido del campo, entonces debemos escribir entre delimitadores normales el contenido no cubierto por la abreviatura, y *concatenar* dicho contenido con el de la abreviatura.

En los ficheros «.bib» el carácter para concatenar cadenas de texto es “#”. Este carácter permite concatenar abreviaturas entre sí, cadenas de texto entre sí y abreviaturas con cadenas de texto. Por ejemplo, si en el fichero «.bib» de nuestro anterior ejemplo tuviéramos un libro llamado “*Comentarios a ‘The TeX Book’ de Knuth*”, y quisiéramos aprovechar la abreviatura que hemos definido para “*The TeX Book*”, podríamos escribir el campo «title» de dicho libro de la siguiente manera:

```

title = {Comentarios a ‘} # TeX # {’ de Knuth},

```

Hay tipos de datos, como el nombre de las revistas científicas, para los que es muy corriente el uso de abreviaturas. De hecho en ciertos campos de conocimiento circulan ficheros en el que se recogen las abreviaturas corrientemente usadas para las revistas más importantes. En tales casos podemos usar dichos ficheros simplemente indicando su nombre en la orden “\bibliography” antes del nombre de nuestra base de datos bibliográfica, y separando ambos nombre por una coma.

La siguiente línea, por ejemplo, provocaría que “MiBiblio.bib” (que cabe suponer que es nuestra base de datos) se concatenara con el fichero distribuido por la *American Mathematical Society* que contiene las abreviaturas habitualmente usadas para las revistas matemáticas:

```

\bibliography{mrabbrev,MiBiblio}

```

En los estilos estándar de BIB_TEX hay ya varias abreviaturas definidas por defecto: Están definidos los nombres de los meses del año en inglés (véase las abreviaturas que mencioné al hablar del campo «month» en la nota 26), así como el nombre de varias revistas dedicadas al campo de la informática³¹.

³⁰En el ejemplo he escrito para la abreviatura “TeX”, pero, como BIB_TEX no distingue entre mayúsculas y minúsculas, habríamos obtenido el mismo resultado escribiendo “TEX” o “tex” tanto en la definición de la abreviatura como más tarde cuando ésta es usada. Podríamos también haber definido “TEX” y luego usar “tex” o “TeX”.

³¹Las abreviaturas definidas en el estilo plain son las siguientes: acmcs (ACM Computing Surveys), acta (Acta Informatica), cacm (Communications of the ACM), ibmjrd (IBM Journal of Research and Development), ibmsj (IBM Systems Journal), ieeese (IEEE Transactions on Software

2.3.2. Referencias cruzadas entre registros

En los ficheros «.bib» se admiten dos tipos de referencias cruzadas:

1. Usando el comando de \LaTeX “\cite”.
2. Mediante el campo «crossref».

En cuanto al uso de “\cite”, podemos usarlo dentro de un campo para hacer referencia a otro registro. Normalmente esto se hace en el campo «note», pero se puede hacer en cualquier otro. Esto provocará que al citar en nuestro documento \LaTeX una referencia que a su vez cita a otra, ambas se consideren citadas en el documento, y se incluyan las dos en la lista de referencias bibliográficas³².

Pero también es posible usar referencias cruzadas de otra manera: para que $\text{BIB}\text{\TeX}$ sepa que el valor de ciertos campos que normalmente forman parte del contenido de un determinado tipo de registro, pero que en un registro concreto no han sido incluidos, debe ser tomado de otro registro distinto. Por ejemplo, si tenemos un libro homenaje a cierto autor, en el que varios autores han colaborado, cada uno de ellos con un artículo. Si quisiéramos volcar todos estos artículos a nuestra base de datos, podríamos:

1. Crear tantos registros como artículos haya e incluir en cada uno de ellos todos los datos requeridos, gran parte de los cuales serán idénticos en todos los registros.
2. Abreviar mediante referencias cruzadas, para los datos idénticos escribirlos una sola vez.

Para seguir el segundo procedimiento, deberíamos empezar por crear un registro del tipo “Proceedings” para el libro homenaje en sí mismo considerado, rellenar en él el valor de todos los campos que podamos, y luego en el registro correspondiente a cada uno de los artículos extraídos de dicho libro (que serían referencias

Engineering), ieeetc (IEEE Transactions on Computers), ieeetcad (IEEE Transactions on Computer-Aided Design of Integrated Circuits), ipl (Information Processing Letters), jacm (Journal of the ACM), jcss (Journal of Computer and System Sciences), scp (Science of Computer Programming), sicomp (SIAM Journal on Computing), tocs (ACM Transactions on Computer Systems), tods (ACM Transactions on Database Systems), tog (ACM Transactions on Graphics), toms (ACM Transactions on Mathematical Software), toois (ACM Transactions on Office Information Systems), topas (ACM Transactions on Programming Languages and Systems), tcs (Theoretical Computer Science)

³²Este es uno de los casos en los que $\text{BIB}\text{\TeX}$ requerirá una compilación adicional. Porque el comando “\cite” incorporado a un registro bibliográfico no será compilado por \LaTeX hasta que se compila una versión que incorpore el contenido del fichero «.bb1». E incluso es posible que al incorporar esta nueva cita, sea preciso, tras la segunda ejecución de \LaTeX , ejecutar de nuevo $\text{BIB}\text{\TeX}$.

de tipo «InProceedings»), usar como valor para el campo «crossref», el de la clave que hayamos asignado al registro «Proceedings»³³.

El efecto de todo ello será que:

1. En los registros «InProceedings» que escribamos para cada uno de los artículos contenidos en el libro, podremos no incluir los campos que ya están en el registro correspondiente al libro general, pues al apuntar «Crossref» al registro correcto, BIB_TE_X sabrá que los datos que faltan en el primer registro deben ser tomados del segundo. Esto, además de reducir el tamaño de la base de datos, y ahorrar tiempo para la introducción de los mismos, contribuye a garantizar que todas las veces que estos datos aparezcan, estén escritos exactamente igual.
2. Si en un documento L^AT_EX citamos al menos dos artículos que apunten a un mismo registro, automáticamente se incluirá en la lista final de referencias bibliográficas, además de los trabajos directamente citados por nosotros, el registro correspondiente al libro que los contiene.

Para que lo anterior funcione correctamente, es preciso que el registro *referenciado* mediante el campo «crossref» (el registro «Proceedings») se encuentre en la base de datos *después* de los registros referenciadores (los registros «InProceedings»).

Los estilos estándar de BIB_TE_X admiten referencias cruzadas en los siguientes casos:

1. Los registros de tipo «InProceedings» pueden hacer referencia a registros «Proceedings» o «Book».
2. Los registros de tipo «InCollection» pueden hacer referencia a registros «Book».

2.4. Herramientas para manejar ficheros «.bib»

Existen varias herramientas para la creación y mantenimiento de bases de datos de BIB_TE_X cuya utilidad fundamental es la de asegurarse de que todos los registros están correctamente escritos y de que no hemos cometido errores ni en el nombre del tipo del registro ni en el de los campos.

Como yo trabajo exclusivamente en sistemas Linux, en este apartado me concentraré en herramientas que funcionen en tales sistemas. Para sistemas MS-DOS, Mac-OS o Microsoft Windows existen otras herramientas que yo no he probado y

³³De hecho, si lo pensamos bien, el único sentido que tiene la presencia del campo «booktitle» en los registros de tipo «Proceedings» es el de que este campo se use para llenar el campo del mismo nombre en los registros «InProceedings» que apunten a este.

sobre las que, en consecuencia, no puedo opinar. Existen asimismo herramientas escritas en java que, en teoría, funcionan en cualquier plataforma, aunque yo no las he probado. He leído buenos comentarios sobre dos de ellas: “jbibtexmanager” y “jabref”, ambas son fáciles de localizar en Internet.

2.4.1. Herramientas gráficas

Existen varias herramientas gráficas que permiten trabajar con ficheros «.bib» mediante ventanas, botones, barras de herramientas, etc. Las utilidades ofrecidas por estas herramientas son bastante similares. Entre ellas he probado (porque están disponibles directamente en Debian):

pybliographer: Posiblemente sea la más completa de todas las que conozco (aunque admito que no es la que más me gusta, pues es muy difícil de manejar sin el ratón, y yo tengo alergia al ratón). Frente a otras herramientas ofrece las siguientes ventajas: que funciona también en modo de consola, que además del formato BIBTEX puede manejar otros formatos de bases de datos bibliográficas y, lo que es más importante: es capaz de convertir bases de datos desde unos formatos a otros.

Gbib: Es mi herramienta favorita en este grupo³⁴. Forma parte del proyecto Gnome, aunque no suele instalarse por defecto. Posiblemente su principal inconveniente esté en el hecho de que no permite controlar el tipo de codificación en el que hay que grabar el fichero, sino que éste es guardado automáticamente en la codificación preestablecida para el sistema³⁵, aunque en realidad este defecto es predicable de las tres herramientas que recojo en la presente sección.

Tkbibtex: Una herramienta muy parecida a las anteriores: permite crear bases de datos nuevas, abrir bases existentes, añadir registros, hacer búsquedas, etc. Apenas le he echado un vistazo.

³⁴Lo que no es mucho decir, pues yo soy usuario de GNU Emacs (del que me ocupo en la próxima sección).

³⁵Esto puede ocasionar una tediosa labor de *reparación*. Sobre todo si se tiene en cuenta que hasta hace poco la codificación estándar para sistemas Linux españoles era «latin1» o «latin15» y en los últimos tiempos las distribuciones empiezan a migrar a «utf8». De modo que si en un sistema instalado recientemente abrimos una base de datos escrita hace algún tiempo, introducimos cambios en ella y los guardamos, cabe la posibilidad de que, si no tenemos control sobre las codificaciones usadas, todos los caracteres no anglosajones del fichero se transformen en caracteres ininteligibles que luego habrá que restuarar a mano.

2.4.2. El modo BIB_TE_X en un editor de texto

Dado que los ficheros «.bib» son ficheros de texto, otra posibilidad es editarlos con algún editor de texto que esté provisto de alguna extensión que facilite de alguna manera el trabajo con ellos.

2.4.3. Otras herramientas disponibles en sistemas linux

Además de las indicadas existen muchísimas otras herramientas disponibles.

3. Los estilos bibliográficos

3.1. Descripción de los estilos estándar de BIB_TE_X

BIB_TE_X se acompaña de cuatro estilos llamados «plain», «abbrv», «alpha» y «unsrt». El estilo esencial es el primero, y los otros funcionan como él en todo salvo en uno o dos puntos concretos.

A continuación describiré estos estilos atendiendo exclusivamente a sus características principales y sin entrar en excesivos detalles (cuya descripción con palabras, por otra parte, ocuparía demasiado espacio). Mi consejo para ver exactamente las características de cada uno de los estilos es escribir un fichero «.bib» de prueba, incluir un registro de cada uno de los tipos admitidos, llenar *todos* los campos posibles de cada registro, e incluir dicha base de datos en un documento de prueba, en el que hayamos incluido un comando “\nocite{*}”.

En la próxima explicación, menciono varias veces la noción *autoría*. Con ella me quiero referir a lo que ya se explicó en la página 16.

Estilo “plain”. Sus características son:

1. La lista bibliográfica final se ordena alfabéticamente atendiendo a la autoría, y si hubiera más de una obra del mismo autor, se toma en cuenta al año de las mismas y después el título. Si sigue habiendo igualdad tras aplicar los criterios anteriores, el último criterio es el del orden en el que fueron citadas y, para obras citadas simultáneamente mediante “\nocite{*}”, el orden que tengan en la base de datos.
2. Las obras incluidas en la lista son numeradas consecutivamente y el número asignado a cada una de ellas, entre corchetes, se convierte en la etiqueta identificativa de la misma que será impresa en el lugar en el que se encuentren los comandos “\cite” existentes en el cuerpo del documento.
3. Los datos de los campos se incluyen completos.

4. Para ciertos campos se añaden determinadas palabras o abreviaturas en inglés. Por ejemplo, tras el contenido del campo «`edition`» se añade la palabra “*edition*”, y hay tipos de registro en los que el nombre del campo (en inglés) forma parte de la referencia.

El estilo “`abbrv`”: Es idéntico al estilo «`plain`» salvo en el hecho de que para ciertos datos se usan abreviaturas y así el nombre de pila de los autores es sustituido por sus iniciales y el nombre de ciertas revistas (que están predefinidas en el estilo y que se refieren a la informática) es sustituido por su abreviatura. Para la mayor parte de los usuarios, que no usan las revistas predefinidas en el estilo, la única diferencia con «`plain`» es que del nombre propio del autor sólo se usan las iniciales.

El estilo “`alpha`”: Se distingue del estilo «`plain`» exclusivamente en el hecho de que la etiqueta de identificación de cada obra en la lista no es un número, sino un texto generado automáticamente a partir de la *autoría* de la referencia, el año de publicación y, en ocasiones, el inicio del título³⁶. Asimismo la lista bibliográfica se ordena alfabéticamente según las etiquetas asignadas, y el comando “`\cite`” imprime, en el lugar en el que se encuentre, la etiqueta asignada a la obra citada.

El estilo “`unsrt`”: Es igual al estilo «`plain`» salvo en el hecho de que en él la lista bibliográfica no se ordena alfabéticamente, sino según el orden en el que las distintas obras que aparecen en ella fueron citadas por primera vez. En este caso, para las obras incluidas en la lista de referencias mediante un comando “`\nocite`” se considerará que fueron citadas en el lugar en el que se encuentre tal comando. Y para el comando “`\nocite{*}`”, se usará el orden en el que las referencias se encuentren en el fichero «`.bib`», pero sólo para las referencias que no hubieran sido citadas antes de “`\nocite{*}`”.

En cuanto a las características concretas de formateo de los campos son similares en todos estos estilos. Los aspectos más llamativos son:

1. Se escribe el nombre de pila antes que el apellido, lo que no deja de ser chocante en aquellos casos en los que la lista de referencias se ordena alfabéticamente a partir de los apellidos (estilos «`plain`» y «`abbrv`», ya que ello provoca una ordenación alfabética tomando como referencia, no la primera palabra de cada párrafo, sino la segunda o la tercera.

³⁶Como regla se toman las tres primeras letras del nombre del autor y las dos últimas cifras del año de publicación. En obras con varios autores se toma la inicial de los tres primeros seguida de un signo “+” en formato de superíndice, si hay más de tres autores. Si con estos criterios a dos referencias se les asignara la misma etiqueta, las obras se ordenan relativamente entre sí, atendiendo a los criterios de ordenación del estilo «`plain`», y a las etiquetas se añade una letra para diferenciarlas entre sí: una “a” a la primera, una “b” a la segunda, etc.

2. Los distintos elementos de la referencia se separan mediante un punto. Lo que también provoca problemas cuando el contenido de un campo no es numérico y no empieza con una mayúscula, pues los estilos sólo ajustan el uso de las mayúsculas en el campo «title».
3. Todos los campos aparecen con letra normal, salvo el título de los libros y trabajos académicos y el nombre de las revistas, que aparecen en cursiva.

3.2. Estilos adicionales

Junto con estos estilos estándar, existen multitud de estilos disponibles. En la CTAN hay varios de ellos preparados para su descarga, y algunas distribuciones de \LaTeX incluyen por defecto ciertos estilos adicionales³⁷.

Entre los estilos adicionales hay algunos cuya instalación es tan corriente que son estilos *casi* estándar. Otros no son tan conocidos, pero resultan interesantes. Muchos de ellos consisten exclusivamente en uno o varios estilos para \BibTeX , y otros incluyen también un *paquete* que hay que incluir en el documento principal, mediante “\usepackage”, para poder usar el estilo.

No voy a explicar todos los estilos adicionales, porque ello sería tedioso, sino exclusivamente los más conocidos e interesantes:

Apalike: Este estilo también fue escrito por Oren Patashnik (el autor de \BibTeX). Es un estilo de tipo “Autor-año” y para su correcto funcionamiento, hay que cargar en los documentos \LaTeX el paquete del mismo nombre que el estilo: «apalike».

Natbib: Natbib es al mismo tiempo un estilo y un paquete, aunque quizás en él destaque más la parte de paquete, por la cantidad de modificaciones que incorpora al funcionamiento del comando “\cite”.

Durante mucho tiempo (hasta que conocí el trabajo de unos compañeros de la Universidad de Murcia que han diseñado «flexbib»), he considerado que «natbib» era la mejor opción. Permite elegir entre citas del tipo Autor-año o citas numéricas. Incluye versiones adaptadas de los estilos estándar (salvo el estilo «alpha») llamadas «plainnat», «abbrvnat» y «unsrtnat»; añade un campo «usr», dispone de una estupenda documentación e incorpora numerosas modificaciones y mejoras en el procedimiento de las citas.

³⁷Por ello, antes de buscar estilos adicionales, conviene que nos aseguremos de cuáles son los estilos incluidos en nuestra distribución de \LaTeX . Lo que se puede hacer por distintos procedimientos que dependen de nuestra concreta instalación. A falta de algún otro procedimiento específico, podemos buscar los ficheros de extensión «.bst» existentes en nuestro sistema; o mejor: los existentes en alguno de los directorios en los que \BibTeX busca los ficheros de estilo. En la nota 7 se explica cómo localizar cuáles son esos directorios en un sistema Unix/Linux.

3.3. Cómo españolizar los estilos estándar

Los estilos estándar de BIB_TE_X están pensados para documentos que se vayan a redactar en inglés y por lo tanto usan partículas, palabras y abreviaturas en inglés. Para conseguir que nuestra lista de referencias figure totalmente en español, sin palabras en inglés, existen varios procedimientos.

1. Lo más cómodo es usar algún estilo o paquete que añada a los paquetes estándar la sensibilidad a «babel». Hay varios paquetes que hacen eso, y más cosas. Podemos citar “abstyles-babel”, diseñado por Tomás Bautista y que es una adaptación de los abstyles, para que sean sensibles a «babel», o los paquetes «babelbib» y «flexbib».
2. Usar algún estilo diseñado para el idioma español. En Internet se pueden localizar algunos, y en la CTAN está disponible el estilo “spain.bst” basado en las indicaciones que sobre referencias bibliográficas se contienen en el “Diccionario de ortografía técnica” de J. Martínez de Sousa.
3. Podemos también generar nuestro propio estilo bibliográfico, preparado para trabajar en un idioma concreto. Al respecto estúdiese lo dicho en la sección 4.
4. Por último, podemos modificar el fichero donde se contiene algún otro estilo, y prepararlo para que trabaje en español. Esta última posibilidad exige un mayor conocimiento del funcionamiento interno de BIB_TE_X. Conviene por lo tanto aprender muy bien lo que se expone en la parte III de este documento.

Para esta parte de la guía, referida al nivel básico, el procedimiento más adecuado es el primero de los mencionados.

Parte II

BIBTEX intermedio

4. Generación automatizada de estilos bibliográficos

Si ninguno de los estilos bibliográficos que acompañan a BIBTEX, o que están disponibles en la CTAN nos convence, podemos generar nuestro propio estilo bibliográfico. Ello se puede hacer escribiendo a mano nuestro propio estilo, lo que exige un profundo conocimiento del lenguaje usado por los ficheros de estilo (véase la parte III), o mediante algunas herramientas dirigidas a la generación de ficheros «.bst». Estas herramientas, cuando son ejecutadas, empiezan preguntando al usuario por las características que debe tener el nuevo estilo (preguntas en inglés, *aof course!*), y, con las respuestas dadas, se genera un fichero «.bst» que contiene el nuevo estilo o, lo que es más normal, un fichero intermedio a partir del cual es posible generar el fichero «.bst».

La herramienta que voy a analizar aquí funciona de esa segunda manera: el fichero intermedio tiene la extensión «.dbj» y la herramienta se llama «makebst» y forma parte del paquete «custom-bib» (disponible en la CTAN).

4.1. Makebst

Makebst es un programa para TEX diseñado por Patrick W. Daly que permite generar ficheros bibliográficos personalizados.

4.2. Uso de los ficheros «.dbj»

5. Paquetes LATEX relacionados con BIBTEX

En la CTAN existen numerosos paquetes que añaden utilidades a BIBTEX o que de alguna manera amplían las posibilidades del sistema.

No me es posible ocuparme de todos, y por ello seleccionaré los que me parecen más interesantes.

- 5.1. Referencias bibliográficas completas en el lugar donde son citadas**
- 5.2. Generación de varias listas bibliográficas**
- 5.3. Otros paquetes**
- 6. Otras cuestiones de interés**
 - 6.1. El comando newblock y la opción openbib**
 - 6.2. Elementos @preamble en los ficheros «.bib»**
 - 6.3. Comentarios y registros desconocidos en los ficheros «.bib»**

Parte III

BIB_TE_X avanzado

7. Los ficheros de estilo como programas

Los ficheros de estilo manejados por BIB_TE_X son en realidad *programas* escritos en un lenguaje innominado que informan a BIB_TE_X acerca de cómo deben formatearse los datos leídos de la base de datos. Como el lenguaje no tiene nombre oficial, cuando me tenga que referir a él lo denominaré BST que es la extensión utilizada por los ficheros de estilo³⁸.

7.1. Cuestiones generales

7.1.1. Reglas generales de sintaxis

Las características generales de BST son las siguientes:

1. No se distingue entre mayúsculas y minúsculas *para los elementos del lenguaje*, entendiéndose por tales los nombres de los comandos, variables y funciones, así como las cadenas de texto usadas como patrón en determinadas funciones internas. Por el contrario, en el contenido de los campos y de las variables de texto, sí se distinguen las mayúsculas de las minúsculas.
2. Los saltos de línea y los tabuladores se tratan como espacios en blanco, y dos o más espacios en blanco se tratan como si fueran sólo uno. Y a diferencia de otros lenguajes no existe ningún carácter reservado para indicar el fin de una instrucción: ni el punto y coma (como en C o Pascal) ni el salto de línea (como en Basic y similares). Los saltos de línea se usan exclusivamente para hacer más comprensible para los seres humanos lo que una función hace.

No obstante lo anterior, de cara a la información que BIB_TE_X emite cuando se produce algún error (y que se almacena en el fichero «.blg» generado tras la ejecución de BIB_TE_X), se recomienda:

- a) Insertar una o más líneas en blanco entre la definición de dos funciones.
- b) No introducir líneas en blanco dentro de una función.

Lo anterior, por supuesto, no se aplica al contenido de las variables de texto. En ellas todos los espacios en blanco son relevantes y se distingue entre

³⁸Eso no significa que el lenguaje se llame BST. Simplemente, si no uso un nombre para referirme a él, en las próximas páginas tendría que emplear demasiados circunloquios.

espacios en blanco y tabuladores³⁹. Asimismo las constantes de texto, que se introducen entrecomilladas, deben escribirse en la misma línea: entre las comillas de apertura y cierre de una cadena de texto no puede haber un salto de línea.

3. Se puede hacer uso de las llaves para agrupar bloques de código.
4. El carácter “%” está reservado para los comentarios: Todo lo que se escriba a la derecha del mismo, hasta el final de la línea, será ignorado.

7.1.2. Estructura de los programas BST

7.2. Objetos del lenguaje BST

7.2.1. La lista de datos

7.2.2. Variables y constantes

7.2.3. Funciones

8. Conclusión: Otros usos de BIB_TE_X

BIB_TE_X fue diseñado para la gestión de la bibliografía. Pero la flexibilidad de formato de los ficheros «.bib», y la potencia de los ficheros de estilo, permiten extender sus posibilidades hasta el punto de que podamos usarlo para muchas otras tareas. Podemos considerar a BIB_TE_X como una herramienta que nos permite insertar en un documento información procedente de una base de datos externa, así como decidir de qué manera debe formatearse tal información.

En [Cascales Salinas y cols. \(2000\)](#) se explica cómo usar BIB_TE_X para generar una base de datos de problemas matemáticos. Y en [Markey \(2005\)](#) se habla del posible uso de BIB_TE_X para generar una lista de publicaciones y una libreta de direcciones. Díez de Arriba y Javier Bezos, por su parte, han diseñado el paquete Gloss que usa BIB_TE_X para generar glosarios, mientras que yo, que me dedico a la enseñanza del Derecho, he usado BIB_TE_X para con una base de datos de exámenes tipo test, y, en publicaciones de tipo jurídico, con una base de datos de sentencias, de tal modo que un simple comando “\cite” pudiera insertar en el texto un rótulo identificativo de una sentencia (del tipo «STS 21-4-56») y, al final del documento, una lista con las sentencias citadas.

Estos usos *alternativos* de BIB_TE_X pueden requerir un *paquete* para implementarlos. Sobre todo si se pretende que su uso no impida, en el mismo documento, el

³⁹Ello con independencia de que si finalmente el contenido de la variable termina formando parte de una lista de referencias insertada en un documento «.tex», los espacios en blanco y tabuladores existentes en ella reciban el tratamiento que L_AT_EX da a tales caracteres.

uso de BIB_TE_X al modo *normal*. Así ocurre en el caso de la generación de glosarios. O pueden exigir el diseño de un tipo nuevo de registro y una función que lo formatee incorporada a alguno de los estilos estándar de BIB_TE_X (como hice para la base de datos de sentencias y para la de exámenes tipo test), o simplemente un uso determinado de los registros normales.

Lo importante es que tengamos claro que BIB_TE_X esconde muchas más posibilidades de lo que a primera vista pudiera parecer.

Apéndice A: GNU Free Documentation License (Licencia GNU para documentación)

Version 1.3, 3 November 2008

Copyright ©2000,2001,2002,2007,2008 Free Software Foundation, Inc.

«<https://fsf.org/>»

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

2. VERBATIM COPYING

3. COPYING IN QUANTITY

4. MODIFICATIONS

5. COMBINING DOCUMENTS

6. COLLECTIONS OF DOCUMENTS

7. AGGREGATION WITH INDEPENDENT WORKS

8. TRANSLATION

9. TERMINATION

10. FUTURE REVISIONS OF THIS LICENSE

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright ©YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Referencias

- Cascales Salinas, B., Lucas Saorín, P., Mira Ros, J. M., Pallarés Ruiz, A., y Sánchez-Pedreño Guillén, S. (2000). *Latex, una imprenta en sus manos*. Aula Documental de Investigación.
- Cascales Salinas, B., Lucas Saorín, P., Mira Ros, J. M., Pallarés Ruiz, A., y Sánchez-Pedreño Guillén, S. (2003). *El libro de latex*. Pearson-Prentice Hall.
- Kopka, H., y Daly, P. W. (2004). *Guide to LaTeX* (4.^a ed.). Addison-Wesley.

- Lamport, L. (1986). *Latex: A document preparation system*. Addison-Wesley.
- Markey, N. (2005). Tame the beast [Manual de software informático]. Descargado de <http://www.ctan.org/tex-archive/info/bibtex/tamethebeast/>
- Mira Ros, J. M. (2004). Bibliografía flexible: el sistema flexbib. *TeXemplares*(6), 8-26. Descargado de w3.mecanica.upm.es/CervanTeX/texemplares6.pdf
- Patashnik, O. (1988a, February). Bibtexing [Manual de software informático]. Descargado de <http://www.ctan.org/tex-archive/biblio/bibtex/contrib/doc/btxdoc.pdf>
- Patashnik, O. (1988b, February). Designing bibtex styles [Manual de software informático]. Descargado de <http://www.ctan.org/tex-archive/biblio/bibtex/contrib/doc/btxhak.pdf>
- Seidel, L. (1998, March). Bases de datos bibliográficas, latex y el idioma español [Manual de software informático]. Descargado de <ftp://tex.unirioja.es/pub/tex/doc/bibliogr.pdf>
- Young, D. (2002, May). Using bibtex [Manual de software informático]. Descargado de www.maths.anu.edu.au/~chrisw/LaTeX/bibtex.pdf