

1. Introducción

La creación de un texto científico largo no es una tarea sencilla. A parte de la dificultad obvia de la complejidad del propio texto aparece la de la tipografía y la presentación.

En la actualidad es muy sencillo crear un texto de cualquier tipo; los procesadores de texto son una herramienta muy popular y sencilla. Escribir un documento, navegar por internet, manipular archivos... Son tareas básicas que cualquier usuario puede realizar sin problemas.

Pero la preparación de un texto científico tiene particularidades especiales que pueden obligarnos a descartar las herramientas convencionales. Un procesador de textos sirve para todo, pero no utilizaremos una navaja suiza para hacer la cena de navidad.

Los requerimientos particulares de un texto científico son los siguientes:

- Es un texto altamente estructurado. Debe comunicar una idea, no contar una historia ni encandilar con sus filigranas.
- La sobriedad y la claridad son elementos importantes. La preparación del texto suele estar relacionada con su credibilidad.
- Debe utilizar reglas tipográficas estrictas. Los tamaños de fuente, los espaciados, los tipos de letra... Todo es parte de un estándar que hay que cumplir.
- Probablemente incluirá gran cantidad de símbolos especiales y caracteres matemáticos. Que las fórmulas se lean con claridad puede convertirse en lo más importante.
- La posición, marcación y rotulado de los gráficos también sigue normas de estricto cumplimiento.
- Se utilizarán formatos de archivos vectoriales, nunca de mapa de bits. Una recta tiene que ser una recta, no una sucesión de puntos.
- La salida del documento será siempre un formato de preimpresión como postscript o pdf. Este punto es el más crítico porque impone los formatos de cualquier cosa que se introduzca en el documento ya sean gráficos, imágenes, esquemas...

Podemos cumplir todos estos requisitos utilizando un procesador de textos pero será muy laborioso y molesto. Los procesadores de textos están orientados a la creación de documentos polivalentes y de fácil creación. Nos interesa un tipo muy concreto de documento para el que ya existen herramientas mucho más especializadas y por consiguiente más adecuadas.

Aprender cualquier cosa requiere sacrificio y tiempo. Es tarea de nuestro sentido crítico juzgar la conveniencia de entrar en algo nuevo

2. T_EX y L^AT_EX

T_EX Es sin duda el formato de tratamiento de textos científicos por excelencia. Es tan completo y polivalente que se puede considerarse un lenguaje de programación. T_EX es un lenguaje de marcas como lo es el HTML. Su misión es únicamente la preparación de textos de alta calidad para publicaciones de todo todo tipo. Su mayor baza

es que los documentos que producen tienen una calidad inmejorable, equivalente a la obtenida con las mejores herramientas de edición profesionales.

\LaTeX es una colección de macros que permiten escribir directamente el documento sin las dificultades asociadas al \TeX puro. La programación \TeX sigue siendo y será por mucho tiempo la herramienta tipográfica definitiva. El código \LaTeX tiene más o menos esta pinta:

```
\section{\TeX\ y \LaTeX}
\label{sec:latex}
```

\TeX Es sin duda el formato de tratamiento de textos científicos por excelencia. Es tan completo y polivalente que se puede considerarse un lenguaje de programación. \TeX es un lenguaje de marcas como lo es el HTML. Su misión es únicamente la preparación de textos de alta calidad para publicaciones de todo todo tipo. Su mayor baza es que los documentos que producen tienen una calidad inmejorable, equivalente a la obtenida con las mejores herramientas de edición profesionales.

\LaTeX es una colección de macros que permiten escribir directamente el documento sin las dificultades asociadas al \TeX puro. La programación \TeX sigue siendo y será por mucho tiempo la herramienta tipográfica definitiva. El código \LaTeX tiene más o menos esta pinta:

El gran éxito de \LaTeX es que cumple con creces los requerimientos planteados en la introducción. Es incluso una gran ayuda para mantener el texto claro y legible. Uno de mis profesores suele decir que “un problema escrito en \LaTeX parece más sencillo”.

Es el formato *de facto* de las revistas y publicaciones científicas. Algunas de ellas incluso han definido sus propios estilos para que los artículos candidatos tengan el formato definitivo antes de su edición. La Sociedad Matemática Americana (AMS) da nombre a algunos de los paquetes de símbolos matemáticos.

Sólo tiene un defecto, requiere un aprendizaje. No me refiero al típico aprendizaje de un programa en el que uno tocando más o menos se hace con el concepto. \LaTeX no es trivial; podemos necesitar material adicional como libros, plantillas, ejemplos.

Pero no es difícil, la única razón por la que aparece un incomprensible sentido de rechazo inicial es porque el texto debe ser compilado, esto es, el documento \LaTeX debe ser interpretado para obtener un documento en algún formato de impresión (dvi o pdf). Esta supuesta complejidad que lo diferencia de los otros formatos de preparación de textos, es lo que le da toda la potencia. Al igual que los lenguajes de programación existe una diferencia entre el código y el producto.

2.1. Problemas asociados con los procesadores de textos

Los procesadores de textos convencionales no son una herramienta adecuada para la creación de textos científicos por varias razones. La primera y quizás más importantes es porque su formato, XML, no está diseñado para ese fin. El XML es un formato pensado para la producción de textos multimedia. No soy ni mucho menos un experto en los formatos *ML pero su diseño busca principalmente polivalencia.

Otro de los motivos es técnico. Los editores de textos actuales son del tipo WY-SIWYG (What You See Is What You Get, lo que ves es lo que tienes). Este planteamiento tan potente es perfecto para la creación rápida de documentos. Permiten hacer

cualquier cosa y para ello se cargan todas las reglas tipográficas habidas y por haber. Es muy difícil mantener la ley y el orden en un documento XML precisamente porque es difícil discriminar las opciones necesarias de entre todas las posibles.

También está el problema asociado al interfaz gráfico. Cuando abrimos un documento en un procesador de textos estamos viendo la cara bonita de un documento que no es ni parecido. El XML es un lenguaje de marcas muy completo pero que no se parece en nada a lo que vemos en pantalla ni a lo que obtenemos en la impresión final. Esto significa que donde nosotros vemos una figura encapsulada con una justificación de 3 cm en realidad hay bastantes líneas de código. Es por eso que los documentos de este tipo son tan monstruosamente grandes. Suelen estar comprimidos por defecto para que no se aprecie tanto su tamaño. El problema surge cuando lo abrimos con nuestro procesador. ¿Qué sucede cuando intentamos hacer un scroll de un documento de 500 páginas? Pues que muy probablemente sea tremendamente lento.

Se dice que el documento \LaTeX es WYSIWYM (What You See Is What You Mean) porque lo que vemos es exactamente el orden y el formato deseados, aunque nos puedan parecer marcas sin sentido en un pequeño archivo de texto.

3. Distribuciones de \LaTeX

\LaTeX es un programa como cualquier otro. El intérprete que convierte de formato escrito al impreso es bastante ligero, lo importante son los paquetes que incluye la instalación. Un entorno de desarrollo \LaTeX completo suele ocupar entre 250 Mb y 300 Mb. Muchos de estos paquetes cambian según el sistema operativo.

3.1. \TeX

Es la distribución principal de \LaTeX para UNIX. Absolutamente todas las distribuciones de Linux tienen un paquete para \TeX . Casi toda la documentación propia de los sistemas UNIX está redactada en \LaTeX , forma parte del mismo sistema operativo.

3.2. $\text{Mik}\TeX$

Es la distribución más famosa del \LaTeX para Windows. Una vez instalemos el paquete básico de $\text{Mik}\TeX$ no tenemos que preocuparnos de nada más, cada vez que necesitemos un paquete extra el propio programa se dará cuenta y lo instalará automáticamente.

La página del proyecto es, además, una gran fuente de información para los usuarios de \LaTeX en Windows, donde no deja de ser un programa ajeno.

4. Editores

Uno de los problemas asociados a la edición de textos en \LaTeX es que, como todos los lenguajes de programación, necesita un editor para crear cómodamente un documento. Podemos utilizar cualquier editor de textos, incluso el Notepad, pero será mucho mejor utilizar un editor especializado o general.

El orden no es de mejor a peor.

4.1. WinEDT

Es un editor de \LaTeX comercial para entorno Windows. Algunos de los usuarios sostienen que es el mejor entorno de edición \LaTeX que existe. Tiene absolutamente todas las herramientas necesarias para la creación y edición de documentos: gestor de proyectos, editor de ecuaciones... Es una herramienta de gran calidad con el único defecto de que no es gratuito. Podemos descargar una versión de prueba de 30 días, suficiente para probar si encaja en nuestras necesidades.

4.2. \TeX maker

Es un editor multiplataforma, con versiones para los tres principales sistemas operativos actuales (Windows, Linux y MacOSX). Utiliza la poderosa biblioteca gráfica Qt y es libre y gratuito. Su punto de partida es Kile que trataremos a continuación.

4.3. Kile

Es el editor de \LaTeX del entorno de escritorio KDE. Es quizás el mejor editor libre y gratuito especializado. Es recomendable tanto para el principiante como para el experto. Es además un proyecto en continuo crecimiento, libre y gratuito. Es uno de los motivos para dar el salto a Linux y la KDE.

4.4. Emacs

El mayor y mejor editor del universo tiene un módulo, Auc \TeX , que modifica los menús y los atajos de teclado para facilitar la edición directa de textos \LaTeX . Estoy redactando este documento en este editor capaz de hacer cosas absolutamente impensables.

Dispone además del paquete *preview-latex* que embebe en el propio editor las fórmulas, los caracteres especiales, los títulos y las imágenes para poder editar el documento de un modo más intuitivo.

5. \LaTeX sin traumas

Es posible editar documentos \LaTeX sin la necesidad de aprender las complicadas reglas de escritura. Cuando utilizamos procesador de textos en realidad estamos utilizando un gestor que manipula un documento XML. ¿Por qué no crear un editor capaz de hacer exactamente lo mismo pero con \LaTeX ?

5.1. LyX

Este editor existe y se llama LyX. El principal inconveniente de este programa es que ha permanecido en el anonimato durante muchos años. Sólo existía en el entorno Linux donde era utilizado por una insignificante minoría. La mayoría de los usuarios de Linux prefiere crear sus propios documentos \LaTeX sin ningún frontend.

Después de casi una década de existencia por fin se publica una versión para Windows equivalente a la existente en Linux. Esto ha sido posible gracias al uso de la librería gráfica Qt.

El objetivo de LyX es crear documentos \LaTeX sin los inconvenientes asociados a la edición directa. Aunque no es exactamente igual a un procesador de textos su uso

es muy parecido. Nos impone desde el primer momento todas las reglas de escritura estructurada típicas de \LaTeX , no nos deja hacer las tonterías habituales ni nos muestra exactamente el documento final como los otros procesadores. Intenta llegar a un híbrido que tome lo mejor de cada mundo.

LyX puede gustar o no, pero hay que reconocer que es un acierto de diseño y que lleva \LaTeX a todos aquellos que no tienen ni idea de programación ni de edición de documentos.

5.2. ScientificWord

Scientific Word es un procesador de textos del estilo de LyX pero bastante más potente. Es un producto comercial y sólo existe en Windows. Es realmente sencillo de utilizar y permite escribir textos matemáticos con una facilidad pasmosa. ScientificWord es sólo un procesador de textos; MacKichan, la empresa creadora, nos ofrece ScientificWorkplace, una combinación entre procesador de textos y herramienta de cálculo simbólico que puede llegar a ser muy útil.

5.3. \TeX macs

Es también un programa libre y multiplataforma que permite editar documentos \LaTeX sin tener que manipular directamente su código. Su principal característica es que se trata de una modificación del código del editor Emacs con lo que tiene todos sus atajos de teclado y sus posibilidades de configuración.

Es multiplataforma, libre y gratuito pero su uso está reservado a desarrolladores más experimentados. Es tan potente que permite “embeber” sesiones de programas como Matlab, Octave, Maple o Maxima dentro del propio editor para mostrar todas las fórmulas directamente en formato \LaTeX . Exporta automáticamente a todos los formatos clásicos de documento (HTML, PDF...)

6. ¿Qué es la metadocumentación?

Se llama meta* siendo * un ‘lo que sea’ a lo que está aún por debajo de la herramienta. Por ejemplo, se llama metaprogramación a la capacidad de crear programas que generan otros programas, metalenguaje a los patrones existentes en los propios lenguajes que pueden servir para crear otros nuevos.

Se dice que \LaTeX sirve no para la creación de documentos sino para la creación de *metadocumentos*. Esto es porque un documento de \LaTeX puede servir para crear muchísimos tipos distintos de documentos.

La salida natural de \LaTeX es el archivo de impresión DVI. Este archivo suele transformarse en algún otro formato de impresión más convencional como postscript. El paso de postscript a PDF (Portable Document Format) es trivial en un sistema operativo UNIX. Pero esta no es la única posibilidad de un documento \LaTeX . Podemos perfectamente producir documentos en formato RTF (Rich Text Format) que pueden ser leídos por cualquier procesador de textos, o pasarlo por un traductor a HTML y crear una página web como he hecho con esta página.

Esta es la principal potencia de \LaTeX , su capacidad para convertirse en cualquier cosa.