El Cómo de los Cómos

Mark F. Komarinski

Lista de las herramientas, procedimientos y pistas necesarios para acelerar el trabajo de los autores de CÓMOs.

1. Sobre este CÓMO

1.1. La finalidad de este CÓMO

Comencé a escribir este documento el 26 de agosto de 1999, tras dos días de total frustración a la búsqueda de las herramientas necesarias para trabajar. Si llegara a servir de ayuda a una sola persona, se habrá cumplido mi objetivo.

La versión más reciente del documento se puede encontrar en mi página web http://www.cgipc.com/~markk , en formato SGML. También existen versiones en otros formatos en el sitio web del LDP http://www.linuxdoc.org (http://www.linuxdoc.org/).

Existen diversas maneras de contribuir al movimiento Linux sin tener que escribir código fuente.Una de las más importantes es la producción de documentación, que permite a cada persona compartir sus conocimientos con otros miles de personas de todo el mundo. Este CÓMO tiene por fin que Usted se familiarice con el funcionamiento del LDP, y con las herramientas necesarias para escribir su propio CÓMO.

1.2. El LDP

Lo que mostramos a continuación es la traducción de un fragmento del Manifiesto LDP (http://www.linuxdoc.org/manifesto.html)

EL Linux Documentation Project (LDP) dedica sus esfuerzos al desarrollo de documentación gratuita y de máxima calidad para el sistema operativo GNU/Linux. El principal objetivo del LDP es colaborar en todos los aspectos de la documentación de Linux. Esto implica la creación de "CÓMOs" y "Guías". Esperamos poder establecer un sistema de documentación para Linux, que sea sencillo de utilizar y de consultar, para lo cual es necesaria la integración de documentos texinfo, las páginas de manual, los CÓMOs y otros documentos.

Si desea saber más sobre el Linux Documentation Project, visite su página web http://www.linuxdoc.org

1.3. Sugerencias

Si tuviese algún comentario o sugerencia que hacer sobre este CÓMO, diríjase a su autor <`markk@linuxdoc.org`>).

1.4. Copyright

- Si se distribuyera una parte de este trabajo, tendrían que incluirse las instrucciones necesarias y un medio para obtener la versión completa de este manual.
- Pueden reproducirse pequeños extractos como ilustraciones para revisiones o citas en otros trabajos, sin la inclusión de este permiso, siempre que se proporcione una referencia apropiada. Se podrían hacer excepciones a estas reglas si se persigue un fin académico. Si ese es el caso, póngase en contacto con el autor. Estas restricciones han sido creadas para protegernos a nosotros como autores, y, en ningún caso, para poner trabas a los lectores y educadores. Todo el código fuente de este documento (junto con el SGML en que fue escrito el documento) se acoge a la licencia GNU, disponible vía FTP anónimo en el archivo de GNU.

1.5. Reconocimientos y agradecimientos

Me gustaría dar las gracias a todos aquellos que aportaron su punto de vista mientras escribía este documento. Entre ellos se encuentran David Lawyer, Deb Richardson, Daniel Barlow, Greg Ferguson, Mark Craig y otros miembros de la lista `<ldp-discuss@lists.linuxdoc.org>`. Algunas secciones han sido extraídas del HOWTO Index (http://www.linuxdoc.org/HOWTO/) (disponible en muchas réplicas del LDP) y de la documentación de sgmltools. Las secciones sobre acceso en red al CVS fueron parcialmente escritas por Serek (`<ser@serek.arch.pwr.wroc.pl>`), y las relacionadas con DocBook, por Jorge Godoy (`<godoy@conectiva.com.br>`). A ambos, un millón de gracias por su inestimable ayuda.

1.6. Convenciones

A lo largo de este texto, las órdenes se muestran siempre en el formato que describimos a continuación: la orden está precedida por el nombre del intérprete de órdenes activo en ese momento; tras el intérprete de órdenes se inserta el símbolo $ cuando se trata de órdenes que hayan de ejecutarse como usuario normal (no superusuario), o el símbolo # cuando sean órdenes que vayan a ejecutarse como superusuario.

2. Introducción al LDP y a SGML

2.1. El LDP

El Linux Documentation Project (LDP) se puso en marcha para proporcionar a los nuevos usuarios una forma más rápida de obtener información sobre un tema concreto. Éste contiene, no sólo una red de libros sobre administración, red y programación, sino también una gran cantidad de temas individuales escritos por aquellos que los han usado. Si desea averiguar cómo imprimir, debe acceder a Printing HOWTO (http://www.linuxdoc.org/HOWTO/Printing-HOWTO.html). Si desea averiguar si su tarjeta Ethernet funciona con Linux, acuda a Ethernet HOWTO (http://www.linuxdoc.org/HOWTO/Ethernet-HOWTO.html), y así con el resto. Inicialmente, muchos de estos trabajos estaban escritos en formato de texto o HTML. Con el tiempo, necesitabamos una forma mejor de gestionar estos documentos, que permitiera leerlos desde una página web, desde un fichero de texto en un CD-ROM, o incluso desde un asistente personal digital. La solución final fue SGML.

2.2. SGML

El Standard Generalized Markup Language (SGML) es un lenguaje que se basa en códigos insertados en un documento. En este sentido, es parecido a HTML, pero aquí acaban las similitudes. La potencia de SGML consiste en que, a diferencia de WYSIWYG (What You See Is What You Get), Usted no define cosas como el color, el tamaño de la fuente o incluso algunos tipos de formato. En cambio, define elementos (párrafos, secciones, listados de números) y deja al procesador de SGML y al programa final que se preocupen por la colocación, los colores, fuentes, etc. En realidad , esto también lo hace HTML , que no es sino es un subconjunto de SGML. SGML está formado por tres partes. La primera es la estructura, que es lo que comúnmente es conocido como la DTD, o Document Type Definition (Definición de tipo de documento). El DTD define las relaciones entre cada uno de los elementos. El DTD DocBook, empleado para crear este documento, es un claro ejemplo. El DTD incluye las reglas que debe seguir el contenido. La segunda parte es el DSSSL o Document

Style Semantics and Specification Language (Lenguaje de especificaciones y de semántica de estilos de documento). El DSSSL ordena al programa que realiza la representación cómo convertir SGML en algo que pueda leer el ojo humano. Por ejemplo, le ordena que convierta la etiqueta <table> en negrita de 14 puntos si el formato final es RTF, o que la sustituya por la etiqueta <hl> si se trata de HTML. Finalmente, está el contenido, que es lo que realmente representa el procesador de SGML y que es lo que ve el usuario. Este párrafo es contenido, pero también lo sería una imagen, una tabla, un listado de números, etc. El contenido es lo que está rodeado por las etiquetas, que separan cada elemento.

2.3. ¿Por qué utilizar SGML en vez de HTML u otros formatos?

SGML hace mucho más que formatear. Puede crear automáticamente índices, tablas de contenidos y enlaces, al mismo documento o a otro distinto. Los paquetes de Jade y OpenJade también le permiten exportar de SGML a LaTeX, info, text, HTML, y RTF. Desde estos formatos básicos, puede crear otros formatos como MS Word, PostScript, PDF y otros. Los programas como LyX le permiten escribir en formato TeX, param luego exportarlo a SGML, y desde SGML a cualquiera que Usted elija. En el fondo, SGML se preocupa más por la manera de trabajar de los elementos que por el aspecto que estos presentan. Esto constituye una gran ventaja, que le permitirá escribir más rápido, ya que no tiene que preocuparse por la colocación de los párrafos, el tamaño de la fuente y otros detalles.

2.4. Para nuevos autores

Si Usted es nuevo en el LDP y desea encargarse de un CÓMO sin mantener o escribir un nuevo CÓMO o un mini-CÓMO, póngase en contacto con el coordinador del CÓMO en la dirección `<ldp-discuss@lists.linuxdoc.org>`. Esto es para asegurarse de que el coordinador del CÓMO pueda saber quién está trabajando con la documentación. (N. del. T.: si se trata de un Cómo original en castellano, vaya a http://www.insflug.org/ y siga las instrucciones.

Una vez hecho esto, debe escribir su documentación en el formato que elija y presentar un borrador a <ldp-submit@lists.linuxdoc.org>, que será revisado por un voluntario del LDP. En muy pocos días recibirá el borrador devuelto junto con las observaciones del voluntario. Tras haber aplicado estos comentarios a su texto, debe enviar esta versión otra vez a la lista ldp-submit para su presentación final en el LDP.

En este momento, otro voluntario del LDP traducirá el documento a DocBook y le enviará el documento DocBook final. De aquí en adelante, todas los envíos al LDP han de estar en formato DocBook. Si tiene preguntas sobre el sistema de marcas o etiquetas, puede preguntarle al voluntario que le ayudó o recurrir a la lista de DocBook del LDP.

2.5. Listas de correo

Hay algunas listas de correo a las que se puede subscribir para participar en el LDP. En primer lugar está <ldp-discuss@lists.linuxdoc.org>, principal grupo de discusión del LDP. Para suscribirse, envie un mensaje con "subscribe" en la línea del asunto <ldp-discuss-request@lists.linuxdoc.org>. Para borrarse, envie un e-mail con "unsubscribe" en la línea del asunto a <ldp-discuss-request@lists.linuxdoc.org>.

Otra lista es la <ldp-docbook@lists.linuxdoc.org>, orientada al marcado y a cuestiones del propio DocBook. Si tiene algún problema con una etiqueta en particular, puede enviar su pregunta aquí para obtener una respuesta. Puede suscribirse a la lista DocBook enviando un mensaje con "subscribe" en la línea del asunto a <ldp-docbook-request@lists.linuxdoc.org>.

3. Herramientas

En este apartado nos referiremos a algunas de las herramientas que necesitará o querrá utilizar para crear su propia documentación LDP. En primer lugar, vamos a describirlas para posteriormente definirlas mejor y explicar cómo se instalan. En el caso de que recurra a alguna otra herramienta que le ayude a escribir LDP, le rogamos que nos lo comunique y añadiremos una nota al respecto.

3.1. DSSSL

Se necesita la versión Norman Walsh, pero el DSL del LDP es opcional.

3.1.1. El DSSSL Norman Walsh

http://nwalsh.com/docbook/dsssl/db152.zip

El Document Style Semantics and Specification Language dice a Jade cómo convertir un documento SGML para impresión o formato de visualización en línea. El DSSSL es lo que transforma, por ejemplo, una etiqueta de título en una etiqueta <H1> en HTML, o en negrita Times Roman de 14 puntos para RTF. La documentación sobre DSSSL se encuentra en http://nwalsh.com/docbook/dsssl/db152d.zip. Debe advertir que al modificar el DSSSL no se modifica el DocBook. Únicamente cambia la forma en la que aparece el texto convertido. El LDP utiliza un DSSSL modificado que proporciona una tabla de contenidos.

3.1.2. El DSSSL del LDP

http://metalab.unc.edu/gferg/ldp/ldp.dsl

El DSSSL del LDP necesita la versión Norman Walsh (ver información previa), aunque se trata de un DSSSL ligeramente modificado para proporcionar características tales como una tabla de contenidos.

3.2. DTD DocBook (versión 3.1)

Se necesita http://www.oasis-open.org/docbook/sgml/3.1/docbk31.zip

La DTD DocBook define las etiquetas y la estructura de un documento SGML de DocBook. Al modificar la DTD, añadiendo por ejemplo una nueva etiqueta, provoque que deje de considerarse DocBook.

3.3. Jade

Jade y OpenJade son dos de los programas que realizan la mayor parte de la representación y validación de cualquier código no basado en la DTD ni en el DSSSL. Es imprescindible uno de ellos y deberá instalarse una vez que hayan sido instalados la DTD y el DSSSL.

3.3.1. Jade

ftp://ftp.jclark.com/pub/jade/jade-1.2.1.tar.gz

Jade es el procesador frontal para SGML. Utiliza el DSSSL y el DTD DocBook para proporcionar la verificación y representación de SGML en el formato destino.

3.3.2. OpenJade

http://openjade.sourceforge.net/

Se trata de una extensión de Jade escrita por la comunidad DSSSL. Algunas aplicaciones precisan Jade, pero se están poniendo al día para admitir cualquier paquete de software.

3.4. Envoltorios Jade

Estas herramientas son opcionales y podrían instalarse una vez que se encuentren instalados Jade, el DSSSL y la DTD.

3.4.1. sgmltools-lite

http://sgmltools-lite.sourceforge.net/

Estamos ante el sucesor del proyecto sgmltools, que se abandonó oficialmente hace más de un año. Desde entonces, Cees de Groot ha elaborado un proyecto un tanto diferente, que actúa como un envoltorio respecto al procesador Jade SGML. Mejora la estética de la sintaxis. El creador fue capaz de instalar el antiguo paquete sgmltools seguido del sgmltools-lite y pudo formatear este documento con bastante facilidad. Existe incluso una página man (página de manual) que explica la sintaxis del sgmltools.

3.4.2. Herramientas Cygnus DocBook

Sitio de Red Hat: http://www.redhat.com/

Red Hat distribuye tres paquetes, que comienzan con la versión 6.2 y que incluyen soporte DocBook y algunas herramientas. La instalación de las herramientas es muy sencilla, lo que le va a permitir centrarse en escribir en lugar de estar luchando innecesariamente con ellas. En primer lugar, tiene que instalar TeTex, Jade, y JadeTeX . Los tres paquetes se encuentran disponibles en el CD de instalación de esta y otras distribuciones. N. del T.: en el caso de Debian, basta con teclear: **bash# apt-get install cygnus-stylesheets**

3.5. Herramientas de edición

Las siguientes herramientas pueden destinarse a la creación, edición o validación de su CÓMO.

3.5.1. LyX

http://www.lyx.org/

LyX permite escribir SGML con la misma facilidad de uso que la de cualquier procesador de texto normal. No se trata de un programa WYSIWYG, sino más bien de una aplicación WYSIWYM (What You See Is What You MEAN), ya que lo que observa en la pantalla no es necesariamente lo que ocurre una vez que el procesador SGML lo ha llevado a cabo. El aspecto que proporciona LyX se parece, aunque no es exactamente igual, a cómo sería la salida de Jade. Sin embargo, se acerca lo suficiente como para que pueda ver el hilo del documento. Las secciones y subsecciones se encuentran numeradas y resaltadas en negrita, y se recurre a fuentes distintas cuando se trata de subrayar algo como, por ejemplo, etiquetas <code> o <ul> . La mayoría de las etiquetas están ocultas en la ventana principal de LyX mientras está editando, ya que LyX escribe en TeX, y luego exporta el TeX a SGML.

Figure 1. Captura de pantalla LyX

3.5.2. Emacs (PSGML)

Opcional - http://www.lysator.liu.se/~lenst/about_psgml/

Emacs tiene un modo de escritura SGML denominado psgml, que es el modo principal diseñado para editar documentos SGML y XML. Ofrece las características de "coloreado de texto" y de "bonita impresión" que hacen sobresalir a las etiquetas SGML, una manera de insertar etiqueta que no tiene nada que ver con el tecleo manual, y la capacidad para dar validez a su documento mientras lo está escribiendo.

Según los usuarios de Emacs, se trata de un avance importante y muchos confían en él por permitir más versatilidad que cualquier otra herramienta de documentación SGML. Funciona igual de bien con DocBook, LinuxDoc y otros DTDs.

3.5.3. VIM

http://www.vim.org

No sería posible ofrecer una idea completa de Emacs sin hablar de vi. El editor VIM (Vi IMproved) tiene las mismas funciones que un vi corriente, pero además incorpora un modo SGML que coloreará y organizará su pantalla con el fin de mostrar dónde se encuentran las etiquetas.

3.5.4. WordPerfect 9 (Corel Office 2000)

http://www.corel.com/

WordPerfect 9 para la plataforma MS Windows admite SGML y DocBook 3.0. WordPerfect 9 para Linux no tiene capacidad SGML.

Esta es la más asequible de las aplicaciones comerciales que admiten SGML.

3.5.5. sgedit

http://www.tksgml.de/

El programa sgedit le permite editar visualmente ficheros SGML. Tiene la ventaja de que no necesitará conocer previamente ni Emacs ni VI , y además la ventaja de que es una aplicación de plataforma cruzada que funciona tanto con Windows como con Linux. Se trata de una aplicación comercial, pero aún no se ha establecido su precio. Habrá licencias gratuitas para uso privado y académico.

Además de la edición visual, sgedit también validará los documentos que se descarguen y que se soliciten mediante la orden Document→Validate.

3.6. Otros/Referencias

En esta sección se encuentran recogidos libros u otras utilidades que no se pueden clasificar fácilmente (por ahora).

3.6.1. DocBook: The Definitive Guide

http://www.docbook.org/

Este libro lo dio a conocer O'Reilly en octubre de 1999, y es una referencia importante para DocBook. Personalmente, no creo que sea un gran libro práctico, y pone mucho énfasis en XML, pero todas las etiquetas DocBook para la versión 3.1 se encuentran detalladas en un formato muy práctico. Puede conseguirlo en cualquier librería. El libro completo también se encuentra disponible online (en formato HTML y SGML) así como en la URL citada.

3.6.2. Aspell

Opcional - http://aspell.sourceforge.net/

Este corrector ortográfico puede funcionar con etiquetas SGML, y únicamente corrije el contenido de esas etiquetas. Los correctores ortográficos por defecto como ispell intentarán corregir las etiquetas, provocando errores en cada nueva etiqueta.

4. Primeros pasos con DocBook

Este apartado trata sobre el nuevo método para escribir la documentación del LDP, utilizando para ello la DTD DocBook 3.1. Veremos cómo conseguir, instalar, y utilizarlas herramientas, junto con una introducción a las etiquetas de DocBook. Debido a que existen más de 300 etiquetas de DocBook, no podremos estudiarlas todas aquí. Quienes estén interesados en hacerlo pueden dirigirse a http://www.docbook.org (http://www.docbook.org/) para obtener más información.

4.1. Descargar e instalar las herramientas

4.1.1. Uso manual de jade/OpenJade

Esta es la forma rápida y directa que debiera funcionar bien con todas las distribuciones, sea cual sea la que Usted esté usando.

1. Cree un directorio base para almacenar todo, tal como `/usr/local/sgml/`. A partir de ahora lo llamaremos $_toolroot.
2. Instale Jade, DocBook DTD, y DSSSL de forma que la base de cada uno esté bajo $_toolroot (creando `$_toolroot/jade-1.2.1`, `$_toolroot/dtd`, `$_toolroot/dssl`).
3. Necesitará definir la variable de entorno SGML_CATALOG_FILES apuntando a los catálogos que tiene bajo `$_toolroot`. Para ello utilice esta orden: **bash$ export SGML_CATALOG_FILES = $_toolroot/dtd/docbook.cat:$_toolroot/dsssl/docbook/catalog:$_toolroot/jade-1.2.1/dsssl/catalog**
4. Ya puede empezar a utilizar Jade. Para crear ficheros HTML individuales:

   $_toolroot/jade-1.2.1/jade/jade -t sgml -i html -d $_toolroot/dsssl/docbook/html/docbook.dsl howto.sgml

5. Para crear un único fichero HTML grande, añada *-V nochunks* a la orden jade.

4.1.2. sgmltools

Al contrario que en versiones anteriores de sgmltools, necesitará la versión 2.x de sgmltools para trabajar con DocBook. Puesto que algunas de las distribuciones más importantes vienen con sgmltools 1.x, tendrá que eliminar el paquete sgmltools 1.x e instalar, o bien una versión 2.0 , o bien una versión CVS. Para conseguir la última versión de código fuente CVS, puede utilizar las órdenes siguientes:

```
bash$ CVSROOT=:pserver:cvs@cvs.sgmltools.org:/home/cvs
bash$ export CVSROOT
bash$ cvs login
bash$ cvs -z6 get sgmltools
```

La contraseña de CVS es 'cvs'. Una vez descargado utilice simplemente

```
bash$ ./compile
bash$ make
bash# make install
```

para instalar sgmltools. Para los sistemas basados en Red Hat (que utilizan RPM) puede usar la orden rpmfind para conseguir el último sgmltools. El programa rpmfind lo encontrará en http://www.rpmfind.net/ (http://www.rpmfind.net). Asegúrese de obtener sgmltools y no sgml-tools, ya que el último es sgml-tools 1.0.9 y sólo trabaja con documentos LinuxDoc. Para sistemas basados en Debian, que ejecuten 2.2 «Potato» y superiores, apt-get le devolverá el paquete correcto:

```
bash# apt-get install sgmltools-2
```

Al igual que ocurre con Linux Red Hat, el paquete sgml-tools está obsoleto. Procure conseguir sgmltools-2.

4.1.3. Cygnus DocBook Tools.

Estas son las herramientas que ofrece Red Hat 6.2. Asegúrese de que los siguientes paquetes están instalados:

- sgml-common
- docbook

- stylesheets

Red Hat tiene la última version en su sitio web:
http://www.redhat.com/support/errata/RHBA-2000022-01.html.

Descargue/obtenga/busque en la red los RPMs e instálelos del modo habitual (entre como root, y a continuación **rpm -Uvh filename**). Una vez los RPMs están instalados, puede usar las siguientes órdenes para exportar DocBook:

```
bash$ db2html nombre_de_fichero
```

Convierte DocBook en HTML. Se crea un subdirectorio con el nombre del fichero (menos la extensión .sgml) y los ficheros HTML se almacenan allí.

```
bash$ db2pdf nombre_de_fichero
```

Convierte DocBook en un fichero PDF.

4.2. Escribir SGML a mano.

Este tema se trata ampliamente en el documento Using DocBook de Jorge Godoy. Aquéllos que estén interesados pueden leerlo en http://metalab.unc.edu/godoy/using-docbook/using-docbook.html para escribir DocBook utilizando su editor de texto favorito.

> **Si escribe SGML a mano:** SGML tiene más de 300 etiquetas, y las utiliza con mucha más frecuencia que HTML. Es recomendable que utilice un CÓMO ya existente como plantilla y vea cómo los han escrito otros autores. También es recomendable que utilice un editor cómodo como PSGML o WordPerfect para Windows, ya que ofrecen lista de muchas de las etiquetas disponibles.

4.3. Escribir SGML utilizando LyX

4.3.1. Documentos nuevos

Puede comenzar fácilmente un nuevo CÓMO utilizando LyX. Use la orden de menú File→New >From Template... para mostrar los listados de plantillas. Seleccione *Templates* a la derecha de la pantalla y seleccione *docbook_template.lyx* en el listado de ficheros. Seleccione OK, y obtendrá un documento nuevo. Introduzca la información, como título, resumen y nombre de autor, y comience a escribir.

Figure 2. Pantalla de Plantilla DocBook de LyX

4.3.2. Documentos ya existentes

Si tiene un documento LyX, TeX, o un documento de texto, puede importarlo a LyX con la orden File→import. Una vez importado vaya a Layout→Document... En la ventana emergente, en Style, seleccione *SGML (DocBook Article)*. Se le preguntará si quiere convertir todo el texto; contéstele afirmativamente. Tendrá que volver a aplicar la mayoría de las etiquetas, pero es una forma secilla de seleccionar texto y cambiar el estilo. Muchas funciones LyX tienen acceso directo desde el teclado para ayudarle.

Figure 3. Pantalla Layout Document

4.3.3. Exportar documentos a SGML

Una vez que el documento está escrito o convertido, guárdelo en formato LyX . Esto le permitirá editar futuras versiones fácilmente. Después, vaya a File→Export→as DocBook... y el fichero será exportado a DocBook

4.4. Escribir SGML utilizando PSGML

4.4.1. Introducción

Si ha instalado una distribución reciente, probablemente ya tendrá instalado PSGML para utilizarlo con Emacs. Para verificarlo, arranque Emacs y busque la documentación PSGML (**C-h i m psgml**).

De ahora en adelante, damos por hecho que tiene instalado PSGML para utilizarlo con una versión reciente de Emacs GNU. Si todo esto no le ha quedado claro, vea el capítulo gratruíto del libro en formato CD de Bob Ducharme, SGML: http://www.snee.com/bob/sgmlfree/.

4.4.2. Actualizar su .emacs para usar PSGML

Si quiere que Emacs GNU entre en modo PSGML cuando abra un fichero con extensión *.sgml* y que esté listo para editar SGML, asegúrese de que PSGML pueda encontrar la DTD Docbook. Si su distribución ya tiene incorporado PSGML para ulilizarlo con Emacs GNU, probablemente no tendrá ningún problema con esto. Si no es así, puede que necesite definir una variable de entorno que le diga a PSGML dónde buscar el catálogo SGML (la lista de los DTDs).

Por ejemplo:

```
 bash$ export
SGML_CATALOG_FILES=/usr/lib/sgml/catalog
```

A continuación añada algo parecido a lo siguiente en su fichero .emacs:

```
;; ******************************************************************
;; configurar modo  psgml
;; utilizar modo psgml-mode en lugar del modo nativo emacs sgml-
mode
;;

(autoload 'sgml-
mode "psgml" "Modo mayor para editar ficheros SGML." t )
(setq auto-mode-alist
  (append
  (list
    '("\\.sgm$" . sgml-mode)
    '("\\.sgml$" . sgml-mode)
    )
  auto-mode-alist))

;; establecer algunas variables psgml

(setq sgml-auto-activate-dtd t)
(setq sgml-omittag-transparent t)
(setq sgml-balanced-tag-edit t)
(setq sgml-auto-insert-required-elements t)
(setq sgml-live-element-indicator t)
(setq sgml-indent-step nil)
```

```
;; crear tipos de fuentes por asignar a categorías de marcado:

(make-face 'sgml-comment-face)
(make-face 'sgml-start-tag-face)
(make-face 'sgml-end-tag-face)
(make-face 'sgml-entity-face)
(make-face 'sgml-doctype-face) ; datos DOCTYPE
(make-face 'sgml-ignored-face) ; datos ignorados por PSGML
(make-face 'sgml-ms-start-face) ; comienzo de secciones marcadas
(make-face 'sgml-ms-end-face) ; fin de sección marcada
(make-face 'sgml-pi-face) ; instrucciones de procesamiento
(make-face 'sgml-sgml-face) ; la declaración SGML
(make-face 'sgml-shortref-face) ; referencias cortas

;; puede ver una lista de colores disponibles con la or-
den emacs-lisp:
;;
;; list-colors-display
;;
;; sírvase añadir sus propios colores, estos no es-
tán muy combinados
(set-face-foreground 'sgml-comment-face "coral")
;(set-face-background 'sgml-comment-face "cornflowerblue")
(set-face-foreground 'sgml-start-tag-face "slateblue")
;(set-face-background 'sgml-start-tag-face "cornflowerblue")
(set-face-foreground 'sgml-end-tag-face "slateblue")
;(set-face-background 'sgml-end-tag-face "cornflowerblue")
(set-face-foreground 'sgml-entity-face "lavender")
;(set-face-background 'sgml-entity-face "cornflowerblue")
(set-face-foreground 'sgml-doctype-face "lavender")
;(set-face-background 'sgml-doctype-face "cornflowerblue")
(set-face-foreground 'sgml-ignored-face "cornflowerblue")
;(set-face-background 'sgml-ignored-face "cornflowerblue")
(set-face-foreground 'sgml-ms-start-face "coral")
;(set-face-background 'sgml-ms-start-face "cornflowerblue")
(set-face-foreground 'sgml-ms-end-face "coral")
;(set-face-background 'sgml-ms-end-face "cornflowerblue")
```

```
(set-face-foreground 'sgml-pi-face "coral")
;(set-face-background 'sgml-pi-face "cornflowerblue")
(set-face-foreground 'sgml-sgml-face "coral")
;(set-face-background 'sgml-sgml-face "cornflowerblue")
(set-face-foreground 'sgml-shortref-face "coral")
;(set-face-background 'sgml-shortref-face "cornflowerblue")

;; asignar fuentes a categorías de marcado

(setq sgml-markup-faces '
  (
  (comment . sgml-comment-face)
  (start-tag . sgml-start-tag-face)
  (end-tag . sgml-end-tag-face)
  (entity . sgml-entity-face)
  (doctype . sgml-doctype-face)
  (ignored . sgml-ignored-face)
  (ms-start . sgml-ms-start-face)
  (ms-end . sgml-ms-end-face)
  (pi . sgml-pi-face)
  (sgml . sgml-sgml-face)
  (shortref . sgml-shortref-face)
  ))

;; le decimos a PSGML que advierta la nueva configu-
ración de fuentes
(setq sgml-set-face t)
;; ... acabada la configuración del modo psgml
;; ******************************************************************
```

Ahora reinicie Emacs.

4.4.3. Prueba de fuego de SGML

Intente la siguiente prueba de fuego. Cree un nuevo archivo, */tmp/test.sgml*, por ejemplo, e introduzca lo siguiente:

```
<!DOCTYPE test [
<!ELEMENT test - - (#PCDATA)>
]>
```

Introduzca **C-cC-p**. Si Emacs consigue analizar su DTD, aparecerá *Parsing prolog...done* en el minibuffer. Utilice *C-c C-e RETURN* para insertar un elemento *<test>* . Si todofunciona correctamente, debería ver en Emacs:

```
<!DOCTYPE test [
<!ELEMENT test - - (#PCDATA)>
]>
<test></test>
```

4.4.4. Escribir un nuevo CÓMO en Docbook

Cree un nuevo archivo para su CÓMO e introduzca lo siguiente:

```
<!DOCTYPE ARTICLE PUBLIC "-//OASIS//DTD DocBook V3.1//EN">
```

Introduzca **C-c C-p** y contenga la respiración. Si todo funciona correctamente, verá a Emacs rumiando durante unos segundos y acto seguido, *Parsing prolog...done* en el minibuffer.

Finalmente, introduzca **C-c C-e RETURN** para insertar un elemento *<article>* y proceda a escribir su CÓMO.

4.4.5. Referencia rápida para Emacs con PSGML

Vea el manual básico de Nik Clayton’s incluido en la documentación de FreeBSD: http://www.freebsd.org/tutorials/docproj-primer/psgml-mode.html

5. Pautas de estilo

Esta sección contiene algunas indicaciones sobre las pautas que LDP ha acordado establecer para dotar a todos sus documentos de un aspecto uniforme, y que todo autor ha de tener en cuenta.

5.1. Formatos de fechas

La etiqueta <pubdate> del encabezamiento debe tener el siguiente formato:

```
v1.0, 21 de abril de 2000
```

5.2. Formatos gráficos

Cuando envíe imágenes a LDP, remita un grupo en formato .eps y otro en formatos .gif o .jpg. Con .gif obtendrá imágenes algo mejores que con .jpg, pero tendrá que tener en cuenta cuestiones en materia de patentes.

5.3. Versiones de DocBook

Actualmente, LDP sólo trabaja con la versión Docbook 3.1, aunque se está estudiando la posibilidad de incorporar DocBook 4.0. Muchos documentos de 3.1 pueden convertirse fácilmente a 4.0 evitando el uso de etiquetas obsoletas.

Debe escribir el encabezamiento de DocBook de la siguiente manera:

```
<!doctype article public "-//OASIS//DTD DocBook V3.1//EN">
```

5.4. Etiquetas Obsoletas

Está desaconsejado el uso de las etiquetas consideradas obsoletas en *DocBook: The Definitive Guide* para la documentación de LDP. La sección "Tips and Tricks" de este libro informa sobre el uso de nuevas etiquetas.

5.5. Minimización de etiquetas

A fin de minimizar etiquetas, se está utilizando </> para indicar el final de una etiqueta en lugar de la forma completa (por ejemplo</para>). No obstante, esto hace el documento más confuso para futuros autores, y no es posible en DocBook XML. Por tanto, procure evitar esta práctica.

5.6. Convenciones

Las siguientes convenciones son válidas para varios tipos de texto:

Si va a mostrar el uso de una orden, dele un formato similar a las líneas de órdenes de usuario. La línea de órdenes debe contener el tipo de shell (bash, tcsh, zsh,etc) seguido de un símbolo $ para las órdenes que hayan de ejecutarse como usuario normal (no como root), o un signo # para los que se ejecuten como root.

Por tanto, una orden debe tener el siguiente aspecto:

```
bash$ orden "me ejecuto como usuario normal"
bash# orden "me ejecuto como superusuario root"
tcsh# setenv DISPLAY :0.0
```

6. Trucos y consejos para DocBook

En esta sección puede encontrar algunas "rarezas" de DocBook con las que puede tropezar al redactar documentos.

6.1. Inclusión de imágenes

Mientras que LinuxDoc no permitía la inclusión de imágenes en los CÓMOs, Docbook ofrece esta posibilidad. A continuación se muestra una manera sencilla de hacerlo:

```
<figure>
<title>LyX screen shot</title>
<mediaobject>
<imageobject><imagedata fileref="lyx_screenshot.eps" for-
mat="eps"></imageobject>
<imageobject><imagedata fileref="lyx_screenshot.jpg" for-
mat="jpg"></imageobject>
<textobject><phrase>Screen shot of the LyX document process-
ing program</phrase></textobject>
</mediaobject>
</figure>
```

Este procedimiento es mejor que <graphic> por dos motivos. En primer lugar, en DocBook 5.0 se sustituye la etiqueta <graphic> por la etiqueta <mediaobject> . Por esto, quizá sea mejor comenzar bien desde un principio. En segundo lugar, <mediaobject> permite diferentes tipos de formato en función de la salida. En el ejemplo, la primera etiqueta <imageobject> es un fichero .eps usado con formatos derivados de TeX como DVI, PS, y PDF. La segunda <imageobject>es una imagen JPEG para mostrarse en pantalla, sobre todo para formato HTML. La etiqueta <textobject> se incluye por si la salida no soporta imágenes (TXT). Considérela una etiqueta <alt> .

6.2. Designación de ficheros HTML

Por defecto, cuando se crea un conjunto de ficheros HTML , el procesador de SGML asigna nombres arbitrarios a los ficheros resultantes. Esto puede confundir a los lectores que deseen marcar una página para realizar modificaciones o para saber qué contiene cada uno de los ficheros. Sean cuales fueren sus razones, aquí tiene un modo de designarlos a su voluntad:

En la primera etiqueta <article> (que debería ser la única) incluya un parámetro id y llámelo índice. La etiqueta debería, pues, tener la siguiente forma:

```
<article id="index">
```

No modifique la primera etiqueta <sect1> pues suele ser una introducción que deseará figure en la primera página. Incluya un parámetro id en el resto de las etiquetas <sect> , y dele un nombre. Los nombres deberían constar únicamente de caracteres alfanuméricos y ser representativos de su contenido.

```
<sect1 id="tips">
```

6.3. Uso de ldp.dsl

El LDP utiliza su propio fichero DSSSL, que permite, por ejemplo, añadir un fondo blanco y generar automáticamente la tabla de contenidos que puede ver al comienzo de los CÓMOs. Puede encontrar la última versión de este fichero en http://metalab.unc.edu/gferg/ldp/ldp.dsl.

Una vez tenga el fichero, puede que deba realizar alguna labor de edición de las primeras líneas, en relación con la ubicación de sus ficheros DSSSL de DocBook. En el ejemplo, se utiliza el conjunto de utilidades Cygnus.

Almacene el fichero `ldp.dsl` en `/usr/lib/sgml/stylesheets` y ábralo con su editor de texto favorito. Debería ver algo así:

```
<!DOCTYPE style-sheet PUBLIC "-
//James Clark//DTD DSSSL Style Sheet//EN" [
<!ENTITY % html "IGNORE">
<![%html;[ <!ENTITY % print "IGNORE">
<!ENTITY docbook.dsl SYSTEM "docbook.dsl❶" CDATA dsssl>
]]>
<!ENTITY % print "INCLUDE">
```

```
<![%print;[
<!ENTITY docbook.dsl SYSTEM "docbook.dsl❷" CDATA dsssl>
]]>
]>
```

❶ Cambie el primer "docbook.dsl" y escriba `/usr/lib/sgml/stylesheets/nwalsh-modular/html/docbook.dsl`

❷ Cambie el segundo "docbook.dsl" y escriba `/usr/lib/sgml/stylesheets/nwalsh-modular/print/docbook.dsl`

Si utiliza otro DSSSL, especifique para estos ficheros la ubicación de los ficheros HTML y print en ese DSSSL. Normalmente se encuentran en directorios llamados "html" y "print".

Cuando termine, podrá generar ficheros HTML:

```
bash$ mkdir HOWTO-HOWTO ; cd HOWTO-HOWTO
bash$ jade -t sgml -ihtml -
d /usr/lib/sgml/stylesheets/ldp.dsl\#html
../HOWTO-HOWTO.sgml
```

La primera orden crea un directorio en el que almacenar los ficheros. La segunda orden (la orden jade) genera ficheros HTML individuales para cada sección de su documento. Si va a utilizar algo como RTF, puede hacer lo siguiente:

```
bash$ jade -t rtf -d /usr/lib/sgml/stylesheets/ldp.dsl ../HOWTO-
HOWTO.sgml
```

7. CVS

El LDP se propone ofrecer acceso CVS a sus autores. Buenas razones para ello serían las siguientes:

1. CVS mantendrá una copia de respaldo local de sus documentos. En caso de que desee enviar un documento a otro autor, éste podrá simplemente recuperarla en CVS y trabajar con ella. Si necesita volver a una versión anterior del documento, también podrá obtenerla en CVS.
2. Es una herramienta magnífica para un grupo de personas que trabaja en el mismo documento. CVS le dirá qué cambios ha hecho otro autor mientras Usted editaba su versión y le permitirá integrar dichos cambios.
3. Guarda un registro de todos los cambios realizados. Estos registros (y las fechas correspondientes) pueden incorporarse automáticamente en el documento haciendo uso de ciertas etiquetas especiales procesadas antes que el procesador SGML.
4. Proporciona medios para que un programa actualice automáticamente el sitio web de LDP, incorporando nueva información a medida que ésta se redacta y se envía. Aún no disponemos de esta opción, pero es uno de nuestros objetivos. Por el momento, las actualizaciones de CVS van indicando al coordinador de los CÓMO que debe actualizar el sitio web, por lo que si utiliza CVS, no tendrá que enviar un e-mail con su código SGML.

Si está utilizando CVS por primera vez, las siguientes páginas web le serán de gran ayuda:

- http://www.sourcegear.com/CVS/Docs/blandy
- https://wroclaw.art.pl/~ser/docs/cvs.html

7.1. Creación de una cuenta en CVS

En primer lugar, tendrá que abrir una cuenta en el repositorio CVS de LDP. Este es el directorio raíz que CVS utiliza para diferentes proyectos (CÓMOs, mini CÓMOs, etc), que a su vez constituyen subdirectorios de él.

Para esto tendrá que crear una contraseña con hash y un nombre de usuario. El hash de la contraseña le permitirá enviar una contraseña cifrada que el grupo CVS no conocerá. Podrá efectuar esto, haciendo uso de la orden siguiente (desde bash o sh):

```
bash$ echo your_password | perl -
e "print crypt(<>,\ join ”,('.', '/', 0..9, 'A'..'Z', 'a'..'z')[rand 64, ranc
```

Envíe la salida de esta orden con su nombre de usuario a: <cvsadmin@cvslist.linuxdoc.org>. De este modo, se creará su propio directorio CVSROOT, y recibirá una respuesta por e-mail. Cuando reciba esta respuesta, acceda a su CVSROOT y asegúrese de que todo está configurado correctamente:

```
bash$ export CVS-
ROOT=:pserver:su_identificador@cvs.linuxdoc.org:/cvsroot
bash$ cvs -d $CVSROOT login
```

(Sustituya *su_identificador* con lo que se le envió en el correo de respuesta).

Se le pedirá que especifique su contraseña, y, posteriormente, se le dará acceso en modo de lectura-escritura al repositorio de CVS. Una vez haya accedido al sistema mediante **cvs login**, su contraseña se almacenará en `.cvsroot` y no tendrá que volver a utilizar **cvs login**. Configure CVSROOT y continúe. Puede acceder al repositorio completo de linuxdoc mediante la siguiente orden:

```
bash$ cvs get LDP
```

También puede acceder al fuente de SGML de su documento mediante las órdenes:

```
bash$ cvs get howto/YOUR-HOWTO.sgml
bash$ cvs get minihowto/YOURDOC.sgml
```

7.2. Información de interés sobre el repositorio de CVS

7.2.1. Acceso anónimo a CVS

Si no necesita abrir una cuenta en CVS (por ejemplo, si lo que desea es publicar documentación del LDP), puede acceder anónimamente al repositorio en modo lectura. Para ello, deberá utilizar la siguiente orden:

```
bash$ cvs -d :pserver:cvs@anoncvs.linuxdoc.org:/cvsroot login
```

Utilice cvs como contraseña. Puede acceder a los módulos de linuxdoc como se ha mencionado anteriormente. Tenga en cuenta que los cambios en el sitio anoncvs pueden retrasarse una media hora con respecto al sitio principal.

7.2.2. Ficheros de CVS via Internet

Puede acceder al repositorio de CVS a través de la red, en http://cvsweb.linuxdoc.org/index.cgi/linuxdoc.

7.2.3. Acceso gráfico a CVS

Podrá obtener una lista de interfaces gráficas para CVS en http://freshmeat.net/appindex. Busque CVS.

7.3. Actualización de ficheros y CVS

CVS dispone de una etiqueta especial, *$Id*, que le permite insertar automática y directamente la fecha y la versión en el documento. Al utilizarla, CVS la convertirá en *$Id: HOWTO-HOWTO.sgml,v 1.4 2000/06/12 20:49:54 markk Exp $*. Al incluirse esta etiqueta en un documento, aquella información se actualizará cada vez que se modifique el fichero, incrementándose, asimismo, la marca de revisión.

Cuando esté listo para enviar los cambios al servidor CVS, utilice la orden **cvs ci -m "comentario" SU-CÓMO.sgml**. El -m "comentario" no es necesario, pero si no lo incluye, se abrirá el editor (normalmente, vi, o el especificado en su variable de entorno EDITOR) y se le dará la oportunidad de añadir comentarios sobre los cambios.

En la lista ldp-discuss, podrá estar al tanto de cuestiones relacionadas con CVS. Por ahora, los envíos a LDP deberán remitirse a `<ldp-submit@lists.linuxdoc.org>`.

8. Distribución de la documentación

8.1. Antes de la distribución

Antes de distribuir su código a millones de lectores potenciales, hay unas cuantas cosas que debería hacer.

Primero, asegúrese de corregir la ortografía de su documento. La mayoría de las utilidades que empleará para escribir en SGML poseen aplicaciones para la corrección ortográfica. Si no es así, siempre está el programa aspell.

Segundo, pida a alguien que revise el documento para verificar su exactitud. La documentación publicada por el LDP debe ser lo más precisa posible en relación con los hechos, ya que millones de usuarios de Linux pueden estar leyéndola. Si participa en alguna lista de correo sobre la materia en cuestión, pida a alguien de la lista que le eche una mano.

Tercero, cree un sitio web donde pueda distribuir su documentación. Esto no es estrictamente necesario, pero es útil para que algunas personas encuentren la ubicación del original de su documentación.

8.1.1. Validación del código SGML

Usando Jade, o cualqier otra orden nsgmls, Usted puede validar su código .sgml contra la DTD para asegurarse de que no existe ningún error.

```
bash$ nsgmls -s HOWTO-HOWTO.sgml
```

Si no hay problemas, se le devolverá a su línea de órdenes.

8.2. Copyright y licencias

Para que un documento sea aceptado por el LDP, debe ajustarse a la sección "LICENSE

REQUIREMENTS" del Manifiesto del LDP, que se encuentra en http://www.linuxdoc.org/manifesto.html. Como autor, puede conservar el copyright y añadir ciertas restricciones (por ejemplo, puede dar el visto bueno a una traducción u otros trabajos derivados). Dispone de un ejemplo en el Manifiesto o en http://www.linuxdoc.org/COPYRIGHT.html. Si elige emplear el copyright estándar, simplemente cópielo en su código fuente en una sección llamada "Copyright y Licencias" o algo similar. Incluya también su propia declaración de copyright (¡es suyo!). Si Usted colabora en un CÓMO ya existente, debe incluir las declaraciones de copyright del autor o autores previos y las fechas en las que trabajaron en ese documento.

Observe que la licencia del Cómo-de-CÓmos obliga a notificar al autor cualquier traducción o trabajo derivado de éste. En mi caso, también coloco explícitamente cualquier código fuente (además del SGML en el que se escribió el CÓMO) bajo licencia GPL. Debería hacer lo mismo, si su CÓMO incluye trozos de código fuente que quiera poner a disposición de los demás usuarios.

8.3. Presentación en LDP

Una vez que su documento ha sido revisado cuidadosamente, puede publicarlo en el LDP. Envíe un e-mail con el código fuente en SGML como un archivo adjunto (utilice gzip si lo desea) a <ldp-submit@lists.linuxdoc.org>.

Asegúrese de incluir el nombre de su CÓMO en la línea del asunto, utilice el cuerpo para explicar los cambios que ha realizado y adjunte su CÓMO. Esto permite a los colaboradores realizar su trabajo con mayor rapidez, de manera que no tiene que esperar mucho para ver su CÓMO actualizado en el sitio web del LDP. Si no tiene noticias en 7 días, le rogamos que se ponga en contacto con nosotros via e-mail para asegurarse de que las cosas están todavía en proceso.

Si su CÓMO contiene extras, imágenes o un catálogo especial, cree un fichero .tar.gz que contenga todos los ficheros, incluyendo el código fuente .sgml, y envíelo como un archivo adjunto a la lista ldp-submit.

8.4. Mantenimiento del CÓMO

Ahora que Usted es autor de un CÓMO, debería mantener el documento y actualizarlo cuando se hagan públicas las nuevas versiones de software. También debería responder a los comentarios y preguntas razonables de sus lectores. No tiene que ayudarlos a todos, especialmente si sus preguntas se responden en el CÓMO. Sin embargo, una buena experiencia con el LDP por parte de los lectores es uno de nuestros objetivos y una excelente manera de aumentar la popularidad de Linux.

9. PUFs (Preguntas de Uso Frecuente) sobre el LDP

9.1. Quiero ayudar en el LDP. ¿Cómo puedo hacerlo?

La forma más fácil es encontrar algo y documentarlo. También puede echar un vistazo entre los CÓMOs sin mantener: quizás encuentra algún tema que domine y sobre el cual pueda seguir documentando.

9.2. Deseo publicar una colección de documentos del LDP en un libro. ¿Cómo es la licencia de los contenidos del LDP?

Por favor, vaya a http://www.linuxdoc.org/COPYRIGHT.html. Tenga en cuenta que está es solo una guía para los autores.Sin embargo, la licencia no puede ser más restrictiva que lo que se lista en esta dirección.

9.3. Encontré un error en un documento del LDP. ¿Puedo arreglarlo?

Contacte con el autor del documento, o con el coordinador del LDP en <ldp-discuss@lists.linuxdoc.org> y cuéntele el problema y cómo piensa que debe arreglarse.

9.4. ¡Pero no sé SGML/No puedo conseguir que funcionen las herramientas/No me gusta SGML!

No importa. Tiene la opción de escribir su primer borrador del CÓMO en el formato que elija, y enviárselo así al LDP. Un voluntario del proyecto revisará el documento, y luego lo convertirá a DocBook por Usted. Una vez hecho esto, le será más fácil encargarse del mantenimiento del CÓMO. Si tiene alguna pregunta, siempre puede dejarla caer al voluntario del LDP o a la lista sobre Docbook del LDP en <ldp-docbook@lists.linuxdoc.org>.