

Edición electrónica: ¿con qué formato?

Hípola, Pedro. «Edición electrónica: ¿con qué formato?». En: *Information world en español*, n° 31, febrero 1995, pp. 1-8.

Están proliferando diversos formatos para llevar a cabo la edición de documentos en soporte legible por máquina. La carrera hacia la hegemonía en el control del "documento electrónico" (v. [IWE-25](#), p. 1) sólo acaba de empezar. Cada empresa o grupo de empresas presenta sus propias normas. ¿Cuál(es) triunfará(n)? En este informe se analizan las características técnicas de los principales formatos y estándares disponibles para poner en el mercado productos de información electrónica.

Cada vez es mayor la cantidad de "documentos" que se editan en soporte legible por ordenador. Y, gracias a la "revolución multimedia", éstos pueden contener información textual-numérica, gráficos, imágenes (fijas o en movimiento) y/o sonido.

Pero puede suceder fácilmente que los diversos productos de información electrónica que se elaboran queden *aislados* del resto por no estar ajustados a normas que faciliten la gestión homogénea de productos heterogéneos. De esta forma se puede producir la paradójica situación de que aumenten los productos y servicios de información electrónica, pero no crezcan en la misma proporción las facilidades para que el público en su conjunto acceda a todos ellos.

En busca de la norma ideal

Como en tantos otros ámbitos de la industria, del comercio, etc., dentro del mundo de la edición electrónica es importante poder contar con *normas* que faciliten el trabajo de todos los sectores implicados en el proceso, en este caso los autores, editores, distribuidores, fabricantes de hardware y de software, usuarios...

Tradicionalmente la mayor parte de las normas disponibles para la gestión de información electrónica servía para resolver problemas informáticos en su más "bajo nivel": compatibilidad de hardware, sistemas de codificación de caracteres, normas para las señales físicas en telecomunicaciones, etc.

Tal nivel de normalización resulta imprescindible. En este sentido la aceptación general de normas como el juego de caracteres *Ascii* (*American standard code for information interchange*) y, posteriormente, *Unicode*, ha supuesto sin duda un avance considerable para facilitar la transferencia y el acceso a la información en soporte digital.

Sin embargo, es interesante observar la creciente importancia que van adquiriendo nuevas "familias" de normas aparecidas en los últimos años, destinadas así mismo a facilitar el intercambio de información electrónica de forma abierta, no sujeta a las especificaciones de un solo fabricante, pero todo ello a un nivel *superior*.

Algunas de estas normas, las menos pretenciosas, tratan de homogeneizar el formato y la estructura de los documentos que incluyen únicamente caracteres alfanuméricos. Otras, más completas, abarcan todas las posibilidades de un entorno multimedia.

Pensemos, por ejemplo, en las normas del *Comité Consultivo Internacional de Teléfonos y Telégrafos (Ccitt)* para la representación de imágenes facsímiles, llamadas *FAX*, que son utilizadas frecuentemente para la elaboración de productos de edición electrónica, como es el caso de algunos discos cd-rom que incluyen imágenes digitalizadas.

En un nivel similar se encuentran otros estándares para la representación de ficheros gráficos que son de uso común en el mundo informático: *Tiff*, *PCX*, *Jpeg*, etc.; los que se utilizan para codificar imágenes en movimiento: *Mpeg*; o sonido: *Midi*, *Wave*, etc.

Su uso tan extendido contrasta, sin embargo, con el bajo nivel en el que se sitúa el esfuerzo normalizador. La información codificada no permite ni la más mínima estructuración de los datos.

En los últimos años, sin embargo, están proliferando diversos sistemas encaminados a facilitar la normalización del "documento electrónico". En los siguientes párrafos nos ocuparemos de estos sistemas, agrupándolos según las *tendencias* que observamos entre quienes están participando en el proceso normalizador (1).

Nuestro objetivo no es hacer una exposición lineal analizando producto a producto, sino presentar las principales líneas de trabajo existentes. Por eso un mismo producto o estándar puede aparecer mencionado en apartados correspondientes a distintas tendencias.

La opción "semántica"

En el mundo bibliotecario y documental es suficientemente conocido el estándar *ISO 2709*, la norma marco de las diversas versiones del formato Marc. Aunque no es una estructura específicamente diseñada para lo que normalmente se entiende por edición electrónica, mencionamos aquí este estándar porque nos parece exponente de una de las filosofías que pueden ser utilizadas a la hora de normalizar el "documento electrónico": asignar a cada elemento del documento un *valor significativo* predeterminado.

Cuando un programa de bibliotecas analiza un registro Marc, puede reconocer el "valor semántico" del contenido de cada campo según cuál es el *designador de contenido* que lo identifica. En este caso, el sistema de *designadores* es la clave para estructurar los elementos según diversas unidades funcionales: el designador indica el valor semántico de las partes.

Sin duda este método supone una importante ayuda si lo que buscamos es llevar a cabo la representación de elementos documentales suficientemente estructurados.

El "rey" de la edición electrónica: *Sgml*

El representante por excelencia de esta filosofía para la estructuración de documentos es el lenguaje *Sgml* (*Standard generalised mark-up language*), que por otra parte viene siendo la norma más extendida dentro del mundo de la edición electrónica. Cada vez son más los textos (para uso interno, o disponibles online, en cd-rom, etc.) que son preparados con

codificación *Sgml*.

Con *Sgml* es posible crear un fichero fácilmente transferible y procesable por un amplio abanico de sistemas, ordenadores, plataformas y medios.

El lenguaje utiliza un conjunto de códigos que indican la estructura del documento. Sus diversos elementos (título principal, títulos secundarios, notas de pie de página, etc.) reciben una serie de codificaciones específicas. Esto hace que cada documento sea como un registro con sus respectivos campos.

En 1986 *Sgml* fue aceptado como norma internacional: *ISO 8879*. Lo que constituye la norma ISO no es el conjunto de etiquetas en sí sino la metodología.

Para crear un documento *Sgml* es necesario hacer uso de una *Definición de Tipo de Documento (DTD)*, que especifica qué elementos pueden aparecer en el documento y qué reglas determinarán su presencia o ausencia: por ejemplo, cuáles son obligatorios y cuáles no.

Cualquier organización puede crear y difundir un *DTD*. Entre los más famosos se encuentra el de la iniciativa *Cals (Computer-aided acquisition and logistic support)*, del *Departamento de Defensa* norteamericano.

Con cualquier procesador de textos es posible editar un fichero Ascii que incluya etiquetas *Sgml*. Como esto puede resultar laborioso, existen editores (software) especiales para crear y manipular ficheros *Sgml*. Pero incluso los procesadores de texto más populares (como es el caso de *WordPerfect* y de *Word*) han comenzado a dar soporte al formato.

Sgml supone un gran avance para los editores, ya que no sólo ahorra tiempo y reduce costos de producción (no es necesario volver a teclear una y otra vez los escritos), sino que el mismo original puede ser utilizado tanto para productos electrónicos como para documentos que se van a imprimir en papel.

Existe una *DTD* específica de *Sgml* utilizada por los sistemas *WWW (World Wide Web)*, denominada *Html (HiperText Mark-up Language)*, que también está teniendo mucho éxito (2) y a la que nos referimos en los últimos párrafos de este artículo.

***Sgml*: el éxito que viene**

Según *InterConsult Inc.*, el mercado de software específico para la utilización de ficheros *Sgml* supuso un volumen de negocio de 520 millones US\$ durante 1993, y alcanzará la cifra de 1.460 millones (casi el triple) en 1998.

Si comparamos estas cifras con el número de documentos que conocemos en formato *Sgml*, alguien puede pensar que estamos ante uno de los trabajos menos productivos del mundo.

¿A qué se debe esto?

Por una parte, muchos usuarios del formato trabajan en proyectos militares y guardan una comprensible discreción sobre los documentos que producen.

Además -y éste es ya un motivo más comercial- muchos productos están aún siendo evaluados por potenciales usuarios, que evitan expresar sus opiniones en público para evitar que tales informaciones puedan llegar a la competencia.

Según declara **Janet Waplington**, de *XSoft*, "no puedo decir quiénes, pero una serie de empresas de sistemas de comunicación, para las que *Sgml* es muy adecuado, están usando nuestros productos. También parece que la industria farmacéutica está analizando muy seriamente *Sgml*, ya que ellos tienen una gran cantidad de información estructurada que es

revisada y buscada constantemente".

"Pienso que pasarán 2 ó 3 años antes de que *Sgml* tenga una influencia real, pues no hay presión para que se implemente. Lo que yo noto ahora, sin embargo, es que crece el número de seminarios sobre *Sgml* y el número de personas que acuden a ellos".

EDI: documentos electrónicos para la gestión

Dentro de los sistemas que han optado por la caracterización semántica, merecen una mención especial los *sistemas EDI* (3).

Aunque no están dirigidos específicamente al mundo de la edición electrónica, vale la pena hacer una breve referencia a ellos por ser un ejemplo claro de lo que hemos denominado la "opción semántica", también por su éxito y porque pueden servir de modelo para desarrollar métodos de normalización documental (4).

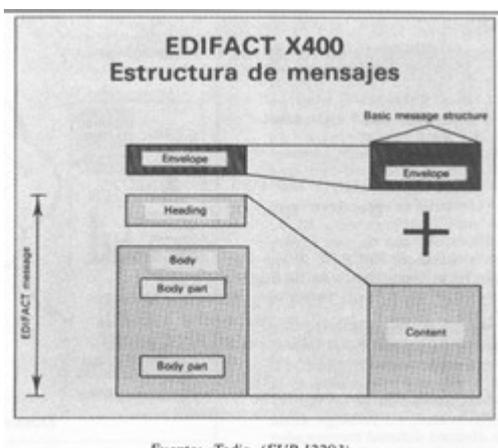
Según la definición contenida en las reglas *Unacid* (*Uniform rules of conduct for interchange of trade data by teletransmission*), publicadas en 1988 por la *International Chamber of Commerce (ICC)*, EDI es "la transferencia directa entre ordenadores, a través de medios electrónicos, de datos de negocios estructurados, esto es, la transferencia de 'documentación' comercial sin papeles".

En efecto, los sistemas EDI tratan de hacer posible el intercambio de pedidos, facturas, acuses de recibo, etc., entre los ordenadores de todo tipo de organizaciones.

Con ello se facilita la automatización de determinadas tareas, eliminando esfuerzo humano, recursos económicos, tiempo, etc.

Estos sistemas están siendo utilizados en diversos sectores del comercio, la industria, el turismo, medios de transporte y en algunas áreas de la Administración.

El esfuerzo normalizador de EDI se centra en la definición de estructuras de datos que no contengan ambigüedades. Aparte de una gran variedad de sistemas "propietarios" (ligados a marcas de ordenadores concretas) incompatibles entre sí, las dos normas más extendidas son *Edifact* en Europa y *Ansi X.12* en Estados Unidos. En el *American National Standards Institute (Ansi)* se ha estado trabajando para compatibilizar ambas normas.



Edifact está compuesto por:

1. el vocabulario Tded (trade data elements directory), articulado en las normas *ISO 7372* y *EN 29732*, que incluye seiscientos conceptos usuales en transacciones

- comerciales, logística y trámites administrativos;
2. un conjunto de reglas de sintaxis al nivel de aplicación de OSI (Open systems interconnection), recogidas en las normas *ISO 9735* y *EN 29735*, y que estructuran el formateo de los mensajes en datos y segmentos;
 3. un directorio de segmentos normalizados: nombre, dirección, tipo de intercambio, etc.;
 4. un conjunto de mensajes normalizados: orden de pedido, factura, aviso de envío, declaración de aduanas, etc.; y
 5. un conjunto de códigos: países, monedas, modos de transporte, condiciones de pago, etc.
 - 6.

Los documentos compuestos "estáticos"

Hasta aquí nos hemos referido a la "opción semántica". Otra tendencia la podemos observar en ciertos sistemas que permiten manipular documentos *compuestos*. Es decir, documentos que, además de materiales textuales, contienen también tablas, gráficos e imágenes.

Cuando estos documentos son gestionados por determinado tipo de software, obtenemos como resultado final una presentación, en la pantalla del ordenador o impresa en papel, tan cuidada como cualquier material impreso tradicional: páginas completamente maquetadas, con la variedad tipográfica necesaria, etc.

La posibilidad de gestionar documentos compuestos está prevista en *Sgml*, y en todos aquellos sistemas que generan o gestionan *documentos compuestos "estáticos"*. Este tipo de documentos, también denominados "pasivos" (5), son los que integran sus distintos elementos perfectamente organizados de cara a poder obtener una presentación formal definitiva, en la pantalla o en el papel. Además pueden estar identificadas las relaciones lógicas entre sus diversas partes: gráficos, notas de pie de página, etc.

Entre estos sistemas podríamos incluir los *lenguajes de descripción de páginas (PDL)*, el más importante de los cuales es *PostScript*, de *Adobe*, y formatos "enriquecidos", como *TEX* (utilizado, por ejemplo, con el editor *LaTEX*) y *RTF (Microsoft)*.

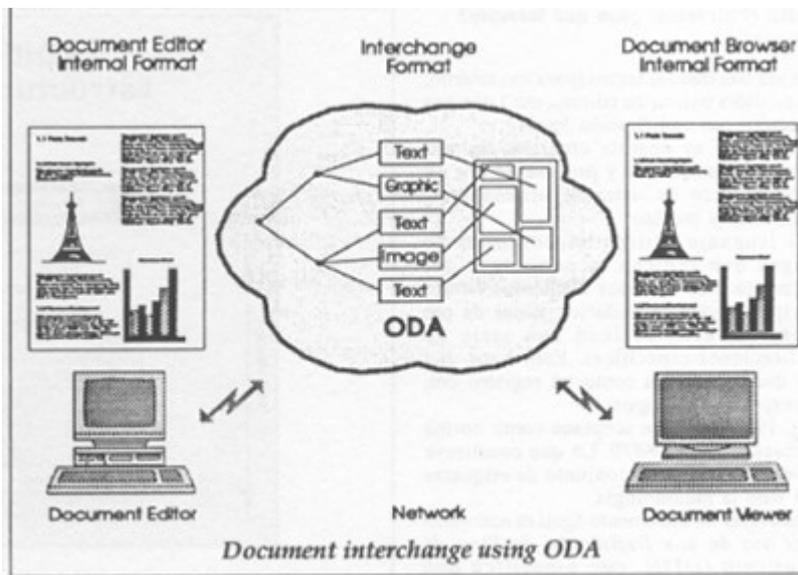
Otros sistemas, más sofisticados, están concebidos para facilitar el *intercambio* de ficheros compuestos. En este grupo hay que citar la norma *ODA*, *PDF-Acrobat (Adobe)*, *Replica (Farallon Computing)*, *Common Ground (No Hands)*, *Envoy (Novell-WordPerfect)*, *Bento (Apple)* y *CDA (Digital)*.

La norma ODA

ODA (Open document architecture) es un estándar (*ISO 8613*) aún poco utilizado, que comenzó a elaborarse en 1982. Entonces la "O" inicial correspondía a *Office*, y su nombre completo era *Oda/Odif (Office document architecture/Office document interchange format)*.

Su objetivo es conseguir que "los documentos, tanto textuales como gráficos, puedan ser transferidos, con todos sus atributos intactos, de un sistema a otro, para que luego puedan ser editados, procesados, almacenados, impresos y transmitidos".

Como decíamos antes, los formatos electrónicos más sencillos usados para el intercambio y visualización de materiales gráficos digitalizados son sólo sistemas para "almacenar" imágenes de páginas que han sido previamente escaneadas o dibujadas. No dan soporte a mayores complejidades.



Por otra parte, aunque los sistemas actuales de tratamiento de textos y autoedición ofrecen facilidades para importar y exportar ficheros, sin embargo, en cuanto éstos son medianamente complejos, porque incluyen gráficos, tablas, etc., los problemas resultan insolubles. Al traspasar los documentos de un sistema a otro muchas veces las tablas se trastocan, se pierde la composición de las páginas que habían sido maquetadas con cierta complejidad...

En definitiva, lo que sucede es que la diversidad de los sistemas "propietarios" (propios de un solo fabricante) muy frecuentemente dificulta el intercambio de documentos complejos, de manera que para el sistema que los recibe no resulta posible presentarlos en pantalla y gestionarlos como si él los hubiera creado.

ODA busca solucionar este problema ofreciendo todo un marco global para la representación de documentos. Es capaz de representar, dentro de un mismo documento, materiales textuales y gráficos. Consta de una estructura física en bloques -título, párrafos, índice de materias, palabras clave, referencias externas, resumen, notas a pie de página...- y de una estructura lógica -páginas, cuadros, etc.-.

La norma ha recibido un mayor impulso desde 1991, fecha en la que se creó el *ODA Consortium*, constituido por *Bull*, *DEC*, *IBM*, *ICL*, *Siemens Nixdorf* y *Unisys*. Después se agregaron otros, como *WordPerfect*.

El *Consortio* ha desarrollado un "toolkit" (conjunto de herramientas software), que se distribuye gratuitamente, el cual transforma documentos al formato de intercambio *Odif* (*Open document interchange format*), para que a continuación se puedan transferir de nuevo esos documentos a cualquier otro sistema que dé soporte a *ODA*.

El estándar está encontrando muchas dificultades para ganar terreno. Por una parte, los fabricantes de software de tratamiento de textos centran hoy más su atención en dar soporte a *Sgml*. Aunque en realidad no se trata de dos normas en competencia, sino que son complementarias, de hecho una está acaparando la atención de todos, mientras que la otra va cayendo en el olvido.

Además, el *Consortio* tiene que afrontar la aparición de más y más versiones de procesadores de textos comerciales. Y faltan, por otra parte, editores y visualizadores para los documentos *ODA*...

Sea por lo que fuere, *ODA* no termina de despegar. **Dallas Powell** -que en *WordPerfect* se responsabiliza de los desarrollos para dar soporte a *ODA* y *Sgml*, así como a su propio producto *Envoy*- asegura que *WordPerfect* ha invertido mucho dinero en su compromiso con

ODA, sin que los resultados hayan sido satisfactorios. "Yo creo ahora que la única manera de que *ODA* alcance sus objetivos es que las autoridades federales norteamericanas y la Comisión Europea ordenen que la norma se use en todos los documentos que ellos manejan. El Departamento de Defensa hizo precisamente eso con la iniciativa *Cals*, y así es como *Sgml* ha sido aceptado (...). Sin el Departamento de Defensa, *Sgml* nunca habría despegado, por muy buena que hubiera sido la especificación técnica" (6).

La idea de **Powell** es buena, pero el gobierno americano distribuyó su presupuesto para el año 1994... ¡en formato *Acrobat*!

Así las cosas, ante el posible fracaso de *ODA*, las empresas especializadas en software para la gestión de documentos electrónicos están enzarzadas en una carrera para ver quién consigue imponer como norma su propio formato de intercambio.

La estrategia de *Adobe*



Adobe Systems Inc. es una compañía centrada en el desarrollo de herramientas software para la gestión de documentos electrónicos. Fue quien lanzó en 1985 el lenguaje de descripción de páginas *PostScript*, que ha sido hasta ahora el principal estándar de facto para el intercambio de documentos electrónicos ya formateados.

Este lenguaje de descripción de páginas ha servido fundamentalmente para que diversos sistemas informáticos puedan imprimir los documentos creados por equipos de diferentes fabricantes.

Pero no ha sido ésta la única aportación de la empresa californiana, pues en 1993 presentó *Acrobat*, un formato con el que quiere llegar más lejos, facilitando aún más el intercambio de documentos electrónicos entre sistemas software y hardware que son incompatibles entre sí.

Haciendo uso de *Acrobat* se pueden ver en la pantalla del ordenador -con gran fidelidad a la tipografía, estilo, gráficos y colores del original- documentos que han sido creados en un sistema completamente incompatible.

Acrobat incluye todas las funcionalidades que ofrece su predecesor *PostScript* para gestionar textos con gran riqueza tipográfica, así como gráficos complejos. Además permite crear índices para búsquedas textuales. Y sirve, en definitiva, para que un fichero *PostScript* sea presentado, con todas sus ilustraciones y tipos de caracteres no habituales, en la pantalla de máquinas que antes habrían sido incapaces de ofrecer tales prestaciones.

El formato propio de *Acrobat* se denomina *PDF (Portable Document Format)*, que puede incluir no sólo textos y gráficos, sino también anotaciones, enlaces hipertexto... *PDF* utiliza sólo caracteres Ascii de 7 bits. Así es posible una portabilidad universal, pues no tiene que depender de las variantes de diferentes juegos de caracteres de 8 bits diseñados para usos específicos internacionales.

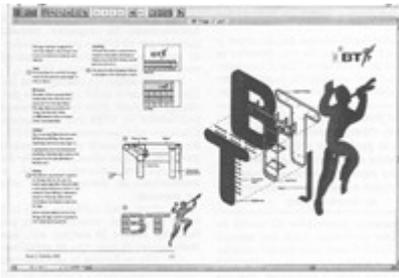


Ilustración 1: Una página del manual de estilo de BT en formato Acrobat PDF

Los ficheros *PDF* se caracterizan por su alto nivel de compresión. La tecnología utilizada aprovecha las técnicas compresoras de la norma *FAX grupo 4*, del *Ccitt*, y las de *Jpeg (Joint photographic experts group)*. En algunos casos lo que antes con *PostScript* ocupaba 80 megas ahora puede caber en un disquete de 3,50".

La versión 2 de *Acrobat*, lanzada al mercado el año pasado, incorporó importantes mejoras. La más importante, quizá, es que se ofrece la posibilidad de establecer enlaces con documentos y aplicaciones externos, utilizando el sistema de *URL (Universal Resource Locator)*, propio de *WWW (7)*. Esta versión se entrega también con un sistema de indexación y recuperación de los textos contenidos en los ficheros PDF.

Adobe está trabajando, con la ayuda de la empresa *Avalanche*, para que *Acrobat* pueda incluir codificación *Sgml*. De esta manera el producto de *Adobe* asumiría las prestaciones que ofrece la norma ISO.



Ilustración 2: Si el sistema no tiene disponibles los tipos de letra, Acrobat los emula

Familia de productos *Acrobat*

- *Acrobat Exchange*: permite generar ficheros en formato PDF a partir de documentos

creados por otras aplicaciones, así como visualizarlos en diversos tamaños, manipularlos, cortar y pegar, hacer "anotaciones electrónicas", búsquedas, navegación... e imprimirlos. Incluye la posibilidad de establecer "enlaces vivos" entre las partes de un fichero PDF. De esta forma se pueden establecer referencias cruzadas. Desde la versión 2 los enlaces pueden ser también a cualquier otro tipo de documento, o a servidores *WWW*, siguiendo en eso el estilo de *Html*.

- *Acrobat Reader*: versión limitada del módulo *Exchange*, que sirve para visualizar los documentos en formato PDF e imprimirlos. No permite crear ficheros PDF, ni manipularlos. Es posible obtener gratuitamente el módulo lector. Se puede encontrar, por ejemplo, en servidores Internet.
- *Acrobat Writer*: controlador que se encarga de la impresión de los ficheros PDF.
- *Acrobat Distiller*: traduce a PDF ficheros estructurados en formato EPS (*PostScript* "encapsulado"). Puede comprimirlos o reducir la resolución de las imágenes. Sin embargo no ofrece la función de visualización ni otras prestaciones de *Exchange*.
- *Acrobat Search*: sirve para realizar búsquedas de textos presentes en ficheros PDF que han sido indexados con *Catalog*.
- *Acrobat Catalog*: crea índices de los textos íntegros contenidos en colecciones de ficheros PDF.

La "competencia" de *Acrobat*

Existen varios productos que en cierta manera compiten con *Acrobat*: *Replica* (de *Farallon Computing*), *Common Ground* (*No Hands*), *Envoy* (*Novell-WordPerfect*) y *Bento* (*Apple*).

Replica está basado en la tecnología *TrueType*, el sistema de caracteres escalables incluido en el *System 7* de *Apple* y en *Windows* a partir de la versión 3.1. En consecuencia, la información de las páginas no se almacena -como tampoco sucede con todos estos productos que se hacen la competencia- en forma de "bit map" (mapa de puntos), sino que el sistema rediseña los juegos de caracteres cada vez que se ha de manipular el documento.

Replica Creator funciona como un driver (controlador) de impresora. Permite recortar y pegar textos formateados y gráficos. Sin embargo, no permite hacer "anotaciones electrónicas". Ofrece, eso sí, ciertas prestaciones de seguridad: encriptación y protección por password.

Replica Viewer, el módulo de visualización, es gratis y puede ser suministrado junto con el documento cuando éste se transfiere de un sistema a otro.

Common Ground organiza la imagen de los documentos utilizando la tecnología *DigitalPaper*, desarrollada por la misma empresa creadora del formato. Describe gráficos y caracteres pixel a pixel. Ofrece la posibilidad, como los productos competidores, de que se entregue un módulo visualizador, el *MiniViewer*, "embebido" en el documento. El sistema completo permite recortar y pegar, así como hacer búsquedas de cadenas de caracteres. Reproduce exactamente el original.

En cuanto a *Envoy*, se ha publicado un análisis extenso en el anterior número de esta revista (8). *Envoy Publisher* se encarga de llevar a cabo la creación de ficheros, actuando también de la misma forma que un driver de impresora. Puede gestionar enlaces hipertexto, pero sólo dentro del mismo documento. *Envoy Viewer* presenta, manipula e imprime los documentos. Por último, *Envoy Run-Time* sirve para distribuir los ficheros con el visualizador embebido.

El producto es muy reciente, y aún es pronto para predecir su posibilidad de éxito. Sin embargo, la posición predominante del procesador de textos *WordPerfect* a nivel mundial

augura una amplia difusión de *Envoy*.

DEC y su CDA

CDA (Compound Document Architecture) es la respuesta de *Digital Equipment Corporation* a estos problemas que estamos analizando. Siguiendo sus especificaciones es posible intercambiar con éxito documentos complejos entre distintas aplicaciones.

Sus reglas de codificación están definidas en el *Ddif (Digital document interchange format)*. Su esquema controla la organización lógica de los contenidos -secciones, capítulos, párrafos...-, la organización física -titulares, columnas, bloques de texto...- y su presentación -tipos de letras...-.

¿Qué formato va a prevalecer?

Resulta aventurado hacer predicciones, pero algunas cosas sí parecen claras. La norma *ODA* es el primer intento importante de favorecer el intercambio de ficheros entre diversos fabricantes, si bien hay que recordar que anteriormente existían especificaciones propias de *IBM*, que fueron las pioneras en este ámbito.

Sin embargo, como ya hemos dicho antes, *ODA* no termina de despegar. Por el contrario, *Acrobat* cada vez tiene mayor presencia en diversos productos de información electrónica y en una buena cantidad de aplicaciones comerciales.

Desde el punto de vista tecnológico, *Envoy* va a la zaga, pero la hegemonía a nivel mundial de *WordPerfect* puede suponer un importante empujón a sus especificaciones.

La experiencia nos dice que el éxito de los nuevos productos no depende sólo de que sean superiores desde el punto de vista técnico. Son muchos los factores que actúan en el desarrollo de los acontecimientos. Y el futuro de este mercado está aún lleno de incertidumbre.

Los documentos compuestos "dinámicos"

Una de las tendencias más destacables de la informática personal durante estos últimos años es el creciente interés por desarrollar lo que se ha denominado "sistemas orientados al documento" (9).

Se trata de entornos operativos que ponen el énfasis en los documentos más que en las aplicaciones. El documento es el centro, y todo gira en torno a él. Éste puede integrar en su seno una serie de elementos -textos, gráficos, tablas, imágenes, sonido...- creados por diferentes aplicaciones y que permanecen ligados a ellas.

Con estos sistemas se pueden gestionar documentos compuestos "dinámicos", también denominados "vivos" (10). Cada parte del documento puede más adelante ser modificada utilizando una aplicación distinta, y conservando siempre su relación con el resto de las partes.

Estos sistemas, que suelen aprovechar las ventajas de las "interfaces gráficas de usuario", son, en palabras de **Marshak**, "verdaderos entornos aplicativos que contienen objetos reprocesables de todo tipo" (11).

Desde el punto de vista orgánico, la información de los documentos compuestos dinámicos se almacena en diferentes ficheros, lo que permite una mayor flexibilidad al sistema: cada aplicación puede operar sobre los correspondientes ficheros.

Entre los sistemas de gestión de documentos compuestos dinámicos destacan *OpenDoc* y *OLE*.

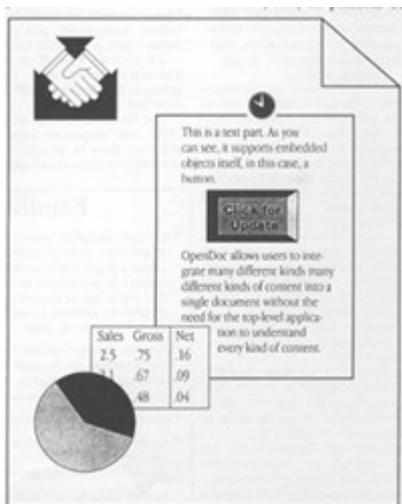


Ilustración 3: Ejemplo de documento compuesto OpenDoc. Una aplicación cualquiera ha servido para crear el gráfico con el "apretón de manos". Luego se ha integrado todo lo demás: el cuadro con el texto -que a su vez contiene un "botón" dentro-, la tabla, el reloj... Cada elemento se ha creado -y se puede modificar- con una aplicación diferente

OpenDoc lo está promoviendo un consorcio de empresas liderado por *IBM*, *Apple* y *Novell-WordPerfect*. En el consorcio se encuentran, además de las tres compañías citadas, *Adobe*, *Hewlett Packard*, *Lotus*, *Oracle*, *Taligent* y *Xerox*.

El consorcio ha creado los *CI Labs (Component Integration Laboratories)*, donde se están desarrollando las especificaciones del sistema. La tecnología de almacenamiento de datos utilizada es la de *Bento*, formato de *Apple*. *IBM* aporta el *System Object Model* para controlar la compatibilidad binaria de los "objetos". Y *WordPerfect* contribuye con su experiencia en integración de productos dentro de *Windows*.

OpenDoc se fundamenta en las técnicas -tan de moda entre los programadores actuales- de "orientación a objetos". En concreto hace uso de la tecnología *Corba (Common object request broker architecture)*. Cada componente de un documento *OpenDoc* es un objeto reprocesable por la aplicación que lo creó.

Su estructura modular organiza la información -que puede ser multimedia: textos, gráficos, imágenes en movimiento, sonidos...- en partes. Estas partes pueden a su vez integrar diversas partes, y éstas otras, etc.

Para referirse a las partes, los creadores de *OpenDoc* no suelen utilizar el término "fichero",

sino que prefieren denominarlas *containers*.

A las aplicaciones necesarias para manipular las partes de un documento se les llama *part handlers*, que pueden ser de dos tipos: *editores* y *visualizadores*. Se trata de una distinción similar a la que ya vimos antes cuando nos ocupamos de *Acrobat*, *Envoy*... Como en los casos anteriores los visualizadores (*viewers*) se distribuyen gratuitamente junto con los documentos.

En abril o mayo de este año han de estar disponibles las herramientas de desarrollo *OpenDoc* para programadores.

Dentro del listado de empresas comprometidas con *OpenDoc* hay una estudiada ausencia: la de *Microsoft*. En esto, como en otras batallas informáticas, la empresa de **Bill Gates** hace la guerra por su cuenta. *Microsoft* defiende en solitario su propio sistema: *OLE (Object linking and embedding)*, que utiliza una filosofía en cierta forma similar a la de *OpenDoc* -al menos desde el punto de vista funcional-, pero que está concebido únicamente para ser explotado dentro de los entornos operativos que está desarrollando la misma empresa.

¿Puede *Microsoft* permitirse el lujo de ofrecer, ella sola y bajo licencia, su propio sistema, compitiendo con *OpenDoc*? La respuesta es: *sí*. Ninguna empresa de software quiere perder el tren de las futuras versiones de Windows ni quedarse fuera de las especificaciones de Windows NT.

Los principales fabricantes de software están ajustando con prisa sus productos a las exigencias de la versión 2.0 de *OLE*. Todos quieren que el portapapeles de los sistemas operativos de *Microsoft* permita "incrustar" sus ficheros dentro de otros ficheros Windows.

De hecho, entre los sistemas que se van a ajustar a la tecnología *OLE* está... ¡*OpenDoc*!

La arquitectura cliente/servidor

Si las partes de un documento compuesto pueden estar distribuidas en diversos ficheros, ¿quién impide que éstos estén situados en diversos ordenadores dentro de una red?

En efecto, cada vez es mayor la importancia de las redes, especialmente para utilizar un tipo de software denominado *groupware*, esto es, software con el que se lleva a cabo el trabajo "en grupo". Los productos *groupware* suelen utilizar la arquitectura cliente/servidor (12), y la mayoría tienden a ser "docucéntricos" (13).

Pues bien, dentro del esquema *cliente/servidor* encontramos otra de las tendencias que queremos ahora analizar.

Existe una buena cantidad de productos, la mayoría poco conocidos en nuestro país, que son capaces de gestionar documentos cuyas partes están distribuidas por una red. Lo normal es que en el servidor se encuentre la aplicación central, que, además de albergar los ficheros principales, puede tener incorporado un sistema de índices y un motor de recuperación de información. Por su parte, cada aplicación cliente cuenta con su propio interface para acceder al servidor (14).

Algunos de los productos que están explotando la arquitectura cliente/servidor son: *Asksam* (*Asksam*), *HotDocs* (*CapSoft*), *Documentum* (*Documentum*), *Excalibur* (*Excalibur*), *Visual Document Library* (*IBM*), *Relational Document Manager* (*Interleaf*), *Notes* (*Lotus*), *Document Direct* (*Mobius*), *ISYS* (*Odyssey*), *PC Docs Open* (*PC Docs*), *DocStor* (*Salix Systems*), *SoftSolutions* (*SoftSolutions*), *Topic* (*Verity*), *Open/Profound* (*Wang*), *Visual Recall* (*XSoft*).

DEN, EDM y Shamrock

El reto de estos sistemas es desarrollar normas de arquitectura abierta que permitan que dos o más aplicaciones puedan comunicarse entre sí compartiendo ficheros. En julio del año pasado *Novell* y *Xerox* hicieron público un acuerdo para poner en marcha los servicios *DEN* (*Document Enabled Networking*), un proyecto de norma abierta que funcionaría inicialmente bajo *Netware*, el software de red de *Novell*. Más adelante podrán existir versiones para otras redes.

El objetivo de *DEN* es proporcionar un estándar abierto que permita a los fabricantes de software crear documentos compuestos que sean gestionables por diversas aplicaciones en un entorno de red. Se trata, en definitiva, de que cualquier aplicación cliente se pueda "entender" con cualquier aplicación servidor.

A las dos empresas promotoras de la norma *DEN* se han añadido después *IDI* (la empresa, filial de *Oclc*, que comercializa *Basis*), *Kodak*, *Oracle*, *PC Docs*, *SoftSolutions* (filial de *Novell*), *Verity* y *XSoft* (filial de *Xerox*).

Existen además dos grupos de empresas que están trabajando en una dirección similar.

El primero de ellos ha desarrollado *Odma* (*Open Document Management API*), un API (interface de programación de aplicaciones) que permite a las aplicaciones cliente entenderse con las aplicaciones de los servidores.

Han participado en el desarrollo de *Odma* las siguientes empresas: *Adobe*, *Andersen*, *Autodesk*, *Borland*, *Documentum*, *Interleaf*, *Microsoft*, *Novell*, *Oracle*, *PC DOCS*, *SoftSolutions*, *Sybase*, *Wordperfect* y *XSoft*.

Las prestaciones de *Odma* son similares a las que se ha propuesto ofrecer *DEN*. Por otra parte, la presencia del gigante *Novell* en ambos proyectos ha hecho pensar que en el futuro *Odma* puede pasar a formar parte de los servicios *DEN* (15). En cualquier caso, pocos auguran continuidad a *Odma*.

Los objetivos de la *Shamrock Document Management Coalition* son mucho más ambiciosos: se busca, desde mediados de 1993, la total interoperatividad entre todos los sistemas. Para ello se está definiendo un conjunto de llamadas y servicios comunes que han de componer la arquitectura *Enterprise Document Management (EDM)*.

La coalición agrupa a *Adobe*, *Aetna Life and Casualty*, *Andersen*, *Coca-Cola*, *Documentum*, *EDS*, *Frame Technology*, *Hewlett-Packard*, *IBM*, *Interleaf*, *Merck*, *Microsoft*, *PC DOCS*, *Saros*, *Sybase*, *Verity*, *ViewStar*, *Wang* y *XSoft*, entre otros. Como puede verse, una curiosa mezcla de productores de software, fabricantes de hardware e importantes usuarios de redes.

En cierta forma la filosofía de *Shamrock* es heredera de otro conocido estándar, *SQL* (*Structured Query Language*), el "lenguaje común" que permite a los programas de gestión de bases de datos relacionales entenderse entre sí.

Para elaborar los servicios de *Shamrock* se está utilizando como "fuente de inspiración" sobre todo la tecnología de una de las empresas presentes en la coalición: *Saros*. No en vano esta empresa, junto con *IBM*, son quienes están liderando el proyecto.

De nuevo la pregunta: ¿triumfará *DEN* o *Shamrock*? Es la lucha entre una iniciativa del poderoso *Novell* contra el proyecto de un nutrido grupo de empresas.

El paradigma hipertexto/hipermedia

La última tendencia que queremos comentar en este artículo es la de los sistemas que permiten ofrecer prestaciones *hipertexto/hipermedia*. Algunos de los sistemas ya analizados en párrafos anteriores lo hacen. Es el caso de *Sgml*, *PDF-Acrobat*, *Envoy*... Pero nuestra intención es ahora ocuparnos de *Html* (*Hypertext markup language*), el lenguaje de etiquetas que hace posible que exista *WWW* (*World Wide Web*), la "telaraña mundial" (16).

Parece como si *Html*, con gran sencillez, fuera capaz de aglutinar todas las tendencias que hemos venido analizando: documentos compuestos dinámicos, capacidad multimedia, arquitectura cliente/servidor, hipertexto e hipermedia.

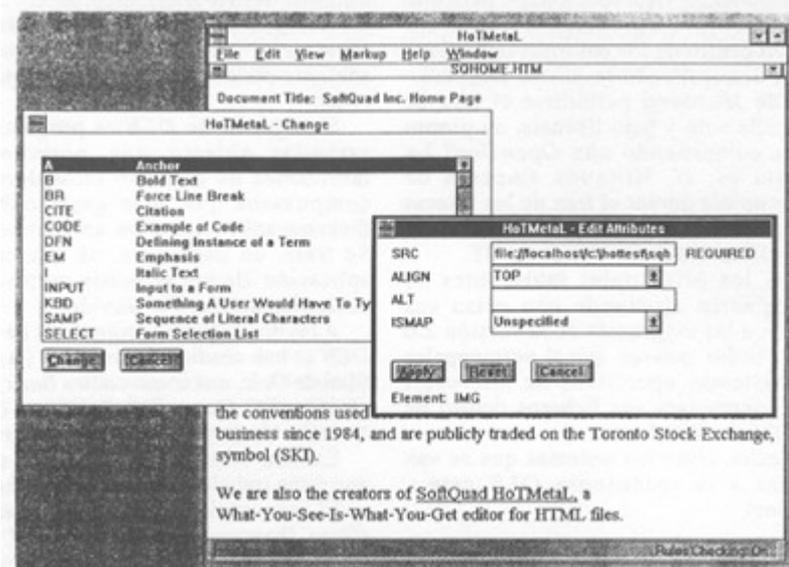


Ilustración 4: HotMetal, un editor especializado en la creación de ficheros Html

Pero lo más espectacular es la facilidad. Un usuario medio es capaz de elaborar documentos completos *Html* tras sólo un par de horas de aprendizaje.

Html utiliza un sistema de etiquetas que se inspira en la norma *Sgml*.

Con *Html* se pueden organizar documentos complejos que contengan textos, gráficos integrados en las páginas del documento, sonidos, imágenes en movimiento, entradas de menús... y, sobre todo, enlaces hipertextuales con otros documentos u otras partes del mismo documento.

Gracias a los servidores *Web* es posible navegar por la inmensidad de Internet saltando de un fichero *Html* a otro. O ver/oir las diversas partes de un documento compuesto *Html* -un texto, una imagen en movimiento, una melodía, un menú *gopher*...-, que puede estar en cualquier lugar del planeta.

Referencias

1. Se utiliza aquí, más desarrollado, el esquema del siguiente trabajo: **Pedro Hípola**, "Los estándares para la edición electrónica", en *Jornada sobre edición electrónica*, actas (en prensa) del seminario organizado por el *Col.legi Oficial de Bibliotecaris-Documentalistes de Catalunya* y *Doc6*, Barcelona, 23 de noviembre de 1994.

2. ver **Pedro Hípola**, "World Wide Web: toda la Internet en un solo 'documento'", en [IWE-29](#), p. 25-27.
3. ver **Pedro Hípola**, "Sistemas EDI", en [IWE-9](#), p. 4-6.
4. ver **Pedro Hípola** y **Félix de Moya**, "Proyectos EDI y normalización documental", en *Revista española de Documentación científica*, octubre-diciembre 1991, 4, p. 408-419.
5. ver **Isidre Canals**, "Los documentos compuestos vivos, base de los futuros *sistemas orientados al documento*", en [IWE-25](#), p. 14-16.
6. ver "Open Document Management", en *Electronic Documents*, vol. 3, n. 10, octubre 1994, p. 18-20.
7. ver **Pedro Hípola**, "World Wide Web: toda la Internet en un solo 'documento'", en [IWE-29](#), p. 25-27.
8. ver **Lluís Codina**, "Envoy: producción y distribución de documentos electrónicos", en [IWE-30](#), p. 18-20.
9. ver **Lluís Codina**, "Situación y tendencias de los microordenadores (1ª parte)", en [IWE-11](#), p. 17-20.
10. ver **Isidre Canals**, "Los documentos compuestos vivos, base de los futuros *sistemas orientados al documento*", en [IWE-25](#), p. 14-16.
11. **R. Marshak**, "Paradigm Shift", 25 febrero 1991, citado por **Isidre Canals** en [IWE-25](#), p. 15.
12. ver **Tomás Baiget**, "Cliente-servidor", en [IWE-14](#), p. 16-17.
13. ver **Lluís Codina**, "Sistemas de gestión documental: situación, problemas actuales y principales novedades tecnológicas", en [IWE-21](#), p. 7-9.
14. ver **Linda Musthaler**, "A tall order for document managers", en *Network World*, 18 de julio de 1994, p. 35-44.
15. ver **Andy Reinhart**, "Managing the new document", en *Byte*, agosto 1994, p. 91-104.
16. ver **Pedro Hípola**, "World Wide Web: toda la Internet en un solo 'documento'", en [IWE-29](#), p. 25-27.