

# EDICION GRAFICA PARA INVIDENTES

Alonso Amo, F.; Montes, C.; Pazos, J.

Facultad de Informática - Universidad Politécnica de Madrid

Campus de Montegancedo

28660-Boadilla del Monte

Madrid

## ABSTRACT

El procesamiento de los fondos bibliográficos para su utilización por parte de personas invidentes es, en la actualidad una tarea compleja y, en muchos casos, imposible. Para que este colectivo tenga acceso a libros de texto, estos han de ser convertidos previamente al alfabeto Braille, bien mecanografiándolos directamente o mediante un OCR, con la ayuda de un operador especializado. Es prácticamente imposible transcribir de forma automática a Braille texto matemático y científico, e incluso una persona altamente cualificada se ve obligada a realizar continuas consultas a los manuales, lo cual conlleva en la mayoría de los casos resultados con discrepancias semánticas que no pueden ser interpretadas por los invidentes.

En este artículo se describe un sistema de edición para invidentes (EDIE), que es capaz de convertir automáticamente a Braille tanto texto común como símbolos científicos y matemáticos, que aparecen en los fondos bibliográficos de las editoriales.

**PALABRAS CLAVE:** Ingeniería de Software, Traductores, Autómatas, Invidencia, Edición

## 1. Introducción

En la actualidad, hay un gran desarrollo en investigación y diseño de productos comerciales dirigidos al gran público para aumentar su calidad de vida, pero pocos esfuerzos se han orientado a la investigación de nuevas tecnologías que permitan a este público minoritario aumentar su calidad de vida e inserción laboral.

La situación descrita es un hecho reconocido por la comunidad científica, hasta el punto de que la propia Comisión Científica de las Comunidades Europeas ha lanzado recientemente el programa de investigación TIDE (Tecnología de la Comunicación, Control e Información para personas con discapacidades y ancianos) [12] encaminado al ámbito de la tecnología de la rehabilitación. Sus objetivos fundamentales consisten en el fomento de las ayudas técnicas, y la adaptación de la tecnología existente, para las personas discapacitadas y ancianos, con lo que coincide con la meta perseguida en el desarrollo de EDIE.

## 2 Características generales del editor

Se pretende definir un Sistema Editor capaz de dar acceso al invidente a documentos textuales y con simbología científica, sin perder su contenido semántico, a través de las herramientas informáticas de adaptación a invidentes. En concreto el sistema deberá:

- Analizar el texto de salida del proceso de edición producido por el paquete de edición y maquetación industrial de una editorial y, en base a los comandos reconocidos en el análisis, efectuar la ejecución apropiada.
- Traducir el texto de edición obtenido en la etapa anterior. Esta traducción deberá ser legible para personas videntes que utilicen el equipo, y comprensible para un invidente en la línea Braille, sintetizador de voz e impresora Braille.
- Formatear el texto en formato estándar para su adaptación a una futura edición por terminal, y a la impresora Braille.
- Producir un sistema de edición que permita el acceso al texto a través de la línea Braille y el sintetizador de voz.

## 3 El prototipo EDIE

Como resulta evidente, el Sistema Editor debe incorporar en su etapa de producción diferentes sistemas de edición y maquetación utilizados en editoriales y centros de documentación. Por ello, y de acuerdo con el ciclo de vida de la metodología IDEAL [10] [8], resulta necesario definir un sistema prototipo cuya implementación, haga viable el Sistema Editor.

EDIE es el sistema prototipo de edición en Braille, diseñado específicamente para tratar la información producida en soporte magnético por el paquete de edición y maquetación PENTA, y utilizando como adaptaciones el sintetizador de voz VERT PLUS o CIBER 232, la línea Braille BRAILLEX-IB 80, y la impresora Braille THIEL.

Se escoge el sistema PENTA por ser uno de los más potentes y el más utilizado por las grandes editoriales. En líneas generales, su funcionamiento es el siguiente: El texto se introduce y manipula utilizando terminales Penta Vue y Penta Saturn, el cual se compone y pagina a través del software Penta Quick y Penta Page. La salida final se envía al Mergenthaler Linitron, que produce la edición definitiva para su posterior impresión. Todo el hardware y software mencionado configuran el sistema Penta [9].

El sistema Penta presenta, asimismo, una complejidad y una completitud suficiente para mostrar un amplio abanico de situaciones comunes en el proceso de edición, por lo que sirve de guía para posteriores estudios de otros paquetes de edición.

La editorial seleccionada, y que utiliza este paquete, ha sido Santillana S.A., porque ofrece una amplia gama de publicaciones en todos los niveles de la enseñanza lo que posibilita al invidente el acceso inmediato a un gran volumen de fondos bibliográficos de carácter educativo.

Las adaptaciones Braille seleccionadas son las más utilizadas actualmente por el personal invidente en el entorno de un sistema personal de procesamiento, no obstante la sustitución por otras no planteará grandes cambios en el sistema prototipo.

#### **4. Especificación del Sistema EDIE**

Para la definición y posterior implementación del sistema prototipo EDIE, se fijaron los siguientes criterios de éxito:

- a) Textos básicos, compuestos por caracteres alfanuméricos y de puntuación. En primera fase, debe reconocer la estructura y significado de todos los caracteres del texto fuente. En segunda fase, debe traducir el texto de entrada a caracteres Braille de salida o a órdenes de formato, dependiendo del significado de la entrada.
- b) Textos científicos, compuestos por texto básico y simbología científica. El sistema debe realizar las fases mencionadas en el punto anterior con este tipo de texto.

Los anteriores criterios dieron lugar a las siguientes especificaciones:

- Reconocimiento y procesamiento de texto básico generado por Penta.
- Traducción a código ASCII-Braille.
- Ejecución de órdenes de formateo Penta en texto básico de edición.
- Reconocimiento y procesamiento de texto científico Penta.
- Traducción a código ASCII-Braille científico.
- Ejecución de órdenes de formateo Penta en texto científico de edición.

Estas especificaciones configuran el diagrama de flujo de datos de la fig. 1.

#### **5. Estructuras de proceso y de datos del Sistema EDIE**

Realmente la información inicial contiene dos tipos básicos de elementos de información a considerar: el texto, que se obtiene como producto final del proceso de edición, y los comandos de control, que permiten al paquete de edición, mediante un reconocimiento de los mismos, realizar el diseño y la maquetación. El sistema ha de permitir, pues, extraer el texto explícito e implícito mediante una interpretación adecuada de los comandos de control, de manera que se mantenga intacta su semántica.

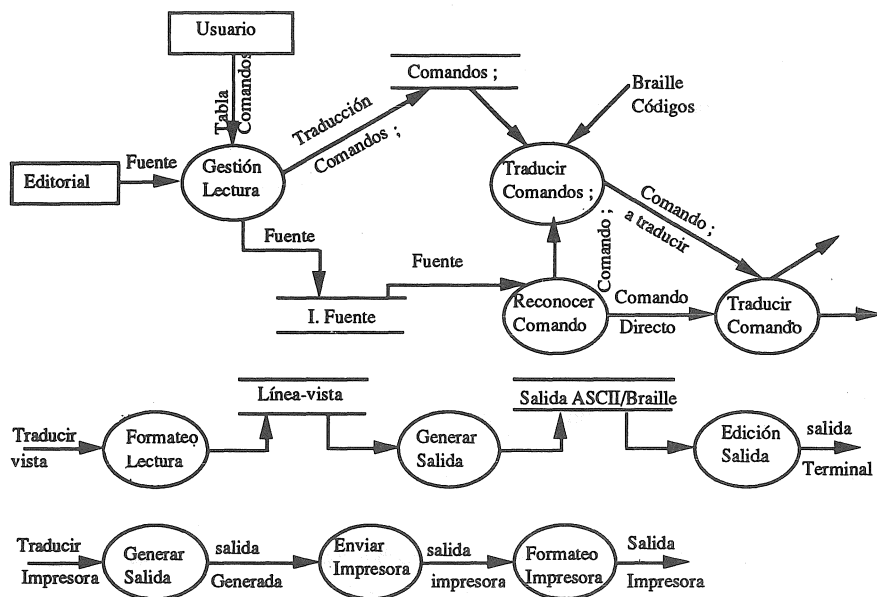


Figura 1. Flujo de datos del Sistema EDIE

Por ello se definieron unas estructuras específicas de información en base a la estructura de estos comandos de control con el fin de realizar un reconocimiento de las acciones que toman sobre el texto final, y así poder efectuar las acciones adecuadas para el mantenimiento de la integridad en la salida.

Este reconocimiento se realiza mediante un **autómata finito** de reconocimiento de cada comando, y a unas tablas **hash** abiertas por los comandos ";".

El problema a tratar es claramente de reconocimiento y traducción por medio de un lenguaje regular, cuya extensión y sintaxis es conocida. Este lenguaje regular es representable por una gramática de tipo 3 de Chomsky [3] [5] [2] en la que las producciones quedan restringidas a:

$$A \longrightarrow \alpha$$

$$A \longrightarrow a\beta$$

donde el lado derecho consta de un símbolo terminal, o de un símbolo terminal seguido por un símbolo no terminal; y se reconoce por una estructura de **autómata finito**.

En el caso concreto que nos ocupa, la **quíntupla** (E,Q,f,p<sub>0</sub>, F) que define el **autómata finito** tiene el siguiente significado:

- E Alfabeto de entrada compuesto por texto y comandos.
- Q Conjunto, no vacío de estados, que se agrupan en dos niveles
  - Nivel Principal: comprende 8 estados según el tipo de comando (fig. 2)

- Nivel Comando: cada uno de los estados de un comando (fig. 3)
- f Función de transición de Q x E (fig. 3)
- p<sub>0</sub> Estado inicial del autómata con valor cero.
- F Conjunto de estados finales (fig. 3)

Tipo	Comando
1	'='
2	','
3	'['
4	'-' y '^'
5	'I'

Figura 2. Tipos de comandos

El autómata finito determinista queda representado por el diagrama de transición de la figura 3, grafo cuyos nodos son los elementos Q y cuyas ramas dirigidas son función de f. Su estructura es especialmente útil en este caso en que se tienen muchos estados [7], y se conoce, en general, su sintaxis y semántica. No obstante, los comandos de tipo 2 (de ;) presentan la peculiaridad de que se conoce su sintaxis, pero es desconocida para el autómata su número (que en principio es ilimitado), y su semántica. Ello es debido, a que vienen definidos por el usuario, de acuerdo con la utilización que se prevea, dando lugar al tratamiento de un lenguaje variable.

Para el reconocimiento de este tipo de lenguajes las técnicas a aplicar son las redes ATN y los diccionarios. En este caso se implementó un diccionario conteniendo los comandos y su traducción, en una estructura de tabla hash abierta [1] [6] [11], y utilizando el comando como clave. La función hash será de la forma:

$$\text{hash}(\text{comando ;}) = (\text{comando ;}) - ' ; ' \wedge$$

asignando a cada comando el código ASCII del primer carácter posterior a ' ; ' menos el valor ASCII ' ^ ', primer valor que se encuentra después de punto y coma. La tabla se ubicará en memoria principal para conseguir una velocidad razonable de traducción, y la carga de sus valores se efectúa mediante estructuras dinámicas lo que permite el acceso inmediato a estos comandos.

Las principales ventajas de este diseño están en que permiten una inserción rápida en tiempo de ejecución de nuevos comandos, efectúa una gestión dinámica de memoria reservando espacio para un nuevo comando sólo en el momento de su aparición, y proporciona un buen tiempo de respuesta en la ejecución del comando.

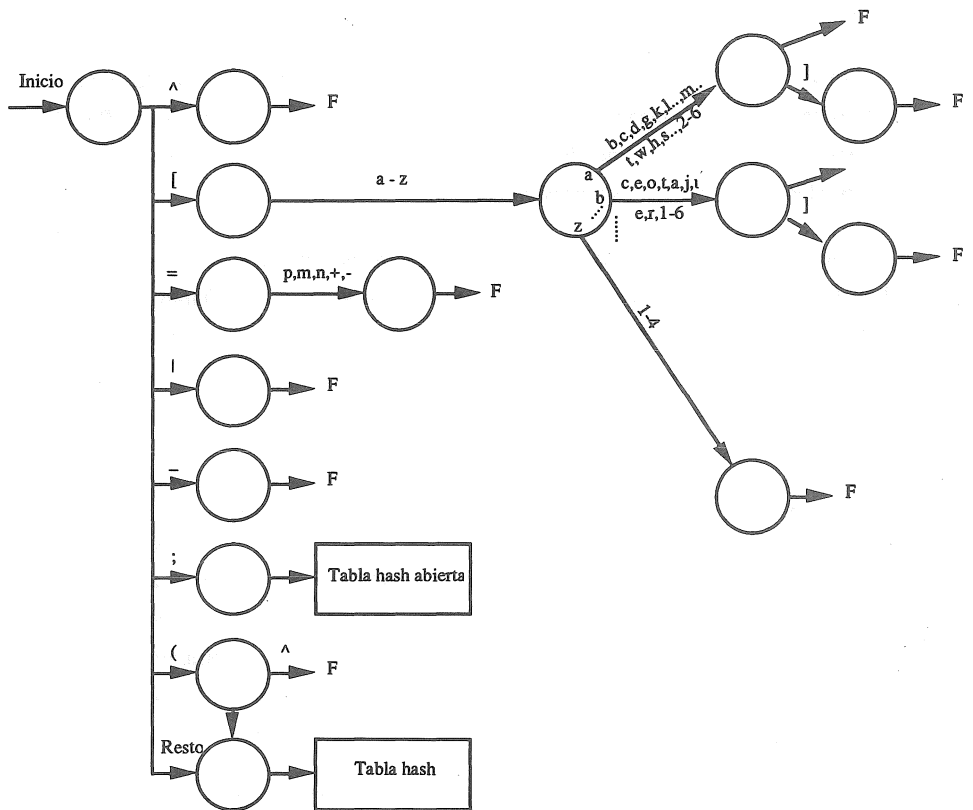


Figura 3. Autómata finito para reconocimiento de comandos

Otra estructura importante, en el diseño del sistema, es la traducción a lenguaje Braille de la correspondiente codificación ASCII. En este punto interesa destacar los diferentes sistemas Braille existentes, por lo que se ha utilizado un diccionario de traducciones, implementado mediante una tabla hash de símbolos [1] [6], siendo la función hash:

$$\text{hash (carácter)} = \text{carácter} - ' '$$

donde la clave la constituye un carácter, y la información la codificación Braille correspondiente. Se está ante un conjunto finito y conocido de entradas al diccionario, que permiten la modificación de las traducciones en función de los diferentes sistemas Braille que se utilicen. En el caso que nos ocupa son dos, uno para la línea Braille y otro para la impresora Braille.

## 6. Edición en línea Braille y sintetizador de voz

El análisis de estas adaptaciones conduce a las siguientes conclusiones:

- La línea Braille es un dispositivo, fundamentalmente hardware, que efectúa una captura de los caracteres visualizados en la línea de pantalla donde se sitúa el cursor. Por ello, no es posible representar en línea Braille nada que no se represente en la pantalla, al menos sin perder alguna de las dos representaciones.
- El sintetizador de voz es un dispositivo, principalmente software, que captura de pantalla los caracteres de la línea en la que se encuentra el cursor. Lee por sílabas, y en el caso de detectar un carácter desconocido deletrea todos los caracteres. Dispone de un juego reducido de caracteres, inferior al conjunto ascii, y por supuesto, no reconoce los caracteres no-ascii.
- En el teclado del sistema se dispone de todos los caracteres ascii, pero no incluye la mayoría de los símbolos propios de las matemáticas, lógica, química, etc.

Teniendo en cuenta, que se trata de editar un texto compuesto por caracteres ascii y notación científica, y considerando las diferentes limitaciones presentadas por cada adaptación para invidentes, se ha optado como solución la de transcribir con la palabra completa el nombre del símbolo (ej.  $\exists$  por 'existe'). De esta forma se llega a una notación legible para las tres representaciones.

Para el manejo del texto de salida se pensó, inicialmente, utilizar un editor comercial, pero el hecho de que el sintetizador de voz requiera que el cursor esté posicionado en la línea que se desea leer, obliga a desechar esta opción, y a tener que diseñar un editor propio, sólo de lectura, que permitan la edición del texto convertido, y el libre movimiento del cursor por el principio de todas las líneas del texto, resolviendo así la lectura del sintetizador.

## 7. Edición en impresora Braille

La edición con impresora Braille (THIEL) se trata independientemente por ser un elemento básicamente de producción Braille, más que de tratamiento interactivo.

Esta impresora presenta dos modos de funcionamiento [13]:

- a) Modo ascii.- Cuenta con un generador de código que traduce los caracteres ascii a código Braille.

- b) **Modo transparente.**- Los datos que recibe son interpretados de derecha a izquierda como caracteres Braille, asignado el bit 0 del carácter al punto 1 de la matriz Braille, el bit 1 al punto 2,... y el bit 7 al punto 8.

La impresora permite pasar de un modo a otro mediante secuencias de escape.

Con la impresora funcionando en modo ascii se diseñará un sistema de conversión Braille de todos los caracteres, incluidos los de notación científica, a código Braille, conforme con el Código Matemático Unificado [4], produciendo este dispositivo una salida Braille que representa fielmente los textos normales y científicos que se traduzcan.

Este sistema de edición en impresora incorporará en este prototipo un formateador estándar Braille para líneas, saltos de líneas, páginas, nº de página, encabezamiento,... de acuerdo con la normativa Braille.

## **8. Resultados y conclusiones**

El sistema prototipo EDIE permite al invidente el acceso a la documentación bibliográfica textual y científica, sobre soporte magnético, producida por el sistema PENTA de edición y maquetación industrial. Ello posibilita la utilización de los fondos editoriales de Santillana y SM, por citar dos importantes empresas del sector que utilizan el sistema PENTA, así como los de otras editoriales que trabajan con este sistema.

Finalizada esta etapa, el Sistema Editor tratará bibliografía producida en entornos Macintosh que está empezando a introducirse en el mundo de las editoriales.

La implementación del sistema se ha realizado en C++, utilizándose la estructura orientada a objetos a nivel de módulo.

En resumen, el desarrollo del Sistema Editor que se describe en este artículo ha planteado, en particular, un análisis específico de los siguientes elementos y estructuras:

- Sistema Penta. Estudio general del Sistema, y dominio de los comandos de control y traducibles (o comandos ;).
- Autómata finito determinista, para el reconocimiento y posterior tratamiento de los comandos de control.
- Tabla hash de listas encadenadas, para el reconocimiento del lenguaje variable de los comandos;

- Diccionario de traducciones, implementado mediante una tabla hash de símbolos, para la traducción a lenguaje Braille de la correspondiente codificación ASCII.
- Línea Braille.- Estudio de este dispositivo, de su código Braille, y su intersección con la pantalla del Ordenador.
- Sintetizador de voz.- Estudio de este dispositivo, e intersección con la pantalla del ordenador.
- Impresora Braille.- Estudio de los modos de funcionamiento ASCII y transparente, y del Código Matemático Braille que se utiliza para la notación científica.

EDIE ha deparado unos resultados muy satisfactorios, hasta el punto de que está siendo utilizado en el Centro de Producción Bibliográfica de la Organización Nacional de Ciegos.

El grado de fidelidad en la transcripción del texto procedente del sistema PENTA alcanza el 100%. Esto quiere decir que si el operario encargado de la maquetación utilizara PENTA perfecta y precisamente, el problema de la conversión estaría resuelto. Ahora bien, en multitud de ocasiones los maquetadores hacen "trampas" al sistema con el fin de simplificar su trabajo (ej. utilizar un código de tabla para recuadrar una frase o no utilizar subíndices sino cambios de tamaño del fuente), lo que conlleva la aparición de una serie de ambigüedades e imprecisiones en el resultado final.

Para resolver este problema, se está desarrollando en la actualidad un filtro-maquetador inteligente, capaz de resolver situaciones ambiguas dependientemente del contexto, y de producir una maquetación del texto adecuada para los invidentes.

En cuanto al tratamiento de los gráficos y figuras que aparecen en el texto, su tratamiento es dependiente de su complejidad. Los gráficos y figuras simples actualmente se reconocerán e interpretarán, generando automáticamente la salida de la gráfica y su descripción. Los gráficos complejos serán referenciados para su incorporación manual, ya que en la actualidad no existe posibilidad de desarrollar un reconocedor universal de imágenes.

## 9. Bibliografía

- [1] Alonso-Amo, F.; Morales Lozano, A.: *Técnicas de Programación - Programación Estructurada, PASCAL, Estructuras de Datos*. Paraninfo. 1987.
- [2] Chòmsky, A.N.: *On Certain Formal Properties of Grammars*. Information Control, Vol. 2, June 1959.
- [3] Crespo: *Informática Teórica*. Vol. 1. Facultad de Informática. UPM. 1982.

- [4] Della Barca, J.J.; Ortiz, M.A.; Rodrigo, F.; Roig, C.: *Código Matemático Unificado para la Lengua Castellana*. Reunión de imprentas Braille de habla hispana. Montevideo, 1987.
- [5] Hopgood, F.R.A.: *Compiling Techniques*. Macdonald. London and American Elsevier, Inc. 1970.
- [6] Horowitz, E.; Sahni, J.: *Fundamentals of Data Structures in Pascal*. Computer Science Press. 1984.
- [7] Kain, R.Y.: *Automate Theory: Machines and Languages*. McGraw-Hill. 1972.
- [8] Maté, J.L.; Pazos, J.: *Ingeniería del Conocimiento. Diseño y Construcción de Sistemas Expertos*. Facultad de Informática, Madrid, 1988.
- [9] Pazos, J.: *Metodología IDEAL para el Diseño e Implementación de Sistemas Expertos*. Actas de las Jornadas de Rank Xerox sobre Sistemas Expertos, 1986.
- [10] PENTA: *Command Reference Manual*. Penta Systems International INC, 1985.
- [11] Sanchis, F.J.; Galán, C.: *Compiladores. Teoría y Construcción*. Paraninfo, S.A. 1981.
- [12] TIDE: *Documentación del Proyecto TIDE de la Comunidad Europea :Information Package*. 1990.
- [13] THIEL: *Manual de utilización e instalación de la Impresora Braille THIEL*. Thiel Gmble Co.,1984.