Organización y recuperación de la información

Documentos de lectura

UP01/79009/00625

UOC
La universidad virtual

Esta recopilación de artículos ha sido seleccionada
por Cristòfol Rovira Fontanals y Lluís Condina Bonilla.

Primera edición: febrero 2002

Av. Tibidabo, 39-43, 08035 Barcelona
Diseño: Manel Andreu
Producción editorial: Eurecamedia, SL
ISBN: 84-8429-438-2
Depósito legal: B-47487-2001

Índice

Fundamentos de teoría de recuparación de información

Lluís Codina*

1. ¿Qué es la RI?

1.1. Definición

La recuperación de información (RI a partir de ahora) es el proceso de selección de información de un fondo documental por medios automáticos o semiautomáticos, es decir, con el uso parcial o intensivo de ordenadores.

Por lo tanto, la RI está en el núcleo de las operaciones más importantes de la gestión documental, a saber, la representación, identificación, selección y extracción de los documentos más relevantes de un fondo documental para solucionar las necesidades de información de sus usuarios.

1.2. Disciplina

También es un campo de estudio interdisciplinario que recibe el nombre de Teoría de Recuperación de Información, a la que contribuyen especialistas procedentes de disciplinas muy variadas, pero sobre todo de la Documentación, la Informática y la Lingüística. Como tal campo de estudio nació justo a partir del momento que pudieron utilizarse ordenadores en la gestión documental, en concreto en la gestión de grandes masas (según el punto de vista de la época) de información textual.

El antecedente más remoto son los trabajos realizados en la década de los cuarenta por un estudioso del lenguaje, G.K. Ziph, descubridor de la llamada, en su honor, *ley de Ziph*, según la cual si estudiamos la frecuencia de las palabras en un corpus lo suficientemente representativo de una lengua obtenemos esta relación:

$$\text{Frecuencia} * \text{Rango} = \text{constante}$$

donde la frecuencia es el número de veces que sucede una palabra y rango es el número de orden de la palabra en la lista de distribución de frecuencias.

Los estudios de Ziph demostraron que se podían descubrir determinadas propiedades estructurales de masas de informaciones textuales. Por ejemplo, ocurre que, según la ley de Ziph, la mayor parte de las palabras sucede unas pocas veces, y que, a la inversa, sólo unas pocas palabras suceden muchas veces.

* Lluís Codina es doctor en Ciencias de la información y profesor titular de universidad de la Universidad Pompeu Fabra de Barcelona. Correo electrónico: lluis.codina@cpis.upf.es.

En la década siguiente, un investigador de la empresa IBM, H.P. Luhn, postuló la creación automática de índices utilizando las propiedades estadísticas de los textos. Entre otras cosas, propuso el concepto de *poder de resolución* de un término en relación con su capacidad para identificar el tema de un documento, capacidad que está directamente relacionada con la frecuencia del término en el documento.

También propuso la idea de que los términos con más "poder de resolución", o capacidad de discriminación de documentos, como se suele decir ahora, son los que están situados en la parte media de la distribución de frecuencias, es decir, los términos que no son ni muy frecuentes ni muy poco frecuentes. Por lo tanto, los términos situados en la franja media son los mejores candidatos a ser utilizados como términos de indización o descriptores en un fondo documental.

La teoría de recuperación de información evolucionó lentamente hasta dar un importantísimo salto cualitativo con la obra de Gerard Salton, desarrollada a lo largo de dos décadas y que llegó hasta principios de los noventa.

En concreto, Salton sistematizó los principios y los principales hallazgos de la Teoría de Recuperación de Información en una obra de 1983 (escrita con colaboración con otro autor, M.J. McGill); trabajo que culminó en su obra posterior de 1989 (esta vez en solitario). En estos dos libros, Salton (1983, 1989) dio una visión sólida y unificada de la disciplina y presentó los procedimientos y los algoritmos más importantes de ésta. Asimismo, este autor desarrolló el sistema SMART, un sistema de indización automática de documentos que todavía hoy está en la base de numerosos motores de búsqueda de Internet.

1.3. Operaciones

Una de las operaciones más significativas de la RI es la representación del contenido semántico de documentos y de las necesidades de información con el uso de grupos de palabras o términos del lenguaje natural.

Dado que estas palabras formarán parte de un índice, esta operación recibe el nombre de *indización* y los términos utilizados en esta representación reciben el nombre de *términos de indización*.

De este modo, las características y propiedades de un documento, incluso de un documento no textual, como una imagen, quedan representadas como un conjunto de términos de indización.

Como es bien sabido, la representación de documentos con términos de indización es una operación que puede realizarse de manera intelectual ("manual"), en cuyo caso dichos términos suelen considerarse descriptores, es decir, térmi-

nos elegidos de un lenguaje documental, como por ejemplo un tesauro, para representar de manera coherente los temas de los documentos.

Así, el objetivo más características de los estudios sobre RI consiste en encontrar la forma de realizar, de manera automática y con la máxima eficiencia posible, el siguiente grupo de funciones u operaciones:

1. Identificar cuáles son los temas más relevantes de un documento.

2. Determinar y asignar los descriptores más adecuados para representar estos temas.

3. Construir lenguajes documentales, como tesauros.

4. Derivar y producir resúmenes de los documentos que, eventualmente, y en según qué circunstancias, puedan sustituir la lectura del documento completo.

5. Determinar cuáles son, y en qué grado, los documentos más relevantes en relación con una necesidad de información determinada.

Como se puede apreciar, los puntos 1 a 4 corresponden a la operación intelectual llamada análisis e indización documental y aspectos relacionados, como la construcción de tesauros. Por lo tanto, gran parte de los trabajos de la RI se han encaminado a automatizar al máximo las tareas típicas de análisis e indización documental que, tradicionalmente, se han hecho de forma "artesanal".

Otros trabajos en RI han estudiado aspectos como el diseño eficiente de interfaces de usuario, la determinación y creación de enlaces hipertextuales de forma automática, la capacidad de obtención de información por medio de las redes de citas o mediante los enlaces entre sus web y la visualización de informaciones inherentemente textuales en forma espacial o gráfica.

Ante el fenómeno de Internet y la sociedad digital, la RI se revela como uno de los campos de estudio interdisciplinario más importantes del futuro, con oportunidades para especialistas de muchas y muy variadas ramas, desde las humanidades hasta las ingenierías.

2. Sistemas de recuperación de información

Toda la moderna teoría de la RI se fundamenta en las siguientes tres ideas nucleares:

1. La representación de la información que contienen los distintos documentos se realiza mediante la asignación de varios conjuntos de términos de indización a cada documento, y no tanto por asignación de los documentos a clases o subclases de un cuadro de clasificación.

2. Las necesidades de información de los usuarios de un sistema documental también pueden representarse mediante conjuntos de términos de indización.

3. Los documentos más relevantes en relación con cada necesidad de información de los usuarios serán aquellos que presenten un mayor grado de similitud con respecto a la necesidad de información.

Los sistemas que realizan la clase de operaciones según las tres ideas nucleares anteriores se denominan sistemas de recuperación de información (o SRI), y su composición es la que se presenta en el cuadro siguiente (cuadro 1), donde podemos ver cuáles son los elementos que forman parte de un SRI típico:

Cuadro 1. Elementos y funciones que forman parte de un sistema de RI

- La entidad necesidad de información (1).
- La entidad documento (2) que, a su vez, forma parte de un fondo documental (3) más amplio.
- La representación de los documentos y de las necesidades de informaciones (4).
- El proceso de comparación (5) entre las representaciones de las necesidades de información y las representaciones de los documentos del fondo documental, con el fin de determinar cuáles son los documentos más relevantes (6) en cada caso.
- La elección del formato más adecuado de visualización (7).
- Finalmente, este proceso tiene lugar en un contexto probabilístico o de descubrimiento (8).

Examinaremos ahora con más detalle los distintos elementos que forman parte de este sistema típico de RI según el modelo que hemos presentado en el cuadro anterior (cuadro 1 y 1a):

2.1. Necesidades de información

Una necesidad de información es, por definición, una entidad inobservable en sí misma, ya que consiste en un estado psicológico. Se supone que este estado psicológico es el inicio de todo el proceso: por algún motivo, un sujeto detecta en sí mismo lo que la teoría llama un "estado anómalo de conocimiento" o *ask* (de *anomalous state of knowledge*).

A partir de aquí, el individuo inicia una conducta de obtención de información seleccionando la fuente de información que considera más adecuada e inte-

rrogando o explorando dicha fuente para encontrar la información que pueda solucionar su estado de conocimiento anómalo o incompleto. Naturalmente, todo este proceso puede realizarse de manera directa o de manera mediada, por ejemplo, con la intervención de un profesional de la documentación.

2.2. Documentos

Un documento es información registrada. En el contexto de la RI se presupone que se trata siempre de documentos cognitivos, es decir, de documentos que contienen obras de creación u obras de pensamiento.

Dicho de otra manera, la RI no suele aplicarse a la gestión de documentos administrativos, ya que tendría un escaso sentido hacerlo. En general, los documentos administrativos se gestionan siguiendo procedimientos archivísticos basados en el uso de cuadros de clasificación y en el concepto de serie documental.

En cambio, como decimos, en RI se da por supuesto, aunque no siempre de forma lo bastante clara, que los documentos de los que hablamos son documentos sobre ciencia, tecnología, cultura, etc., es decir, la clase de documentos con un contenido mínimamente complejo como para justificar el uso de las no menos complejas tecnologías y procedimientos propios de la RI en particular y de la gestión documental en general.

2.3. Fondo documental

Las operaciones de RI tienen sentido en el contexto de un fondo documental no trivial. Cuando se trata de encontrar información en una colección compuesta por decenas de documentos, las operaciones de RI, en cambio, no tienen un significado especial, ya que podría explorarse todo el fondo documental de manera secuencial y, para hacerlo, no son precisos procedimientos ni técnicas especiales.

La RI empieza a tener sentido en colecciones que contienen miles de documentos, y tiene más sentido cuanto mayor sea el fondo. En el límite, la RI podría aplicarse al conjunto universal de todos los documentos producidos en algún momento por la humanidad, una perspectiva no tan fantástica como podría parecer a primera vista si reflexionamos sobre las futuras posibilidades de la WWW.

2.4. Representaciones de documentos y representaciones de necesidades de información

En un sistema de RI no se pueden (o no resulta conveniente) comparar directamente documentos y necesidades de información. En realidad, lo que se compara son representaciones de cada una de las dos entidades mencionadas. La

razón es que es ineficiente o simplemente imposible comparar de forma directa dos elementos de naturaleza heterogénea: recordemos que una necesidad de información es un estado psicológico, inobservable por definición, mientras que un documento es un conjunto de informaciones, de morfología variable, registrado en algún tipo de soporte material. Por lo tanto, para que la comparación sea posible es necesario convertir ambas entidades en una representación homogénea o que contenga elementos homogéneos.

Desde el punto de vista cognitivo, la representación de los documentos puede consistir en una ficha bibliográfica, articulada, por ejemplo, en una descripción formal tipo ISBD más una descripción característica formada por uno o más campos de descriptores.

Ahora bien, en el momento en que se representa así un documento, incluso un documento no textual como por ejemplo una imagen, entonces, desde el punto de vista del ordenador, esta ficha no es más que un conjunto de palabras o, más exactamente, de términos de indización. Formalmente, por lo tanto, en un sistema de RI un documento es un conjunto *D*, cuyos elementos son términos de indización, según este modelo general:

$$D_i = \{t_1, t_2, \ldots t_n\}$$

Donde $t_1, t_2, \ldots t_n$ son palabras simples (p.e., "economía") o compuestas (p.e., "economía política") que expresan las propiedades semánticas, es decir, el contenido temático, del documento D_i. Por ejemplo, supongamos para simplificar que el documento D_i contiene cinco temas relevantes, t_1, t_2, t_3, t_4, t_5; entonces la representación de D_i sería la siguiente:

$$D_i = \{t_1, t_2, t_3, t_4, t_5\}$$

Supongamos que el documento anterior trata de "la legislación sobre economía y trabajo en Cataluña y en España". Entonces, t_1, t_2, t_3, t_4, t_5 podrían corresponder, respectivamente, a:

Economía	(t_1)
Cataluña	(t_2)
España	(t_3)
Trabajo	(t_4)
Legislación	(t_5)

y por lo tanto, el documento podría representarse así:

$$D_i = \{\text{economía, Cataluña, España, trabajo, legislación}\}$$

La cuestión interesante aquí es que las necesidades de información también pueden representarse por palabras o términos de indización, según el modelo general:

$$P_j = \{t_1, t_2, \ldots t_n\}$$

que, como podemos ver, es idéntico a la forma en que representamos también los documentos. En concreto, supongamos que P_j representa la siguiente necesidad de información: "legislación sobre trabajo y mujeres en Cataluña". La representación de la pregunta P_j sobre la base de las palabras o términos de indización que la forman sería la siguiente:

P_j = {Cataluña, trabajo, legislación, mujeres}

En el siguiente punto relacionamos documentos y preguntas con un ejemplo simple de cálculo de relevancia.

2.5. Proceso de comparación

Uno de los dogmas centrales de la RI consiste en la idea de que, para seleccionar y extraer el documento más útil para solucionar una necesidad de información, hay que comparar las propiedades o características del documento con las propiedades o características de la necesidad de información. El documento que más se parezca a la necesidad de información será el más útil o, en términos técnicos, el más relevante.

Tal como hemos visto en el punto anterior, a partir de la forma de D_i y de P_j es fácil establecer una comparación entre los dos conjuntos (el conjunto de términos de indización que representa los documentos y el conjunto de términos de indización que representa las preguntas) y concluir que, en concreto, los dos conjuntos poseen tres elementos en común.

Supongamos que en la base de datos hubiera otros dos documentos con algunos elementos en común con la pregunta en cuestión, por ejemplo los documentos D_h y D_g. Supongamos que D_h sólo tiene dos elementos en común (dos términos de indización en común) y que D_g tiene, en cambio, cuatro elementos en común (cuatro términos de indización en común). Entonces, el subsistema de comparación del sistema de RI podría ordenar así los documentos, por orden decreciente de semblanza pregunta/documento:

1. D_g
2. D_i
3. D_h

Lo que hemos obtenido entonces es una ordenación de los documentos sobre la base del grado de probabilidad de cada documento de satisfacer la necesidad de información. Este grado de probabilidad se puede considerar una medida de relevancia, que se ha estimado a partir del número de elementos en común, o términos de indización, entre la necesidad de información y el documento.

Este modelo es muy simple, pero sustenta gran parte de los sistemas de RI que pueden encontrarse en el mercado, si bien también es cierto que la mayoría presenta importantes modificaciones sobre este modelo simple que dan mayor potencia al sistema.

2.6. Relevancia

La relevancia es una de las propiedades más interesantes de los documentos y, al mismo tiempo, una de las más difíciles de definir de manera operativa. Intuitivamente, podemos afirmar que un documento es mucho más relevante cuanto mejor puede solucionar una necesidad de información. Ahora bien, definida de esta manera, se pone de manifiesto que la relevancia no es una propiedad exclusiva del documento, sino, en realidad, un tipo de coproducción entre las características del documento, las características de la necesidad de información y las características de la persona que formula la pregunta.

Además, la relevancia tiene grados, ya que un documento no se limita a ser relevante o a no serlo, sino que la relevancia de un documento puede situarse en cualquier punto de un continuo entre 0 y 1, como 0,3, 0,6, 0,8 ó 0,9, por ejemplo, y en el que el 0 representa la ausencia de relevancia y el 1 la relevancia total. Naturalmente, nada nos impide representar esta misma escala con los límites de 0 y 1000; de 0% y 100%, etc.

El punto importante aquí es que si diferentes documentos poseen diferentes grados de relevancia ante una pregunta o una necesidad de información, entonces no tiene mucho sentido entregar los documentos en respuesta a esta pregunta de manera aleatoria o por un orden no muy significativo desde un punto de vista semántico, como el título o la fecha de creación, y eso era justamente lo que hacían la inmensa mayoría de los sistemas de gestión documental antes de Internet y lo que todavía hacen algunos sistemas.

Por el contrario, una vez aceptado el principio teórico de la relevancia, lo que hacen los mejores sistemas de RI es intentar calcular de la manera más eficiente posible el grado de relevancia de cada documento ante una necesidad de información y entregarlos al usuario ordenados según este grado. De hecho, en grandes fondos documentales, la eficiencia del cálculo de relevancia es un factor crítico que determina la calidad total del sistema. ¿Qué importancia tiene que una respuesta a una necesidad de información contenga una lista de diez mil documentos si los documentos relevantes están distribuidos de manera aleatoria entre estos diez mil documentos? El usuario no dispone de ninguna forma de saber cuándo tiene que detener su exploración de este conjunto de diez mil documentos, dado que el documento más relevante podría ser, precisamente, el último de la lista.

2.7. Descubrimiento

Resulta difícil apreciar la naturaleza de la RI sin entender la siguiente cuestión: la RI no sirve –necesariamente– para saber más cosas de una entidad previamente conocida, sino para *descubrir* qué entidades cumplen una condición.

Sin entender esta diferencia no se puede entender cuál es, entonces, la aportación específica de un *software* documental comparado con un *software* ofimático están-

dar. En concreto, es imposible distinguir entre un sistema de gestión de bases de datos documental y un sistema de gestión de bases de datos relacional. Otra forma de enfocar este punto consiste en señalar que el entorno de trabajo típico del *software* ofimático es de tipo determinista, es decir, se sabe siempre qué se quiere y se sabe que tales acciones producirán siempre tales resultados. En cambio, en el entorno típico de la RI no siempre se sabe qué se quiere, ni siquiera se sabe si habrá entidades que puedan satisfacer las condiciones indicadas en la petición de información.

La petición de información típica de un entorno ofimático sigue este tipo o modelo general: "qué valor asume la variable *V* de la entidad *E*, previamente conocida". Por ejemplo, "cuál es el importe total de las ventas del mes de abril de la delegación de París". El valor que se quiere saber es "el importe total"; la variable que tiene este valor es "las ventas del mes de abril", y la entidad previamente conocida es "la delegación de París". Aquí tenemos un entorno determinista: dada la clase de pregunta, siempre tiene que haber una respuesta y una única respuesta.

La petición de información típica de un entorno a RI sigue, en cambio, a este otro modelo general: "qué entidades, E_1, E_2... E_n, desconocidas por definición, son susceptibles de satisfacer la condición *C* o el complejo de condiciones C_1, C_2... C_n". Por ejemplo, "qué documentos pueden solucionar una necesidad de información sobre psicología y cine".

Las entidades desconocidas son por definición los hipotéticos documentos relevantes, y el complejo de condiciones que tienen que satisfacer los documentos para ser considerados relevantes son, en este caso, tres: tratar de psicología (1), tratar de cine (2) y que la relación lógica entre (1) y (2) sea la que se expresa con un AND booleano (3).

Aquí tenemos un típico entorno probabilístico: puede haber o no una respuesta, y en caso de haberla no tiene por qué ser necesariamente única, sino que lo más habitual es que haya una colección de documentos (respuestas) diferentes, cada uno con un grado de relevancia diferente. Finalmente, aunque el sistema sea capaz de entregar documentos relevantes, eso puede significar que, en lugar de solucionar de manera definitiva la necesidad de información, se planteen al usuario nuevos interrogantes, por lo tanto, nuevos "estados anómalos de conocimientos", la necesidad de efectuar nuevas operaciones de RI, etc.

2.8. Presentación y visualización de la información

Una vez seleccionados y ordenados los documentos por su grado de relevancia, el sistema de RI puede tener uno o más formatos de presentaciones, llamados habitualmente *vistas*.

Cada vista puede representar los intereses o las necesidades de varios grupos de usuarios, o varios estilos de visualización. Por ejemplo, en el primer sentido es habitual que haya una vista para los administradores del sistema, otra para usuarios finales, etc.

Algunos motores de búsqueda de Internet permiten elegir entre respuestas resumidas o detalladas (podéis ver, por ejemplo, HotBot, –http://www.hotbot.com–). En bases de datos como la Special Collections de NL Search (http:/ www.nlsearch.com), hay tres vistas diferentes de los documentos, según la fase de la búsqueda, y es más detallada cada vez hasta llegar al documento completo en la última fase.

Algunos bancos de imágenes también permiten elegir el formato de visualización de las imágenes recuperadas, aunque sea para elegir entre las dimensiones y el número de imágenes que tiene que presentar el sistema de manera simultánea (consultad, por ejemplo, Corbis, http://www.corbis.com).

Por su parte, las técnicas de visualización de la información consisten en mostrar de forma gráfica informaciones que son inherentemente textuales. Por ejemplo, la empresa Cartia (www.cartia.com) ha desarrollado un sistema para representar en forma de mapa espacial los temas de cualquier grupo de documentos y lo han aplicado a varios ámbitos, uno de los cuales es la información de prensa (http://www.newsmaps).

La empresa Inxight (www.inxight.com) ha producido una interfaz de visualización, llamada Hiperbolic, que puede aplicarse a fondos documentales. Se puede ver una demostración aplicada a la base de datos de fuentes de información de Lexis-Nexis (www.lexis-nexis.com/lncc/hyperbolic/).

3. Algoritmos básicos de la RI

3.1. Inteligencia aparente

Como es bien sabido, los sistemas informáticos ni entienden ni pueden interpretar el significado de los textos y, a pesar de todo, los sistemas informáticos de RI desarrollan tareas que simulan inteligencia o, por lo menos, algún grado de comprensión del significado de la información textual.

Esto es posible porque, en general, la capacidad de los ordenadores para resolver cualquier tarea o cualquier problema, desde el más simple hasta el más complicado, está basada en lo mismo: el descubrimiento o la determinación de un procedimiento que permita descomponer los pasos necesarios para la resolución de la tarea en un número finito de suboperaciones, en las que cada una no requiera ninguna inteligencia ni, por lo tanto, ninguna capacidad de comprensión o de interpretación de nada, ni de la información textual ni de la información de cualquier otro tipo.

3.2. Algoritmos

Donde sí hay inteligencia, y mucha, es en la persona o en el equipo de personas que han sabido descomponer la resolución de un problema en este núme-

ro finito de pasos al que nos referimos y que, en matemáticas y en ciencias de la computación, tiene un nombre concreto: algoritmo.

Por lo tanto, podemos definir un algoritmo como un método de resolución de problemas que consta de un número finito de pasos bien enunciados. En matemáticas, el procedimiento para resolver una suma, una raíz cuadrada o una división son ejemplos de algoritmos.

En informática, cualquier programa de ordenador consiste en uno o más algoritmos, codificados en un lenguaje de programación que un ordenador pueda leer. Por lo tanto, antes de que un programador pueda escribir un programa, es preciso que alguien, este mismo programador u otro, haya encontrado el algoritmo para resolver el problema que el programa informático tratará de solucionar.

En RI hay un buen número de algoritmos que se han ido descubriendo y perfeccionando desde hace unos treinta años. Estos algoritmos suelen presentarse bajo su forma lógica más abstracta, es decir, de forma independiente de su implementación en lenguajes de programación concretos, y así es como los presentaremos también aquí.

Aquí examinaremos algoritmos para la indización automática de documentos y para el cálculo de relevancia. Ahora bien, el lector debe entender, por lo tanto, que tal como se presentan estos algoritmos no podrían implementarse en ningún ordenador, sino que antes sería necesario traducirlos a alguno de los lenguajes de programación existentes, por ejemplo, a Visual Basic, Java, C, etc.

4. Evaluación de sistemas de RI

Sin embargo, antes de entrar en consideraciones sobre la indización automática es necesario que dediquemos un tiempo a ver cómo se evalúa el rendimiento de los sistemas de RI.

Las dos medidas más utilizadas suelen ser la tasa de recordación (*recall*) y la tasa de precisión (*precision*).

Las fórmulas son las siguientes:

$$\text{Recordación} = \frac{\text{Número de documentos relevantes recuperados}}{\text{Número total de documentos relevantes presentes en el fondo documental}} \times 100$$

$$\text{Precisión} = \frac{\text{Número de documentos relevantes recuperados}}{\text{Número total de documentos recuperados}} \times 100$$

Ejemplo para la tasa de recordación

Supongamos que en una colección hay diez documentos relevantes sobre el tema *X*, y que, como consecuencia de una operación de recuperación de información sobre el tema *X*, por la razón que sea (mala indización del fondo, mala indización de la pregunta, insuficiencias del lenguaje documental, etc.), se obtienen sólo seis documentos; entonces la fórmula anterior nos indica que la tasa de recordación en este caso ha sido del 60%. Reiteradas medidas del rendimiento de este fondo con la consiguiente media nos darían la tasa de recordación global del fondo.

Ejemplo para la tasa de precisión

Supongamos que se han obtenido diez documentos en respuesta a una operación de recuperación de información, pero que cinco de estos documentos no eran relevantes. Entonces, la tasa de precisión sería del 50%.

La tasa de recordación proporciona una medida de la habilidad del sistema para recuperar documentos relevantes, mientras que la tasa de precisión proporciona una medida de la habilidad del sistema para evitar el ruido.

Naturalmente, el objetivo consiste en diseñar sistemas que proporcionen al mismo tiempo un 100% de recordación y un 100% de precisión, es decir, sistemas que recuperen todos los documentos relevantes y sólo los documentos relevantes, pero en la práctica estos dos indicadores se comportan de manera antagónica, ya que las medidas para incrementar la recordación tienden a reducir la precisión, y viceversa.

La razón es la siguiente: si queremos asegurar la precisión del sistema adoptaremos medidas que tiendan a aumentar la especificidad de la indización. Por ejemplo, si un documento trata sobre “gladiolos”, entonces diseñaremos un sistema de indización que tienda a indizar el documento con el descriptor “gladiolos”, y no con el descriptor “flores” y ni mucho menos con el descriptor “plantas” o “jardines”, etc. De esta manera tendremos un sistema muy preciso, pero cuando alguien pida documentos sobre “flores” no recuperará documentos relevantes sobre el tema.

En general, podemos observar que los motores de búsqueda generalistas que funcionan en Internet, como AltaVista o HotBot, proporcionan altas tasas de recordación, es decir, probablemente tienden a recuperar muchos documentos relevantes, pero como es fácil de comprobar, la tasa de precisión es extremadamente baja, dado que una parte ínfima de los documentos recuperados son relevantes.

En cambio, los sistemas muy especializados, como las agencias de selección y evaluación de recursos digitales tales como BUBL (www.bubl.ac.uk), ADAM (www.adam.ac.uk) o Cercador (www.cercador.com) dan un rendimiento inverso. A cada petición de información proporcionan muchos menos recursos

y, por lo tanto, probablemente, tasas de recordación muy bajas, pero las tasas de precisión se aproximan en gran medida al 100%.

También resultan útiles para discutir los problemas de evaluación de los sistemas de RI los conceptos, adoptados de la teoría estadística, de los falsos positivos y de los falsos negativos.

Un documento es un falso positivo cuando se recupera pero no es relevante, es decir, se ha recuperado *de facto* pero no tendría que haberse recuperado, ya que no es realmente relevante.

Un documento es un falso negativo cuando, aun siendo relevante, no se recupera. Es decir, no se ha entregado al usuario aunque es un documento relevante.

Los motivos de los rendimientos inadecuados en las tasas de recordación y precisión, y por lo tanto en el fenómeno de los falsos positivos y de los falsos negativos, son varios, pero pueden señalarse cuatro factores, los tres primeros propios de entornos donde se realiza una indización de tipo intelectual o mixto y el cuarto en entornos de indización automática pura. Son los siguientes:

a) Indización deficiente del documento

Por ejemplo, el documento trataba del tema *X* pero, en cambio, por error no se ha asignado este descriptor. El documento no se recuperará cuando se pida información sobre *X*. El caso contrario: un documento en realidad no trata el tema *Y*, pero le ha sido asignado el descriptor y, por lo tanto, proporcionará ruido cuando alguien pida información sobre *Y*.

b) Indización deficiente de la necesidad de información

La indización de las necesidades de información presenta el mismo problema. Tal vez el usuario desconoce que el tema para el que está buscando información se representa con el descriptor *X*, por lo que utiliza un descriptor menos adecuado, por ejemplo más general, y eso le proporcionará una tasa muy baja tanto de precisión como de recordación, etc.

c) Grado insuficiente de especificidad del lenguaje documental

El lenguaje documental utilizado en la representación de los documentos puede ser inadecuado. Por ejemplo, podría haber varios documentos en el fondo documental sobre “gladiolos”, “rosas”, “amapolas”, etc., pero el lenguaje documental sólo tiene en cuenta el descriptor “flores”, o peor todavía, “plantas”, con lo que los documentos no quedan representados en su nivel de especificidad adecuado.

d) Deficiente algoritmo de relevancia

Si el sistema entrega muchos documentos como respuesta a la pregunta, entonces el rendimiento final de la calidad del sistema vendrá determinado por el acierto en el cálculo de relevancia. Siempre que el sistema entregue más de cincuenta documentos, la relevancia será un factor esencial.

Supongamos que se han utilizado los términos *X*, *Y* para indizar la pregunta, y supongamos que el cálculo de relevancia otorga un gran peso, es decir, un valor positivo, a los documentos que tienen muchas veces cualquiera de los dos términos o los dos términos a la vez, pero sin discriminar entre estas dos posibilidades.

El documento más relevante podría tener pocas ocurrencias de *X* y pocas ocurrencias de *Y*, por ejemplo, a causa de la creatividad del autor, quien tal vez posee un amplio vocabulario. Como resultado, el sistema podría desplazar el documento más relevante a las últimas posiciones de la lista y privilegiar documentos donde sólo *X* (pero no *Y*) aparece muchas veces. Éste, por ejemplo, es uno de los síndromes habituales de los motores de búsqueda de Internet.

5. Indización automática

El objetivo de los procedimientos de indización automática es imitar lo mejor posible la indización intelectual (indización humana). La indización intelectual se caracteriza por la capacidad de trabajar en el nivel de los conceptos, mientras que la indización automática trabaja, en principio, en el nivel de cadenas de caracteres.

5.1. Conceptos frente a cadenas de caracteres

Es decir, para un indizador humano, las expresiones (1) "aumento de precios en un periodo determinado"; (2) "índice de carestía" y (3) "incremento periódico de precios" significan lo mismo, al menos desde el punto de vista de la indización documental y, por lo tanto, un indizador humano no tiene ningún problema para realizar una igualdad entre los tres términos anteriores –(1), (2), (3)– y el término (4) "inflación". Por lo tanto, para un indizador humano, la relación entre los términos anteriores es una igualdad de este tipo:

(1) = (2) = (3) = (4)

En virtud de ésta, el término (4), por ejemplo, puede ser declarado término preferido y, por lo tanto, descriptor autorizado para representar este concepto.

A partir de este momento, la aparición de las expresiones (1), (2), (3) u otras semánticamente equivalentes en un documento permite al indizador humano realizar la inferencia válida de que el documento se debe indizar con el descriptor (4) "inflación", aunque esta palabra, *inflación* (es decir, esta cadena de caracteres, desde la lógica del ordenador), no aparezca en el documento.

En cambio, para un ordenador, lo significativo son las cadenas de caracteres, por lo que la relación entre (1), (2), (3), (4) es la de una desigualdad simétrica entre todos ellos.

5.2. Documento ejemplo

Partiremos de un documento ejemplo sencillo, que llamaremos **Doc1**, y de un ejemplo de indización intelectual de este documento para discutir el posible rendimiento de los distintos procedimientos de indización automática.

Figura 1. Documento ejemplo Doc1

La información como propiedad

Por Dr. Lluís Codina

La información no es una sustancia ni un objeto, sino una propiedad de los mensajes bien formados, a saber, la propiedad de dar a conocer algún aspecto de la realidad.

En este sentido, estamos de acuerdo con la teoría de la información de Alfred Dretske, según la cual, en realidad, una información falsa no es una información, en el mismo sentido que un pato de madera no es un pato.

Por este motivo, podemos decir también que, en el contexto de la teoría de los símbolos, los mensajes son una clase de sistemas de información.

Estadísticas del documento

- Número total de palabras: 100
- Número total de palabras únicas: 55

5.3. Indización intelectual

A partir de un hipotético documento como éste, una indización intelectual típica para representar el documento sería como la que presenta la figura 2:

Figura 2. Descriptores asignados al documento Doc1 con indización intelectual

1. Información
2. Mensajes
3. Teoría de la información
4. Semiótica
5. Sistemas de información
6. Lluís Codina
7. Alfred Dretske

A un indizador humano, al menos para un indizador entrenado, le resulta fácil identificar tanto los descriptores simples como los compuestos ("información" frente a "sistemas de información"), y también asignar un descriptor por inferencia y no por mera transcripción de palabras ("semiótica", como resul-

tado de la expresión teoría de los símbolos); finalmente, el indizador humano no se deja engañar y no asigna el descriptor "patos", a pesar de que el término aparece dos veces en el texto del documento.

En conclusión, un indizador humano, de manera rutinaria:

a) detecta tanto descriptores simples como compuestos;

b) asigna descriptores, aunque la palabra no aparezca en el documento;

c) no asigna descriptores, aunque la palabra esté presente en el documento.

En cambio, para un ordenador, conseguir *a*, *b* y *c* sería una auténtica proeza. Pese a todo, más adelante veremos que los ordenadores pueden conseguir un resultado bastante parecido.

5.4. Indización automática I. Indización simple

La indización que realizaría una máquina podría ser de tres tipos básicos, cada uno de éstos según algoritmos sucesivamente más sofisticados. Los veremos a continuación.

En este punto examinaremos el llamado algoritmo simple, que aparece representado en el cuadro siguiente:

Algoritmo 1. Modelo de indización simple

1. Identificar cuáles son las cadenas de caracteres únicas del documento.
2. Cada cadena de carácter única es un término de indización.
3. Asignar cada una de las cadenas de caracteres únicas al documento como un término de indización del documento.

El algoritmo precedente es de una gran simplicidad conceptual, pero su implementación es engañosamente simple. En primer lugar, aquí hemos obviado algunas cuestiones, ya que son rutinarias en programación, tales como prever cómo se iniciará y cómo se acabará el proceso, cuál será la entrada de la información, cuál será la salida, etc., pero que habrá que tener presentes en el momento de escribir el programa correspondiente.

En segundo lugar, habrá que especificar en el programa informático qué se considerará que es una cadena de caracteres y qué no lo es, por ejemplo:

a) La expresión "sistema de información", ¿es una, son dos o son tres cadenas de caracteres?

b) ¿Los espacios en blanco y los signos de puntuación son siempre separadores de cadenas de caracteres? Por ejemplo, ¿el punto (.), la barra (/), el guión (-) son

siempre separadores de cadenas de caracteres? Por lo tanto, ¿expresiones como "E.U." serán una o dos cadenas de caracteres? ¿Qué pasará con datos como "01-10-1999", o con expresiones como "importación/exportación"?, etc.

c) Será preciso especificar qué es una cadena de caracteres única. En el caso más simple, son cadenas o términos únicos las cadenas idénticas. "Información", por ejemplo, aparece varias veces en el texto, pero es una misma cadena, por lo tanto, es un término único; sin embargo, ¿qué pasaría con "información" e "informaciones"? ¿Son uno o dos términos únicos?

Por tanto, aunque no sea evidente a primera vista, incluso un algoritmo conceptualmente tan simple como el algoritmo 1 requiere un cierto análisis, ya que, como ya hemos indicado, se trata de que una máquina que no puede interpretar las palabras sea capaz, en cambio, de identificarlas en un texto gracias a instrucciones del estilo: "toda cadena de letras entre espacios en blanco es una cadena de caracteres; toda cadena de letras entre un espacio en blanco y un punto (.) es una cadena de caracteres", etc.

Cada una de las diferentes palabras de un documento o de una base de datos recibe el nombre de palabra única.

En cualquier caso, la indización que produciría un algoritmo simple de indización coincidiría con el resultado de la figura 4, es decir, los términos de indización asignados coincidirían con la lista de palabras únicas del documento, tal como se presenta en la figura siguiente:

Figura 3. Resultado de la indización del documento Doc 1 con un algoritmo simple (palabras únicas del documento)

a	dr	por
alfred	dretske	podemos
alguno	el	propiedad
con	los	cual
pato	en	que
éste	es	realidad
aspecto	estamos	saber
bien	falsa	según
clase	formados	sentido
codina	madera	símbolos
como	la	sino
conocer	lluís	sistemas
contexto	mismo	son
acuerdo	mensajes	sustancia
información	motivo	también
de	ni teoría	
de los	no	un
decir	objeto	una
dar		

Podemos observar diferentes aspectos de esta clase de indización:

En primer lugar, se ha multiplicado el número de términos de indización asignados al documento. Hemos pasado de los siete términos de la indización intelectual a cincuenta y cinco con indización automática simple.

En segundo lugar, y como consecuencia directa de lo anterior, este documento tendrá muchas más posibilidades de ser recuperado, pero en muchas de estas posibilidades será un falso positivo, es decir, proporcionará ruido. El caso más evidente será si alguna vez este documento se recupera como consecuencia de una pregunta sobre patos.

En tercer lugar, y en contraste con lo anterior, este documento será un falso negativo cada vez que algún usuario pida documentos sobre "semiótica", dado que este término no aparece en el texto y, por lo tanto, el sistema automático de indización no ha podido identificar el concepto.

En cuarto lugar, a causa del algoritmo utilizado, se ha perdido mucha información, ya que este algoritmo sólo ha sido capaz de identificar palabras simples, como "información", pero no como "sistema de información" o como "Alfred Dretske".

Aunque, como decimos, este algoritmo parezca muy simple e, incluso, de resultados muy limitados, es uno de los más utilizados actualmente. Es el que utilizan muchos motores de búsqueda de Internet, y el más implementado en la mayor parte del parque de los sistemas de gestión documental de las empresas.

También hay que indicar que, a menudo, este algoritmo de indización automática se complementa con una indización intelectual, con lo que el resultado final sería, en realidad, una combinación de términos de indización de la figura 2 y la figura 3. A pesar de todo, ésta no es la práctica mayoritaria en las empresas, sino en centros de documentación y bibliotecas. Por lo tanto, en muchas empresas, el rendimiento máximo de sus sistemas de RI es el que da el algoritmo que hemos discutido aquí.

Un programa muy representativo de este algoritmo sería el sistema de gestión de bases de datos File Maker (www.filemaker.com), muy popular como solución departamental, también en pequeñas y medianas empresas y en algunos centros de documentación.

5.5. Indización automática II. Indización adelantada

El algoritmo que discutiremos a continuación presenta una importante mejora en relación con el anterior, y en la figura siguiente indicamos sus características (seguimos, sobre todo, el modelo de Gerard Salton).

Algoritmo 2. Modelo de indización adelantada

1. Identificación de las cadenas de caracteres para determinar la primera lista de candidatos a términos de indización.
2. Eliminación de las palabras vacías de esta lista, es decir, de los términos muy frecuentes.
3. Creación de raíces con las cadenas de caracteres para crear los términos de indización.
4. Mezcla de términos sinónimos.
5. Cálculo de frecuencias absolutas.
6. Cálculo del peso o importancia de los términos en cada documento.
7. Eliminación, como candidatos a descriptores, de los términos con un índice de discriminación que quede por debajo de un umbral determinado.
8. Asignación de los descriptores ponderados a cada documento.

En este algoritmo, el primer paso es idéntico al anterior y los problemas que hay que resolver en su implementación son exactamente los mismos, a saber, será preciso determinar algún procedimiento eficiente para determinar de manera correcta qué es y qué no es una cadena de caracteres válida. En el segundo paso, en cambio, ya encontramos una operación nueva: la eliminación de las llamadas palabras vacías (*stop words*).

Las palabras vacías son palabras con una frecuencia tan elevada que no tienen ninguna capacidad para discriminar documentos y, por lo tanto, es mejor retirarlas de entrada de la lista de candidatos a descriptores. Determinar cuáles son las palabras vacías en cada caso puede hacerse de dos formas diferentes: *a priori*, *a posteriori* y, cómo no, con una combinación de los dos métodos.

En el método *a priori* un operador humano introduce en el sistema una lista, llamada a veces diccionario de palabras vacías, que contiene todas aquellas partes de una lengua que tienen una función gramatical pero un pobre significado semántico independiente, como por ejemplo pronombres, artículos, adverbios, etc. Para muchas lenguas, incluyendo el castellano, el catalán y el inglés, suelen aparecer unas trescientas palabras.

En el método *a posteriori*, las palabras vacías se determinan por cálculo de frecuencia. De esta manera, se retiran de la lista de candidatos todas aquellas palabras que aparecen, por ejemplo, en más del 80% de los documentos. Así se detectan palabras vacías que de otra manera pasarían desapercibidas. Por ejemplo, en un fondo documental sobre economía, probablemente convendrá considerar el término "economía" como palabra vacía. Naturalmente, nada impide combinar los dos métodos.

Según Salton, de esta manera la lista inicial de términos candidatos queda reducida en un 40% o un 50%. En nuestro caso, de 55 palabras pasamos a 29, es decir, efectivamente se ha producido una reducción de un poco más del 40%, como podemos ver en la figura 4.

Figura 4. Primer grupo de candidatos a descriptores: resultado de la eliminación de las palabras vacías de la lista inicial del documento Doc1

alfred	dr	propiedad
pato	dretske	realidad
aspecto	falsa	saber
bien	formados	según
clase	madera	símbolos
codina	lluís	sistemas
conocer	mismo	sustancia
contexto	mensajes	teoría
información	motivo	
decir	objeto	
dar		

El tercer paso consiste en fusionar los términos que tienen las mismas raíces. De esta manera, si, por ejemplo, en el documento hubiera dos palabras como "información" e "informaciones", quedarían reducidas a una sola forma: "informacio*" (donde el asterisco indica un truncamiento).

El cuarto paso consiste en detectar posibles sinónimos. Por ejemplo, si en el documento tuviéramos dos palabras como "ordenador" y "computador", en este paso quedarían fusionadas en una única palabra a efectos del cálculo de frecuencia del que hablaremos a continuación. Es decir, se consideraría que, en lugar de dos palabras, habría un mismo término con dos ocurrencias. Este paso puede resolverse con el uso de un tesauro o con una lista de sinónimos.

En el quinto paso se realiza el cálculo de las frecuencias absolutas de cada uno de los términos de la lista resultante. Éste es un paso previo al cálculo del peso o índice discriminatorio de cada término.

Según este índice, los diferentes términos de un documento pueden tener una capacidad discriminatoria diferente, que indica la posible utilidad de cada término como descriptor. Un término es mucho mejor descriptor cuanto mejor sirve para discriminar grupos de documentos. Por ejemplo, un término como "sistema" es probablemente un mal descriptor, dado que debe de estar presente en un gran número de documentos y, por lo tanto, tiene un índice de discriminación muy bajo. En cambio, probablemente, el término "teoría de sistemas" tiene un índice de discriminación mucho más elevado.

Por lo tanto, en el paso sexto se calcula el índice de discriminación o peso de cada término de la lista de descriptores. El cálculo que propone Salton, y que siguen varios sistemas de indización automática, es el siguiente:

ft × fid = índice de discriminación del término

donde

ft = Frecuencia absoluta del término en el documento
fid = Frecuencia inversa del documento

La frecuencia absoluta es el número de veces que aparece el término en el documento. Por ejemplo, en nuestro caso, la lista de frecuencias absolutas es la siguiente:

Figura 5. Frecuencias absolutas de los términos candidatos a descriptores del documento Doc1

alfred	1	dr	1	propiedad	3
pato	2	dretske	1	realidad	2
aspecto	1	falsa	1	saber	1
bien	1	formatos	1	sentido	2
clase	1	madera	1	símbolos	1
codina	1	lluís	1	sistemas	1
conocer	1	mismo	1	sustancia	1
contexto	1	mensajes	2	teoría	2
información	7	motivo	1		
decir	1	objeto	1		
dar	1				

Sólo con esta lista ya se puede apreciar que los términos más frecuentes corresponden bastante con el tema de los documentos y, por lo tanto, si adoptáramos como descriptores todos los términos de frecuencia superior a 1, por ejemplo, no quedaría una mala representación del documento, como podemos ver (indicamos ahora la frecuencia en la izquierda):

7 información
3 propiedad
2 pato
2 mensajes
2 realidad
2 sentido
2 teoría
(todos los otros términos son de frecuencia = 1)

Ahora bien, el sexto paso no se limita a adoptar la frecuencia absoluta como indicador de la adecuación de un término como descriptor, sino que, como hemos visto por la fórmula anterior, relaciona esta frecuencia con la denominada frecuencia inversa del documento (fid). Ésta se calcula así:

$$\text{fid}(i) = \frac{\text{Número de documentos en el fondo documental}}{\text{Número de documentros con el término } i}$$

donde fid(*i*) significa que la frecuencia inversa del documento para el término *i* se obtiene dividiendo el número total de documentos de la base de datos por el número de documentos que tienen el término *i*.

En realidad, Salton recomienda, por razones de comodidad para la manipulación del resultado, una pequeña variante:

$$\text{fid}(i) = \frac{\log 2\ (\text{número de documentos en el fondo documental})}{\log 2\ (\text{número de documentros con el término } i) + 1}$$

En cualquier caso, la fid de un término sirve para indicar su peso relativo, ya que relaciona su frecuencia en todo el fondo documental con el número total de documentos. Multiplicando este factor (fid) de cada término (que es una medida global) con la frecuencia absoluta en el documento (que es una medida local) se pretende lo siguiente: otorgar más peso a los términos que tienen una alta presencia local y una baja presencia global. Por ejemplo, si el término "información" tiene una presencia muy alta en el documento, pero también tiene una frecuencia muy alta en todo el fondo documental, podría obtener un peso relativo más bajo que el término "propiedad", el término "mensajes" o el término "dretske".

En el séptimo paso, los candidatos a descriptor con un índice de discriminación por debajo de un determinado umbral quedarían eliminados. Este índice tiene que establecerse de forma empírica según las características de cada fondo. Podemos suponer que, de la lista de los 29 descriptores, probablemente una tercera parte quedarían excluidos como candidatos a descriptores.

A partir de aquí (paso octavo) es imposible saber cómo quedaría esta lista, ya que el cálculo depende de las características concretas del fondo del que forme parte, pero tal vez podría parecerse a algo como esto:

Figura 6. Lista hipotética de descriptores del documento Doc1, con el algoritmo 2

alfred
patos
contexto
información
dretske
mensajes
objeto
propiedad
realidad
saber
sentido
símbolos
sistemas
sustancia
teoría

Finalmente, además, cada descriptor quedaría asignado al documento con un índice numérico de su peso o capacidad discriminatoria como tal descriptor que podría utilizarse después en el cálculo de relevancia del documento. Este índice, resultado del cálculo del sexto paso, podría ser un número entre 0 y 1, de manera que, por ejemplo, el descriptor "información" podría tener un índice de 0,4, mientras que el descriptor "mensaje" podría tener un índice de 0,5, etc.

Es un resultado bastante mejor que el que daba al modelo simple de indización automática, pero no es mejor todavía que la indización intelectual.

Básicamente, persisten los mismos problemas: este procedimiento no reconoce unidades superiores a la palabra simple (no reconoce "teoría de la información) y, probablemente, el término "pato" quedaría asignado como un descriptor a este documento, que no trata en absoluto de patos.

Numerosos motores de búsqueda de Internet parecen aplicar un algoritmo como éste, o muy parecido, en su procedimiento de análisis e indización automática, aunque nunca es posible estar completamente seguros desde el momento en que las empresas que administran estos motores no proporcionan los detalles exactos de sus algoritmos.

5.5.1. Variaciones sobre el algoritmo 2

Ahora bien, según Salton, hay posibilidad de añadir uno o dos pasos más al algoritmo 2 que estamos examinando ahora y que todavía podrían mejorar el resultado. En concreto, en algunas ocasiones Salton ha presentado un modelo de indización automática que incorpora el señalado aquí como 5a y 6a y que destacamos en cursiva):

Algoritmo 2a. Modelo de indización adelantada. Segunda variación

1. Identificación de las cadenas de caracteres, para determinar la primera lista de candidatos a términos de indización.
2. Eliminación de las palabras vacías de esta lista, es decir, de los términos muy frecuentes.
3. Creación de raíces con las cadenas de caracteres para crear los términos de indización.
4. Mezcla de términos sinónimos.
5. Cálculo de frecuencias absolutas.

5a. *Eliminación de términos poco frecuentes.*

6. Cálculo del peso o importancia de los términos en cada documento.

6a. *Formación de frases (descriptores compuestos) con términos muy frecuentes, mediante cálculo de concurrencias de términos en las diferentes oraciones del documento.*

7. Eliminación, como candidatos a descriptores, de los términos con un índice de discriminación que quede por debajo de un umbral determinado.
8. Asignación de descriptores ponderados a cada documento.

Se supone que, gracias al paso 5a se eliminaría de los candidatos a descriptores un término como "patos". Ahora bien, esto sería cierto siempre que habláramos de un fondo documental especializado y en el que, por lo tanto, términos ajenos a la especialidad del fondo no aparecerían con frecuencia. Si suponemos que estamos hablando de un fondo especializado en temas de información y comunicación, entonces es plausible suponer que el término "pato" sería muy infrecuente y quedaría, por lo tanto, eliminado. Sin embargo, sólo es una hipótesis que, en todo caso, seguro que en un fondo indiscriminado en su conjunto como Internet no se cumple.

Por su parte, gracias al paso 6a, se supone que, también en condiciones ideales, saldrían descriptores compuestos como "sistemas de información". Ahora bien, igual que en el caso anterior, sólo es una hipótesis que a veces se cumple y a veces no, según las características del fondo; en todo caso, no se cumple siempre al 100%.

Sea como sea, en el caso más favorable, ahora el resultado que tendríamos, si aplicáramos el algoritmo 2a, podría ser el siguiente:

Figura 7. Lista hipotética de descriptores del documento Doc1, con el algoritmo 2 a

contexto
información
mensajes
objeto
propiedad
realidad
saber
sentido
símbolos
sistemas de información
sustancia
teoría de la información

Las observaciones que podemos hacer a este resultado son las siguientes: en primer lugar, ha mejorado en el sentido de que ha eliminado algunos términos inadecuados, como el famoso "pato" (pero recordemos que esto sólo es una hipótesis). En segundo lugar, ha añadido algunos términos compuestos, como "sistemas de información" y "teoría de la información", que sin duda mejoran la indización. Ahora bien, por los mismos principios según los cuales han desaparecido algunos descriptores inadecuados, también podrían desaparecer los descriptores "Alfred" y "Dretske". Finalmente, no es plausible, por lo menos sin el uso de un tesauro externo, que el descriptor "semiótica" quedara asignado al documento.

5.6. Indización automática III. Indización inteligente

Para que la indización automática alcance un mejor rendimiento, quedaría añadir al procedimiento avanzado algunas operaciones y mejoras que podrían conducir a una indización no ya adelantada, sino inteligente.

Ahora bien, todo lo que se dirá a partir de ahora existe sólo o bien en sistemas propietarios que, por alguna razón, no han llegado al mercado como soluciones estandarizadas, o bien en productos de tipo experimental.

Parece que la mejora de los procedimientos de análisis e indización documental tendría que provenir de combinar dos herramientas más en este tipo de procesos:

1. Instrumentos de análisis lingüístico
2. Sistemas expertos
3. Tesauros

Los instrumentos de análisis lingüístico permitirían detectar candidatos a descriptores con más fundamento que los simples datos estadísticos de los términos, aunque éstos continuarían siendo útiles. Por ejemplo, con técnicas de lingüística computacional y terminología podrían detectarse candidatos a descriptores formados no sólo por palabras simples, como "información", sino también por palabras compuestas, como "sistemas de información", a partir de la determinación de las características sintácticas, semánticas y morfológicos de los textos y de reglas de formación de expresiones gramaticalmente válidas, y no sólo sobre la base de propiedades estadísticas de los textos.

Por su parte, un sistema experto podría aplicar reglas de producción del estilo "si... entonces...", para asignar descriptores de un tesauro o identificar sinónimos también con la ayuda de un tesauro. Por ejemplo, una regla de producción del sistema experto podría servir para deducir que:

si <el término "diafragma" aparece en un contexto próximo al término "óptica">, entonces, <el documento se puede indizar con el término "diafragmas ópticos">.

En caso necesario, el uso de un tesauro como parte integrante del sistema experto ayudaría a formar clases de sinonimia y a elegir en cada caso el término preferido como descriptor, así como a elegir el término más adecuado según el nivel de especificidad, etc.

O bien podría aplicar reglas que determinaran que "Alfred Dretske" es un nombre propio que identifica a un autor y que este autor es bastante relevante para ser utilizado como descriptor. Por ejemplo, reglas según las cuales:

si <dos cadenas conexas empiezan con mayúsculas> y <van precedidas de la expresión "según", entonces <se trata de un nombre propio y el documento puede indizarse con este nombre propio>.

6. Conclusiones

En relación con la indización automática de documentos, Internet ha demostrado que en los algoritmos digamos "clásicos" como los que hemos examinado aquí había una gran cantidad de ideas preconcebidas.

Por ejemplo, nunca se había pensado en un entorno tan heterogéneo como la WWW. En este entorno, el bajo rendimiento habitual de los motores de búsqueda convencionales demuestra el importantísimo papel, de momento insustituible, de la selección y filtrado de calidad previos que han desarrollado tradicionalmente las bibliotecas y centros de documentación. En estos entornos tan controlados de antemano, gracias a la intervención humana de selección y filtrado previos, algunos de estos algoritmos pueden llegar a funcionar razonablemente bien, pero en cambio no funcionan nada bien en el entorno heterogéneo y sin ningún tipo de filtro de la WWW.

En el futuro, los sistemas "inteligentes" de indización sólo podrán incrementar su eficiencia, es decir, sólo serán verdaderamente inteligentes sobre la base de: primero, considerar también las propiedades lingüísticas de los textos, y no sólo las estadísticas; segundo, incorporar el uso de instrumentos de control terminológico como los tesauros.

Esta última sería una relación muy adecuada de esfuerzo intelectual (o sea, hecho por personas) y de automatismo (es decir, de operaciones hechas por máquinas). Parece que es por aquí por donde irá el futuro de la RI. Con esfuerzo intelectual se construirían los tesauros, pero una vez construidos, podrían clonarse tantas veces como fuera necesario, y su uso pasaría a ser automático en lugar de manual, ya que los tesauros se consultarían y aplicarían como resultado de reglas de producción de sistemas expertos.

En cualquier caso, y como ya hemos indicado en otra parte, la RI es un campo de trabajo y de estudios interdisciplinario, cuya importancia no dejará de aumentar a medida que Internet vaya estando cada vez más presente en la vida de los ciudadanos.

Lluís Codina
(sitio web del autor: http://camelot.upf.es/~lcodina)

7. Fuentes seleccionadas de información

7.1. Bibliografía

Chowdhury, G.G. *Introduction to modern information retrieval*. London: Library Association, 1999, 451 p.

Kowalski, G. *Information retrieval systems: theory and implementation*. Boston: Kluwer, 1997, 282 p.

Salton, G.; Mcgill, M.J. *Introduction to modern information retrieval*. New York: McGraw-Hill, 1983, 448 p.

Salton, G. *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Reading (MA): Addison-Wesley, 1989, 530 p.

Lancaster, F.W. *Indexing and abstracting in theory and practice*. Champaing (IL): University of Illinois, 1998, 412 p.

Frakes, W.B.; Baeza-yates, R. (eds). *Information retrieval: data structures & algorithms*. Englewod Cliffs: Prentice Hall, 1992, 504 p.

Soergel, D. *Organizing information: principles of data base and retrieval systems*. Orlando: Academic Press, 1985, 450 p.

Losee Jr., R.M. *The science of information*. San Diego: Academic Press, 1990, 293 p.

Ellis, D. *New horizons in information retrieval*. London: The Library Association, 1990, 138 p.

Gillman, Peter (ed.). *Text retrieval: the state of the art*. London: Taylor Graham, 1990, 208 p.

Buckland, M. *Information and information systems*. Westport: Greenwood Press, 1991, 225 p.

Chorafas, D. N. *Intelligent multimedia databases: from object orientation and fuzzy engineering to intentional database structures*. Englewood Cliffs, New Jersey: Prentice Hall, 1994, 360 p.

Blair, D.c. *Language and representation in information retrieval*. Amsterdam: Elsevier, 1990, 335

Codina, L. "Sistemas automáticos de recuperación de información textual". En: Gomez Guinovart, J. *Aplicaciones lingüísticas de la informática*. Santiago de Compostela: Tórculo Edición, 1994, p. 63-86

Codina, L. "Recuperación de información e hipertextos: sus bases lógicas y su aplicación a la documentación periodística". En: Fuentes, M. Eulàlia (ed.). *Manual de Documentación periodística*. Madrid: Síntesis, 1995, p. 213-230

Codina, L. "Teoría de recuperación de información: modelos fundamentales y aplicación a la gestión documental". *Information world en español*, n. 38, octubre 1995, p. 18-22

7.2. Sitios web

Visualization Bookmars
(Lista de recursos sobre visualización de la información)
http://research.cis.drexel.edu/classes/ynsis300/visualization.html

Sics: Intelligent Software Agents
http://www.sics.se/isl/abc/survey.html

Search Engine Watch
http://www.searchenginewatch.com

Cataloguing and Indexing
http://www.desire.org/results/discovery

Center for Networked Information Discovery and Retrieval
http://www.cnidr.org

Lluís Codina. "Fundamentos de teoría de recuperación de información".

Information Retrieval Techniques

Paul B. Kantor

Introduction

Information retrieval (IR) technique stands today at a crossroads. Originally an outgrowth of librarianship, it has expanded into fields such as office automation, genome databases, fingerprint identification, medical image management, knowledge finding in databases, and multimedia management. Now with the national emphasis on digital libraries, it stands as the key issue of modem librarianship and challenges research in areas as diverse as artificial intelligence (AI), natural-language processing, and the statistical theory of inference. By "digital library" we mean the ensemble consisting of (1) a collection of texts, images, or data in digitized form; (2) a set of systems for indexing and navigating or retrieving in that collection; and (3) one or more defined communities of users. I propose that the global internetworked set of such libraries be thought of as "The Digital Library." This review aims to identify the current trends in the automation of indexing, of retrieval, and of the interaction between the systems and the users. The central issues are: (1) what the system does to describe the documents for purposes of retrieval; (2) how the system computes the degree of match between a given document and the current state of the query; and (3) what the system does with the information it obtains from the users.*

There are a number of general sources. Belkin & Croft have last reviewed information retrieval in *Arist*. Frakes & Baeza–Yates cover a range of issues, several more technical than those covered here. Harman (1992; 1993a) reports on the Trecs (Text Retrieval Conferences), which present the state of the art for some two dozen systems that all undertook retrieval from some 750,000 documents for a set of 100 preassigned topics or problems. Heaps addresses some computational aspects of Irt. The International Organization for Standardiztion (ISO) presents the important Standard Generalized Markup Language (SGML), which serves to let an algorithm identify conceptual slots for concepts, such as title, author, personal name, etc., in a machine-readable document. Pearl is an important source on the general question of probabilistic

* This paper has been influenced by almost everyone with whom I have discussed information retrieval over many years, in particular Richard Blankenbecler, Abraham Bookstein, and William Cooper, and, at Rutgers, Nicholas Belkin and Tefko Saracevic. One cannot review this literature without being struck by the prescience of Hans Peter Luhn and by the enormous impact of Gerry Salton and, more recently, Bruce Croft. Parts of this work were done during a visit at the University of Michigan School of Information and Library Studies, during fall 1993. *Annual Review of Information Science and Technology (ARIST),* Volume 29, 1994. Martha E. Williams, Editor. Published for the American Society for Information Science (ASIS). By Learned Information, Inc., Medford, N.J.

reasoning, which is having a strong impact on the interpretation of current algorithms.

In this review I describe matching algorithms in terms of what they calculate, avoiding a discussion of why the designers have chosen to calculate in this way. This reflects my view that the proof of a theoretical formulation can be found only in the performance of the algorithms that are realizing that formulation and not in the formulation's self–evidence or rhetorical strength. Robertson et al. (1982) have integrated several probabilistic approaches. An excellent source for the basic concepts of probability theory itself is Feller. The classic source for Bayesian analysis and inference remains Lindley. A more recent source on the same issues is Van Der Gaag.

Salton remains a classic which deserves study by anyone new to the field. Some additional material is added in Salton & Mcgill, reflecting the advance of the field during that period. Sparck Jones is an important source for understanding how the conventions of this field have arisen. More recently, Stanfill & Waltz surveyed a number of approaches. Tague et al. offer a formal model of an IR system. Other models, either explicit or implicit, can be found in the works of most of the authors cited here. Van Rijsbergen (1979) remains an important source for some of the probabilistic and Bayesian arguments.

In preparing this review the most recent conference proceedings proved to be the richest sources. In addition to the Trec conferences already mentioned, the annual conferences of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM Sigir) are excellent sources; the most recent is the 1993 meeting (Korfhage et al.). *The Journal of the American Society for Information Science* and the independent journal *Information Processing and Management* are the leading current resources. Other relevant work is widely scattered, as reflected in the bibliography. In selecting articles I have tried to represent the major themes currently important in the development of information retrieval technique at the expense of omitting many interesting papers.

Location, Navigation, and Indexing

Libraries, Catalogs, and Databases

In a classical library a book is in one place, collocated by author and subject, insofar as the book can be assigned a primary subject and a single author. To compensate for this loss of accessibility, libraries maintain a set of catalogs (author, title, subject, shelf list, authority files) to increase the chance of a user's being able to locate a needed book without having to examine (even superficially) every book in the library. It is well known in librarianship that this application of mixed controlled (subject headings) and free (title words) access both aids and confounds the readers. Controlled vocabulary, with all its power, relies on the user's ability to speak the same language as the indexers.

The IR technique-problem is to create in cyberspace the benefits that location and examination of documents provide in physical space.

The first contribution of the computer is that all of the contents of all of the several catalogs can be read by the computer and can be indexed both by their tagged fields and as free text. This permits a user to find all records that have "history" as a controlled descriptor or as a term in the title or in a note field. A full inverted file, in which each posting for a term is accompanied by a "document ID," and field or other tag information as well as position information, supports complex Boolean searches and proximity searches. In the final analysis, the results are pointers to the true information-bearing materials. The materials still must be "physically fetched." However, there is some work on OPACs (online public access catalogs) that is conceptually relevant for IR in general, and it is mentioned below as needed.

The next level of development is the bibliographic database, which typically contains citation information, descriptor and keyword fields, and the full text of an abstract. If we regard these abstracts as the documents of the database, we see that they are not located in one particular place as far as the user is concerned. In a sense the abstract is "located" in a cyberspace, at *all* the points through which it can be accessed.

We are accustomed to thinking of the bibliographic database as an access tool, but it is in fact the prototype of the digital library. It has been treated that way ever since the experiments by Cleverdon, in which the performance of a retrieval system was judged on the basis of the abstracts retrieved, without reference to the real documents (see also Cleverdon et al.).

Digital Libraries and Ideal Retrieval

Whatever is meant by a digital library (Lunin & Fox; as this chapter is being prepared, numerous conferences on digital libraries are being scheduled for 1994), it is clear that its contents, like the abstracts in a database, are located anywhere in cyberspace that the user can find them. Further, just as users cannot look at every book in a conventional library, so they cannot look at every retrievable document in a digital library.

The central role of IR techniques is to bring this potentially suffocating quantity of information under control and make it available to the immediate users and to the world at large. To do this, IRT should ideally produce results similar to those that would be achieved if the users could scan all retrievable documents and rank them in order of usefulness. The possible results of such an (impossible) scanning may have a very complicated structure. For example, there might be nonoverlapping heaps, each labeled "If I can have this, I don't need anything more" there would be an enormous heap labeled "not interesting"; and there might be overlapping heaps, ranked lists, and so on. Infor-

mation retrieval as practiced today simplifies this problem by assuming that every retrievable item is either "relevant" or "not relevant" (for further information on relevance see Schamber). This consists of the weak assertion that relevance is binary and the strong assertion that relevance is a property of documents in isolation, unaffected by the other documents that have been retrieved. The weak assertion is easily corrected using a fractional indicator of relevance. The strong assertion is not easily corrected in present systems.

Thus, of the whole complex structure that the (impossible) scan of a complex dataspace might reveal, current IRT concentrates on reproducing the ranked lists of documents, with the more relevant ones ranked closer to the top. We note that a Boolean (set) retrieval is a very weak form, of ranked list, in which all the retrieved documents are given one rank (call it "1") and all the other documents are given another rank (call it "2"). With these restrictions, IRT seeks to make this ranked list perform as we1l as possible.

Documents, Queries, Structures, and concepts

The evolution of IRT, from the classical library to present research, may be viewed as an effort to first move away from concept-based indexing and then to move back toward it with content–based algorithms. In a sense the issue is whether knowledge of the contents of a document takes the place of knowledge of its concepts.

A document (in its entirety rather than as represented by its abstract) has in it both concepts and a conceptual structure. There are relationships among the concepts present in the document, and often those relationships are the essential meaning of the document. The relationships are what must be preserved under paraphrase in order for us to believe that the document has been "understood". Representation of the conceptual structures is being pursued under the general title of "fact retrieval" or "rnessage understanding." The most widely used method, is a frame-based approach. In this approach the set of conceptual relations is schematized, and the instances of concepts (actors, objects) for the present document are labeled by the slot in the frame which they seem to fill. This topic is outside of the scope of this review.

When we set aside the structure, a document may be thought of as a "sack of concepts" or set of concepts. These concepts no longer have specific roles to play or specific relationships to each other. They have become in fact a set of "labels." Historically, indexers translate these labels into the language of some controlled vocabulary. The words of that vocabulary are in turn assigned as indicators of the content. This could be expressed numerically as:

$$Ind(d,c) = \begin{cases} 1 \text{ if concept } c \text{ describes document } d \\ 0 \text{ otherwise} \end{cases}$$

The index file contains entries labeled by the various values of c (the concept), and the postings for that entry are precisely the pointers to the documents for which the relationship *Ind* has the value 1.

The relationship *Ind* can be thought of as defining a table or matrix, and each entry in that table can be anything at all so long as it is determined only by the document and the concept. Typically we think of entering some numerical indication of the degree to which the concept describes the document. We ought not at this point enter anything that is dependent on the entire corpus of documents or on the entire universe of concepts. To do so would destroy the locality of the definition. Such nonlocal concepts (the so-called inverse document frequency is a prime example) can be introduced at retrieval time without loss of generality. Related discussions are given by Fuhr (1992). Paice & Jones offer a different view, in which structure is rnaintained. A formal model for IR systems is given by Tague et al.

The basic event for analyzing the workings of an IR system is the retrieval instance. An instance is characterized by the presence at the system of a user with a single coherent (possibly complex) need or problem of some kind such that the user (at least) is able to decide whether a retrieved item helps to solve that problem. Our knowledge of this situation, and in particular the inputs governing the behaviour of the system, are limited to the utterances of the user. However, the user draws on a rich understanding of the situation to form such utterances as "document A is not relevant" or "I'm interested in rocket science."

In any specific retrieval instance, the information need will at first be represented by a query statement. When such a statement is "processed" by a human interpreter, it yields a structure and a (very small) set of concepts. When the structure is removed, each query may be represented as a set of numbers, which can be thought of as the degree to which each of several concepts is present in or implied by that query.

Matching documents to queries

When two sets of numbers refer to the same index set, the set of concepts, they can he treated as the components of an abstract vector. In particular, an inner product can be calculated. This inner product can be taken as an indicator of the match between a specific document and a specific query.

$$\text{Match}(d,q)= \sum_{\text{concepts c}} d(c)q(c)$$

We have switched from the more complex notation *Ind*(*d*,*c*) to the more compact *d*(*c*). In this notation c labels a specific concept, and d is somewhat overworked since it stands for the document whose label is d. It can be called a "document vector in concept space". In the same way, *q*(*c*) represents the degree to which the concept c is "called for" by the query q.

In practice, the vectors *d, q* may have to be normalized to correct for the difference between documents with many concepts and those with few. In fact, as discussed below, most systems label the components of the vectors by terms, but they intend to label concepts, and we continue to use the character *c* in this part of the discussion. For another perspective on the representation of concepts, see Hearst (1993a; 1993b). For more detail on vector space modeling see Wong Et Al. (1987). For a description in the language of probabilistic indexing (which seeks to assess the probability that assigning concept c to document d would be effective), see Fuhr (1989b).

Weight of Concepts

In principle, finding matches with regard to some concepts may be more important than finding matches with regard to others. This can be represented mathematically by introducing a weight into the sum. The weight can also be thought of as a table whose nonzero elements lie only on the diagonal.

$$Match(d, q) = \sum_{conceptsc} d(c)W(c)q(c)$$

or

$$Match(d, q) = \sum_{conceptsc,\ c'} d(c)W(c, c')q(c')$$

with

$$W(c, c') = 0 \text{ if } c \neq c'.$$

The problem of choosing weights for terms so that retrieval works well is discussed by Salton & Buckley (1988). The problem of choosing those weights so that documents are ranked in the best order and at the same time the relevant fraction is predicted is considered by Cooper et al. See also Fuhr (1989a; 1990), Fuhr & Buckley, and Fuhr et al. For an application of the probabilistic indexing ideas, in the context of selecting an optimal indexing and weighting, see Fuhr & Buckley.

Relations between Concepts

In principle concepts may be related to each other, either in general, in a specific domain, or perhaps only in the case of a specific query. Such relations can be set out in a thesaurus and can be represented by tables of numbers T(*c, c'*) whose entries indicate the degree to which the concepts are related to each other. In principle a standard thesaurus gives rise to several different tables, which can be labeled by the type of relation ("see also"; "related concept"; the pair of relations: "broader/ narrower concept" can be represented together in a single table).

The entries of this table can be inserted into the sum defining the match:

$$\text{Match}(d, q) \sum_{conceptesc,\ c',\ c''} d(c)W(c, c')T(c,c'')q(c'').$$

In short, the idea of matching a query to a document, when both are thought of as sets of concepts, can be well represented by linear expressions which can be thought of as operations on (suitably normalized) vectors in some abstract space. We may say that the vector q is transformed through a matrix multiplication to:

$$q \rightarrow q' = WTq.$$

When training data are available (i.e., documents of known relevance), the problem of designing an IR system may be restated as: what choice of Pff works best for a specific query q at discriminating the documents whose relevance is known? This may be thought of as finding the best q' for each specific q or as the more general problem of handling all q at once (see discussions in Wong et al. (1987) and Wong & Yao). The general discrimination problem can be treated by linear programming (along the lines discussed by Mangasarian). When it is cast in quadratic form, the method of Yang & Chute results. For a theoretical framing in terms of a general probabilistic model, see Bookstein. For an extension to nonlinear (polynomial) functions involving the *tf* and *idf* measures, see Fuhr (1989b). Another perspective is found in Robertson et al. (1994) who use an intractable formal model to suggest specific term weighting formulae.

Restoring query structure

Under the influence of Boolean-based online retrieval systems, users of retrieval systems have learned to formulate queries with a modest logical structure. Typically this takes the form of the conjunction (or intersection, or logical *AND*) of several parts, each of which is a disjunction (union, logical *OR*) of its own components. Each of these components could be a concept. Similar concepts (really, substitutable concepts) are banded together in the parts, and some representative of each must be present in the whole. This may be represented as

$$\text{Structure } (q|c_1,\ldots c_m) = \bigcap_{\text{subparts of } q} \left[\bigcup_{c_i \varepsilon \text{ subpart}} c_i \right].$$

When the relation of a document to a concept takes only the values 1 and 0, these logical operations are easily represented mathematically. There are several ways to extend those representations to relations that take values between 0 and 1. Without going into the rationale for developing each of them, we summarize them below.

All of these methods respect DeMorgan's rules relating unions and intersections, that the complement of the intersection is the union of the complements. One retains the other key properties of Boolean algebra, while the other two do not (see also Bookstein; Kantor (1981)). For a different approach, based on object–oriented query languages, see Bertino et al. Chiariaramella & Nie consider an alternative approach based on modal logics. Jacobs et al.

(1993) consider a Boolean approximation method. Lee et al. discuss evaluation of Boolean operators and structural relations. Prade & Testemale consider generalizing database algebra to deal with vagueness in query representation. Schäuble (1993) considers dynamic variation in the data structures and its impact on query formulation.

Fuzzy-Set Representations

Fuzzy subset theory (Zadeh) replaces the Boolean notion that an object either does or does not belong to a set with the notion of a generalized membership function, which can assume any value between 0 and 1. A fuzzy subset S of a set of elements (x S X) is defined by a membership function:

$$0 \leq \int_S (x) \leq 1.$$

If fs (x) = 1, we say that x is definitely in S, and if f_S (x) = 0, we say that it is definitely not in S. The operations of union and intersection have been defined for fuzzy sets by the rules:

$$\int_{S \cap T} (x) = \min_{x \epsilon X} \{\int_S (x), \int_T (x)\}$$

$$\int_{S \cup T} (x) = \max_{x \epsilon X} \{\int_S (x), \int_T (x)\} .$$

A structured query can then be interpreted by transforming each term of the query to, the corresponding one-component vector in concept space, calculating all of the similarity measures using *W*and *T* and computing the score overall according to the rules of fuzzy logic.

For example (neglecting for the moment the possibility of off-diagonal effects), we can calculate the fuzzy logic value of the similarity of the query:

$$q = fish \cap oil(\cup scales)$$

and a document *d* as:

$$Sim(q, d: FL) = \min \{d(fish), \max \{d(oil), d(scales)\}\}$$

Buckles & Petry discuss the use of fuzzy concepts in databases. Negoita is a source on fuzzy sets and expert systems generally. Ogawa et al. (1991) apply fuzzy concepts to document retrieval, and H.J. Zimmerman is a general source on fuzzy-set theory.

Product-Based Representations

These combine two measures of relatedness according to the rules (here f_Z represents the match of some specific document to the concept z):

$$\int_{xANDy} = \iint_{x\,y}$$

$$\int_{xORy} = 1-\left(1-\int_{x}\right)\left(1-\int_{y}\right)$$

This really calculates the *OR* function from the *AND* function by DeMorgan's rules. In practice (Croft et al., 1991; Turtle) the strict product penalizes absent terms too heavily. This can be prevented by adjusting the definition of fx so that it ranges from, say 0.3 to 1. For more on the development of this inference scheme in information retrieval see Turtle & Croft (1991; 1992) and Croft et al. (199 D.

Sum-Based Representations: p-Norms

The p-Norm rules combine two measures of the match between a document and a concept according to the rules:

$$\int_{xORy} = \frac{1}{2^{1/p}}\left(\int_{x}^{p}+\int_{y}^{p}\right)^{1/p}$$

$$\int_{x} = 1-\int_{x}$$

$$\int_{xANDy} = 1-\int_{\bar{x}OR\bar{y}}$$

When $p = 1$, the rule for *OR* is the same as a linear combination. In a typical application, values of p are between 1 and 2. lf p becomes very large, this rule approaches the fuzzy-set rule. Again the DeMorgan relation is respected, but this time *OR* is taken as fundamental, and *AND* is computed. As for the product form, it is found (Fox & Shaw) that better performance is obtained when f is restricted, typically to the range 05-1.0. This approach to modeling structure was developed by Fox.

Numerical Methods, Structured Queries, and Sets of Concepts

Using any of these three approaches, one may convert a query in, for example, the form:

$$\text{query} = \bigcap_{parts}\left(\bigcup_{concepts\ \varepsilon\ part}\text{concepts}\right)$$

into a numerical expression of the match between a given document, described by the numbers $d(c)$, and the query as shown here. The indicated operations are simply applied to the numbers representing the importance of each of

the concepts in the document. In this fashion, numerically based methods can encompass the fundamental Boolean operations.

In sum, we see that by admitting a fairly complex linear structure and then embellishing it with selected extensions to represent the residual structure of a query, the representation of documents as vectors labeled by the concepts, and of queries as structures of concepts, can support a rich program of retrieval. We note that this kind of matching scheme, which we have expressed in terms of linear algebra and slight nonlinear generalizations, can be arrived at from a variety of theoretical perspectives, such as natural-language concepts, probabilistic or statistical retrieval, and frame–based reasoning. In fact, most of he methods have been applied directly to terms rather than to concepts. We now consider what happens when the concepts are no longer assumed to be known.

Algorithms and Content vs. Concepts

The algorithms that process a text for retrieval can be said to "know" the terms in that text with near certainty. In English (or European languages generally), for example, a term corresponds at first to a word. That is, a term is anything delimited by white space or punctuation. In some approaches (recently, Cavnar) terms, are defined arbitrarily as n-*grams* of characters. This approach sacrifices all prior knowledge of the meaning of terms. Fujii & Croft report that 1-*grams*, i.e., character-based retrieval, works nearly as well as word-based retrieval for Japanese texts and is computationally easier. On the other hand, phrases, such as "health care" could be seen as terms in a more general sense. Such noun phrases are difficult to incorporate in indexes because there are so many of them.

Whether terms are identified with words, with n–*grams*, or with phrases, the processing of both the query and the retrievable documents follows the same lines as those described in our discussion of concepts, but instead of indexing the weights and relations by c, representing "concepts," we are forced to index them by *t*, representing *terms that actually appear in the text of the document or the query*. Because they are present in the text, we call them the "contents" of the text as opposed to, the concepts which are intended or perceived by a human agent.

Preprocessing is a step in extracting concepts from contents. Older methods of preprocessing a document replace all words by their uppercase forms and replace many words by their stems. Finally, a list of common words (the stop words) are often not indexed, although they may be counted as place holders in determining the position of words. All of this helps to focus on the concepts that the words represent without regard to, the specific morphological variants (i.e., forms) in which they appear. We mention that the indexing of a text in this way can consume an enormous amount of space as phrases and

word combinations are added to, the inverted file. A compression scheme to alleviate this problem is suggested by Linoff & Stanfill.

Formulating Concepts in Terms of Contents

The linear formulation discussed above is the famous vector method incorporated in the Smart system and its descendants. Documents are represented by vectors whose elements are related to the frequency with which terms appear in them, and queries are expressed on the same basis. Certain basic principles had already been proposed by Luhn (1959) prior to the development of the Smart system. The most important are: (1) the degree to which the document and the term are related should increase as the frequency of the term in the document increases (the term frequency or tf principle); and (2) the usefulness of a term in discriminating among documents should decrease as the number of documents in which it appears increases (the inverse document frequency, or *idf* principle). While *tf* is a strictly local concept, *idf* is a corpus-based concept. In the simplest interpretations, the sum defining the match contains terms corresponding to the terms in a query, and each term involves the product $d(t)W(t)$ for some t which appears in the query. Hence such methods are generically referred to as *tf.idf* methods.

Some recent work (Cooper et al; Fuhr, 1989b) asks whether systematic search in a space of alternate formulas for *tf*, for *idf*, and for the product leads to improved retrieval performance. Of course, expanding the range of alternatives can improve performance, and the results are not yet definitive. (As noted, Cooper et al. ask whether some combinations of the variables can predict the fraction of a document, in a given range of the ranking, that will be judged relevant to the problem at hand. This does not bear on the ranking problem per se but is of interest in the development of theoretical models (Robertson et al., 1982) regarding that fraction). Tong & appelbaum seek to define term weights based on the classificatory power of specific query terms, as determined from a training set of documents. The definition of power is based on the concept of Classification Analysis with Regression Trees (CART). lt is somewhat like the development of optimal strategies for a sequential binary choice game such as Twenty Questions.

Term Refinement

In addition to this work, which seeks to fine tune the general *tf.idf* concept, there is much work on restoring the conceptual dimension to a system that processes text strictly in terms of its content. The problem of disambiguation is important. The same character string, for example, “bank” has quite a different meaning in a discussion of flight vs. a discussion of finance. In effect, a single term is resolved into several different terms which have different weights and different relations to other terms. There is a consensus that human readers disambiguate these cases by using the surrounding text to esta-

blish a context (see Vorhees et al. for further discussion). A discussion in the context of inference–based retrieval is given by Krovetz & Croft.

Stemming as a Relation between Terms

Stemming can be represented by direct conflation in the postings (that is, all trace of the morphology is destroyed in building the table of document–term relations), or it can be represented by off–diagonal weights $W(t, t') = 1$. The latter has the same effect as far as computation of the match between documents and queries, but the issue of weighting terms must be considered in more detail. For example, consider the popular "inverse document frequency" weighting scheme,

$$W(t, t') = 1\mathrm{n}\left(\frac{|Document\ Set|}{|Set(t\ is\ in\ the\ document)|}\right).$$

If stemming is done first, this will include all variant forms of the term. (In this equation /*Set*/ represents the number of elements in the set.) To accomplish the same effect if stemming is implemented through the off-diagonal weight matrix, the calculation must include a count of all of the documents in which the related terms appear, and to preserve the same meaning, this must refer to the union of the several sets. This is a more complex calculation, although there is no difficulty in principle.

Terms, Proximity, and Concepts

Even systems that are Boolean, in the sense that they retrieve sets without ranking them, admit more than just Boolean operations on the set of terms. First, they admit and use predefined conceptual entities by permitting the user to specify that a search be confined to the title, keywords, and so on. The rapid development of markup languages, which identify the parts of a text in a machine-readable way, greatly facilitates the automation of text processing for retrieval (see, for example, Goldfarb). Second, for the reconstruction of concepts from free text, they admit proximity operators. Typical are the DIALOG operators $t(nW)\ t'$ and $t(nN)t'$.

The first operator specifies that there be no more than n terms between t and t' and that they be in the specified order; the second does not require the specified order. Given the linguistic conventions of English, this tool lets the system move a long way toward the representation of concepts by terms. In particular, it brings together essential ingredients of a phrase (the open-class words) while the remaining parts of the phrase may be an array of closed-class words or "boiler plate" unique to the authors style and point of view.

These are relations among terms that are expressed only at search time and calculated explicitly during retrieval. Insofar as I know, no system indexes text by such proximity reIationships. M. Zimmerman discusses proximity correla-

tion. In all the cases considered here the relations among terms are defined by the user at retrieval time.

More Complex Relations among Terms: Corpus-Independent Case

The corpus-independent approach to developing term-term relationships requires study of the lexicon of the underlying language in which the documents are written. The approaches range from thesaurus construction to natural–language processing. Efforts in this direction (Voorhees) have used the WordNet (Miller et al, 1990a, 1990b) semantic network structure as a basis. Results to date have not been as successful as hoped, and it has been suggested that the specificity of relationships required cannot be found in a universal network of term–term relations. If so, further improvement along these lines will require the development of domain–specific networks of term-term relations corresponding to the underlying concepts of the domains. This is akin to the widespread finding in AI that the solution to a specific problem requires development of domain–specific intelligence. Other papers that address the issue of domain–independent term-term relations include those by Evans & Lefferts, Jacobs et al. (1991), Krovetz (1992), and Lesk.

Formulating Term-Term Relations with the Aid of the Corpus

If universal term–term nets (or semantic nets) approach the issue of extracting concepts from one side, we may say that corpus-specific methods approach it from the other. These methods suppose that the corpus already represents only one domain. They seek to build relations among terms from the co-occurrence of terms within documents. Returning to the fundamental matrix $Ind(d,t)$, note that while the rows of this matrix or table represent "document vectors," the columns are vectors representing the terms. Thus, various metric schemes similar to those for expressing the relations between documents and queries may be used to express relations between terms and terms. One natural form for the term–term matrix is:

$$T(t, t') = \sum_{d \in corpus} Ind(d, t) Ind(d, t')$$

More complex formulations might give documents different weights, based on, for example, the number of terms occurring in them. In general, one would expect documents containing many terms to yield more spurious relations among terms, and might wish to assign them a lower weight. In addition to some works cited earlier, Van Rijsbergen (1977) discusses a theoretical basis for some of this work (see also Can & Ozkarafian (1987; 1990) and Peat & Willett). For a natural–language perspective, see Strzalkowski & Perez Carballo. Wong et al. (1993) consider the computation of term associations by neural network methods.

Reducing the Dimension of the Concept Space

Since the dimension of the term space is very large, the term-term relations can also be used to reduce the dimensionality of this space. This is conceptually similar to principal component analysis or factor analysis in statistics. Such approaches are best illustrated by the Latent Semantic Analysis of Deerwester et al. and Dumais. Essentially a matrix such as $T(t, t')$ is analyzed to find a space of lower dimension that contains most of the significant parts of the matrix. The corresponding axes in the original term space deserve to be called concepts even if we cannot name or label those concepts in natural language alone.

Another way to reduce the dimension of the concept space is to use adaptive classification techniques that do not rely on linear operations. The Kohonen feature map (Gallant et al.) is a technique to develop a few vectors that describe a much larger number of objects. The vectors are constrained to lie on some surface in the original space and are adjusted so that they are near the original entities (e.g., the concepts) and also are not too close to each other. This latter condition is met by imposing a so–called "conscience" mechanism which effectively causes the vectors to repel each other if they get too close. The resulting concept vectors can then be used to support a classification of documents and queries, as suggested in the first part of this chapter. For another perspective, see Tzeras & Hartmann and Yang & Chute.

Reducing the Cardinality of the Document Space: Clustering

Rather than use the $Ind(d, t)$ table to define a relation among the terms, one could use it to define a relation among the documents:

$$U(d, d') = \sum_{t \in inverted\ file} Ind(d, t) Ind(d', t)$$

Any relation of this kind may be used to group the documents into so-called document clusters. The most intellectually satisfying but computationally intensive approach is to define clusters, all of whose members have a specified level of similarity to one another. In this regard, Willet remains an important source. For an introduction to clustering, see Kaufman & Rousseeuw. These clusters may then be represented by either a selected specific document vector or by the average of all of the vectors corresponding to documents in the cluster. The issues of how to build clusters, which vectors to include, and when to draw the boundary have been with IR from the beginning (Rocchio).

Clusters of documents, like clusters of terms, represent concepts. While each document no doubt contains many concepts, the cluster will rank some concepts more highly (e.g., because they appear in more of the documents). Thus, although a cluster may not represent a specific concept, it is more specific in concept space than is an individual document. Recent discussions of issues related to clustering are given by Botafogo and by Fuller et al.

Formulating Concepts with the Assistance of the Users

In concert with the development of algorithms to find concepts in texts, there is a major thrust to enlist the user in the real–time definition of concepts. This draws on the (often unstated) realization that the user possesses a more powerful mechanism for scanning, reasoning, and evaluating than any mechanism the program could provide.

Hence, the best system, at least for now, will have a Person in The Loop (PiTL). The issue is to find the best ways to extract useful informa–tion from the person and to find the best ways to use that information in iterated retrieval. Methods under investigation include: (1) iconic and graphic metaphors for collections of items (Rose et al.); (2) efforts to represent the links among concepts or documents that are stored within the machine; (3) thumbnail representation of the graphics within a document; and (4) representation of title page layout to ease the user's scanning process (Hoffman et al.).

The twofold challenge of the PiTL idea is: (1) to tell the user what the system has found so that the user's feedback is really helpful, and (2) to incorporate the users feedback in a way that improves performance. General papers dealing with these issues include those by Agosti et al., Anick, Furnas et al, and Harman (1988). A review of gateway devices is given by Efthimiadis (1990).

Automatic Summarization and Theme Sentence Extraction

Automatic summarization by extraction of key sentences (Luhn, 1958, 1959; Salton & Buckley, 199la, 1991b; Salton et al., 1993) and by explicit formulation of sentences (Mckeown; Mckeown et al.) provides a way for the system to describe the dataspace to the user. This helps the PiTL to navigate and to clarify the information need. The scatter/gather approach (Cutting et al., 1992, 1993) tries to move the user through successive layers of query refinement by offering titles or phrases typical of a subset of the documents and following the user's selections. All of these can be seen as different ways to make the user a more effective component of the retrieval system and he1p the user to refine and express the concepts of the query in paths that the system can use. Fox et al. trace the evolution of the interactive approach in the setting of an online catalog.

Relevance Feedback and Query Expansion

Research on putting the user in the loop includes study of the weights to attach to terms in the documents that a user finds relevant. This is a difficult area since the value of a single term may not be a definable concept. That is, the effects of terms are generally seen in combination, and we cannot say that each of the components of the combination has a value of its own unless we know how those values combine to achieve retrieval. Pursuing this issue takes us deep into probabilistic retrieval. Roughly, it is known that under term-in-

dependence assumptions or maximum-entropy assumptions, there will be (in a specific logistic sense) a value attributable to each term appearing in the collection (Kantor, 1984), but the assumptions of these models are not yet verified. There are not yet satisfactory studies of the joint distribution of terms in the subsets *C* (good or relevant) and *B* (bad or not relevant) of documents for a set of queries. However, the large Trec collections couId support an ambitious study of these issues. The essential issues of relevance feedback today are: (1) given that a document is judged relevant, how should the weights, the query vector, or both be changed in a vectorial formulation; and (2) if the system admits structured queries originally, can the feedback information be incorporated into the structure, or must it be treated in a different way from the original information?

Note that end-user retrieval and mediated retrieval often incorporate new information into a semi–Boolean structure by adding synonyms as they are found in early retrievals or by introducing negation to block unwanted documents including unwanted terms. An early study of the effect of query expansion is found in Smeaton & Van Rijsbergen. Recent papers dealing with the issues of query revision, in technical terms, include those by Efthimiadis (1993), Pedersen, Robertson, and Robertson et al. (1986). The relation of the query expansion to a concept space is discussed by Qiu & Frei. Haines & Croft explore relevance feedback in the inference network framework (discussed earlier).

Complex System Codes vs. Underlying Principles

Like computer codes in other areas of science and engineering, the cutting-edge retrieval codes have been built up over years of modification by large teams of scientists and programmers. The result is that a program is not adequately characterized by an enumeration of its design principles. Another system, adhering to the same principles, may perform quite differently.

Examples of the areas in which undocumented program differences arise are: (1) choice of the dimension to which some space is to be reduced; (2) selection of the conscience parameters for a Kohonen algorithm; (3) selection of the interval within which "document frequency parameters" are allowed to vary; (4) specific choice of stemming rules and stop lists; and (5) hand built sets of terms to represent concepts.

Each of these choices has an effect on the performance of the system, but systems are so complex and the space of options so large that there is little chance of isolating the effect of any one choice. In practice, systems are run, bugs are tracked down and eliminated, and features are improved until time or resources run out.

Thus, information retrieval method has become a mature technology with both a science and a practice. This makes it possible for the science to develop more rapidly as it learns which practices are most effective. At the same time, because the practices are realized in enormously complex programs, it is difficult to design experiments that will determine which principles most contribute to the success of those practices. Because human users are more intelligent than the systems they use, perhaps several quite different systems are equally effective for users who have adapted to them.

Evaluation of Systems: Beyond Precision and Recall

Precision and Recall

The classic concepts of precision and recall (Cleverdon) are still used today. Briefly, all calculations are based on the assumption that for each query every retrievable document is either good or bad ("relevant" or "not relevant"). It is assumed that the total number good, *G* is known in advance. It is assumed that every retrieved document (up to some point of exhaustion) is judged as to its relevance. When the documents are ranked according to some parameter, with ranks 1,2, …, we can define g(*r*) to be the number of good documents among the first r documents. The precision at this point is *g/r*, and the recall is g(*r*)/*G*. These two functions of *r* can be summarized by a graph showing precision as a function of recall. There is a remarkably large literature on how to draw this graph.

In real–world retrieval, the value of *G* is estimated typically by pooling the retrieved documents for several systems. In the Trec–2 (Harman, 1994) experiments, for example, the top 100 documents retrieved by each of more than 30 participating systems were actually reviewed and the systems were judged using this definition of recall. For the better systems, the performance was slightly above 40% precision at 40% recall. When the systems were trained on one body of full texts and then applied to another (the so–called "routing" task), system performance rose to over 50% precision at 40% recall. This measure is the average over all the queries of the precision for that query at 40% recall (see Harman, 1993a; 1993b; 1994).

Characterizing the Power of Systems

All previous efforts to reduce the performance of systems to a single number have proved unsatisfactory. Even for a simple–set retrieval system, the performance is essentially two dimensional. For ranking systems the performance is characterized by a curve. Recent new work which suggests ways to deal with this problem includes that of Frei & Schäuble, who introduce the notion of "usefulness," and Hull, who summarizes ways to deal with the nonparametric nature of the scores as currently calculated. A typical score is the average precision averaged over several levels of recall and averaged over all of the queries to which the system has been applied.

The search for ways to summarize a system's performance may proceed along either of two paths. One is to develop a theory of precision–recall curves that reveals a single parameter at work. There is no published work on this issue, and it presents interesting possibilities for evaluation research.

The other path is to find a nonparametric characterization of the relation between two systems. Such an approach might determine whether system A performs better than system B while ignoring the question of how much better. This could provide a new path to the selection and evaluation of systems. However, in the final analysis it will be necessary to say "how good" a system is in order to justify its cost of development and operation.

As the world of information becomes a single linked entity, the notion of recall (or even the "relative recall" defined in terms of the number of retrieved good documents) will have to fall out of use. Rather, much in the spirit of combination of information and of data fusion, we will be led to ask: given the results of retrieval using algorithms *A, B, C,* suitably combined, should we do any more work? This question is best answered in principle, by reporting, for each possible increment of value (*v*) the probability *Pr*(*v* 1 adding *S* to *A, B, C*) that we can realize that added value by using algorithm S in addition to A, B, and C. Of course, answering this question depends on knowing how to use several systems together, which we consider below. The concept of decision rules for stopping has been applied to the single search by Kantor (1987).

For any given bi–corpus, the first choice system will be the one that maximizes expected value. An alternative rule would be to choose the system that maximizes the probability of getting the amount of value that we need. Each of these would be calculated for the case of no prior search. However, an IR technique or system might remain important because it is the right second choice for many problems where the performance of the first algorithm is not good enough. In general then, evaluation of systems will enter the work–day world in which the cost of using the system is balanced against the expected value to he derived from using it, and precision–recall measures appear, if at all, as interme–diary steps in the estimation of value.

Future: Combination of Information And Use of Exogenous information

Combination of Evidence, Data Fusion, and Combination of Information

It has long been known that human indexers and searchers do not agree in their assignment of index terms, retrieval of relevant documents, or even in judgements of relevance (Katzer et al.). Initially this was viewed as a flaw in the performance of rules or of humans, but more recently it has been situated in the general problem of detection and decision or inference. The fact that an indicator of relevance (such as a human judge) is imperfect opens the possibility that combinations of several indicators will yield a better performance than any of them, alone (see Saracevic & Kantor and Kantor (1992) for an imple-

mentation of this idea when the indicators are distinct trained searchers). In what follows we refer to each indicator as a "scheme", which might be either a particular searcher, a particular automated system, or even a particular combination of several systems. To simplify the example, suppose that retrieval is set–wise, and that scheme 1 retrieves good documents with probability d_1, while scheme 2 does so –With probability d_2. That is, each good document has a probability d, to be retrieved by scheme 1. The expected number of good documents retrieved by scheme 1 is d_1C.

Let the corresponding probabilities for retrieving bad documents be f_1 and f_2. The a priori odds that a document is relevant are $G : B$. As before, G represents the number of good documents, and B represents the number of bad documents in the dataspace. The corresponding odds for documents retrieved by scheme 1 are: $dG : fB$. Thus, as long as $d > f$, the odds of finding a good document in the retrieved set are better than the a priori odds. If the two retrieval systems are stochastically independent with respect to "goodness," the odds for documents retrieved by both schemes are $d_1\, d_2\, G$: $f_1\, f_2\, B$. Thus, if both schemes are effective, the intersection of the retrieved sets is even more effective. Hence, the intersection of the retrieved sets should be ranked above the remainder of their union. Available data (Kantor, 1992) indicate that this independence assumption is too strong because it predicts too strong an improvement in the odds.

The Inquery system is built explicitly on combination of evidence (Pearl; Turtle & Croft, 1991) as the logical foundation for its numerical algorithms. The idea of data fusion appears in the work of Fox & Shaw Belkin et al. (1994), and Kantor (1994). All of these are examples of what is now called in statistics the "combination of information." In a nutshell, these methods propose that several imperfect Indicators of the relevance of a document to a query instance can be combined to provide a more reliable indicator. A presentation in Bayesian language is given by Thompson (1994).

In the Inquery system the fundamental indicators are the presence or absence of terms in the document, together with constructed concepts, and some attempt to add phrase–finding capabilities. In the data–fusion approaches, the indicators are the ranked outputs of several separate systems, each working the same dataspace for the same query.

In general, expanding a parameter space always opens the possibility for improving performance. So it is not surprising that these methods do yield a posteriori better performance than those methods of the systems or indicators that they combine. To be specific, we might combine a vector space method with a p–norm method, with relative weights a and $1 - \alpha\cdot$. The original system, whose parameters characterize the rule for forming term frequencies and inverse document frequency, had some parameter space P. The new one has parameters in P X $\{(\alpha, p)\}$. This new space is larger and contains the original

subspace as the special case a = 1 (*p* doesn't matter). So every performance that could be achieved before can still be achieved, and so can other values of the performance. If even one of these value is better than the original, then a combination of systems has improved system performance.

Thus, it is not surprising that in each instance such combinations can often yield a performance that is better than the best achieved by the original systems (Kantor, 1994). An important research issue is to find rules of combination that are stable and effective across a full range of queries and /or documents.

Feedback of Exogenous Information: Relations among Documents

As described above, developments in interfaces and feedback are permitting users to play a role in the definition of concepts at retrieval time, either explicitly or implicitly. These concepts can be captured in two ways. One is to permit users to annotate the descriptions of documents. This information then becomes internal to the system (part of the contents) and can he processed in the same way as all other contents. Users can also provide information in the form of direct links among documents. This type of link can reflect a conceptual relationship without making it explicit. In particular, the link is supplied exogenously (from outside the system), and its specific nature (or the name of the concept which it reflects) is never available to the system, per se. This idea has been proposed for library catalogs by Koenic (see also Croft & Turtle). This type of link is realized for catalogs in the Adaptive Network Library Interface System (Kantor, 1993; Zhao & Kantor). It represents a way to include concepts in the system without naming them, by simply storing direct links among the retrievable documents.

This concept is related to the hypercatalog concept discussed by Fijerppe, with the specific proviso that the links cannot be found algorithmically from the contents alone –i.e., they are available only because there are human users of the system.

Human Scanning: a New Source of Ideas?

In all studies of system performance, human judgement is the final standard of effectiveness. The implication is that we strive to make systems as good as people, but people are extremely variable in their judgement of the match between concepts and documents and in all other decision tasks related to IR. It maybe argued that this is due to an essential imprecision in the expression of ideas in language or to an essential imprecision in the ideas themselves. In any case, we can follow this insight in two complementary directions. One affects the evaluation process; the other suggests a new source for development of algorithms.

Fractional Relevance Judgments and Inconsistency of Human Evaluations

Evaluation is performed against a "golden rule" in which the items deemed relevant are to be recovered before those deemed not relevant. When the system is serving the real user in a real retrieval instance, this seems the best possible approach. However, when the system is being judged offline, as in the Trec conferences, there should he some way to improve on this by using several judges for each query–document pair. The relevance score will then be a fraction (the number of judges asserting relevance divided by the total number of judges). This is a direct operational "fuzzification" of the notion of relevance. With this information the notion of precision–recall, or of cumulated relevance, can be suitably generalized. Rather than the total number of relevant documents *G*, one would speak of the total relevance scores of all the documents and so forth. This will move us toward a more "impersonal" characterization of what the system wants to achieve.

Human Scanning and Recognition Primitives

We have asserted that the user is being put back into the loop for query reformulation and the interactive refinement of concepts from content because the human outperforms any existing computer. This suggests that we might use the study of human scanning (of text, images, data) for new insights into the identification of concepts.

Although beyond the scope of this article, there are important and puzzing results from image processing (Julesz), which suggest: that the human eye responds instantly to characteristics that are, from the point of view of algorithm development, very high order indeed. This suggests that the rational path for algorithm development that has been followed so far (see Myler & Weeks) may lack crucial recognition primitives (basic elements to which the human mind responds) that would facilitate the retrieval of relevant images.

By a modest leap, I propose that there are also recognition primitives at work when a reader with suitable training apprehends that "a concept is present" in a text. The research necessary to identify these primitives is likely to involve studies of psychology and language and may need to develop very slowly, beginning with the study of how children acquire language. There is already work under way on "reverse–engineering" the human process of language acquisition to understand what kinds of computations are being performed. These efforts aim to develop computer algorithms which exhibit the same successes and failures as human language learners. In this connection ¡t is suggestive to note that a text that has been preprocessed (stop words removed and stemmed) looks a great deal like the utterances of an infant who has not yet mastered the role of function words in language. The development of concept recognition in algorithmic form will have to progress, with great effort, from matching the abilities of infants, through progressive improvements, to matc-

hing or exceeding the abilities of the author and readers of an article such as this one.

A Phase Change in the World of Information

Physicists speak of "phase changes" when, for example, a gas condenses into a liquid or a liquid suddenly becomes a rigid crystalline structure. The development of machine–readable records and of the worldwide network creates the potential for a phase transition in the state of human knowledge.

With no increase in the amount of what is "known and recorded" (as opposed to simply known by one person privately) there could be a dramatic phase transition in what is "known to the world." With worldwide access to virtually all publicly available electronic information, mutually relevant and synergistic atoms of information could be instantly brought together. In the hands (minds?) of the right users this information could solve problems and improve conditions for people anywhere in the world. Of course, the ultimate limiting factor is the capacity of the human mind to absorb and transform information. The effect of this global change will be to sharply reduce the latency or cycle time between the creation and application of relevant information.

As a practical matter, the path to this global phase transition will be built by work on problems less grandiose than improving the human condition. For example, the phase transition might occur first in surveillance activities. It would be excellent, on the other hand, to have such progress in the health sciences. The enormous engineering files of the Department of Energy or the Strategic Defense Initiative are also ripe for conversion to digital form and for the phase change to take place.

However, this is a chicken or egg proposition. Our IR technologies will only flourish as enormous, potentially coherent dataspaces become available, but the motivation for putting them into machine–readable form will not be strong until the IR technologies are mature. So the impulse may come from agencies that have a less public mission –e.g., those charged with monitoring the flow of electronic messages for indications of terrorism and conspiracy. At the same time, we must he wary of the potential for superficiality that is encouraged by the possibility of "surfing" the Internet or wandering through some virtual reality. The systems that give us the incredible cornputational power and the sophisticated retrieval algorithms needed to wander in this way were created by people schooled in rigorous thought. I hope that the acquisition and organization of information will not come to be seen as an adequate substitute for such rigor. (For a brief polemic on this issue, see Florman.) The transformative power of this potential phase change is beyond my powers of speculation. The possibility of making it a reality turns what could he a dry academic pursuit into an exciting quest.

Bibliography

Agosti, M.; Colloti, R.; Gradenigo, G. (1991). "A Two–Level Hypertext Retrieval Model for Legal Data". In: Bookstein, A.; Chiaramella, Y.; Salton, G.; Raghavan, V.V., (eds.). SIGIR'91: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) 14th Annual International Conference on Research and Development in Information Retrieval; (October 1991, 13–16,); Chicago, W. New York, NY: ACM; (1991; pag.316–325). ISBN: 0–89791448–1.

Anick, P.G. (1992). "Lexicon Assisted Information Retrieval for the Help–Desk". In: Proceedings of IEEE CAIA–92 Workshop on Artificial Intelligence for Customer Service and Support; Monterey, CA.

Anwyl, P.; Kanaya, M.; Morita, T. (1990). "Automatic Keyword Assignment in English Documents". In: Proceedings of the 4lst Conference of the Information Processing Society of Japan. (March, 1987–1988).

Badecker, W.; Caramazza, A. (1989). "A Lexical Distinction between Inflection and Derivation". Linguistic Inquiry. (Winter 1989, num.20, vol. I, pag.108–116.); ISSN: 0024–3892.

Belkin, N.; Nicholas, J.; Cool, C.; Croft, W.B.; Callan, J.P. (1993). The Effect of Multiple Query Representations on Information Retrieval System Performance. See reference: Korfhage et al., (eds.). (pag. 339–346).

Belkin, N.; Nicholas, J.; Croft, W.B. (1987). "Retrieval Techniques". In: Williams, Martha E., ed. Annual Review of Information Science and Technology; (Volume 22). Amsterdam, The Netherlands: Elsevier Science Publishers for the American Society for Information Science; (1987; pag.109–145). ISSN: 0066–4200; ISBN: 0–444–70302–0.

Belkin, N; Nicholas, J.; Ingwersen, P.; Petjersen, A.M.; (eds.) (1992). "SIGIR'92: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/ SIGIR)". 15th Annual International Conference on Research and Development in Information Retrieval; (1992, June 21–24,); Copenhagen, Denmark. New York, NY: ACM; (1992; pag.353). ISBN: 0–89791–523–2.

Belkin, N.; Nicholas, J.; Kantor, P.B.; Cool, C; Quatrain, R. (1994). Combining Evidence for Information Retrieval. See reference: Harman, ed., (1994; pag. 35–44).

Bertino, E.; Negri, M.; Pelacatti, G.; Sbatella, L. (1992). "Object–Oriented Query Languages: The Notion and the Issues". IEEE Transactions on KnowIedge and Data Engineering. (1992, June, num.4, vol. 3, pag.223–237,). ISSN: 1041–4347.

Bookstein, A. (1983). "Outline of a General Probabilistic Retrieval Model". Journal of Documentation. (1983, num.39, vol.2, pag.63–72). ISSN: 0022–0418.

Botafogo, R.A. (1993). Cluster Analysis for Hypertext Systems. See reference: Korhfage et al., (eds.), (pag.116–125).

Botafogo, Rodrigo.; Scheiderman, B. (1991). "Identifying Aggregates in Hypertext Structures". In: Hypertext'91: Proceedings of the Association for Computing Machinery (ACM) 3rd Conference on Hypertext; (1991, December 15–18); San Antonio, TX. New York, NY: ACM, (1991; pag.63–74). ISBN: 0–89791–461–9.

Bruza, P.D.; Van Der Gaac, L.C. (1993). Efficient Context–Sensitive Plausible Inference for Information Disclosure. See reference: Korfhage et al., (eds.), (pag.12–21).

Buckland, M.; Butler, M. H.; Norgard, B.A.; Plaunt, Christian, J. (1993). "OASIS: Prototyping Graphical Interfaces to Networked Information". In: Bonzi, Susan, ed. ASIS'93: Proceedings of the American Society for Information Science (ASIS) 56th Annual Meeting; (1993, October 24–28, vol.30); Columbus, OH. Medford, Nj: Learned Information, Inc. for ASIS; (1993; pag.204–210). ISSN: 0044–7870; ISBN: 0938734–78–4.

Buckles, B.P.; Petry, F.E. (1982). "A Fuzzy Representation of Data for Relational Databases". Fuzzy Sets and Systems. (1982, May, num.7, vol.3, pag.213–226). ISSN: 0165–0114.

Buckley, C.; Salton, G.; Allan, J. (1993). Automatic Retrieval with Locality Information Using Smart. See reference: Harman, (ed.), (1993ª; pag.59–72).

Burkowski, F.J. (1992). Retrieval Activities in a Data base Consisting of Heterogeneous Collections of Structured Text. See reference: Belkin et al., (eds.), (1992; pag.112–125).

Callan, J.P.; Croft, W.B. (1993). An Evaluation of Query Processing Strategies Using the Tipster Collection. See reference: Korfhage et al., (eds.), (pag.347–355).

Can, F.; Ozkarahan, E.A. (1987). "Computation of Term/Document Discrimination Values by Use of the Cover Coefficient Concept". Journal of the American Society for Information Science. (1987, num.38, vol.3, pag.171–183). ISSN: 0002–8231.

Can, F.; Ozkarahan, E.A. (1990). Concepts and Effectiveness of the Cover–Coefficient–Based Clustering Methodology for Text Databases. ACM Transac-

tions on Database Systems. (1990, December, num.15, vol.4, pag.483–517). ISSN: 0362–5915.

Cavnar, W. B. (1994). N–Gram–Based Text Filtering for Trec–2. See reference: Harman, (ed.), (1994; pag.171 –180).

Chang, S.C.; Dediu, H.; Azzam, H.; DU, M.W. (1993). Multilevel Ranking in Large Text Collections Using Fairs. See reference: Harman, (ed.), (1993; pag.329–336).

Cherik, M. (1989). Optimal Decision and Detection in the Decentralized Case. Cleveland, OH: Case Western Reserve University; (1989; pag.211). (Ph.D.dissertation). Available from: University Microfilms Ann Arbor, MI. (UMI order no. 90–04501).

Chiaramella, Y.; Nie, J.(1990). "A Retrieval Model Based on an Extended Modal Logic and Its Application to the RIME Experimental Approach". In: Vidick, jean–Luc, (ed.). Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) l3th International Conference on Research and Development in Information Retrieval; (1990, September 5–7); Brussels, Belgium. New York, NY: ACM; (1990; pag.25–43). ISBNT.: 0–89791–408–2.

Cleverdon, C.W. (1962). Reputt on the Testing and Analysis of an Investigation into the Comparative Efficiency of Indexing Systems. Cranfield, England: Collego of Aeronautics; (1962; pag.305). LC: 63–60414.

Cleverdon, C.W.; Mills, J.; Keen, E.M. (1966). Factors Determining the Performance of Indexing Systems. Cranfield, England: ASLIB; (1966, 2 Vols., vol.1: Design; vol.2: Test Results). LC: 67–81732; OCLC: 3911240.

Cooper, W.S.; Chen, A.; Cey, F.C. (1994). Full Text Retrieval Based on Probabilistic Equations with Coefficients Fitted by Logistic Regression. See reference: Harman, (ed.), (1994; pag.57–66).

Croft, W.B.; Krovetz, R.; Turtle, H.R. (1990). "Interactive Retrieval of Complex Documents". Information Processing and Management. (1990, num.26, vol.5, pag.593–613). ISSN: 0306–4573.

Croft, W.B.; Smith, L.A.; Turtle, H.R. (1992). A Loosely Coupled Integration of a Text Retrieval System and an Object–Oriented Database System. See reference: Belkin et al., (eds.), (1992; pag.223–232).

Croft, W.B.; Turtle, H.R. (1989). "A Retrieval Model for Incorporating Hypertext Links". In: Hypertext '89: Proceedings of the Association for Computing Machinery (ACM) 2nd Conference on Hypertext; (1989, November 5–8);

Pittsburgh, PA. New York, NY: ACM; (1989; pag.213–224). ISBN: 0–89791–339–6.

Croft, W.B.; Turtle, H. R.; Lewis, D.D. (1991). "The Use of Phrases and Structured Queries in Information Retrieval". In: Bookstein, A.; Chiaramella, Y.; Salton, G.; Raghavan, V.V., (eds.). SIGIR'91: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) l4th Annual International Conference on Research and Development in Information Retrieval; (1991, October 13–16); Chicago, IL. New York, NY: ACM; (1991; pag.32–45). ISBN: 089791–448–1.

Crouch, C.J.; Yang, B. (1992). Experiments in Automatic Statistical Thesaurus Construction. See reference: Belking et al., (eds.), (1992; pag. 77–88).

Cutting, D.R.; Karcer, D. R.; Pedersen, J.0. (1993). Constant Interaction–Time Scatter/Gather Browsing of Very Large Document Collections. See reference: Korfhage et al., (eds.) (pag.126–134).

Cutting, D.R.; Karger, D.R.; Pedersen,J.O.; Tuckey, J.W. (1992). Scatter/Gather: A Cluster–Based Approach to Browsing Large Document Collections. See reference: Belkin et al., (eds.), (1992; pag.318–329).

Cutting, D.; Douglass, R.; Pedersen, J.0. (1990). "Optimizations for Dynamic Inverted Index Maintenance". In: Vidick, jean–Luc, ed. SIGIR'90: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) l3th International Conference on Research and Development in Information Retrieval; (1990, September 5–7); Brussels, Belgium. New York, NY: ACM; 1990. 405–411. ISBN: 089791–408–2.

Deerwester, S.; Dumais, S.T.; Furnas, G.W.; Landauer,T.; Harshaman, R. (1990). "Indexing by Latent Semantic Analysis". Journal of the American Society for Information Science. (1990, num.41, vol.6, pag.391–407). ISSN: 0002–8231.

Drabenstott, K.M.; Vizine–Coetz, D. (1990). "Search Trees for Subject Searching in Online Catalogs". Library Hi–Tech. (1990, num.8, vol.3, pag.7–20). ISSN: 0737–8831.

Dumais, S.T. (1994). Latent Semantic Indexing and Trec–2. See reference: Harman, (ed.), (1994; pag.105–116).

Efthimiadis, E.N. (1990). "Online Searching Aids: A Review of Front Ends, Gateways and Other Interfaces". Journal of Documentation. (1990, September, num.46, vol.3, pag.218–262). ISSN: 0022–0418.

Efthimiadis, E.N. (1993). A User–Centred Evaluation of Ranking Algorithms for Interactive Query Expansion. See reference: Korfhage et al., (eds.), (pag.146–159).

Evans, D.A.; Lefferts, Robert, G. (1994). Design and Evaluation of the Clarit–Trec–2 System. See reference: Harman, (ed.), (1994; pag.137–150).

Feller, W. (1968). An Introduction to Probability Theory and Its Applications. (3rd ed.). New York, NY: John Wiley & Sons; (1968. 2 vols.), LC: 68–11708.

Florman, S.C. (1994). "Odysseus in Cyberspace". Technology Review. (1994, April, num.97, vol.3, pag.65). ISSN:0040–1692.

Fox, E.A. (1983). Extending the Boolean and Vector Space Models of Information Retrieval with P–norm Queries and Multiple Concept Types. Ithaca, NY: Cornell University; (1983; pag.86). (Ph.D. dissertation). Available from: University Microfilms, Ann Arbor, MI. (UMI order no. 83–28584).

Fox, E.A.; France, R.K.; Sahle, E.; Daciud, A.; Cline, B.E. (1993). Development of a Modern Opac: From Revtolc to Marian. See reference: Korfhage et al., (eds.), (pag.248–259).

Fox. E.A.; Shaw, J.A. (1994). Combination of Multiple Searches. See reference: Harman, (ed.), (1994; pag.243–252).

Frakez, W.B.; Baeza–Yates, R. (1992). Information Retrieval: Data Structures and Algorithms. Englewood Cliffs, Nj: Prentice Hall; (1992; pag.504). ISBN: 0–13–463837–9.

Frei, H.P.; Schäuble, P. (1991). "Determining the Effectiveness of Retrieval Algorithms". Information Processing and Management. (1991, num.27). 2/U153–164. ISSN: 0306–4573.

Fuhr, N. (1989a). "Models for Retrieval with Probabilistic Indexing". Information Processing and MInformation Retrieval, Computational and Theoretical Aspects. New York, NY: Academic Press; (1978; pag.344). ISBN: 0~12335750–0.

Fuhr, N. (1989b). "Optimum Polynomial Retrieval Functions Based on the Probability Ranking Principle". ACM Transactions on Information Systems. (1989, July, num.7, vol.3, pag.183–204). ISSN:0734–2047.

Fuhr, N. (1990). "A Probabilistic Framework for Vague Queries and Imprecise Information in Databases". In: McLeod, D.; Sacks–Diavis, R.; Schek, H., (eds.) Proceedings of the 16th International Conference on Very Large Databases;

(1990, August 13–16); Brisbane, Australia. Palo Alto, CA: Morgan Kaufmann; 1990. 696–707. ISBN: 0–55860–1,19–X.

Fuhr, N. (1992). Integration of Probabilistic Fact and Text Retrieval. See reference: Belkin et al., (eds.), (1992; pag.211–222).

Fuhr, N. (1993). A Probabilistic Relational Model for the Integration of IR and Databases. See reference: Korfhage et al., (eds.), (pag.309–317).

Fuhr, N.; Buckley, C. (1993). Optimizing Document Indexing and Search Term Weighting Based on Probabilistic Models. See reference: Harman, (ed.), (1993; pag.89–99).

Fuhr, N.; Pfeifer, U.; Bremkamp, C.; Pollmann, M. (1994). Probabilistic Learning Approaches for Indexing and Retrieval with the Trec–2 Collection. See reference: Harman, (ed.), (1994; pag.67–74).

Fujii, H.; Croft, W. B. (1993). A Comparison of Indexing Techniques for Japanese Text Retrieval. See reference: Korfhage et al., (eds.), (pag.237–246).

Fuller, M.; Mackie, E.; Sacks–Davis, R.; Wilkinson, R. (1993). Structured Answers for a Large Structured Document Collection. See reference: Korfhage et al., (eds.), (pag.204–213).

Furnas, G.W.; Landauer, T.K.; Gomez, L.M.; Dumais, S.T. (1987). "The Vocabulary Problem in Human–System Communication". Communications of the ACM. (1987, num.30, vol.11, pag.964–971). ISSN: D001–0782.

Gallant, S.I.; Caid, W.R.; Carleton, J.; Gutschow, T.W.; Hecht–Nielsen, R.; Qing, K.P.; Sudbeck, D. (1994). Feedback and Mixing Experiments with MatchPlus. See reference: Harman, (ed.), (1994; pag.101–104).

Goldfarb, C. F. (1990). The SGML Handbook. New York, NY: Oxford University Press; (1990; pag.664). ISBN: 0–19–853737–9.

Haines, D.; Croft, W. B. (1993). Relevance Feedback and Inference Networks. See reference: Korfhage et al., (eds.), (pag.2–11).

Harada, T.; Hosonoko, K.; Tamura, S. (1989). "Developing an Automatic Construction Method of the BT–NIT Relations from Japanese Compound Nouns". SIGNL Record of the Information Processing Society of Japan. (1989, num.70, vol.5, pag.1–8). (In Japanese).

Harman, D. (1988). Towards Interactive Query Expansion. In: Chiaramella, Y., (ed.). "Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval". (ACM/SIGIR) Ilth International Conference on Research and Development in Information Retrieval; (1988,

June, pag.13–15); Grenoble, France. New York, NY: ACM; 1988. 321–331. ISBN: 2–7061–0309–4.

Harman, D. (1992). "The Darpa Tipster Project". SIGIR Forum. (1992 Fall; num.6, vol.2, pag.26–28). Available from: Association for Computing Machinery, 1515 Broadway, New York, NY 10036.

Harman, D., (ed). (1993a.) The 1st Text Retrieval Conference (Trec–1); (1992, November 4–6); Gaithersburg, MD. Gaithersburg, MD: National Institute of Standards and Technology; 1993. 518p. (NIST Special Publication 500–207). ISBN: 0–16–042265–5; NTIS: PB 93–191641.

Harman, D. (1993b). Overview of the First Trec Conference. See reference: Korfhage et al., (eds.), (pag.36–47).

Harman, D., (ed.). (1994). The 2nd Text Retrieval Conference (Trec–2); (1993, August 31–September 2); Gaithersburg, MD. Caithersburg, MD: National Institute of Standards and Technology; 1994. 497p. (NIST Special Publication 500–215). CODEN: NSPUE2; NTIS: P1394–178407.

Harney, R. C., (ed.). (1990). Sonsor Fusion III: Sponsored by SPIE–the International Society for Optical Engineering; (1990, April 19–20); Orlando, FL. Bellingham, WA: SPIE; 1990. 230p. (Proceedings/SPIE vol.1306). ISSN: 0277–786X; ISBN: 0–8194–0357–1.

Heaps, H.S. (1978). Information Retrieval, Computational and Theoretical Aspects. New York, NY: Academic Press; (1978; pag.344). ISBN: 0~12335750–0.

Hearst, M.A. (1992). "Automatic Acquisition of Hyponyms from Large Text Corpora". In: COLING–92: Proceedings of the 15th International Conference on Computational Linguistics; (1992, August 23–28); Nantes, France. Crenoble, France: ICCL; 1992. 539–545. OCLC: 27857672.

Hearst, M.A. (1993a). "Cases as Structured Indexes for Full–Length Documents". In: Proceedings of the American Association for Artificial Intelligence 1993 Spring Symposium on Case–Based Reasoning and Information Retrieval; Stanford, CA.

Hearst, M.A. (1993b). TextTiling. Berkeley, CA: University of California at Berkeley; (1993). (Sequoia 2000 Technical Report 93/24).

Hearst, M.A.; Plaunt, C. (1993). Subtopic Structuring for Full–Length Document Access. See reference: Korfhage et al., (eds.), (pag.59–68).

Hjerppe, R. (1986). "Hypercatalog and Three Meta–Sckernata for Database Views: Knowledge Organising, Collection Derived and User Establihed Structures. In: Kinsella, Janet, ed. Online Public Access to Library Files. Oxford, England: Elsevier International Bulletins; 1986. 101–110. ISBN: 0–946395–25–X.

Hoffman, M.M.; O'Corman, L.; Story, C.A.; Arnold, J.Q.; Macdonald, N.H. (1993). "The RightPages Service: An Image–Based Electronic Library". Journal of the American Society for Information Science. (1993, September; num.44, vol.8, pag.446–452). ISSN: 0002–8231.

Hull, D. (1993). Using Statistical Testing in the Evaluation of Retrieval Experiments. See reference: Korfhage et al., (eds.), (pag.329–338).

International Organization for Standardization. (1986). Information Processing Systems–Text and Office Systems–Standard Generalized Markup Language (SGML). Geneva, Switzerland: International Organization for Standardization,– (1986, October). (International Standard 8879).

Jacobs, P.S.; Krupka, G.R.; Rau, L.F. (1991). "Lexico–Semantic Pattern Matching as a Companion to Parsing in Text Understanding". In: Proceedings of the 4th DARPA Speech and Natural Language Workshop; (1991, February). San Mateo, CA: Morgan Kaufmann; 1991. 337–342. ISBN: 1–55860–207–0.

Jacobs, P.S.; Krupka, G.R.; Rau, L.F. (1993). A Boolean Approximation Method for Query Construction and Topic Assignment in Trec. See reference: Harman, (ed.), (1993a; pag.297–308).

Janes, J.W. (1993). "On the Distribution of Relevance judgments". In: Bonzi, S., (ed.). ASIS '93: Proceedings of the American Society for Information Science (ASIS) 56th Annual Meeting; (vol.30; 1993 October 24–28); Columbus, OFI. Medford, Nj: Learned Information, Inc. for ASIS; 1993. 104–114. ISSN: 0044–7870; ISBN: 0–938734–78–4.

Julesz, B. (1991). "Early Vision and Focal Attention". Reviews of Modern Physics. (1991, July); (num.63, vol.3, pag.735–772). ISSN: 0034–6861.

Kantor, P.B. (1981). "The Logic of Weighted Queries". IEEE Transactions on Systems, Man and Cybernetics. (1981; num.11, vol.12, pag.816–821). ISSN: 0018–9472.

Kantor, P.B. (1984). "Maximum Entropy Principle and the Optimal Design of Automated Information Retrieval Systems". Information Technology: Research and Development. (1984; num.3, vol.2, pag.88–94). ISSN: 0144–817X.

Kantor, P.B. (1987). "A Model for the Stopping Behaviour of Users of Online Systems". Journal of the American Society for Information Science. (1987; num.38, vol.3, pag.211–214). ISSN: 0002–8231.

Kantor, P.B. (1992). "Two Heads Are Better Than One: The Potential of Data Fusion Concepts for Improvement of Online Searching". In: Williams, M.E., (comp.). Proceedings of the 13th National Online Meeting; (1992, May 5–7); New York, NY. Medford, Nj: Learned Information, Inc.; (1992; pag.147–151). ISBN: 0–938734–63–6.

Kantor, P.B. (1993). "The Adaptive Network Library Interface: A Historical Overview and Interim Report". Library Hi Tech. (1993; num.11, vol.3, pag.81–92). ISSN: 0737–8831.

Kantor, P.B. (1994). "Data Fusion in Information Retrieval: Issues of justification, Simulation and Evaluation". New Brunswick, Nj: Rutgers University; 1994. (Alexandria Project Laboratory Technical Report APLab/ TR–94/1). Available from: APLab/SCILS, Rutgers University, 4 Huntington St., New Brunswick, Nj 08903.

Kantor, P.B.; Blanckendecker, R.; Cherick, M. (1988). Sensor Calculus. Cleveland, OH: Tantalus Inc.; (1988, June, pag.88). (Report Tantalus/CT–88/3). NTIS: AD–197250. Also available from: Tantalus Inc., 362 N. 4th Ave., Highland Park, Nj 08904.

Kantor, P.B.; Lee, J.J. (1986). "The Maximum Entropy Principle in Information Retrieval". In: Rabitti, F., (ed.) Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) 9th Conference on Research and Development in Information Retrieval; (1986, September 8–10); Pisa, Italy. New York, NY: ACM; 1986. 269–274. ISBN: 0–89791–187–3.

Katzer, J.; McGill, M.J.; Tessier, J.A.; Frakes, W; Dascupta,P. (1982). "A Study of the Overlap among Document Representations". Information Technology: Research and Development. (1982, October; num.1, vol.4, pag.261–274). ISSN: 0144–817X.

Kauffman, L.; Rousseeuw, P.J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. New York, NY: Wiley; (1990; pag.342). ISBN: 0–471–87876–6.

Kilpelainen, P.; Linden, G.; Mannila, H.; Nikunen, E. (1990). "A Structured Document Database System". In: Furuta, R., (ed.) EP90: Proceedings of the International Conference on Electronic Publishing, Document Manipulation and Typography; (1990, September 18–20); Gaithersburg, MI). Cambridge, England: Cambridge University Press; (1990; pag.139–151). ISBN: 0–521–40246– 8.

Koenig, M.E.D. (1990). "Linking Library Users: A Culture Change in Librarianship". American Libraries. (1990, October; num.21, vol.9, pag.844–845, 847, 849). ISSN: 0002–9769.

Korfhage, R.; Rasmussen, E.; Willet, P.; (eds.). (1993). "SIGIR '93: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval". (ACM/SIGIR) 16th Annual International Conference on Research and Development in Information Retrieval; (1993, June 27–july 1); Pittsburgh, PA. New York, NY: ACM; (1993; pag. 361). ISBN: 0–89791–605–0.

Kraft, D., (ed.).(1950). Journal of the American Society for Immigration Science. New York, NY: John Wiley & Sons. (Formerly titIed American Documentation). ISSN: 0002–8231.

Krovetz, R. (1992). "Sense–Linking in a Machine Readable Dictionary". In: Proceedings of the Association for Computational Linguistics 30th Annual Meeting; (1992, June 28–july 2); Newark, DE. Morristown, Nj: Association for Computational Linguistics; (1992; pag.330–332).

Krovetz, R. (1993). Viewing Morphology as an Inference Process. See reference: Korfhage et al., (eds.). (pag.191–202).

Krovetz, R.; Croft, W.B. (1992). "Lexical Ambiguity and Information Retrieval". ACM Transactions on Information Systems. (1992, April; num.10, vol.2, pag.115–141). ISSN: 1046–8188.

Kupiec, J.M. (1992a). "Hidden Markov Estimation for Unrestricted Stochastic Context–Free Grammars". In: ICASSP–92: Proceedings of the 1992 IEEE International Conference on Acoustics, Speech and Signal Processing; (1992, March 23–26); San Francisco, CA. New York, NY: IEEE; (1992, vol.1, pag.177–180). ISBN: 0–7803–0532–9.

Kupiec, J.M. (1992b). "Robust Part–of–Speech Tagging Using a Hidden Markov Model". Computer Speech and Language. (1992 July; num.6, vol.3, pag.225–242). ISSN: 0885–2308.

Kupiec, J.M. (1993). Murax: A Robust Linguistic Approach for Question Answering Using an On–line Encyclopaedia. See reference: Korhfage et al., (eds.), (pag.181–190).

Lee, J.H.; Kim, W.Y.; Kim, M.; Lee, Y.J. (1993). On the Evaluation of Boolean Operators in the Extended Boolean Retrieval Framework. See reference: Korfhage et al., (eds.), (pag.291–297).

Lee, J.J.; Kantor, P.B. (1991). "A Study of Probabilistic Information Retrieval Systems in the Case of Inconsistent Expert Judgments". Journal of the American Society for Information Science. (1991; num.42, vol.3, pag.166–172). ISSN: 0002–8231.

Lesk, M.E. (1986). "Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone". In: Proceedings of SIGIDOC. (1986; pag.24–26).

Lindley, D.V. (1965). Introduction to probability and Statistics from a Bayesian Viewpoint. Cambridge, England: Cambridge University Press; (1965, 2 vols.); LC: 64–24313.

Linoff, C.; Stanfill, C. (1993). Compression of Indexes with Full Positional Information in Very Large Text Databases. See reference: Korfhage et al., (eds.), (pag.88–95).

Lovins, J.B. (1968). "Development of a Stemming Algorithm". Mechanical Translation and Computational Linguistics. (1968, March; num.11, page.22–31). ISSN: 0543–2073.

Luhn, H.P. (1958). "The Automatic Creation of Literature Abstracts". IBM journal of Research and Development. (1958, April; num.2, vol2, page.159–165). ISSN: 0018–8646.

Luhn, H.P. (1959). "Auto–Encoding of Documents for Information Retrieval Systems". In: Boaz, M., (ed.) Modem Trends in Documentation. London, England: Pergamon Press; (1959; pag.45–58). LC: 59–10081.

Lunin, L.; Fox, E.A., (eds.). (1993). Perspectives on Digital Libraries Journal of the American Society for Information Science. (1993, September; num.44, vol.8, pag.440–491). ISSN: 0002–8231.

Mancas–Arian, O.L. (1965). "Linear and Non–Linear Separation of Pattems by Linear Programming". Operations Research. (1965; num.13, pag.444–452). ISSN: 0030–364X.

Maruyama, H.; Watanabe, H.; Ocino, S. (1990). "An Interactive Japanese Parser for Machine Translation". In: Karlgren, H., (ed.) Proceedings of the 13th International Conference on Computational Linguistics. Helsinki, Finland: University of Helsinki; (1990; vol.2, pag.257–262). ISBN: 952–90–2028–7.

Mckeown, K.R. (1985). Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text. Cambridge, England: Cambridge University Press; (1985; pag.246). ISBN: 0–52130116–5.

Mckeown, K.R.; Feiner, Steven K.; Robin J.; Seligmann, Doree, D.; Taneblatt, M. (1992). "Generating Cross–References for Multimedia Explanation". In: Proceedings of the 10th National Conference on Artificial Intelligence; (1992, July 12–16); San Jose, CA. Menlo Park, CA: AAAI Press; (1992; pag.9–16). ISBN: 0–262–51063–4.

Miller, G.A.; Beckwith, R.; Fellbaum, C.; Gross, D.; Miller, K.J. (1990a). Five Papers on WordNet. Princeton, Nj: Princeton University Computer Science Laboratory; (1990, July).

Miller, G.A.; Beckwith, R.; Fellbaum, C.; Gross, D.; Miller, K.J. (1990b). "Introduction to WordNet: An On–line Lexical Database". International Journal of Lexicography. (1990; num.3, vol.4, pag.235–244). ISSN: 0950–3846.

Myler, H.R.; Weeks, A.R. (1993). The Pocket Handbook of Image Processing Algorithms. In C. Englewood Cliffs, Nj: Prentice–Hall; (1993). ISBN: 0–13–649240–3.

Negoita, C.V. (1985). Expert Systems and Fuzzy Systems. Menlo Park, CA: Benjamin/Cummings; (1985; pag.190). ISBN: 0–8053–6840–X.

O'Connor, J. (1975). "Retrieval of Answer–Sentences and Answer–Figures from Papers by Text Searching". Information Processing and Management. (1975; num.11vol.5/7, pag.155–164). ISSN:0306–4573.

Oddy, R.N.; Balakrishnan, B. (1991). PThomas: "An Adaptive Information Retrieval System on the Connection Machine". Information Processing and Management. (1991; num.27, vol.4, pag.317–335). ISSN: 03064573.

Ocawa, Y.; Bessho, A.; Hirose, M. (1993). Simple Word Strings as Compound Keywords: An Indexing and Ranking Method for Japanese Texts. See reference: Korfhage et al., (eds.), (pag.227–236).

Ocawa, Y.; Morita, T.; Kobayashi, K. (1991). "A Fuzzy Document Retrieval System Using the Keyword Connection Matrix and Its Learning Method". Fuzzy Sets and Systems. (1991; num.39, vol.2, pag.163–179). ISSN: 0165–0114.

Ozkarahan, E.A.; Can, F. (1986). "An Automatic and Tunable Document Indexing System". In: Rabitti, F., (ed.). Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) 9th Conference on Research and Development in Information Retrieval; (1986, September 8–10); Pisa, Italy. New York, NY: ACM; 1986. 234–243. ISBN: 0–89791–187–3.

Paice, C.D. (1990). "Constructing Literature Abstracts by Computer: Techniques and Prospects". Information Processing and Management. (1990;num.26, vol.1, pag.171–186). ISSN:0306–4573.

Paice, C.D.; Jones, P.A. (1993). The Identification of Important Concepts in Highly Structured Technical Papers. See reference: Korfhage et al., (eds.), (pag.69–78).

Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. San Mateo, CA: Morgan Kaufmann; (1988; pag.552). ISBN: 0–934613–73–7.

Peat, H.J.; Willm, P. (1991). "The Limitations of Term Cooccurrence Data for Query Expansion in Document Retrieval Systems". Journal of the American Society for Information Science. (1991; num.42, vol.5, pag.378–383). ISSN: 0002–8231.

Pedersen, G.S. (1993). A Browser for Bibliographic Information Retrieval Based on an Application of Lattice Theory. See reference: Korfhage et al., (eds.), (pag.270–279).

Popcivic, M.; Willet, P. (1992). "The Effectiveness of Stemming for Natural–Language Access to Slovene Textual Data". Journal of the American Society for Information Science. (1992; num.43, vol.5, pag.384–390). ISSN: 0002–8231.

Porter, M.F. (1980). An Algorithm for Suffix Stripping Program. (1980, July; num.14, vol.3, pag.130–137). ISSN: 0033–0337.

Prade, H.; Testemale, C. (1984). "Generalizing Database Relational Algebra for the Treatment of Incomplete or Uncertain Information and Vague Queries". Information Sciences. (1984, November; num34, vol.2, pag.115–143).ISSN: 0020–0255.

Qui, Y.; Frei, H.P. (1993). Concept Based Query Expansion. See reference: Korfhage et al., (ed.), (pag.160–169).

Raymond, D.R. (1990). "LECTOR–An Interactive Formatter for Tagged Text. Waterloo, Ontario: University of Waterloo"; (1990). (Technical Report OED–90–02). Available from: Centre for the New Oxford Dictionary and Text Research, University of Waterloo.

Ro, J.S. (1988). "An Evaluation of the Applicability of Ranking Algorithms to Improve the Effectiveness of Full–Text Retrieval. Part 1. On the Effectiveness of Full–Text Retrieval". Journal of the American Society for Information Science. (1988; num.39, vol.2, pag.73–78). "Part II. On the Effectiveness of Ranking Algorithms on Full–Text Retrieval". Journal of the American Society for Information Science. (1988;num.39, vol.3, pag.147–160). ISSN:0002–8231.

Robertson, S.E. (1990). On Term Selection for Query Expansion. journal of Documentation. (1990, December; num.46, vol.4, pag.359–364). ISSN: 00220418.

Robertson, S.E.; Bovey, J.D.; Thompson, C.L.; Macaskill, M.J. (1986). "Weighting, Ranking and Relevance Feedback in a Front–End System". Journal of Information Science. (1986; num.12, pag.71–75). ISSN: 0165–5515.

Robertson, S.E.; Maron, M.E.; Cooper, W.S. (1982). "Probability of Relevance: A Unification of Two Competing Models for Document Retrieval". Information Technology: Research and Development. (1982, January; num.1, vol.1, pag.1–21). ISSN: 0144–817X.

Robertson, S.E.; Walker, S.; Hancock–Beaulieu, Micheline; Gull, A.; Lau, M. (1993). Okapi at Trec. See reference: Harman, (ed.), (1993a; pag.21–30).

Robertson, S.E.; Walker, S.; Jones, S.; Hancockbeaulieu, M.; Gatford, M. (1994). Okapi at Trec–2. See reference: Harman, (ed.), (1994; pag.21–34).

Rocchio, J.J., JR. (1971). Relevance Feedback in Information Retrieval. In: Salton, Gerard, (ed.) The Smart Retrieval System: Experiments in Automatic

Document Processing. Englewood Cliffs, Nj: Prentice–Hall; (1971; pag.313–323). LC: 70–159122.

Rose, D.E.; Mander, R.; Oren, T.; Ponceleón, D.B.; Salomon, GMA; Wong, Y.Y. (1993). Content Awareness in a File System Interface: Implementing the 'Pile" Metaphor for Organizing Information. See reference: Korfhage et al., (eds.), (pag.260–269).

Salton, G. (ed.). (1971). The Smart Retrieval System: Experiments in Automatic Document Processing. Englewood Cliffs, Nj: Prentice–Hall; (1971; pag.556). LC: 70–159122.

Salton, G.; Allan, J.; Buckley, C. (1993). Approaches to Passage Retrieval in Full Text Information Systems. See reference: Korfhage et al., (eds.), (pag.49–58).

Salton, G.; Allan, J.; Buckley, C. (1988). "Term–Weighting Approaches in Automatic Text Retrieval". Information Processing and Management. (1988; num.24, vol.5, pag.513–523). ISSN: 0306–4573.

Salton, G.; Allan, J.; Buckley, C. (1990). "Improving Retrieval Performance by Relevance Feedback". Journal of the American Society for Information Science. (1990; num.41, vol.4, pag.288–297). ISSN: 0002–8231.

Salton, G.; Allan, J.; Buckley, C. (1991a). "Automatic Text Structuring and Retrieval: Experiments in Automatic Encyclopedia Searching". In: Bookstein, A.; Chiaramella, Y.; Salton, C.; Raghavan, V.V., (eds.). SIGIR'91: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) 14th Annual International Conference on Research and Development in Information Retrieval; (1991, October 13–16); Chicago, IL. New York, NY: ACM; (1991; pag.21–30). ISBN: 089791–448–1.

Salton, G.; Allan, J.; Buckley, C. (1991b). "Global Text Matching for Information Retrieval". Science. (1991, August 30; num.253, vol.5023, pag.1012–1015). ISSN: 0036– 8075.

Salton, G.; Allan, J.; Buckley, C.; Fox, E.A. (1983). "Automatic Query Formulations in Information Retrieval". Journal of the American Society for Information Science. (1983; num.34, vol.4, pag.262–280). ISSN: 0002–8231.

Salton, G.; McGill, M.J. (1983). Introduction to Modem Information Retrieval. New York, NY: McGraw–Hill; (1983; pag.448). ISBN: 007–054484–0.

Saracevic, T., (ed.). (1963). Information Processing and Management. Oxford, England: Pergamon Press. (Formerly titled Information Storage and Retrieval). ISSN: 0306–4573.

Saracevic, T.; Kantor, P.B. (1988). "A Study of Information Seeking and Retrieving. III. Searchers, Searches, and Overlap". Journal of the American Society for Information Science. (1988; num.39, vol.3, pag.197–216). ISSN: 0002–8231.

Schamber, L. (1994). "Relevance and Information Behaviour". In: Williams, M.E., (ed.) Annual Review of Information Science and Technology: vol.29. Medford, Nj: Learned Information, Inc. for the American Society for Information Science; (1994; pag.3–48). ISSN: 0066–4200; ISBN: 0938734–91–1. –

Schäuble, P. (1989). The Compatibility of Retrieval Functions, Preference Relations, and Document Descriptions. Zurich, Switzerland: Department Informatik; (1989). (Technical Report num.113).

Schäuble, P. (1993). Spider: A Multiuser Information Retrieval System for Semistructured and Dynamic Data. See reference: Korfhage et al, (eds.), (pag.318–327).

Smeaton, A.F.; Van Rijsbergen, C.J. (1983). "The Retrieval Effects of Query Expansion on a Feedback Document Retrieval System". Computer Journal. (1983; num.26, vol.3, pag.239–246). ISSN: 0010–4620.

Sparck Jones, K., (ed.). (1981). Information Retrieval Experiment. London, England: Butterworths; (1981; pag.352). ISBN: 0–408–10648–4.

Stanfill, Craig; Waltz, D.L. (1992). "Statistical Methods, Artificial Intelligence, and Information Retrieval". In: Jacobs, Paul S., (ed.). Text–Based Intelligent Systems: Current Research and Practice in Information Extraction and Retrieval. Hillsdale, Nj: Lawrence Erlbaum Associates; (1992; pag.215–225). ISBN: 0–8058–1188–5.

Strzalkowski, T. (1993). Natural Language Processing in Large–Scale Text Retrieval Tasks. See reference: Harman, (ed.), (1993a; pag.173–188).

Strzalkowski, T.; Pérez Carballo, J. (1994). Recent Developments in Natural Language Text Retrieval. See reference: Harman, (ed.), (1994; pag.123–136).

Tague, J.; Salminen, A.; Mcci–ellan, C. (1991). "Complete Formal Model for Information Retrieval Systems". In: Bookstein, A.; Chiaramella, Y.; Salton, C.; Raghavan, V.V., (eds.) SIGIR'91: Proceedings of the Association for Computing Machinery Special Interest Group on Information Retrieval (ACM/SIGIR) 14th Annual International Conference on Research and Development in Information Retrieval; (1991 October 13–16); Chicago, IL. New York, NY: ACM; (1991, pag14–20). ISBN: 0–89791–448–1.

Thompson, P. (1990). "A Combination of Expert Opinion Approach to Probabilistic Information Retrieval. Part 1: The Conceptual Model". Information

Processing and Management. (1990; num.26, vol.3, pag.371–382). "Part II. Mathematical Treatment of CEO Model 3". Information Processing and Management. (1990; num.26, vol.3, pag.383–394). ISSN: 0306–4573.

Thompson, P. (1994). Description of the PRC CEO Algorithm for Trec2. See reference: Harman, (ed.), (1994; pag.271–274).

Tong, R.M.; Applebaum, L.A. (1994). Machine Learning for Knowledge–Based Document Routing. See reference: Harman, (ed.), (1994; pag.253–264).

Turtle, H.R. (1991). Inference Networks for Document Retrieval. Amherst, MA: University of Massachusetts; (1991; pag.211). (Ph.D. dissertation). Available from: University Microfilms, Ann Arbor, MI. (UMI order no. 91–20950).

Turtle, H.R.; Croft, W.B. (1991). "Evaluation of an Inference Network–Based Retrieval Model". ACM Transactions on Information Systems. (1991, July; num. 9, vol.3,pag.187–222). ISSN: 0734–2047.

Turtle, H.R.; Croft, W.B. (1992). "A Comparison of Text Retrieval Models". Computer Journal. (1992, june; num.35, vol.3, pag.279–290). ISSN: 0010–4620.

Tzeras, K.; Hartmann, S. (1993). Automatic Indexing Based on Bayesian Inference Networks. See reference: Korfhage et al., (eds.), (pag.22–34).

Van Der Gaag, L.C. (1990). Probability–Based Models for Plausible Reasoning. Amsterdam, The Netherlands: University of Amsterdam; (1990). (Ph.D. dissertation).

Van Rijsbergen, C.J. (1977). "A Theoretical Basis for the Use of Co–Occurrence Data in Information Retrieval". Journal of Documentation. (1977, june; num.330, pag.106–119). ISSN: 0022–0418.

Van Rijsbergen, C.J. (1979). Information Retrieval. (2nd edition). London, England: Butterworths; (1979; pag.208). ISBN: 0–408–70929–4.

Van Rijsbergen, C.J.; Harper, DJ.; Porteer, M.F. (1981). "The Selection of Good Search Terms". Information Processing and Management. (1981; num.17, vol.2, pag.77–91). ISSN: 0306–4573.

Voorhees, E.M. (1993). Using WordNet to Disambiguate Word Senses for Text Retrieval. See reference: Korfhage et al., (eds.), (pag.171–180).

Voorhees, E.M.; Hou, Y.W. (1993). Vector Expansion in a Large Collection. See reference: Harman, (ed.), (1993; pag.343–351).

Voorhees, E.M.; Leacock, C.; Towell, G. (1992). "Learning Context to Disambiguate Word Senses". In: Proceedings of the 3rd Computational Learning Theory and Natural Learning Systems Conference.

Wang, P.; Soercel, D. (1993). "Beyond Topical Relevance: Document Selection Behavior of Real Users of IR Systems". In: Bonzi, Susan, ed. ASIS'93: Proceedings of the American Society for Information Science (ASIS) 56th Annual Meeting: vol. 30; (1993, October 24–28); Columbus, OH. Medford, Nj: Learned Information, Inc. for ASIS; (1993; pag.87–92). ISSN: 0044–7870; ISBN: 0–938734–78–4.

Willet, P. (1980). "Document Clustering Using an Inverted File Approach". Journal of Information Science. (1980; num.2, vol.5, pag.223–231). ISSN: 01655515.

Wittgenstein, L. (1953). Philosophical Investigations. Oxford, England: Basil Blackwell; (1953; pag.232).

Wong, S.K.M.; Cai, Y.J.; Yao, Y.Y. (1993). Computation of Term Associations by a Neural Network. See reference: Korfhage et al., (ed.), (pag.107–115).

Wong, S.K.M.; Yao, Y.Y. (1990). "Query Formulation in Linear Retrieval Models". Journal of the American Society for Information Science. (1990; num.41, vol.5, pag.334–341). ISSN: 0002–8231.

Wong, S.K.M.; Ziarko, W.; Raghavan, V.V.; Wong, P.C.N. (1987). "On Modeling of Information Retrieval Concepts in Vector Spaces". ACM Transactions on Database Systems. (1987, june; num.12, vol.2, pag.299–321). ISSN: 03625915.

Yang, Y.; Chute, C.G. (1993). An Application of Least Squares Fit Mapping to Text Information Retrieval. See reference: Korfhage et al., (eds), (pag.281–290).

Zadeh, L.A. (1994). "Fuzzy Logic, Neural Networks and Soft Computing". Communications of the ACM. (1994, March; num.37, vol.3, pag.77–84). ISSN: 0001–0782.

Zhao, S.; Kantor, P.B. (1993). "Development of an Adaptive Network Library Interface: Progress Report and System Design Issues". In: Bonzi, Susan, (ed.). ASIS '93: Proceedings of the American Society for Information Science (ASIS) 56th Annual Meeting: vol.30; (1993, October 24–28); Columbus, OH. Medford, Nj: Learned Information, Ine. for ASIS; (1993; pag.211–216). ISSN: 0044–7870; ISBN: 0–938734–784.

Zimmerman, H.J. (1991). Fuzzy Set Theory and Its Applications. (2nd edition). Boston, MA: Academic Publishers; (1991; pag.399). ISBN: 0–7923–9075–X.

Zimmerman, M. (1993). Proximity–Correlation for Document Ranking: The Para Group's Trec Experiment. See reference: Harman, (ed.), (1993a; pag. 353–364).

Paul B. Kantor (1994). "Information Retrieval Techniques". En: *Annual Review of Information Science and Technology* (vol. 29, pàg. 53-90). New Jersey: Information Today, Inc.

Metodología general de análisis y desarrollo de bases de datos documentales*

Lluís Codina†

Introducción

El objetivo de este documento es proporcionar un instrumento intelectual que ayude a enfocar problemas de información susceptibles de ser resueltos total o parcialmente con la utilización de sistemas de gestión de bases de datos. Se han publicado versiones previas, con pequeñas diferencias respecto a este documento en diversas publicaciones científicas y académicas de Documentación (ver bibliografía).

A propósito de las metodologías

En el contexto de los sistemas de información, el término *metodologías* suele generar equívocos a menudo. Es frecuente que los lectores esperen de ellas cosas que, en realidad, no pueden dar. En concreto, suelen esperar lo mismo que proporcionan, por ejemplo, los algoritmos en matemáticas, es decir, una solución segura a un problema bien planteado.

Por desgracia, en el desarrollo de sistemas de información no existe nada parecido ni a los algoritmos ni a las recetas de cocina. ¿Para qué sirve entonces una metodología en este contexto? Mi propia experiencia me dice que una metodología sirve, exactamente, para que el resultado final se deba *en lo más posible a la planificación consciente y, en lo menos posible, al azar o al método de ensayo y error. Nada más, pero nada menos*.

No parece necesario insistir mucho en que, mediante la planificación consciente, un profesional tiene derecho a esperar un grado de éxito mucho mayor que si toma las decisiones al azar o por el método del ensayo y error. Por contra, por muy correcta que sea una metodología, un lego no hará nada bueno con ella.

† Profesor titular de universidad. Universidad Pompeu Fabra. Dep. de Ciencias Políticas y Sociales. Sección Científica de Biblioteconomía y Documentación. Correo electrónico: lluis.codina@cpis.upf.es.

Por tanto, permítame el lector insistir en que la diferencia entre utilizar una metodología o no utilizarla está en qué proporción la parte final del producto puede atribuirse: a), al azar; b), al ensayo y error; c), a la planificación consciente.

De ello se desprende que siempre se desliza algo de azar en el diseño de sistemas de información, así como siempre existe la necesidad de recurrir al ensayo y error para refinar el resultado final. La cuestión clave radica en que *la parte de planificación consciente debe ser la que tenga mayor influencia en el resultado final,* tanto por razones de eficiencia como por razones de economía.

Lo contrario, que el azar y el ensayo y error tengan un gran peso, sólo puede producir sistemas desastrosos, principalmente porque los sistemas mal diseñados e ineficientes son mucho más probables, porque hay un número virtualmente infinito de hacer mal cualquier cosa, que los bien diseñados y eficientes y siempre que dejamos algo al mero azar sucede lo más probable. Esto no es más que una forma un poco más fisicalista de enunciar la conocida Ley de Murphy.

Por otro lado, es también habitual que las metodologías suenen como un mero puñado de consejos de sentido común, lo cual induce a algunos a un peligroso menosprecio hacia ellas. El problema radica en que, si bien muchos aspectos de las metodologías parecen de sentido común, su contrario también lo parece. Así pues, con una metodología, por lo menos sabemos cuáles de las muchas cosas que *parecen* razonables *son probablemente* razonables. Pongamos un ejemplo, supongamos que alguien afirma, muy serio, que el mejor procedimiento para diseñar una base de datos es escoger un buen equipo informático, después elegir un programa que sea compatible con el mismo y, a continuación, diseñar la base de datos.

No sé qué les parece a ustedes, pero yo sé de mucha gente a la cual el consejo le ha parecido tan adecuado que lo han llevado a la práctica con resultados, por supuesto, bastante lamentables. No les hubiera sucedido así si hubieran conocido uno de los aspectos más básicos del diseño de sistemas de información que aconseja comenzar siempre un proyecto estudiando siempre los aspectos lógicos y no los físicos, o comenzar por la fase de análisis y no por la de implantación, etc. Sin embargo, cuando se explican esa clase de principios en un aula, invariablemente, todo el mundo cree que está recibiendo un mensaje de sentido común.

Qué es una metodología

Por otro lado, unas meras reflexiones o unos consejos no son, a pesar de todo, una auténtica metodología. ¿Qué cosas forman parte, por tanto, de una auténtica metodología? Entendemos que, en sistemas de información, toda metodología debe contemplar, como mínimo, tres elementos o tres grupos de elementos, que aquí llamaremos *aparatos*:

a) Aparato conceptual.
b) Aparato instrumental.
c) Aparato procedimental.

El primer aparato, o grupo de elementos conceptuales, tiene la misión de proporcionar a los responsables de desarollo de sistemas de información unas bases conceptuales mínimas que faciliten su entendimiento de todo el proyecto y que faciliten, así mismo, la comunicación entre los diferentes actores involucrados en el proceso. En el aparato conceptual se definen las entidades básicas que intervienen en el proyecto y se proporcionan puntos de vista estratégicos.

El aparato instrumental es el responsable de proveer los instrumentos de análisis y de diseño, es decir, es aquella parte de la metodología que, precisamente, a veces se ha confundido, incorrectamente, con un algoritmo.

Finalmente, el aparato procedimental establece las fases y los procedimientos básicos, señalando sus objetivos, así como identifica y describe los productos que deben obtenerse de cada fase de análisis, incluido el producto final.

Así pues, y de acuerdo con lo expuesto, se describirá aquí una metodología de desarrollo de bases de datos documentales que no es un algoritmo, es decir, que no libera, mágicamente, de la obligación de tener una buena formación para poder aplicarla con éxito, pero que ayuda a reducir al mínimo posible los riesgos debidos a la improvisación.

Por otro lado, importa señalar que la metodología que se expone aquí se ha obtenido, básicamente, por la utilización de tres tradiciones científicas y académicas distintas, que este autor ha intentado fusionar en una metodología unificada y, hasta cierto punto, consistente. Se trata de las siguientes tradiciones académicas y/o tecnológicas:

a) La tradición del análisis de sistemas, proviniente de las ciencias informáticas. Unos de los autores más representativos y cualificados sería Yourdon (1993).

b) La tradición de la teoría de sistemas y de la metodología general de resolución de problemas. Concretamente, se ha utilizado teoría general de sistemas adaptada a problemas de información (Baiget, 1986; Currás, 1988) y aportaciones de la SSM (*Soft System Metodology*), una metodología elaborada principalmente, pero no únicamente, por Checkland (Checkland, 1981; Checkland y Scholes, 1990; Lewis, 1994, Underwood, 1996).

c) La tradición, naturalmente, de los métodos y procedimientos de trabajo de las ciencias de la documentación.

Una vez expuestas estas consideraciones de tipo meta-metodológicas, se exponen en las secciones siguientes los elementos de una metodología que, a su

vez, tiene sus fundamentos teóricos en un modelo conceptual sobre sistemas de información documental expuesto con más detalle en otro lugar (Codina, 1994a y Codina 1994b).

Aparato conceptual

Un primer punto de partida muy útil en el diseño de todo sistema de información y, por tanto, también en el diseño de una base de datos documental, consiste en definir el sistema de información (o la base de datos en nuestro caso) como un sistema simbólico, S1, que mantiene registros sobre otro sistema del mundo real, S2, denominado sistema objeto, y al cual representa.

De este modo, el proceso de análisis y diseño puede concebirse como el intento de obtener un modelo de aquella parte de la realidad, o sistema objeto (S2) que resulta de interés para el sistema de información (S1). Tenemos entonces el par conceptual <sistema de información, sistema objeto>, o <S1, S2>, y la relación que les une es que el primero (S1) es un modelo del segundo (S2), exactamente en el mismo sentido en que un mapa será un buen sistema de información justo en la medida en que sea un buen modelo del territorio sobre el que informa.

El segundo punto de partida consiste en considerar que, desde el punto de vista de los intereses de la Documentación, todo sistema objeto (S2) se compone de dos subsistemas, que denominamos:

a) Sistema de actividades humanas (SAH).
b) Sistema de entidades registrables (SER).

El SAH es el sistema social –es decir un sistema formado por personas y cosas– que justifica la existencia del sistema de información, porque en él desarrollan sus actividades los futuros usuarios que necesitarán que exista un sistema de información. En ocasiones, nos puede convenir considerar que, a su vez, dentro del SAH debemos distinguir entre el poseedor o propietario del sistema y los usuarios o beneficiarios del sistema (Checkland, 1982).

Por ejemplo, si pensamos en una biblioteca universitaria como en un sistema de información, entonces el sistema objeto que modela (y por tanto, el SAH) es la universidad, la cual necesita de la biblioteca (así como de otros recursos documentales) para sus actividades de creación y difusión del conocimiento. ¿En qué sentido la biblioteca es un modelo de la universidad? En el sentido en que los temas y disciplinas científicas que cubre la biblioteca, la clase de documentos que adquiere, los procedimientos de trabajo, los servicios que presta, etc., son un reflejo de las características de la universidad.

Si consideramos la base de datos de temas de actualidad de una empresa periodística, la propia empresa periodística es el SAH del sistema, en este caso, el po-

seedor del sistema, y el público interesado en la consulta de esa base de datos formará parte también del SAH, en este caso, como beneficiarios del sistema.

Como el entorno de un sistema siempre influye en él de alguna forma, los diseñadores de la base de datos, aunque deberán concentrarse en las características de la información de actualidad a tratar en la base de datos, también deberán conocer las características de su entorno, esto es, de la empresa. Los ejemplos podrían multiplicarse fácilmente. Por ejemplo, si se trata de diseñar la base de datos de un museo, el SAH será el museo en cuestión, etc.

Por su parte, el sistema de conocimiento o de entidades registrables (SER) está formado por los documentos o las entidades sobre los cuales el sistema de información debe mantener algún tipo de registros.

En el caso de la base de datos de una empresa periodística, por seguir con otro de los ejemplos mencionados, el SER consistirá, según decisión de los poseedores del sistema, o bien en las informaciones de actualidad que publica esa empresa o bien en alguna otra entidad. Por ejemplo, una de las agencias de noticias más importantes de nuestro país, la Agencia EFE, produce bases de datos no solamente sobre noticias de actualidad sino sobre biografías, la Unión Europea, etc.

Con los dos principios fundamentales anteriores se dispone ya de un mínimo aparato conceptual que permite iniciar la discusión de los otros elementos de la metodología. Se observará que algunas herramientas del aparato instrumental, tal como en el modelo entidad-relación (que se explica más adelante) incluyen también aspectos conceptuales. En realidad, es en buena parte arbitrario decidir qué elementos pertenecen al aparato conceptual y qué elementos pertenecen al procedural o al instrumental. Aquí se he hecho una elección concreta, pero probablemente son posibles otras interpretaciones.

Aparato instrumental

El aparato instrumental de una metodología proporciona los instrumentos de análisis que puede utilizar el analista. En concreto, tres son los instrumentos principales que se pueden emplear: el modelo entidad-relación, desarrollado originalmente por Chen (1976), el diccionario de datos y la norma ISBD.

Modelo Entidad-Relación

El modelo entidad-relación (o modelo E-R) ayuda a detectar sin ambigüedades las entidades que formarán parte de la base de datos, es decir, los objetos que forman parte del sistema de conocimiento. Estas entidades son las que tendrán que ser descritas en la base de datos e importa, por tanto, idenfiticarlas con la mayor precisión posible.

Además, el modelo E-R proporciona una terminología adecuada para las primeras fases de diseño y un método para discriminar entre entidad y atributo de entidad, cosa que a veces puede resultar trivial pero que en otras ocasiones no lo es en absoluto. El modelo E-R utiliza los siguientes conceptos:

- Entidad
- Atributo
- Relación

Según este modelo, si las bases de datos representan a cosas u objetos del mundo real, tales cosas deben ser identificables y deben tener algunas propiedades. A las cosas sobre las cuales almacena información una base de datos se las denomina entidades, y pueden ser cosas materiales (libros, personas, etc.) o conceptuales (ideas, teorías científicas, etc.).

La única restricción aplicable es que las entidades que han de estar representadas en una base de datos deben ser identificables y, por tanto, debe ser posible señalar a una cualquiera de ellas sin ambigüedad.

Los atributos, por su parte, son las propiedades relevantes que caracterizan a una entidad. En este sentido, el término relevantes significa lo siguiente: relevantes para el problema de información que se está considerando. Teniendo en cuenta que, en principio, los atributos de una entidad son virtualmente ilimitados, será labor del documentalista seleccionar en cada caso cuáles son los que se consideran más relevantes.

El modelo distingue entre tipo de entidad y ocurrencia de entidad. Un tipo de entidad define un conjunto de entidades constituidas por datos del mismo tipo, mientras que una ocurrencia de entidad es una entidad determinada y concreta. Cuando se diseña una base de datos el objetivo del documentalista debe consistir en definir un tipo de entidad, que obtiene estudiando ocurrencias concretas de entidades.

Un registro es una representación de una entidad en la base de datos y, por lo tanto, cada registro describe a una entidad. Por ejemplo, en una base de datos bibliográfica cada documento se describe en un registro.

Por tanto, si los registros describen entidades del mundo real, los campos corresponden a los atributos de la entidad. De este modo, si un tipo de entidad posee los atributos A, B, C, el modelo de registro debe poseer los campos A, B, C.

En este punto, necesitamos diferenciar entre los siguientes conceptos:

1. *Etiqueta* del campo
2. *Valor* del campo
3. *Dominio* del campo

La etiqueta es el nombre del campo, es decir, una constante que identifica una zona del registro. El valor se refiere al contenido concreto de un campo concreto y puede ser distinto para cada campo de cada registro. El dominio, por su parte, es el conjunto del cual puede tomar sus valores un campo. Por ejemplo, el dominio del campo *Año de publicación*, es el conjunto formado por los años de publicación de documentos.

Figura 1: Un registro representando a un libro

Título	Internet: manual de referencia
Autor	Harley Hahn; Rick Stout
Fuente	Madrid: Osborne Mc Graw-Hill, 1994
Año	1994
Páginas	692
ISBN	84-481-1882-0
Descriptores	Internet, Redes telemáticas, Bases de datos, Correo electrónico, Telnet, Usenet, FTP, Wais, World Wide Web

Veámoslo con otro ejemplo. De acuerdo con el registro de la figura 1, el segundo campo o zona de información se puede analizar así:

Nombre del campo:	Autor
Valor del campo:	Harley Hahn; Rick Stout
Dominio del campo:	El conjunto de los nombres de responsables intelectuales de los documentos.

Generalizaciones y abstracciones

Al igual que distinguimos ente tipo y ocurrencia de entidad, debemos diferenciar también entre modelo de registro y ocurrencia de registro. Un tipo de entidad se forma por abstracción y/o generalización. Abstracción o generalización significa que se ignoran ciertos aspectos distintos de diversas ocurrencias de entidad y se forma con todas ellas un tipo unitario, o que se generalizan a todas las entidades ciertos rasgos que presentan regularmente ciertas entidades.

Por ejemplo, supongamos que aplicando el modelo E-R a un problema de información (por ejemplo, una base de datos para automatizar el archivo de un medio de comunicación), nos muestra como primer resultado los siguientes tipos de entidades:

1. Artículos de revistas
2. Artículos de prensa diaria
3. Capítulos de libros
4. Libros

5. Informes
6. Fotografías de personajes
7. Fotografías de sucesos
8. Fotografías de estudio
9. Infografías

Una simple generalización reduce los nueve tipos de entidades a dos, puesto que las entidades 1 a 5, pueden reducirse, por abstracción, a una sola: *Documentos escritos*, y los tipos de entidades 5 a 9 al tipo de entidad: *Documentos gráficos*. La entidad *Documentos escritos* deberá tener un atributo denominado *Tipo de documento*, que permitirá describir qué clase de documento es: artículo, libro, etc. Por su parte, la entidad *Documentos gráficos*, deberá tener también un campo denominado *Tipo de documento*, que permitirá indicar si es una fotografía de personas, fotografía de paisajes, o si es una infografía, etc.

Relaciones

Las entidades del mundo real pueden tener relaciones entre ellas y, mientras las entidades suelen nombrarse mediante sustantivos, las relaciones se nombran mediante verbos. Por ejemplo, consideremos el caso de una base de datos sobre teatro español. Un análisis intuitivo nos revelaría la existencia de dos entidades relevantes para el sistema: *[obras de teatro]* y *[autores teatrales]*, y veríamos que entre ambas entidades existe la relación <escriben>, que significa más explícitamente que *[autores teatrales]* <escriben> *[obras de teatro]*.

Un aspecto importante de la relación es su grado, el cual indica el número de elementos que pueden participar en cada uno de los extremos de la relación, en este caso *[autores]* y *[obras de teatro]*. Este grado puede ser de uno a uno (1:1), de uno a muchos (1:N) y de muchos a muchos (N:M). Una manera típica de representar estas relaciones y su grado es utilizando diagramas y expresiones textuales. n estos diagramas, las entidades se representan como rectángulos y las relaciones como rombos. A su vez, las entidades se identifican con sustantivos y las relaciones con verbos.

Así, por ejemplo, la relación que existe entre el número de ISBN y un libro es una relación de 1:1 (que se lee "relación de uno a uno") porque un número de ISBN se asigna a un solo libro, y cada libro tiene un solo número de ISBN.

En cambio, la relación entre profesores y universidades es de 1:N, ("de uno a muchos") porque cada profesor pertenece a una sola universidad, y una universidad tiene muchos profesores.

Finalmente, una relación de N:M ("de muchos a muchos") sería la que existe entre autores de teatro y obras de teatro, porque un autor puede escribir diversas obras de teatro, y una obra de teatro puede estar escrita por varios autores

y justamente ese es el significado de las letras N y M que hemos puesto en el diagrama anterior.

Además, la participación de la entidad puede o no ser obligatoria, lo cual significa que una entidad obligatoria interviene siempre en la relación. Por ejemplo, en la relación entre ISBN y libros, la participación de la entidad *[libros]* es obligatoria, porque siempre que hay un número de ISBN hay un libro, en cambio lo contrario no es cierto, porque hay libros que no tienen número de ISBN.

Esta última parte del análisis entidad-relación (grado y participación) es muy importante en el diseño de bases de datos de gestión que suelen utilizar tecnología relacional, porque ayuda a modelar los datos de la empresa y a representarlos en tablas normalizadas.

En cambio, en sistemas documentales no es tan importante porque éstos no suelen utilizar tecnología relacional, ni necesitan modelar relaciones complejas entre entidades, como las que se dan en los sistemas de gestión administrativos.

En muchos sistemas documentales, las entidades, de hecho, no mantienen relaciones entre ellas que deban ser reflejadas en el modelo E-R.

Por ejemplo, en una típica bases de datos documental sobre literatura científica y técnica no suele existir ninguna relación entre las entidades representadas (típicamente artículos de revista y monografías) que deba ser tenida en cuenta en el modelo E-R.

En tales situaciones, el modelo E-R aporta una importante claridad conceptual y proporciona una terminología común a todos los miembros que participan en el diseño. Sin embargo, el propósito de las herramientas de diseño no es tanto proporcionar soluciones para situaciones que son bien conocidas, sino para las situaciones no conocidas o menos típicas y, en este sentido, el modelo E-R puede resultar de ayuda también para determinar otros elementos del diseño.

Por ejemplo, y volviendo al caso anterior, donde se nos pide diseñar una base de datos sobre teatro español. Supongamos que tenemos dudas sobre el siguiente aspecto: no sabemos si considerar que el autor (y todos sus datos biográficos) son atributos de la obra de teatro, o bien si considerar que autor y obras de teatro son entidades distintas, como hemos dado por supuesto en el diagrama.

Si adoptáramos el primer punto de vista, tendríamos que diseñar un único modelo de registro, donde los atributos del autor serían otros tantos campos, junto con los atributos de la obra de teatro. En cambio, si adoptamos el segundo punto de vista, necesitaremos diseñar dos modelos de registro, uno para obras de teatro y otro para autores. Puede ser que la simple intuición no indi-

que cuál es el camino correcto en éste o en otros casos parecidos, pero si queremos estar seguros de no equivocarnos en nuestra decisión, siempre podemos aplicar el siguiente procedimiento:

1. En caso de duda, tratar las cosas como entidades distintas.
2. Determinar la relación entre entidades.
3. Determinar su grado.
4. Si la relación es de grado 1:1, entonces se trata de una sola entidad y un solo modelo de registro es suficiente para representarla. Por ejemplo, el número de ISBN es, de hecho, un atributo de la entidad libro, y para representarla es suficiente un solo registro, con un atributo que incluya el número de ISBN.
5. Si la relación es de grado N:1, o N:M, se trata de dos entidades y, por lo tanto, necesitamos dos modelos de registro, uno para cada entidad, y cada uno de ellos debe contar con un campo con un dominio común.

En nuestro ejemplo, la aplicación de esa regla nos indicaría que la decisión acertada consiste en utilizar dos modelos de registro: uno para representar obras de teatro y otro para representar autores teatrales. El campo con un dominio común podría ser el campo Autor, que debería figurar en ambos registros.

¿Qué sucedería si no procedíeramos como indica esta norma? En tal caso, la carga de datos sería poco eficiente, porque para autores muy prolíficos tendríamos que entrar los mismos datos tantas veces como obras de teatro hubiera escrito.

En general, si un autor ha escito *n* obras de teatro, tendríamos que repetir sus datos *n* veces. Además, la redundancia, como es sabido, genera inmediatamente inconsistencias, y tendríamos enseguida, por ejemplo, diversas fechas de nacimiento para un mismo autor. Es evidente que si no detectamos ese error de diseño a tiempo, no tardará en hacerse evidente en algún momento de la fase de carga de datos, pero no debería ser menos evidente que si podemos evitar el error en la fase de diseño estaremos trabajando con mucha mejor calidad (ahora que está tan de moda este tema) que si necesitamos llegar a la implantación para detectar los errores, tal vez después de meses de trabajo que, de golpe, se revelarán inútiles.

Una advertencia final sobre el modelo E-R. Primero, cuando se utiliza para diseñar bases de datos relacionales, las reglas para tomar decisiones son más complejas, porque la descomposición de datos a la que obliga el modelo relacional implica la necesidad de representar no sólo las entidades, sino también las relaciones entre entidades mediante una tabla más. Los interesados en esos aspectos de diseño pueden consultar Jackson (1990).

En general, la tecnología relacional debería ser necesaria cuando se trata sobre todo de modelar actividades (relaciones) y los datos relativos a cada entidad

son relativamente simples o están muy estructurados. La mayoría de las actividades de gestión administrativa de una empresa son de esa clase y por eso utilizan sistemas relacionales. En cambio, deberíamos utilizar sistemas documentales en la situación simétricamente opuesta a la anterior, es decir, cuando se trata de modelar depósitos de conocimiento más que actividades, y los datos no son en realidad datos, sino información no estructurada o extremadamente compleja. La mayoría de las actividades de la Documentación responden a ese perfil y por eso utilizan sistemas documentales.

El diccionario de datos

El diccionario de datos es una herramienta que ayuda al diseñador de una base de datos a garantizar la calidad, la fiabilidad, la consistencia y la coherencia de la información introducida en la base de datos, de tal manera que el diccionario de datos marcará decisivamente el rendimiento y la calidad global del sistema de información.

Consiste en la lista detallada de cada uno de los campos que forman los distintos modelos de registro de la base de datos. A cada campo de cada modelo de registro se le aplica una parrilla de análisis que contempla, como mínimo, los siguientes aspectos:

1. Dominio
2. Tipo
3. Indización
4. Tratamiento documental
5. Lengua
6. Otros controles de validación u observaciones
7. Ejemplos válidos

Por ejemplo, supongamos, a efectos de esta explicación, una base de datos documental imaginaria sobre noticias de actualidad con sólo tres campos: <Título>, <Descriptores> y <Fecha de publicación>. El diccionario de datos tendría entonces esta forma:

Etiqueta: Título

Dominio:
Título del documento. El título se transcribe de la siguiente forma: *Título: antetítulo: subtítulo*.

Tipo:
Alfanumérico

Indización:
Indizado

Tratamiento documental:
Lenguaje libre

Lengua:
Lengua del documento

Controles de validación:
No puede quedar vacío. Si por alguna razón, el documento careciera de título, el documentalista asignará un título descriptivo.

Etiqueta: Descriptores

Dominio:
Palabras clave normalizadas que expresan los conceptos principales contenidos en el documento, según el siguiente principio general: si el artículo contiene *n* conceptos relevantes se asignan n descriptores, procurando no asignar més de 20 descriptores por documento.

Tipo:
Alfanumérico

Indización:
Indizado

Tratamiento documental:
Lenguaje controlado

Lengua:
Del centro de documentación

Controles de validación:
No puede quedar vacío y sólo admite valores extraídos de una lista de términos autorizados.

Etiqueta: FPublicación

Dominio:
La fecha de publicación de la noticia, indicada con el siguiente formato: DD/MM/AAAA.

Tipo:
Fecha

Indización:
Indizado

Tratamiento documental:
No procede

Lengua:
No procede

Controles de validación:
No admite valores fuera de rango.

Estudiando el ejemplo de diccionario de datos anterior, formado únicamente por tres campos, podemos observar cuatro aspectos importantes para el diseño de bases de datos lo siguiente:

1. Que el *Dominio*, en el contexto del diccionario de datos, se refiere al conjunto del que un campo puede obtener sus valores.

2. Que el *Tipo* se refiere, en cambio, al tipo de dato que admite el campo. Los tipos de datos suelen ser: numérico, alfanumérico, fechas y lógico.

Recordemos que un tipo de dato (data type) define un conjunto de operaciones válidas y un rango de valores aceptable. Por ejemplo, el tipo de datos "alfanumérico" define operaciones de comparación de cadenas de caracteres, entre otras, así como cualquier letra de la a a la z y cualquier número del 0 al 9, así como cualquier combinación de esos caracteres en palabras, frases, párrafos, etc. En cambio, no admite operaciones aritméticas, aunque admita números. Por el contrario, un tipo de dato "numérico" admite sólo números así como cualquier operación aritmética, etc.

Por su parte, un campo de fechas sólo admite fechas en un formato establecido y permite búsquedas por rangos de fechas o por valores superiores o inferiores a una fecha dada. Un campo lógico sólo admite uno de dos valores: Sí o No; Verdadero o Falso.

3. Que el *Tratamiento documental* establece si se debe utilizar algún lenguaje documental para entrar los valores del campo, como así sucede en el campo Descriptores, donde el diccionario de datos establece que ese campo sólo admite palabras clave autorizadas extraídas de un thesauruso de una lista de autoridades.

4. Que la *Lengua* puede ser, o bien la lengua del documento, o bien la del centro de documentación. Eso significa, en el caso de un documento escrito en inglés, que el título estaría en inglés, pero los descriptores en castellano, siempre de acuerdo con el diccionario de datos precedente.

La descripción funcional, por su parte, debe incluir los siguientes elementos:

1. Qué clase de información se tratará y cómo entrará la información en el sistema.

2. Qué procesos documentales se llevarán a cabo.

3. Qué servicios y productos generará el sistema, y/o a qué aplicaciones podrá dar soporte.

El primer punto debe describir en qué consisten las entradas del sistema. El punto dos debe proporcionar una idea sobre qué procesos de tratamiento documental automatiza la base de datos, y el punto siguiente debe explicar en qué consisten las salidas del sistema.

La ISBD y los modelos canónicos

Por otro lado, no deberíamos olvidar que, en Documentación, la experiencia previa ha dejado bien sentados cuáles son los atributos de algunas entidades e incluso cuál es la forma más conveniente de representarlos. Podemos hablar entonces de situaciones canónicas que han generado un modelo. La mejor herramienta de análisis y de diseño, en tal caso, consiste precisamente en aplicar ese modelo bien conocido y testeado.

Por ejemplo, los atributos estructurales de cualquier clase de documento pueden ser adecuadamente modelados siguiendo la norma internacional ISBD. Recordemos que esa norma internacional representa un gran esfuerzo de abstracción para proporcionar un marco general de descripción, válido para cualquier clase de documento, desde una partitura musical, hasta una filmación audio-visual, pasando por un archivo de ordenador, un fonograma o un artículo de revista, de manera que las ISBD constituyen una herramienta de diseño de primera magnitud para cualquier problema documental donde debamos representar documentos.

Sobre el uso de las ISBD, cabe advertir que algunos centros de documentación se han sentido intimidados ante la aparente complejidad de la norma y la supuesta obligación de adoptarla como un todo, incluyendo la prolija puntuación que prescribe y, en tal sentido, se ha argumentado que utilizar la norma ISBD solo tiene sentido en el contexto de la lectura pública.

Entiendo que tal postura es un error: primero, porque siempre podemos utilizar la estructura de las ISBD como una orientación en el análisis de los documentos convencionales así como una fuente de inspiración para situaciones más exóticas, independientemente de que incorporemos o no la norma en toda su complejidad, es decir, incluyendo decir todos los niveles de descripción y todas las prescripciones de puntuación, máxime cuando el hecho de separar zonas mediante campos libera de la necesidad de utilizar la puntuación prescrita.

Además, en caso necesario, el programa documental debería permitir presentar la salida de los datos en formato ISBD (o en cualquier otro formato), desde el momento en que la estructura repetitiva de los registros permite incorporar

instrucciones del tipo: "el valor del campo *Título* se transcribe seguido por un punto, espacio y una raya", etc.

Aparato procedimental

El principio general de diseño de sistemas de información indica que todo proyecto comienza siempre por un diseño lógico y que, una vez aprobado éste, se procede al diseño físico o implantación, en un proceso que es tan circular como lineal, ya que la fase de diseño, por ejemplo, puede obligar a repensar aspectos de la fase de análisis.

El aspecto importante aquí es que la metodología nos dice claramente que el proceso de creación de una base de datos debe ir siempre desde los aspectos lógicos hacia los aspectos físicos, y no al revés, como suele suceder, ya que, en la práctica, existen muchas formas de violar ese principio general a causa de malos hábitos de trabajo.

Otra manera de enfocar incorrectamente este proceso consiste en querer abordar directamente el diseño del sistema de información e, incluso en querer visualizarlo por completo en nuestra mente, sin antes saber nada del sistema objeto.

El resultado, claro está, será una visión caótica. Todas las interrogantes se agolparán en nuestra mente y seremos incapaces de despejar una sola de ellas.

Lo correcto en ambos casos es comenzar a diseñar los aspectos lógicos (nivel conceptual) e ignorando de momento los aspectos físicos; así como comenzar por analizar el sistema objeto y sólo después de conocerlo bien, podemos iniciar el diseño del sistema de información.

Así pues, el proceso de diseño de un sistema de información debe ajustarse siempre al siguiente ciclo de vida que, por otro lado, es universal para todo sistema de información:

1. Análisis
2. Diseño
3. Implantación

Otra forma de enfocar el ciclo de vida de un proyecto de desarrollo es indicar que la dirección del diseño debe proceder de lo conocido a lo desconocido, y no al revés, como sucede cuando se desea visualizar el sistema de información antes de conocer el sistema de actividades humanas y el sistema de conocimiento.

Finalmente, y por la misma razón, la dirección del diseño debe ir de lo general a lo específico y de los aspectos lógicos a los aspectos físicos, y nunca al revés,

es decir, nunca se debe empezar a discutir o a considerar cuestiones concretas (¿cómo se imprimirá la información?) o físicas (¿qué tamaño tendrán las estanterías de los documentos?) antes de plantear las cuestiones generales (¿cuál es el propósito de la base de datos?) o lógicas (¿qué entidades formarán parte de la base de datos?). El siguiente cuadro sinóptico sintetiza estas ideas:

Figura 3: Cuadro sinóptico de la dirección del diseño en el ciclo de vida de un sistema de información

- De lo conocido a lo desconocido.
- De los aspectos lógicos a los aspectos físicos.
- De lo general a lo concreto.

En cuanto, al ciclo de vida, cada una de las tres fases enunciadas antes (Análisis, Diseño, Implantación) puede dividirse en cuantas subfases sean necesarias según el proyecto concreto y la clase de sistema que se está diseñando.

En el caso de una base de datos documental, las dos primeras fases se pueden subdividir en otras dos subfases (a y b). Las fases de implantación pueden subdividirse en cuatro subfases (a, b, c, d, e). Nuevamente debe indicarse que tales divisiones tienen siempre algo de arbitrario. Aquí se hace una propuesta concreta, pero pueden ser válidas otras formas de dividir el ciclo de vida. En concreto, en esta metodología se propone la división de fases del cuadro sinóptico de la figura 4:

Figura 4: Cuadro sinóptico del ciclo de vida de una base de datos documental

1. *Análisis*
1a. Análisis del sistema de actividades humanas
1b. Análisis del sistema de conocimiento

2. *Diseño*
2a. Diseño del modelo conceptual
2b. Determinación de los procedimientos de tratamiento documental (descripción, análisis e indexación documental, etc.) si es el caso.

3. *Implantación*
3a. Elaboración del presupuesto y del calendario de implantación, en su caso.
3b. Selección del soporte informático (*software y hardware*) de acuerdo con los requerimientos expresados en el modelo conceptual de la base de datos producido en la fase 2a y de acuerdo con los requerimientos expresados en 2b.
3c. Instalación, pruebas de rendimiento y re-elaboración, en su caso, de los puntos previos de este ciclo de vida.
3d. Elaboración del libro de estilo de la base de datos.
3e. Carga de datos, formación de usuarios y promoción del producto.

Aunque expresado en fases y enumeradas secuencialmente el proceso parece estrictamente lineal, en realidad, el proceso de diseño también tiene mucho de circular, porque aunque siempre se empieza por la fase de análisis y se sigue con la de diseño, llegados a la fase 2b, por ejemplo, es posible que el diseñador desee considerar de nuevo algunos aspectos de 2a, o que necesite aclarar mejor algunas cuestiones de 1b, etc.

En este sentido, debe hacerse notar que la metodología no excluye totalmente el procedimiento del ensayo y error, como ya se advirtió, sino que lo integra como un modo natural de refinar el producto.

En particular, es prácticamente imposible producir un modelo conceptual correcto en el primer intento, y la experiencia indica que lo más probable es que el modelo elaborado en los puntos 2a y 2b haya que rehacerlo más de una vez, por lo menos en alguno de sus aspectos, principalmente a la vista de las primeras pruebas de rendimiento (3c).

Naturalmente, tiene que llegar un momento en el cual el diseñador dé por finalizado el proceso, pero la cuestión de cuántas veces conviene iterarlo antes de darlo por bueno, no puede establecerse *a priori*, sino que, antes bien, es una cuestión sensible al contexto y que debe decidir el diseñador en cada caso.

En todo caso, es importante que se llegue a la fase de implantación con un modelo lo más sólido posible porque a partir de tal fase ya no resulta tan fácil reconsiderar el proyecto, por lo menos no sin pagar algún precio, de manera que el punto 3c debería considerarse el punto de despegue, de alguna manera, el punto de no retorno del proyecto.

La fase de implantación puede llevarla a cabo un equipo distinto del que hizo el diseño. De hecho, en algunas empresas, sobre todo en empresas medianas y grandes, puede ocurrir que la fase de implantación corra a cargo del departamento de informática, aunque el análisis y el diseño lo haya hecho el de documentación. En empresas pequeñas, lo más habitual es que todo el proceso lo ejecute un mismo equipo o una misma persona.

Cada una de las fases precedentes (Análisis, Diseño, Implantación) tiene unos objetivos, debe producir unos resultados concretos y utilizar unas herramientas determinadas.

La fase de análisis

El objetivo de esta fase es conocer bien aquella parte del mundo real, llamada sistema objeto, que justifica y requiere la creación del sistema de información, de una base de datos en este caso.

Como ya vimos anteriormente, a efectos de análisis, el sistema objeto se considera dividido en:

– Un sistema de actividades humanas (SAH)
– Un sistema de conocimientos (SCO).

Por lo tanto, y dado que las características del sistema de actividades humanas (SAH) determinarán las características de la base de datos, deberá conocerse lo mejor posible antes de iniciar cualquier actividad de diseño.

El resultado que debe producir esta fase de análisis es una descripción textual que puede incluir gráficos de ser necesario, sobre el SAH, que suele denominarse *Informe de funciones* o *Informe de oportunidad*[‡], y que debe incluir, como mínimo, los siguientes aspectos:

1. Propósito y objetivos del SAH
2. Actores principales del SAH
3. Actividades más relevantes del SAH
4. Entorno del SAH
5. Características de las entidades registrables (SER)

La herramienta principal aquí es la realización de entrevistas con representantes del SAH y el análisis de cualquier documentación, del y sobre el SAH, que pueda aportar una comprensión global del sistema. Entre tales documentos podemos citar organigramas, documentos fundacionales, memorias, etc.

Aunque el *Informe de funciones* consiste, básicamente en una descripción textual, puede incluir, si el documentalista lo considera necesario, diagramas o gráficos que faciliten su comprensión.

El *Informe de funciones* no debe ser muy extenso, sino, que tal como indica su nombre, debe consistir únicamente en una descripción que recoja los aspectos esenciales de la naturaleza y de las actividades del SAH. Además, como una base de datos documental no persigue el modelado de esas actividades, probablemente cinco o seis párrafos deberían ser suficientes para aportar el conocimiento necesario para los objetivos perseguidos.

Este modelo podrá formar parte del producto final, pero no es necesario que sea así, ya que, principalmente su misión es asegurarse de que el responsable del proyecto y otros actores que intervengan en él tienen una adecuada concepción de la naturaleza del SAH.

‡ En otras versiones de esta metodología, a este informe se le denominaba "Modelo esencial". Como es fácil suponer, el nombre es lo de menos. Ahora se opta por el nombre de "Informe de funciones" o "Informe de oportunidad" para utilizar expresiones más estandarizadas.

Por su parte, el propósito de la fase del análisis del sistema de conocimiento consiste en conocer el componente clave en este caso del sistema objeto, a saber, los documentos o las cosas sobre las cuales la base de datos deberá recoger información.

El resultado de esta fase debe consistir en la identificación clara y sin ambigüedades de los documentos o las cosas (entidades) sobre las cuales la base de datos deberá mantener información, así como debe poner de manifiesto las propiedades más relevantes de esas entidades.

La herramienta más adecuada para esta fase, es el modelo entidad-relación (modelo E-R), un modelo bastante intuitivo que, sin embargo, resulta de gran utilidad para enfocar este tipo de análisis. Este modelo se explicará en el apartado dedicado a las herramientas.

La fase de diseño

El propósito de la fase de diseño es obtener un *Modelo Conceptual* de la base de datos y una *Propuesta de tratamiento documental*. El primero contiene los elementos necesarios para orientar el proceso de implantación. El segundo establece criterios y orientaciones sobre el proceso de descripción y de representación del contenido semántico de los documentos o entidades de los que tratará la base de datos.

Los dos modelos mencionados son el resultado de la fase de diseño y deben ser aprobados por quien encargó el proyecto, antes de que puedan servir como guías de implantación. Por tanto, el modelo conceptual no sólo debe ser acertado, sino que, además debe parecerlo.

El *Modelo Conceptual* debe contener, por lo menos, los siguientes elementos:

1. La parte esencial del informe de funciones, mencionando el objetivo y propósitos de la base de datos e identificando a los usuarios del sistema.
2. Una definición del dominio de la base de datos.
3. Una identificación de las entidades representadas en la base de datos.
4. El diccionario de datos
5. La política de control terminológico o tratamiento documetal.

El *dominio* de la base de datos es el conjunto de los temas o entidades sobre los que mantiene información la base de datos. Como todo dominio, puede definirse por extensión o por comprensión. Por tanto, puede ser tan breve como el nombre de una o más disciplinas científicas, por ejemplo, el dominio de la base de datos LISA Plus son las *Ciencias de la Documentación*. O puede consistir en una frase, por ejemplo, el dominio de la base de datos TESEO se enuncia diciendo que está formado por *las tesis doctorales publicadas por universidades españolas*.

Las herramientas para producir el documento anterior son, entre otras, las siguientes:

1. El informe de funciones o de oportunidad.
2. El modelo entidad-relación.
3. El diccionario de datos.

La definición raíz expresa qué es la base de datos o, si se quiere, expresa la clase de problemas que podrá solucionar y a qué categoría de usuarios dará servicio. Esta descripción debe mencionar a los usuarios de la base de datos. No debe ser más larga de tres o cuatro párrafos. La información necesaria para construir la definición raíz se obtuvo del *Modelo esencial*, que forma parte de la fase de análisis y que vimos en su momento.

Un ejemplo sencillo podría ser la definición raíz de la base de datos documental de un medio de comunicación que podría adoptar la siguiente forma: "El propósito de esta base de datos es satisfacer las necesidades de información retrospectiva de los redactores del diario, permitiéndoles recuperar selectivamente cualquier información publicada anteriormente por el diario".

Al igual que en la identificación del dominio de la base de datos, elaborar la definición raíz puede ser una tarea fácil e intuitiva, resultado de un mero análisis técnico, o bien puede ser producto de una refinada decisión política. Lo que es importante es que, sea cual sea el proceso de decisión, ésta quede documentada y expresa y formalmente detallada por escrito.

La fase de implantación

Una vez aprobado el modelo conceptual de la base de datos, puede procederse a su implantación, la cual suele seguir el siguiente proceso:

1. Se selecciona el sistema informático (software + hardware) que pueda satisfacer mejor los requerimientos del modelo conceptual y del modelo de normativa de indización. De ser necesario, se examinarán varios programas candidatos hasta que exista una razonable certeza de que el programa elegido se ajusta bien a los requerimientos del modelo conceptual. Se realiza la primera instalación y se nombra a un administrador de la base de datos que, a partir de ahora, será el máximo responsable de ella.
2. Se realizan pruebas con una colección-test de documentos o de entidades a ser representadas para comprobar la consistencia de los modelos y esquemas de registros.
3. Se realizan los cambios o ajustes necesarios, hasta obtener el modelo final.
4. Definición de una política de mantenimiento y explotación.
5. Se edita la versión 1 del Libro de estilo de la base de datos, que incluye:
   a) La versión definitiva del modelo conceptual.
   b) La normativa de tratamiento documental, en su caso.

6. Se procede a la formación del personal técnico y de los usuarios finales.
7. Acciones de promoción, en su caso.

Conclusiones

El valor de esta metodología radica, como ya se dijo al principio, en que ayuda a que el producto final sea más resultado del diseño consciente que de las fuerzas ciegas del azar y/o del ensayo y error, pero, particularmente entendemos que su utilidad aumenta conforme se aplica a situaciones poco canónicas o a situaciones atípicas, como las que el entorno cambiante de nuestra profesión introduce en cada momento y, al parecer, tal como el nuevo horizonte de las autopistas de la información y de un futuro mundo digital parece prometer.

Esperamos que, entonces, la aplicación de esta clase de metodologías sirva para que los profesionales de nuestro campo puedan demostrar los beneficios de una adecuada formación académica, del trabajo bien realizado y de la planificación, porque en nuestro campo de actividades también es rigurosamente cierto que el éxito se debe invariablemente a "un diez por ciento de inspiración y un noventa por ciento de transpiración".

Lluís Codina (1999). *Metodología general de análisis y desarrollo de bases de datos documentales* (document reprografiat). Barcelona.

Estudio de caso. Proyecto: sistema de información sobre recursos digitales en Internet de la editorial ACME

Lluís Codina

1. Escenario del proyecto

La editorial Acme, una editorial especializada en publicaciones sobre comunicación audiovisual, multimedia y temas culturales, proyecta crear un sitio web que actúe como portal para los ámbitos temáticos señalados.

Una de las secciones que esperan que proporcione un mayor interés a su portal es un servicio de selección, descripción y evaluación de recursos digitales que sea fácil de consultar por el público y que permita a ese público hacer búsquedas selectivas por múltiples criterios: título, temas, idioma, etc.; así como búsquedas en las que se combinen dos o más de esos criterios.

Además, ese servicio deberá dar soporte a las diversas redacciones de la editorial que publica revistas sobre cine, cultura y pensamiento, humanidades, etc. Finalmente, y si el servicio es eficiente, la base de datos será el núcleo de una serie de guías sobre temas culturales en Internet que la editorial piensa ir publicando periódicamente.

Después de un proceso de análisis se ha llegado a la conclusión que se necesitará una base de datos documental para dar soporte al servicio, dada la diversidad de formas de explotación que se prevén. En concreto, la base de datos servirá para que un equipo de editores, con formación en documentación, pueda crear y mantener el sistema de información sobre recursos digitales.

Además, a través de un servidor web y de un programa que actúe como pasarela, la misma base de datos podrá ser consultada desde Internet utilizando un navegador web estándar, como Nestscape o Explorer.

2. Proyectos comparables

Buscopio, de la Editorial Prisa <www.buscopio.com>.

Guíame, de Esade <www.guiame.es>.

Sosig, www.sosig.ac.uk.

ADAM <adam.ac.uk>, de la administración inglesa.

Cercador, de Gran Enciclopedia Catalana <www.cercador.com>

EB, de la Enciclopedia Británica <www.eb.com>.

3. Definición funcional

3.1. Objetivos

La base de datos de recursos digitales Acme tiene el objetivo de facilitar las labores de creación, mantenimiento y explotación del servicio de información de la editorial Acme sobre recursos digitales en Internet, el cual se ofrece como una de las partes principales de su lugar web.

El propósito estratégico del servicio es, en primer lugar, fidelizar a sus lectores e incrementar su cuota de mercado en los servicios ofrecidos a través de Internet. En segundo lugar, la base de datos debe proporcionar soporte a las actividades de las diversas redacciones de Acme.

3.2. Público

El público destinatario de este servicio de información es:

1. Los lectores de sus publicaciones y, en general, el público interesado en los temas propios de las actividades de Acme.
2. Los redactores, jefes de redacción y directores de las publicaciones de Acme.

Para ello, un equipo de especialistas en comunicación social, periodismo y documentación examinan periódicamente lugares web, los evalúan críticamente y, si su calidad supera un umbral mínimo, esos lugares web quedan descritos y registrados como recursos digitales en la base de datos.

Al mismo tiempo, la base de datos de recursos digitales es el sistema de información que proporciona la posibilidad de realizar las consultas y otras formas de explotación, como la creación de índices de diversos tipos o la publicación de guías, etc.

3.3. Controles terminológicos

Un número de campos de la base de datos requieren controles terminológicos o tratamiento documental.

En general, se ha optado por dos tipos de controles:

– *Sistemas de clasificación*, que suelen ser grupos cerrados de términos. En estos campos, se debe optar por elegir un término como representación sintética. Se considera que el grupo de valores forma parte de un cuadro de clasificación sencillo, de un solo nivel.
– *Sistemas de indización asociativa*, formados por (candidatos a) descriptores, que son sistemas abiertos formados por términos de indización que serán

descriptores o candidatos a descriptores, en tanto aún nos encontramos en fase de creación del tesauro de la base de datos. En estos campos, se asignan tantos descriptores como temas relevantes presente el documento. Se trata aquí de realizar una representación exhaustiva, situada al mismo nivel de especificidad del documento.

En los campos con tratamiento terminológico basado en clasificaciones, existirán listas cerradas de valores admitidos, de manera que solamente se podrán entrar los valores presentes en esa lista, como forma de control.

En los campos basados en descriptores, éstos se asignarán siguiendo la normativa UNE de construcción de tesauros, de manera que, después de la indización de algunos centenares de recursos, se podrá inciar la recolección de términos de indización para contruir el futuro tesauro de la base de datos Acme.

3.4. Diccionario de datos

La entidad que describe la base de datos Acme son recursos digitales de Internet. Un recurso digital se presenta siempre como algún servicio o producto de información accesible mediante una URL.

A efectos del tratamiento en campos se han dividido los atributos de los recursos digitales en dos grupos:

– Atributos relevantes del recurso, como título del recurso, tema, etc.
– Elementos de control y gestión, como fecha de alta, modificación, etc.

La siguiente tabla lista ambos grupos de recursos:

Tabla de atributos:

Título Tipo de recurso Autor Fuente Lugar Idioma Clasificación Descriptores Descripción Valoración Última visita URL	**Elementos relevantes de la entidad**
Operador Número de registro Fecha de alta Fecha de modificación	**Elementos de control**

3.4.1. Diccionario de datos detallado

A continuación se analiza cada uno de los campos de la base de datos en base a sus parámetros esenciales. Algunas indicaciones de los campos pueden implementarse a través del sistema de gestión de la base de datos, como la *etiqueta*, el *tipo* de datos o la *obligatoriedad*; pero otros parámetros deben ser observados por el operador humano que realiza el análisis de la información y/o la carga de datos. Es el caso del *dominio* y del *tratamiento documental*.

La lista de campos con su tratamiento sistemático es la siguiente:

Campo Título

Etiqueta	**Titulo**
Dominio	Título propio o título atribuido del recurso, seguido del título traducido a la lengua del centro, en su caso. El título traducido se indica entre paréntesis. Ejemplo: Internet Movie Database (Base de datos de cine de Internet)
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del recurso/ Del centro
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	Si el recurso tiene subtítulo o un elemento que actúa como tal, debe indicarse también, en la forma <Título: subtítulo>.

Campo Tipo de recurso

Etiqueta	**Tipo**
Dominio	La clase de recurso digital: base de datos, institución, documento, etc.
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	Lenguaje controlado cerrado. Lista de valores admitidos: Base de datos Directorio Documento Institución Publicación periódica
Obligatorio	Sí
Observaciones	–

Campo Autor

Etiqueta	Autor
Dominio	Nombre de la persona o institución responsable intelectual del recurso. En caso de tratarse del nombre de una persona, entrar en forma invertida: Apellido, Nombre
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del recurso
Tratam. Docum.	NP
Obligatorio	No
Observaciones	–

Campo Fuente

Etiqueta	Fuente
Dominio	Nombre de la institución o empresa responsable de la edición del recurso en su forma actual
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	Ejemplos: Universidad Pompeu Fabra; Institut Català de Tecnología; IBM, etc.

Campo Lugar

Etiqueta	Lugar
Dominio	Topónimo de la institución fuente. A nivel internacional, con indicación del país, por ejemplo, <Francia>. A nivel nacional, con indicación de Ciudad, Comunidad Autónoma y país. Ejemplo: <Barcelona. Cataluña. España>
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	NP
Obligatorio	Sí. Si no se ha identificado, se indicará así: “no identificado”
Observaciones	–

Campo Idioma

Etiqueta	Idioma
Dominio	Lengua del recurso o lenguas del recurso
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	Lenguaje controlado abierto. Lista de valores más frecuentes: Catalán Castellano Francés Inglés
Obligatorio	Sí
Observaciones	–

Campo Clasificación

Etiqueta	Clasificacion
Dominio	Indicación sintética de la categoría o categorías temáticas principales del recurso
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	Lenguaje controlado cerrado. Lista de valores admitidos: Arte Cine Cultura Fotografía Humanidades Literatura Multimedia Música Teatro Televisión
Obligatorio	Sí
Observaciones	NP

Campo Descriptores

Etiqueta	Descriptores
Dominio	Términos de indización normalizados
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	Lenguaje controlado mediante tesauro
Obligatorio	Sí
Observaciones	La norma general consiste en asignar tantos descriptores como temas relevantes presente el recurso, siguiendo la norma UNE de creación de tesauros monolingues

Campo Descripción

Etiqueta	Descripcion
Dominio	Descripción textual del tema, contenido, orientación, etc., del recurso
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	Del centro
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo Valoración

Etiqueta	Valoracion
Dominio	Puntuación alcanzada por el recurso, en una escala de 1 a 3
Tipo de datos	Numérico
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo URL

Etiqueta	URL
Dominio	URL del recurso
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo Última visita

Etiqueta	Visitado
Dominio	Fecha de la última vez que fue comprobado el recurso
Tipo de datos	Fecha
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo Número de registro

Etiqueta	**RecNo**
Dominio	Número de registro
Tipo de datos	Numérico
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo Operador

Etiqueta	**Operador**
Dominio	Primer apellido de la persona que ha entrado los datos
Tipo de datos	Alfanumérico
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo Fecha de alta

Etiqueta	**Created**
Dominio	Fecha de creación del registro
Tipo de datos	Fecha
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

Campo Fecha de modificación

Etiqueta	**Modified**
Dominio	Fecha de la última modificación del registro
Tipo de datos	Fecha
Indización	Sí
Lengua	NP
Tratam. Docum.	NP
Obligatorio	Sí
Observaciones	–

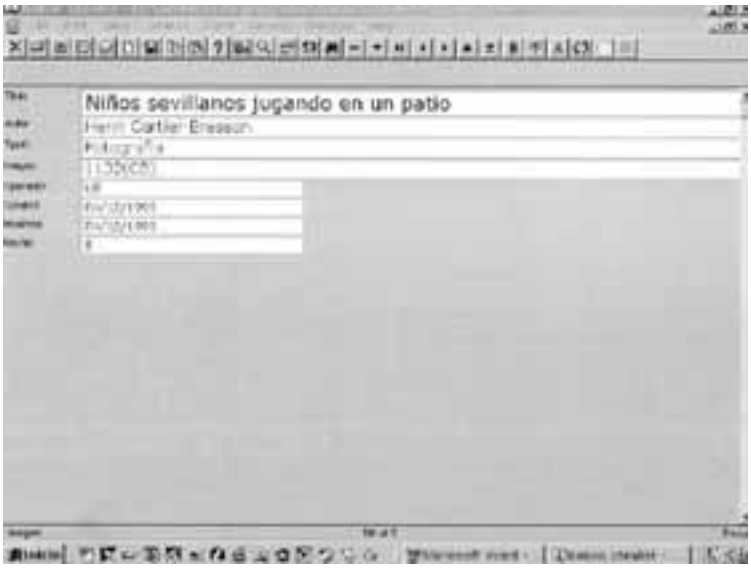
Anexos. Capturas de pantalla

Anexo 1

Un registro de la base de datos SOSIG

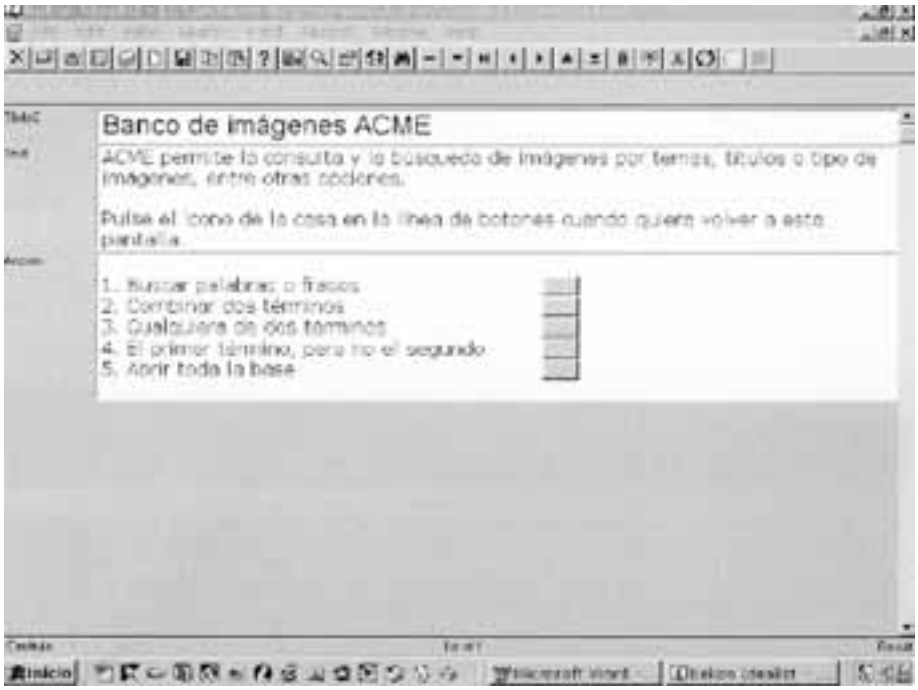
Anexo 2

Primera versión de un registro de la base de datos Imagen



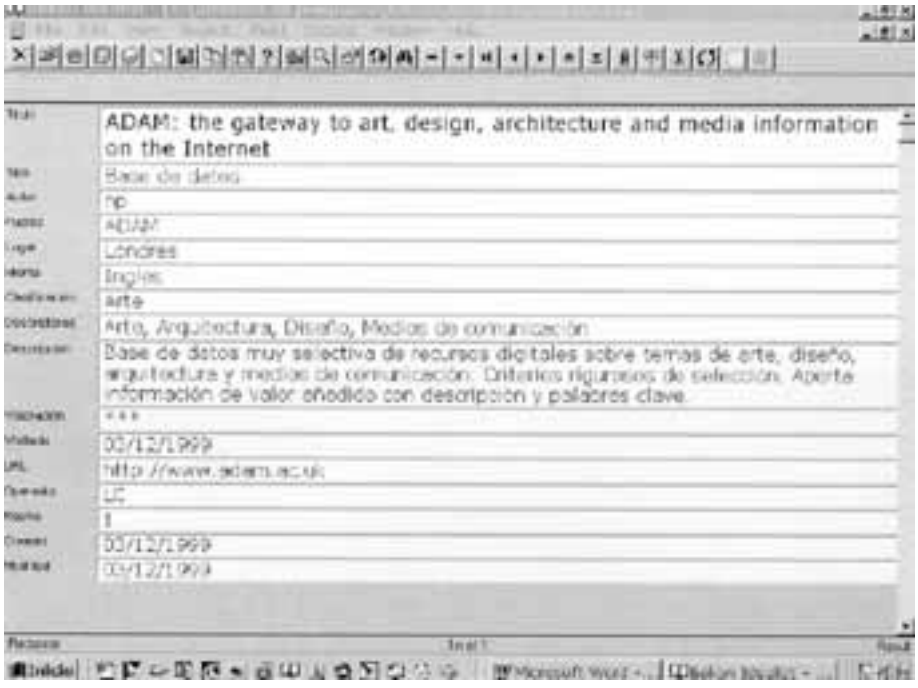
Anexo 3

Aspecto de la carátula de la base de datos Imagen



Anexo 4

La base de datos de recursos digitales del estudio de caso (Proyecto ACME)



Lluís Codina (1999). "Estudio de caso. Proyecto: sistema de información sobre recursos digitales en Internet de la editorial ACME" (novembre 1999).

El hipertexto: la recuperación de información por navegación en la web

Cristòfol Rovira
cristofol.rovira@cpis.upf.es

Introducción

Una herramienta es ergonómica cuando se integra de tal manera en las condiciones físicas y psíquicas del usuario que se hace transparente, cuando es tan fácil de utilizar que no es necesario prestarle atención y el usuario puede centrarse en la tarea que tiene que llevar a cabo.

El hipertexto es el elemento central de la ergonomía en la consulta de los documentos de Internet. El servicio World Wide Web[1] utiliza la navegación hipertextual para acceder de manera ágil a la información, formando una red interminable de documentos interrelacionados. Sólo hay que hacer clic en una palabra subrayada[2] para obtener de manera inmediata[3] el documento referenciado. La web es altamente ergonómica desde el punto de vista físico gracias al hipertexto. En cambio, desde el punto de vista cognitivo surgen problemas de desorientación cuando el lector activa de manera constante y desordenada las referencias disponibles. Parece como si la alta ergonomía física derivara en una baja ergonomía psíquica.

El hipertexto no es una idea nueva. Hay que buscar sus orígenes en los trabajos de Vannevar Bush durante la década de los años cuarenta, antes de la aparición de los ordenadores. Cualquier documento en papel incorpora elementos hipertextuales que permiten romper la lectura secuencial y lineal. Las notas a pie de página son el ejemplo más representativo. Por ello a menudo se ha definido el hipertexto como "la generalización de las notas a pie de página". Cuando el acceso a las referencias se convierte en el eje que estructura un documento, surge el hipertexto.

Durante los años ochenta, diferentes *softwares* han materializado la visionaria idea de Bush, poniendo a disposición del gran público la creación y consulta de documentos digitales no lineales. El más popular fue Hypercard, un excelente programa de creación de hipertextos que durante muchos años la empresa Apple regalaba al comprar un ordenador Macintosh. Las ayudas en línea de cualquier programa Windows son otro ejemplo de documento hipertextual ampliamente utilizado. Incluso los tratamientos de textos, como Word Perfect, permiten incorporar saltos hipertextuales en sus documentos.

Por otra parte, la web de Internet implementa el hipertexto sobre bases comerciales y tecnológicas completamente diferentes. Se trata de un sistema abierto y, por lo tanto, no es preciso adquirir un producto comercial para crear, gestionar y consultar documentos hipertextuales. La web es una tecnología de uso público que utiliza el hipertexto para consultar y recuperar la información de Internet por navegación.

Este profundo cambio de contexto comporta muchas ventajas y algunos inconvenientes. La principal ventaja es que no hay barreras comerciales o tecnológicas para hacer y leer documentos hipertextuales. Todo el mundo puede crear sus páginas. El principal inconveniente es que esta ausencia de limitaciones tecnológicas se ha traducido a menudo en una ausencia de calidad en el diseño hipertextual de los documentos de la Red.

El objetivo de este capítulo es analizar cómo son los documentos hipertextuales de la web; mostrar qué errores de diseño pueden interferir en una recuperación de la información y, finalmente, ver cuáles son las estrategias de navegación más efectivas para consultar la inmensa red de documentos hipertextuales de Internet.

2.1. Precisiones terminológicas

Antes de nada, debemos hacer algunas puntualizaciones terminológicas, ya que existe una cierta confusión en torno al concepto de hipertexto y los que le son próximos. Por una parte, un hipertexto es un documento digital que aprovecha la ventaja de la computabilidad para permitir un acceso asociativo a la información. De este modo, se rompe la secuencialidad que impone el soporte en papel. Por otra parte, el hipertexto es también el programa informático que hace posible la creación y lectura de estos nuevos documentos digitales. Finalmente, el modelo teórico de organización de la información de manera no secuencial también se llama hipertexto. Con propiedad, el término hipertexto sólo tiene esta última acepción, ya que:

> El documento digital con prestaciones de hipertexto se llama hiperdocumento y los programas informáticos para crearlo, modificarlo y consultarlo son los sistemas de gestión de hipertextos (SGH).

Un segundo foco de confusiones tiene su origen en la dicotomía hipertexto/hipermedia. En principio, un hipermedio sería un documento digital de acceso asociativo en cualquiera de las monologías de la información (texto, imagen y sonido). En cambio, un hipertexto estaría formado exclusivamente por texto. En la práctica, el término hipertexto alcanza esta significación más general desplazando el término hipermedia, que cada vez se utiliza menos. Nosotros utilizaremos los dos términos como sinónimos con el significado general que implica un contenido en cualquier morfología de la información.

2.2 Evolución del hipertexto: de Bush a la web de Internet

Hay un consenso generalizado (Concklin, 1987) en situar los inicios del hipertexto en los trabajos de Vannevar Bush durante la década de los cuarenta. Este autor ideó el sistema Memex, basado en microfichas, para organizar y recupe-

rar información textual. Aunque el Memex era materialmente inviable, Bush (1945) sentó las primeras bases conceptuales de lo que después se conocería como hipertexto.

En el otro extremo, el uso generalizado de los sistemas hipertextuales se ha producido a mediados de la década de los noventa gracias al servicio World Wide Web de Internet. La utilización del hipertexto como sistema de navegación en las redes telemáticas está popularizando un sistema de organización y recuperación de la información con más de cincuenta años de historia, aunque, en gran parte, los objetivos originales no se han conseguido todavía.

La evolución del hipertexto entre estos dos hitos se ha producido gracias a las aportaciones de diferentes autores, que desde distintas bases epistemológicas han centrado sus estudios en ampliar las capacidades cognitivas humanas por medio de la tecnología. Incluso el mismo Bush (1945) creó el sistema Memex para facilitar el acceso asociativo y no secuencial a la información, intentando imitar la supuesta manera como funciona el cerebro humano.

El objetivo de Bush era almacenar documentos y proporcionar los medios para hacer una lectura no secuencial de los mismos, saltando de un fragmento de texto a otro, en función de las necesidades del lector, y siguiendo un rastro de vínculos predefinidos.

Los siguientes pasos en el desarrollo del hipertexto llegaron veinte años después con la aparición de los ordenadores. Douglas Engelbart creó, a finales de la década de los sesenta, el sistema NLS (On Line System) con segmentos de textos relacionados con vínculos. Paralelamente, Ted Nelson desarrollaba el sistema Xanadu con el objetivo de automatizar la parte material del trabajo intelectual, integrando y relacionando textos, anotaciones, referencias y notas a pie de página.

El término *hipertexto* apareció en el año 1965, y fue creado por T. Nelson (1974). El origen de este término se atribuye a la influencia de la terminología utilizada en las películas de ciencia ficción, ya que el significado de la partícula "hiper" no tiene una relación clara con el concepto de hipertexto. Los términos supratexto, metatexto o *supertexto* habrían resultado conceptualmente más exactos, pero obviamente menos cinematográficos.

El primer sistema de gestión de hipertextos comercializado para microordenadores fue el PC's Guide, de la empresa OWL, en el año 1986. Sin embargo, el hipertexto llega al gran público gracias a Hypercard de Macintosh a partir de 1987. La política de Apple de regalar Hypercard a los compradores de ordenadores Macintosh provocó la popularización del nuevo concepto. Posteriormente, otros sistemas aparecieron en el mercado, con funcionalidades diferentes pero con prestaciones parecidas a Hypercard: un documento digital consultable de manera no secuencial por medio de vínculos entre fragmentos de información.

Poco a poco, el hipertexto se convirtió en la estructuración natural del documento digital en las enciclopedias electrónicas y los libros multimedia. Pero el uso masivo del hipertexto y del documento digital en general se ha producido gracias a la red Internet. El servicio World Wide Web ha convertido Internet en un espacio virtual mundial de documentos hipertextuales.

2.3. El hipertexto en la web

Si diseccionamos un hipertexto identificaremos tres elementos básicos: nodos, vínculos y ancorajes. Los nodos se llaman "páginas" en el contexto de Internet y son los documentos individuales que forman un hipertexto. Los vínculos son las conexiones lógicas con estos documentos y, finalmente, los ancorajes son los puntos físicos de salida y de llegada de un vínculo. En la web los ancorajes de salida toman la forma de palabras subrayadas, iconos gráficos o mapas táctiles, y los ancorajes de llegada son normalmente las zonas iniciales de las páginas web.

Por lo tanto, un hipertexto es un documento digital organizado en forma de red mediante nodos, vínculos y ancorajes. Como decíamos en el apartado 2.2, se utiliza la palabra "hiperdocumento" para denominar el documento digital con prestaciones hipertextuales y "programa de gestión de hipertextos", para el programa informático capaz de crear y consultar un documento hipertextual. En el contexto tecnológico de Internet, un hiperdocumento puede identificarse con un sitio web (w*eb* site) y las tareas de los programas de gestión de hipertextos son asumidas por los navegadores[4] y por los editores de HTML[5].

El hipertexto ideal

Conklin (1987, pág. 19) propone una serie de características básicas que definen y delimitan el hipertexto ideal:

1. Una base de datos en red formada por nodos de información textual y gráfica.
2. Los nodos de la base de datos se visualizan en la pantalla del ordenador por medio de ventanas. Una ventana corresponde a un nodo y sólo se puede ver un pequeño número de ventanas al mismo tiempo en pantalla.
3. Las ventanas son manejadas siguiendo las convenciones estándar (abrir, cerrar, desplazar, etc.).
4. Las ventanas contienen vínculos que representan conexiones a otros nodos de la base de datos. Los vínculos contienen texto para explicitar el contenido del nodo apuntado y la acción de activarlo hace que se abra una ventana y se muestre su contenido.

5. El usuario puede crear nuevos nodos o modificar los existentes (anotaciones, comentarios, reelaboraciones, etc.).

6. La base de datos puede visualizarse de tres maneras:

   a) Siguiendo los vínculos y visualizando los contenidos de los nodos apuntados.

   b) Mediante búsquedas por palabras clave u otros atributos de los nodos.

   c) Por medio de la visualización gráfica de la red de conexiones.

La tecnología web también es abierta en el formato físico de los hiperdocumentos. Los ficheros generados para los programas clásicos de gestión de hipertextos tienen un formato propietario que sólo el mismo programa puede generar o modificar. En cambio, los documentos web están en formato ASCII, un formato estándar que cualquier editor de textos puede generar. El diseño gráfico, tipográfico e hipertextual de los hiperdocumentos web viene determinado por un conjunto de etiquetas o marcas textuales, el llamado lenguaje HTML, intercaladas en el texto del documento. Por lo tanto, físicamente el fichero tiene formato ASCII, pero gracias a las etiquetas el formato lógico es HTML. Los navegadores[6] son programas informáticos de uso público que interpretan las etiquetas y construyen las páginas web con todos los atributos propios de un nodo hipertextual. Estos programas tienen una doble función: gestionar la conexión telemática en Internet[7], y permitir la consulta o navegación de los documentos hipertextuales.

2.4. Herramientas de navegación

Sólo con nodos, vínculos y ancorajes obtendríamos un hipertexto de muy difícil consulta, ya que una pantalla de ordenador no ayuda a obtener una visión del contexto del nodo activo. En cambio, el grosor de las páginas de un libro ofrece un referente elemental para saber lo que ya se ha leído y lo que falta por leer.

Por lo tanto, en un hipertexto son imprescindibles las ayudas a la navegación, ya que evitan que se pierda el rumbo de la lectura y facilitan el encuentro de nuevas rutas de navegación. Los sumarios e índices son las "herramientas de navegación" de los documentos en papel. Permiten localizar las zonas del documento que tratan de un determinado tema al mismo tiempo que ofrecen una visión global del contenido. En las obras de referencia son prestaciones esenciales.

Sin embargo, la principal ayuda a la "navegación sobre el papel" está tan integrada en la esencia de los documentos en papel que se ha descubierto con la aparición del hipertexto: la estructuración secuencial de la información. Los

documentos en papel tienen una única ruta de lectura[8] y, por lo tanto, no hay pérdida. En cambio, la esencia de un hipertexto es precisamente la multiplicidad de rutas de lectura, las inmensas posibilidades de rumbos de navegación. Por ello en la navegación hipertextual las ayudas a la navegación son de vital importancia.

En la web, los tradicionales sumarios o tablas de contenido adoptan la forma de menús. Es muy común que la primera página de un sitio web contenga un menú de los principales apartados. Otras veces aparecen menús de forma fija en una parte de color a la izquierda de la página. Dado que no hay un itinerario preestablecido, la estructura jerárquica del menú ofrece los puntos de referencia necesarios para decidir el objetivo de la navegación y para la orientación en caso de eventuales rupturas de este itinerario elegido.

Página principal de ICTnet, donde por medio de gráficos se muestra un primer nivel jerárquico de los menús.

Normalmente, los menús de las páginas web sólo muestran un nivel jerárquico. Es difícil encontrar sitios web con una representación global de su contenido por medio de unos menús completamente desarrollados o de mapas de contenido que muestren, de una manera gráfica, todas las páginas y sus vínculos. Esta circunstancia dificulta la visión global de un sitio web. Recomendamos solucionarlo haciendo un primer recorrido exploratorio para localizar los diferentes submenús.

Con el fin de ofrecer una visión global, los mapas de contenido deberían ocupar una sola pantalla y ser accesibles desde todos los nodos del hiperdocumento. En los sitios web muy extensos es justificable la ausencia de estos mapas globales, ya que en el poco espacio disponible de una pantalla es materialmente imposible representar de manera inteligible un contenido amplio. La solución consiste en complementar mapas globales poco desarrollados con mapas locales en los que se representen los diferentes apartados con más detalle.

Menú principal de la Yale Web Style Guide, un sencillo menú HTML de un solo nivel jerárquico que representa una web de más de quinientas páginas.

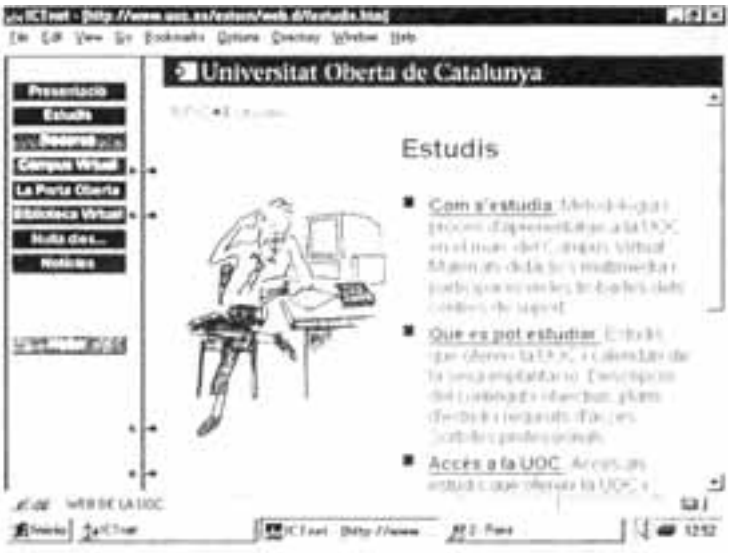
Otra posibilidad es la de integrar un mapa global combinándolo con uno local por medio de un *Fish-eye view* o "mapa de ojo de pez". En este mapa se puede ver la red de nodos en torno al nodo activo. Los nodos conceptualmente más próximos son representados de manera más detallada que los más alejados.

Se trata de un tipo de mapa de difícil implementación con tecnología web, ya que exige un tratamiento específico para cada nodo. Algunos programas de creación de sitios web, como Front Page, incorporan prestaciones que generan mapas del contexto del nodo activo. Sin embargo, no son auténticos mapas de ojo de pez porque sólo representan los nodos que están a distancia de un salto hipertextual de la página activa, tanto los que apuntan contra la página activa como los que son apuntados. Tampoco podemos decir que esta prestación sea una auténtica herramienta de navegación, ya que sólo se activa contra páginas residentes en el disco duro para ayudar en su construcción.

La rama jerárquica es otra herramienta de navegación, derivada del mapa de ojo de pez, que incorporan algunas páginas web de las últimas generaciones. Consiste[9] en representar el camino de nodos que habría que recorrer para llegar a una determinada página siguiendo la estructura jerárquica del hiperdocumento. Cada nodo del camino se representa con su título en forma de palabra activa; de esta manera, se ofrecen dos prestaciones: informar de la rama jerárquica en la que se encuentra la página y permitir saltar directamente a cualquier página de nivel superior. El buscador-índice Yahoo![10] incorpora esta sencilla y efectiva herramienta que facilita la navegación por la estructura de vínculos jerárquicos de un sitio web.

Los gráficos GIF clicables (o mapas táctiles) permiten implementar los mapas de contenido de una manera gráfica. El resultado es un nodo mucho más impactante que otro realizado con texto e hiperlinks. También se utilizan GIF animados o aplets de Java para conseguir que los gráficos cambien de color al situar el cursor encima de los mismos.

Por otro lado, la estructura de *frames* o subpantallas[11] se utiliza a menudo para mostrar menús de manera fija a la izquierda o en la parte superior de la pantalla. Aunque ocupan una parte importante del valioso espacio de la pantalla, los *frames* permiten que el usuario mantenga siempre a la vista el menú que corresponde a la página que está consultando. Sin embargo, cuando ya se conoce el sitio web, se agradece la posibilidad de quitar el menú y aprovechar todo el espacio para el contenido de la página.



En la web de la Universitat Oberta de Catalunya el menú aparece de manera fija a la izquierda. En la parte superior aparece representado la rama jerárquica a la cual pertenece la página activa.

El botón de retroceso es probablemente la herramienta de navegación más utilizada. Permite deshacer el último salto hipertextual, es soportado por el propio navegador y está situado en la barra superior de botones. Gracias a esta opción puede afirmarse que todos los vínculos web son de doble sentido. Además, haciendo clic en este botón sucesivas veces puede deshacerse todo el camino recorrido. Es una herramienta básica, ya que muy a menudo no hay suficiente información para prever si interesará o no el nodo al que apunta un determinado vínculo. Entonces hay que activar el salto hipertextual, visitar el nodo referenciado y si no interesa su contenido, hacer clic sobre el botón de retroceso.

Las idas y venidas pueden evitarse ampliando la información sobre el nodo de destino que da la palabra subrayada o icono. Esta herramienta de navegación, llamada vínculo etiquetado, activa una frase explicativa al poner el cursor sobre el ancoraje de salida. Su función es evitar que el lector entre en páginas que no están de acuerdo con su ruta de navegación y que, por lo tanto, tendrían poco interés. En la tecnología web el etiquetado de vínculos se implementa por medio de un guión JavaScript que muestra una frase en la barra inferior de estado al poner el cursor encima de la palabra subrayada. La versión 4.0 de HTML propone una solución más elegante para el etiquetado de nodos que los navegadores todavía no soportan.

En cambio, todos los navegadores soportan la historia de los nodos visitados. Se trata de una herramienta de navegación que hace un listado por orden cronológico de todas las páginas visitadas hasta el momento. Cada página viene expresada por su título interno[12] en forma de vínculo activo que nos permite acceder de manera directa a cada una de las páginas visitadas. Así, podemos saltar directamente a aquella página que hemos visitado hace veinte minutos.

2.5. Nodos

En un hipertexto, cada nodo es un documento relativamente independiente. La multiplicidad de rutas de navegación obliga a los nodos a no depender excesivamente de la información del contexto. El título materializa esta especificidad e independencia de cada nodo y es un punto de referencia crítico, con más implicaciones que los títulos de los apartados de un documento secuencial. En el momento de construir la red de conexiones, el título permite identificar los nodos más adecuados y en el momento de las consultas hace más amigable el establecimiento de referencias. Por ejemplo, sería muy poco práctico hacer un listado de nodos visitados con los nombres de los ficheros en lugar de los títulos de las páginas.

No hay un consenso sobre cuál es la dimensión óptima de los nodos. A pesar de todo, cuanto más extensos sean, más difícil resultará hacer la red de conexiones, y cuanto menores, más fraccionada, inconexa y dependiente de los vínculos resultará la información. Por lo tanto, hay que llegar a un compromiso y conseguir que cada nodo contenga el desarrollo completo de una idea sin superar las tres pantallas de ordenador[13].

En la tecnología web a menudo no se respetan ni los títulos ni el tamaño de los nodos. Podemos encontrar extensas páginas con un título interno poco significativo que en realidad no son nodos hipertextuales, sino documentos secuenciales puestos en la Red. En estos casos no se pueden hacer muchas recomendaciones, ya que en realidad no estaremos consultando un hipertexto, sino un documento tradicional.

Las páginas web extensas incorporan vínculos hipertextuales internos que permiten saltar de un punto a otro de la misma página. Esta prestación se utiliza en documentos largos para acceder desde el sumario, colocado al comienzo de la página, al desarrollo de cada uno de los puntos.

Una segunda utilidad del vínculo interno es permitir el salto desde el final al inicio de la página. Es una prestación muy generalizada que evita utilizar el *scroll* (barra de desplazamiento) para volver al comienzo de la página. El anco-raje de partida por este vínculo es muchas veces la palabra “inicio” o una flecha que señala hacia arriba.

Las guías de estilo sobre la creación de sitios web recomiendan dividir la página en tres zonas: encabezamiento, cuerpo y pie de página. También sugieren in-

cluir determinados elementos en cada zona. Hay un cierto consenso en recomendar que se ponga el título[14] en el encabezamiento y el nombre del autor, fecha de creación y fecha de la última actualización a pie de página.

Además, para facilitar la navegación, también se recomienda que todas las páginas incluyan (en el encabezamiento o a pie de página) un vínculo directo con la página inicial, también llamada *home page*, donde aparece un menú que ofrecerá una visión de conjunto y que integra las páginas en su contexto.

Otra herramienta de navegación de uso generalizado es el marcaje de nodos. Es una prestación soportada por los navegadores que permite guardar la dirección del nodo activo para acceder al mismo de manera directa en futuras ocasiones. En la web resulta una herramienta imprescindible, ya que la red Internet contiene una cantidad inmensa de hiperdocumentos difícilmente controlable sin el mantenimiento efectivo de una lista de direcciones o *bookmarks*.

En relación con esta herramienta, recomendamos clasificar las direcciones por temas y colocar las nuevas direcciones en el apartado correspondiente cuando se haga el marcaje.

Sólo las últimas versiones de los navegadores permiten clasificar una nueva dirección en el apartado que le corresponde en el mismo momento de hacer el marcaje. En las versiones anteriores hay que hacerlo en dos pasos, primero marcar y después abrir la lista y clasificar. Por ello recomendamos seguir los dos pasos en el momento en que se decida guardar la dirección y acumular la tarea de clasificación.

2.6. Estructuras hipertextuales

La navegación web es tan fácil que se pierden las fronteras entre los hiperdocumentos. Es tan sencillo saltar entre dos páginas del sitio web activo como saltar a un hiperdocumento a miles de kilómetros de distancia. Esta circunstancia resulta muy desorientadora para los navegantes neófitos porque normalmente el diseño gráfico tampoco contiene indicaciones explícitas sobre si se abandona o no el actual sitio web al activar un vínculo.

Para identificar las fronteras entre los hiperdocumentos de la Red es muy interesante tener en cuenta la estructura de vínculos sobre la cual están construidos los hipertextos. La organización de los vínculos de un hipertexto está basada en tres estructuras básicas: la jerárquica, la de red y la secuencial.

Los vínculos jerárquicos unen los nodos en función del grado de especificidad del tema tratado. Permiten navegar desde una exposición general a una específica y viceversa. La representación de la estructura jerárquica de todos los nodos unidos por relaciones jerárquicas constituye un mapa de contenido[15] que muestra la manera y la profundidad en el desarrollo del tema tratado. Es la misma función que cumplen los sumarios de los documentos secuenciales.

Los vínculos asociativos rompen esta jerarquía de nodos y materializan una estructura en forma de red. Expresan diferentes tipos de conexiones entre el contenido de dos nodos: complementariedad, resumen, ampliación, fundamentación, causa, efecto, etc.

En Internet, los vínculos jerárquicos se materializan en menús que relacionan dos páginas de un mismo sitio web. Son los indicadores más fiables para identificar hasta dónde llega el sitio web activo y para establecer las fronteras entre los diferentes hiperdocumentos de la red Internet.

A menudo resulta difícil identificar cuáles son los vínculos jerárquicos porque en Internet todos los ancorajes de partida tienen la misma forma (palabra subrayada o icono), con independencia de que se trate de un vínculo jerárquico, de causa, de efecto, de resumen, etc. Sin embargo, el diseño gráfico puede ayudar. Cuando el ancoraje quede integrado en el texto, probablemente se trate de un vínculo asociativo, y cuando forme parte de un menú, es posible que sea jerárquico.

Para identificar el tipo de vínculo recomendamos observar la barra de estado[16]. Al poner el cursor sobre el ancoraje de partida, la barra de estado muestra la dirección web a la que apunta el vínculo. Comparando esta dirección con la de la página activa podemos deducir si se trata de un vínculo externo o interno. Generalmente, los vínculos externos son asociativos; en cambio, los internos pueden ser jerárquicos o asociativos y será preciso guiarse por el diseño gráfico.

La última estructura es la secuencial, que conecta los nodos de manera lineal. Cada nodo tiene dos vínculos: el nodo anterior del recorrido propuesto y otro posterior. Muy a menudo el ancoraje de partida adopta la forma de flecha. Si apunta a la izquierda, indicará un salto al nodo anterior, y si apunta a la derecha, a la posterior.

La estructura secuencial puede tener tres funciones: unir un subconjunto de nodos que van seguidos; permitir un recorrido clásico por todo el hiperdocumento y, finalmente, proponer una "visita guiada" por los nodos más significativos.

En la web la estructura secuencial se utiliza a menudo para evitar páginas largas. Cuando una página ocupa más de tres o cuatro pantallas conviene hacer dos nodos independientes y unidos con vínculos secuenciales. Sin embargo, no es aconsejable seguir este principio para transformar un documento secuencial en documento hipertextual. Un documento secuencial ha sido creado para leerse en un determinado orden. Por lo tanto, cada apartado implica la lectura de todos los anteriores y el autor puede fundamentar su argumentación sobre todo lo que ya ha expuesto. El hilo argumental se desarrolla dando por implícitas partes del discurso.

En cambio, el autor de un hipertexto no controla la ruta que seguirá el lector. Por lo tanto, es preciso que cada nodo desarrolle de manera completa una sola

idea, sin depender en exceso de otras partes del hiperdocumento y explicitando todas las posibles relaciones semánticas con otros nodos.

En definitiva, la fragmentación en nodos es sólo un paso en el proceso de creación de un hiperdocumento. Será preciso reestructurar y probablemente volver a redactar los contenidos para adecuarlos al nuevo medio. En la web los contenidos no se han adaptado siempre a las exigencias del medio hipertextual, lo que ha dado como resultado páginas excesivamente largas y mal cohesionadas.

2.7. Estrategias de navegación

La recuperación de información es el proceso que permite obtener de un fondo documental los documentos adecuados para unir demanda de información expresada inicialmente en lenguaje natural. Las demandas se hacen a partir de las propiedades formales del documento (autor, título, editorial, año de publicación, etc.) o a partir de las propiedades semánticas, a partir de su contenido (documentos que traten de...).

Generalmente, el término de "recuperación de información" se utiliza cuando se obtiene la información al interrogar una base de datos documental o un motor de búsqueda de Internet. En otros capítulos de esta obra se ha tratado con detalle este proceso de recuperación de información por interrogación.

La consulta de un hipertexto también se puede considerar como un proceso de recuperación de información, pero no por interrogación, sino por navegación. También se parte de una demanda de información y se obtiene un conjunto de documentos (nodos) que satisfacen esta demanda. No obstante, las posibilidades de recuperación por navegación están limitadas a las propiedades semánticas de los documentos expresadas en los vínculos que relacionan nodos de acuerdo con sus contenidos. En cambio, en la recuperación por interrogación, el contenido o materia de los documentos es sólo uno de los posibles criterios de búsqueda. Como decíamos antes, un motor de búsqueda también puede ser interrogado a partir de las propiedades formales de los documentos digitales, como el año de publicación, la editorial o el autor.

Además, la mecánica del proceso es completamente diferente en ambos casos. En la interrogación de una base de datos se obtiene un listado de documentos después de entrar un conjunto de palabras clave en el formulario del sistema interrogado. En cambio, en la navegación se obtienen sucesivos nodos del hipertexto a partir de una ruta de navegación elegida y materializada en la activación sucesiva de vínculos.

Hay dos maneras básicas de realizar una ruta de navegación: por anchura y por profundidad o largura. En la navegación por anchura se activan todos los vínculos del nodo activo. En viajes de ida y vuelta se consultan todos los nodos referenciados antes de decidir por dónde continuará la ruta de navegación.

La estrategia de profundidad consiste en adoptar la actitud contraria y elegir el vínculo que más interesa de cada nodo, avanzando por un único camino, sin considerar las ramificaciones.

En la recuperación de información por navegación en la web habrá que combinar anchura y profundidad según los resultados que se obtengan en cada momento. La adecuación del vínculo al tema de interés marcará la estrategia que deberá utilizarse. Así, se avanzará en profundidad en las páginas que traten temas próximos al que estamos buscando para activar sólo los vínculos que pueden conducir a documentos más pertinentes. En cambio, habrá que adoptar una estrategia de anchura, y explorar todos los vínculos, en las páginas que contengan selecciones de recursos sobre el tema de interés. Probablemente, muchas de las páginas referenciadas encajarán con nuestros intereses y algunas contendrán nuevas selecciones de recursos que será preciso explorar de nuevo por anchura.

Las secciones de "recursos de interés" que incorporan muchos sitios web hacen muy rentable la localización de recursos pertinentes porque abren el camino para la obtención de muchos otros. Muy a menudo, el problema es localizar el primer recurso adecuado a nuestras necesidades. Por este motivo, recomendamos utilizar los motores de búsqueda y los índices para localizar las primeras páginas de una sesión de navegación y a continuación avanzar en profundidad o anchura según el interés de los vínculos.

En la navegación hipertextual en la web es muy fácil perder el rumbo, sobre todo cuando se encuentran muchos recursos interesantes y se multiplican las ramificaciones para explorar. Hay que guardar en la lista de direcciones las páginas en las que se ejecuta una estrategia de anchura y clasificarlas inmediatamente[17], ya que así se evita perder los puntos de referencia esenciales del proceso de navegación.

También hay que guardar en la lista de direcciones los vínculos interesantes sobre temas que no tienen nada que ver con lo que estamos buscando. Muy a menudo, la pérdida de rumbo se debe al cambio provisional del objetivo de navegación a causa de hallazgos valiosos e inesperados. No es recomendable seguir los vínculos que abren nuevas rutas de navegación, ya que llevan fácilmente a una situación de "desbordamiento cognitivo". La navegación se hace imposible porque no puede mantenerse el control sobre las dos o más rutas de navegación con las múltiples ramificaciones de cada una. Por lo tanto, recomendamos guardar en la lista de direcciones estos vínculos tan interesantes que nos llevarán a la perdición (nunca mejor dicho) y dejar para más tarde su exploración.

Los navegadores permiten guardar la dirección de un vínculo sin necesidad de entrar en la página referenciada. Hay que situar el cursor sobre el vínculo y hacer clic sobre el botón derecho del ratón. Se abre un menú móvil con la opción de guardar la dirección asociada al vínculo. Guardando las direcciones de esta manera se evita abrir rutas paralelas de navegación sin perder la referencia de

páginas interesantes que se pueden explorar en otro momento. Como puede comprobarse en la tabla siguiente, la tecnología web ofrece herramientas de ayuda a la navegación soportadas de maneras muy diversas.

Ayuda a la navegación	Soportado por
Menús, índices o sumarios	Lenguaje HTML
Botón de retroceso	Navegador
Historia de los nodos visitados	Navegador
Mapa táctil de contenido	Lenguaje HTML
Menú global	Lenguaje HTML
Menús desplegables	JavaScript
Rama jerárquica	Lenguaje HTML
Marcaje de nodos	Navegador
Incorporar anotaciones	CG1
Vínculos etiquetados	JavaScript Nueva propuesta de HTML
Retorno directo a la página inicial	Lenguaje HTML
Retorno directo al inicio de página	Lenguaje HTML
Visión del contexto de vínculos de un nodo	Programas de creación de sitios web (Front Page)

2.8. Colaboración hipertextual

La interactividad de hipertexto puede ir mucho más allá de la elección del vínculo más adecuado a las necesidades del usuario. En los hipertextos conectados en red, el lector puede convertirse en autor incorporando nuevos documentos al hiperdocumento. Se trata de una prestación interesante, a menudo utilizada en las redes locales para soportar grupos de trabajo, que también está disponible en el entorno web.

Foro de discusión de Extra!-Net, donde cualquier usuario puede pasar de navegante a autor incorporando nuevos mensajes al foro.

Por medio de programas del tipo CGI es posible crear nuevas páginas en un sitio web remoto que contengan mensajes textuales o gráficos. Sólo hay que acceder a un formulario de introducción de datos y cumplimentarlo, y de manera inmediata el programa genera una página HTML que recoge la nueva aportación. Estos programas a menudo se utilizan para soportar conferencias electrónicas o grupos de discusión en los que las intervenciones quedan estructuradas en forma de cadenas de preguntas y respuestas. La consulta de estas cadenas de intervenciones se realiza por medio de vínculos hipertextuales que permiten el acceso a los mensajes almacenados.

Conclusiones

En el fondo, la única novedad que aporta el hipertexto es la inmediatez en la obtención de los documentos referenciados. Esta sencilla prestación ha generado una nueva forma de organizar y "leer" la información con importantes consecuencias desde el punto de vista cognitivo que provocarán cambios culturales profundos. El habla y la escritura son secuenciales, pero la memoria no, como tampoco lo es el proceso intelectual para la creación de un discurso estructurado. Los creadores del hipertexto (V. Bush, T. Nelson) buscaban una herramienta para contribuir a este proceso haciendo explícitas las conexiones asociativas previas al discurso elaborado.

El hipertexto de Internet se ha apartado de estos orígenes, ya que la web recoge de manera muy parcial los resultados de muchos años de investigaciones sobre la forma idónea de organizar y consultar un documento hipertextual. A pesar de todo, el entorno tecnológico abierto de la web ha popularizado de manera definitiva la navegación hipertextual. Este uso generalizado del hipertexto es el origen de un cambio cultural con repercusiones todavía no delimitadas. Sin embargo, la web ofrecerá toda su potencialidad cuando la velocidad de transmisión de las redes y la resolución de los monitores no sean elementos disuasorios para la consulta de documentos digitales.

Notas

1 A partir de ahora "web".
2 En la tecnología web se llama *hiperlink*.
3 Si la Red no está colapsada.
4 Las últimas generaciones de navegadores, como Comunicator o Netscape Gold, incorporan también un editor de HTML.
5 Puesto que una página web es un documento ASCII con las correspondientes etiquetas HTML, podemos considerar que cualquier editor ASCII es un editor de HTML.
6 Como Netscape o Internet Explorer.
7 La parte "cliente" de la conexión.
8 La única ruta ortodoxa posible, la que el autor espera que haga el lector.
9 Podéis ver el gráfico 3.
10 http://www.yahoo.com
11 La estructura de *frames* (marcos o recuadros) divide la ventana del navegador en diferentes subventanas. En cada subventana se visualiza un Fichero HTML.
12 El título interno de una página HTML aparece en la parte superior del marco de la ventana del navegador. Es el resultado de las etiquetas HTML <title> y <title>.

[13] Indicación de la *Yale Guide*.
[14] Que puede coincidir o no con el título interno.
[15] Hemos tratado los mapas de contenido en un punto anterior.
[16] Barra que está en la parte inferior de la pantalla de los navegadores y que informa de aspectos técnicos de la navegación.
[17] Podéis consultar el apartado de herramientas de navegación.

Bibliografía

Bush, V. "As we may think". *Atlantic Monthly*. Núm. 176, julio 1945, p. 101-108.

Conklin, J. "Hypertext: An Introduction and Survey". *IEEE Computer*. Vol. 20, núm. 9, septiembre 1987, p. 17-41.

Díaz, Paloma; N. Catenazzi; I. Aedo. *De la Multimedia a la Hipermedia*. Madrid: Ra-Ma, 1996.

Engelbart, D.C. "A Conceptual Framework for the Augmentation of Man's Intellect". Howerton (ed.) *Vistas in Information Handing*. Londres: Spartan Books, 1963.

Lynch, P.; Horton, S. *Yale C/AIM Web Stile guide* [en línea]. Rev. 1/97. Yale University, 1997. [Consulta: 10 de marzo, 1998] <http://info.med.yale.edu/caim/manual/index.html>

Nelson T.H. *Dream Machines*. South Bend, IN: The Distributers, 1974.

Nielsen, J. *Hypertext and hypermedia*. Boston: Academic Press, 1991.

Nielsen, J. T*he Alertbox: Current Issues in Web Usability* [en línea]. [Consulta: 10 de marzo, 1998] <http://www.useit.com/alertbox/>

Rada, Roy. *Hypertext: From text to Expertext*. Londres: McGraw-Hill, 1991.

Shneiderman, B.; Kearsley G. *Hypertext Hands-On!: An Introduction to a New Way of Organizing and Accesing Information*. Reading, Massachussetts: Addison-Wesley, 1989.

Cristòfol Rovira (1998). "L'hipertext: la recuperació d'informació per navegació al web". En: **Jaume Baró i Queralt** (ed.) (1998). *Cercar i col locar informació en el World Wide Web* (cap. 2, págs. 57-79). Barcelona: Llibres de l'Índex, Quaderns de Comunicació, 7.

Organizing Information

The biginning od all understnading is classification.
Hyaden White

In this chapter:

- Organizational Charges
- Organizing Web Sites and Intranets
- Creating Cohesive Organization Systems

Out understanding of the world is largely determined by our ability to organize information. Where do you live? What do you do? Who are you? Our answers reveal the systems of classification that form the very foundations of our understanding. We live in towns within states within countries. We work in departments in companies in industries. We are parents, children, and siblings, each an integral part of a family tree.

We organize to understand, to explain, and to control. Our classification systems inherently reflect social and political perspectives and objectives. We live in the *first* world. They live in the *third* world. She is a freedom fighter. He is a terrorist. The way we organize, label, and relate information influences the way people comprehend that information.

As information architects we organize information so that people can find the right answers to their questions. We strive to support casual browsing and directed searching. Our aim is to apply organization and labelling systems that make sense to users.

The Web provides us with a wonderfully flexible environment in which to organize. We can apply multiple organization systems to the same content and escape the physical limitations of the print world. So why are many large web sites so difficult to find information? Why can't the people who design these sites make it easy to find information? These common questions focus attention on the very real challenge of organizing information.

Organizational Challenges

In recent years, increasing attention has been focused on the challenge of organizing information. Yet, this challenge is not new. People have struggled with the difficulties of information organization for centuries. The field of librarianship has been largely devoted to the task of organizing and providing access to information. So why all the fuss now?

Believe it or not, we're all becoming librarians. This quiet yet powerful revolution is driven by the decentralizing force of the global Internet. Not long ago, the responsibility for labelling, organizing, and providing access to information fell squarely in the laps of librarians. These librarians spoke in strange languages about Dewey Decimal Classification and the Anglo-American Cataloging Rules. They classified, catalogued, and helped us find the information we needed.

The Internet is forcing the responsibility for organizing information on more of us each day. How many corporate web sites exist today? How many personal home pages? What about tomorrow? As the Internet provides us all with the freedom to publish information, it quietly burdens us with the responsibility to organize that information.

As we struggle to meet that challenge, we unknowingly adopt the language of librarians. How should we *label* that content? Is there an existing *classification system* we can borrow? Who's going to *catalog* all of that information?

We are moving towards a world where tremendous numbers of people publish and organize their own information. As we dos so, the challenges inherent in organizing that information become more recognized and more important. Let's explore some of the reasons why organizing information in useful ways is so difficult.

Ambiguity

Classification systems are built upon the foundation of language, and language is often ambiguous. That is, words are capable of being understood in two or more possible ways. Think about the word *pitch*. When you say *pitch*, what do I hear?

There are actually more than 15 definitions, including:

- A throw, fling, or toss
- A black, sticky substance used for waterproofing.
- The rising and falling of the bow and stern of a ship in a rough sea.
- A salesman's persuasive line of talk.
- An element of sound determined by the frequency of vibration.

This ambiguity results in a shaky foundation for our classification systems. When we use words as labels for our categories, we run the risk that users will miss our meaning. This is a serious problem. See Chapter 5, *Labeling Systems*, for more on this issue.

It gets worse. Not only do we need to agree on the labels and their definitions, we also need to agree on which documents to place in which categories. Consider the common tomato. According to Webster's dictionary, a tomato is *a red or yel-*

lowish fruit with a juicy pulp, used as a vegetable: botanically it is a berry. Now I'm confused. Is it a fruit or a vegetable or a berry?*

If we have such problems classifying the common tomato, consider the challenges involved in classifying web site content. Classification is particularly difficult when you're organizing abstract concepts such as subjects, topics, or functions. For example, what is meant by *alternative healing* and should it be cataloged under *philosophy* or *religion* or *health and medicine* or all of the above? The organization of words and phrases, taking into account their inherent ambiguity, presents a very real and substantial challenge.

Heterogeneity

Heterogeneity refers to an object or collection of objects composed of unrelated or unlike parts. You might refer to grandma's homemade broth with its assortment of vegetables, meats, and other mysterious leftovers as heterogeneous. At the other end of the scale, homogeneous refers to something composed of similar or identical elements. For example, Oreo cookies are homogeneous. Every cookie looks and tastes the same.

An old fashioned library card catalog is relatively homogeneous. It organizes and provides access to books. It does not provide access to chapters in books or collections of books. It may not provide access to magazines or videos. This homogeneity for a structured classification system. Each book has a record in the catalog. Each record contains the same fields: author, title, and subject. It is a high-level, single-medium system, and works fairly well.

Most web sites, on the other hand, are highly heterogeneous in two respects. First, web sites often provide access to documents and their components at varying levels of *granularity*. A web site might present articles and journals and journal databases side by side. Links might lead to pages, sections of pages, or to other web sites. Second, web sites typically provide access to documents in *multiple formats*. You might find financial news, product descriptions, employee home pages, image archives, and software files. Dynamic news content shares space with static human resources information. Textual information shares space with video, audio, and interactive applications. The web site is a great multimedia melting pot, where you are challenged to reconcile the cataloging of the broad and the detailed across many mediums.

* "The tomato is technically a berry an thus a fruit, despite an 1893 U.S Supreme Court decision that declared it a vegetable. (John Nix, an importer of West Indies tomatoes, had brought suit to lift a 10 percent tariff, mandated by Congress, on imported vegetables. Nix argued that he tomato is a fruit. The Court held that since a tomato was consumed as a vegetable rather than a desert like fruit, it was a vegetable.)" "Best Bite of Summer" by Denise Grady. *Self*, July 1997. Vol. 19 (7), pp. 12-125.

The heterogeneous nature of web sites makes it difficult to impose highly structured organization systems on the content. It doesn't make sense to classify documents at varying levels of granularity side by side. An article and a magazine should be treated differently. Similarly, it may not make sense to handle varying formats the same way. Each format will have uniquely important characteristics. For example, we need to know certain things about images such as file format (GIF, TIFF, etc.) and resolution (640x480, 1024x768, etc.). It is difficult and often misguided to attempt a one-size-fits-all approach to the organization of heterogeneous web site content.

Differences in Perspectives

Have you ever tried to find a file on a coworker's desktop computer? Perhaps you had permission. Perhaps you were engaged in low-grade corporate espionage. In any case, you needed that file. In some cases, you may have found the file immediately. In others, you may have searched for hours. The ways people organize and name files and directories on their computers can be maddeningly illogical. When questioned, they will often claim that their organization system makes perfect sense. "But it's obvious! I put current proposals in the folder labeled */office/clients/red* and old proposals in */office/clients/blue*. I don't understand why you couldn't find them!"

The fact is that labeling and organization systems are intensely affected by their creators perspectives. We see this at the corporate level with web sites organized according to internal divisions or org charts. In these web sites, we see groupings such as *marketing*, *sales*, *customer support*, *human resources*, and *information systems*. How does a customer visiting this web site know where to go for technical information about a product they just purchased? To design usable organization systems we need to escape from our own mental models of content labeling and organization.

You must put yourself into the shoes of the intended user. How do they see the information? What types of labels would they use? This challenge is further complicated by the fact that web sites are designed for multiple users, and all users will have different perspectives or ways of understanding the information. Their levels of familiarity, with your company and your web site vary. For these reasons, it is impossible to create a perfect organization system. One site does not fit all! However, by recognizing the importance of perspective and striving to understand the intended audiences, you can do a better job of organizing information for public consumption than your coworker on his or her desktop computer.

Internal Politics

Politics exist in every organization, individuals and departments constantly position for power or respect. Because of the inherent power of information

organization in forming understanding and opinion, the process of designing information architectures for web sites and intranets can involve a strong undercurrent of politics. The choice of organization and labeling systems can have a big impact on how users of the site perceive the company, its departments, and its products. For example, should we include a link to the library site on the main page of the corporate intranet? Should we call it *The Library* or *Information Services* or *Knowledge Management*? Should information resources provided by other departments be included in this area? If the library gets a link on the main page, then why not corporate communications? What about daily news?

As an information architect, you must be sensitive to your organization's political environment. In certain cases, you must remind your colleagues to focus on creating an architecture that works for the user. In others, you may need to make compromises to avoid serious political conflict. Politics raise the complexity and difficulty of creating usable information architectures. However, if you are sensitive to the political issues at hand, you can manage their impact upon the architecture.

Organizing Web Sites and Intranets

The organization of information in web sites and intranets is a major factor in determining success, and yet many web development teams lack the understanding necessary to do the job well. Our goal in this chapter is to provide a foundation for takling even the most challenging information organization projects.

Organization systems are composed of *organization schemes* and *organization structures*. An organization scheme defines the shared characteristics of content items and influences the logical grouping of those items. An organization structure defines the types of relationships between content items and groups.

Before diving in, it s important to understand information organization in the context of web site development. Organization is closely related to navigation. Labeling, and indexing. The hierarchical organization structures of web sites often play the part of primary navigation system. The labels of categories play a significant role in defining the contents of those categories. Manual indexing is ultimately a tool for organizing content items into groups at a very detailed level. Despite these closely knit relationships, it is both possible and useful to isolate the design of organization systems, which will form the foundation for navigation and labeling systems. By focusing solely on the logical grouping of information, you avoid the distractions of implementation details and design a better web site.

Organization Schemes

We navigate through organization schemes every day. Phone books, supermarkets, and television programming guides all use organization schemes to facilitate access. Some schemes are easy to use. We rarely have difficulty finding a friend's phone number in the alphabetical organization scheme of the white pages. Some schemes are intensely frustrating. Trying to find marshmallows or popcorn in a large and unfamiliar supermarket can drive us crazy. Are marshmallows in the snack aisle, the baking ingredients section, both, or neither?

In fact, the organization schemes of the phone book and the supermarket are fundamentally different. The alphabetical organization scheme of the phone book's white pages is exact. The hybrid topical/task-oriented organization scheme of the supermarket is ambiguous.

Exact organization schemes

Let's start with the easy ones. Exact organization schemes divide information into well defined and mutually exclusive sections. The alphabetical organization of the phone book's white pages is a perfect example. If you know the last name of the person you are looking for, navigating the scheme is easy. *Porter* is in the P's which is after the O's but before the Q's. This is called "known-item" searching. You know what you're looking for and it's obvious where to find it. No ambiguity is involved. The problem with exact organization schemes is that they require the user to know the specific name of the resource they are looking for. The white pages don't work very well if you're looking for a plumber.

Exact organization schemes are relatively easy, to design and maintain because there is little intellectual work involved in assigning items to categories. They are also easy to use. The following sections explore three frequently used exact organization schemes.

Alphabetical. An alphabetical organization scheme is the primary organization scheme for encyclopedias and dictionaries. Almost all nonfiction books, including this one, provide an alphabetical index. Phone books, department store directo-ries, bookstores, and libraries all make use of our 26-letter alphabet for organizing their contents. Alphabetical organization often serves as an umbrella for other organization schemes. We see information organized alphabetically by last name, by product or service, by department, and by format. See Figure 3-1 for an example.

Chronological. Certain types of information lend themselves to chronological organization. For example, an archive of press releases might be organized by the date of release (see Figure 3-2). History books, magazine archives, diaries, and television guides are organized chronologically. As long as there is agreement on when a particular event occurred, chronological schemes are easy to design and use.

Figure 3-1. An alphabetical index supports both rapid scanning for a known item and more casual browsing of a directory.

Geographical. Place is often an important characteristic of information. We travel from one place to another. We care about the news and weather that affects us in our location. Political, social, and economic issues are frequently location-dependent. With the exception of border disputes, geographical organization schemes are fairly straightforward to design and use. Figure 3-3 shows an example of a geographic organization scheme.

Figure 3-2. Press release are obvious candidates for chronological organization schemes. The date of announcement provides important context for the release. However, keep in mind that users may also want to browse the releases by title or search by keyword. A complementary combination schemes is often necessary.

Ambiguous organization scbemes

Now for the tough ones. Ambiguous organization schemes divide information into categories that defy exact definition. They are mired in the ambiguity of language and organization, not to mention human subjectivity. They are difficult to design and maintain. They can be difficult to use. Remember the tomato? Do we put it under fruit, berry, or vegetable?

Figure 3-3. In this example, the map presents a graphic organization scheme. Users can select a location from the map using their mouse.

However, they are often more important and useful than exact organization schemes. Consider the typical library catalog. There are three primary organization schemes. You can search for books by author, by title, or by subject. The author and title organization schemes are exact and thereby easier to create, maintain, and Lise. However, extensive research that library patrons use ambiguous subject-based schemes such as the Decimal and Library of Congress Classification Systems much more frequently.

There's a simple people find ambiguous organization schemes so useful: *We don't always know what we're looking for*. In some cases, you simply don't the correct label. In others, you may only have a vague information need that you can't quite articulate. For these reasons, information seeking is often iterative and interactive. What you find at the beginning of your search may influence what you look for and find later in your search. This information seeking process can involve a wonderful element of associative learning. Seek and ye shall find, but if the system is well-designed, you also might learn along the way. This is web surfing at its best.

Ambiguous organizatio supports this serendipitous mode of information seeking by grouping items in intellectually meaningful ways. In an alphabetical scheme, closely grouped items may have nothing in common beyond the fact that their names begin with the same letter. In an ambiguous organization scheme, someone other than the user has made n intellectual decision to group items together. This grouping of related items supports an associative learning process that may enable the user to make new connections and reach better conclusions. While ambiguous organization schemes require more work and introduce a messy element of subjectivity, they often prove more valuable to the user than exact schemes.

The success of ambiguous organization schemes depends on the initial design of a classification system and the ongoing indexing of content items. The

classification system serves as a structured container for content items. It is composed of a hierarchy of categories and subcategories with scope notes that define the types of content to be included under each category. Once this classification system has been created, content items must be assigned to categories accurately and consistently. This is a painstaking process that only a librarian could love. Let's review a few of the most common and valuable ambiguous organization schemes.

Topical. Organizing information by subject or topic is one of the most challenging yet useful approaches. Phone book yellow pages are organized topically. That's why they're the right place to look when you need a plumber. Academic courses and departments, newspapers, and the chapters of most nonfiction books are all organized along topical lines.

While few web sites should be organized solely by topic, most should provide some sort of topical access to content. In designing a topical organization scheme, it is important to define the breadth of coverage. Some schemes, such as those found in an encyclopaedia, cover the entire breadth of human knowledge (see Figure 3-4 for an example). Others, such as those more commonly found in corporate web sites, are limited in breadth, covering only those topics directly related to that company's products and services. In designing a topical organization scheme, keep in mind that you are defining the universe of content (both present and future) that users will expect to find within that area of the web site.

Figure 3-4. Research-oriented web sites such as the Argus Clearinghouse rely heavily on their topical organization scheme. In this example, the scope note for the Arts and Humanities categony is presented as well as the list of subcategories. This helps the user to understand the reasoning behind the inclusion or exclusion of specific subcategories.

Task-oriented. Task-oriented schemes organize content and applications into a collection of processes, functions, or tasks. These schemes are appropriate when it's possible to anticipate a limited number of high-priority tasks that users will want to perform. Desktop software applications such as word processors and

spreadsheets provide familiar examples. Collections of individual actions are organized under task-oriented menus such as *Edit*, *Insert*, and *Format*.

On today's Web, task-oriented organization schemes are less common, since most web sites are content rather than application intensive. This should change as sites become increasingly functional. Intranets and extranets lend themselves well to a task orientation, since they tend to integrate powerful applications as well as content. Figure 3-5 shows an example of a task-oriented site.

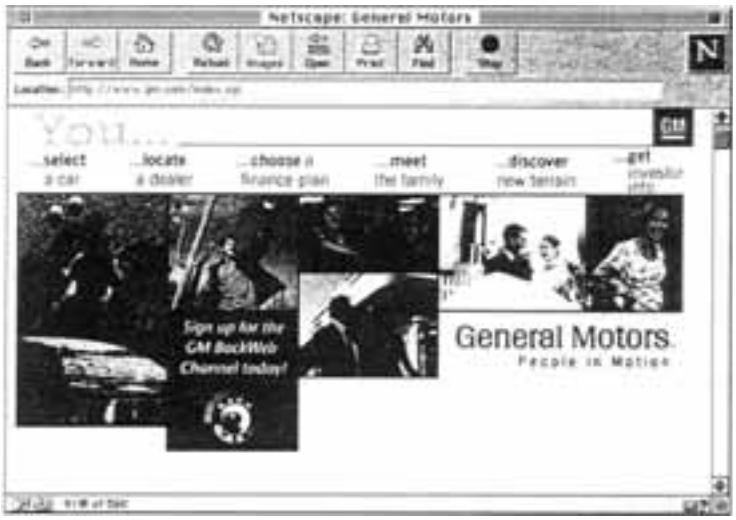


Figure 3-5. In this example, General Motors anticipates some of the most important needs of users by presenting a task-based menu of action items. This approach, enables GM to quickly funnel a diverse user base into specific action-oriented areas of the web site.

Audience-specific. In cases where there are two or more clearly definable audiences for a web site or intranet, an audience-specific organization scheme may make sense. This type of scheme works best when the site is frequented by repeat visitors who can bookmark their particular section of the site. Also, it works well if there is value in customizing the content for each audience. Audience-oriented schemes break a site into smaller, audience-specific minisites, thereby allowing for clutter-free pages that present only the options of interest to that particular audience. See Figure 3-6 for an example.

Audience-specific schemes can be open or closed. An open scheme will allow members of one audience to access the content intended for other audiences. A closed scheme will prevent members from moving between audience-specific sections. A closed scheme may be appropriate if subscription fees or security issues are involved.

Metaphor-driven. Metaphors are commonly used to help users understand the new by relating it to the familiar. You need not look further than your *desktop* computer with its *folders*, *files*, and *trash can* or *recycle bin* for an example. Applied to an interface in this way, metaphors can help users understand content and function intuitively. In addition, the process of exploring possible metaphor-driven organization schemes can generate new and exciting ide-

Figure 3-6. This area of the SIGGRAPH 97 conference web site is designed to meet the unique needs of media professionals covering the conference. Other SIGGRAPH audiences with special needs include contributors and exhibitors.

as about the design, organization, and function of the web site (see "Metaphor Exploration" in Chapter 8, *Conceptual Design*).

While metaphor exploration can be very useful while brainstorming, you should use caution when considering a metaphor-driven global organization scheme. First, metaphors, if they are to succeed, must be familiar to users. Organizing the web site of a computer hardware vendor according to the internal architecture of a computer will not help users who don't understand the layout of a motherboard.

Second, metaphors can introduce unwanted baggage or be limiting. For example, users might expect a virtual library to be staffed by a librarian that will answer reference questions.

Most virtual libraries do not provide this service. Additionally, you may wish to provide services in your virtual library that have no clear corollary in the real world. Creating your own customized version of the library is one such example. This will force you to break out of the metaphor, introducing inconsistency into your organization scheme.

Figure 3-7 shows a more offbeat metaphor example.

Hybrid schemes

The power of a pure organization scheme derives from its ability to suggest a simple mental model for users to quickly understand. Users easily recognize an audience-specific or topical organization. However, when you start blending elements of multiple schemes, confusion is almost guaranteed. Consider the example of a hybrid scheme in Figure 3-8. This hybrid scheme includes elements of audience-

Figure 3-7. In this offbeat example. Bianca has organized the contents of her web site according to the metaphor of a physical shack with rooms. While this metaphor-driven approach is fun and conveys a sense of place, it is not particularly intuitive. Can you guess what you'll find in the pantry? Also, note that features such as Find Your Friend don't fit neatly into the metaphor.

The Mix-Up Library

Adult	*audience-oriented*
Arts and Humanities	*topical*
Community Center	*metaphor-based*
Get a Library Card	*functional*
Learn About Our Library	*functional*
Science	*topical*
Social Science	*topical*
Teen	*audience-oriented*
Youth	*audience-oriented*

Figure 3-8. A hybrid organization scheme.

specific, topical, metaphor-based, and task-oriented organization schemes. Because they are all mixed together, we can't form a mental model, instead, we need to skim through each menu item to find the option we're looking for.

Examples of hybrid schemes are common on the Web. This happens because it is often difficult to agree upon any one scheme to present on the main page, so people throw the elements of multiple schemes together in a confusing mix. There is a better alternative. In cases where multiple schemes must be presented on one page, you should communicate to designers the importance of retaining the integrity of each scheme. As long as the schemes are presented separately on the page, they will retain the powerful ability to suggest a mental model for users (see Figure 3-9 for an example).

Organization Structures

Organization structure plays an intangible yet very important role in the design of sites. While we interact with organization structures every day,

Figure 3-9. Notice the audience-oriented scheme (contributors, exhibitors, media) has been presented as a pure organization scheme, separate from the others on this page. This approach allows you to present multiple organization schemes on the same page. This approach allows you to present multiple organization schemes on the same page without causing confusion.

we rarely think about them. Movies are linear in their physical structure. We experience them frame by frame from beginning to end. However, the plots themselves may be non-linear, employing flashbacks and parallel subplots. Maps have a spatial structure. Items are placed according to physical proximity, although the most useful maps cheat, sacrificing accuracy for clarity.

The structure of information defines the primary ways in which users can navigate. Major organization structures that apply to web site and intranet architectures include the hierarchy, the database-oriented model, and hypertext. Each organization structure possesses unique strengths and weaknesses. In some cases, it makes sense to use one or the other. In many cases, it makes sense to use all three in a complementary manner.

The hierarchy: A top-down approach

The foundation of almost all good information architectures is a well-designed hierarchy. In this hypertextual world of nets and webs, such a statement may seem blasphemous, but it's true. The mutually exclusive subdivisions and parent-child relationships of hierarchies are simple and familiar. We have organized information into hierarchies since the beginning of time. Family trees are hierarchical. Our division of life on earth into kingdoms and classes and species is hierarchical. Organization charts are usually hierarchical. We divide books into chapters into sections into paragraphs into sentences into words into letters. Hierarchy is ubiquitous in our lives and informs our understanding of the world in a profound and meaningful way. Because of this pervasiveness of hierarchy, users can easily and quickly understand web sites that use hierarchical organization models. They are able to develop a mental model of the site's structure and their location within that

structure. This provides context that helps users feel comfortable. See Figure 3-10 for an example of a simple hierarchical model.

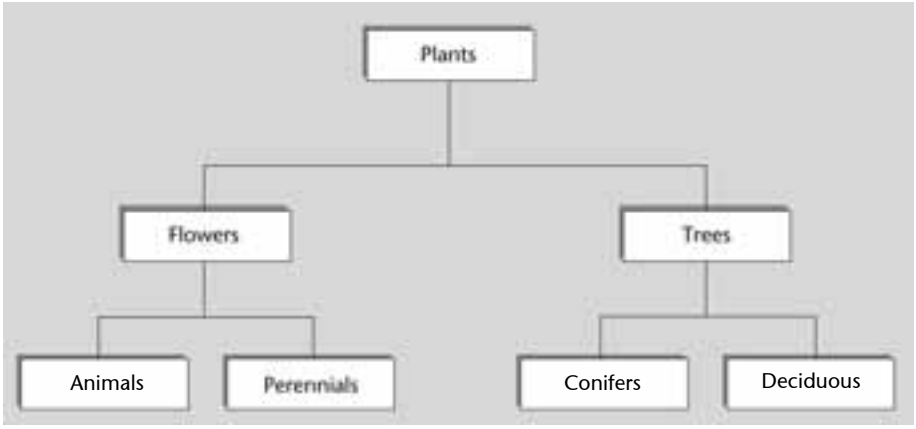


Figure 3-10. A simple hierarchical organization model.

Because hierarchies provide a simple and familiar way to organize information, they are usually a good place to start the information architecture process. The top-down approach allows you to quickly get a handle on the scope of the web site without going through an extensive content inventory process. You can begin identifying the major content areas and exploring possible organization schemes that will provide access to that content.

Designing hierarchies

When designing information hierarchies on the Web, you should remember a few rules of thumb. First, you should be aware of, but not bound by, the idea that hierarchical categories should be mutually exclusive. Within a single organization scheme, you will need to balance the tension between exclusivity and inclusivity. Ambiguous organization schemes in particular make it challenging to divide content into mutually exclusive categories. Do tomatoes belong in the fruit or vegetable or berry category? In many cases, you might place the more ambiguous items into two or more categories, so that users are sure to find them. However, if too many items are cross-listed, the hierarchy loses its value. This tension between exclusivity and inclusivity does not exist across different organization schemes. You would expect a listing of products organized by format to include the same items as a companion listing of products organized by topic. Topic and format are simply two different ways of looking at the *same* information.

Second, it is important to consider the balance between breadth and depth in your information hierarchy. Breadth refers to the number of options at each level of the hierarchy. Depth refers to the number of levels in the hierarchy. If a hierarchy is too narrow and deep, users have to click through an inordinate number of levels to find what they are looking for (see Figure 3-11). If a hierarchy is too broad and shallow, users are faced with too many options on the main menu and are unpleasantly surprised by the lack of content once they select an option.

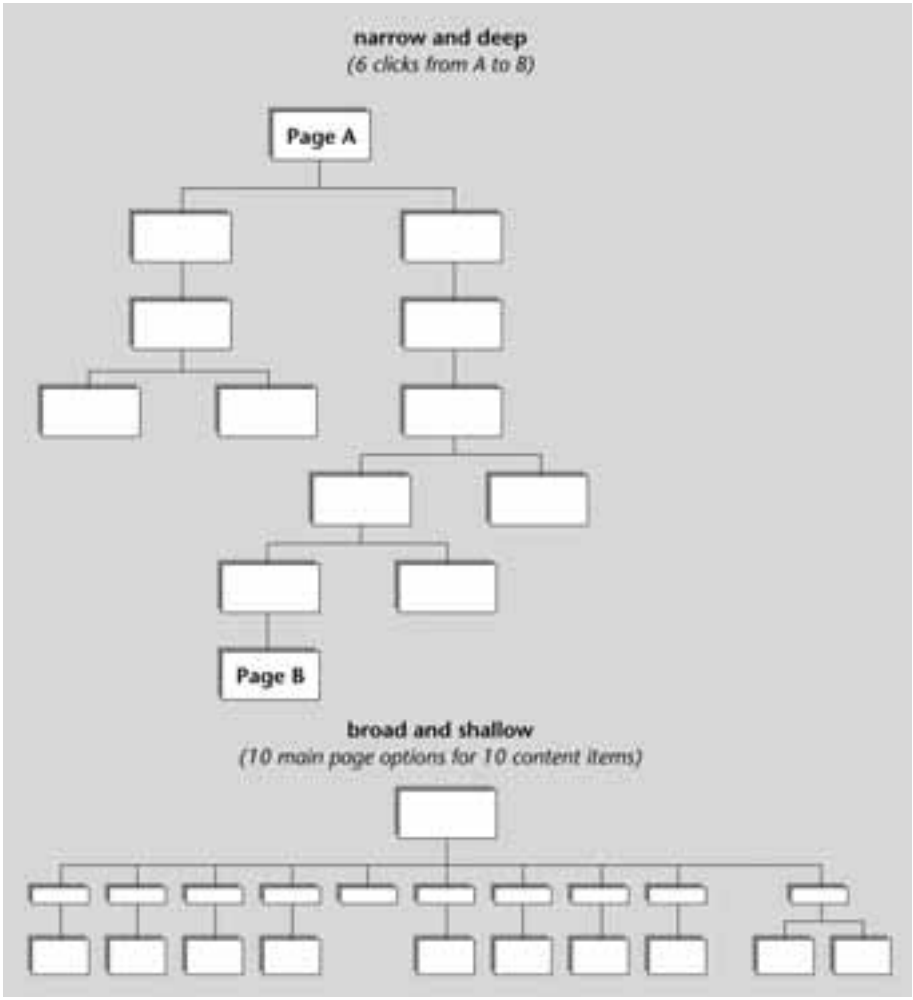


Figure 3-11. In the narrow and deep hierarchy, users are faced with six clicks to reach the deepest content. In the broad and shallow hierarchy, users must choose from ten options to reach a limited amount of content.

In considering breadth, you should be sensitive to the cognitive limits of the human mind. Particularly with ambiguous organization schemes, try to follow the seven plus-or-minus two rule.† Web sites with more than ten options on the main menu can overwhelm users.

In considering depth, you should be even more conservative. If users are forced to click through more than four or five levels, they may simply give up and leave your web site. At the very least, they'll become frustrated.

For new web sites and intranets that are expected to grow, you should lean towards a broad and shallow rather than narrow and deep hierarchy. This approach allows for the addition of content without major restructuring. It is less problematic to add items to secondary levels of the hierarchy than to the main page, for a couple of reasons. First, the main page serves as the most prominent and important navigation interface for users. Changes to this page can really hurt the mental model they have formed of the web site

† **G. Miller**. "The Magical Number Seven, plus or Minus Two: Some Limits on our Capacity for Processing Information". *Psychological Review* 63. Nº 2 (1956): 81-97.

over time. Second, because of its prominence and importance, companies tend to spend lots of care (and money) on the graphic design and layout of the main page. Changes to the main page can be more time consuming and expensive than changes to secondary pages.

Finally, when designing organization structures, you should not become trapped by, the hierarchical model. Certain content areas will invite a database or hypertext-based approach. The hierarchy is a good place to begin, but is only one component in a cohesive organization system.

Hypertext

Hypertext is a relatively new and highly nonlinear way of structuring information. A hypertext system involves two primary types of components: the items or chunks of information which are to be linked, and the links between those chunks. These components can form hypermedia systems that connect text, data, image, video, and audio chunks. Hypertext chunks can be connected hierarchically, non-hierarchically, or both (see Figure 3-12).

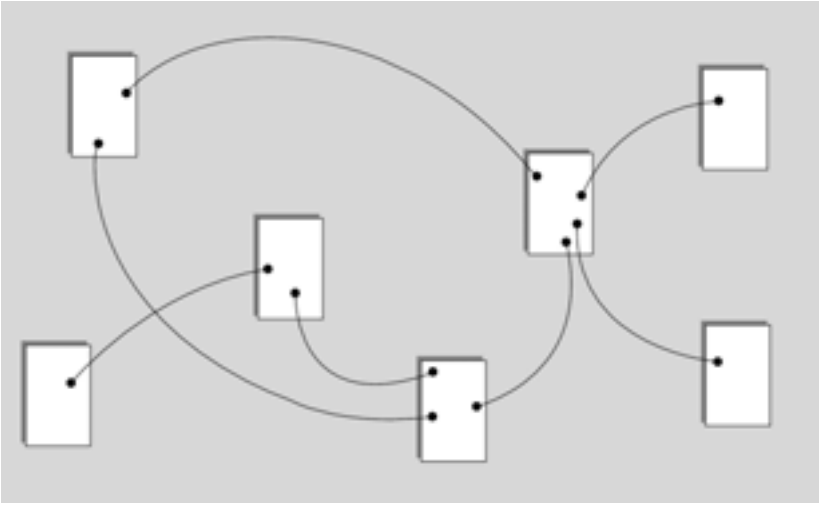


Figure 3-12. In hypertext systems, content chunks are connected via links in a loose web of relationships.

Although this organization structure provides you with great flexibility, it presents substantial potential for complexity and user confusion. As users navigate through highly hypertextual web sites, it is easy for them to get lost. It's as if they are thrown into a forest and are bouncing from tree to tree, trying to understand the lay of the land. They simply can’t create a mental model of the site organization. Without context, users can quickly become overwhelmed and frustrated. In addition, hypertextual links are often personal in nature. The relationships that one person sees between content items may not be apparent to others.

For these reasons, hypertext is rarely a good candidate for the primary organization structure. Rather, hypertext can be used to complement structures based upon the hierarchical or database models.

Hypertext allows for useful and creative relationships between items and areas in the hierarchy. It usually makes sense to first design the information hierarchy and then to identify ways in which hypertext can complement the hierarchy.

The relational database model: A bottom-up approach

Most of us are familiar with databases. In fact, our names, addresses, and other personal information are included in more databases than we care to imagine. A database is a collection of records. Each record has a number of associated fields. For example, a customer database may have one record per customer. Each record may include fields such as customer name, street address, city, state, ZIP code, and phone number. The database enables users to search for a particular customer or to search for all users with a specific ZIP code. This powerful field-specific searching is a major advantage of the database model. Additionally, content management is substantially easier with a database than without. Databases can be designed to support time-saving features such as global search and replace and data validation. They can also facilitate distributed content management, employing security measures and version control systems that allow many people to modify content without stepping on each others' toes.

Finally, databases enable you to repurpose the same content in multiple forms and formats for different audiences. For example, an audience-oriented approach might benefit from a context-sensitive navigation scheme in which each audience has unique navigation options (such as returning to the main page of that audience area). Without a database, you might need to create a separate version of each HTML page that has content shared across multiple audiences. This is a production and maintenance nightmare! In another scenario, you might want to publish the same content to your web site, to a printed brochure, and to a CDROM. The database approach supports this flexibility.

However, the database model has limitations. The records must follow rigid rules. Within a particular record type, each record must have the same fields, and within each field, the formatting rules must be applied consistently across records. This highly structured approach does not work well with the heterogeneous content of many web sites. Also, technically it's not easy to place the entire contents (including text, graphics, and hypertext links) of every HTML page into a database. Such an approach can be very expensive and time consuming.

For these reasons, the database model is best applied to subsites or collections of structured, homogeneous information within a broader web site. For example, staff directories, news release archives, and product catalogs are excellent candidates for the database model.

Designing databases

Typically, the top-down process of hierarchy design will uncover content areas that lend themselves to a database-driven solution. At this point, you

will do well to involve a programmer, who can help not only with the database implementation but with the nitty-gritty data modeling issues as well (see Figure 3-13).

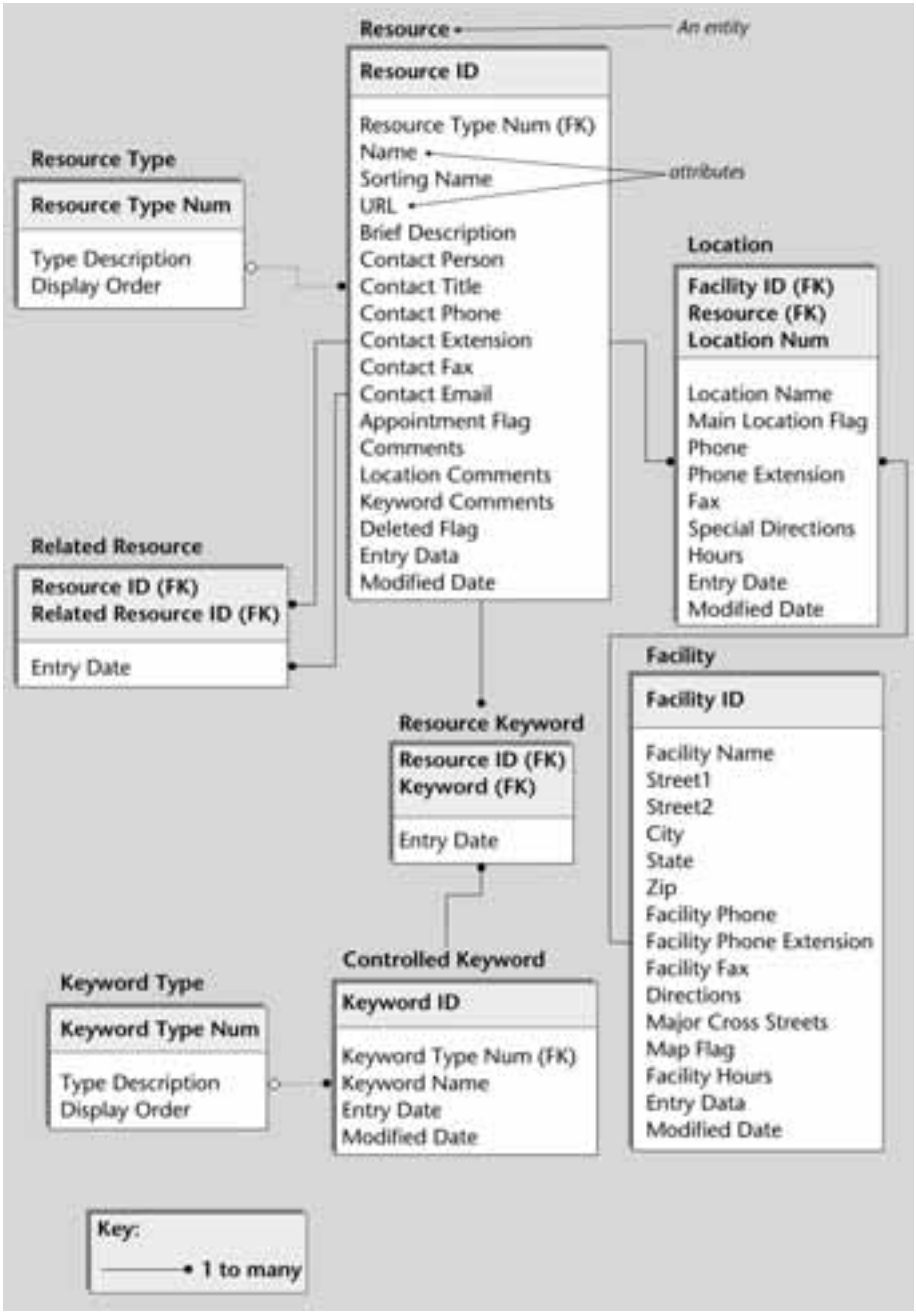


Figure 3-13. This entity relationship diagram (ERD) shows a structured approach to database design. We see that entities (e.g. Resource) have attributes (e.g., Name, URL). Ultimately entities and attributes become records and fields in the database. An ERD also shows relationships between entities. For example, we see that each resource is available at one or more locations. The ERD is used to visualize and refine the data model, before design and population of the database. (This entry relationship diagram courtesy of InterConnect of Ann Arbor, a technical consulting and development firm.)

Within each of the content areas identified as candidates for a database-driven solution, you will need to begin a bottom-up approach aimed at identifying the content and structure of individual record types.

For example, a staff directory may have one record for each staff member. You will need to identify what information will be made available for each individual. Some fields such as name and office phone number may be required. Others such as email address and home phone number may be optional. You may decide to include an expertise field that includes keywords to describe the skills of that individual. For fields such as this, you will need to determine whether or not to define a controlled vocabulary.

A controlled vocabulary specifies the acceptable terms for use in a particular field. It may also employ scope notes that define each term.

For example, the table below lists the controlled vocabulary for keywords in the ecology area of the Argus Clearinghouse web site (see *http://www.clearing-house.net*). The scope notes explain that ecology is "the branch of biology dealing with the relation of living things to their environments." (See Figure 5-2 for an example of scope notes in action.) This information is useful for the staff who index resources and the users who navigate the web site.

Controlled Vocabulary **Argus Clearinghouse: Environment: Ecology**	
biodiversity	coastal zone management
conservation	ecology (general)
environment	environmental health
environmental resources	environmental science
environmental studies	land use
reef conservation	roadkill
water resources	wetlands conservation
wildlife	wildlife management
wildlife rehabilitation	

Use of a controlled vocabulary imposes an important degree of consistency that supports searching and browsing. Once users understand the controlled vocabulary, they know that a search on *biodiversity* should retrieve all relevant documents. They do not also need to try *biological diversity*. In addition, this consistency allows you to automatically generate browsable indexes. This is a great feature for users, is not very difficult to implement, and is extremely efficient from a site maintenance perspective (see Figure 3-14).

However, creating and maintaining a controlled vocabulary is not a simple task. In many cases, complementing a simple controlled vocabulary that divides the items into broad categories with an uncontrolled keyword field provides a good balance of structure and flexibility. (For more on creating controlled vocabularies, see "Controlled vocabularies and thesauri" in Chapter 5.)

Once you've constructed the record types and associated controlled vocabularies, you can begin thinking about how users should be able to navigate this information. One of the major advantages of a database-driven approach is the power and flexibility it affords for the design of searching and browsing

Figure 3-14. You can leverage a controlled vocabulary, to automatically generate browsable indexes. In this example, after selecting Environmental Health from a menu of acceptable terms in the Ecology category, the user is presented with a list of relevant resources. These resources have been manually, indexed according to the controlled vocabulary.

Figure 3-15. A database of organizational resources brings power and flexibility to the Henry Ford Health System web site. Users can browse by organizational resource or keyword, or perform a search against the collection of records. The browsing indexes and the records themselves are generated from the database. Site-wide changes can be made at the press of a button. This flexibility is made possible by a database-driven approach to content organization and management.

systems (see Figure 3-15). Every field presents an additional way to browse or search the directory of records.

The database-driven approach also brings greater efficiency and accuracy to data entry and content management. You can create administrative interfaces that eliminate worry about HTML tags and ensure standard formatting across records through the use of templates. You can integrate tools that perform syntax and link checking. Of course, the search and browse indexes can be re-built automatically after each addition, deletion, or modification.

Content databases can be implemented in a variety of ways. The database management software can be configured to produce static HTML pages in

batch mode or to generate dynamic HTML pages on-the-fly as users navigate the site. These implementation decisions will be influenced by technical performance issues (e.g., bandwidth and CPU constraints) and have little impact upon the architecture.

Creating Cohesive Organization Systems

As you've seen in this chapter, organization systems are fairly complex. You need to consider a variety of exact and ambiguous organization schemes. Should you organize by topic, by task, or by audience? How about a chronological or geographical scheme? What about using multiple organization schemes?

You also need to think about the organization structures that influence how users can navigate through these schemes. Should you use a hierarchy or would a more structured database-model work best? Perhaps a loose hypertextual web would allow the most flexibility? Taken together, in the context of a large web site development project, these questions can be overwhelming. That's why it's important to break down the site into its components, so you can tackle one question at a time. Also, keep in mind that all information retrieval systems work best when applied to narrow domains of homogeneous content. By decomposing the content collection into these narrow domains, you can identify opportunities for highly effective organization systems.

However, it's also important not to lose sight of the big picture. As with cooking, you need to mix the right ingredients in the right way to get the desired results. Just because you like mushrooms and pancakes doesn't mean they will go well together. The recipe for cohesive organization systems varies from site to site. However, there are a few guidelines to keep in mind.

In considering which organization schemes to use, remember the distinction between exact and ambiguous schemes. Exact schemes are best for known-item searching, when users know precisely what they are looking for. Ambiguous schemes are best for browsing and associative learning, when users have a vaguely defined information need. Whenever possible, use both types of schemes. Also, be aware of the challenges of organizing information on the Web. Language is ambiguous, content is heterogeneous, people have different perspectives, and politics can rear its ugly head. Providing multiple ways to access the same information can help to deal with all of these challenges.

When thinking about which organization structures to use, keep in mind that large web sites and intranets typically require all three types of structure. The top-level, umbrella architecture for the site will almost certainly be hierarchi-

cal. As you are designing this hierarchy, keep a lookout for collections of structured, homogeneous information. These potential subsites are excellent candidates for the database model. Finally, remember that less structured, creative relationships between content items can be handled through hypertext. In this way, all three organization structures together can create a cohesive organization system.

Designing Navigation Systems

Louis Rosendfeld
Peter Morville

Just wait, Gretel, until the moon rises, and then we shall see the crumbs of bread which I have strewn about. they will show us our way home again.
Hansel and Gretel

In this chapter:

- Browser Navigation Features
- Building Context
- Improving Flexibility
- Types of Navigation Systems
- Integrated Navigation Elements
- Remote Navigation Elements
- Designing Elegant Navigation Systems

As our fairy tales suggest, getting lost is often a bad thing. It is associated with confusion, frustration, anger, and fear. In response to this danger, we have developed navigation tools to prevent people from getting lost. From bread crumbs to compass and astrolabe to maps, street signs, and global positioning systems, people have demonstrated great ingenuity in the design and use of navigation tools.

We use them to chart our course, to determine our position, and to find our way back. They provide a sense of context and comfort as we explore new places. Anyone who has driven through an unfamiliar city as darkness falls understands the importance that navigation tools play in our lives.

On the Web, navigation is rarely a life or death issue. However, getting lost in a large web site can be confusing and frustrating. While a well-designed hierarchical organization scheme will reduce the likelihood that users will become lost, a complementary navigation system is often needed to provide context and to allow for greater flexibility of movement within the site.

Navigation systems can be designed to support associative learning by featuring resources that are related to the content currently being displayed. For example, a page that describes a product may include *see also* links to related products and services (this type of navigation can also support a company's marketing goals). As users move through a well-designed navigation system, they learn about products, services, or topics associated to the specific content they set out to find.

Any page on a web site may have numerous opportunities for interesting *see also* connections to other areas of the site. The constant challenge in naviga-

tion system design is to balance this flexibility of movement with the danger of overwhelming the user with too many options.

Navigation systems are composed of a variety of elements. Some, such as graphical navigation bars and pop-up menus, are implemented on the content-bearing pages themselves. Others, such as tables of contents and site maps, provide remote access to content within the organization structure. While these elements may be implemented on each page, together they make up a navigation system that has important site-wide implications. A well-designed navigation system is a critical factor in determining the success of your web site.

Browser Navigation Features

When designing a navigation system, it is important to consider the environment the system will exist in. On the Web, people use web browsers such as Netscape Navigator and Microsoft Internet Explorer to move around and view web sites. These browsers sport many built-in navigation features.

Open URL allows direct access to any page on a web site. *Back* and *Forward* provide a bidirectional backtracking capability. The *History* menu allows random access to pages visited during the current session, and *Bookmark* enables users to save the location of specific pages for future reference. Web browsers also go beyond the Back button to support a "bread crumbs" feature by color-coding hypertext links. By default, unvisited hypertext links are one color and visited hypertext links are another. This feature helps users understand where they have and haven't been and can help them to retrace their steps through a web site.

Finally, web browsers allow for a prospective view that can influence how users navigate. As the user passes the cursor over a hypertext link, the destination URL appears at the bottom of the browser window, ideally hinting about the nature of that content (see Figure 4-1). If flies and directories have been carefully labeled, prospective view gives the user context within the content hierarchy. If the hypertext link leads to another web site on another server, prospective view provides the user with basic information about this off-site destination.

Much research, analysis, and testing has been invested in the design of these browser-based navigation features. However, it is remarkable how frequently site designers unwittingly override or corrupt these navigation features. For example, designers often modify the unvisited and visited link colors with no consideration for the bread crumbs feature. They focus on aesthetics, attempting to match link colors with logo colors. It's common to see a complete reversal of the blue and purple standard.

Figure 4-1. In this example, the cursor is positioned over the Investor Info button. 7be prospective view window at the bottom shows the URL of the Investor Info page.

This is a classic sacrifice of usability* for aesthetics and belies a lack of consideration for the user and the environment. It's like putting up a green stop sign at a road intersection because it matches the color of a nearby building.

Given proper understanding of the aesthetic and usability issues, you can in fact modify the link colors and create an intelligent balance† Unfortunately, this convention has been violated so frequently, the standard may no longer be standard.

A second common example of inadvertently disabling valuable browser navigation features involves prospective view. Image maps have become a ubiquitous navigation feature on web sites. The graphic navigation bar allows the aesthetically pleasing presentation of navigation options. Unfortunately, server-side image maps completely disable the prospective view feature of web browsers. Instead of the destination URL preview, the XY coordinates of the image map are presented. This information is distracting, not useful. Again, a solution that balances aesthetics and usability is available. Through an elegant use of tables (or by using client-side image maps), you can present a graphical navigation bar that leverages the browser-based prospective view feature.

Once you are sensitive to the built-in navigation features of web browsers, it is easy to avoid disabling or duplicating those features. In fact, it is both possible and desirable to find ways to leverage them. In designing navigation systems, you should consider all elements of that system. Web browsers are an extremely common and integral part of the user's navigation experience. From a philosophical perspective, we might say that web pages do not exist in the absence of a web browser. So, don't override or corrupt the browser!

---

* Analysis of a usability test that explored the impact of graphic design on users' ability to find information lead to the following conclusion: "Of all the graphic design elements we looked at, the only one that is strongly tied to user success was the use of browser-default link color.... Our theory is that use of the default colors is helpful because users don't have to relearn every time they go to a new site." Jared Spool et al., *Web Site Usability* (Andover, M.A: User Interface Engineering, 1997).

† For an example, see Michigan Comnet *http://cmnet.org/*. The link colors have been modified slightly to match the logo colors, but the red:purple/unvisited link standard is maintained.

Building Context

With all navigation systems, before we can plot our course, we must locate our position. Whether we're visiting Yellowstone National Park or the Mall of America, the *You Are Here* mark on fixed-location maps is a familiar and valuable tool. Without that landmark, we must struggle to triangulate our current position using less dependable features such as street signs or nearby stores. The *You Are Here* indicator can make all the difference between knowing where you stand and feeling completely lost.

In designing complex web sites, it is particularly important to provide context within the greater whole. Many contextual clues in the physical world do not exist on the Web. There are no natural landmarks and no north and south. Unlike physical travel, hypertextual navigation allows users to be transported right into the middle of a large unfamiliar web site. Links from remote web pages and search engine result pages allow users to completely bypass the front door or main page of the web site. To further complicate matters, people often print web pages to read later or to pass along to a colleague, resulting in even more loss of context.

You should always follow a few rules of thumb to ensure that your sites provide contextual clues. First, all pages should include the organization's name. This might be done as part of the title or header of the page. As a user moves through the levels of a site, it should be clear that they are still within that site. Carrying the graphic identity throughout the site supports such context and consistency. In addition, if a user bypasses the front door and directly accesses a subsidiary page of the site, it should be clear which site lie or she is on.

Second, the navigation system should present the structure of the information hierarchy in a clear and consistent manner and indicate the location within that hierarchy. See Figure 4-2 for an example.

Figure 4-2. The navigation system for the Argus Clearinghouse clearly shows the path the user has taken through the hierarchy and indicates the user's current location. This helps the user to build a mental model of the organization scheme that facilitates navigation and helps them feel comfortable.

Improving Flexibility

As discussed in the previous chapter, hierarchy is a familiar and powerful way of organizing information. In many cases, it makes sense for a hierarchy to form the foundation for organizing content in a web site. However, hierarchies can be fairly limiting from a navigation perspective. If you have ever used the ancient information browsing technology and precursor to the World Wide Web known as Gopher, you will understand the limitations of hierarchical navigation. In Gopherspace, you were forced to move up and down the tree structures of content hierarchies (see Figure 4-3). It was not practical to encourage or even allow camps across branches (lateral navigation) or between multiple levels (vertical navigation) of a hierarchy.

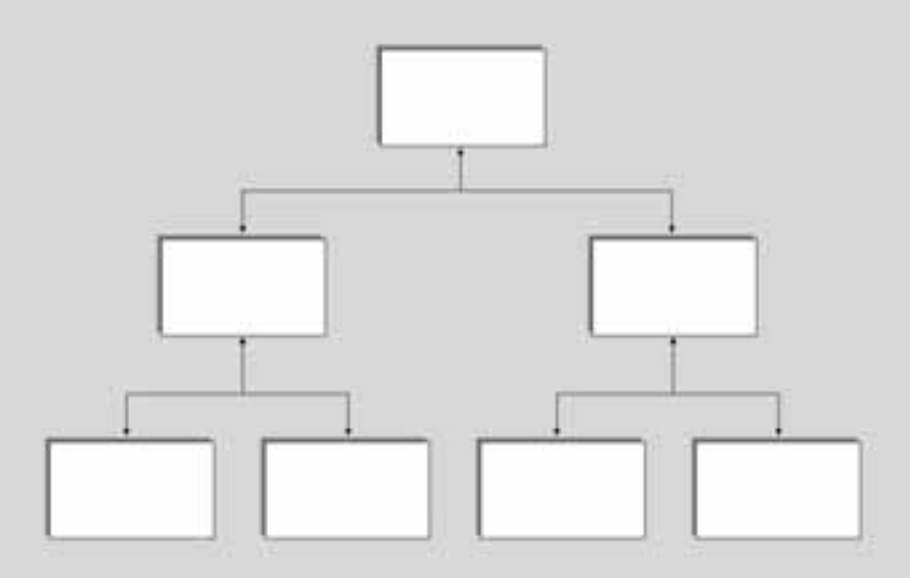


Figure 4-3. On a Gopher site, you could only, move up or down through the tree structure of the hierarchy.

The Web's hypertextual capabilities removed these limitations, allowing tremendous freedom of navigation. Hypertext supports both lateral and vertical navigation (see Figure 4-4). From any branch of the hierarchy, it is possible and often desirable to allow users to laterally move into other branches. For example, as you explore the Programs and Events section of a conference web site, you may decide to register for that conference. A hypertext link should allow you to jump to Registration without first retracing your steps back up the Programs and Events hierarchy.

It is also possible and often desirable to allow users to move vertically from one level in a branch to a higher level in that same branch (e.g., from a specific Program back to the main Programs and Events page) or all the way back to the main page of the web site.

The trick with designing navigation systems is to balance the advantages of flexibility with the dangers of clutter. In a large, complex web site, the com-

plete lack of lateral and vertical navigation aids can be very limiting. On the other hand, too many navigation aids can bury the hierarchy and overwhelm the user. Navigation systems should be designed with care to complement and reinforce the hierarchy by providing added context and flexibility.

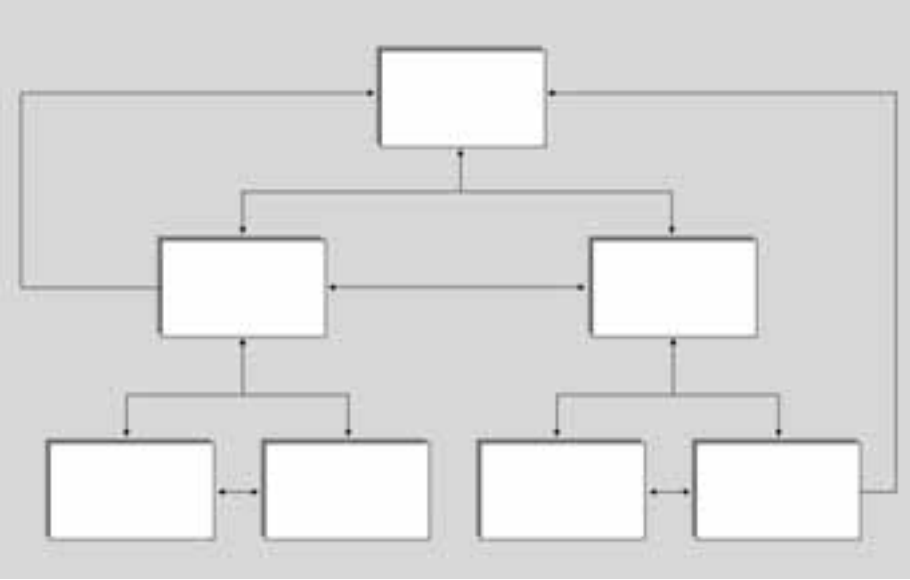


Figure 4-4. In a hypertext system, navigation links can completely bypass the hierarchy. You can enable users to get anywhere from anywhere. However as you can see from this diagram, things can get confusing pretty quickly. It begins to look like an architecture from M. C. Escher.

Types of Navigation Systems

A complex web site often includes several types of navigation systems. To design a successful site, it is essential to understand the types of systems and how they work together to provide flexibility and context.

Hierarchical Navigation Systems

Although we may not typically think of it this way, the information hierarchy is the primary navigation system. From the main page to the destination pages that house the actual content, the main options on each page are taken directly from the hierarchy (see Figure 4-5). As noted earlier, the hierarchy is extremely important, but also rather limiting. It is these limitations that often require additional navigation systems.

Global Navigation Systems

A global or site-wide navigation system often complements the information hierarchy by enabling greater vertical and lateral movement throughout the entire site. At the heart of most global navigation systems are some standard rules that dictate the implementation of the system at each level of the site.

Figure 4-5. The six options at the bottom of the MVAC homepage constitute the top level of the hierarchical organization scheme. Note that these icons combine text and images so that users don't have to guess what's hidden behind each. In addition, the imagery and associated color schemes can be repeated appropriately through out the web site, providing both context and consistency.

The simplest global navigation system might consist of a graphical navigation bar at the bottom of each page on the site. On the main page, the bar might be unnecessary, since it would duplicate the primary options already listed on that page. On second level pages, the bar might include a link back to the home page and a link to the feedback facility, as in Figure 4-6.

A slightly more complex global navigation system may provide for area-specific links on third level pages and below. For example, if a user explores the products area of the web site, the navigation bar could include *Main Page*, *Products*, and *Search*. The obvious exception to this rule-based system is that pages should not include navigation links to themselves. For example, the main page of the products area should not include a *Products* link. However, this is a great opportunity for the site's graphic designer to devise the navigation bar to show that you are currently on the main page of the products area. Designers often leverage a folder tab or button metaphor to accomplish this effect. (On the Argus web site, we use the @ sign from our corporate logo, as seen in Figure 4-7.)

Figure 4-6. The MVAC Web site employs a very simple, icon-based global navigtion system.

Figure 4-7. For the Argus web site, graphic designers from Q LTD came up with a creative and elegant solution to show context within the navigation system by leveraging the @ sign from our corporate logo. In this example, the @ sign indicates that the Publications page is within the What We Do area.

As you can see, this type of rule-based global navigation system can easily be applied throughout the entire web site. The navigation system and the graphic design system should be integrated to provide both flexibility and context. Note that the relative locations of the options should remain the same from one version of the bar to another and that, since people read from left to right, *Main Page* should be to the left of the other options. Both these factors enhance the context within the hierarchy.

Local Navigation Systems

For a more complex web site, it may be necessary to complement the global navigation system with one or more local navigation systems. To understand the need for local navigation systems, it is necessary to understand the concept of a *subsite*‡. The term sub-site was coined by Jakob Nielsen to identify the recurrent situation in which a collection of web pages within a larger site invite a common style and shared navigation mechanism unique to those pages.

For example, a software company may provide an online product catalog as one area in their web site. This product catalog constitutes a sub-site within the larger web site of the software company. Within this sub-site area, it makes sense to provide navigation options unique to the product catalog, such as browsing products by name or format or market.

However, it is also important to extend the global navigation system throughout the sub-site. Users should still be able to jump back to the main page or

‡ **Jakob Nielsen**. *The Rise of the Sub-Site*. Sept. 1996 (*http://www.useit.com alertbox/9609.html*).

provide feedback. Local navigation systems should be designed to complement rather than replace the global navigation system (see Figure 4-8).

Figure 4-8. In this example, the bulleted options are pall of a simple local navigation system that guides users through information about the Digital Dissertations project. The graphical buttons at the lower left of the page are part of the global navigation system.

This integration can be challenging, particularly when the global and local navigation systems provide too many options. Alone they may each be manageable, but together on one page, the variety of options may overwhelm the user. In some cases, you may need to revisit the number of global and local navigation options. In others, the problem may be minimized through elegant page design.

Ad Hoc Navigation

Relationships between content items do not always fit neatly into the categories of hierarchical, global, and local navigation. An additional category of *ad hoc* links is more editorial than architectural. Typically an editor or content specialist will determine appropriate places for these types of links once the content has been placed into the architectural framework of the web site. In practice, this usually involves representing words or phrases within sentences or paragraphs (i.e., prose) as embedded hypertext links. This approach can be problematic if these ad hoc links are important, since usability testing shows "a strong negative correlation between embedded links (those surrounded by text) and user success in finding information"[**]. Apparently, users tend to scan pages so quickly that they often miss these less conspicuous links. You can replace or complement the embedded link approach with external links that are easier for the user to see.

Embedded Links

As you can see, embedded links are surrounded by text.
Users often miss these links.
One Solution to the Embedded Link Problem is to give links their own separate lines within the paragraph.
Another solution is to create a separate menu of ad hoc links at the top or bottom of the page that point to useful related resources:

- Embedded Links
- Users
- One Solution to the Embedded Link Problem

[**] Spool et al., 41-42.

The approach you use should be determined by the nature and importance of the ad hoc links. For non-critical links provided as a point of interest, embedded links can be an elegant, unobtrusive solution.

When using ad hoc links, it's important to consider whether the linked phrase provides enough context for the user. In Figure 4-9, it's fairly obvious where the Digital Dissertations Pilot Site link will take you. However, if 1861 or 1997 were underlined, you would be hard pressed to guess where those links would lead. In designing navigation systems for the Web, context is king.

Figure 4-9. Moderation is the primary rule of thumb for guiding the creation of embedded ad hoc links. Used sparingly, (as in this example), they can complement the existing navigation systems by adding one more degree of flexibility. Used in excess, ad hoc links can add clutter and confusion.

Integrated Navigation Elements

In global and local navigation systems, the most common and important navigation elements are those that are integrated into the content-bearing pages of the web site. As users move through the site or sub-site, these are the elements they see and use again and again. Most integrated navigation elements fit into one of two categories: navigation bars and pull-down menus.

Navigation Bars

You can implement navigation bars in many ways and use them for the hierarchical, global, and local navigation systems. In simplest form, a navigation bar is a collection of hypertext links grouped together on a page. Alternatively, the navigation bar may be graphical in nature, implemented as an image map or as graphic images within a table structure.

The decision to use text versus graphic navigation bars falls primarily within the realms of graphic design and technical performance rather than information architecture. Graphic navigation bars tend to look nicer but can significantly slow down the page loading speed (although, if you're able to reuse the same global navigation bar throughout the site, loading speed will only be hurt once, since the image will be cached locally). If you do use graphic nav-

igation bars, you need to be sensitive to the needs of users with low bandwidth connections. You should also consider those users with text-only browsers (there are still quite a few out there) and those users with high-end browsers who turn off the graphical capabilities to get around more quickly. Appropriate use of the <ALT> attribute to define replacement text for the image will ensure that your site supports navigation for these users.

However, key issues related to the architecture should also influence this decision. For example, it is usually much easier to add options to a text menu than a graphic-based menu. If you anticipate substantial growth or change in a particular area, it may make sense to employ a textual navigation bar, like the one in Figure 4-10. Cost is also an issue, since graphic navigation bars require more work to create and change than text-based bars. In many cases, you might employ a graphic bar for global navigation and a textual menu for local navigation. A good graphic designer will strike an elegant balance between form and function in creating these navigation bars.

It is often best to place the navigation bar towards the top and/or bottom of the page, rather than at the side[††]. Placement at the top provides immediate access to the navigation system as well as an instant sense of context within the site. This supports the scenario in which a user quickly scans the first paragraph and decides to move on to other areas of the site. Placement at the bottom assumes navigation once the page has been fully read. Placement at both the top and bottom should be determined by the length of the content.

Graphical navigation bars may employ several techniques for conveying content and context, including textual labels and icons. Textual labels are the easiest to create and by far most clearly indicate the contents of each option. Icons, on the other hand, are relatively difficult to create and often fail to indicate the contents of each option. It's difficult to represent abstract concepts through images. A picture may say a thousand words, but often they're the wrong words. Icons can successfully be used to complement the textual labels. Since repeat users may become so familiar with the icons that they no longer take the time to read the textual labels, icons are useful in facilitating rapid menu selection for them. See Figure 4-11 for an example.

However, hidden minefields may plague an iconic system. First, the Internet's global nature introduces the potential for confusion or even anger, since an image may have very different meanings from one culture to another. Second, the iconic system may work well for a limited number of menu options, but if the decision is made to add one or more options, creating an appropriate icon can be very challenging. While icons certainly work well sometimes, the skillful use of a color system can facilitate rapid menu selection without

[††] One usability study showed that "Sites with navigation buttons or links at the top and bottom of pages did slightly better than sites with navigation buttons down the side of the page." Spool et al., 24.

the inherent problems of iconic systems. (For more about the use of icons, see Chapter 5, *Labeling Systems*.

Figure 4-10. C/ Net provides a high-profile example of the use of text-based navigation options.

Figure 4-11. This navigation bar, which appears at the bottom of the page, demonstrates an interesting blend of graphic icons (with labels) and textual options. The global navigation icons provide a splash of color, while their labels ensure usability. The textual local navigation options allow for the creation of many footer navigation bars without restrictive costs.

Frames

Frames present an additional factor to consider in the application of textual or graphical navigation bars. Frames allow you to define one or more independently scrollable "panes" within a single browser window. Hypertextual links within one pane can control the content displayed in other panes within that same window. This enables the designer to create a static or independently scrolling navigation bar that appears on every page in that area of the web site. This frame-based navigation bar will be visible to the user in the same location in the browser window even while scrolling through long documents. By separating the navigation system from content in this way, frames can provide added context and consistency as users navigate a web site.

However, frames present several serious problems, both from the consumer's and producer's perspective. Architects should proceed very carefully in considering frame-based navigation solutions. Let's review a few of the major considerations.

Screen real estate

Static navigation bars implemented through frames often take up significant portions of valuable screen real estate (see Figure 4-12). No matter how far the user scrolls, the navigation bar always stays with them. The addition of winking, blinking banner advertisements into the static navigation bar often compounds this problem. On a large, high resolution monitor this may be only a minor irritation. On a standard 640 x 480 monitor, these frames can be really annoying. If you're going to use a frame-based navigation bar, keep it relatively small and non obtrusive. You should also consider a vertical rather than horizontal frame, since left-to-right reading lends itself to narrow text columns like those found in newspapers and magazines.

Figure 4-12. *The Wall Street journal's* Interactive Edition makes use of frames. It's a relatively elegant implementation, but it limits screen real estate and disables basic navigation features.

The page model

The Web is built upon a model of pages, with each page having a unique address or URL. Users are familiar with the concept of pages. Frames confuse this issue, by slicing up pages into independent panes of content. By violating the page model, the use of frames frequently disables important browser navigation features such as bookmarking, visited and unvisited link discrimination, and history lists. Frames can also confuse and frustrate users executing simple tasks such as using the back button, reloading a page, and printing a page. While web browsers have improved in their ability to handle frames, they can't remove the confusion caused by violating the page model.

Display speed

Right off the bat, a web page with multiple panes will take a hit on display speed. Since each pane is a separate file with its own URL, loading and displaying each pane requires a separate client-server interaction. In other words, the user spends a lot of time watching "Host Contacted" messages fly by at the bottom of the screen. This problem is compounded by heavy graphics use.

Complex design

In theory, there are some compelling reasons to try frames. You can make global navigation bars or section headers (or advertisements) visible to the user at all times. However, in practice, designing user-friendly web sites using frames is quite challenging. Frames add a layer of complexity that many architects and designers deal with unsuccessfully. You must think about the multiple ways users will access your frame-based documents. What if they come from another frame-based documents. Then you face the danger of frames within frames. In addition, while most web browsers now support frames, different browsers on different computer platforms display the frames and their contents slightly differently. This requires more testing and more careful design. Before using frames, make sure you consider the additional overhead in architecture and design.

Pull-Down Menus

Pull-down menus compactly provide for many navigation options. The user can expand what appears as a single-line menu to present dozens of options (as shown in Figure 4-13). The most common pull-down menus on the Web are implemented using the standard interactive forms syntax. Users must choose an option from the menu and then hit a Go or Submit button to move to that destination.

You can implement a more sophisticated version of the pull-down menu (also know as the *pop-up menu*) on the Web by using a programming language such as Java or JavaScript. As the user moves the cursor over a word or area on the page, a menu pops up. The user can directly select an option from that menu.

Use pull-down and pop-up menus with caution. These menus allow designers to pack lots of options on one page. This is usually what you are working hard to avoid. Additionally, menus hide their options and force the user to act before being able to see those options. However, when you have a very straightforward, exact organization scheme, these menus can work well.

Remote Navigation Elements

Remote navigation elements or supplemental navigation systems such as tables of contents, indexes, and site maps are external to the basic hierarchy of a web site and provide an alternative bird's-eye view of the site's content. Increasingly, we are seeing these remote navigation elements displayed outside of the main browser window, in either a separate target window or in a Java-based remote control panel. While remote navigation elements can enhance access to web site content by providing complementary ways of navigating,

Figure 4-13 This pull-down menu enables users to select a location without first going to a separate web page. This approach avoids further cluttering the main page with a long list of locations.

they should not be used as replacements or bandages for poor organization and navigation systems. In many ways, remote navigation elements are similar to software documentation or help systems. Documentation can be very useful but will never save a bad product. Instead, remote navigation elements should be used to complement a solid internal organization and navigation system. You should provide them but never rely on them.

The Table of Contents

The table of contents and the index are the state of the art in print navigation. Given that the design of these familiar systems is the result of testing and refinement over the centuries, we should not overlook their value for web sites.

In a book or magazine, the table of contents presents the top few levels of the information hierarchy. It shows the organization structure for the printed work and supports random as well as linear access to the content through the use of chapter and page numbers. Similarly, the table of contents for a web site presents the top few levels of the hierarchy. It provides a broad view of the content in the site and facilitates random access to segmented portions of that content. A web-based table of contents can employ hypertext links to provide the user with direct access to pages of the site.

You should consider using a table of contents for web sites that lend themselves to hierarchical organization. If the architecture is not strongly hierarchical, it makes no sense to present the parent-child relationships implicit in a structured table of contents. You should also consider the web site's size when deciding whether to employ a table of contents. For a small site with only two or three hierarchical levels, a table of contents may be unnecessary.

The design of a table of contents significantly affects its usability. When working with a graphic designer, make sure he or she understands the following rules of thumb:

1. Reinforce the information hierarchy so the user becomes increasingly familiar with how the content is organized.
2. Facilitate fast, direct access to the contents of the site for those users who know what they want.
3. Avoid overwhelming the user with too much information. The goal is to help, not scare, the user.

The *Search/Browse* area of the Argus Clearinghouse, shown in Figure 4-14, provides an example of a table of contents.

Graphics can be used in the design and layout of a table of contents, providing the designer with a finer degree of control over the presentation. Colors, font styles, and a variety of graphic elements can be applied to create a well-organized and aesthetically pleasing table of contents. However, keep in mind that a graphic table of contents will cost more to design and maintain and may slow down the page loading speed for the user. When designing a navigation tool such as a table of contents, form is less important than function.

Figure 4-14. This table of contents allows users to select a category (e.g., Arts & Humanities) or jump directly to a subcategory (e.g., architecture). Because of the clean page layout, users can quickly scan the major and minor categories for the topic they're interested in.

The Index

For web sites that aren't conducive to strong hierarchical organization, a manually created index can be a good alternative to the more structured table of contents. Similar to an index found in print materials, a web-based index presents keywords or phrases alphabetically, without representing the hierarchy. Unlike a table of contents, indexes generally are flat and present only one or two levels of depth. Therefore, indexes work very well for users who already know the name of the item they are looking for. A quick scan of the alphabetical listing will get them where they want to go.

A major challenge in indexing a web site involves the level of granularity of indexing. Do you index web pages? Do you index individual paragraphs or concepts that are presented on web pages? Or do you index collections of web pages? In many cases, the answer may be *all of the above*. Perhaps a more valuable question is: *What terms are users going to look for?* Its answers should guide the index design. To answer this question, you need to know your audience and understand their needs. Before launch of the site, you can learn more about the terms that users will look for through focus group sessions and individual user interviews. After launch, you can employ a query tracking tool that captures and presents all search terms entered by users. Analysis of these actual user search terms should determine refinement of the index. (To learn more about query tools, see Chapter 9, *Production and operations*.)

In selecting items for the index, keep in mind that an index should point only to destination pages, not navigation pages. Navigation pages help users find (destination pages) pages through the use of menus that begin on the main page and descend through the hierarchy. They are often heavy on links and light on text. In contrast, destination pages contain the content that users are trying to find. The purpose of the index is to enable users to bypass the navigation pages and jump directly to these content-bearing destination pages.

A useful trick in designing an index involves term rotation, also known as permutation. A permuted index rotates the words in a phrase so that users can find the phrase in two places in the alphabetical sequence. For example, in the SIGGRAPH 96 index shown in Figure 4-15, users will find listings for both *New Orleans Maps* and *Maps* (*New Orleans*). This supports the varied ways people look for information. Term rotation should be applied selectively. You need to balance the probability of users seeking a particular term with the annoyance of cluttering the index with too many permutations. For example, it would probably not make sense to present Sunday (Schedule) as well as Schedule (Sunday). If you have the time and budget to conduct focus groups or user testing, that's great. If not, you'll have to fall back on your common sense.

Figure 4-15. The SIGGRAPH 96 index allows for multiple levels of multiple levels of granularity. Selecting "New Orleans" will take you to a page that introduces this adventurous city and includes a number of links. One of those links takes you to a New Orleans map. Since this map is judged to be an important content item, it is also presented in the index.

The Site Map

While the term *site map* is used indiscriminately in general practice, we define it narrowly as a graphical representation of the architecture of a web site. This definition excludes tables of contents and indexes that use graphic elements to enhance the aesthetic appeal of tools that are primarily textual. A real site map presents the information architecture in a way that goes beyond textual representation.

Unlike tables of contents and indexes, maps have not traditionally been used to facilitate navigation through bodies of text. Maps are typically used for navigating physical rather than intellectual space. This is significant for a few reasons. First, users are not familiar with the use of site maps. Second, designers are not familiar with the design of site maps. Third, most bodies of text (including most web sites) do not lend themselves to graphical representations. As we discussed in Chapter 3, *Organizing Information*, many web sites incorporate multiple organization schemes and structures. Presenting this web of hypertextual relationships visually is difficult. These reasons help explain why we see few good examples on the Web of site maps that can improve navigation systems.

Figure 4-16 shows a site map from *http://www.sgml.net*. To learn more about automatically generated site maps, see *http://www.webreview.com/97/05/16/arch/index.html*.

If you decide to try a site map, consider physical versus symbolic representation. Maps of the physical world do not present the exact geography of an area. Accuracy and scale are often sacrificed for representative contextual clues that help us find our way through the maze of highways and byways to our

Figure 4-16. In this example of an automatically generated site map, go bars represent pages within a web site. Users must roll their cursor over a Sold bar to see the title of the page. Do you think this approach is more useful than a text-based table of contents?

destination. Often, the higher the level of abstraction, the more intuitive the map. This rule of thumb holds true for all of the remote navigation elements of web sites. When consulting a table of contents or index or site map, a user doesn't need to see every single link on every single page. They need to see the important links, presented in a clear and meaningful way.

The Guided Tour

A guided tour serves as a nice tool for introducing new users to the major content areas of a web site. It can be particularly important for restricted access web sites (such as online magazines that charge subscription fees) because you need to show potential customers what they will get for their money.

A guided tour should feature linear navigation (new users want to be guided, not thrown in), but a hypertextual navigation bar may be used to provide additional flexibility. The tour should combine screenshots of major pages with narrative text that explains what can be found in each area of the web site. See Figure 4-17 for an example.

Remember that a guided tour is intended as an introduction for new users and as a marketing opportunity for the web site. Many people may never use it, and few people will use it more than once. For that reason, you might consider linking to the tour from the gateway page[‡‡] rather than the main page. Also, you should balance the inevitable big ideas about how to create an exciting, dynamic, interactive guided tour with the fact that it will not play a central role in the day to day use of the web site.

[‡‡] Web sites sometimes have a getaway page that first-time users encounter before reaching the main page. This gateway might serve as a plash page with fancy graphics and animation, as an audience-selection page that sends users to the appropriate area of a site, or as a preview page that shows users what they will get if they subscribe to that particular web site.

Figure 4-17. In this example, the navigation options on each screen allow users to move through the guided tour in a non-linear manner.

Designing Elegant Navigation Systems

Designing navigation systems that work well is challenging. You've got so many possible solutions to consider, and lots of sexy technologies such as pop-up menus and dynamic site maps can distract you from what's really important: building context, improving flexibility, and helping the user to find the information they need.

No single combination of navigation elements works for all web sites. One size does not fit all. Rather, you need to consider the specific goals, audience, and content for the project at hand, if you are to design the optimal solution.

However, there is a process that should guide you through the challenges of navigation system design. It begins with the hierarchy. As the primary navigation system, the hierarchy influences all other decisions. The choice of major categories at the highest levels of the web site will determine design of the global navigation system. Based on the hierarchy, you will be able to select key pages (or types of pages) that should be accessible from every other page on the web site. In turn, the global navigation system will determine design of the local and then ad hoc navigation systems. At each level of granularity, your design of the higher-order navigation system will influence decisions at the next level.

Once you've designed the integrated navigation system, you can consider the addition of one or more remote navigation elements. In most cases, you will need to choose between a table of contents, an index, and a site map. Is the hierarchy strong and clear? Then perhaps a table of contents makes sense. Does the hierarchy get in the way? Then you might consider an index. Does

the information lend itself to visualization? If so, a site map may be appropriate. Is there a need to help new or prospective users to understand what they can do with the site? Then you might add a guided tour.

If the site is large and complex, you can employ two or more of these elements. A table of contents and an index can serve different users with varying needs. However, you must consider the potential user confusion caused by multiple options and the additional overhead required to design and maintain these navigation elements. As always, it's a delicate balancing act.

If life on the high wire unnerves you, be sure to build some usability testing into the navigation system design process, Only by learning from users can you design and refine an elegant navigation system that really works.

Bibliography

Tufte, Edward R. *Envisioning Information*, 3rd Edition. Cheshire, CT: Graphics Press, 1990.

Tufte, Edward R. *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press, 1992.

Tufte, Edward R. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Cheshire, CT: Graphics Press, 1997.

Wurman, Richard Saul. *Information Architects*. Zurich, Switzerland: Graphis Press Corp, 1996.

Organization

Blair, David C. *Language and Representation in Information Retrieval*. New York: Elsevier Science Publishers, 1990.

"Cataloguing Policy and Support Office Home Page." Library of Congress. *http://www.ocl.org/ocl/fp/ddchome.htm*.

"Dewey Decimal System Home Page." OCLC Forest Press. 1997. *http://www.ocl.org/oclc/fp/ddchome.htm*.

Friedlander, Amy, ed. *D-Lib Magazine. The Magazine of Digital Library Research*. Reston, VA: Corporation for National Research Initiatives. *http://www.dlib.org/*.

Gorman, Michael and Paul W. Winkler, eds. *Anglo-American Cataloging Rules*, 2nd Edition, 1998 Revision ed. Chicago, IL: American Library Association, 1988.

"Hypertext Now: Archives." Eastgate Systems. *http://www.eastgate.com/HypertextNow/.*

Lakoff, George and Mark Johnson. *Metaphors We Live By*. Chicago: University of Chicago Press, 1983.

Meadow Charles T. *Text information Retrieval Systems*. San Diego: Academic Press, 1992.

Richmond, Alan and Lucy Richmond. "The WDVL: Resource Location". Web Developer's Virtual Library, Cyberweb Software.*http://Stars.com/Location/.*

Rosenfeld, Louis. "Particles, Waves, and the Site Visualization", Web Architect. *Web Review Magazine*, July 1987, *http://www.review.com/97/07/11/arch/index.html*.

Rowlev, Jennifer E. *Organizing Knowledge*, 2nd Edition. Brookfield, VT: Ashgate Publishing, 1992.

Rosenfeld, Louis and Morville, Peter. *Information Architecture for the World Wide Web*. Cambridge, KöIn, Paris, Sebastopol, Tokyo: O'REILLY.


UOC
La universidad virtual


71H009625EC