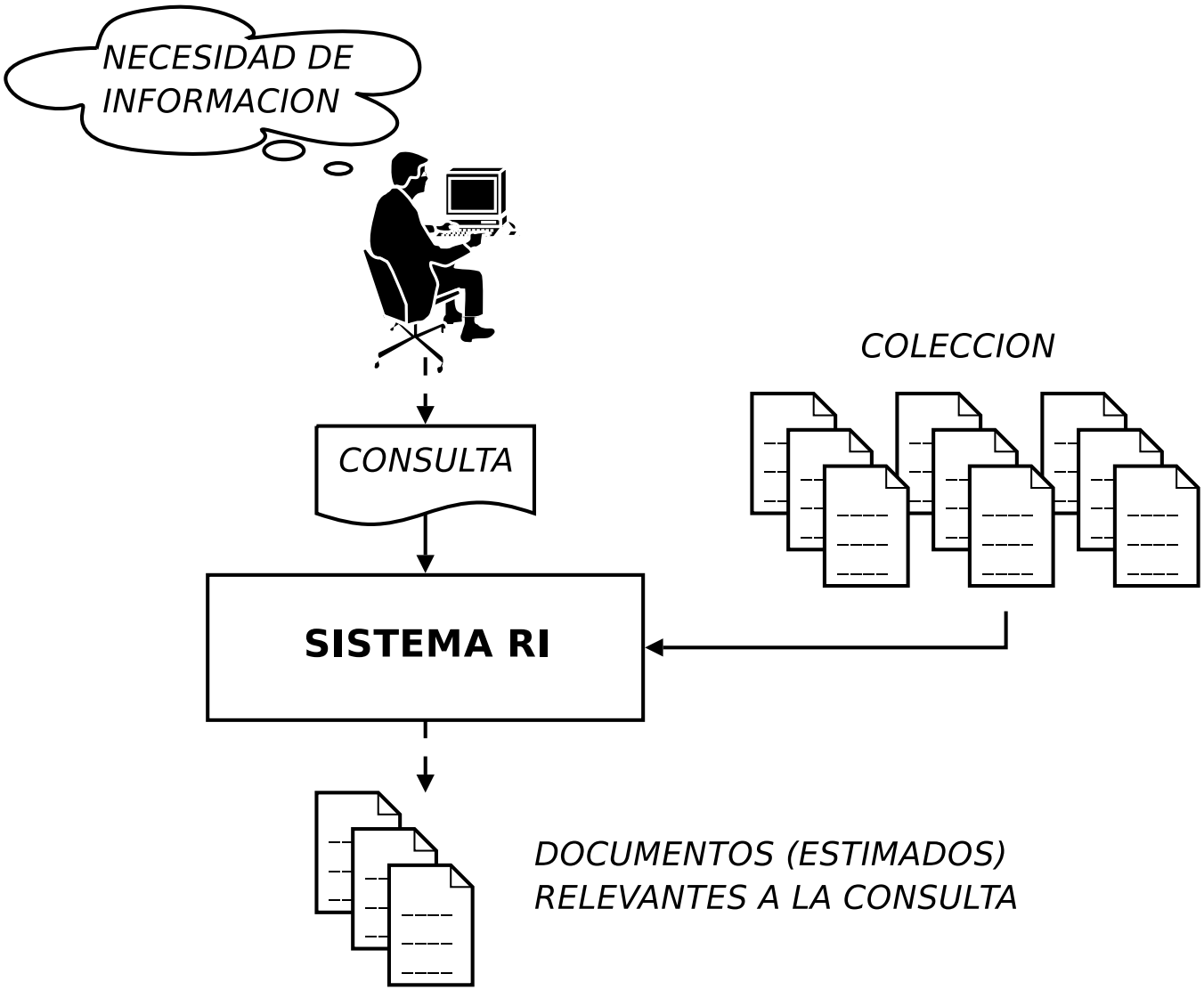
Índice

Recuperación de Información (RI)

- A.k.a. *Information Retrieval (IR)*
- **Def.:** área de la ciencia y la tecnología que trata de la representación, almacenamiento, organización y acceso a elementos de información
- **Objetivo:** dada una **colección de documentos** y una **necesidad de información** del usuario —expresada como una **consulta (query)**—, devolver un conjunto de documentos relevantes para dicha necesidad de información (i.e., cuyo contenido satisface dicha necesidad)
  - No devuelve la información deseada, sólo indica los documentos donde parece estar
- P.ej., buscadores web

Proceso de RI



Terminología

- **Documento:** unidad de texto almacenada y disponible para su recuperación; p.ej., páginas web, artículos de prensa, tesis, ...
  - Granularidad variable: documento completo, capítulos, párrafos, ...
- **Colección:** repositorio de documentos en los que buscar
- **Términos:** unidades léxicas (palabras) que componen un documento/consulta
- **Consulta (query): representación** en forma de términos, de la necesidad de información del usuario
- **Relevancia de un documento:**
  - Establecida por el sistema respecto a la *consulta*
  - Juzgada por el usuario respecto a la **necesidad de información** en su cabeza (**subjetividad**)
- **Ordenación (ranking):** los documentos suelen devolverse ordenados por relevancia

Bases de Datos vs. Sistemas de RI

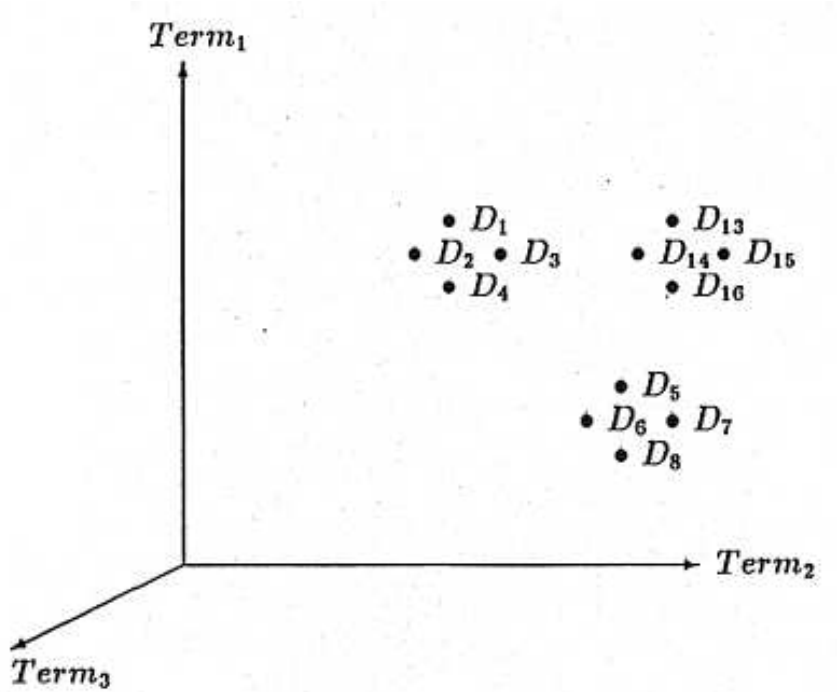
	BD	**RI**
Información	datos estructurados semántica bien definida	lenguaje natural (desestructurado) semántica ambigua
Consulta	formalizada (álgebra relacional)	lenguaje natural
Resultados	todos los relevantes (completitud) todos son relevantes (ningún error)	no necesariamente contiene errores: objetivo • maximizar relevantes devueltos • minimizar no relevantes devueltos

Tareas de RI

- **Recuperación ad hoc**:
  - P.ej., *buscadores web*
  - La colección no varía (estática) o varía poco (semiestática) —web
  - Consultas variables (dinámicas), puntuales y específicas
- **Categorización (clasificación de documentos):**
  - Asignar un doc. a una/más clases fijadas a priori
  - Los documentos van llegando poco a poco (colección dinámica)
  - Necesidades (perfiles o *profiles*) complejas, estables en el tiempo (estáticas), y que reflejan varias necesidades a la vez
  - 2 [sub]tareas diferenciadas:
    - Enrutamiento (routing): ordenación por similaridad con el perfil
    - Filtrado (filtering): acepta o rechaza (binario)

Tareas de RI (cont.)

- **Clustering:**



  - Generar automática. clases (*clústers*) a partir de un conjunto de docs.
    - Maximizar similaridad intra-clúster
    - Minimizar similaridad inter-clúster

- De aquí en adelante: **recuperación ad hoc**

Paradigma *Bag-of-Terms*

- **Def.:** representación de documentos/consultas como **conjunto** de *términos índice* (a.k.a. *términos de indexación* o *palabras clave*)
- **Ppo. de composicionalidad de Frege**: "la semántica de un objeto puede obtenerse a partir de la semántica de sus componentes"
  - Si una palabra aparece en un texto, dicho texto trata dicho tema
  - **Si una consulta y un documento comparten uno/más términos índice, el documento debería tratar el tema de la consulta**

Peso de un Término

- No todos los términos tendrán la misma **importancia/representatividad**: **peso (weight)** w_{ij} de un término t_i en un documento d_j
- Factores para cómputo del peso de un término:
  - Frecuencia dentro del documento
  - Distribución dentro de la colección
  - Longitud del documento
  - Forma combinarlos varía según *modelo* y formúla empleados

Peso de un Término (cont.): Frecuencia en el Documento

- Si un término aparece muchas veces en un doc., se puede suponer que el doc. está más relacionado con este tema $\longrightarrow$ **mayor peso**
- P.ej., si en un documento aparece *chocolatina* repetidamente, es lógico pensar que dicho documento habla sobre chocolatinas
- **Frecuencia del término t_i en el documento d_j (tf_{ij}):** número de veces que aparece el término t_i en el documento d_j

Peso de un Término (cont.): Distribución en la Colección

- A mayor número de documentos en los que aparece un término, menor su poder de discriminación $\longrightarrow$ **menor peso**
- P.ej., si *chocolatina* aparece en gran parte de los documentos de la colección, poco ayuda a diferenciar unos de otros
- **Frecuencia inversa de documento del término t_i (idf_i):**

$$idf_i = log \frac{N}{n_i}$$

donde N es el número total de documentos de la colección, y n_i el número de dichos documentos en los que aparece el término t_i.

Peso de un Término (cont.): Longitud del Documento

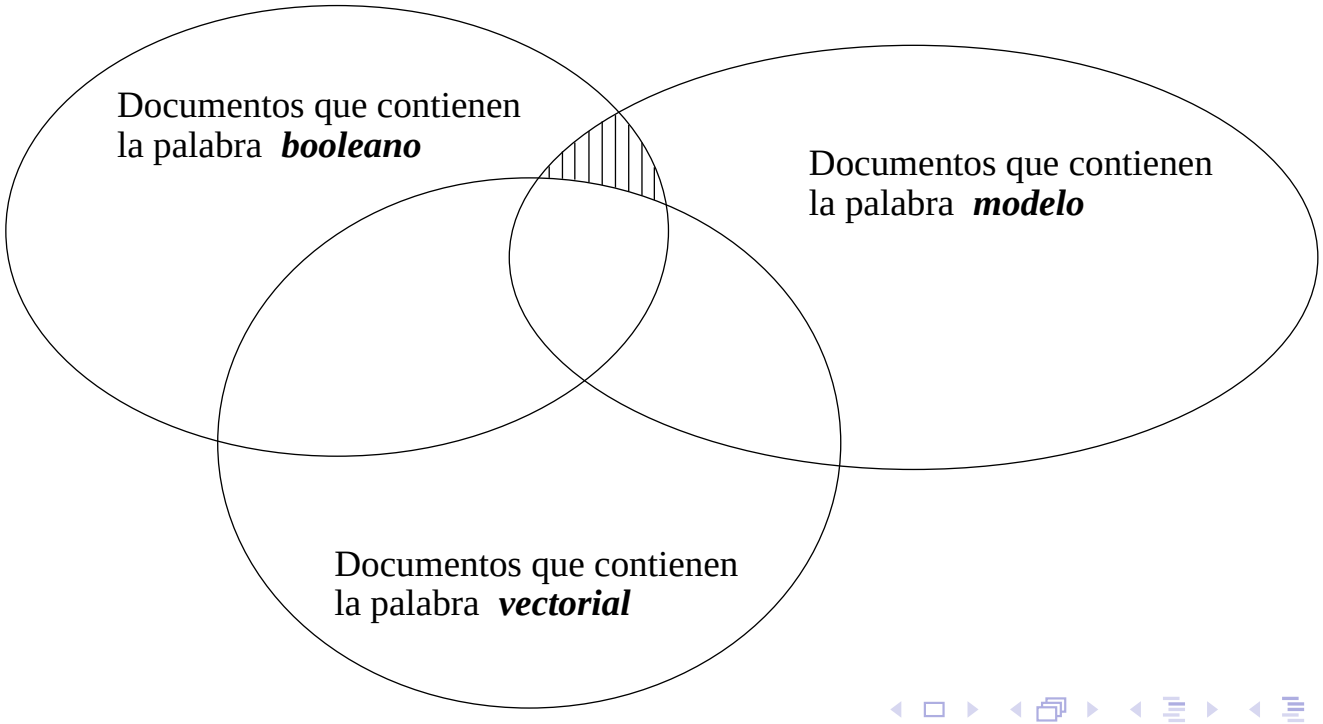
- A mayor longitud, mayor probabilidad de correspondencias, por lo que, a priori, dichos docs. partirían con ventaja → **ponderar según longitud**
- No siempre tenido en cuenta

Introducción

- Establece:
  - Cómo representar los documentos
  - Cómo representar las necesidades de información
  - Cómo compararlos

Modelo Booleano

- Base matemática: **teoría de conjuntos** y **álgebra de Boole**
- Consulta: expresión booleana de términos ligados por operadores booleanos (AND, OR y NOT)
- Devuelve aquellos documentos que satisfacen la consulta
  - **Binario** (sí/no relevante): **no hay gradación de la relevancia**
- Ejemplo: *modelo* AND *booleano* AND NOT *vectorial*



Modelo Booleano (cont.)

- **Ventajas:**
  - Sencillo
  - Preciso
  - Veloz
- **Desventajas:**
  - Formalizar consulta como expresión booleana:
    - Fácil quedarse corto/largo
    - Pequeñas modificaciones en la consulta pueden provocar grandes variaciones en los resultados. P.ej., AND vs. OR
  - Binario:
    - No permite correspondencias parciales
    - Sin gradación de la relevancia/similaridad: resultados no ordenados + todos los términos valen lo mismo (como si $w_{ij} = \{0, 1\}$)
- Muy popular en el pasado. Hoy relegado a sistemas precisan correspondencias exactas: P.ej., sistemas de información legislativa

Modelo Booleano (cont.): Booleano Extendido

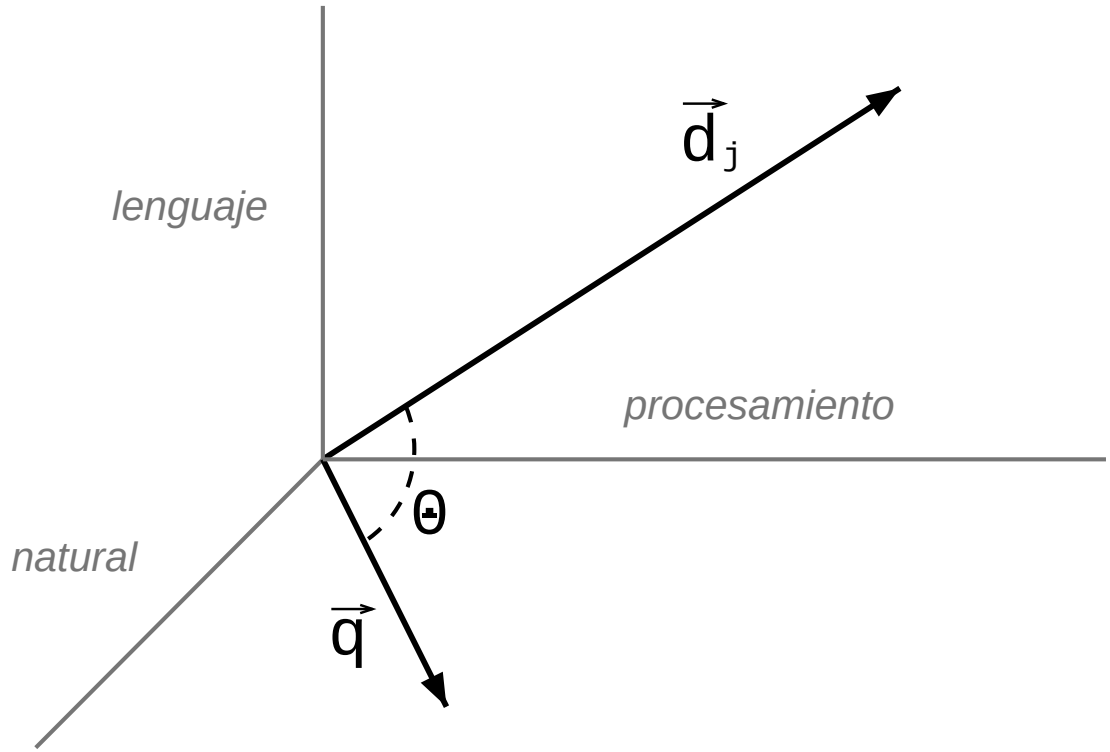
- Variante del modelo booleano clásico
  - Permite **ordenación por relevancia** de los docs.:
    1. Operar estrictamente a nivel de conjs. booleanos
    2. Ordenar el conj. resultante mediante **medida de similaridad**

Modelo Vectorial

- Base matemática: **álgebra vectorial**
- **Consultas y documentos representados como vectores** en un espacio multidimensional
  - Definido por los términos del vocabulario: 1 dimensión por término
  - P.ej. Vocabulario tamaño $T \rightarrow$ espacio T-dimensional
    - Documento d_j: vector $\overrightarrow{d_j} = (w_{1j}, w_{2j}, \ldots, w_{tj})$
    - Consulta q: vector $\overrightarrow{q} = (w_{1q}, w_{2q}, \ldots, w_{tq})$

    siendo $w_{ij} \geq 0$ y $w_{iq} \geq 0$ los pesos del término t_i en el documento d_j y la consulta q, respectivamente.

Modelo Vectorial (cont.)



- Si los vectores de consulta y documento están próximos, asumimos que documento es similar a la consulta (i.e., posiblemente relevante)

Modelo Vectorial (cont.)

- **Medida proximidad:** coseno del ángulo Θ formado por los vectores:

$$sim(d_j, q) = cos(\Theta) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{t} w^2{}_{ij}} \times \sqrt{\sum_{i=1}^{t} w^2{}_{iq}}}$$

$|\vec{q}|$ es constante para una consulta dada, luego puede simplificarse

- **Peso de un término:** clásico, esquema *tf-idf*:

$$w_{ij} = tf_{ij} \times idf_i$$

Modelo Vectorial (cont.)

- **Ventajas:**
  - Sencillo
  - Preciso (buenos resultados)
  - Permite **correspondencias parciales**
  - Permite **gradación relevancia/similaridad**: *ranking* de docs.
- **Referencia para comparación** de otros sistemas

Modelo Probabilístico

Véase Tutorial

Proceso

1. **Normalización** (*conflation*):
   1. Tokenización
   2. Eliminación de *stopwords*
   3. Paso a minúsculas y eliminación de signos ortográficos
   4. *Stemming*
   5. Selección de términos índice
2. **Indexación**

Normalización

- Proceso de generación de términos índice de docs./consultas mediante transformaciones sucesivas del texto (*operaciones de texto*)
- **Objetivo:** reducción del texto a una **forma canónica** para facilitar las correspondencias

Normalización (cont.): Tokenización

- **Identificación de las palabras del texto**:
  - Def. *palabra*: "secuencia de caracteres de palabra delimitada por separadores"
- Herramientas sencillas y rápidas

Normalización (cont.): Eliminación de *Stopwords*

- **Def. stopwords:** palabras de escasa o nula utilidad d.p.d.v. recuperación:
  - Escaso contenido semántico; p.ej.: artículos, preposiciones, etc.
  - Frecuencia excesiva (nula capacidad discriminante); p.ej.: formas verbales de *ser* o *estar*
- Su eliminación permite un **considerable ahorro de recursos de almacenamiento**:
  - Mínima parte del vocabulario pero gran parte de los términos del texto (*Ley de Zipf*)
  - Listas preestablecidas de *stopwords*
- En las consultas, eliminar también información de *metanivel*

  P.ej., *"Encuentre los documentos que describan ..."*

Normalización (cont.): Paso a Minúsculas y Eliminación de Signos Ortográficos

- P.ej., ” ... *Paso a Minúsculas ...* ” → ” ... *paso a minusculas ...* ”
- **Objetivo:** facilitar las correspondencias

Normalización (cont.): *Stemming*

- **Def.:** reducción de una palabra a su *stem* o raíz supuesta eliminando su terminación según una **lista de sufijos**
  - *Stem* o raíz contiene semántica básica

$$\left.\begin{array}{l}\texttt{reloj} \\ \texttt{relojes} \\ \texttt{relojero}\end{array}\right\} \rightarrow \texttt{reloj-}$$

- **Objetivo:**
  - Principal: permitir correspondencias entre variantes
  - Secundario: reducir recursos almacenamiento (reducir vocabulario)
- *Stemmer* de Porter
  - Demo: `http://maya.cs.depaul.edu/~classes/ds575/porter.html`
  - `Snowball` (descargables): `http://snowball.tartarus.org`
- Nivel de normalización
  - *Superficial*: sólo morfología flexiva simplificada; p.ej., sólo plurales
  - *Profundo*: flexiva y derivativa (agresivo); p.ej., Porter

Normalización (cont.): *Stemming*

- **Def.:** reducción de una palabra a su *stem* o raíz supuesta eliminando su terminación según una **lista de sufijos**
  - *Stem* o raíz contiene semántica básica

$$\left.\begin{array}{l}\texttt{reloj}\\ \texttt{relojes}\\ \texttt{relojero}\end{array}\right\} \rightarrow \texttt{reloj-}$$

- **Objetivo:**
  - Principal: permitir correspondencias entre variantes
  - Secundario: reducir recursos almacenamiento (reducir vocabulario)
- *Stemmer* de Porter
  - Demo: `http://maya.cs.depaul.edu/~classes/ds575/porter.html`
  - `Snowball` (descargables): `http://snowball.tartarus.org`
- Nivel de normalización
  - *Superficial*: sólo morfología flexiva simplificada; p.ej., sólo plurales
  - *Profundo*: flexiva y derivativa (agresivo); p.ej., Porter

Normalización (cont.): Selección de Términos Índice

- Los términos resultantes son tomados como términos índice asociándoseles, si es necesario, el *peso* correspondiente
- Representación:
  - **A texto completo**
  - Selección de términos: manual/automáticamente
    - P.ej., indexación con vocabulario controlado

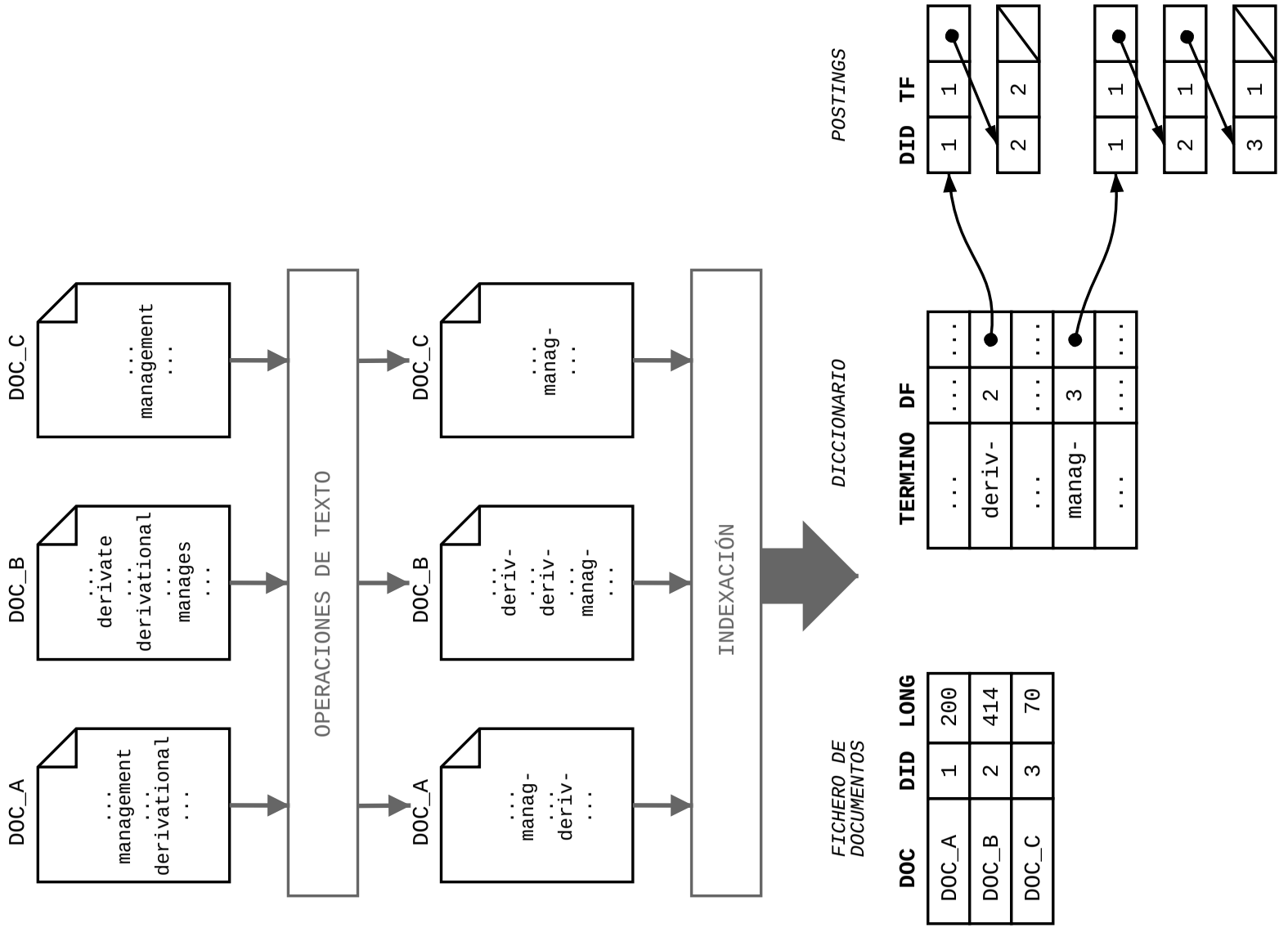
Indexación: Componentes Sistema de RI

- **Índice invertido:** estructuras auxiliares donde almacenar las representaciones de los docs.
  - **Objetivo:** acelerar la búsqueda
  - **Componentes:**
    - **Vocabulario/diccionario:** lista de términos índice en la colección
    - **Postings:** lista de docs. en los que aparece cada término
    - **Fichero de documentos:** lista de documentos del sistema

    Asociado a cada entrada (término/*posting*/documento) se almacenan los datos necesarios para el cálculo del peso/relevancia

- **Motor de indexación:** componente software encargado del manejo de los índices

Indexación (cont.): Generación del Índice



ejemplo_Generacion_Indice.pdf

Introducción

- **Proceso:**
  1. El usuario plasma su **necesidad de información** en una **consulta**
  2. El sistema obtiene la **representación interna** de la consulta aplicando *operaciones de texto* (las mismas que durante *indexación*)
  3. El sistema compara la **representación interna** obtenida con los documentos indexados
- El sistema parte de la consulta formulada por el usuario. **Peligro:**
  - Si la consulta está formulada **de forma incorrecta o insuficiente**
  - Si usuario y autor del documento emplean **términos diferentes (variación lingüística)**
- **Solución paliativa:** *expansión de consultas*

Expansión de Consultas

- A.k.a. **query expansion**
- **Def.:** procesos automáticos/semiautomáticos para la reformulación/refinamiento de la consulta inicial mediante la adición de nuevos términos
  - Relacionados con los términos iniciales
  - Asociados a los documentos [supuestamente] relevantes

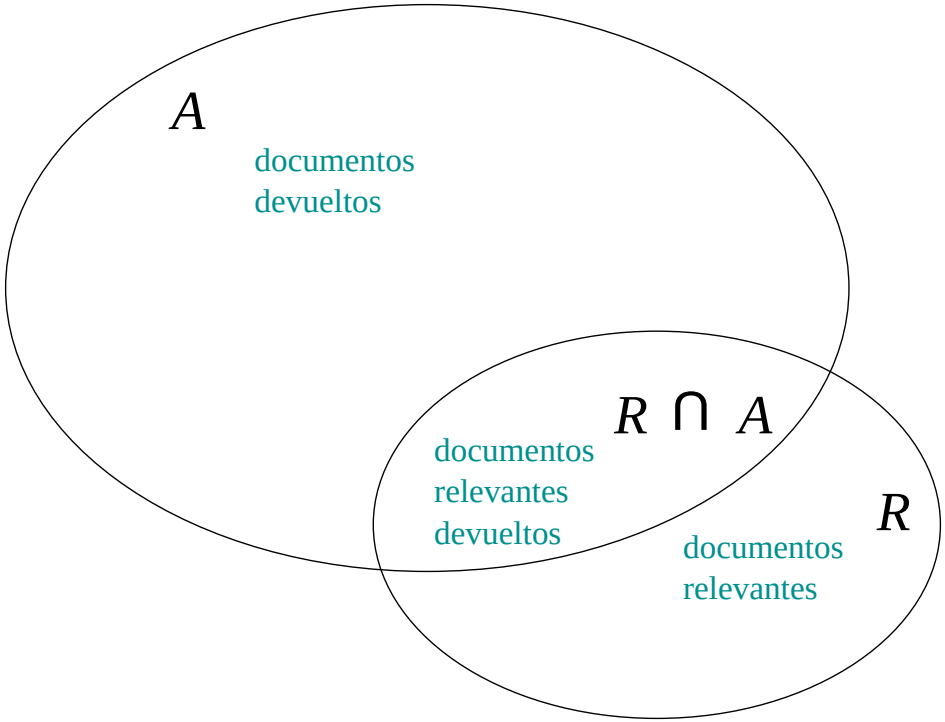
Expansión de Consultas (cont.): Mediante Tesauros

- **Def. tesauro (thesaurus):** base de datos lexicográfica que almacena una representación jerárquica de un lexicón de acuerdo a las relaciones semánticas existentes entre sus palabras
  - P.ej., WordNet (más utilizado)
- **Proceso:** reformular consulta inicial
  - Manualmente: navegando por su estructura eligiendo los términos con los que expandir
  - Automáticamente: p.ej., expandiendo un término con sus sinónimos
- Por lo general **no se logran mejoras** en los resultados
  - Introducción de **ruido** durante la expansión: necesario aplicar técnicas de *desambiguación del sentido de las palabras (WSD)*

Expansión de Consultas (cont.): Mediante Realimentación

- A.k.a. **relevance feedback**
- **Proceso:**
  1. Se lanza consulta inicial contra el sistema
  2. Se toman los primeros documentos devueltos por el sistema
     - Manual: usuario los examina y determina cuáles son relevantes
     - Automático: los *n* primeros documentos se suponen relevantes (*expansión ciega* o *por pseudo-relevancia*)
  3. Partiendo de los documentos estimados como relevantes:
     - **Se añaden nuevos términos** que aparezcan en ellos
     - **Se modifican los pesos** de los términos de la consulta inicial
- **Buen comportamiento** general (uso ampliamente extendido)

Métricas de Evaluación



- D, conjunto de **documentos** en la colección
- R, conjunto de **documentos relevantes** en la colección
- $\bar{R} = D - R$, conjunto de **documentos no relevantes** en la colección
- A, conjunto de **documentos recuperados** por el sistema
- $A \cap R$, conjunto de **documentos relevantes recuperados** por el sistema

Métricas de Evaluación (cont.)

- **Precisión (precision):** porcentaje de documentos recuperados que son relevantes:

$$precision \;\; Pr = \frac{A \cap R}{A}$$

  Capacidad para recuperar sólo documentos relevantes.

- **Cobertura (recall):** porcentaje de documentos relevantes que son recuperados:

$$recall \;\; Re = \frac{A \cap R}{R}$$

  Capacidad para recuperar todos los documentos que son relevantes.

- **Medida-F o F_1 (F-measure/balanced F-score):** combina ambas

$$F_1 = \frac{2 \; Re \; Pr}{Re + Pr}$$

  Valora Re/Pr por igual.

- A nivel de **consulta** o **conjunto de consultas** (media)

Colecciones de Referencia/Evaluación

- Composición: 3 elementos
  1. Documentos
  2. Consultas
  3. Lista de los documentos relevantes para cada consulta
- Más importantes (asociadas a instituciones/congresos):
  - TREC
  - CLEF

Colecciones de Referencia (cont.): TREC

- Text REtrieval Conference (`http://trec.nist.gov/`)
  - National Institute of Standards and Technology (NIST)
  - Defense Advanced Research Projects Agency (DARPA)
- **Objetivo:** facilitar infraestructura, herramientas y metodologías para la **evaluación a gran escala** de sistemas de RI (**inglés**)
  - Diferentes sesiones (*tracks*): *ad hoc*, *terabyte (web)*, filtrado, etc.
  - Consultas: ∼50 por año y *track*
  - Colecciones:

Ad hoc track:	*L.A. Times*	131,896 artículos	475 MB	media 527 words
Terabyte track:	GOV2	25,205,179 webs (.gov)	426 GB	media 17.7 KB

  - Evaluación:
    - **Técnica pooling:** revisan manualmente K *top* docs. (K=100)
    - **Def. relevancia:**

      "Si estuviera redactando un informe sobre el tema del topic en cuestión y pudiese usar para dicho informe la información contenida en el documento examinado, entonces dicho documento será considerado relevante"

Colecciones de Referencia (cont.): CLEF

- Cross-Language Evaluation Forum (`http://www.clef-campaign.org/`)
- TREC europeo:
  - Lenguas europeas (y otras):
    - Colecciones: inglés, francés, español, alemán, holandés, italiano, portugués, búlgaro, checo, húngaro, ruso, sueco y finlandés
    - Topics (a mayores): chino, japonés, griego, arameo, hindi, bengalí, marathi, oromo, tamil, telugu
  - *Tracks* mono e interlingües —*Recuperación de Información InterLingüe* (RIIL)

Colecciones de Referencia (cont.): Ejemplo de *Topic*

```
<top>
<num> C044 </num>
<ES-title> Indurain gana el Tour </ES-title>
<ES-desc> Reacciones al cuarto Tour de Francia ganado por Miguel Indurain.
</ES-desc>
<ES-narr> Los documentos relevantes comentan las reacciones a la cuarta
victoria consecutiva de Miguel Indurain en el Tour de Francia. Los
documentos que discuten la relevancia de Indurain en el ciclismo mundial
después de esta victoria también son relevantes. </ES-narr>
</top>
```

- A partir de los cuales se generan (automática o manualmente) las consultas finales
- **3 elementos:**
  - *Título*: breve título
  - *Descripción*: frase de descripción
  - *Narrativa*: pequeño texto especificando los criterios que utilizarán los revisores para establecer la relevancia de un documento respecto a la consulta

Colecciones de Referencia (cont.): Ejemplo de *Documento*

```
<DOC>
<DOCNO>EFE19940101-00002</DOCNO>
<DOCID>EFE19940101-00002</DOCID>
<DATE>19940101</DATE>
<TIME>00.34</TIME>
<SCATE>VAR</SCATE>
<FICHEROS>94F.JPG</FICHEROS>
<DESTINO>ICX MUN EXG</DESTINO>
<CATEGORY>VARIOS</CATEGORY>
<CLAVE>DP2404</CLAVE>
<NUM>100</NUM>
<PRIORIDAD>U</PRIORIDAD>
<TITLE> IBM-WATSON
FALLECIO HIJO FUNDADOR EMPRESA DE COMPUTADORAS
</TITLE>
<TEXT> Nueva York, 31 dic (EFE).- Thomas Watson junior, hijo del fundador
de International Business Machines Corp. (IBM), falleció hoy,
viernes, en un hospital del estado de Connecticut a los 79 años de
edad, informó un portavoz de la empresa.
Watson falleció en el hospital Greenwich a consecuencia de
complicaciones tras sufrir un ataque cardíaco, añadió la fuente.
El difunto heredó de su padre una empresa dedicada principalmente
a la fabricación de máquinas de escribir y la transformó en una
compañía líder e innovadora en el mercado de las computadoras. EFE
PD/FMR
01/01/00-34/94
</TEXT>
</DOC>
```

Colecciones de Referencia (cont.): Ejemplo de *Qrel*

```
<QID> <ITER> <DOCNO> <REL>

...

44 0 EFE19940722-13111 0
44 0 EFE19940722-13237 0
44 0 EFE19940722-13274 0
44 0 EFE19940723-13494 1
44 0 EFE19940723-13513 1
44 0 EFE19940723-13688 0
44 0 EFE19940724-13915 0
44 0 EFE19940724-14076 1
44 0 EFE19940724-14077 1
44 0 EFE19940724-14084 1
44 0 EFE19940724-14086 1
44 0 EFE19940724-14093 1
44 0 EFE19940724-14098 1
44 0 EFE19940724-14101 0
44 0 EFE19940724-14104 1
44 0 EFE19940724-14130 1

...
```

PLN & IR: Introducción

- **Def. Procesamiento del Lenguaje Natural (PLN):** tratamiento computacional del lenguaje humano
  - Objetivo: computadora comprenda el lenguaje humano
- **IR como tarea de NLP:** "comprender" el contenido de los documentos

PLN & IR: Introducción (cont.)

- **Principal problema de IR: variación lingüística**
  - El mismo concepto puede expresarse de muy diferentes maneras
  - Impide establecer correspondencias
- Diferentes niveles de variación:
  - Morfológica: modificaciones **flexivas** y **derivativas**

    `cantas` / `cantó` `cantar` / `cantante`
  - Semántica: **polisemia**

    `banda (de música)` / `banda (franja)`
  - Léxica: **sinonimia**

    `rápido` / `veloz`
  - Sintáctica: modificaciones de la **estructura sintáctica**

    `cambio climático` / `cambio del clima`

PLN & IR: Introducción (cont.)

- El lenguaje no es un mero repositorio de palabras (*bag-of-terms*)
  - Comunicar conceptos, entidades, y relaciones, de múltiples maneras
  - Las palabras se combinan en unidades lingüísticas de mayor complejidad, cuyo significado no siempre viene dado por el significado de sus palabras componente (*ppo. composicionalidad*)
- **Solución: técnicas de NLP**
  - Principalmente para **inglés**

Tratamiento de la Variación Lingüística

- En general, dos enfoques diferenciados:
  - **Normalización:** reducir las diferentes variantes de un término a una *forma canónica* común
    - Ej. sustituir una palabra por su *stem* (*stemming*) o lema (*lematización*)
  - **Expansión:** añadir a la consulta variantes de sus términos originales
    - Ej. añadir sinónimos

Tratamiento de la Variación Morfológica: *Stemming* (cont.)

- Tratado anteriormente ▸ Normalización: stemming
- Ventajas
  - Simplicidad
- Desventajas:
  - Problemas con idiomas de morfología compleja. Ej. español:
    - Adjetivos/nombres: +20 grupos variación género +10 grupos número
    - Verbos: 3 grupos regulares, ±40 irregulares; 118 formas flexivas cada grupo
  - Pérdida de información de cara a procesamiento futuro
  - Sobre-*stemming*: palabras no relacionadas dan igual *stem*

$$\left.\begin{array}{l} \textit{general} \\ \textit{generous} \end{array}\right\} \rightarrow \texttt{gener-}$$

  - Sobre-stemming: palabras sí relacionadas dan *stems* diferentes

$$\begin{array}{ll} \textit{recognize} & \rightarrow \texttt{recogn-} \\ \textit{recognition} & \rightarrow \texttt{recognit-} \end{array}$$

Tratamiento de la Variación Morfológica (cont.): Otros

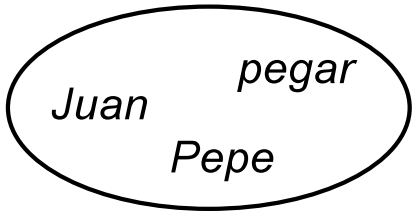
- Expansión de la consulta con variantes:
  - Google: busca simultáneamente el término en singular y plural
- **Lematización:** sustituir palabra por su lema
  - Mejora resultados con idiomas de morfología compleja
  - Reduce la pérdida de información

Tratamiento de la Variación Léxico-Semántica

- Técnicas de *desambiguación del sentido de las palabras*
  - Necesaria alta efectividad: ∼90% (¿60%?)
- **WordNet/EuroWordNet** principalmente
- Aproximaciones:
  - Indexación por sentidos (*synsets*) en lugar de palabras
  - Distancias conceptuales
  - **Expandir consulta con términos relacionados** (sinónimos, etc.)
    - Poco efectivas salvo con consultas cortas o incompletas
    - Google: operador ∼ (por ahora sólo inglés)
      ∼ape → {ape, monkey, gorilla, chimpanzee, ...}

Tratamiento de la Variación Sintáctica: Introducción

- **Problema:** *bag-of-terms* insuficiente:



$$\begin{cases} \textit{Juan pegó a Pepe ?} \\ \textit{Pepe pegó a Juan ?} \end{cases}$$

- **2 aproximaciones:**
  1. Representaciones complejas en base a estructuras sintácticas: árboles y grafos
     - Coste muy alto: inadecuado para uso práctico
  2. **Frases como términos índice:**
     - **Hipótesis:** frases denotan conceptos/entidades más significativos que las palabras
     - Términos más precisos y descriptivos
     - Uso combinado con palabras

Tratamiento de la Variación Sintáctica (cont.): Identificación y Extracción

- Técnicas estadísticas
  - Secuencias de palabras coocurren frecuentemente
  - Análisis estadístico (frecuencias, coocurrencias, etc.)
  - No base lingüística (a veces resultados extraños)
  - Mayor simplicidad
- Sintácticas
  - Secuencias de palabras satisfacen relaciones sintácticas
  - Análisis sintáctico (complejidad diversa)
  - Sí base lingüística (teóricamente superiores)
  - Mayor complejidad
- Aproximar sintaxis mediante distancias
  - Palabras cercanas se suponen relacionadas sintácticamente

Tratamiento de la Variación Sintáctica (cont.): Representación y Correspondencias

- Como conjuntos de palabras
- Almacenar árbol de análisis
  - Técnicas de comparación de árboles: gran complejidad
- Almacenar sólo las relaciones sintácticas interesantes
  - Sustantico–modificador
  - Sujeto–verbo
  - Verbo–Objeto
  - ...

Tratamiento de la Variación Sintáctica (cont.): Buscadores

- Operador de frase (secuencia exacta)
  - Google/Yahoo: `""`
  - Ej. `"coche rojo"`
- Operador comodín (palabra completa)
  - Google/Yahoo: `*`
  - Conjuntamente con operadores de frase. Ej.:

    `"* tomates"`={`"dos tomates"`, `"varios tomates"`, etc.}
    `"* * tomates"`={`"los dos tomates"`, `"quiero varios tomates"`, etc.}

  - Buscar citas aproximadas
- Operador de proximidad:
  - Anteriormente explícito (`NEAR`, `ADJ`): exige que la otra palabra esté dentro de un radio dado
  - Actualmente implícito: la proximidad entre sí dentro del documento de los términos de la consulta aumenta la relevancia

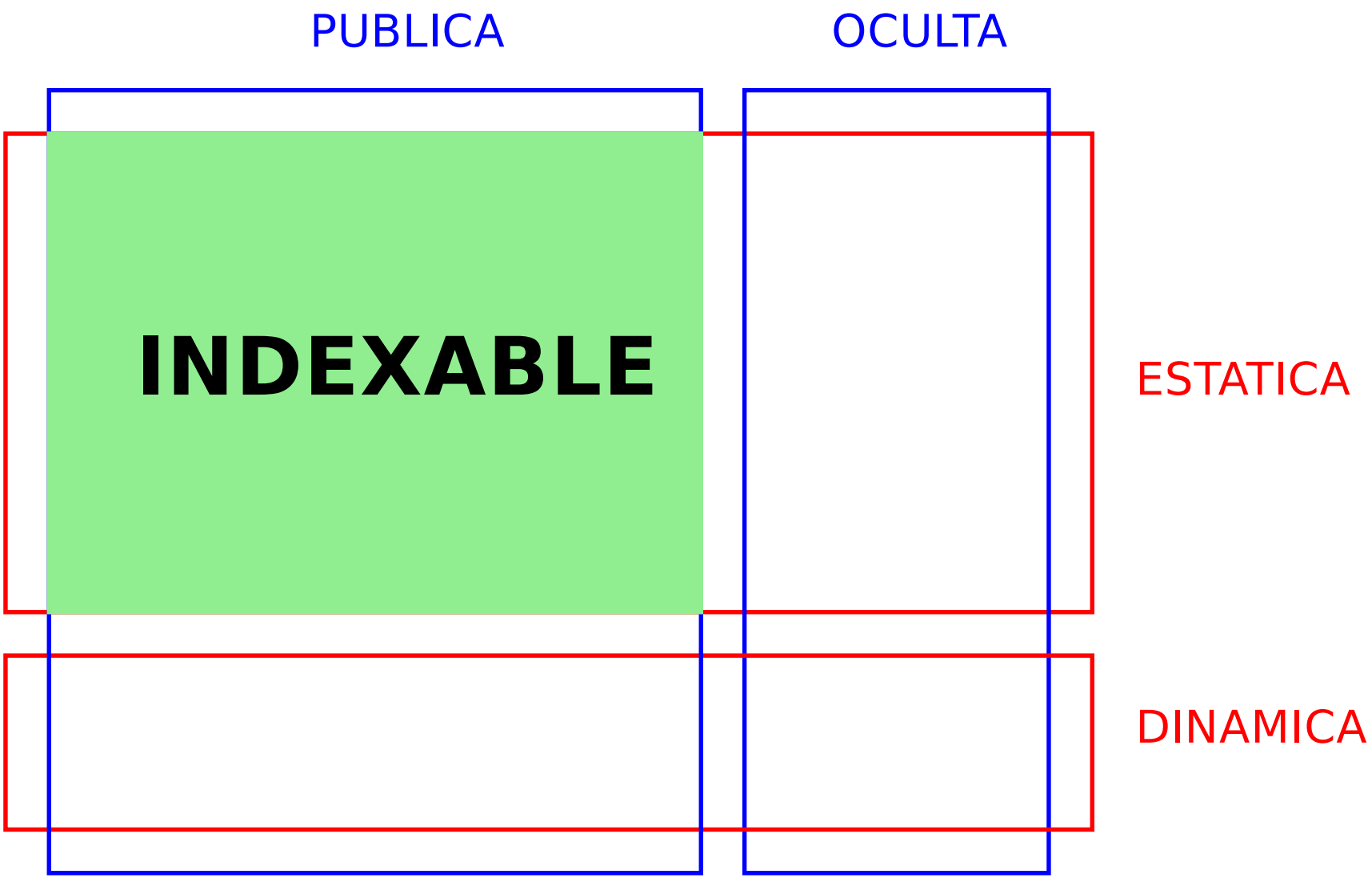
Búsqueda de Información en la Web: Introducción

- Tamaño web (febrero 2007): ∼30,000,000,000 páginas
- ¿Cómo encontrar algo?
- Sitios web especializados en buscar otros sitios web (**buscadores**):
  - **Directorios**: jerarquizados por temas y categorías

    Google Directory (`http://directory.google.com/`)
    Yahoo! Directory (`http://dir.yahoo.com`)
  - **Motores de búsqueda** (o buscadores): búsqueda por palabras clave

    Google (`http://www.google.es`)
    Yahoo! (`http://www.yahoo.es`)

Estructura de la Web



Breve Historia de los Buscadores

1990	Archie	Primer buscador de Internet (FTPs)
1992 Dic	Veronica	Buscador de Gopher (menús jerarquizados)
1993 Jun	Wanderer	Primer buscador web
1993 Dic	RBSE	Primero en calcular medida relevancia
1994 Ene	Galaxy	Primer directorio
1994 Abr	Yahoo	Directorio revisado manualmente
1994 Abr	WebCrawler	Salto tecnológico: indexar texto completo
1994 Jul	Lycos	Índice masivo
1995 Feb	Infoseek	Netscape. Amigable, servicios adicionales
1995 Jun	Metacrawler	Primer metabuscador
1995 Dic	Altavista	Muy veloz. Lenguaje natural y ops. lógicos

Breve Historia de los Buscadores (cont.)

1996 Abr	Olé	Primer buscador hispano
1996 May	HotBot	Tecnología de búsqueda de alto rendimiento
1998	MSN Search	Buscador de Microsoft
1998 Sep	Google	Nuevo salto tecnológico: algoritmo *pagerank*
1999	Baidu	Buscador chino
2005 Nov	Live Search	Nueva plataforma *Windows Live* de Microsoft
2006	Quaero	"Buscador europeo"

`http://manuales.ojobuscador.com/historia`

Directorios

- **Def.:** sitio web que contiene un índice o lista de páginas web estructuradas jerárquicamente en base a categorías y subcategorías temáticas

  Yahoo! Directory (`http://dir.yahoo.com`)
  Google Directory (`http://directory.google.com/`)

- Estructura navegable
- Generalmente creado/revisado a mano
  - Categorización automática
- Han ido perdiendo importancia frente a los motores de búsqueda
- Actualmente son un "complemento" a éstos
- Para búsquedas muy generales

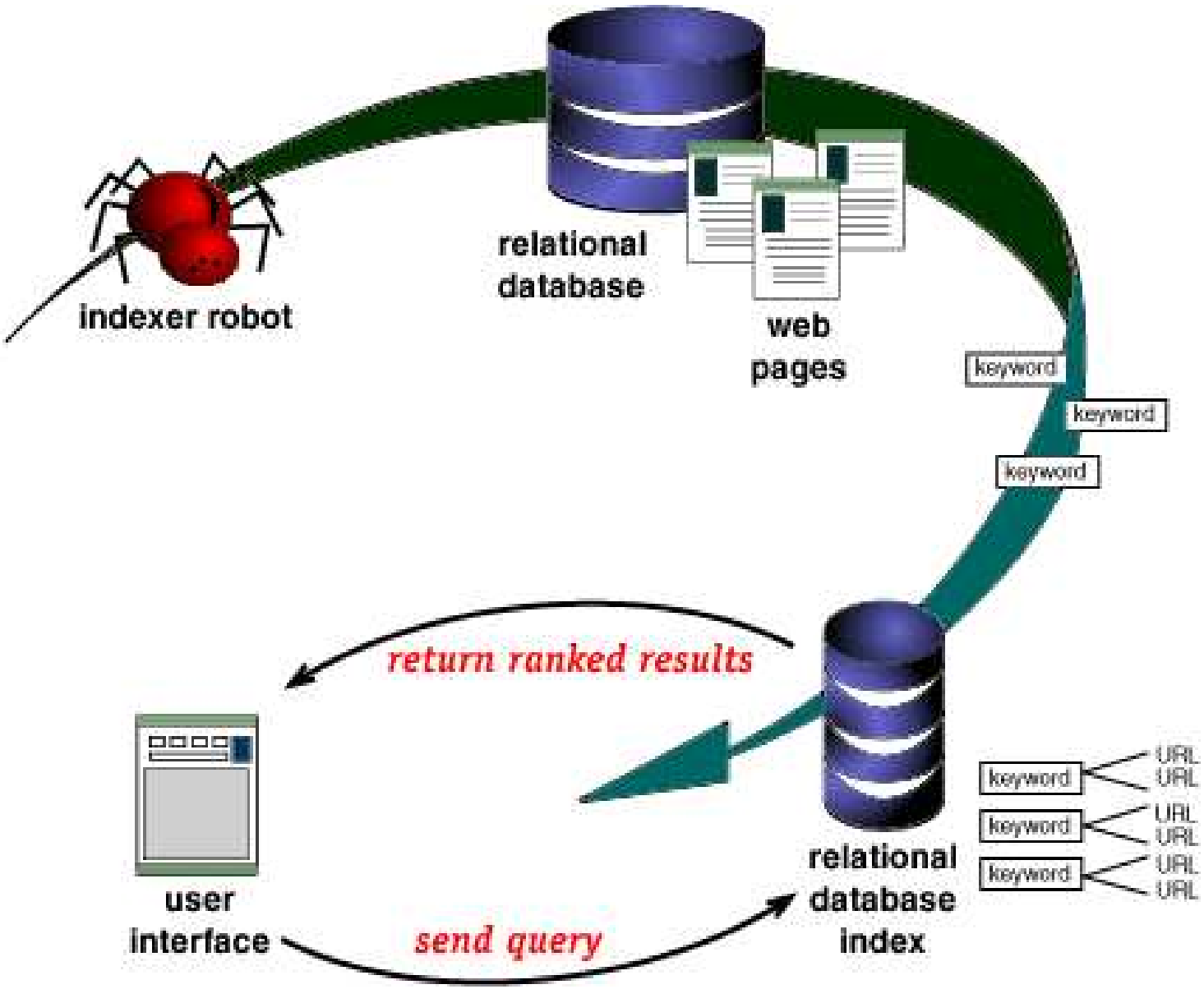
Motores de Búsqueda

- **Def.:** sitio web que contiene una base de datos (índice) donde las páginas web han sido indexadas en base a palabras clave y sobre la cual podemos realizar búsquedas (consultas o *queries*)

  Google (`http://www.google.es`)
  Yahoo! (`http://www.yahoo.es`)

- Similar a **recuperación ad hoc**
- Para búsquedas concretas

Motores de Búsqueda (cont.): Funcionamiento



Motores de Búsqueda (cont.): Arquitectura

- **Robots:** Programas que recorren la red buscando documentos:
  - Analizan su contenido (total o parcial)
  - Devuelven las palabras clave o descriptores que lo describen (a indexar)
- **Base de datos:** índice de palabras clave o descriptores asociados a cada documento
  - Actualización periódica (robots)
- **Interfaz de consulta:** parte que ve el usuario
  - Introducir consulta
  - Presentar resultados

Buscadores Instalables

- Permiten indexar y buscar tus propios contenidos
- "*Spin-offs*" de los principales buscadores
  - Sólo contenidos del PC del usuario
  - Google Desktop (`http://desktop.google.es`)
  - Yahoo! Desktop Search (`http://desktop.yahoo.com`)
- Motores de búsqueda independientes
  - Algunos también contenidos web
  - `http://en.wikipedia.org/wiki/Information_retrieval`

PageRank: Introducción

- Los modelos tradicionales de RI no tienen en cuenta la estructura de hipertexto de la red
- Un buen modelo de RI para web debe tene en cuenta:
  - La estructura de las páginas web
  - El texto de los hiperenlaces (que se asocia a la página de destino)
  - La persistencia en el tiempo de una página
  - La **popularidad** de una página web
- El **algoritmo PageRank de Google** permite calcular la popularidad de una página web en base a los enlaces que apuntan a ella
- Desarrollado en la Univ. de Stanford por Larry Page y Sergei Brin:

  S. Brin & L. Page. (1998). The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *WWW7/Computer Networks* 30(1-7):107-117. `http://dbpubs.stanford.edu:8090/pub/1998-8`.

PageRank: Algoritmo

- Algoritmo de análisis de grafos que asigna a cada página web un **valor numérico (PageRank) en función de su "popularidad"**
- **Formalmente**, el *PageRank* es una distribución de probabilidades que pretende representar la probabilidad de que un usuario llegue a una página en particular recorriendo enlaces de forma aleatoria.
  - e.g., un *PageRank* de 0.5 indica que existe un 50% de probabilidades de que una persona alcance dicha página pulsando links al azar
- **Informalmente**, **enlace de página B a página A = "voto" de página B a A**
  - Una página con muchos votos debería ser muy popular/importante
  - Un voto desde una página muy popular vale más que un voto desde una página poco popular: es un **algoritmo retroalimentado** que requiere de varias pasadas (iteraciones)
- **Matemáticamente**, el *pagerank* de una página es un **valor del autovector principal de la matriz de adyacencia de la web**...

PageRank: Algoritmo (cont.)

- . . . lo que se traduce en la fórmula

$$PR(A) = \frac{1-d}{N} + d\left(\frac{PR(B_1)}{L(B_1)} + \frac{PR(B_2)}{L(B_2)} + \ldots + \frac{PR(B_m)}{L(B_m)}\right)$$

donde

  - A es la página cuyo *pagerank* vamos a calcular
  - B_i son las páginas que contienen enlaces a A
  - $PR(X)$ es el *pagerank* de la página web X
  - $L(X)$ es el número de enlaces diferentes de la página X
  - d es el probabilidad de que una persona siga los enlaces de cualquier página (alrededor de 0,85), por lo que $1-d$ es la probabilidad de que salte arbitrariamente a cualquier otra página
  - N es el número total de páginas web

- La fórmula se recalcula iterativamente hasta converger (los valores se estabilizan)

Links sobre PageRank

- Phil Craven. Google's PageRank Explained and how to make the most of it:
  `http://www.webworkshop.net/pagerank.html`
- Ian Rogers. The Google Pagerank Algorithm and How It Works:
  `http://www.ianrogers.net/google-page-rank/`
- **Demo gráfica online:**
  `http://www.search-this.com/pagerank-decoder/`
- Calulcadora:
  `http://www.webworkshop.net/pagerank_calculator.php3`
- Demo gráfica Matlab:

`http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=14258&objectType=file`