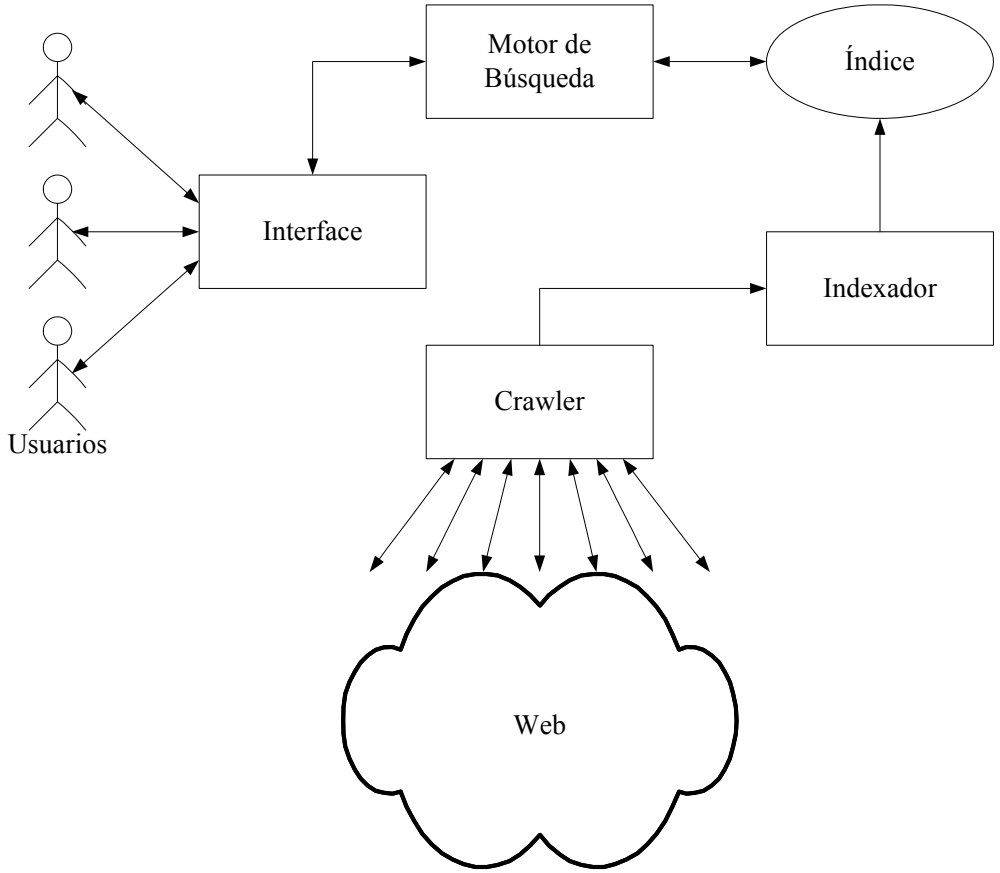
# Sistema de Recuperación de Información (SRI)

UCR – ECCI

CI-2414 Recuperación de Información

Prof. Kryscia Daviana Ramírez Benavides

# Funcionamiento de un Buscador



# Componentes Básicos

- *Crawler:* Recorre la Web buscando las páginas a indexar.
- *Indexador:* Mantiene un índice con esa información.
- *Motor de Búsqueda:* Realiza las búsquedas en el índice.
- *Interfaz:* Interactúa con el usuario.

# Componentes Básicos (cont.)

- Estos componentes son de una *arquitectura centralizada*.
- El *crawler* (robot, spider, entre otros) corre *localmente* en la máquina de búsqueda, recorriendo la Web mediante pedidos a los servers y trayendo el texto de las páginas Web que va encontrando.
- El *indexador* corre *localmente* y mantiene un *índice* sobre las páginas que le trae el *crawler*.
- El *motor de búsqueda* también corre *localmente* y realiza las búsquedas en el *índice*, retornando básicamente los *URLs rankeados*.

# Componentes Básicos (cont.)

- La *interfaz* corre en *el cliente* (cualquier parte de la Web) y se encarga de recibir la consulta, mostrar los resultados, retroalimentación, etc.

# Componentes Básicos: Crawler

- También se le llama *robot* o *araña* (*spider*).
- Se ejecuta en una máquina local y envía peticiones a los servidores Web, visitas periódicas.
- Comienza con un conjunto de *URLs*, ya sea haciendo un recorrido en *profundidad* o *anchura*.
- Permite que se le proporcionen direcciones de sitios Web.

# Componentes Básicos: Crawler (cont.)

- Los buscadores combinan las técnicas anteriores con medidas de *popularidad* para decidir el orden en que se visitan las páginas.
- El objetivo es recorrer las páginas de *mayor calidad*.
- Popularidad se mide como el número de enlaces que apuntan a la página.
- Límite en la profundidad por sitio Web es el límite al número de páginas que puede devolvernos el buscador.
- *Frecuencia* entre visitas es *variable* (días – meses).

# Componentes Básicos: Crawler (cont.)

- En el intervalo entre actualizaciones los buscadores pueden devolver *enlaces inválidos* (porcentaje de 2-9%).
- Los administradores de sitios Web pueden controlar el comportamiento de los *crawlers*, por ejemplo: impedirles que indexen determinadas páginas.
- No se indexan páginas generadas dinámicamente o protegidas con contraseñas.
- Los *crawlers* pueden tener *problemas* para indexar páginas con *frames* o mapas de imágenes.

# Componentes Básicos: Indexador

- Encargado de obtener la representación interna de cada página encontrada por el *crawler*.
- Hasta que una página no ha sido indexada no está disponible para ser devuelta como resultado de una búsqueda.
- Técnicas de análisis:
  - Lista de parada (palabras vacías, *stopwords*).
  - Extracción de raíces.
- Organización de los índices: *archivos invertidos*
  - Lista ordenada de palabras o vocabulario.
  - Documentos en que aparece cada una de las palabras.

# Componentes Básicos: Indexador (cont.)

- *Información básica:* Frecuencia de aparición de cada palabra en cada página Web.
- *Información adicional (Google):* Posición de la palabra en la página, uso de mayúsculas, tipo de letra.
  - *Ventajas:*
    - Consultas de proximidad o por grupos de palabra.
    - Técnicas de pesado más elaboradas (título o primeras líneas, negritas o cursiva)
  - *Desventaja:*
    - Costo de espacio.
- *Información de cada página:* fecha de creación, tamaño, título, primeras líneas.

# Componentes Básicos: Indexador (cont.)

- Texto ligado a los enlaces:
  - *Mayoría de buscadores:* Asocian el texto con las páginas en la que aparece.
  - *Google:* Además lo asocia con la página a la que hace referencia el enlace.
    - *Razones:*
      - Descripciones precisas del contenido de una página.
      - Pueden corresponder a documentos que no pueden ser indexados por un motor de búsqueda (imágenes, programas).
      - Aumenta la cobertura sobre la Web.
    - *Desventaja:*
      - Se incrementa el riesgo de devolver páginas con problemas (no se comprueba la validez del enlace).

# Componentes Básicos: Motor de Búsqueda

- *Misión:*
  - Analizar la consulta de usuario.
  - Buscar en el índice las páginas relacionadas.
  - Ordenarlas según la relevancia estimada, criterios de localización, frecuencia de aparición y popularidad de las páginas.
- *Localización:*
  - La hipótesis es que cualquier página relevante a una consulta mencionará las palabras utilizadas en dicha consulta desde el comienzo.
  - Relevancia mayor cuando las palabras aparecen en el título o primeras líneas.

# Componentes Básicos: Motor de Búsqueda (cont.)

- *Frecuencia o número de apariciones de las palabras de la consulta en la página:*
  - A mayor frecuencia se da mayor relevancia.
  - Se impone un *límite* a la frecuencia que se tiene en cuenta:
    - Pueden descartarse algunas páginas si incluyen palabras con frecuencia excesivamente elevada.
    - Tiene como objetivo evitar los intentos de mejorar la posición de una página en las listas de resultado.
- *Popularidad:*
  - La hipótesis es que cuando mayor sea el número de hiperenlaces que apuntan a una página, mejor es esa página.
  - El inconveniente es que no se distingue la calidad de la página donde aparece el hiperenlace.

# Componentes Básicos: Motor de Búsqueda (cont.)

- *PageRank de Google:*
  - *PageRank* de una página en función de:
    - Número de enlaces que apuntan a la página.
    - Calidad de las páginas que la apuntan (su *PageRank*).
    - Número de enlaces que salen de dichas páginas
  - El resultado se califica como páginas de calidad aquellas citadas en pocos sitios si dichos sitios tienen a su vez una alta calidad.
- *DirectHit de Excvite:*
  - Cuenta las páginas que visita cada usuario después de realizar una consulta.
  - El número de usuarios que visita cada página se utiliza para calcular el grado de relevancia.

# Componentes Básicos: Motor de Búsqueda (cont.)

- Otros elementos para calcular relevancia:
  - Tipo de letra de las palabras: tamaño, cursiva, negrita.
  - En consultas de varias palabras: proximidad dentro de la página.

# Componentes Básicos: Interfaz de Usuario

- Se divide en dos partes:
  - Interfaz de consulta.
  - Interfaz de respuesta.
- *Interfaz de consulta básica:*
  - Caja de texto donde el usuario puede escribir una o varias palabras.
  - Misma interfaz aunque puede variar la *semántica*:
    - Páginas que contenga *al menos una de las palabras de la consulta*, por ejemplo: AltaVista, Excite.
    - Páginas que incluyan *todos los términos de la consulta*, por ejemplo: Google, HotBot.

# Componentes Básicos: Interfaz de Usuario (cont.)

- *Interfaz de consulta básica:*
  - Falta de conocimiento del usuario sobre la visión lógica del texto:
    - Utilización de listas de parada, extracción de raíces.
    - Diferenciación mayúsculas/minúsculas.
  - Esta información aparece en la ayuda. Pero casi ningún usuario la lee.
- *Búsquedas complejas:*
  - Operadores booleanos, comodines, búsquedas en expresiones o en proximidad.
  - Interfaces avanzadas: idioma, ámbito geográfico, dominio de Internet, rango de fechas, tipo específico de datos, etc.

# Componentes Básicos: Interfaz de Usuario (cont.)

- *Interfaz de respuesta:*
  - Diez páginas más relevantes.
  - Para cada página: URL, tamaño, fecha indexación, título y primeras líneas.
  - Google:
    - Extracto que incluye algunas o varias de las palabras de la consulta, esto muestra el contexto en el que se usan dichas palabras.
    - Acceso a la copia de la página (almacenada en el momento de la indexación).
- Otras posibilidades:
  - Cambiar el número de documentos devueltos, orden.
  - Buscar documentos similares a otro.
  - Refinar la búsqueda.
  - Traducción automática.

# Problemas con la Arquitectura Centralizada

- El *crawling* es la actividad más lenta, y posiblemente en el futuro el tamaño de la Web y su dinamismo hagan imposible que se realice en forma centralizada. Actualmente las mejores máquinas indexan sólo un 26% de la Web. Esto no se resuelve haciendo *crawling* con varias máquinas.
- Los servidores Web se saturan con pedidos de diferentes *crawlers*.
- Las redes se saturan innecesariamente porque los *crawlers* traen toda la página, para luego descartar casi todo su contenido.
- Cada *crawler* recolecta independientemente, sin cooperar.

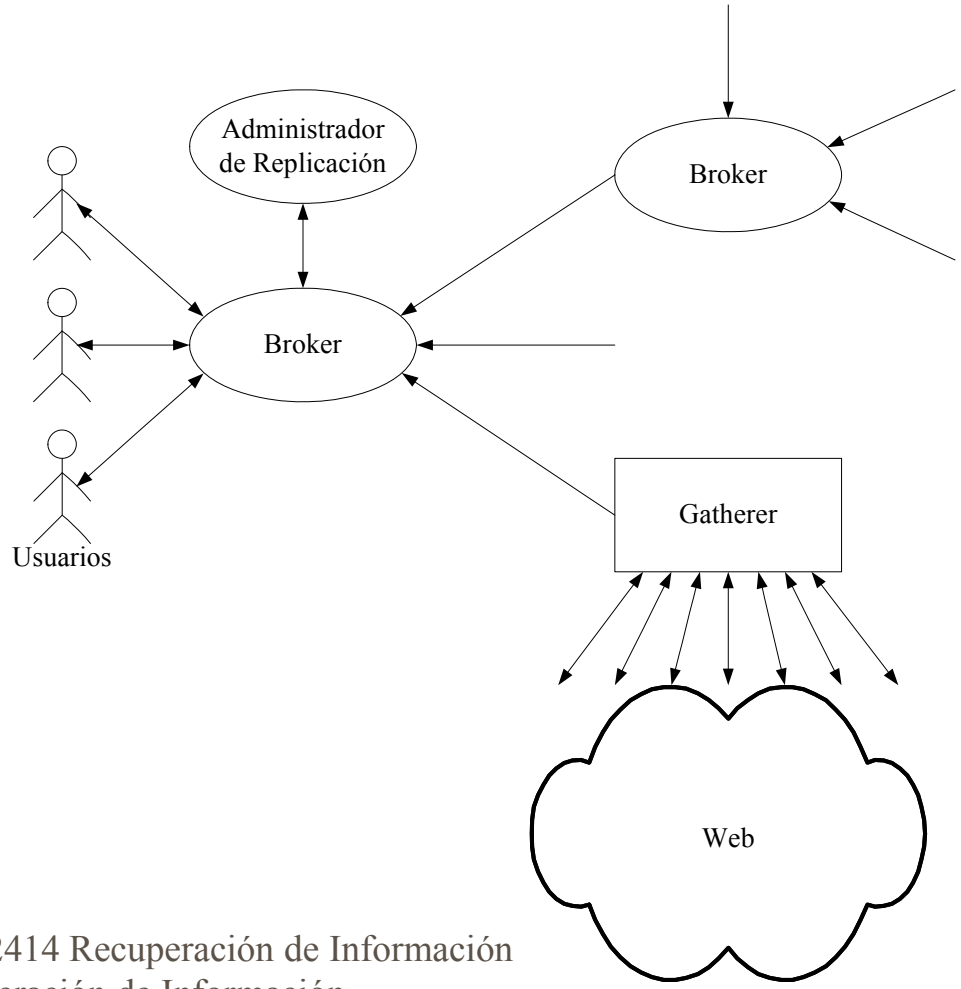
# Arquitectura Distribuida

- Se basa en dos tipos de componentes:
  - *Gatherer:* Recolecta páginas de ciertos servidores Web y extrae la información a indexar, periódicamente.
  - *Broker:* Indexa la información de determinados *gatherers* y otros *brokers* y responde consultas.
- Se pueden organizar de distintas formas. Por ejemplo: un *gatherer* puede ser local y no generar tráfico, y enviar la información a varios *brokers*, evitando repetir la recolección.
- Los *brokers* pueden construir índices y enviar esos índices a otros *brokers*, que emergerán varios, evitando transmitir tanta información y re-indexar tanto.

# Arquitectura Distribuida (cont.)

- El problema es que requiere cooperación de parte de los sitios Web.
- Esta arquitectura se está utilizando en diferentes organizaciones: CIA, NASA, agencias de gobierno de USA, etc.
- Las máquinas de propósito general siguen confiando en la arquitectura centralizada.

# Arquitectura Distribuida (cont.)



# Referencias Bibliográficas

- La información fue tomada de:
  - Libro de texto del curso.