Capítulo 4
Estrategias de Indexación Paralelas

4.1. Arquitectura del Servidor

Un sistema distribuido utiliza un modelo de red de $workstations$ o estaciones de trabajo. Las workstations están fuertemente conectadas por medio de una tecnología de red $switching$. Las estaciones de trabajo poseen su propia memoria y disco local. La ventaja de este modelo distribuido, es que todas las comunicaciones entre los procesadores se realizan a través del paso de mensajes, que elimina la interferencia del proceso de control de memoria del sistema operativo, y que los discos locales son accedidos directamente por los procesadores sin necesidad de atravesar la red. La Figura 4.1 muestra el modelo de red de workstations.

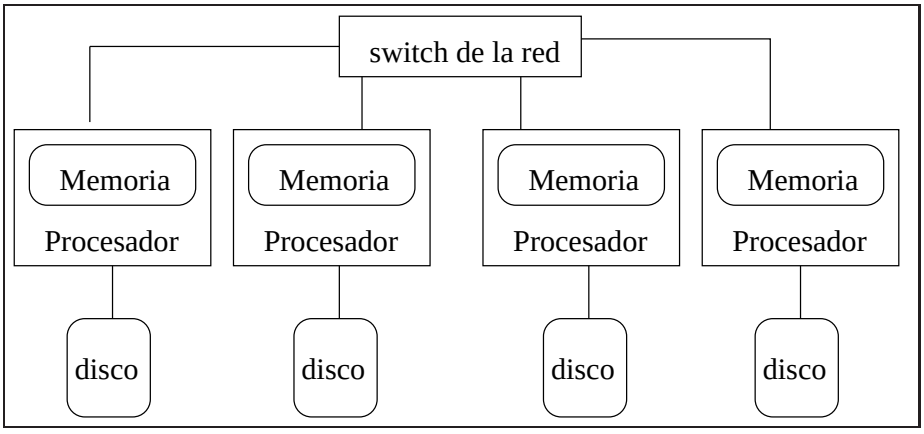

Figura 4.1: Modelo de red de workstations.

El sistema adopta un paradigma de cliente-servidor donde uno o varios procesos clientes requieren servicio de un proceso servidor. Un proceso servidor normal-

mente está escuchando requerimientos desde una dirección conocida. Es decir que el servidor permanece dormido hasta que se solicita una conexión. En ese momento el servidor se “despierta” y sirve al cliente, realizando las acciones apropiadas.

De acuerdo a estas propiedades, el servidor es una entidad pasiva, escuchando los requerimientos de conexión, mientras que los clientes son entidades activas, inicializando conexiones. En el modelo utilizado para este trabajo, el servicio consiste en procesar las consultas. Para ello, el servidor debe acceder a la base de datos de texto, y resolver las consultas provenientes de diferentes usuarios de la red. Como se muestra en la Figura 4.2, se asume un servidor compuesto por P procesadores y al menos una máquina broker que actúa como intermediaria entre los procesadores del servidor y los usuarios. La máquina broker es la encargada de recibir las consultas provenientes de los usuarios, y con alguna metodología debe enviarlas a las máquinas del servidor.

El broker ofrece un servicio concurrente para las consultas que arriban al sistema. Está constituido por diferentes tareas que en paralelo envían varias consultas al servidor y reciben los resultados.

Por cada consulta recibida, uno de los P procesadores del servidor cumple el rol de máquina ranker, la cual es seleccionada por el broker durante la distribución de la consulta. Esta máquina es la encargada de realizar el ranking final de documentos que serán entregados como resultado al usuario.

La máquina broker, dependiendo de la estrategia de distribución que se esté utilizando, también debe seleccionar una máquina receptora, a través de la cual, las consultas arribarán a servidor BSP.

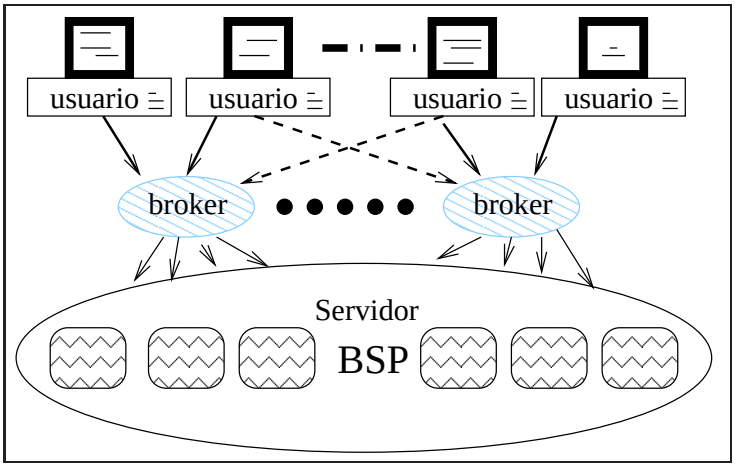

Figura 4.2: Arquitectura del servidor BSP.

4.2. Relación Broker-Servidor

Los clientes solicitan requerimientos a una o mas máquinas broker, que a su vez distribuyen estos requerimientos entre los procesadores del servidor. Los requerimientos son consultas que se resuelven utilizando la lista invertida como estructura de datos.

Para poder explotar el paralelismo disponible, se minimiza la cantidad de trabajo secuencial realizada por la máquina broker. Se restringe su funcionalidad a recibir los requerimientos de los usuarios, distribuir las consultas entre los procesadores, recibir los identificadores de los documentos mejor rankeados y pasarlos al usuario.

Las dos operaciones básicas relacionadas con las respuestas a las consultas de los usuarios, están a cargo del servidor paralelo. Esto es, la recuperación de los identificadores de los documentos y sus respectivos rankings. Ambas tareas son realizadas en paralelo, donde el broker es responsable de planificarlas de manera tal que mantenga un balance de carga en los procesadores tan cercano a $1/P$ como sea posible. Es decir que todos los procesadores tengan aproximadamente

la misma cantidad de trabajo.

Por cada consulta, el broker utiliza una estrategia de balance de carga particular denominada *the last loaded processor*[71], para elegir un procesador donde se realizará el ranking final de documentos (procesador ranker). Posteriormente este procesador le envía al broker la lista de los identificadores de documentos mejor rankeados. Como es probable que los términos de una consulta se encuentren ubicados en distintos procesadores, el broker selecciona al ranker (utilizando la estrategia mencionada)de manera tal que se mantenga un buen balance de carga. Para ello, las operaciones de ranking son distribuidas uniformemente entre los procesadores. Esto se realiza manteniendo un contador por cada procesador. Estos contadores mantienen el número de operaciones de ranking planificadas para consultas que no han sido terminadas aún. La tarea de ranking para una consulta dada es planificada en un procesador que posea información sobre los términos de la misma, y de éstos se selecciona el que tiene el menor valor en el contador. La Figura 4.3 muestra el pseudo-código ejecutado por el broker.

Los detalles de las operaciones ejecutadas por el servidor paralelo BSP son descriptas en las siguientes secciones y están condicionados a los diseños de las estrategias de listas invertidas que se utilice. En general, el servidor ejecuta una secuencia de superpasos en las cuales se procesan lotes de consultas en cada procesador junto con la operación de ranking de documentos.

4.3. Estrategia de Indexación Local

El proceso para crear las listas invertidas, utilizando una organización local de los documentos, es sencillo ya que una vez que se dispone de los documentos de interés, de los cuales se asume que se encuentran en una base de datos de texto, sólo se debe hacer un recorrido sobre ellos para obtener sus términos y luego

```
1:  while (true) do
2:      msg = RecibeMensaje();                  ▷ Espera hasta recibir un mensaje
3:      SWITCH(msg.origen())                    ▷ De donde proviene el mensaje
4:      Case USER:                              ▷ Es un nuevo mensaje
5:          proc=rankingProcesador(msg.ConsultaTérminos)   ▷ Obtiene el
                procesador ranking
6:          for (cada término en msg.ConsultaTérminos) do
7:              term.ranker(proc)          ▷ Le hace conocer el procesador ranker
8:              Distribuye los términos de acuerdo a la estrategia de particionado
                    de los datos y el índice
9:          end for
10:     EndCase
11:     Case SERVIDOR:                          ▷ Llega una respuesta del servidor
12:         Actualizar los contadores
13:         Enviar la respuesta al usuario
14:     EndCase
15: end while
```

Figura 4.3: Pseudo-código ejecutado por el broker.

incorporarlos en la lista invertida. Es decir que las listas invertidas se construyen basándose en los documentos que cada procesador posee. Cada máquina busca en dichos documentos los términos de interés, calculando la cantidad de veces que el término se repite.

Indexador

La estrategia local es la mas sencilla porque, como se explicó anteriormente, las listas son construidas utilizando los documentos que cada procesador posee. Cada máquina contiene una tabla de vocabulario con T términos, y asumiendo que la lista asociada secuencial para un término t tiene un largo N, en promedio cada lista para el mismo término en cada procesador tendrá un largo cercano a N/P. Las listas locales en cada procesador son ordenadas lexicográficamente para facilitar la posterior resolución de consultas.

Para reducir la carga de trabajo que soportan los procesadores del servidor

durante la resolución de consultas, el indexador es el encargado de actualizar las listas invertidas, y para ello mantiene una cola con las URLs de los documentos recuperados por el crawler, donde los documentos referenciados por las URLs son procesados uno a la vez, debido a la forma en que los procesadores del servidor trabajan (cada procesador construye su propia lista invertida usando los documentos locales).

Por lo tanto en esta estrategia, la tarea del indexador consiste en seleccionar la máquina que recibirá los términos de un documento junto con sus correspondientes listas asociadas.

El indexador debe procesar los documentos (operación de parsing) para extraer los términos relevantes junto con cierta información adicional necesaria para realizar la operación de ranking (identificador de documento, frecuencia, etc.). Durante la operación de parsing, el indexador excluye las palabras vacías o $stopwords$, para ganar espacio e incrementar la velocidad de búsqueda, debido a que son palabras que pretenden ser spam.

Luego de que cada documento es procesado, cada palabra es convertida en un identificador de palabra $wordID$ y se agrega a una lista invertida temporal que es enviada al procesador seleccionado, quien tiene que agregar esta nueva información a su lista local (ver Figura 4.4). Como consecuencia, este proceso implica un alto grado de comunicación entre los procesadores del servidor BSP y el indexador.

Los procesadores que actualizan sus listas locales, pueden ser seleccionados en forma aleatoria o mediante una función de hash, pero para poder controlar que todos los procesadores mantengan aproximadamente la misma cantidad de documentos procesados, el indexador puede tener un contador para cada máquina que indique el número de documentos procesados. Por lo tanto, al momento de en-

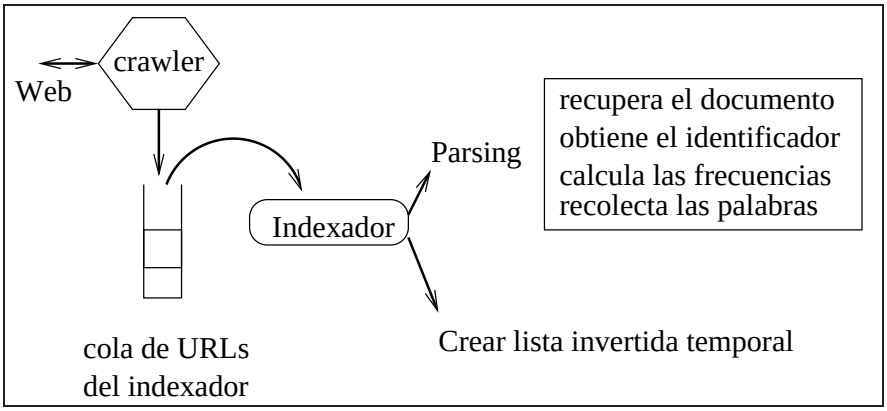

Figura 4.4: Tarea del Indexador.

viar los términos de un nuevo documento ingresado en el sistema, el indexador selecciona un procesador cuyo contador posea el menor valor.

Máquina de Búsqueda

Debido a cómo se construyen las listas invertidas en esta estrategia, una consulta formada por un único término debe ser resuelta computando paralelamente las listas invertidas locales de cada máquina que forma parte del servidor.

Para procesar una consulta generada por una máquina del usuario, ésta es transmitida a la máquina broker, quien debe elegir una máquina del servidor (máquina receptora) para enviarle la consulta. La máquina receptora es seleccionada en forma circular, es decir que la primera consulta generada por la máquina de usuario será recibida por la máquina 1 del servidor, la segunda consulta será recibida por la máquina 2 del servidor y así sucesivamente. La máquina broker también es la encargada de seleccionar la máquina ranker del servidor, que realizará el ranking final para seleccionar los mejores documentos. Esta última es seleccionada teniendo en cuenta la carga de trabajo que cada máquina del servidor posee [26].

Una vez que la máquina del servidor, escogida por el broker, recibe la nue-

va consulta, el procesamiento de dicha consulta siguiendo el modelo BSP es como se muestra en la Figura 4.5.

```
1: Start [Primer Superpaso](Máquina receptora)
2:     Realizar un broadcast de la consulta
3: Ending
4: Start [Segundo Superpaso](Todas las máquinas del servidor BSP)
5:     Buscar en sus listas invertidas locales los términos pertenecientes a dicha
           consulta
6:     if (existen documentos que contengan algún término) then
7:         preranking : recupera los documentos locales
8:         Enviar a la máquina ranker la información obtenida
9:     end if
10: Ending
11: Start [Tercer Superpaso](Máquina ranker)
12:     Recibir los resultados parciales
13:     ranking final: seleccionar los mejores documentos
14:     Enviar el resultado al broker
15: Ending
```

Figura 4.5: Procesamiento de consultas: estrategia local.

4.4. Estrategia de Indexación Global

La estrategia de indexación global consiste, a diferencia de la anterior, en distribuir uniformemente entre los procesadores del servidor, en forma aleatoria (hashing), cada palabra de la tabla de vocabulario junto con sus listas asociadas. Es decir, que la colección completa de documentos es utilizada para construir una única tabla de vocabulario que es idéntica a la secuencial para luego repartir la información que ésta posee entre las máquinas del servidor.

Luego de mapear los términos a los procesadores, cada uno tendrá aproximadamente T/P términos, donde P es la cantidad de máquinas y T el número de palabras diferentes encontradas en los documentos.