



TAREA #2

Bases de Datos Distribuidas

Descripción breve

Fragmentación horizontal primaria en bases de datos relacionales de Tamer Ozsü

- Urrutia González Brenda
 - Olea García Alan
- Juárez Anguiano Mario Alexis

Equipo 03
02/06/2025

Fragmentación Horizontal Primaria en Bases de Datos Relacionales

(Basado en Tamer Özsu, Principles of Distributed Database Systems, 3ª ed., 2011)

La fragmentación horizontal primaria consiste en dividir una relación (tabla) en subconjuntos de filas (tuplas), llamados fragmentos horizontales. Cada fragmento contiene un subconjunto de las filas de la relación original, de acuerdo a ciertos criterios, como condiciones de selección.

¿Qué es el algoritmo COM_MIN?

COM_MIN es un algoritmo para diseñar fragmentación horizontal primaria que minimiza la comunicación entre nodos en un sistema distribuido.

Se basa en analizar las consultas que se ejecutan con frecuencia para decidir cómo fragmentar.

Utiliza la información de accesos (qué consultas acceden a qué filas) para definir predicados de fragmentación.

Su objetivo es que las consultas que se ejecutan localmente tengan que comunicarse lo menos posible con otros nodos.

Cómo funciona COM_MIN: pasos básicos

Recolectar la información de acceso:

- Se obtiene un conjunto de consultas representativas que describen el patrón de acceso a la base de datos distribuida. Cada consulta puede tener condiciones (predicados) sobre la tabla.

Identificar predicados simples:

- Se extraen los predicados de selección simples (por ejemplo, $A < 10$, $B = 'X'$) usados en las consultas.

Generar predicados completos y mutuamente excluyentes:

- Se combinan y refinan los predicados para que sean conjuntos disjuntos (mutuamente excluyentes) que cubran toda la tabla (completitud). Estos definirán los fragmentos.

Asignar fragmentos a nodos:

- Según el análisis del patrón de acceso, se asignan fragmentos a nodos donde las consultas que usan esos fragmentos se ejecutan con mayor frecuencia.

Minimizar comunicación:

- Se busca que la suma total de comunicaciones entre nodos para las consultas sea mínima. Esto se logra creando fragmentos que correspondan a grupos de filas accedidas frecuentemente juntas.

Ejemplo sencillo:

Si tenemos una tabla Empleados, podríamos fragmentarla en:

- Empleados_México: todos los empleados cuya ubicación sea México.
- Empleados_USA: todos los empleados cuya ubicación sea EE.UU.

Características clave:

- Basada en condiciones de selección: cada fragmento se obtiene aplicando un predicado de selección sobre la relación original.
- Propiedades para que sea correcta:
 1. Completitud: la unión de todos los fragmentos debe reconstruir la relación original.
 2. Reconstructibilidad: debe poder reconstruirse la relación original a partir de los fragmentos.
 3. No solapamiento: cada tupla debe aparecer en un solo fragmento (a menos que sea una fragmentación redundante).

Ejemplo en términos de predicados:

La fragmentación horizontal primaria se basa en predicados de selección que determinan qué tuplas van en cada fragmento. Cada fragmento se obtiene aplicando un predicado específico a la relación original.

Por ejemplo, supongamos que tenemos la tabla Empleados con los siguientes atributos:

Empleados(ID, Nombre, Departamento, País)

Queremos fragmentar esta tabla en función del país donde trabaja cada empleado. Podemos definir dos predicados de selección:

p1: País = 'México'

p2: País = 'EE.UU.'

Entonces, la fragmentación horizontal primaria se expresa así:

Empleados_México = $\sigma_{\{\text{País}=\text{'México'}\}}(\text{Empleados})$

Empleados_USA = $\sigma_{\{\text{País}=\text{'EE.UU.'}\}}(\text{Empleados})$

Esto significa:

- Empleados_México contendrá todas las filas (tuplas) donde el país sea 'México'.
- Empleados_USA contendrá todas las filas donde el país sea 'EE.UU.'.

Ventajas:

- Mejora la eficiencia: las consultas pueden acceder solo a los fragmentos relevantes.
- Facilita la distribución de la base de datos: cada fragmento puede almacenarse cerca de donde más se usa.

Bibliografías

Özsu, M. T., & Valduriez, P. (2011). Principles of distributed database systems (3rd ed.). Springer.