

Según ChatGPT:

### ¿Qué es la Fragmentación Horizontal Primaria?

La **fragmentación horizontal** consiste en **dividir una relación (tabla)** en subconjuntos de **tuplas (filas)**, según un criterio de selección. Cada fragmento contiene **un subconjunto de filas** que **cumplen una condición específica**.

Cuando hablamos de **fragmentación horizontal primaria**, nos referimos a que:

- La fragmentación **ocurre sobre una relación base** (tabla original), **sin haber sido previamente fragmentada**.
- Es la primera división que se hace de los datos.
- Se basa en **predicados de selección** (por ejemplo, Región = 'Norte', Edad > 30, etc.).

---

### Enfoque de Tamer Özsu (según *Principles of Distributed Database Systems*)

Según Özsu y Valduriez, la **fragmentación horizontal primaria** debe cumplir con **ciertas propiedades clave** para ser válida:

#### ✅ 1. Completitud (Completeness)

Cada tupla de la relación original **debe aparecer en al menos uno de los fragmentos**.

→ No se puede perder información.

#### ✅ 2. Reconstructibilidad (Reconstructability)

Debe ser posible **reconstruir la relación original** mediante **la unión de los fragmentos**.

→ Normalmente esto se hace con la **operación UNION**:

```
sql                                                                    Copiar  Editar
R = F1 U F2 U ... U Fn
```

#### ✅ 3. Disyunción (Disjointness) (*opcional pero recomendable*)

Cada tupla debe pertenecer a **exactamente un fragmento**.

→ Esto garantiza que no haya **duplicación de datos** entre fragmentos.

→ Se logra cuando los **predicados son mutuamente excluyentes**.

---

### ¿Para qué sirve la fragmentación horizontal primaria?

- **Distribuir datos según la ubicación geográfica** o lógica de uso.
- **Optimizar el rendimiento** al acceder solo a los datos relevantes.
- **Reducir el tráfico de red** en sistemas distribuidos.

- Facilitar la **paralelización de consultas**.

---

### Ejemplo simple

Supón una tabla Clientes con clientes de distintos países:

```
sql                                                                    Copiar  Editar





CREATE TABLE Clientes (
  ID INT,
  Nombre VARCHAR(50),
  País VARCHAR(50)
);
```

Una **fragmentación horizontal primaria** según el país podría ser:

- Fragmento F1: SELECT \* FROM Clientes WHERE País = 'México'
- Fragmento F2: SELECT \* FROM Clientes WHERE País = 'Argentina'
- Fragmento F3: SELECT \* FROM Clientes WHERE País = 'Colombia'

---

### Pasos para diseñar una buena Fragmentación Horizontal Primaria (según Özsu y prácticas recomendadas)

1.  **Análisis del uso de la base de datos**
  - Examina las **consultas más frecuentes** y patrones de acceso.
  - Determina **cómo y desde dónde se accede** a los datos.
2.  **Definición de predicados de fragmentación**
  - Utiliza **predicados de selección simples o compuestos**.
  - Asegúrate de que sean **mutuamente excluyentes**, si quieres disyunción.
3.  **Validación de propiedades**
  - Verifica que los fragmentos cumplen:
    - **Compleitud**
    - **Reconstrucción**
    - (Opcional) **Disyunción**
4.  **Implementación de los fragmentos**

- Crea **sub-tablas físicas** o vistas en distintas ubicaciones si la base está distribuida.
- Asegúrate de manejar correctamente los **índices y claves primarias/foráneas**.

## 5. Revisión y ajuste

- Evalúa el rendimiento.
- Si cambian los patrones de uso, **reformula la fragmentación**.

---

## Errores comunes al fragmentar

- Fragmentar por una columna irrelevante (por ejemplo, que no se usa en consultas).
- No asegurar la **reconstrucción** de la tabla original.
- Tener **fragmentos con datos solapados** sin control.
- Crear fragmentos con **predicados muy amplios o mal balanceados**, generando fragmentos desiguales.

---

## Conclusión

La **fragmentación horizontal primaria** es una técnica **fundamental en bases de datos distribuidas**, y Özsu la considera uno de los **pilares del diseño** de estos sistemas. Una buena fragmentación mejora el rendimiento y la escalabilidad, pero debe ser cuidadosamente diseñada con base en el **análisis del sistema y los patrones de acceso**.