

# PROYECTO SEMESTRAL

## Bases de Datos Distribuidas Gestión de Terminales Terrestres



**ISWD - 553**

**GR1CC**

**GRUPO 8**

**Integrantes:** Pasquel Johann, Torres Jorge

## 1. Escenario

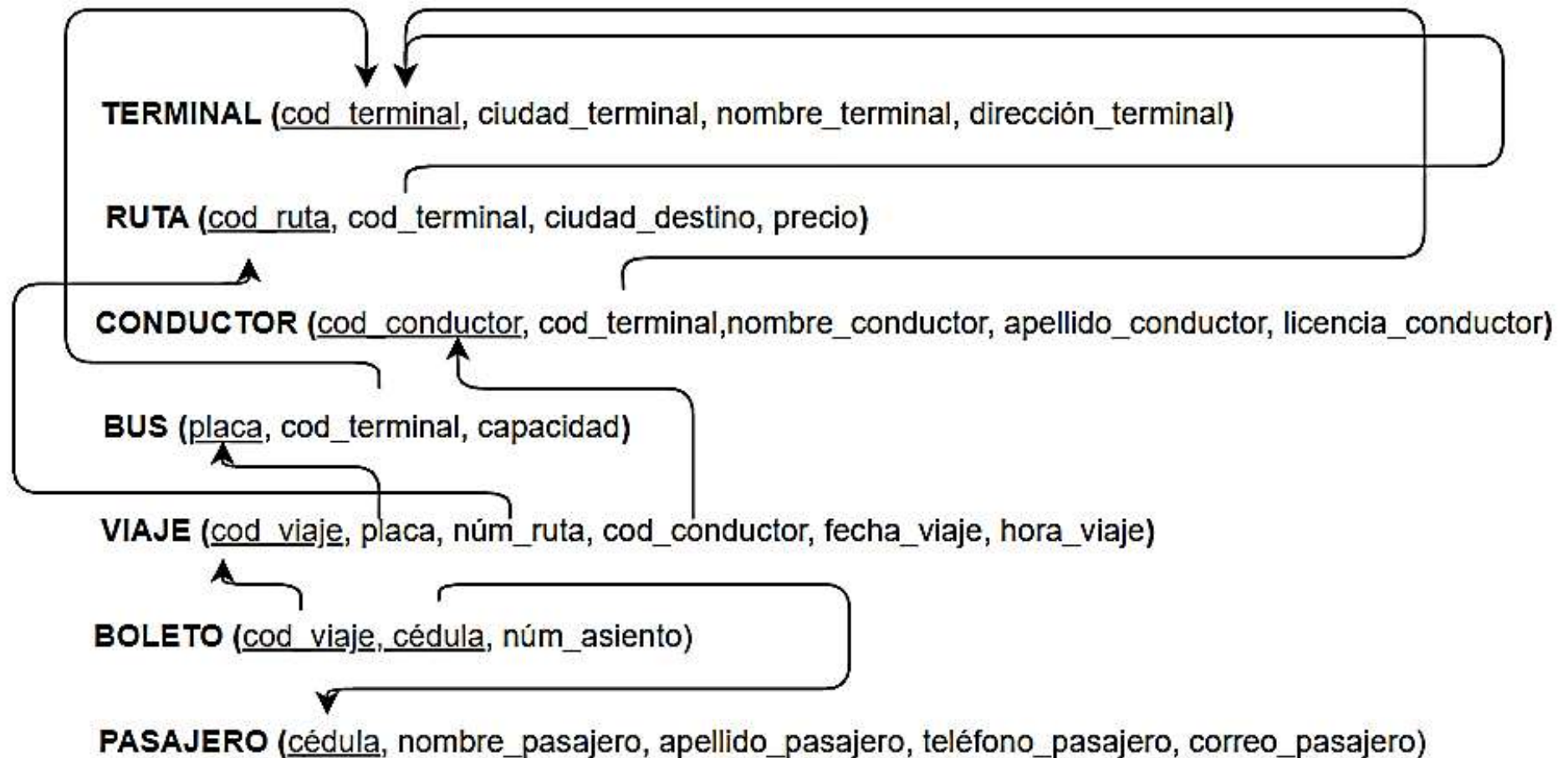
El sistema gestiona la operación de terminales de buses en dos sedes: **Quito e Ibarra**. La sede Quito ejecuta la base de datos **Terminal\_Quito** en el servidor **VLADIMIRJON** y la sede Ibarra ejecuta **Terminal\_Ibarra** en el servidor **ASUS-8KR2UI2**. El objetivo operativo es permitir que cada sede trabaje con autonomía sobre su información local (buses, rutas, conductores, viajes y pasajeros), manteniendo a la vez una visión integrada para consultas y reportes a nivel institucional.

En cuanto a la infraestructura lógica, ambas instancias de SQL Server están interconectadas mediante servidores vinculados (**Linked Servers**) y habilitan transacciones distribuidas (MSDTC) cuando una operación involucra datos de más de una sede. El firewall fue ajustado para permitir la comunicación requerida por estas transacciones.

La estrategia de distribución de datos combina **fragmentación y replicación**. Sobre estas particiones y réplicas se exponen **vistas particionadas** que permiten a las aplicaciones consultar de forma uniforme, y en los casos permitidos, ejecutar operaciones de actualización enrutadas hacia el fragmento correspondiente.

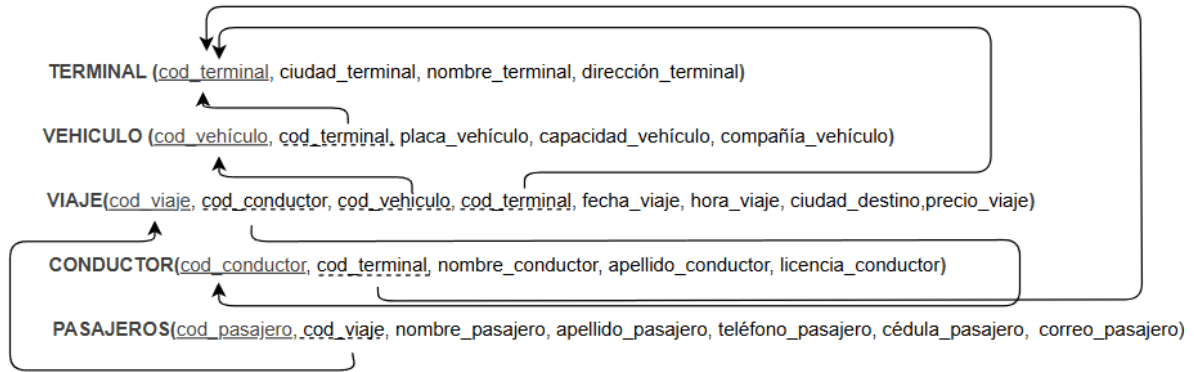
## 2. Grafo relacional

# Sistema de Gestión de Terminales Terrestres



Cambios realizados:

## Sistema de Gestión de Terminales Terrestres



Se incorporó BOLETO como entidad puente entre PASAJERO y VIAJE, resolviendo explícitamente la relación muchos-a-muchos: ahora un pasajero puede comprar varios boletos para la misma ruta en fechas distintas (cada boleto apunta a un viaje específico). A la par, se clarificó el rol de RUTA como catálogo del trayecto y de VIAJE como la instancia programada (fecha/hora, conductor y vehículo), lo que permite que una ruta tenga múltiples viajes a lo largo del tiempo y que cada viaje soporte múltiples boletos.

### 3. Campo y condición de fragmentación

#### 3.1 Campo y condición de fragmentación

3.1.1 Campo de fragmentación: *cod\_terminal*

3.1.2 Condición de fragmentación  $i = \{1, 2\}$

#### Esquema de fragmentación

#### 3.2 Fragmentaciones verticales

##### 3.2.1 Conductor

$$\text{ConductorDatos} = \Pi_{\text{cod\_conductor}, \text{nombre\_conductor}, \text{apellido\_conductor}} (\text{Conductor})$$

$$\text{ConductorTerminal} = \Pi_{\text{cod\_conductor}, \text{cod\_terminal}, \text{cedula\_conductor}} (\text{Conductor})$$

##### 3.2.2 Fragmentación horizontal primaria

$$\text{Ruta}_i = \sigma_{\text{cod\_terminal} = "i"} (\text{Ruta})$$

Donde  $i = \{1, 2\}$

##### 3.3 Fragmentación horizontal primaria

$$\text{ConductorTerminal}_i = \sigma_{\text{cod\_terminal} = "i"} (\text{Conductor})$$

Donde  $i = \{1, 2\}$

### 3.4 Fragmentación horizontal primaria

$$Bus_i = \sigma_{cod\_terminal = "i"}(Bus)$$

Donde  $i = \{1, 2\}$

### 3.5 Fragmentaciones horizontales derivadas

#### 3.5.1 Viaje respecto a Ruta

$$Viaje_i = Viaje \times Ruta_i$$

#### 3.5.2 Boleto respecto a Viaje

$$Boleto_i = Boleto \times Viaje_i$$

#### Cambios realizados:

Campo y condición de fragmentación

Campo de fragmentación

*ciudad\_terminal*

Condición de fragmentación

$i = \{Quito, Ibarra\}$

Esquema de fragmentación

5.1 Fragmentación horizontal primaria

Donde

$$Terminal_i = \sigma_{ciudad\_terminal = "i"}(Terminal)$$

$i = \{Quito, Ibarra\}$

En la versión anterior se usó **ciudad\_terminal** como atributo de fragmentación y se planteó fragmentar horizontalmente **TERMINAL**; en el diseño final esto se corrige: **TERMINAL** se maneja como tabla replicada en ambas sedes y, por tanto, no se fragmenta. La fragmentación horizontal primaria se limita a las tablas operativas, utilizando cod\_terminal como predicado de corte para localizar registros por sede.

## 4. Esquema de replicación

Dentro del diseño del sistema distribuido, la tabla **TERMINAL** y **PASAJERO** han sido replicadas en ambas sedes (**Quito e Ibarra**). **TERMINAL** funciona como una tabla ya determinada y no admite cambios; por otra parte, **PASAJERO**, aunque replicada, sí permite incorporar nuevos registros conforme al escenario operativo de los terminales (nuevos pasajeros, cambios de datos, borrado de datos).

- **Tipo de replicación:** Se ha implementado una replicación unidireccional.
- **Nodo de gestión:** El nodo de gestión es **Quito**, lo que significa que todas las modificaciones o cargas iniciales sobre la tabla **TERMINAL** y **PASAJERO** se realizarán desde esta sede, y posteriormente se distribuyen a Ibarra.

### Cambios realizados:

En el diseño anterior se replicaba solo la tabla TERMINAL en Quito e Ibarra bajo un esquema unidireccional administrado desde Quito, principalmente para consultas; en el diseño actual se replica TERMINAL y PASAJERO en ambas sedes, manteniendo TERMINAL como estructura estable, mientras PASAJERO, aun replicada, admite altas y actualizaciones según la operación.

### 5. Esquema de Ubicación

	QUITO	IBARRA
TERMINAL	TERMINAL	TERMINAL
RUTA	Ruta_1	Ruta_2
CONDUCTOR	ConductorTerminal_1 ConductorDatos	ConductorTerminal_2
BUS	Bus_1	Bus_2
VIAJE	Viaje_1	Viaje_2
BOLETO	Boleto_1	Boleto_2
PASAJERO	PASAJERO	PASAJERO

**Cambios realizados:**

	QUITO	IBARRA
TERMINAL	TerminalQuito	TerminalIbarra
VIAJE	ViajeInformación_Quito ViajeTiempo	ViajeInformación_Ibarra
PASAJERO	PasajerosInformación_Quito PasajerosContato	PasajerosInformación_Ibarra
CONDUCTOR	Conductor_Quito	Conductor_Ibarra
VEHÍCULO	Vehículo_Quito	Vehículo_Ibarra

TERMINAL se mantiene replicada en Quito e Ibarra; RUTA, BUS y VIAJE pasaron de definiciones dispersas (p. ej., Vehículo\_/ViajeInformación\_/ ViajeTiempo) a fragmentos por sede (Ruta\_1/Ruta\_2, Bus\_1/Bus\_2, Viaje\_1/Viaje\_2). PASAJERO dejó de estar fragmentado en tablas de “información” y “contacto” para convertirse en una única tabla replicada en ambas sedes.

CONDUCTOR adoptó un esquema mixto: vertical (ConductorDatos centralizado en Quito) y horizontal por sede (ConductorTerminal\_1 en Quito y ConductorTerminal\_2 en Ibarra).

Además, se incorporó BOLETO como fragmento por sede (Boleto\_1/Boleto\_2) para materializar la relación entre pasajeros y viajes. En conjunto, el modelo final uniforma nombres, elimina particiones innecesarias, y distribuye solo lo operativo por sede, manteniendo replicado lo común.