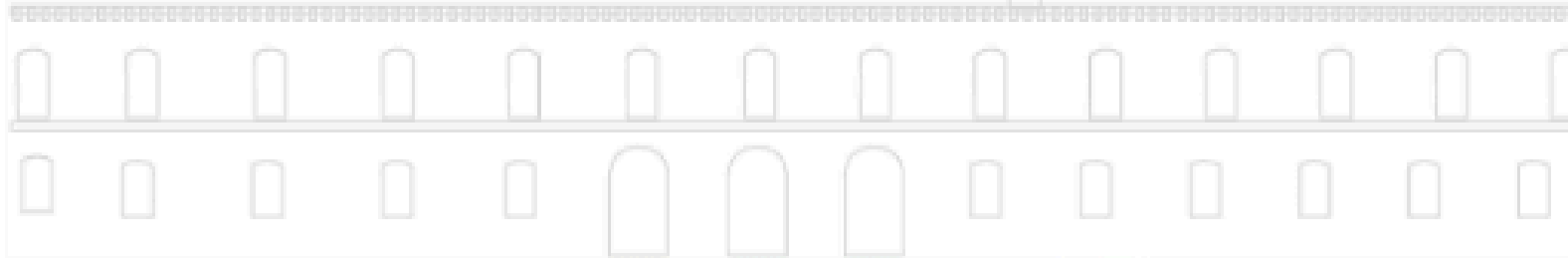
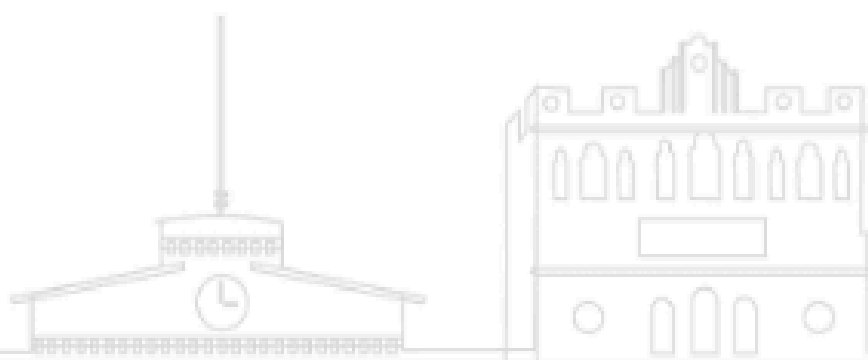


Reporte de Practica No. 3.2

ALUMNO: Cesar Rodriguez Garcia
Dr. Eduardo Cornejo-Velázquez



I. Marco Teórico

Tema	Concepto Clave	Sintaxis o Comando Relevante
Fragmentación Vertical	División de las columnas de una tabla en fragmentos para distribuirlos en diferentes nodos, mejorando el rendimiento en consultas que solo necesitan un subconjunto de atributos.	No hay un comando SQL directo para la fragmentación; es un proceso de diseño (crear tablas con subconjuntos de columnas).
Procesos ETL	Extracción (obtener datos), Transformación (limpiar/adaptar datos), Load (cargar datos). Es el proceso de mover datos entre sistemas.	Involucra SELECT (Extracción), código intermedio (Transformación) y LOAD DATA INFILE (Carga).
SELECT ... INTO OUTFILE	Exporta el resultado de una consulta directamente a un archivo en el sistema de archivos del servidor. Útil para la fase de Extracción del ETL.	SELECT col1, col2 FROM tabla INTO OUTFILE '/ruta/archivo.csv' FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n';
LOAD DATA INFILE	Importa datos desde un archivo del sistema de archivos a una tabla en la base de datos. Útil para la fase de Carga del ETL.	LOAD DATA INFILE '/ruta/archivo.csv' INTO TABLE tabla FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n';
SELECT con tablas de dos bases de datos	Permite unir o consultar datos de tablas que residen en diferentes esquemas (bases de datos) en el mismo servidor MySQL.	SELECT t1.col FROM db1.tabla1 t1 JOIN db2.tabla2 t2 ON t1.id = t2.fk_id;

II. Esquema Conceptual Local de Cada Nodo

Basado en la fragmentación vertical (misma entidad, diferentes atributos/columnas en cada nodo) y la replicación de ciertas entidades (como **vehículo**).

Nodo	Esquemas (Bases de Datos)	Tablas (y Atributos Implícitos)
LCS1-Principal	flotilla_db	flotilla (Datos generales de la flotilla)
		vehiculo_principal (Atributos de identificación del vehículo: ID, placa, modelo, etc.)
		documento (Documentos legales/administrativos del vehículo: póliza, verificación, etc.)
LCS2-Mantenimiento	mantenimiento_db	vehiculo_mtto (Atributos relacionados con estado/uso : ID, placa, kilometraje actual, estado_motor, etc.)
		mantenimiento (Registros de servicios: ID_vehiculo, fecha, tipo_servicio, costo, etc.)
LCS3-Rutas	rutas_db	vehiculo_ruta (Atributos relacionados con operación/conductor : ID, placa, capacidad_carga, etc.)
		conductor (Datos del conductor: ID, nombre, licencia, etc.)
		ruta (Detalle de rutas: ID, origen, destino, distancia, etc.)

		transaccionCombustible (Cargas de combustible: ID_vehiculo, fecha, litros, costo, etc.)
--	--	------------------------------------------------------------------------------------------------

III. Script de Creación de Nodos (Base de Datos y Tablas)

Nodo LCS1-Principal

```
CREATE DATABASE IF NOT EXISTS flotilla_db;
```

```
USE flotilla_db;
```

```
CREATE TABLE vehiculo_principal (
  id_vehiculo INT PRIMARY KEY,
  placa VARCHAR(10) UNIQUE NOT NULL,
  modelo VARCHAR(50),
  anio INT
);
```

```
CREATE TABLE documento (
  id_documento INT PRIMARY KEY,
  id_vehiculo INT,
  tipo VARCHAR(50),
  fecha_vencimiento DATE,
  FOREIGN KEY (id_vehiculo) REFERENCES vehiculo_principal(id_vehiculo)
);
```

-- Nodo LCS2-Mantenimiento

```
CREATE DATABASE IF NOT EXISTS mantenimiento_db;
```

```
USE mantenimiento_db;
```

```
CREATE TABLE vehiculo_mtto (
  id_vehiculo INT PRIMARY KEY,
  placa VARCHAR(10) UNIQUE NOT NULL,
  kilometraje_actual INT,
  estado_motor VARCHAR(20)
);
```

IV. Scripts de Extracción y Carga de Datos (Proceso ETL)

1. Extracción (de LCS1-Principal)

```
USE flotilla_db;

SELECT id_vehiculo, placa, '0' AS kilometraje_actual, 'OK' AS estado_motor

INTO OUTFILE '/tmp/datos_mtto.csv'

FIELDS TERMINATED BY ',' ENCLOSED BY '"'

LINES TERMINATED BY '\n'

FROM vehiculo_principal;
```

```
SELECT id_vehiculo, placa, '1000' AS capacidad_carga

INTO OUTFILE '/tmp/datos_ruta.csv'

FIELDS TERMINATED BY ',' ENCLOSED BY '"'

LINES TERMINATED BY '\n'

FROM vehiculo_principal;
```

2. Carga (en LCS2-Mantenimiento y LCS3-Rutas)

-- LCS2-Mantenimiento

```
USE mantenimiento_db;

LOAD DATA INFILE '/tmp/datos_mtto.csv'

INTO TABLE vehiculo_mtto

FIELDS TERMINATED BY ',' ENCLOSED BY '"'

LINES TERMINATED BY '\n'

(id_vehiculo, placa, kilometraje_actual, estado_motor);
```

-- LCS3-Rutas

```
USE rutas_db;
```

```
LOAD DATA INFILE '/tmp/datos_ruta.csv'
```

```
INTO TABLE vehiculo_ruta
```

```
FIELDS TERMINATED BY ',' ENCLOSED BY ''
```

```
LINES TERMINATED BY '\n'
```

```
(id_vehiculo, placa, capacidad_carga);
```

V. Script de Consulta de Datos a Dos Tablas

1. Consulta en LCS2-Mantenimiento

```
USE mantenimiento_db;
```

```
SELECT
```

```
    v.placa,
```

```
    v.kilometraje_actual,
```

```
    m.fecha_servicio,
```

```
    m.tipo_servicio
```

```
FROM
```

```
    vehiculo_mtto v
```

```
INNER JOIN
```

```
    mantenimiento m ON v.id_vehiculo = m.id_vehiculo
```

```
WHERE
```

```
    m.fecha_servicio > DATE_SUB(CURDATE(), INTERVAL 6 MONTH);
```

2. Consulta a Dos Bases de Datos (LCS1 y LCS2)

```
USE flotilla_db;
```

```
SELECT
```

```
    vp.placa,
```

```
    d.tipo,
```

```
    d.fecha_vencimiento AS Doc_Vencimiento_LCS1,
```

```
    vm.kilometraje_actual AS Kilometraje_LCS2
```

```
FROM
```

```
    vehiculo_principal vp
```

```
JOIN
```

```
    documento d ON vp.id_vehiculo = d.id_vehiculo
```

```
JOIN
```

```
    mantenimiento_db.vehiculo_mtto vm ON vp.placa = vm.placa
```

```
WHERE
```

```
    vp.placa = 'ABC-123';
```