PR1 – Análisis y Diseño de un Almacén de Datos

Vinicio Naranjo Mosquera

16/11/2024

Contents

1 Análisis de los Requisitos

El análisis de los requisitos se basa en identificar las necesidades específicas que tiene la organización respecto al análisis de la información de viajes de vehículos de alquiler en Nueva York. En esta fase, se debe pensar tanto en las necesidades actuales como en posibles necesidades futuras, para así cubrir de manera integral el análisis de la información. La necesidad principal de la organización es disponer de información integrada para su análisis y difusión mediante herramientas de inteligencia de negocio. Esto facilitará la toma de decisiones para los usuarios potenciales y permitirá cumplir el objetivo de analizar los viajes de vehículos de alquiler en Nueva York. A continuación, se especifican las preguntas que el sistema debe ser capaz de responder para cubrir las necesidades de los usuarios potenciales.

1.1 Preguntas Clave para el Análisis de Viajes de Vehículos de Alquiler

1. **¿Cuántos viajes de vehículos de alquiler (tanto FHV como taxis amarillos) se inician en cada distrito y zona de Nueva York en un periodo específico?**
   - Permite analizar la demanda en distintas zonas de la ciudad, útil para planificación de servicios y asignación de recursos.
2. **¿Cuál es el tiempo promedio de viaje y la distancia recorrida para los servicios de FHV y taxis amarillos por cada zona de recogida y destino?**
   - Proporciona información sobre la duración y longitud de los viajes, importante para el análisis de eficiencia y costos de operación.
3. **¿Cuáles son los patrones de uso de los diferentes tipos de pago (tarjeta, efectivo, etc.) en las distintas zonas de la ciudad?**
   - Ayuda a entender las preferencias de pago por zona y permite adaptar estrategias de servicio según el comportamiento del cliente.
4. **¿Qué zonas experimentan los tiempos de espera más largos antes de recoger a un pasajero?**
   - Identificar áreas con posible escasez de vehículos, lo que facilita ajustar la distribución de estos para mejorar el servicio.
5. **¿Cómo varía la cantidad de viajes según las franjas horarias (mañana, tarde, noche) en diferentes distritos?**
   - Permite observar patrones de demanda por horarios, esencial para optimizar la disponibilidad de vehículos en distintos momentos del día.
6. **¿Cuál es el impacto de la congestión (congestion surcharge) y otras tarifas en el precio total de los viajes?**
   - Permite analizar cómo las tarifas adicionales afectan el costo para los usuarios y evaluar el impacto económico de la congestión.
7. **¿Cuántos viajes de vehículos de alquiler terminan en áreas específicas como aeropuertos (LaGuardia, JFK) y centros turísticos (Times Square)?**
   - Facilita el monitoreo de la actividad en puntos de interés clave, útil para planificar servicios especiales o campañas en estos lugares.
8. **¿Cuál es la evolución mensual de los viajes en áreas con alta demanda, como el centro de Manhattan o Brooklyn?**
   - Conocer esta tendencia mensual ayuda a realizar previsiones de demanda a largo plazo y definir estrategias de crecimiento o ajustes en la flota.
9. **¿Qué patrones de viaje existen para los servicios de FHV y taxis amarillos durante eventos específicos (como ferias, festivales)?**
   - Permite evaluar el impacto de eventos locales en la demanda de transporte y coordinar recursos adicionales durante estos eventos.

10. **¿Cuántos viajes resultan en un tiempo de espera corto (por debajo de 5 minutos) y en qué zonas se encuentran?**

    - Identificar zonas con menor tiempo de espera permite analizar áreas con alta disponibilidad de vehículos, optimizando la asignación para evitar saturación en esas zonas.

1.2 Conclusión

Estas preguntas están diseñadas para proporcionar una visión integral de la operación de los vehículos de alquiler en Nueva York, incluyendo aspectos de tiempo, distancia, zona, preferencia de pago y costos asociados. Este enfoque permite analizar patrones de demanda y comportamiento de los usuarios en diferentes momentos y ubicaciones específicas, cumpliendo con el objetivo de proporcionar información integrada para una toma de decisiones informada sobre el servicio de vehículos de alquiler en Nueva York.

2 Análisis de la Fuente de Datos

2.1 Análisis `yellow_tripdata`

Descripción del Archivo

- **Nombre de archivo**: `yellow_tripdata-001.zip` (fichero comprimido, contiene varios CSV).
- **Tipo**: Fichero de texto plano (CSV) con delimitador de coma (‘,‘).
- **Formato**: Incluye cabecera en la primera fila con los nombres de los campos.
- **Período**: Contiene datos de múltiples meses de viajes de taxis amarillos, ejemplo mostrado desde octubre 2023 a enero 2024.

Estructura de Campos

Nombre de campo	Descripción	Tipo	Ejemplo
VendorID	ID del proveedor del taxi	Numérico	2
tpep_pickup_datetime	Fecha y hora de recogida	Fecha/hora	2024-01-01 00:57:55
tpep_dropoff_datetime	Fecha y hora de llegada	Fecha/hora	2024-01-01 01:17:43
passenger_count	Número de pasajeros	Numérico	1.0
trip_distance	Distancia recorrida	Numérico	1.72
RatecodeID	Código de tarifa	Numérico	1.0
store_and_fwd_flag	Indicador de almacenamiento y reenvío	Texto	N
PULocationID	ID de la ubicación de recogida	Numérico	186
DOLocationID	ID de la ubicación de destino	Numérico	79
payment_type	Tipo de pago	Numérico	2
fare_amount	Importe de la tarifa	Numérico	17.7
extra	Cargo extra	Numérico	1.0
mta_tax	Impuesto MTA	Numérico	0.5
tip_amount	Propina	Numérico	0.0
tolls_amount	Importe de peajes	Numérico	0.0
improvement_surcharge	Recargo por mejoras	Numérico	1.0
total_amount	Importe total	Numérico	22.7
congestion_surcharge	Recargo por congestión	Numérico	2.5
Airport_fee	Tarifa de aeropuerto	Numérico	0.0

Table 1: Estructura de campos de `yellow_tripdata`

#	Nombre del fichero	Total registros
1	yellow_tripdata_2023-10.csv	2,964,624
2	yellow_tripdata_2023-11.csv	3,522,285
3	yellow_tripdata_2023-12.csv	3,339,715
4	yellow_tripdata_2024-01.csv	3,376,567
Total		**13,203,191**

Table 2: Volumetría de los archivos yellow_tripdata

Fichero	Registros	Columnas	Datos Estimados (MB)
yellow_tripdata-001.zip	13,203,191	19	1,526.2

Table 3: Estimación de volumen para la carga inicial

Volumetría

Estimación de Volumen para Carga Inicial

Observaciones

- **Transformación de Fechas**: Es importante normalizar el formato de fechas para que sea coherente con otros archivos.
- **Tratamiento de Nulos**: Asegurar que los valores nulos o vacíos se traten de manera uniforme, especialmente en el campo store_and_fwd_flag.
- **Carga Inicial**: Dado el volumen significativo de datos, se sugiere cargar los datos en el sistema en particiones mensuales para optimizar el rendimiento.

2.2 Análisis de fhv_tripdata

Descripción del Archivo

- **Nombre de archivo**: fhv_tripdata-001.zip (fichero comprimido, contiene varios CSV).
- **Tipo**: Fichero de texto plano (CSV) con delimitador de coma (‘,‘).
- **Formato**: Incluye cabecera en la primera fila con los nombres de los campos.
- **Período**: Contiene datos de múltiples meses de viajes de vehículos de alquiler (FHV) desde octubre 2023 a enero 2024.

Estructura de Campos

Nombre de campo	Descripción	Tipo	Ejemplo
dispatching_base_num	Número de licencia TLC.	Numérico	B00254
pickup_datetime	Fecha y hora de inicio del viaje (activación del taxímetro).	Fecha/hora	2023-10-01 00:24:00
dropoff_datetime	Fecha y hora de final del viaje (desactivación del taxímetro).	Fecha/hora	2023-10-01 00:38:39
PUlocationID	ID de la ubicación de recogida	Numérico	48.0
DOlocationID	ID de la ubicación de entrega	Numérico	107.0
SR_flag	Indicador (s/n) de si el viaje es compartido	Texto	
Affiliated_base_number	Número de base a la que está afiliado el vehículo	Numérico	B00254

Table 4: Estructura de campos de fhv_tripdata

Volumetría

#	Nombre del fichero	Total registros
1	`fhv_tripdata_2023-10.csv`	1,628,438
2	`fhv_tripdata_2023-11.csv`	1,343,846
3	`fhv_tripdata_2023-12.csv`	1,376,748
4	`fhv_tripdata_2024-01.csv`	1,290,116
Total		**5,639,148**

Table 5: Volumetría de los archivos `fhv_tripdata`

Estimación de Volumen para Carga Inicial

Fichero	Registros	Columnas	Datos Estimados (MB)
fhv_tripdata-001.zip	5,639,148	7	199

Table 6: Estimación de volumen para la carga inicial de `fhv_tripdata`

2.3 Análisis de `taxi_zone_lookup`

Descripción del Archivo

- **Nombre de archivo**: `taxi_zone_lookup.csv`
- **Tipo**: Fichero de texto plano (CSV) con delimitador de coma (‘,‘).
- **Formato**: Incluye cabecera en la primera fila con los nombres de los campos.
- **Registros Totales**: 265.

Estructura de Campos

Nombre de campo	Descripción	Tipo	Ejemplo
LocationID	ID de la ubicación	Numérico	1
Borough	Distrito o condado de la ubicación	Texto	Queens
Zone	Nombre de la zona	Texto	Jamaica Bay
service_zone	Zona de servicio	Texto	Boro Zone

Table 7: Estructura de campos de `taxi_zone_lookup`

Volumetría

Fichero	Registros	Columnas	Datos Estimados (KB)
taxi_zone_lookup.csv	265	4	13.97

Table 8: Estimación de volumen para la carga inicial de `taxi_zone_lookup.csv`

2.4 Análisis de `current_base.tsv`

Descripción del Archivo

- **Nombre del archivo**: `current_base.tsv`
- **Formato**: Texto plano con separador de tabulación (`\t`).
- **Registros totales**: 100 (ejemplo basado en análisis previo).
- **Objetivo**: Proveer información sobre las bases de vehículos de alquiler.

Estructura de Campos

Nombre del campo	Descripción	Tipo de Dato	Ejemplo
License Number	Número de licencia del vehículo	Alfanumérico	B00254
Entity Name	Nombre de la entidad	Texto	Base Transportation
Type of Base	Tipo de base	Texto	FHV
Address	Dirección de la base	Texto	123 Main St

Table 9: Estructura de campos de `current_base.tsv`

Observaciones

- Se requiere limpieza de datos para estandarizar los valores del campo `Type of Base`.
- Los valores nulos en `Entity Name` deben ser tratados mediante imputación o eliminación.

Volumetría

Archivo	Registros	Columnas	Tamaño Estimado (KB)
`current_base.tsv`	100	4	5.3

Table 10: Volumetría del archivo `current_base.tsv`

2.5 Análisis de `Rate_code.tab`

Descripción del Archivo

- **Nombre del archivo**: `Rate_code.tab`
- **Formato**: Texto plano con separador de tabulación (`\t`).
- **Registros totales**: 6.
- **Objetivo**: Proveer códigos de tarifa y sus descripciones.

Estructura de Campos

Nombre del campo	Descripción	Tipo de Dato	Ejemplo
RatecodeID	Identificador del código de tarifa	Numérico	1
Description	Descripción del código de tarifa	Texto	Standard Rate

Table 11: Estructura de campos de `Rate_code.tab`

Observaciones

- No se requieren transformaciones significativas; los datos están limpios y bien estructurados.

Volumetría

Archivo	Registros	Columnas	Tamaño Estimado (KB)
`Rate_code.tab`	6	2	0.8

Table 12: Volumetría del archivo `Rate_code.tab`

2.6 Análisis de `Payment_type.xls`

Descripción del Archivo

- **Nombre del archivo**: `Payment_type.xls`
- **Formato**: Hoja de cálculo Excel.
- **Registros totales**: 6.
- **Objetivo**: Proveer información sobre métodos de pago.

Estructura de Campos

Nombre del campo	Descripción	Tipo de Dato	Ejemplo
PaymentID	Identificador del método de pago	Numérico	1
Payment_type	Descripción del método de pago	Texto	Credit card

Table 13: Estructura de campos de `Payment_type.xls`

Observaciones

- Se deben estandarizar las descripciones en `Payment_type` para evitar duplicados.
- Los valores faltantes deben ser manejados antes de la carga al sistema.

Volumetría

Archivo	Registros	Columnas	Tamaño Estimado (KB)
`Payment_type.xls`	6	2	0.3

Table 14: Volumetría del archivo `Payment_type.xls`

3 Análisis Funcional y Selección de Arquitectura

Elección de la Arquitectura

Considerando los requisitos y la naturaleza del proyecto, he seleccionado la **Arquitectura de Data Mart Bus con Dimensiones Conformadas** (enfoque de Kimball) como la más adecuada. Esta arquitectura es especialmente útil para proyectos que requieren múltiples data marts y análisis multidimensional. A continuación, se describen las características principales de esta arquitectura:

- **Área de Staging**: Zona de preparación donde se realizan las tareas de Extracción, Transformación y Carga (ETL) para asegurar la calidad y consistencia de los datos antes de que ingresen al almacén de datos central.
- **Almacén de Datos Central**: Un repositorio centralizado donde se integran y normalizan todos los datos relevantes de viajes, eliminando redundancias y facilitando la generación de métricas precisas.
- **Data Marts Específicos**: Se crean data marts especializados para áreas de análisis clave, como zonas geográficas, tipos de vehículos y métodos de pago. Esto permite que cada área acceda a los datos específicos que necesita sin duplicación.
- **Dimensiones Conformadas**: Dimensiones comunes, como `fecha`, `ubicación` y `tipo de vehículo`, que se comparten entre data marts, garantizando consistencia y permitiendo análisis comparativos.
- **Herramientas OLAP y BI**: La arquitectura está diseñada para soportar herramientas OLAP, permitiendo consultas multidimensionales y análisis intuitivo para los usuarios.

Justificación de la Elección

- **Flexibilidad y Escalabilidad**: Esta arquitectura permite la incorporación de nuevos data marts en respuesta a las necesidades de análisis cambiantes, manteniendo la integridad de las dimensiones compartidas.
- **Eficiencia en el Desarrollo Incremental**: Los data marts pueden desarrollarse y desplegarse en etapas, proporcionando valor al negocio de forma temprana.
- **Soporte para Análisis Multidimensional**: La estructura del Data Mart Bus y el uso de herramientas OLAP son ideales para responder a preguntas clave sobre el comportamiento de los viajes y otros aspectos operacionales.

3.1 Tabla de Requisitos Funcionales

#	Requisito	Prioridad	Exigible/Deseable
1	Se extraerá de forma adecuada la información de las fuentes de datos.	1	E
2	Se creará un almacén de datos centralizado para integrar todos los datos de viajes.	1	E
3	Los datos se transformarán y limpiarán para asegurar su calidad y consistencia en el almacén.	1	E
4	Se implementará una arquitectura de Data Mart Bus con dimensiones conformadas para mantener consistencia.	1	E
5	El sistema debe permitir la consulta multidimensional a través de herramientas OLAP.	2	D
6	Se deben generar informes detallados sobre patrones de demanda por zona y franja horaria.	2	D
7	Se proporcionarán visualizaciones de datos para representar tendencias y patrones clave.	2	D
8	El sistema debe permitir consultas ad hoc para análisis personalizados por parte de los analistas.	1	E
9	Se establecerán medidas de seguridad y control de acceso a los datos sensibles.	2	D
10	El sistema debe soportar cargas incrementales de datos para optimizar la actualización.	1	E
11	El sistema debe ser escalable para manejar incrementos en el volumen de datos.	2	D
12	Se deben implementar alertas y notificaciones para eventos críticos en los datos.	3	D

Table 15: Requisitos funcionales para la factoría de información

3.1.1 Justificación de las Prioridades

- **Prioridad 1 (Críticos para la Operación)**: Requisitos esenciales que garantizan la disponibilidad, consistencia y accesibilidad de los datos, permitiendo que el sistema funcione y responda a las preguntas clave del análisis de viajes.
- **Prioridad 2 (Importantes para el Análisis)**: Requisitos que agregan funcionalidad analítica y seguridad adicional, aumentando la utilidad del sistema para los analistas sin ser críticos para la operación inicial.
- **Prioridad 3 (Deseables pero No Urgentes)**: Características avanzadas, como alertas y notificaciones, que no son imprescindibles para el funcionamiento básico del sistema, pero contribuyen al análisis proactivo.

3.2 Otros Requisitos Funcionales

Para asegurar que el sistema cumpla con las expectativas de escalabilidad, rendimiento y cumplimiento legal, se consideran los siguientes requisitos adicionales:

- **Escalabilidad**: El sistema debe estar diseñado para manejar incrementos en el volumen de datos conforme crece el servicio de transporte o aumenta el número de registros por viaje.

- **Disponibilidad y Rendimiento**: El almacén de datos debe garantizar una disponibilidad del 99.9% y responder a consultas en un tiempo razonable, no superior a cinco segundos en promedio.
- **Cumplimiento de Normativas de Protección de Datos**: El sistema debe adherirse a las regulaciones locales e internacionales de protección de datos, incluyendo el tratamiento de datos personales de los pasajeros.
- **Control de Calidad de los Datos**: Implementar un mecanismo de control de calidad que verifique la precisión y consistencia de los datos en cada actualización, asegurando que los análisis sean fiables.

3.3 Conclusión

La selección de la **Arquitectura de Data Mart Bus con Dimensiones Conformadas** es ideal para este proyecto debido a su flexibilidad, escalabilidad y compatibilidad con herramientas OLAP, lo cual es esencial para satisfacer las necesidades de análisis en un sistema de transporte urbano como el de Nueva York. Los requisitos funcionales detallados y priorizados garantizan que el sistema pueda responder de forma eficiente a las preguntas planteadas, y las recomendaciones adicionales en seguridad, escalabilidad y calidad de datos proporcionan una estructura robusta y confiable para el análisis de información.
Este enfoque asegura que la factoría de información no solo cumpla con los objetivos inmediatos, sino que también esté preparada para adaptarse a las necesidades cambiantes y de crecimiento de la organización.

4 Diseño Conceptual

Dado que los datos de `yellow_tripdata` y `fhv_tripdata` tienen atributos distintos, se han creado dos tablas de hechos separadas:

4.1 Tabla de Hechos `FACT_YELLOW_TRIP`

Esta tabla se centra exclusivamente en los datos de taxis amarillos.

Atributo	Descripción	Fuente
`trip_id`	Identificador único del viaje	Generado
`pickup_datetime`	Fecha y hora de inicio del viaje	`tpep_pickup_datetime`
`dropoff_datetime`	Fecha y hora de fin del viaje	`tpep_dropoff_datetime`
`trip_duration`	Duración del viaje	Calculado (diferencia entre `pickup` y `dropoff`)
`pickup_location_id`	ID de la ubicación de recogida	`PULocationID`
`dropoff_location_id`	ID de la ubicación de destino	`DOLocationID`
`vendor_id`	Identificador del proveedor del taxi	`VendorID`
`rate_code_id`	Código de tarifa	`RatecodeID`
`payment_type_id`	Tipo de pago	`payment_type`
`distance`	Distancia recorrida durante el viaje	`trip_distance`
`fare_amount`	Importe de la tarifa del viaje	`fare_amount`
`total_amount`	Importe total del viaje (incluye propinas)	`total_amount`
`congestion_surcharge`	Recargo por congestión	`congestion_surcharge`

4.2 Tabla de Hechos FACT_FHV_TRIP

Esta tabla se centra exclusivamente en los datos de vehículos de alquiler (FHV).

Atributo	Descripción	Fuente
trip_id	Identificador único del viaje	Generado
pickup_datetime	Fecha y hora de inicio del viaje	pickup_datetime
dropoff_datetime	Fecha y hora de fin del viaje	dropoff_datetime
trip_duration	Duración del viaje	Calculado (diferencia entre pickup y dropoff)
pickup_location_id	ID de la ubicación de recogida	PUlocationID
dropoff_location_id	ID de la ubicación de destino	DOlocationID
dispatching_base_id	ID de la base que despachó el viaje	dispatching_base_num
affiliated_base_id	ID de la base afiliada del vehículo	affiliated_base_number
sr_flag	Indicador de viaje compartido	SR_flag

4.3 Dimensiones Comunes

4.3.1 Dimensión DIM_TIME

Esta dimensión permite analizar los datos por tiempo y se comparte entre ambas tablas de hechos.

Atributo	Descripción	Fuente
date_id	Clave única	Generado
year	Año	pickup_datetime
month	Mes	pickup_datetime
day	Día del mes	pickup_datetime
hour	Hora del día	pickup_datetime
weekday	Día de la semana	pickup_datetime

4.3.2 Dimensión DIM_LOCATION

Describe las ubicaciones relacionadas con los viajes (recogida y destino), basada en taxi_zone_lookup.csv.

Atributo	Descripción	Fuente
location_id	Clave única de ubicación	PUlocationID / DOlocationID
borough	Distrito	taxi_zone_lookup.csv
zone	Nombre de la zona	taxi_zone_lookup.csv
service_zone	Zona de servicio	taxi_zone_lookup.csv

4.3.3 Dimensión DIM_LICENSE

Representa las bases y licencias relacionadas con los viajes.

Atributo	Descripción	Fuente
license_id	Clave única de licencia o base	dispatching_base_num
base_name	Nombre de la base	dispatching_base_num

4.4 Dimensión DIM_RATE (Solo para yellow_tripdata)

Describe los códigos de tarifa (RatecodeID), basada en Rate_code.tab.

Atributo	Descripción	Fuente
rate_code_id	Clave única para el código de tarifa	RatecodeID
description	Descripción del código de tarifa	Rate_code.tab

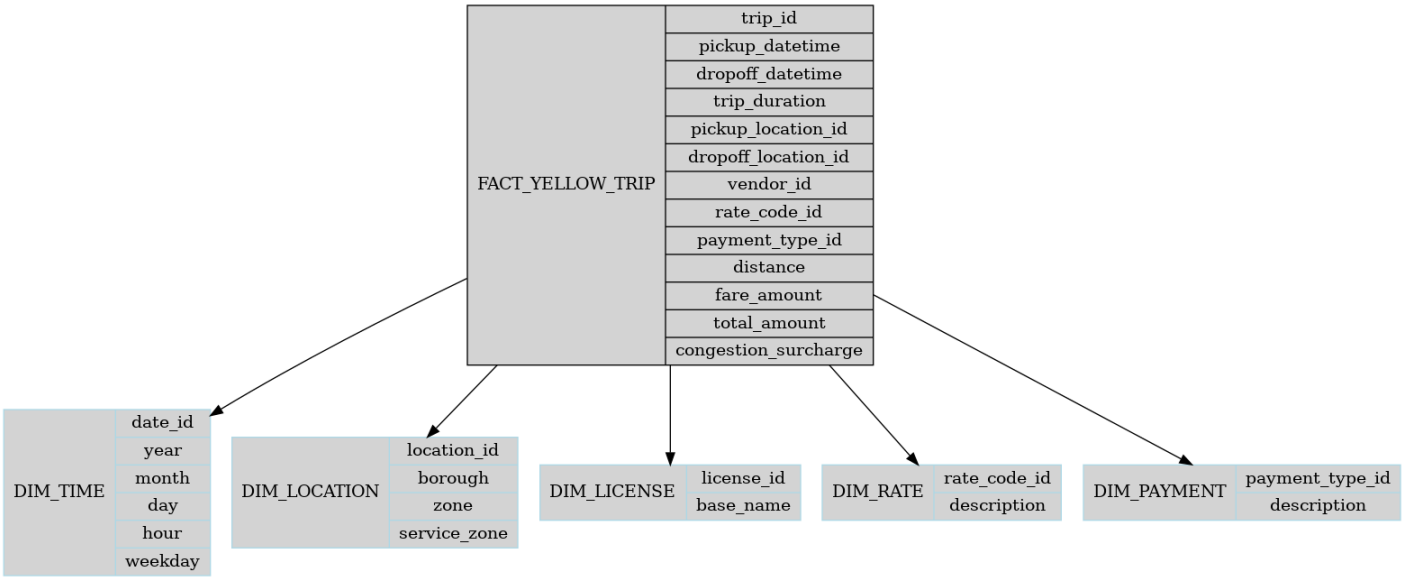
4.5 Dimensión DIM_PAYMENT (Solo para yellow_tripdata)

Describe los métodos de pago, basada en Payment_type.xls.

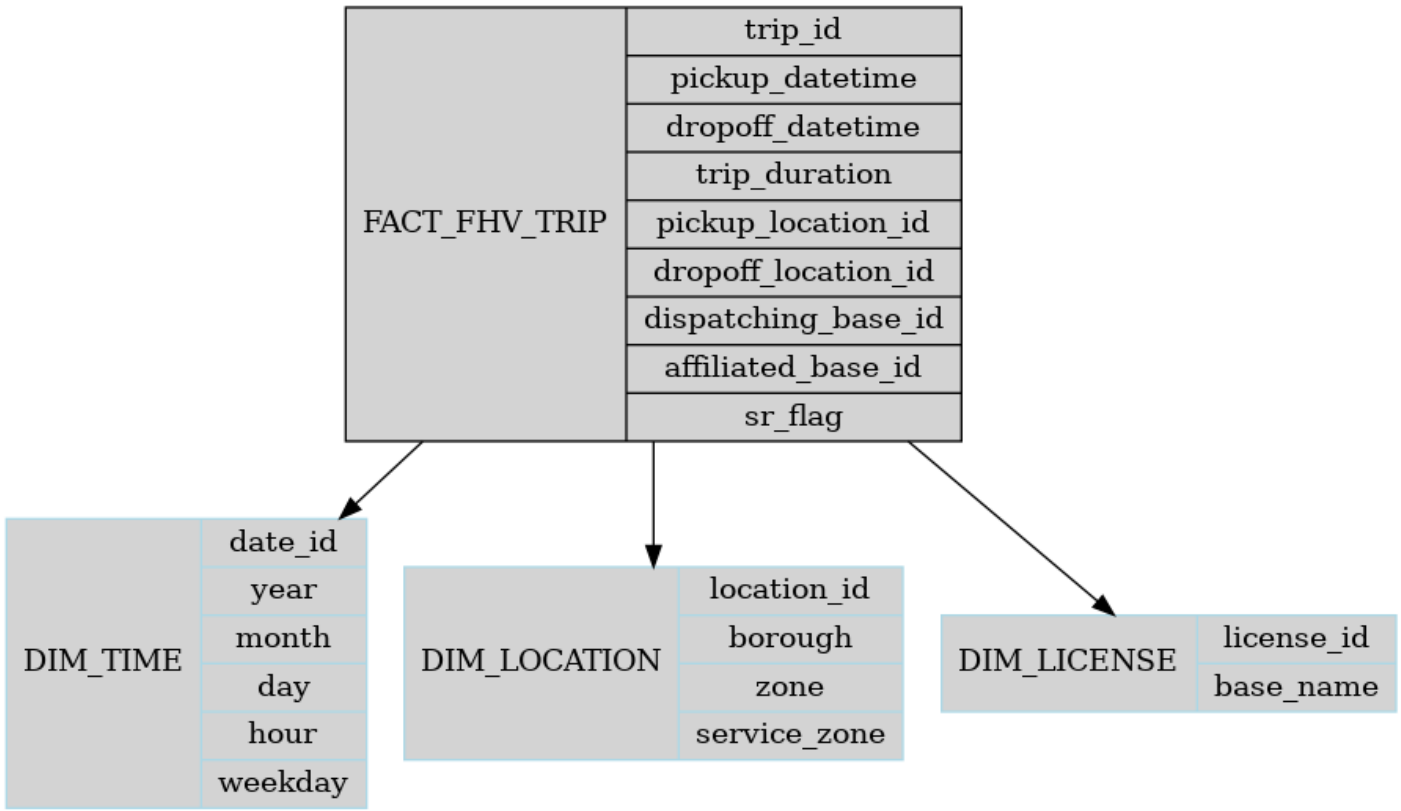
Atributo	Descripción	Fuente
payment_type_id	Clave única para el tipo de pago	payment_type
description	Descripción del método de pago	Payment_type.xls

4.6 Esquema en Estrella

4.6.1 Esquema para FACT_YELLOW_TRIP



4.6.2 Esquema para FACT_FHV_TRIP



5 Diseño Lógico

Se presenta un diseño lógico revisado y optimizado para el modelo de datos `FACT_FHV_TRIP` y sus dimensiones relacionadas.

5.1 Esquema Lógico de `FACT_FHV_TRIP`

El siguiente código crea la tabla `FACT_FHV_TRIP`, que almacena información sobre los viajes realizados por vehículos de alquiler. A continuación, se describen sus columnas principales:

- **Identificador Único:** La columna `fhv_trip_id` utiliza un campo autoincremental `SERIAL` como clave primaria.
- **Relación con Licencias:** La columna `license_id` es una clave foránea que referencia a la tabla `DIM_LICENSE`, asociando cada viaje con una licencia específica.
- **Relación con Tiempos:**
  - `pickup_time_id`: Clave foránea que referencia a la tabla `DIM_TIME`, indicando el momento de inicio del viaje.
  - `dropoff_time_id`: Clave foránea que referencia a la tabla `DIM_TIME`, indicando el momento de finalización del viaje.
- **Relación con Ubicaciones:**
  - `pickup_location_id`: Clave foránea que referencia a la tabla `DIM_LOCATION`, indicando el lugar de recogida.
  - `dropoff_location_id`: Clave foránea que referencia a la tabla `DIM_LOCATION`, indicando el lugar de destino.
- **Indicador de Viaje Compartido:** La columna `sr_flag` es un campo `CHAR(1)` que indica si el viaje fue compartido (`Y` para sí, `N` para no).
- **Número de Afiliación:** La columna `affiliated_num` almacena un número de afiliación único como `VARCHAR(20)`.
- **Duración del Viaje:** La columna `duration_minutes` es un campo `INT` que almacena la duración del viaje en minutos. Este campo es obligatorio.

Estructura

```
CREATE TABLE FACT_FHV_TRIP (
    fhv_trip_id SERIAL PRIMARY KEY,
    license_id INT NOT NULL,
    pickup_time_id INT NOT NULL,
    dropoff_time_id INT NOT NULL,
    pickup_location_id INT NOT NULL,
    dropoff_location_id INT NOT NULL,
    sr_flag CHAR(1),
    affiliated_num VARCHAR(20),
    duration_minutes INT NOT NULL,
    FOREIGN KEY (pickup_time_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (dropoff_time_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (pickup_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (dropoff_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (license_id) REFERENCES DIM_LICENSE(license_id)
);
```

Mejoras

- **Relación directa con `DIM_TIME`:** Se utilizan `pickup_time_id` y `dropoff_time_id` como claves foráneas para garantizar consistencia.
- **Duración en minutos:** Se almacena como un número entero (`duration_minutes`) para simplificar cálculos.
- **Índices:** Se sugieren índices en claves foráneas para acelerar las consultas.

5.2 Esquema Lógico de DIM_TIME

La tabla de tiempo se ajusta para incluir un campo `datetime` y jerarquías temporales para facilitar análisis agregados.

Estructura

```
CREATE TABLE DIM_TIME (
    time_id SERIAL PRIMARY KEY,
    datetime TIMESTAMP NOT NULL, -- Fecha y hora completas
    year INT NOT NULL,
    month INT NOT NULL,
    day INT NOT NULL,
    hour INT NOT NULL,
    weekday VARCHAR(10),
    date DATE NOT NULL -- Solo la fecha
);
```

Mejoras

- **Campo `datetime`:** Representa la fecha y hora completa, simplificando análisis a nivel granular.
- **Jerarquías temporales:** Los campos como `year`, `month`, y `day` permiten consultas agregadas por periodos.

5.3 Esquema Lógico de DIM_LOCATION

La tabla de ubicaciones incorpora soporte opcional para coordenadas geoespaciales.

Estructura

```
CREATE TABLE DIM_LOCATION (
    location_id SERIAL PRIMARY KEY,
    borough VARCHAR(50),
    zone VARCHAR(100),
    service_zone VARCHAR(50),
    coordinates GEOGRAPHY -- Opcional: Coordenadas geoespaciales
);
```

Mejoras

- **Campo `coordinates`:** Permite análisis geoespaciales avanzados utilizando extensiones como PostGIS.
- **Compatibilidad:** Los atributos existentes (`borough`, `zone`, `service_zone`) se mantienen para consultas estándar.

5.4 Esquema Lógico de DIM_LICENSE

La tabla de licencias se actualiza para relacionarse con `DIM_LOCATION` y manejar datos geoespaciales.
El siguiente código crea la tabla `DIM_LICENSE`, que almacena información relacionada con licencias. A continuación, se describen sus columnas principales:

- **Identificador Único:** La columna `license_id` utiliza un campo autoincremental `SERIAL` como clave primaria.
- **Número de Licencia:** La columna `license_num` es un campo `VARCHAR(20)` que almacena un número único para cada licencia. Este campo es obligatorio.
- **Nombre de la Entidad:** La columna `entity_name` permite almacenar el nombre de la base o entidad asociada a la licencia, con un máximo de 100 caracteres.
- **Indicador de Autorización:** La columna `SHL_endorsed` es un valor booleano que indica si la licencia cuenta con autorización especial.
- **Relación con Ubicación:** La columna `location_id` es una clave foránea que referencia a la tabla `DIM_LOCATION`, permitiendo asociar la licencia con una ubicación específica.
- **Coordenadas Opcionales:** La columna `coordinates` es de tipo `GEOGRAPHY`, utilizada para almacenar coordenadas geográficas relacionadas con la ubicación.

Estructura

```
CREATE TABLE DIM_LICENSE (
    license_id SERIAL PRIMARY KEY,
    license_num VARCHAR(20) NOT NULL,
    entity_name VARCHAR(100),
    SHL_endorsed BOOLEAN,
    location_id INT,
    coordinates GEOGRAPHY,
    FOREIGN KEY (location_id) REFERENCES DIM_LOCATION(location_id)
);
```

Mejoras

- **Relación con** `DIM_LOCATION`**:** Permite asociar licencias con zonas específicas.
- **Campo** `coordinates`**:** Admite análisis geoespaciales adicionales si no están cubiertos por `DIM_LOCATION`.

5.5 Relaciones del Esquema en Estrella

El modelo en estrella se refuerza con relaciones consistentes entre `FACT_FHV_TRIP` y sus dimensiones.

Índices Clave

```
CREATE INDEX idx_fhv_pickup_time ON FACT_FHV_TRIP (pickup_time_id);
CREATE INDEX idx_fhv_dropoff_time ON FACT_FHV_TRIP (dropoff_time_id);
CREATE INDEX idx_pickup_location ON FACT_FHV_TRIP (pickup_location_id);
CREATE INDEX idx_dropoff_location ON FACT_FHV_TRIP (dropoff_location_id);
CREATE INDEX idx_license_location ON DIM_LICENSE (location_id);
```

5.6 Manejo de Coordenadas Geoespaciales

El uso de datos geoespaciales se maneja a través del tipo `GEOGRAPHY`. Esto permite análisis como proximidad y distancia.

Habilitación de PostGIS

```
CREATE EXTENSION postgis;
```

Ejemplo de Consulta de Proximidad

Encuentra ubicaciones dentro de un radio de 1 km:

```
SELECT *
FROM DIM_LOCATION
WHERE ST_DWithin(coordinates, ST_GeogFromText('POINT(-73.982419 40.736565)'), 1000);
```

Ejemplo de Cálculo de Distancias

Calcula la distancia entre dos puntos:

```
SELECT ST_Distance(
    ST_GeogFromText('POINT(-73.982419 40.736565)'),
    ST_GeogFromText('POINT(-73.985130 40.748817)')
) AS distance_in_meters;
```

5.7 Diseño Lógico para FACT_YELLOW_TRIP

El siguiente código crea la tabla FACT_YELLOW_TRIP. A continuación, se describen las características principales:

- **Identificador Único:** La columna yellow_trip_id utiliza un campo autoincremental SERIAL como clave primaria.
- **Relación con Dimensiones:**
  - vendor_id: Clave foránea para la tabla DIM_VENDOR.
  - pickup_time_id y dropoff_time_id: Relación con la tabla DIM_TIME.
  - pickup_location_id y dropoff_location_id: Relación con la tabla DIM_LOCATION.
  - rate_code_id: Relación con la tabla DIM_RATE.
  - payment_type_id: Relación con la tabla DIM_PAYMENT.
- **Precisión en Importes Monetarios:** Se utiliza DECIMAL(10,2) para garantizar precisión en cálculos monetarios.
- **Validaciones de Campos Numéricos:** Restricciones CHECK aseguran valores no negativos en los importes como:
  - trip_distance, fare_amount, extra, mta_tax, tip_amount, tolls_amount, y total_amount.
- **Validación de Pasajeros:** Restricción CHECK para garantizar un rango razonable (1-10 pasajeros) en la columna passenger_count.
- **Indicador de store_and_fwd_flag:** Permite analizar registros almacenados antes de ser enviados.

```
CREATE TABLE FACT_YELLOW_TRIP (
    yellow_trip_id SERIAL PRIMARY KEY,
    vendor_id INT NOT NULL,
    pickup_time_id INT NOT NULL,
    dropoff_time_id INT NOT NULL,
    pickup_location_id INT NOT NULL,
    dropoff_location_id INT NOT NULL,
    rate_code_id INT,
    payment_type_id INT,
    passenger_count INT NOT NULL CHECK (passenger_count >= 1 AND passenger_count <= 10),
    trip_distance DECIMAL(10,2) NOT NULL CHECK (trip_distance >= 0),
    fare_amount DECIMAL(10,2) NOT NULL CHECK (fare_amount >= 0),
    extra DECIMAL(10,2) CHECK (extra >= 0),
    mta_tax DECIMAL(10,2) CHECK (mta_tax >= 0),
    tip_amount DECIMAL(10,2) CHECK (tip_amount >= 0),
    tolls_amount DECIMAL(10,2) CHECK (tolls_amount >= 0),
    improvement_surcharge DECIMAL(10,2) CHECK (improvement_surcharge >= 0),
    congestion_surcharge DECIMAL(10,2) CHECK (congestion_surcharge >= 0),
    total_amount DECIMAL(10,2) NOT NULL CHECK (total_amount >= 0),
    store_and_fwd_flag CHAR(1),
    FOREIGN KEY (pickup_time_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (dropoff_time_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (pickup_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (dropoff_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (rate_code_id) REFERENCES DIM_RATE(rate_code_id),
    FOREIGN KEY (payment_type_id) REFERENCES DIM_PAYMENT(payment_type_id),
    FOREIGN KEY (vendor_id) REFERENCES DIM_VENDOR(vendor_id)
);
```

5.7.1 Mejoras Incorporadas

- **Precisión en Importes Monetarios:** Se utiliza DECIMAL(10,2) para garantizar precisión en cálculos monetarios.
- **Validaciones de Campos Numéricos:** Restricciones CHECK aseguran valores no negativos en los importes.
- **Indicador de store_and_fwd_flag:** Añade análisis operativo relevante para registros almacenados.
- **Relación con DIM_VENDOR:** Clave foránea para análisis por proveedor.
- **Validación de Pasajeros:** Restricción CHECK para garantizar un rango razonable (1-10 pasajeros).

5.7.2 Índices y Rendimiento

```
CREATE INDEX idx_yellow_pickup_time ON FACT_YELLOW_TRIP (pickup_time_id);
CREATE INDEX idx_yellow_dropoff_time ON FACT_YELLOW_TRIP (dropoff_time_id);
CREATE INDEX idx_yellow_pickup_location ON FACT_YELLOW_TRIP (pickup_location_id);
CREATE INDEX idx_yellow_dropoff_location ON FACT_YELLOW_TRIP (dropoff_location_id);
CREATE INDEX idx_yellow_payment_type ON FACT_YELLOW_TRIP (payment_type_id);
CREATE INDEX idx_yellow_rate_code ON FACT_YELLOW_TRIP (rate_code_id);
```

5.7.3 Definición de las Dimensiones Asociadas

DIM_VENDOR

```
CREATE TABLE DIM_VENDOR (
    vendor_id INT PRIMARY KEY,
    vendor_name VARCHAR(100)
);
```

DIM_RATE

```
CREATE TABLE DIM_RATE (
    rate_code_id INT PRIMARY KEY,
    description VARCHAR(100)
);
```

DIM_PAYMENT

```
CREATE TABLE DIM_PAYMENT (
    payment_type_id INT PRIMARY KEY,
    description VARCHAR(50)
);
```

5.7.4 Ejemplo de Consultas

Ingresos Totales por Ubicación y Mes

```
SELECT
    dl.zone AS pickup_zone,
    dt.month AS trip_month,
    SUM(fyt.total_amount) AS total_revenue
FROM FACT_YELLOW_TRIP fyt
JOIN DIM_LOCATION dl ON fyt.pickup_location_id = dl.location_id
JOIN DIM_TIME dt ON fyt.pickup_time_id = dt.time_id
GROUP BY dl.zone, dt.month
ORDER BY total_revenue DESC;
```

Viajes por Método de Pago

```
SELECT
    dp.description AS payment_method,
    COUNT(*) AS trip_count
FROM FACT_YELLOW_TRIP fyt
JOIN DIM_PAYMENT dp ON fyt.payment_type_id = dp.payment_type_id
GROUP BY dp.description
ORDER BY trip_count DESC;
```

Distancia Promedio por Proveedor

```
SELECT
    dv.vendor_name,
    AVG(fyt.trip_distance) AS avg_distance
FROM FACT_YELLOW_TRIP fyt
JOIN DIM_VENDOR dv ON fyt.vendor_id = dv.vendor_id
GROUP BY dv.vendor_name
ORDER BY avg_distance DESC;
```

6 Diseño Físico

En esta sección, se detalla el diseño físico del almacén de datos, desarrollado para apoyar el análisis de los registros de viajes de la *New York City Taxi and Limousine Commission* (NYC TLC). Este diseño incluye tanto las dimensiones clave como las tablas de hechos necesarias para manejar grandes volúmenes de datos de manera eficiente.

6.1 Diseño de Dimensiones

6.1.1 DIM_LOCATION

La tabla `DIM_LOCATION` captura información geográfica esencial, utilizada frecuentemente en consultas relacionadas con zonas de recogida y entrega.

Definición:

```
CREATE TABLE DIM_LOCATION (
    location_id INT PRIMARY KEY,
    borough NVARCHAR(50) NOT NULL,
    zone NVARCHAR(100) NOT NULL,
    service_zone NVARCHAR(100) NOT NULL
);
```

Decisiones de diseño:

- **Identificador único**: `location_id` se define como clave primaria para garantizar la integridad de los datos.
- **Tamaño de campos**: Los campos `zone` y `service_zone` tienen una longitud inicial de 100 caracteres, que puede ajustarse a 150 si fuera necesario.
- **Optimización**: Se implementaron índices para acelerar las consultas por borough y zonas de servicio:

  ```
  CREATE INDEX idx_borough ON DIM_LOCATION(borough);
  CREATE INDEX idx_service_zone ON DIM_LOCATION(service_zone);
  ```

6.1.2 DIM_TIME

La dimensión temporal, `DIM_TIME`, está diseñada para proporcionar la granularidad necesaria para análisis por fechas y horas.

Definición:

```
CREATE TABLE DIM_TIME (
    time_id INT PRIMARY KEY,
    date DATE NOT NULL,
    year INT NOT NULL,
    month INT NOT NULL,
    day INT NOT NULL,
    hour INT NOT NULL,
    weekday NVARCHAR(10) NOT NULL
);
```

Decisiones de diseño:

- **Granularidad**: Se incluyeron niveles como año, mes, día y hora para consultas en diferentes periodos de tiempo.
- **Mejoras futuras**: Consideré la posibilidad de agregar campos como `is_weekend`, útil para identificar días no laborales.
- **Índices compuestos**: Para optimizar consultas comunes, definí:

```
CREATE INDEX idx_year_month ON DIM_TIME(year, month);
CREATE INDEX idx_date ON DIM_TIME(date);
```

6.1.3 DIM_LICENSE

La tabla DIM_LICENSE recopila datos sobre las licencias de vehículos y sus bases afiliadas.

Definición:

```
CREATE TABLE DIM_LICENSE (
    license_id INT PRIMARY KEY,
    affiliated_base NVARCHAR(20),
    base_name NVARCHAR(100),
    service_type NVARCHAR(50)
);
```

Decisiones de diseño:

- **Normalización**: Se diseñó una tabla adicional DIM_SERVICE_TYPE para gestionar los tipos de servicio:

```
CREATE TABLE DIM_SERVICE_TYPE (
    service_type_id INT PRIMARY KEY,
    description NVARCHAR(50) NOT NULL
);
```

- **Unicidad**: Para evitar duplicados, implementé una restricción única en affiliated_base:

```
CREATE UNIQUE INDEX uq_affiliated_base ON DIM_LICENSE(affiliated_base);
```

6.2 Diseño de Tablas de Hechos

6.2.1 FACT_FHV_TRIP

La tabla FACT_FHV_TRIP almacena datos de los viajes realizados por vehículos de alquiler con licencia.

Definición:

```
CREATE TABLE FACT_FHV_TRIP (
    fhv_trip_id BIGINT PRIMARY KEY,
    license_id INT NOT NULL,
    pickup_datetime_id INT NOT NULL,
    dropoff_datetime_id INT NOT NULL,
    pickup_location_id INT NOT NULL,
    dropoff_location_id INT NOT NULL,
    sr_flag CHAR(1),
    affiliated_num NVARCHAR(20),
    duration_minutes INT NOT NULL,
    FOREIGN KEY (license_id) REFERENCES DIM_LICENSE(license_id),
    FOREIGN KEY (pickup_datetime_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (dropoff_datetime_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (pickup_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (dropoff_location_id) REFERENCES DIM_LOCATION(location_id)
);
```

Decisiones de diseño:

- **Validación**: Se definió una restricción para asegurar que la duración de los viajes (duration_minutes) sea mayor que cero:

```
CHECK (duration_minutes > 0)
```

- **Índices**: Se crearon índices en los campos de ubicación para optimizar las consultas:

```
CREATE INDEX idx_pickup_location ON FACT_FHV_TRIP(pickup_location_id);
CREATE INDEX idx_dropoff_location ON FACT_FHV_TRIP(dropoff_location_id);
```

6.2.2 FACT_YELLOW_TRIP

La tabla `FACT_YELLOW_TRIP` se diseñó para los viajes de taxis amarillos.

Definición:

```
CREATE TABLE FACT_YELLOW_TRIP (
    yellow_trip_id BIGINT PRIMARY KEY,
    vendor_id INT NOT NULL,
    pickup_datetime_id INT NOT NULL,
    dropoff_datetime_id INT NOT NULL,
    pickup_location_id INT NOT NULL,
    dropoff_location_id INT NOT NULL,
    rate_code_id INT,
    passenger_count INT NOT NULL,
    trip_distance DECIMAL(10,2) NOT NULL,
    fare_amount DECIMAL(10,2) NOT NULL,
    extra DECIMAL(10,2),
    mta_tax DECIMAL(10,2),
    tip_amount DECIMAL(10,2),
    tolls_amount DECIMAL(10,2),
    congestion_surcharge DECIMAL(10,2),
    total_amount DECIMAL(10,2) NOT NULL,
    store_and_fwd_flag CHAR(1),
    FOREIGN KEY (vendor_id) REFERENCES DIM_VENDOR(vendor_id),
    FOREIGN KEY (pickup_datetime_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (dropoff_datetime_id) REFERENCES DIM_TIME(time_id),
    FOREIGN KEY (pickup_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (dropoff_location_id) REFERENCES DIM_LOCATION(location_id),
    FOREIGN KEY (rate_code_id) REFERENCES DIM_RATE(rate_code_id)
);
```

Decisiones de diseño:

- **Restricciones**: Se definieron validaciones para asegurar valores positivos en tarifas y cantidades de pasajeros:

```
CHECK (fare_amount >= 0),
CHECK (total_amount >= 0),
CHECK (passenger_count BETWEEN 1 AND 10)
```

- **Optimización**: Se añadieron índices en los campos de ubicación para consultas rápidas.

6.3 Conclusión

El diseño físico presentado en esta sección asegura un equilibrio entre rendimiento, escalabilidad y flexibilidad. Se han priorizado las consultas analíticas y la facilidad de mantenimiento, sentando una base sólida para la explotación de datos en el ámbito del transporte urbano.

7 Diagrama Entidad-Relación (ER)

El siguiente diagrama representa el modelo Entidad-Relación (ER) correspondiente al diseño físico del almacén de datos. Este modelo incluye las tablas de hechos y dimensiones necesarias para soportar el análisis multidimensional.

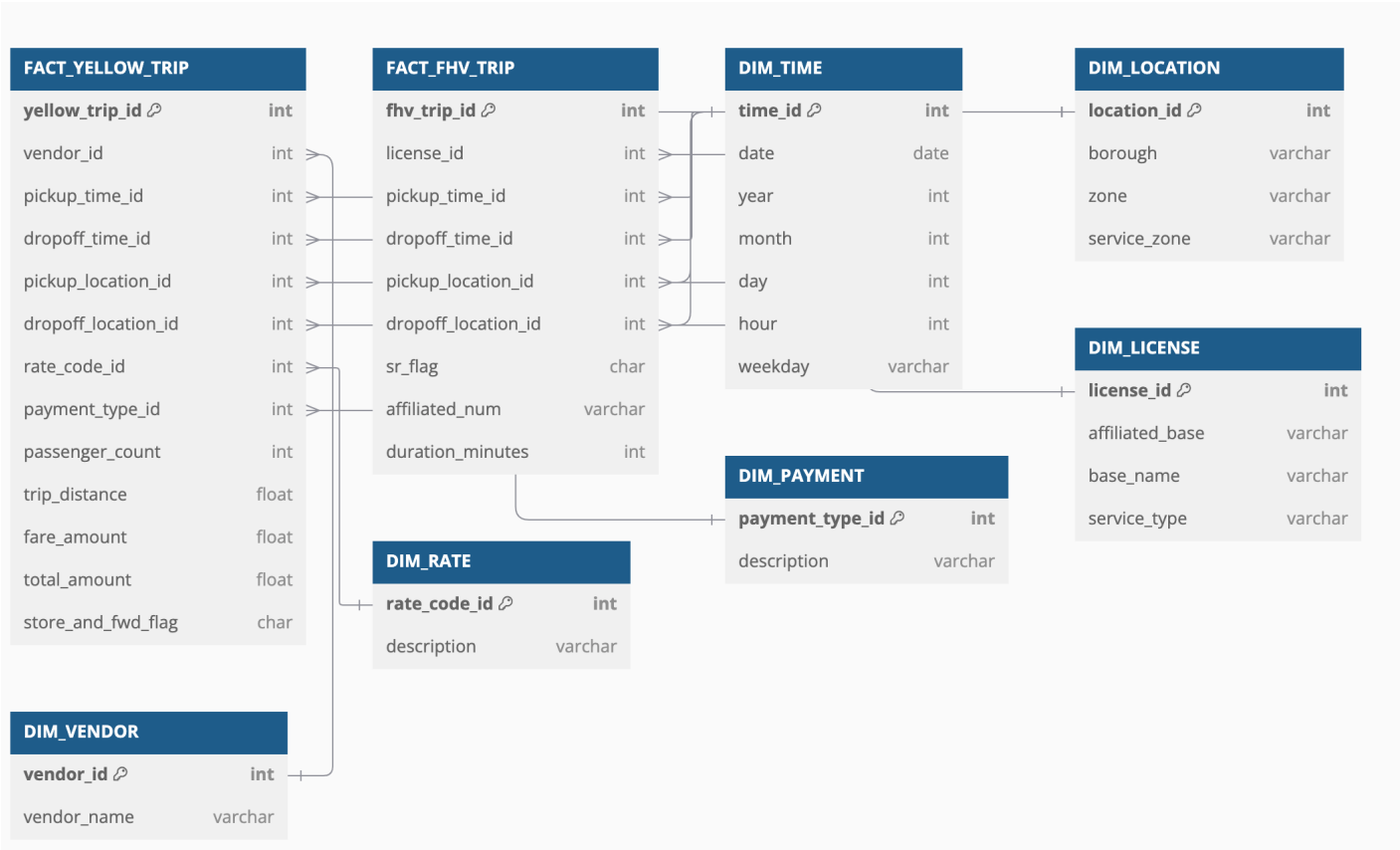


Figure 1: Diagrama Entidad-Relación (ER) del diseño físico