

Projekt Data Warehouse

Bearbeiter: Benedikt Grothues, Julius Seiffert

Datum: 01.07.2016

Data Warehouse

Fakultät Informatik und Medien

Inhalt

Datenquellen	3
Intern	3
Extern	3
Phase 1	4
Adapt-Schema	5
Phase 2	7
Basisdatenbank	7
Mehrdimensionales Datenbankschema	9

Datenquellen

Für das Data Warehouse Projekt wurden folgende Datenquellen ausfindig gemacht.

Intern

- Tlc_green_trips_20<xx> - Fahrten der grünen Taxis in New York von 01.01.2009 bis 31.12.2018
- Tlc_yellow_trips_20<xx> - Fahrten der gelben Taxis in New York von 01.08.2013 bis 31.08.2018
- Taxi_zone_geom – Zonen, in denen Fahrer Passagiere aufnehmen, oder absetzen

Verwendete Spalten aus der Tabelle tlc_green_trips:

Spaltenname	Datentyp	Inhalt
PICKUP_DATETIME	datetime	Datum, Uhrzeit taximeter einschaltzeit
PICKUP_LONGITUDE	number	Längengrad, Beginn der Taxifahrt
PICKUP_LATITUDE	number	Breitengrad, Beginn der Taxifahrt
DROPOFF_LONGITUDE	number	Längengrad, Ende der Taxifahrt
DROPOFF_LATITUDE	number	Breitengrad, Ende der Taxifahrt
PASSENGER_COUNT	number	Anzahl der Passagiere
TIP_AMOUNT	number	Höhe des Trinkgelds
TOTAL_AMOUNT	number	Gesamtbetrag, den der Passagier zu bezahlen hat - ohne Trinkgelder in bar

Verwendete Spalten aus der Tabelle tlc_yellow_trips:

Spaltenname	Datentyp	Inhalt
PICKUP_DATETIME	datetime	Datum, Uhrzeit taximeter einschaltzeit
PASSENGER_COUNT	number	Anzahl der Passagiere
TIP_AMOUNT	number	Höhe des Trinkgelds
TOTAL_AMOUNT	number	Gesamtbetrag, den der Passagier zu bezahlen hat - ohne Trinkgelder in bar
PICKUP_LOCATION_ID	string	TLC-Taxizone, Beginn der Fahrt
DROPOFF_LOCATION_ID	string	TLC-Taxizone, Ende der Fahrt

Aus der Tabelle taxi_zone_geom:

Spaltenname	Datentyp	Inhalt
ZONE_ID	string	Datum, Uhrzeit taximeter einschaltzeit
ZONE_NAME	string	Name der Taxizone als Volltext
BOROUGH	string	Borough, in der sich die Taxizone befindet
ZONE_GEOM	geography	GeometrieDaten der Taxizone GIS

Extern

- Wetterdaten: git – jfk_weather.csv – Wetterdaten des JFK Flughafen von 01.01.2009 bis 30.06.2016
- <https://catalog.data.gov/> - Sehenswürdigkeiten in NewYork

Phase 1

Datenverarbeitungsanforderungen: Auswertung der Taxidaten im Zusammenhang mit gegebenen Wetterdaten und Daten über den Ort von Sehenswürdigkeiten (points of interest, POI) in New York. Ziel ist die Analyse von Zusammenhängen zwischen Wetter und den zu Besuchenden Sehenswürdigkeiten. Dazu sollen die Taxifahrten, welche an bestimmten Orten ihr Ziel haben über die Tage mit den Wetterdaten ausgewertet werden. Für die Klassifikation des Wetters ist auf den Wassertyp und die Tagesdurchschnittstemperatur zu schauen.

Die Genauigkeit der geografischen Position von Beginn und Ende der Taxifahrten, sowie dem Ort der Sehenswürdigkeit wird auf die Taxizone reduziert.

Über die Geografischen Koordinaten (Longitude, Latitude) der POIs kann auf die Taxizone geschlossen werden. Dasselbe gilt für die grünen Taxis. Die Fahrtentabelle der gelben Taxis ist bereits mit den Zonen versehen

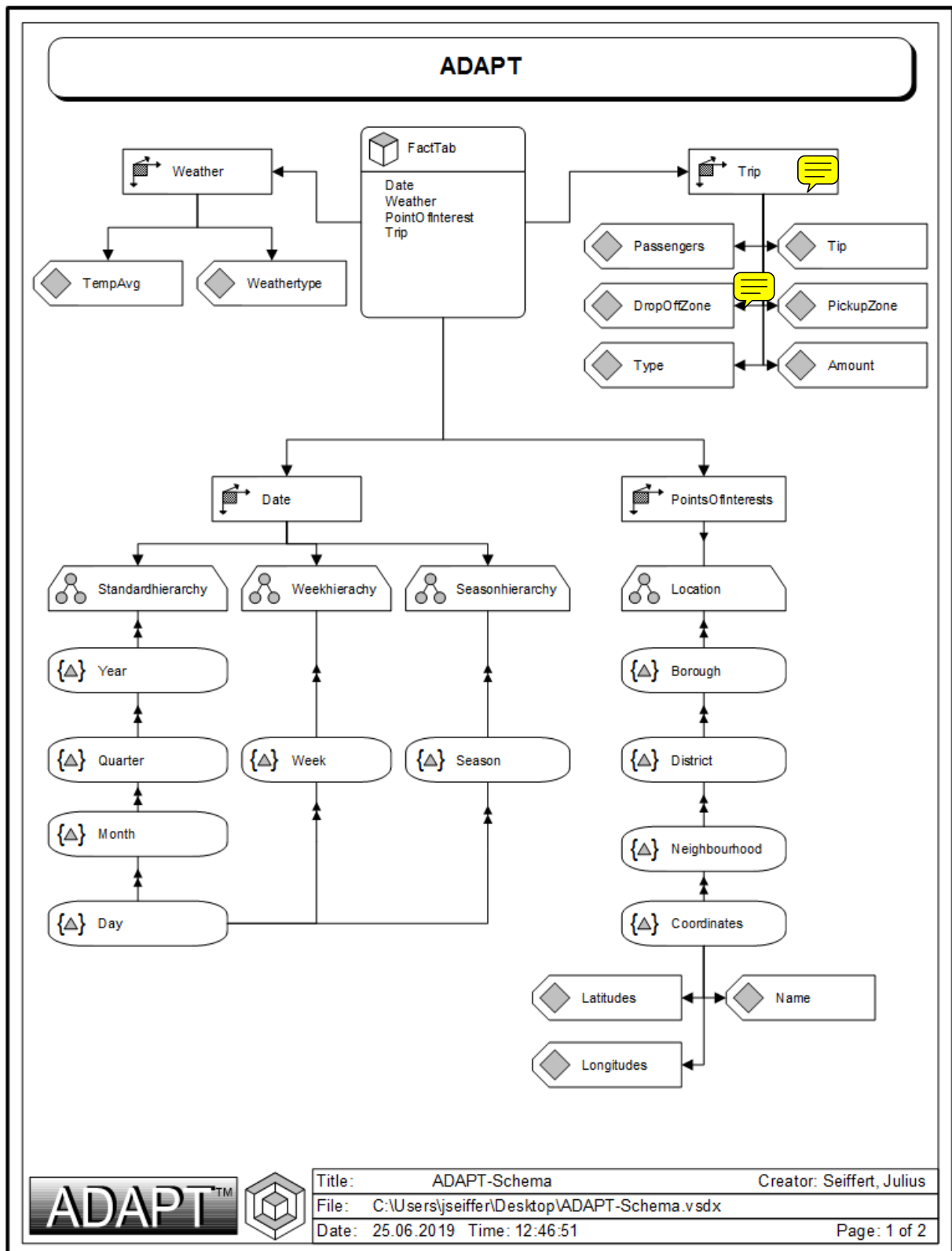
Dieses Datawarehouse soll in folgenden Fragestellungen unterstützen:

1. Welche Sehenswürdigkeiten werden am meisten zu welchem Wetter angefahren?
2. Bei welchem Wetter wird am meisten Taxi gefahren?
3. Bei welchen Sehenswürdigkeiten bekommt der Fahrer das höchste Trinkgeld?
4. Welche Sehenswürdigkeiten werden in welchen Jahreszeiten am häufigsten besucht.

Hierfür werden folgende Fakten und Maßzahlen benötigt.

- Datum (Fakt)
- Wassertyp (Fakt)
- Temperatur (Fakt)
- Sehenswürdigkeiten (Fakt)
- Summe der Fahrten - Maßzahl
- Summe der Passagiere - Maßzahl
- Trinkgeld pro Passagier bei Kreditkarte – Maßzahl

Adapt-Schema



Im ADAPT-Schema lassen sich die 4 Dimensionen und die Faktentabelle sehen. Bei den Dimensionen werden die Hierarchiestufen, sowie die Attribute angezeigt.

Bei der Dimension Date, gibt es als Besonderheit 3 Hierarchien:

- **Standardhierarchie:** Bei der Standardhierarchie wird das Datum nach Jahr, Quartal, Monat und Tag unterteilt.
- **Wochenhierarchie:** Bei der Wochenhierarchie wird einem Tag eine Wochennummer zugeordnet. Die Woche kann nicht in der Standardhierarchie untergebracht werden, da eine Wochennummer selbst zwei Jahre beinhalten kann.
- **Jahreszeithierarchie:** Bei dieser Hierarchie wird die Jahreszeit eingeführt. Diese kann nicht in den anderen Hierarchien untergebracht werden, da eine Jahreszeit Wochen und Jahre überschneiden kann.

Bei der Dimension **PointsOfInterests** gibt es eine Hierarchie, in der die Koordinaten der Sehenswürdigkeiten einem *Borough*, einem *District* und einem *Neighbourhood* zugeordnet werden kann.

Die Dimension **Trip** hat keine Hierarchiestufen. Als Attribute werden hier die Passagieren einer Fahrt, das Trinkgeld, den Aufnahmeort, den Zielort, die Art des Taxis und den Betrag der Fahrt aufgelistet.

Die Dimension **Weather** hat keine Hierarchiestufen. Hier werden als Attribute die Durchschnittstemperatur und die Art des Wetter abgebildet.

Phase 2

Basisdatenbank

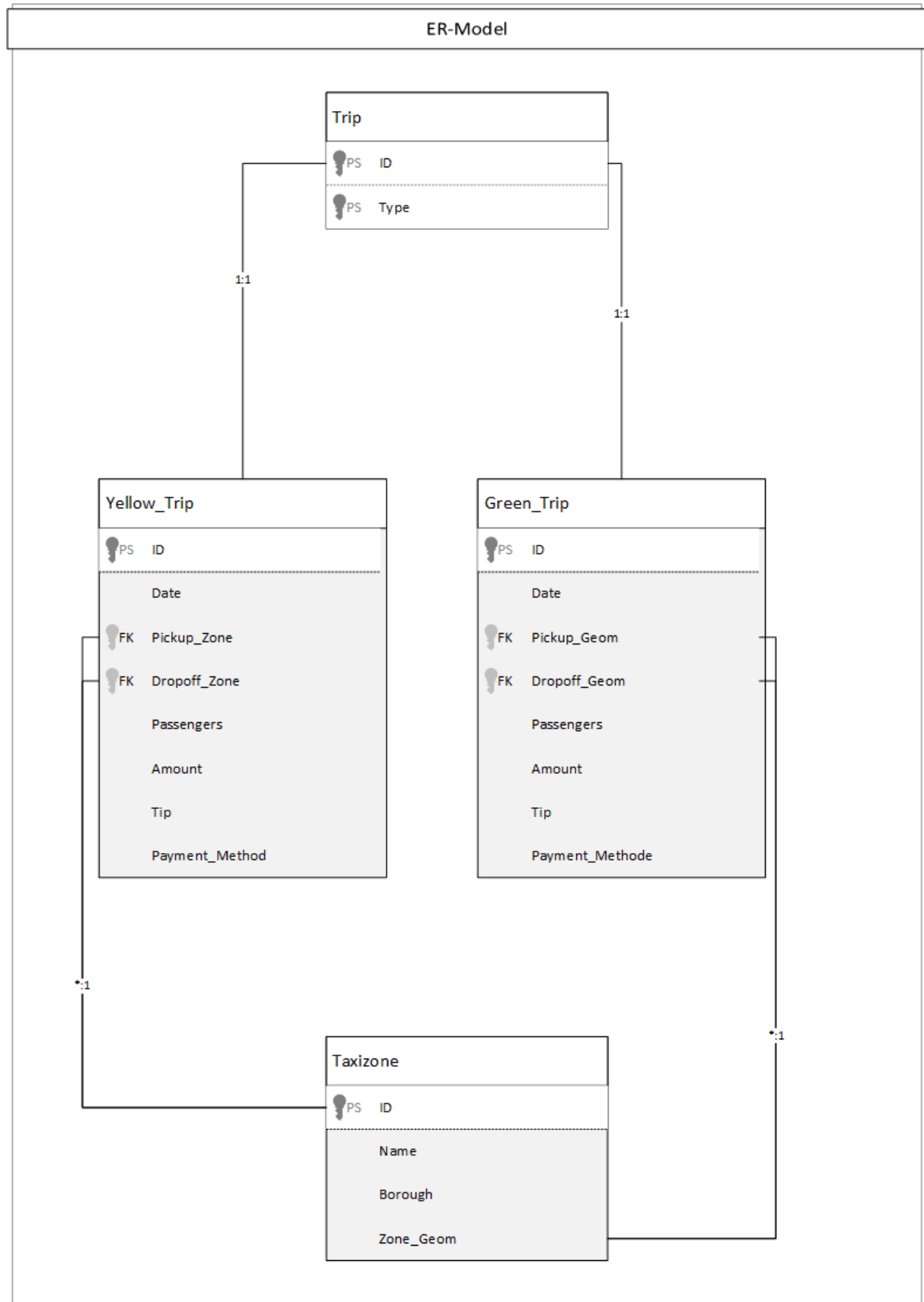



Abbildung 1: ER-Modell

In Phase 2 wurde die Basisdatenbank, sowie das MDM-Schema des Data Warehouse entworfen.

In der Basisdatenbank wird in der Tabelle **Trip** auf die Tabellen **Yellow_Trip** und **Green_Trip**. Je nach Tabelle bekommt **Trip** ein Attribut *Type*.

Bei den Fahrten der gelben Taxis wird der die Zonen ID des Aufnahme- und Zielortes mit angegeben. Hier kann direkt auf die Tabelle **Taxizone** referenziert werden.

Bei den Fahrten der grünen Taxis werden leider nur die Koordinaten des Aufnahme- und des Zielortes angegeben. Hier können die Koordinaten dem Attribut *Zone_Geom* zugeordnet werden, welches einen größeren geometrischen Bereich abdeckt. 

Mehrdimensionales Datenbankschema

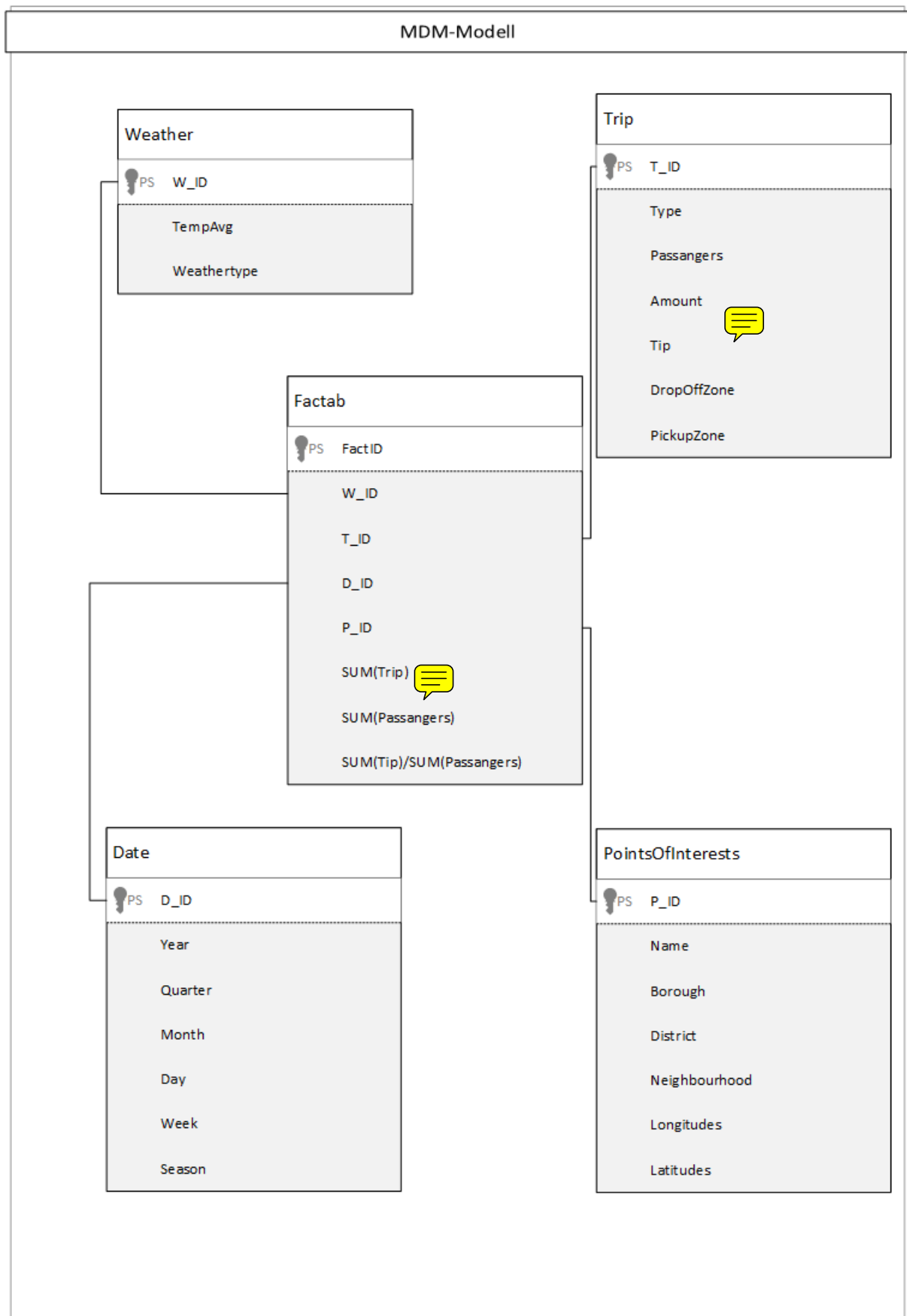


Abbildung 2: MDM-Schema

Als mehrdimensionales Datenbankschema wird das Starschema benutzt. Hierbei gibt es eine Faktentabelle und mehrere Dimensionstabellen, die allerdings nicht weiter unterteilt werden.

Die Tabelle **Facttab** enthält die Fremdschlüssel zu den anderen Dimensionen, sowie die Maßzahlen.

Die Dimensionen enthalten alle in der eigenen Dimensionstabelle Ihre Hierarchiestufen. Hier werden keine extra Tabellen angelegt, um über Tabellen auf die verschiedenen Hierarchiestufen zuzugreifen, wie es beim Snowflake-Schema der Fall gewesen wäre.