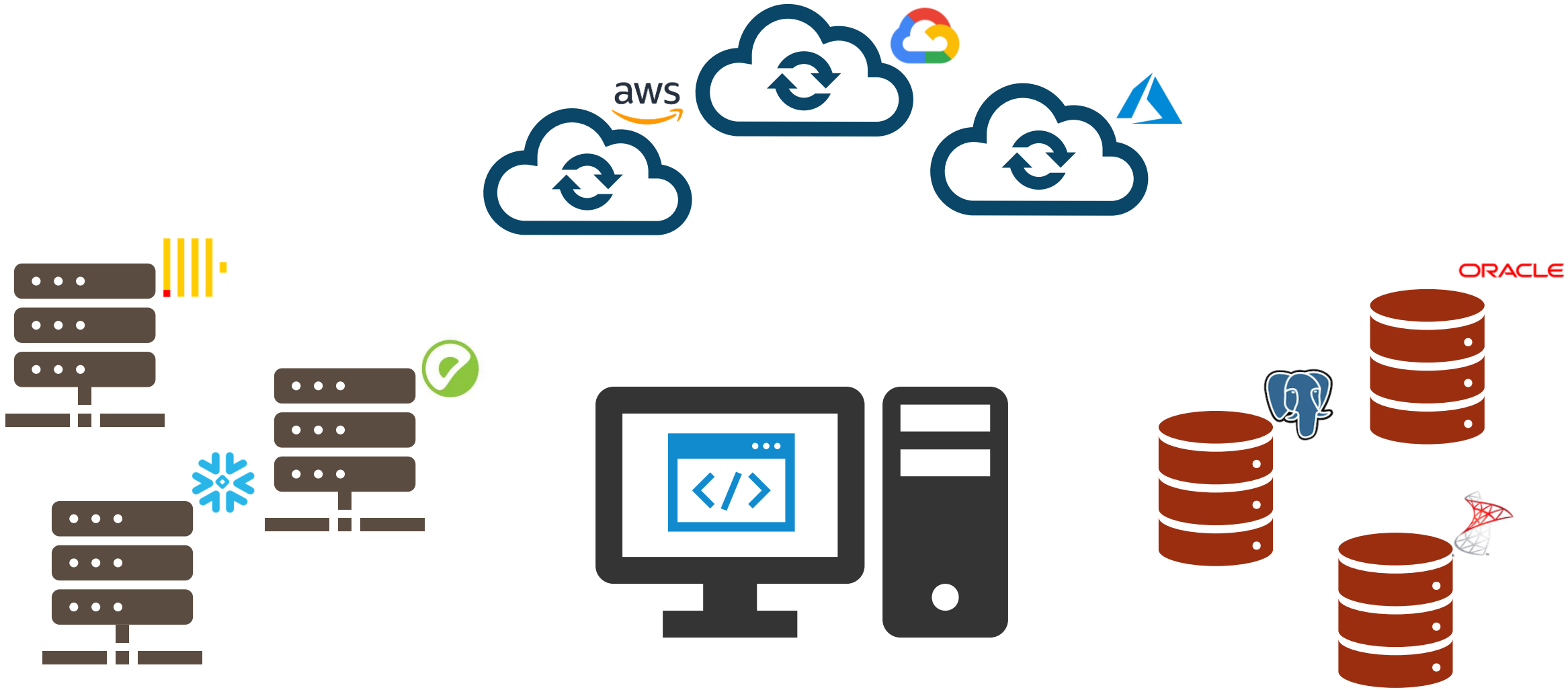
FOREIGN TABLES MODELLING IN UI

TATIANA KRUPENYA AND SERGE RIDER

dbeaver.com

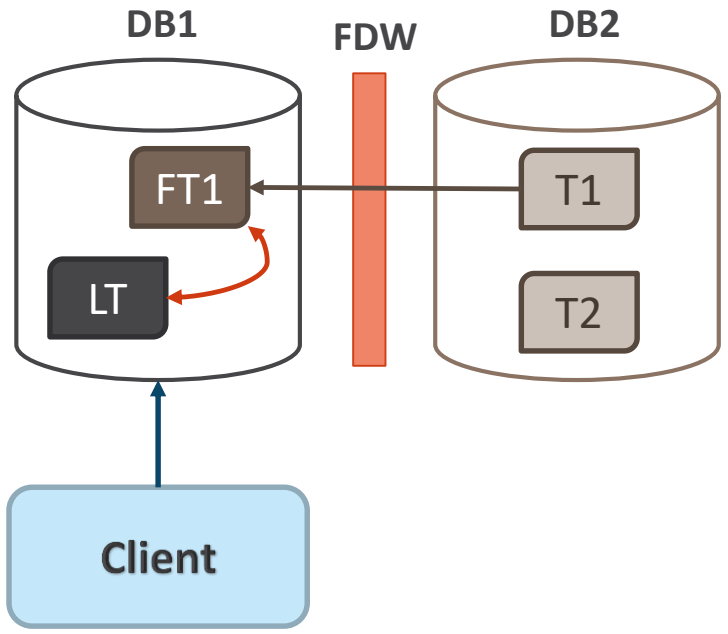
MULTI-DATABASE WORLD



CROSS-DATABASE LINKS

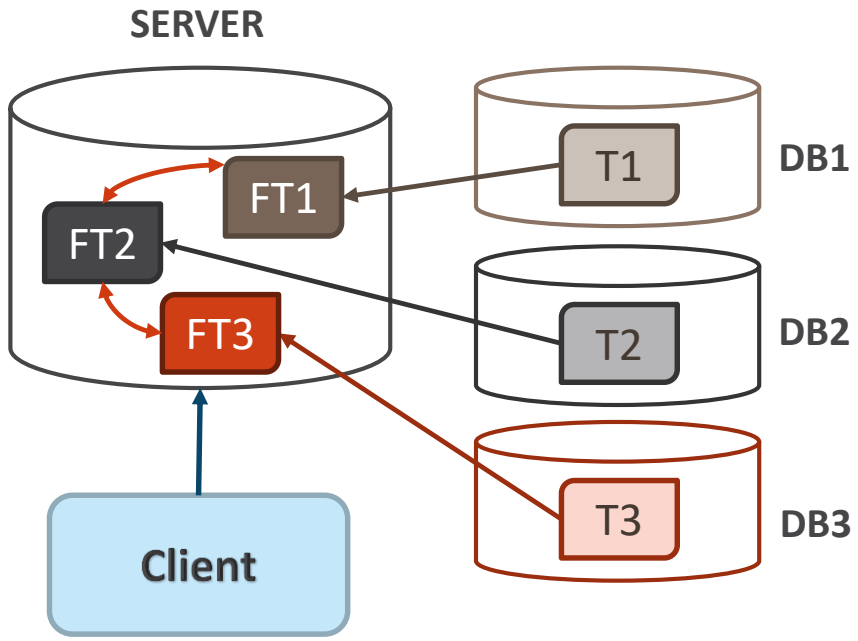
Native

- PostgreSQL FDW
- Oracle Database Links
- DB2 Federated tables



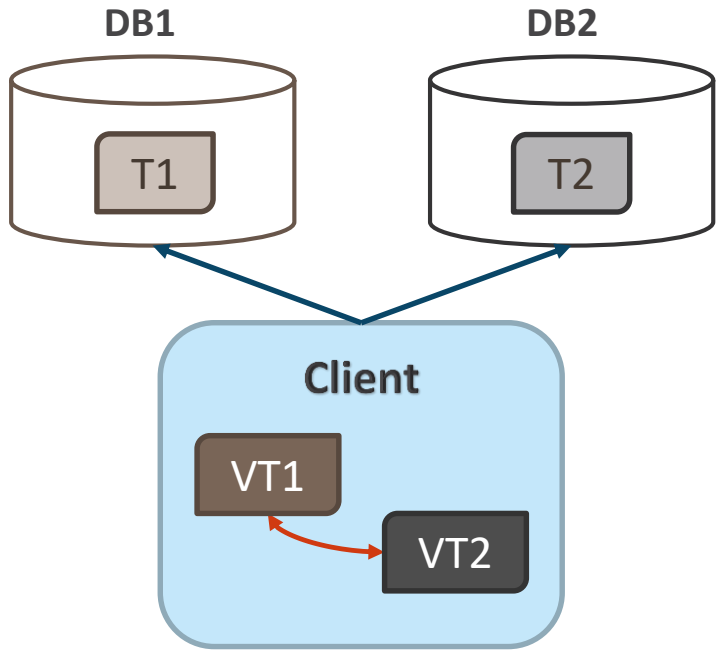
"Composite" Server

- Presto DB



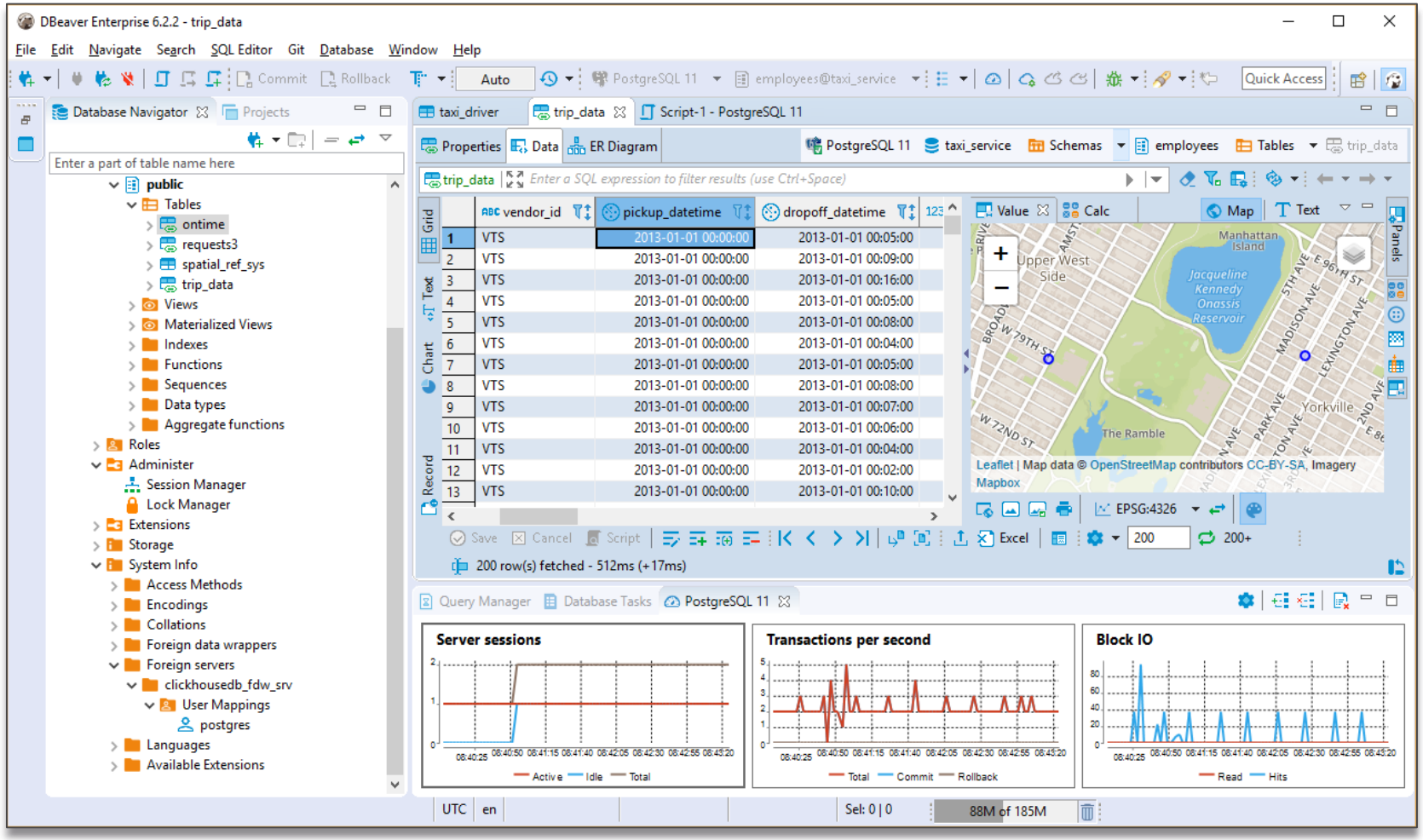
Client side

- Unity JDBC
- DBeaver VFK



DBEAVER FOR POSTGRESQL

- Metadata viewer and editor
- Custom data types rendering/editing
- GIS data viewer
- Session manager
- Locks manager
- SQL query execution plan
- Administrative tools: vacuum, analyze
- User/role permissions editor
- Server health dashboards
- Foreign data wrappers management
- Backup/restore wizards



STORY ABOUT THE LOST WALLET

A person has taken a taxi around 11 am near the Water str. 77 and left a wallet in the car.

Can we find it?

WHAT DO WE HAVE?

Data about the taxi drivers:

- Only information about taxi drivers
- A few gigabytes of data
- Structured data

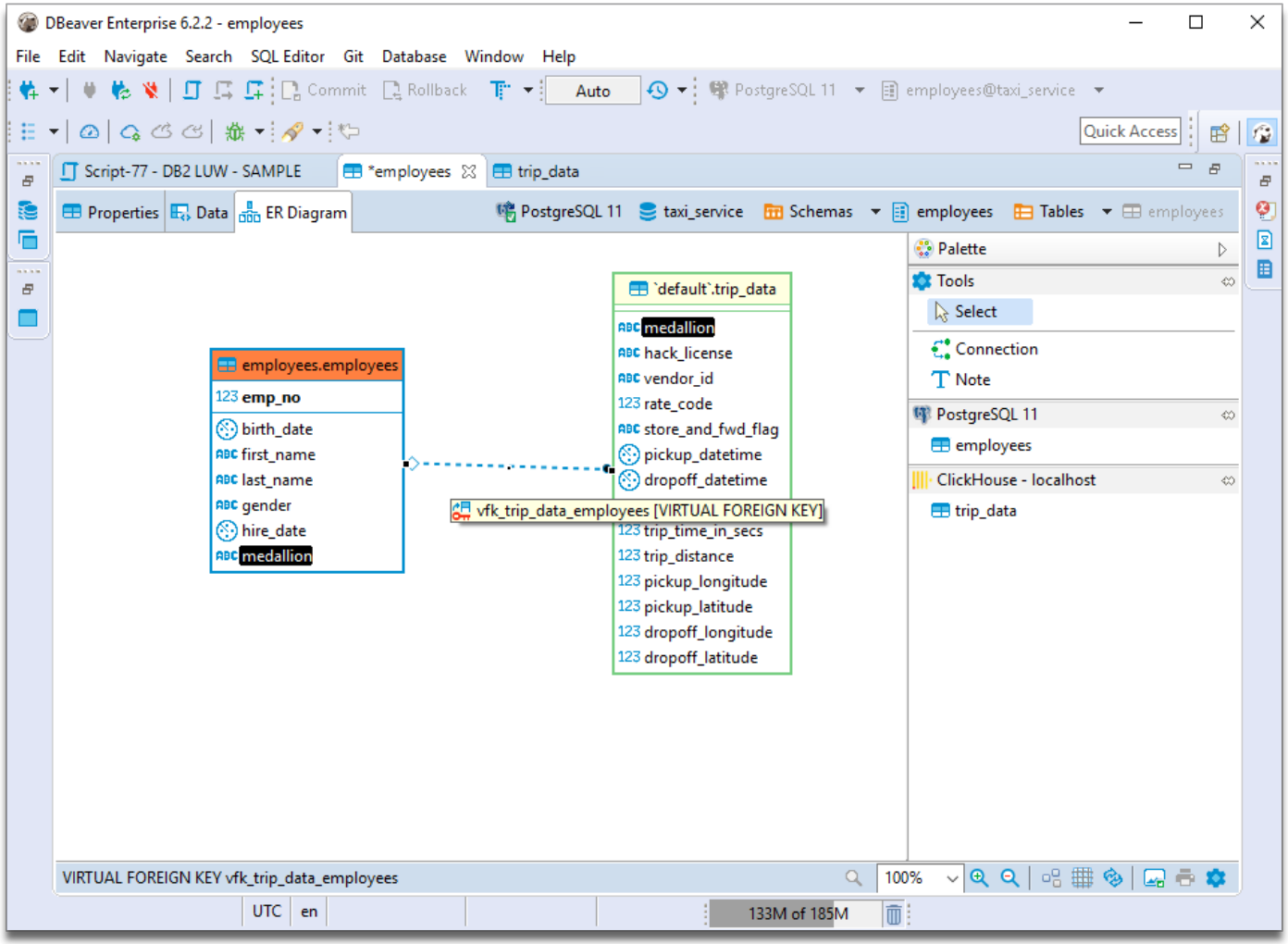
Statistic data about the trips:

- Data about all taxi trips
- A few terabytes of data
- Support analytical queries

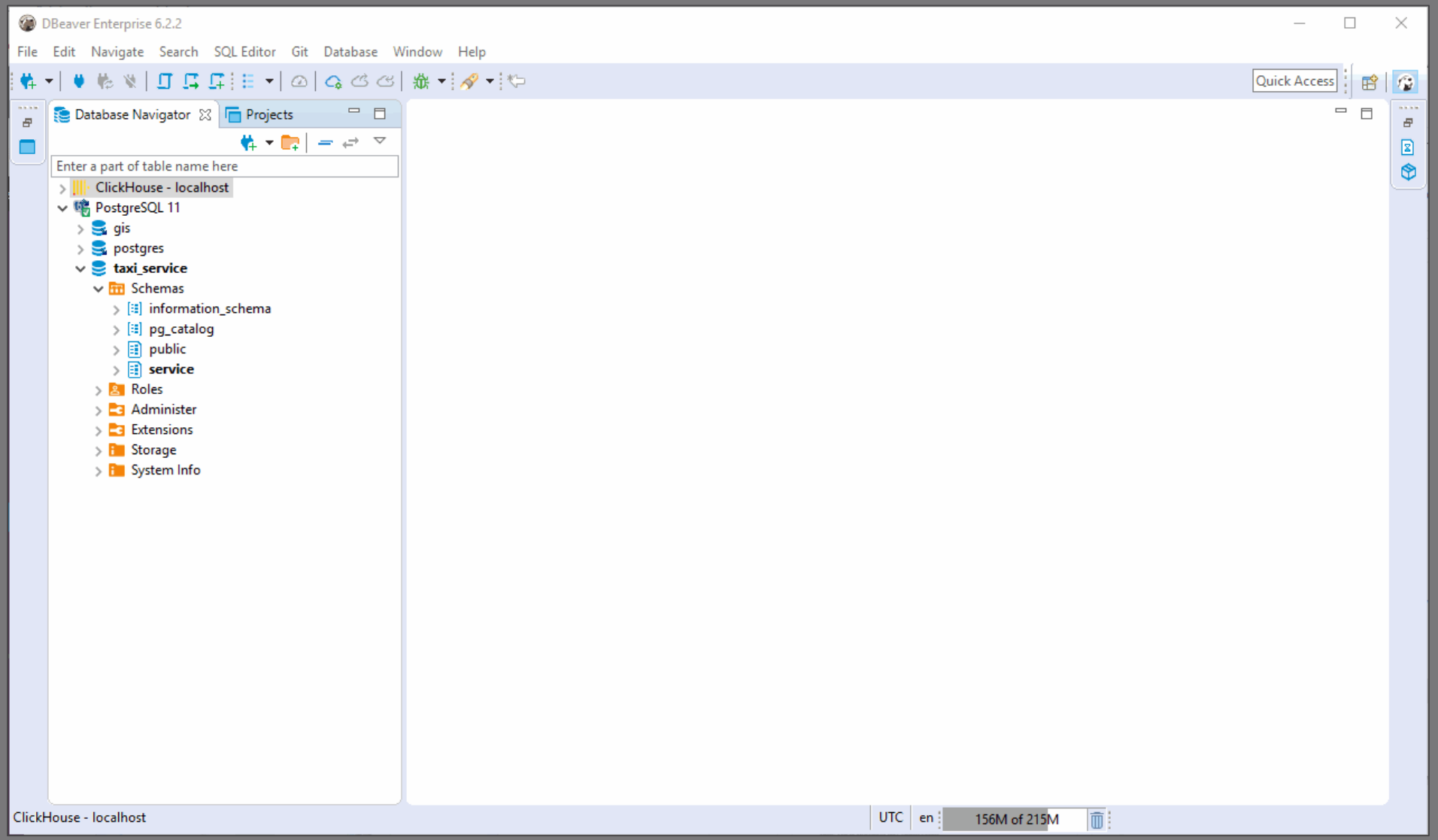
MAKE VIRTUAL CONNECTIONS

Virtual foreign keys

- Allows navigation across tables
- Show referenced data from different databases in one table
- Can connect any databases, including non-relational
- Don’t provide referential integrity



HOW TO CREATE VFK



MAKE REAL CONNECTIONS

Foreign Data Wrappers

Benefits:

- ✓ Create logical tables and use them just like local ones
- ✓ Join tables from different databases

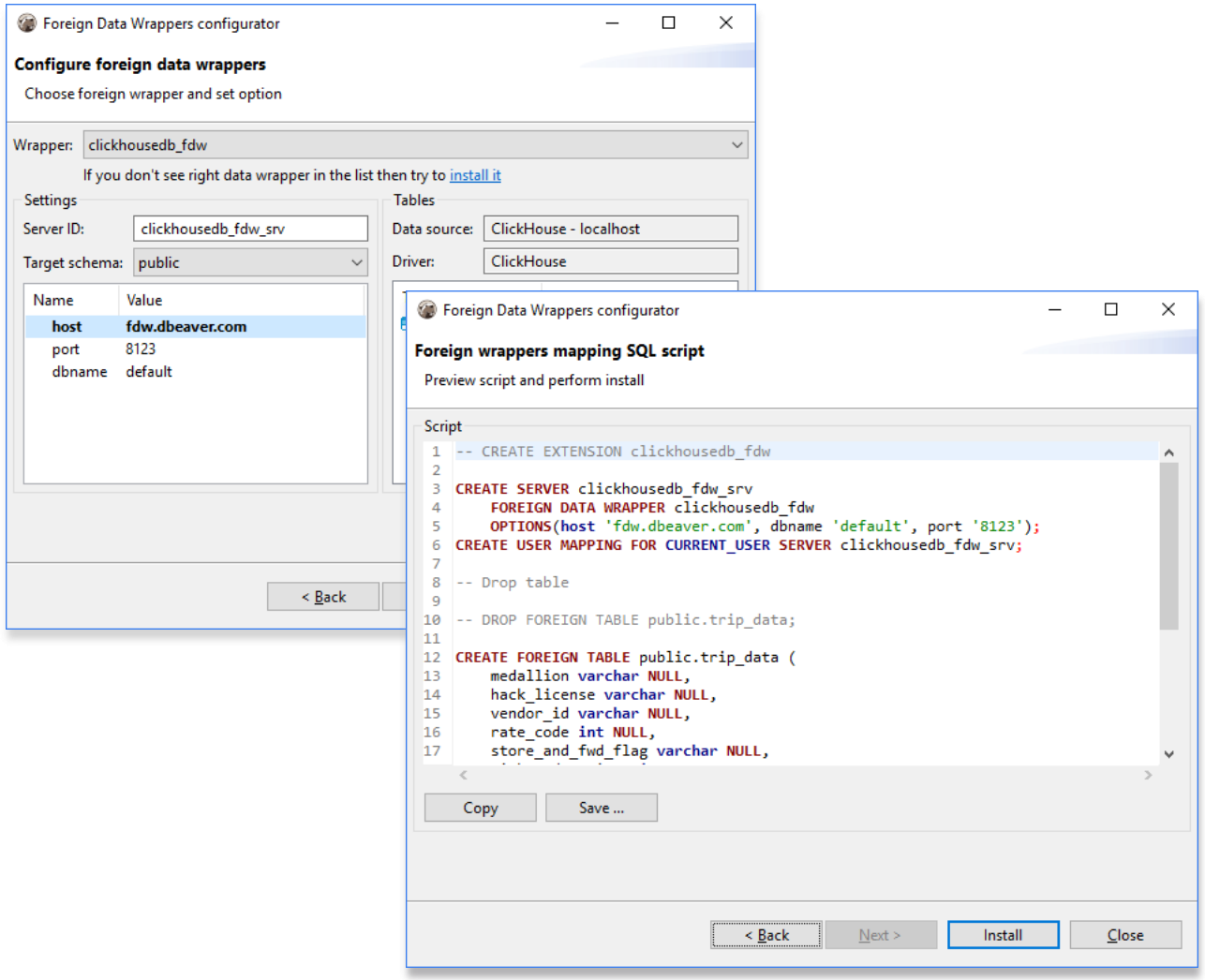
Problems:

- ✗ FDW extension installation can be tricky
- ✗ Difficult to configure mappings
- ✗ Potential performance problems

FDW INSTALL SCRIPT

```
CREATE SERVER clickhousedb_fdw_srv
FOREIGN DATA WRAPPER clickhousedb_fdw
OPTIONS(host '127.0.0.1', dbname '${database}', port '39523');
CREATE USER MAPPING FOR CURRENT_USER SERVER clickhousedb_fdw_srv;
-- Drop table
-- DROP FOREIGN TABLE public.ontime;
CREATE FOREIGN TABLE public.ontime (
"Year" int NULL,
"Quarter" int NULL,
"Month" int NULL,
"DayofMonth" int NULL,
"DayOfWeek" int NULL,
"FlightDate" timestamp NULL,
"UniqueCarrier" varchar NULL,
"AirlineID" int NULL,
"Carrier" varchar NULL,
"TailNum" varchar NULL,
"FlightNum" varchar NULL,
"OriginAirportID" int NULL,
"OriginAirportSeqID" int NULL,
"OriginCityMarketID" int NULL,
"Origin" varchar NULL,
"OriginCityName" varchar NULL,
"OriginState" varchar NULL,
"OriginStateFips" varchar NULL,
"OriginStateName" varchar NULL,
"OriginWac" int NULL,
"DestAirportID" int NULL,
"DestAirportSeqID" int NULL,
"DestCityMarketID" int NULL,
"Dest" varchar NULL,
"DestCityName" varchar NULL,
"DestState" varchar NULL,
"DestStateFips" varchar NULL,
"DestStateName" varchar NULL,
"DestWac" int NULL,
"CRSDepTime" int NULL,
"DepTime" int NULL,
"DepDelay" int NULL,
"DepDelayMinutes" int NULL,
"DepDel15" int NULL,
"DepartureDelayGroups" varchar NULL,
"DepTimeBlk" varchar NULL,
"TaxiOut" int NULL,
"WheelsOff" int NULL,
"WheelsOn" int NULL,
"TaxiIn" int NULL,
"CRSArrTime" int NULL,
"ArrTime" int NULL,
"ArrDelay" int NULL,
"ArrDelayMinutes" int NULL,
"ArrDel15" int NULL,
"ArrivalDelayGroups" int NULL,
"ArrTimeBlk" varchar NULL,
"Cancelled" int NULL,
"CancellationCode" varchar NULL,
"Diverted" int NULL,
"CRSElapsedTime" int NULL,
"ActualElapsedTime" int NULL,
"AirTime" int NULL,
"Flights" int NULL,
"Distance" int NULL,
"DistanceGroup" int NULL,
"CarrierDelay" int NULL,
"WeatherDelay" int NULL,
"NASDelay" int NULL,
"SecurityDelay" int NULL,
"LateAircraftDelay" int NULL,
"FirstDepTime" varchar NULL,
"TotalAddGTime" varchar NULL,
"LongestAddGTime" varchar NULL,
"DivAirportLandings" varchar NULL,
"DivReachedDest" varchar NULL,
"DivActualElapsedTime" varchar NULL,
"DivArrDelay" varchar NULL,
"DivDistance" varchar NULL,
"Div1Airport" varchar NULL,
"Div1AirportID" int NULL,
"Div1AirportSeqID" int NULL,
"Div1WheelsOn" varchar NULL,
"Div1TotalGTime" varchar NULL,
"Div1LongestGTime" varchar NULL,
"Div1WheelsOff" varchar NULL,
"Div1TailNum" varchar NULL,
"Div2Airport" varchar NULL,
"Div2AirportID" int NULL,
"Div2AirportSeqID" int NULL,
"Div2WheelsOn" varchar NULL,
"Div2TotalGTime" varchar NULL,
"Div2LongestGTime" varchar NULL,
"Div2WheelsOff" varchar NULL,
"Div2TailNum" varchar NULL,
"Div3Airport" varchar NULL,
"Div3AirportID" int NULL,
"Div3AirportSeqID" int NULL,
"Div3WheelsOn" varchar NULL,
"Div3TotalGTime" varchar NULL,
"Div3LongestGTime" varchar NULL,
"Div3WheelsOff" varchar NULL,
"Div3TailNum" varchar NULL,
"Div4Airport" varchar NULL,
"Div4AirportID" int NULL,
"Div4AirportSeqID" int NULL,
"Div4WheelsOn" varchar NULL,
"Div4TotalGTime" varchar NULL,
"Div4LongestGTime" varchar NULL,
"Div4WheelsOff" varchar NULL,
"Div4TailNum" varchar NULL,
"Div5Airport" varchar NULL,
"Div5AirportID" int NULL,
"Div5AirportSeqID" int NULL,
"Div5WheelsOn" varchar NULL,
"Div5TotalGTime" varchar NULL,
"Div5LongestGTime" varchar NULL,
"Div5WheelsOff" varchar NULL,
"Div5TailNum" varchar NULL
)
SERVER clickhousedb_fdw;
```

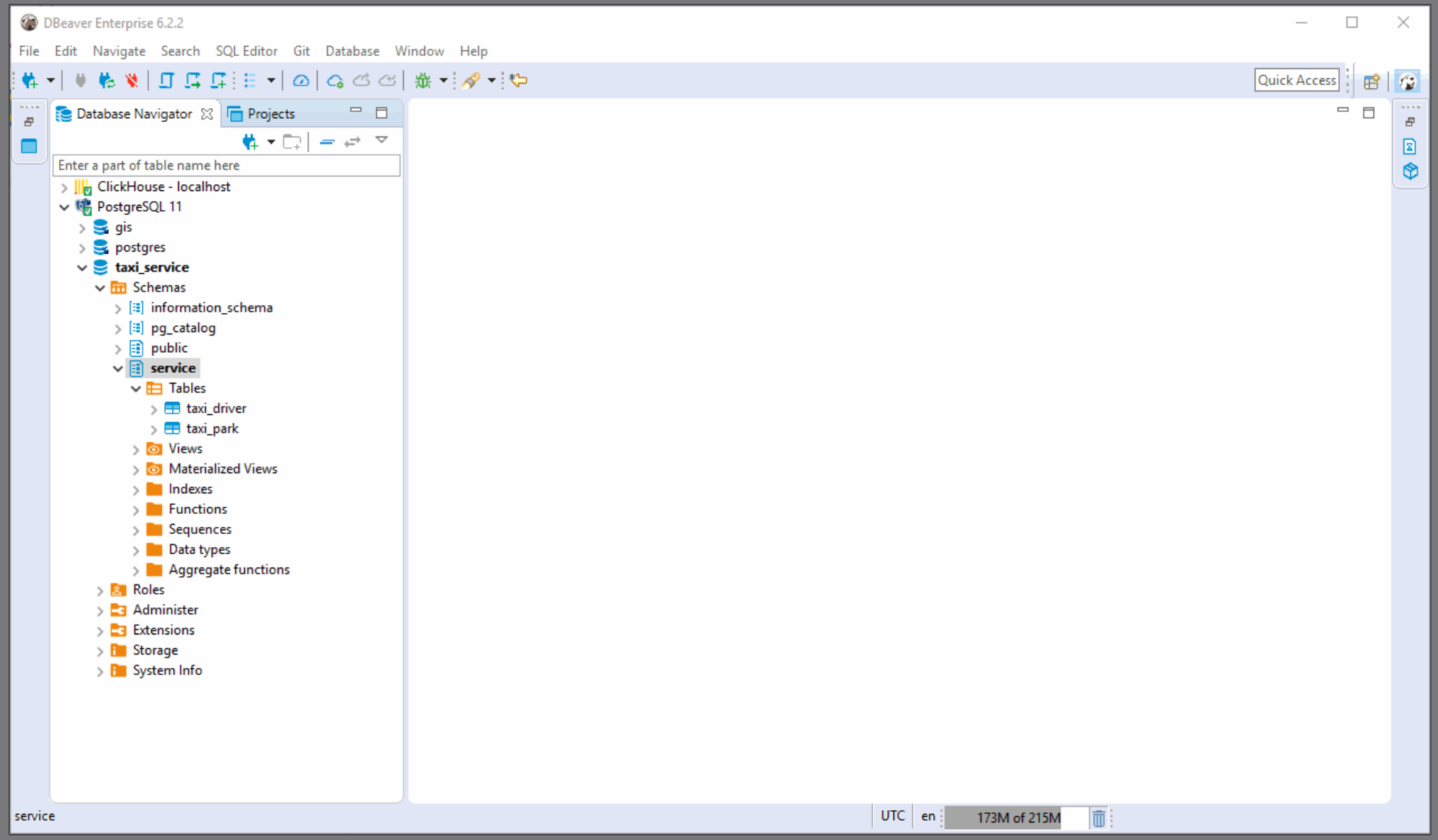
MAKE FDW IN DBEAVER



Why is it easier with DBeaver?

- ✓ Connections (VFK) visualization
- ✓ Foreign data viewer
- ✓ Generate FDW script in a few extra clicks
- ✓ Automatic data types mappings
- ✓ Automatic assignment to proper FDW extension

HOW TO CREATE FDW



BE CAREFUL WITH JOINS

- Foreign keys between local and foreign tables exist only in DBeaver logical model.
- Performance tuning may be tricky. Do not use complex joins/subqueries in cross-database selects.
- Some FDW may work slowly or fail when using range/equals conditions.

RETURN TO OUR PROBLEM

WHAT DO WE KNOW?

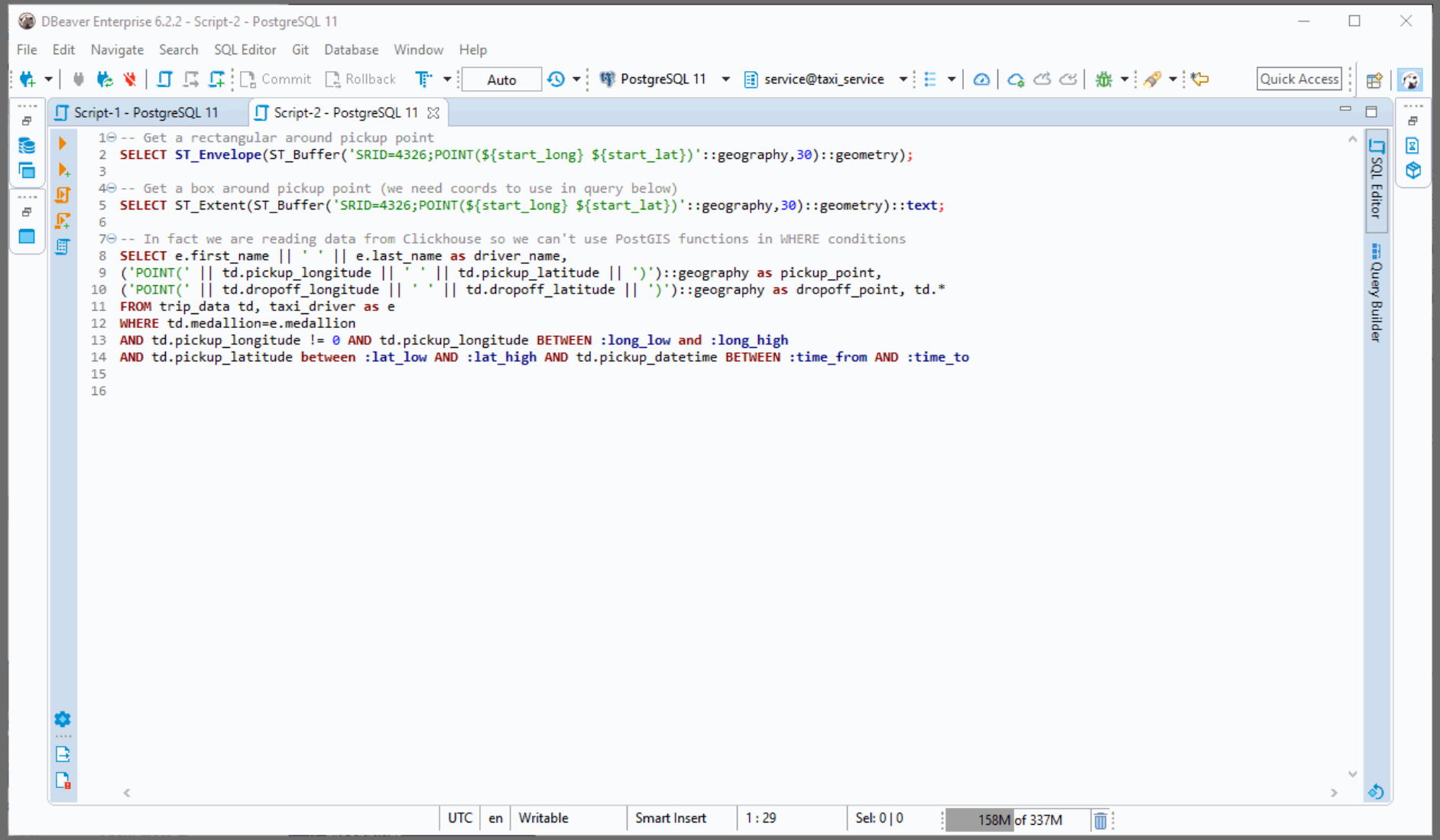
- Pick-up location: Water street 77
- Drop-off location: Columbus Circle
- Pick-up time: around 10:45

SOLUTION

1. Find all trips which started in this area in 30min interval around pick-up time
2. Choose trips which ended in approximate drop-off location
3. Join trip data with drivers data to find driver name/ID.

PROFIT!!!

HOW DOES IT WORK?



CONCLUSION

- FDWs are very handy in some cases
- Be careful with functions. Queries with foreign tables can't use PG functions if foreign database is not PostgreSQL.
- Usually FDW don't provide best performance but they are much easier than any custom development.

USEFUL LINKS

USEFUL LINKS

- FDW list: https://wiki.postgresql.org/wiki/Foreign_data_wrappers
- Clickhouse FDW: https://github.com/Percona-Lab/clickhousedb_fdw
- New York TLC database (tip data): https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

FOLLOW US

- Twitter: https://twitter.com/dbeaver_news
- GitHub: https://github.com/dbeaver/
- DBeaver EE: https://dbeaver.com/
- DBeaver CE: https://dbeaver.io/

CONACT US

- CEO: tati@dbeaver.com
- CTO: serge@dbeaver.com