W A L D

A Modern & Sustainable Analytics Stack

PyConDE / PyData 2023, April 17th

Florian Wilhelm

inovex

Dr. Florian Wilhelm

inovex • HEAD OF DATA SCIENCE

@FlorianWilhelm

Florian Wilhelm

FlorianWilhelm.info

florian.wilhelm@inovex.de

- Mathematical Modelling
- Modern Data Warehousing & Analytics
- Personalisation & RecSys
- Uncertainty Quantification & Causality
- Python Data Stack
- OSS Contributor & Creator of PyScaffold

inovex

inovex

is an innovation and quality-driven IT project house with a focus on digital transformation.

- Application Development (Web Platforms, Mobile Apps, Smart Devices and Robotics, UI/UX design,Backend Services)
- Data Management and Analytics (Business Intelligence, Big Data, Searches, Data Science and Deep Learning, Machine Perception and Artificial Intelligence)
- Scalable IT-Infrastructures (IT Engineering, Cloud Services, DevOps, Replatforming, Security)
- Training and Coaching (inovex Academy)

Using technology to inspire our clients.
And ourselves.

www.inovex.de

Berlin · Karlsruhe · Pforzheim · Stuttgart · München · Köln · Hamburg · Erlangen

Data Warehouse

vs

Data Lake

&

ETL

vs

ELT

inovex

COMPARISON

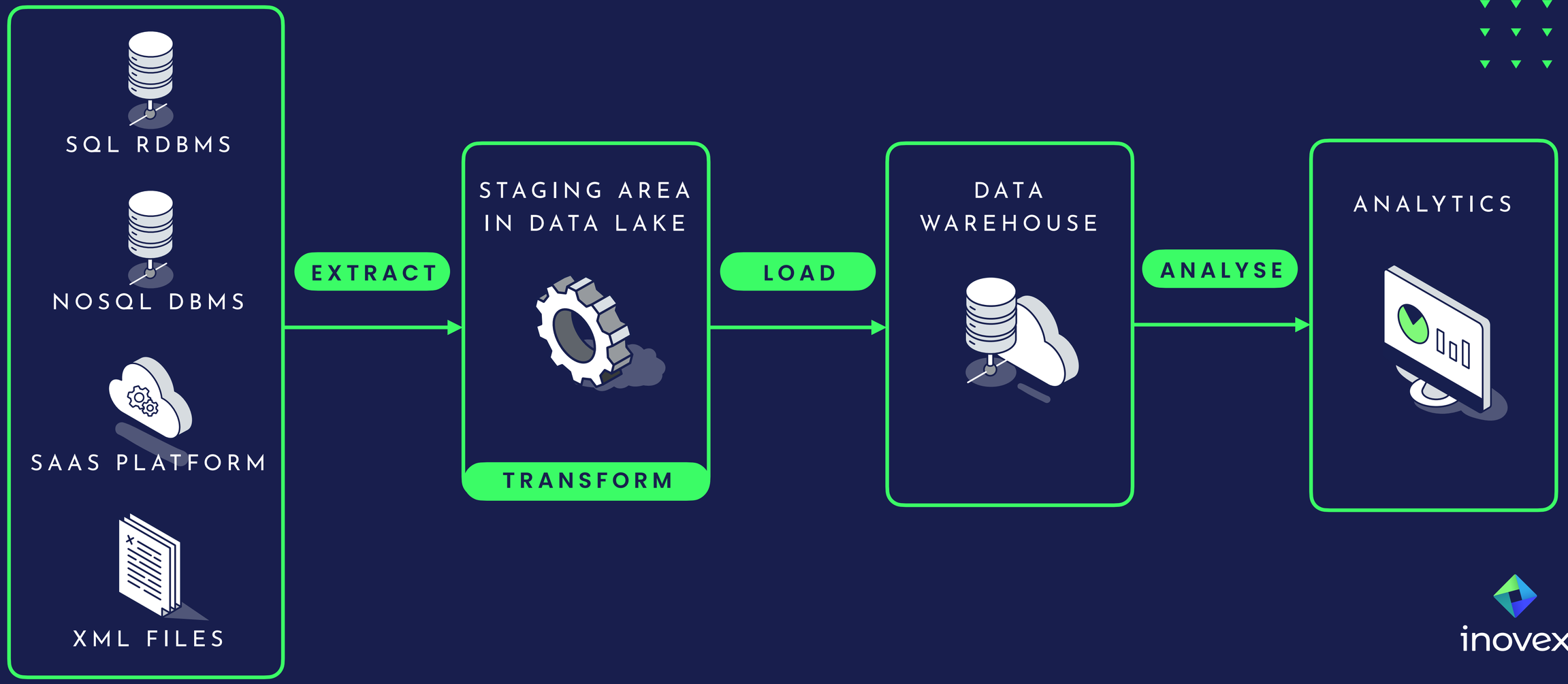
Data Lake vs Data Warehouse

	Data Lake	Data Warehouse
Data Structure	Raw	Processed
Purpose of Data	Not yet determined	Currently in use
Users	Data scientists / engineers	Business professionals
Accessibility	Highly accessible and quick to update	More complicated and costly to make changes

inovex

COMPARISON

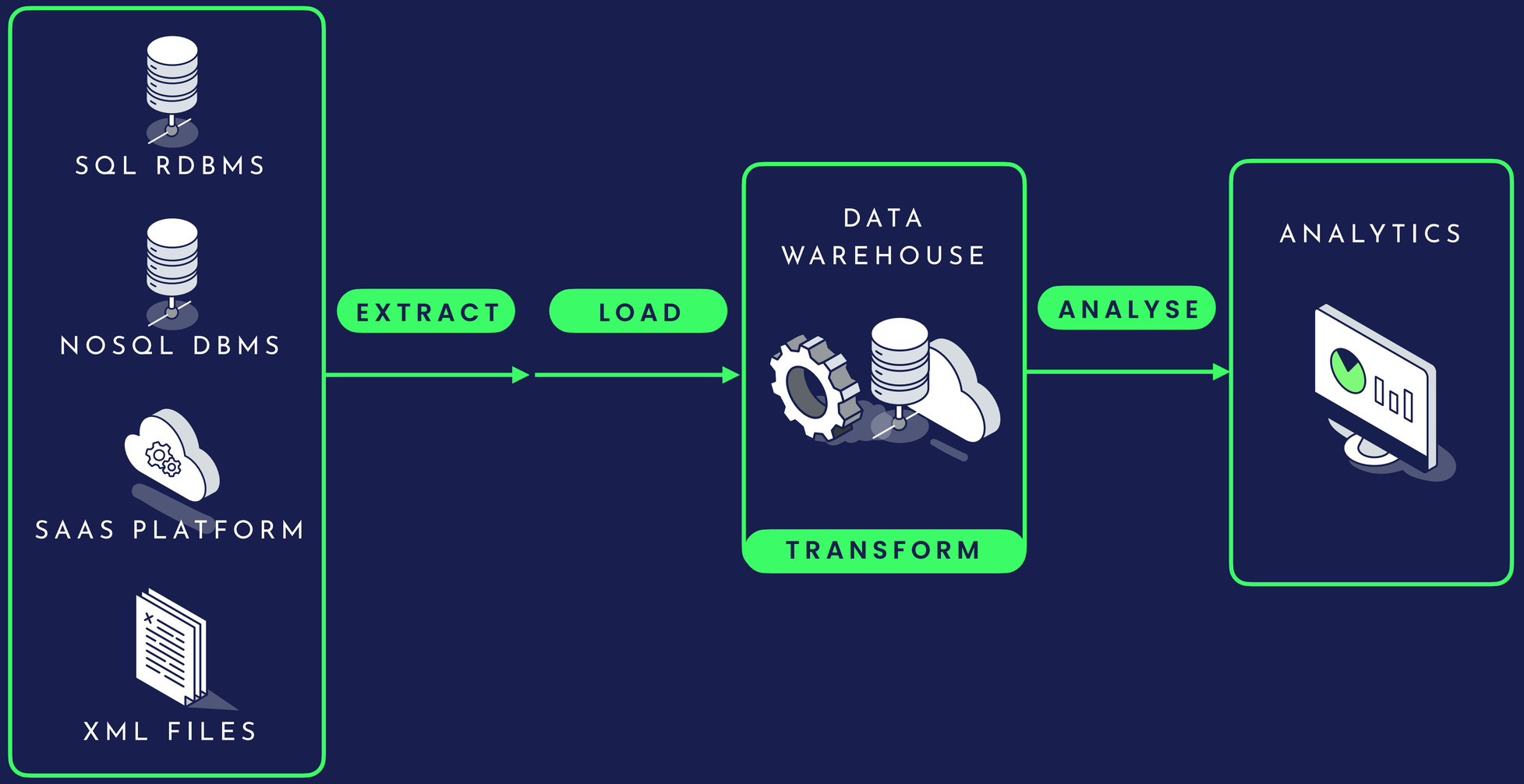
ETL — Extract, Transform, Load



inovex

COMPARISON

ELT — Extract, Load, Transform



inovex

PROGRESS

The Evolution of Databricks & Snowflake

databricks

DATA LAKE

& ETL

snowflake

DATA WAREHOUSE

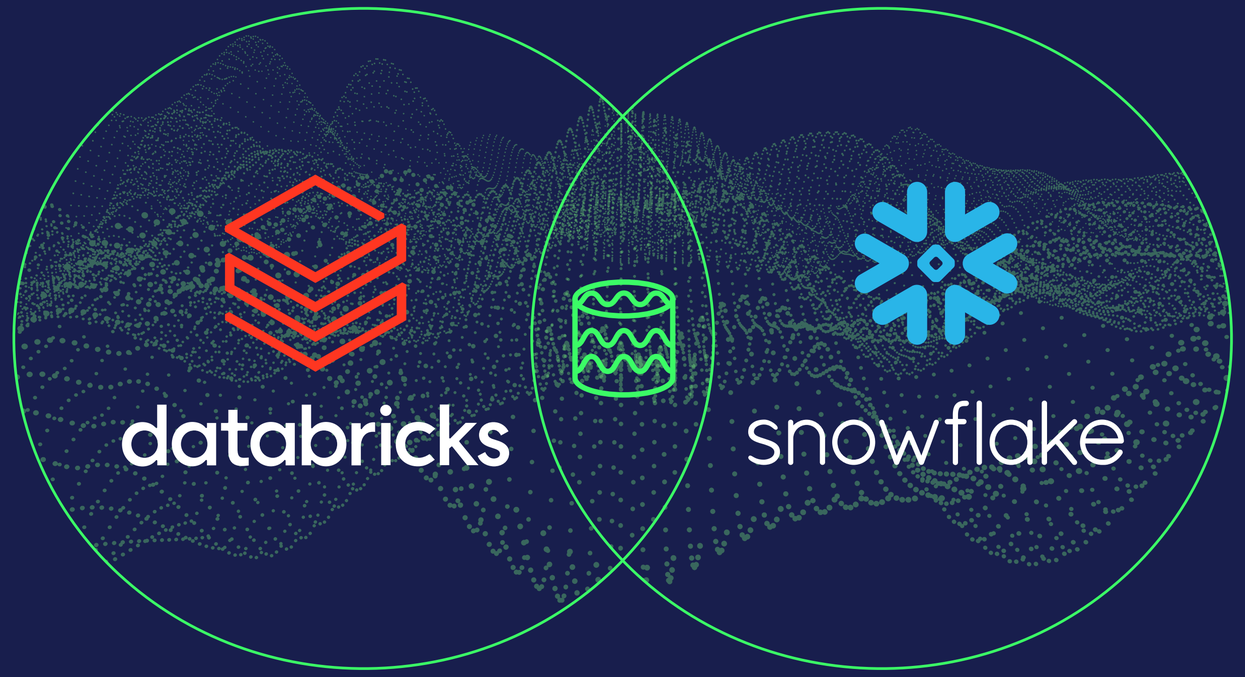
& ELT

inovex

PROGRESS

The Evolution of Databricks & Snowflake

DATABRICKS
LAKEHOUSE



SNOWFLAKE
DATA CLOUD

UNIFICATION
OF DATA WAREHOUSE & LAKE

inovex

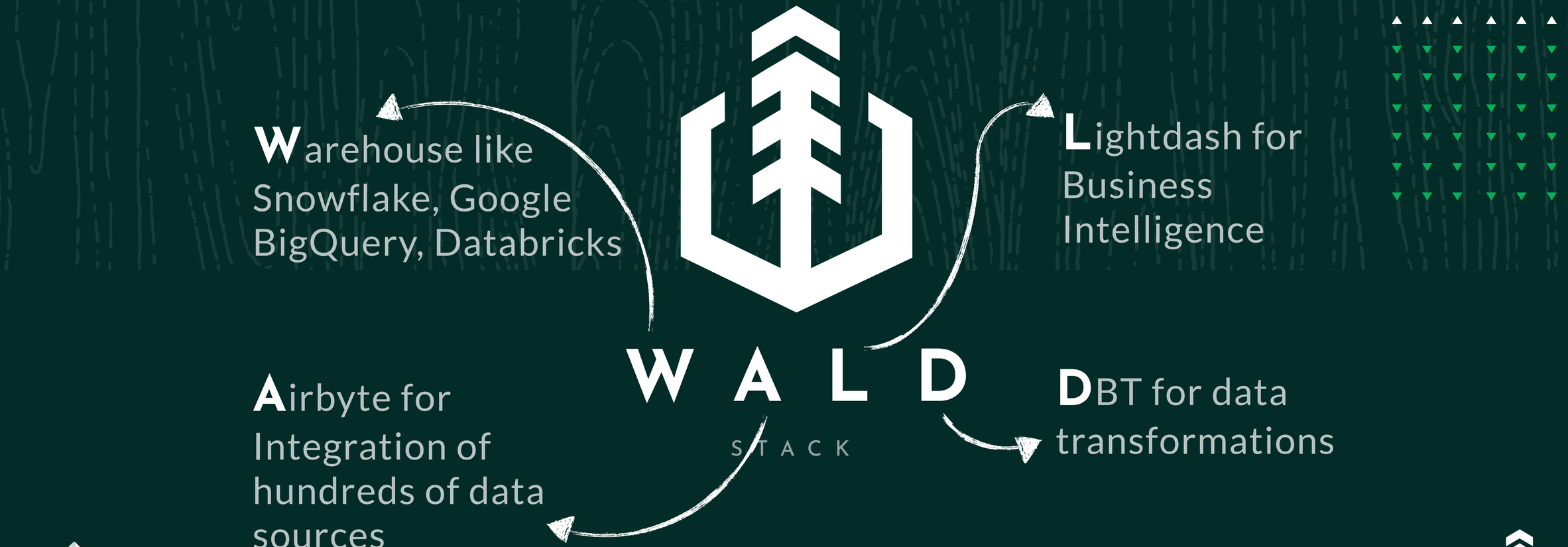
WALD

STACK

inovex

WALD STACK

A Modern & Sustainable Analytics Stack



inovex

WALD STACK

WALD based on Google Big Query

- fully managed enterprise data warehouse
- only available on Google Cloud Platform
- uses dbt-bigquery to run Python UDFs with Dataproc using PySpark

inovex

WALD STACK

WALD based on Databricks

- uses Databrick’s SQL warehouse
- available on all major cloud providers
- uses dbt-databricks to run Python UDFs with PySpark

inovex

WALD STACK

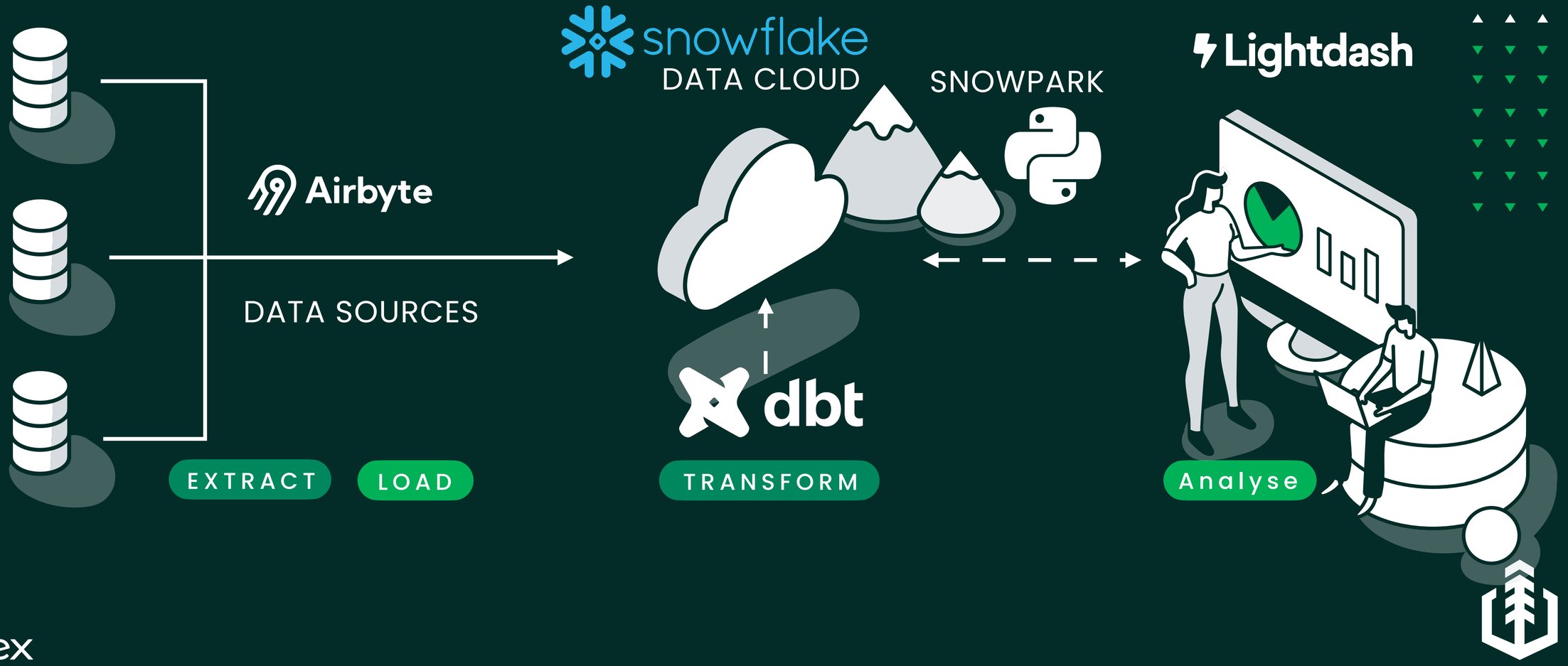
WALD based on Snowflake

- comes with all the benefits of Snowflake, e.g. governance, easy collaboration, security, near zero maintenance
- runs on all major cloud providers
- tons of great documentation
- the easiest to use and full-featured option to get started with WALD

inovex

WALD STACK

Modern Analytics Stack



inovex

WALD STACK

Example Use-Case

find all code & details under

https://waldstack.org

inovex

FORMULA 1

Analysis and Position Prediction

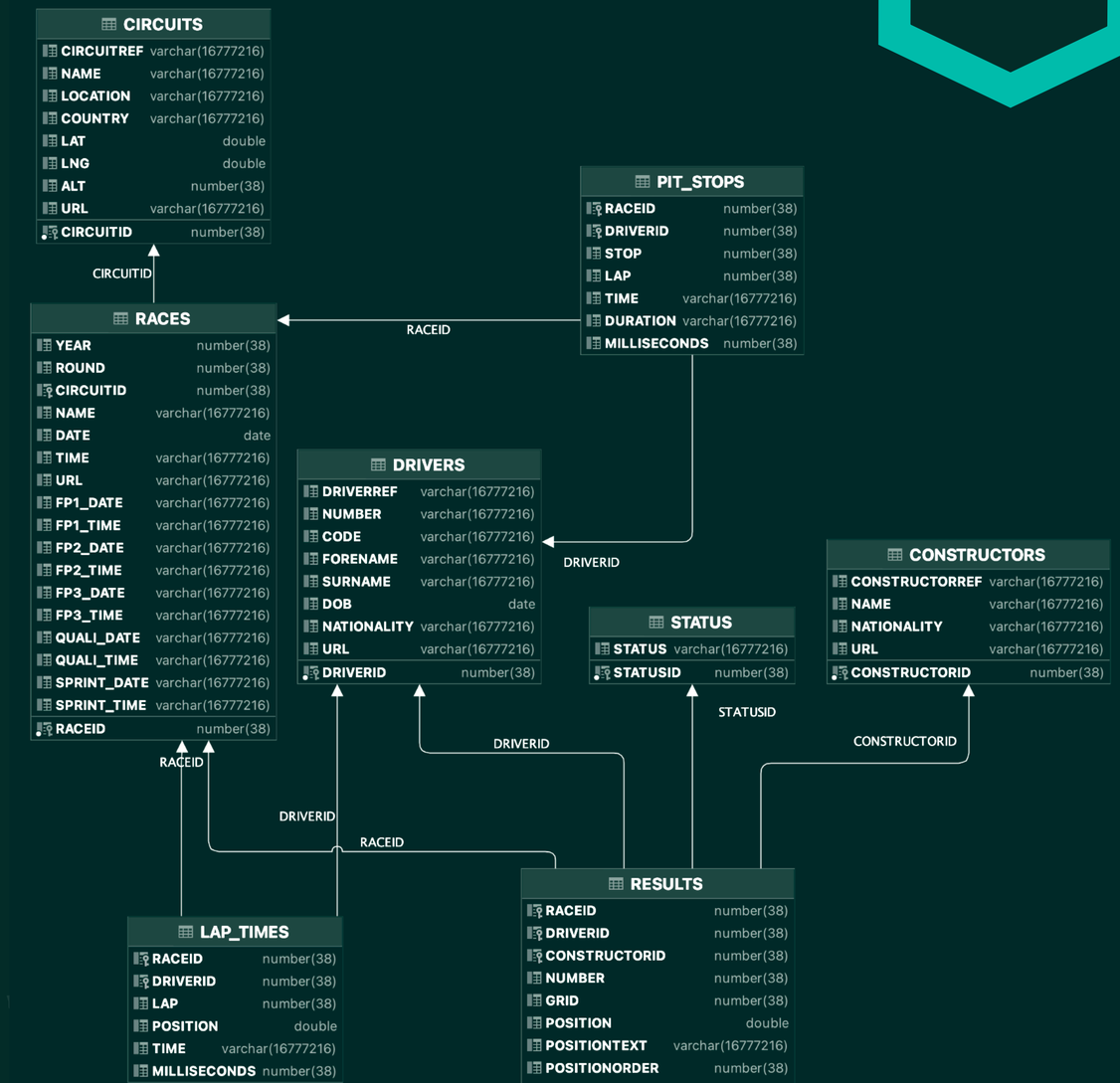
1. Upload data using Airbyte to Snowflake
2. Use dbt for some basic analysis
3. Use dbt with Snowpark for some ML to predict future position
4. Visualise our results with Lightdash

inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results



FORMULA 1

Rough Overview of the Data

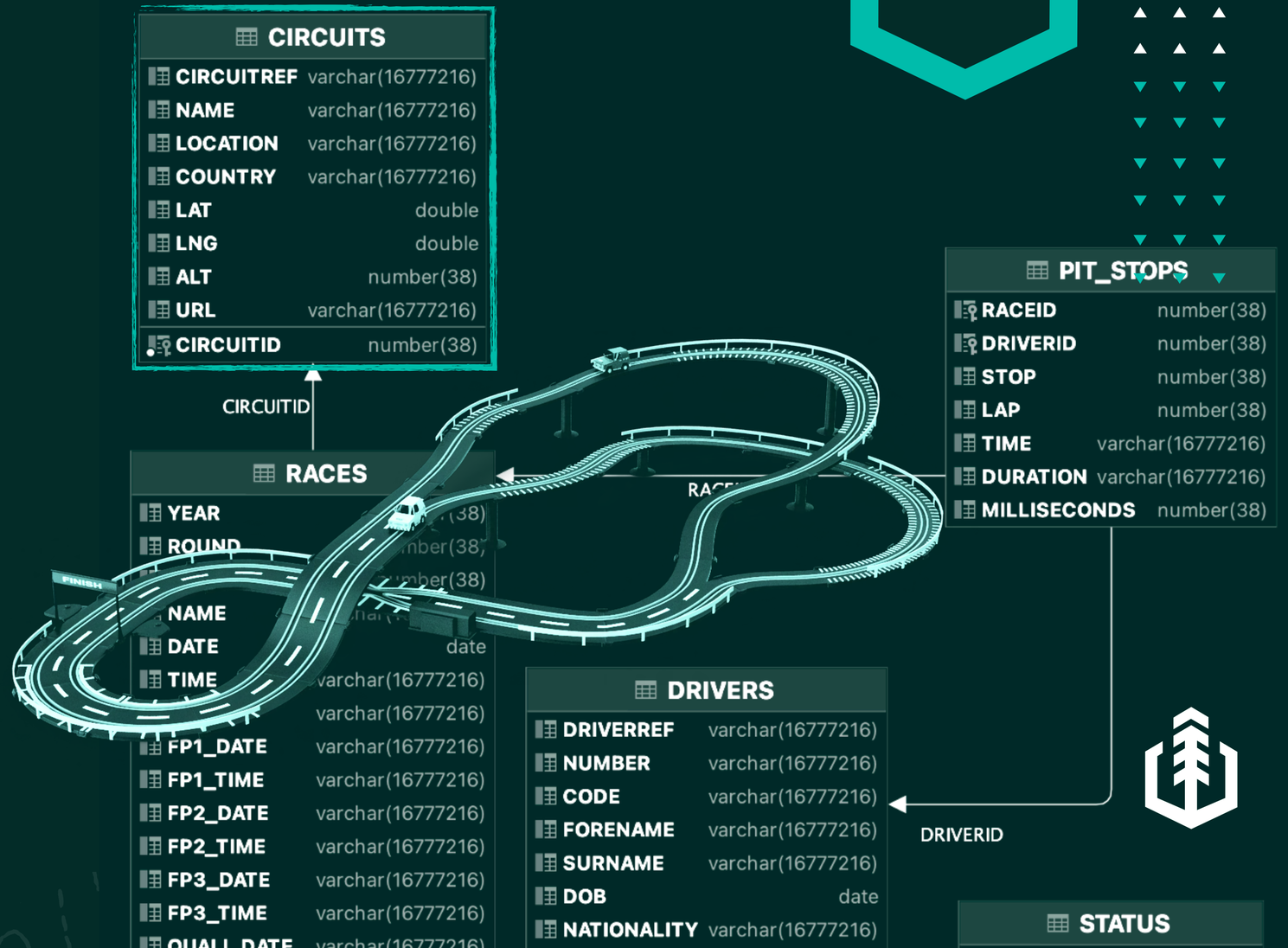
- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results

inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results

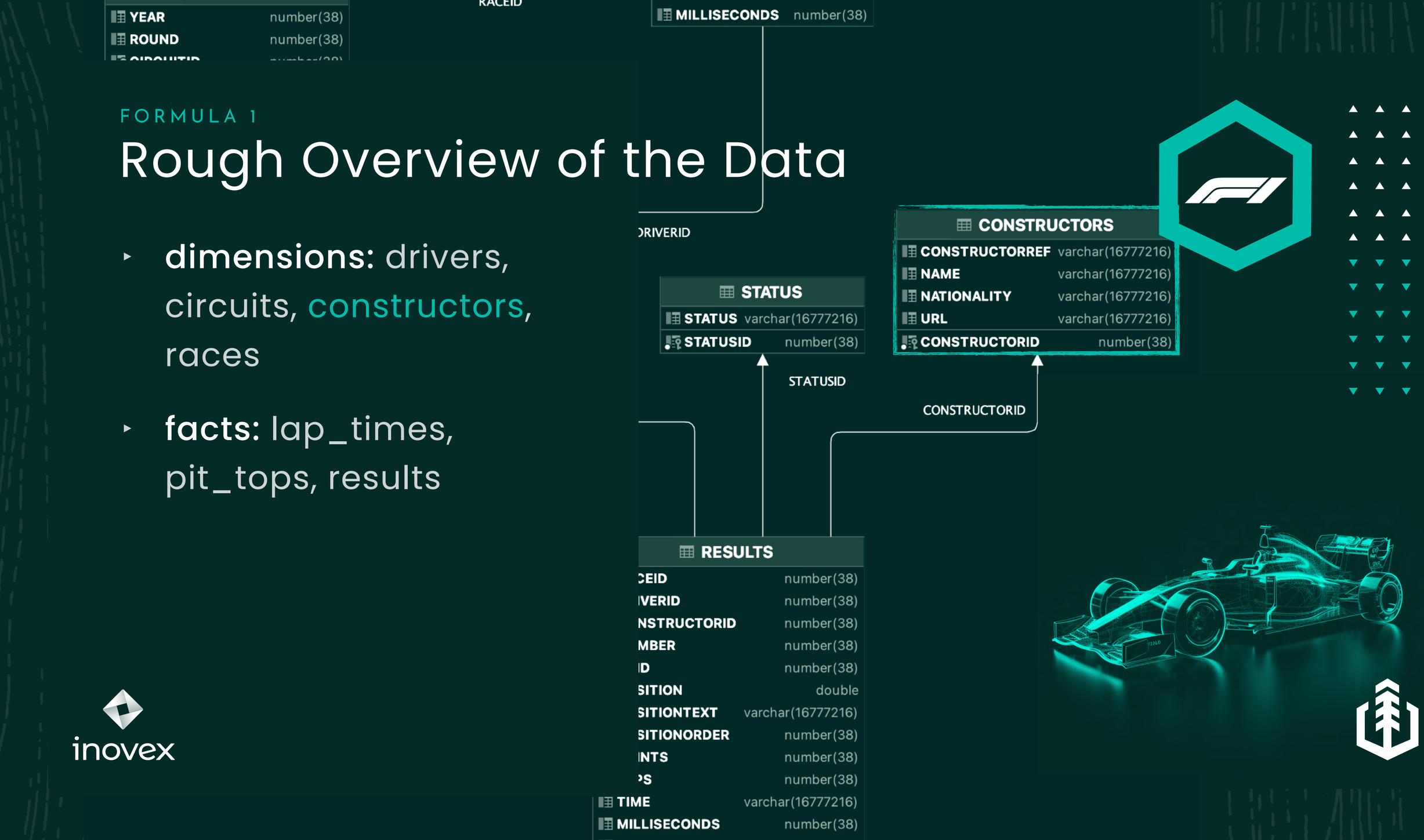


inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results

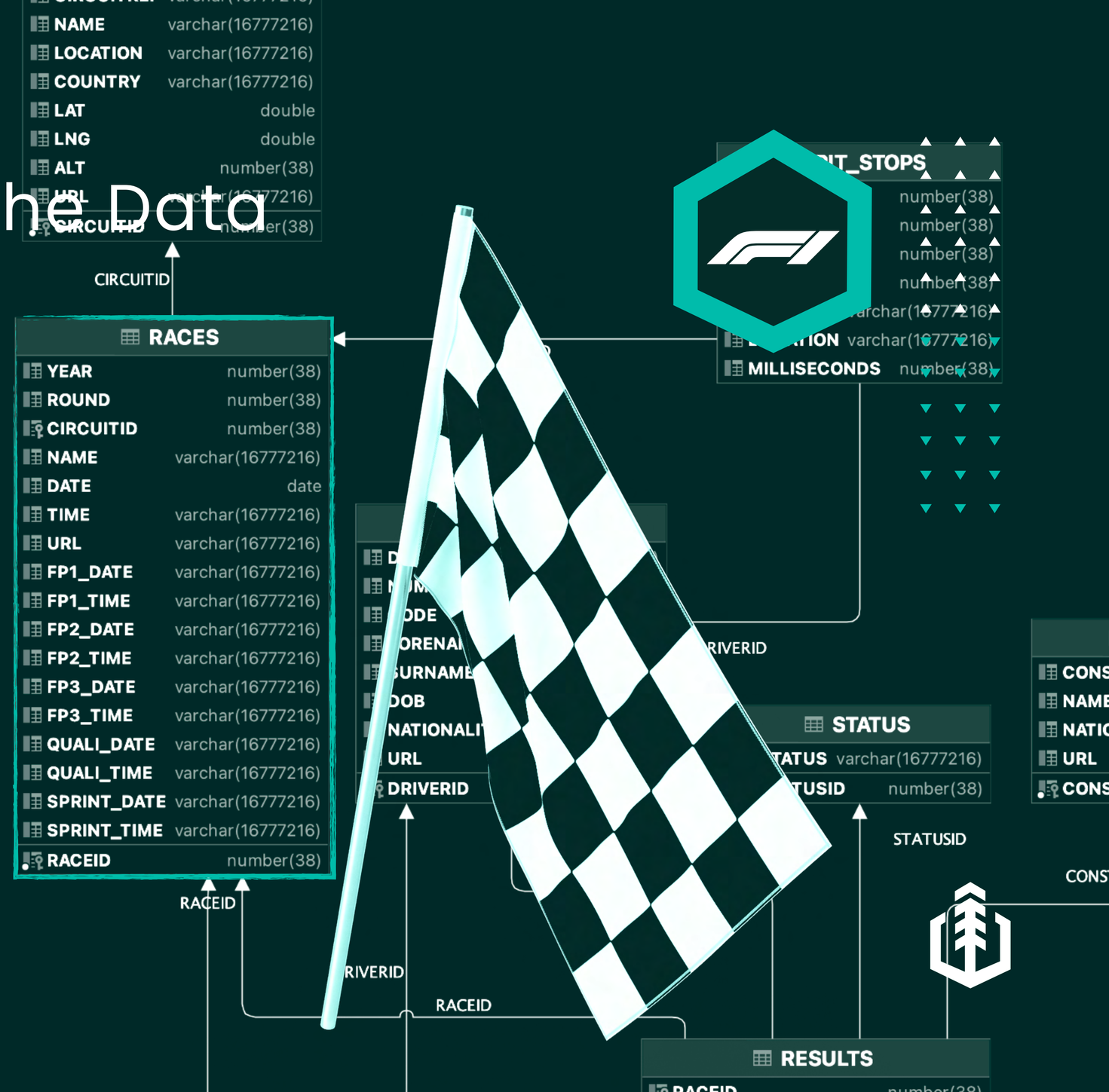


inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results

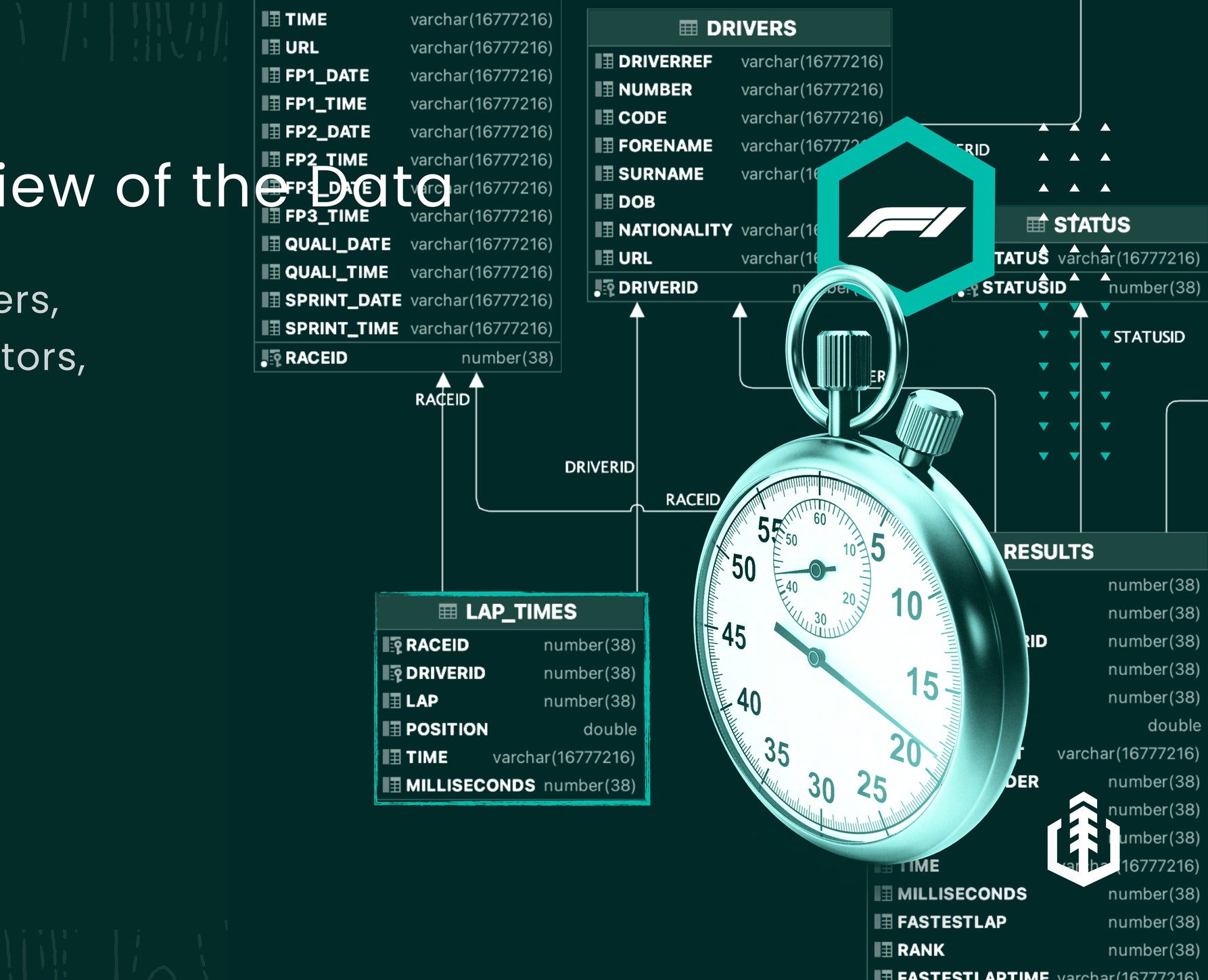


inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results

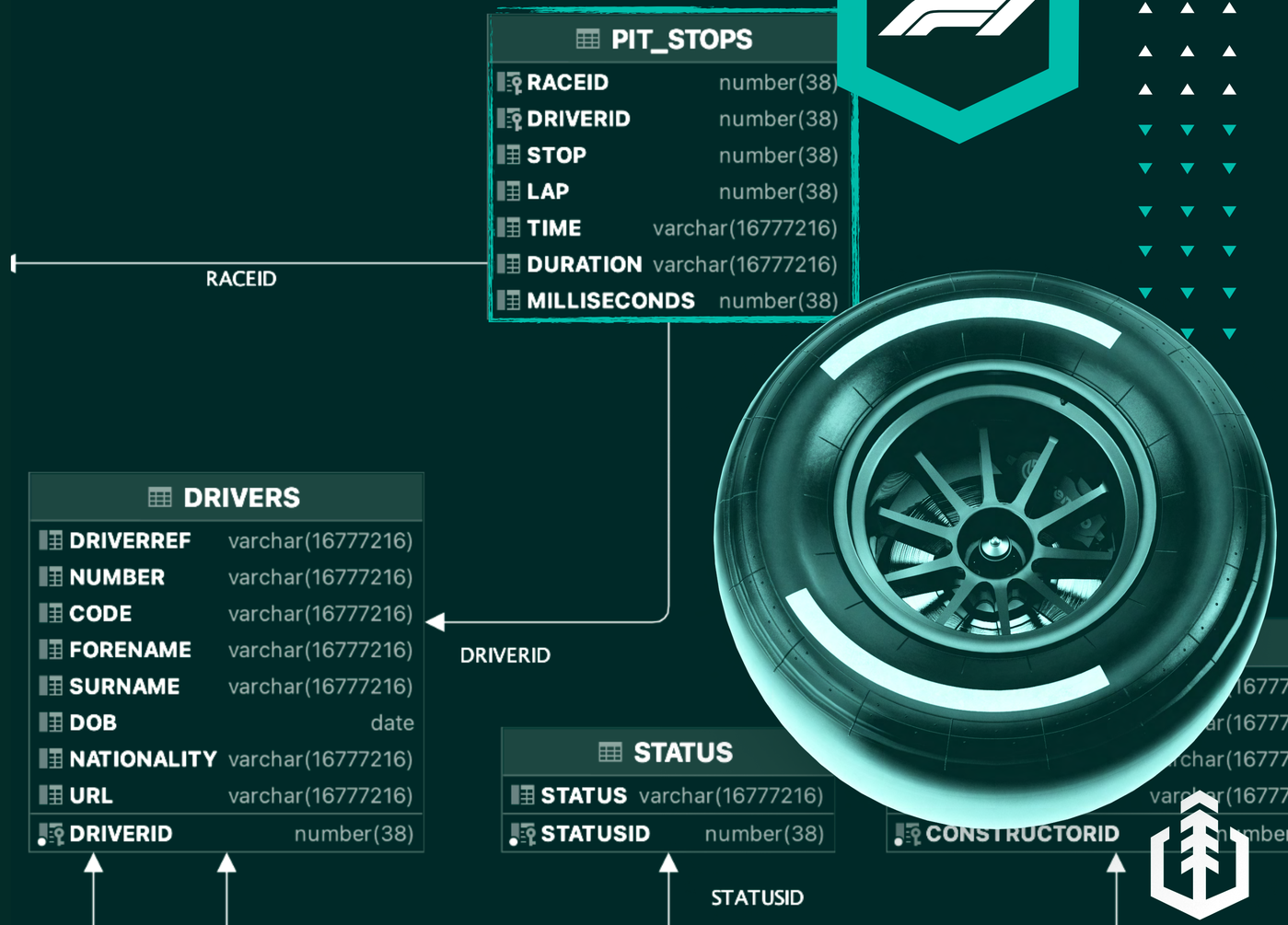


inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results

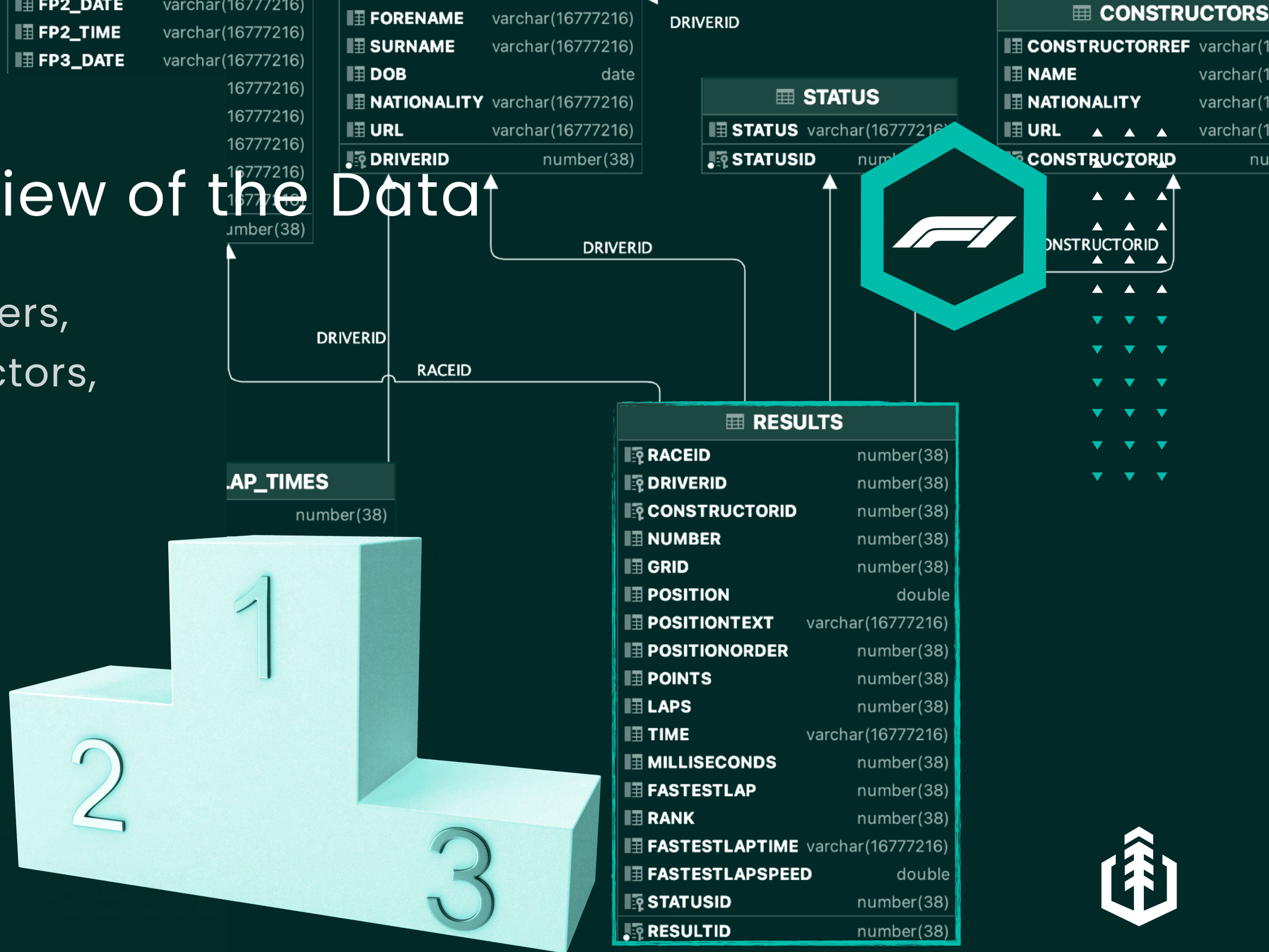


inovex

FORMULA 1

Rough Overview of the Data

- **dimensions:** drivers, circuits, constructors, races
- **facts:** lap_times, pit_tops, results



inovex

SNOWFLAKE

Snowflake Setup

- activate the Anaconda Packages (Preview Feature)
- create a warehouse to be used for dbt
- that’s it already!

inovex

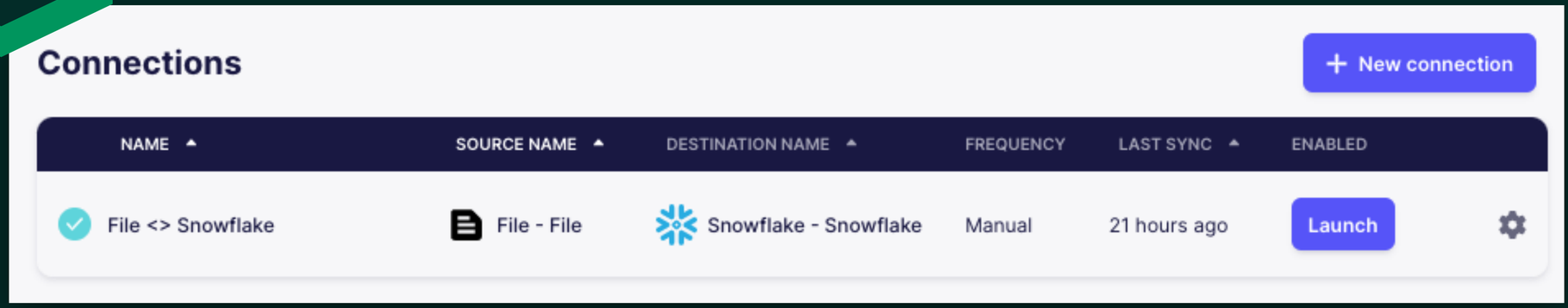
AIRBYTE

Ingesting the Data with Airbyte

Simply select

- file as source
- Snowflake as destination

and hit Launch



inovex

PYTHON + SNOWPARK

Setup Python/Snowpark for dbt

- just install along dbt also
  - **`dbt-snowflake`**
  - **`snowflake-snowpark-python`**
  - **`snowflake-connector-python`**
- and configure Snowflake in your profiles.yml

inovex

PYTHON + SNOWPARK

Training a Python Model in dbt/Snowpark

```python
def model(dbt, session):
    dbt.config(
        packages=["numpy", "scikit-learn", "pandas"],
        materialized="table",
        tags="train",
    )
    session.sql("create or replace stage MODELSTAGE").collect()
    test_train_df = dbt.ref("train_test_dataset")
    test_train_pd_df = test_train_df.to_pandas()
    X_train, X_test, y_train, y_test = train_test_split(
        test_train_pd_df.drop(["target"], axis=1), test_train_pd_df[target_col],
        train_size=0.7, random_state=42
    )
    model = LogisticRegression()
    model.fit(X_train, y_train)
    save_file(session, model, "@MODELSTAGE/", "model_v1.joblib")
```

inovex

DBT/SNOWPARK

Predicting with a trained Model in dbt/Snowpark

```python
def model(dbt, session):
    dbt.config(
        packages=["snowflake-snowpark-python", "scikit-learn", "pandas"],
        materialized="table",
        tags="predict",
    )
    model = load_model_from_stage(session)

    @udf
    def predict(p_df: T.PandasDataFrame[int, int]) -> T.PandasSeries[int]:
        pred_array = model.predict(p_df)
        return pd.Series(pred_array)

    snow_df = dbt.ref("hold_out_dataset_for_prediction").select(*FEATURE_COLS)

    return snow_df.withColumn("predicted", predict(*FEATURE_COLS))
```

inovex

LIGHTDASH

Visualisation with Lightdash

Lightdash pulls dimensions and metrics from dbt.

What are the avg lap times split by year and grand prix?

METRIC DIMENSIONS

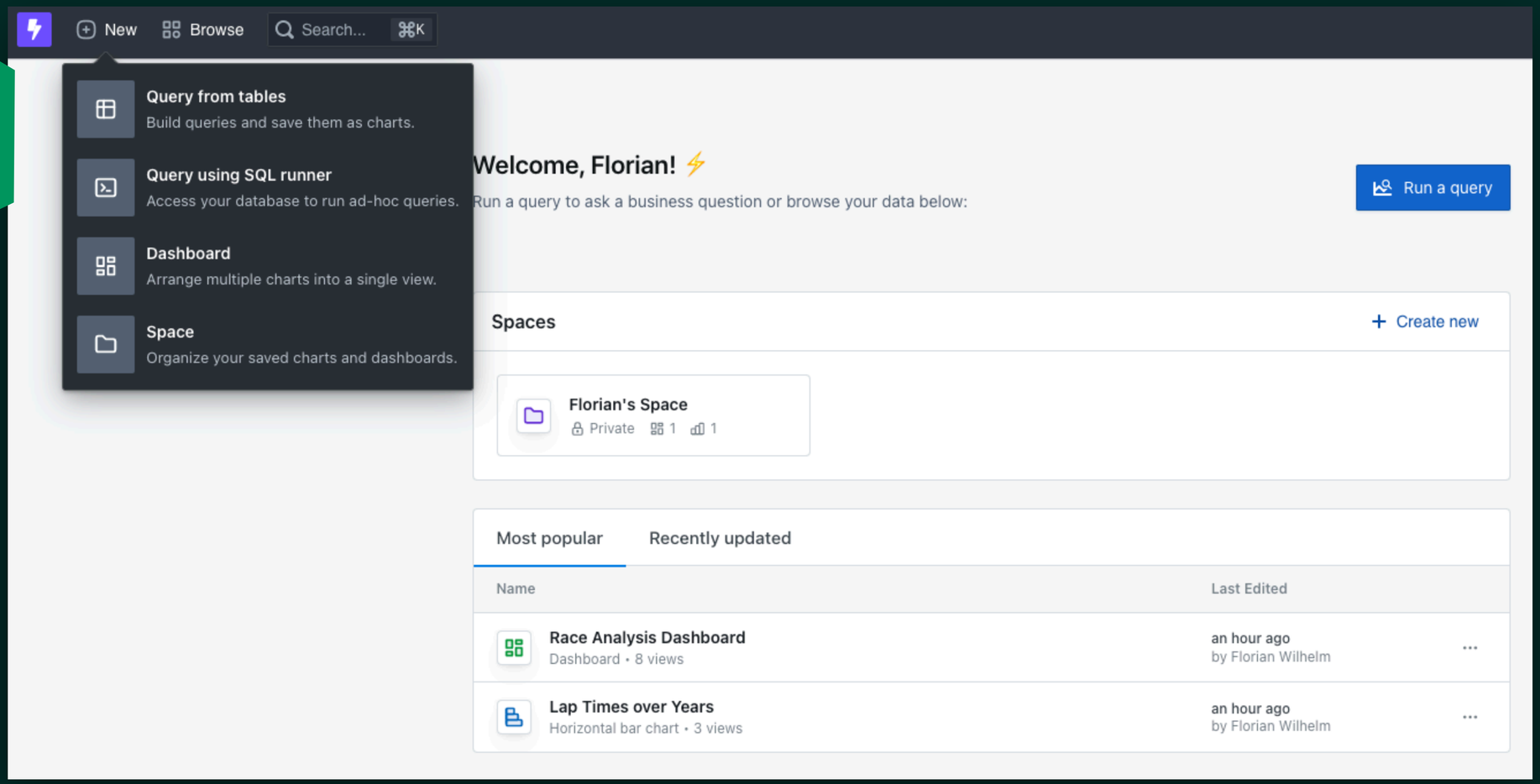
Lightdash

- runs ad-hoc queries
- creates charts from queries
- creates dashboards from charts
- organises everything in spaces

inovex

LIGHTDASH

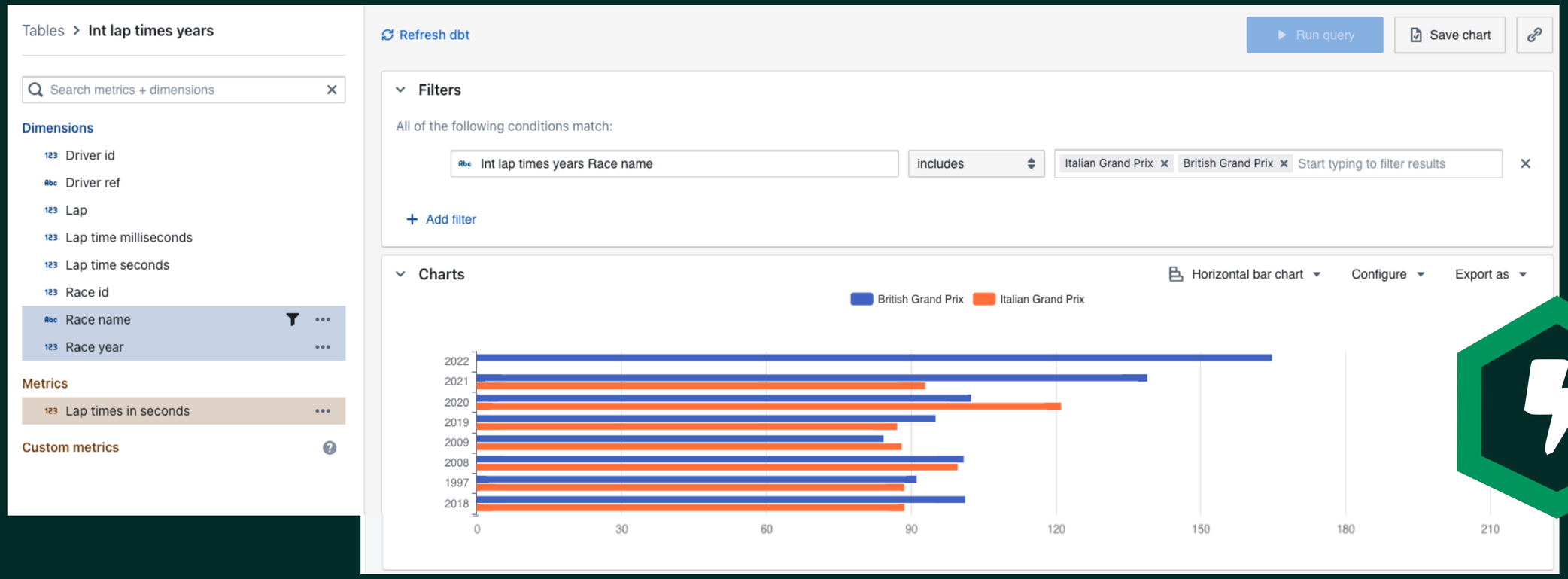
Main View of Lightdash



inovex

LIGHTDASH

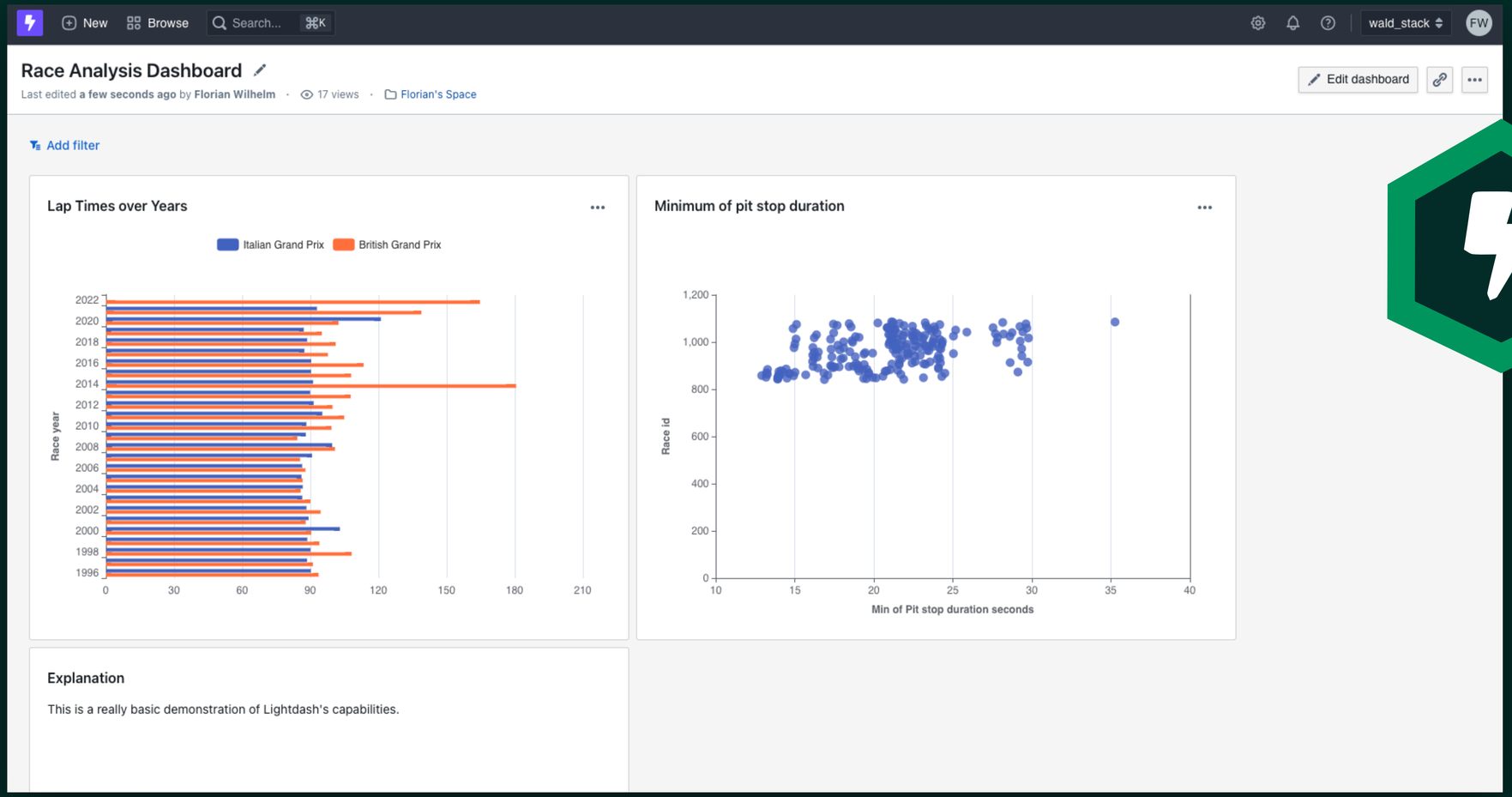
Analysing the Average Lap Times



inovex

LIGHTDASH

Example of a Simple Dashboard



inovex

FORMULA 1

Analysis and Position Prediction

inovex

One more thing...

inovex

ON MORE THING ON...

Streamlit

- provides faster way to build and share data apps
- was recently acquired by Snowflake
- might be a nice addition to Lightdash for interactive data apps

snowflake

Streamlit

inovex

WALD STACK

Conclusion

The WALD stack combines 4 powerful technologies that work very well together.

The WALD stack is flexible enough for many use-cases from analytics to building a full-fledged end-to-end data product at scale.

inovex

Dr. Florian Wilhelm, 2023

CHEERS TO THE COMMUNITY

Credits & Resources

- python-snowpark-formula Github repo by Hope Watson from dbt labs
- kind support by Michael Gorkow & Marko Cavar from Snowflake
- these awesome slides by Michael Hofmann from inovex

inovex

PyConDE / PyData 2023

Thank you!

Dr. Florian Wilhelm
Head of Data Science

florian.wilhelm@inovex.de
inovex.de
@inovexlife
@inovexgmbh

waldstack.org

inovex