Streaming Data Engineering

Feature Extraction from Road Traffic Data

Hochschule München University of Applied Sciences

Dr. Cristian Axenie

Outline

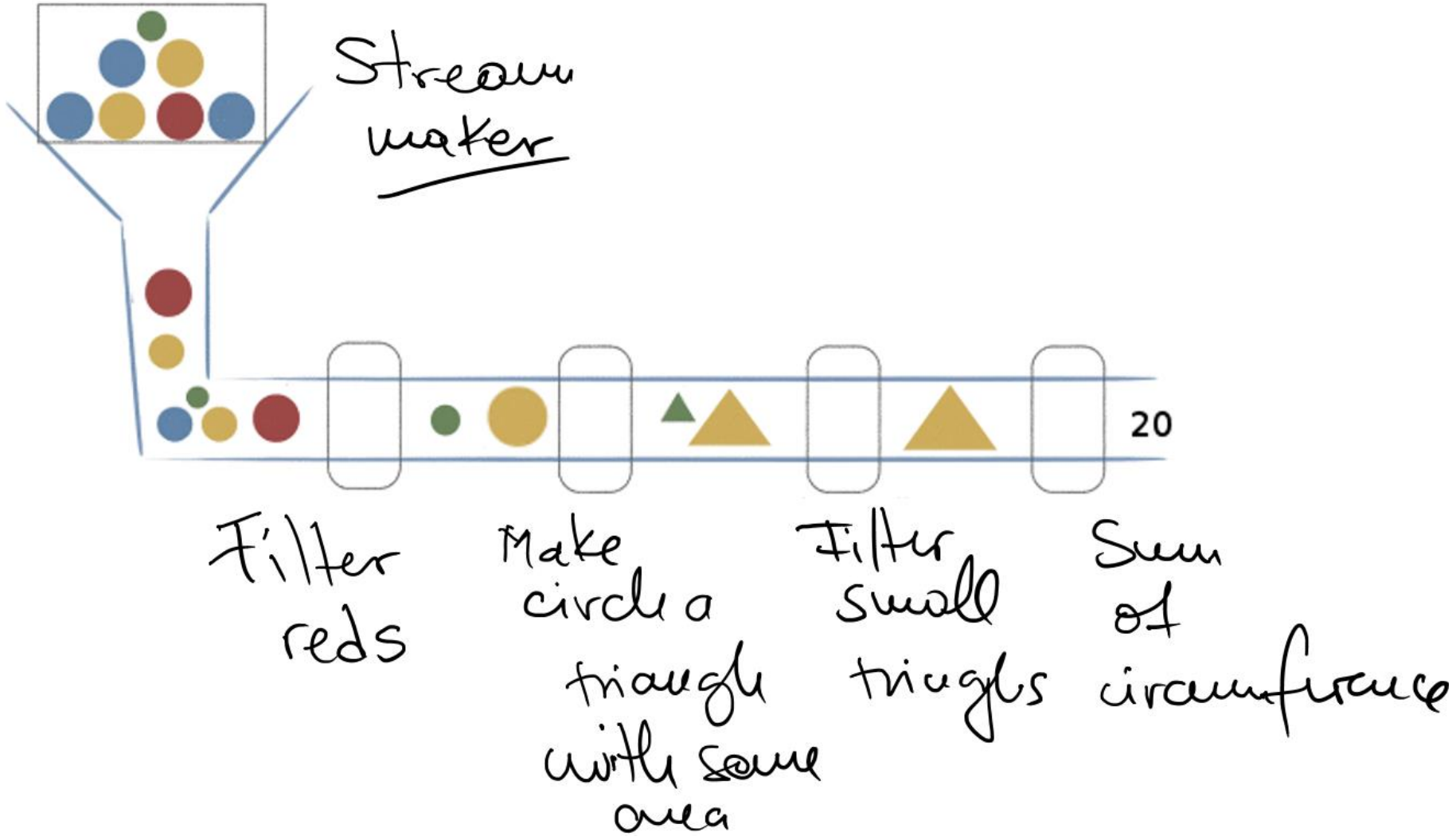
- Streaming Data Engineering
  - *Concepts*
  - *Functions*

- A concrete example: Road Traffic Monitoring and Analysis
  - *Applied concepts*
  - *Technology*

- Conclusions from practice

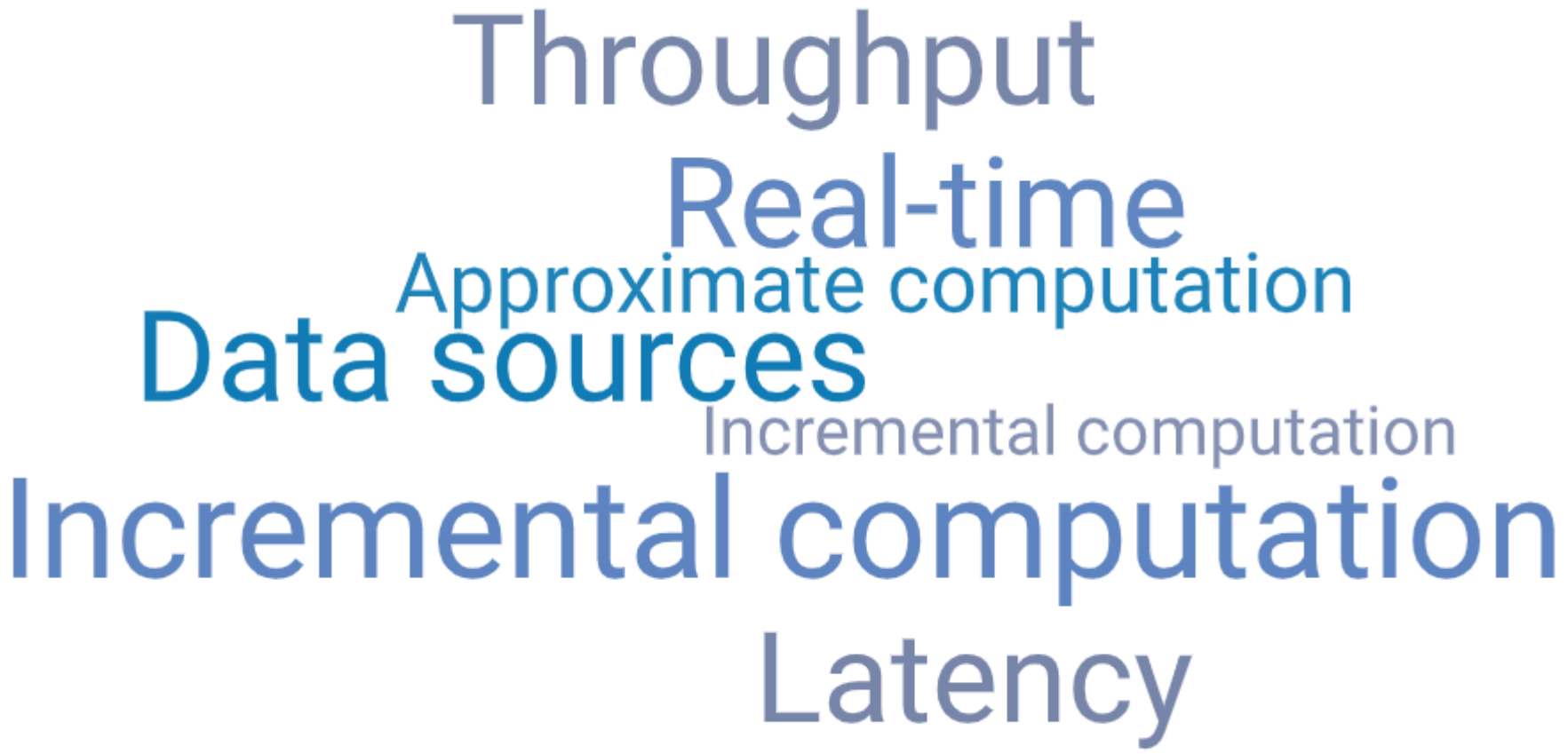
Streaming Data Engineering

An intuitive introduction



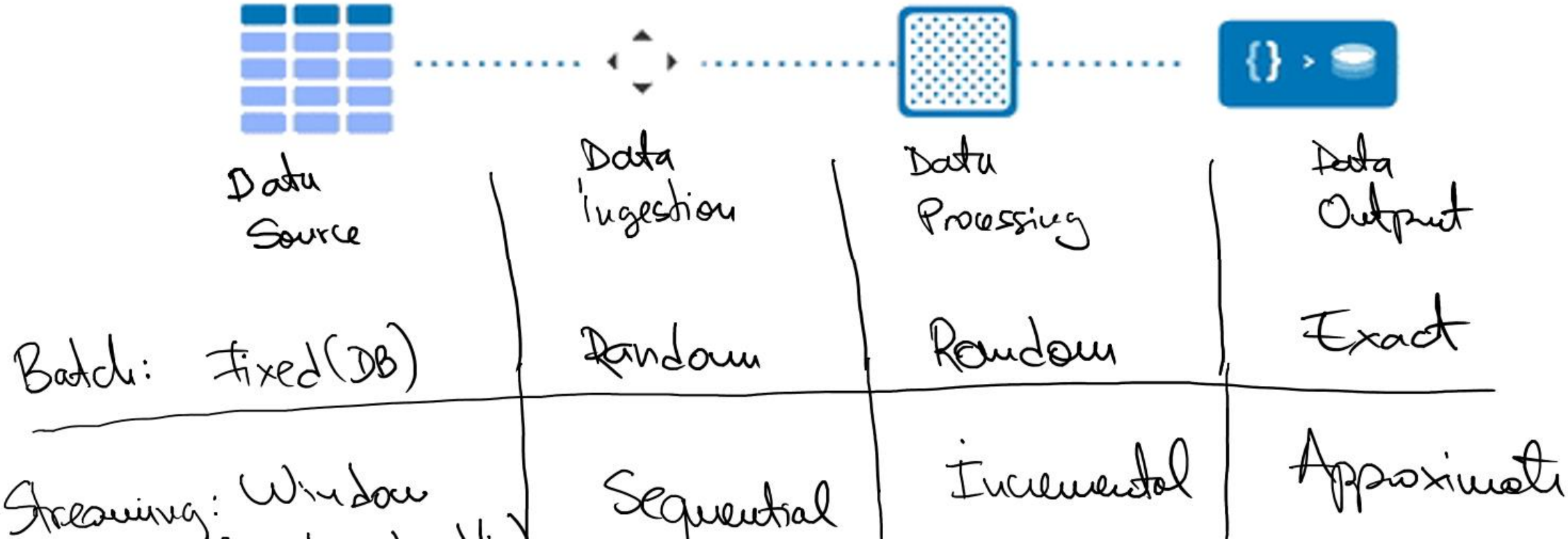
Streaming Data Engineering

Concepts



Streaming Data Engineering

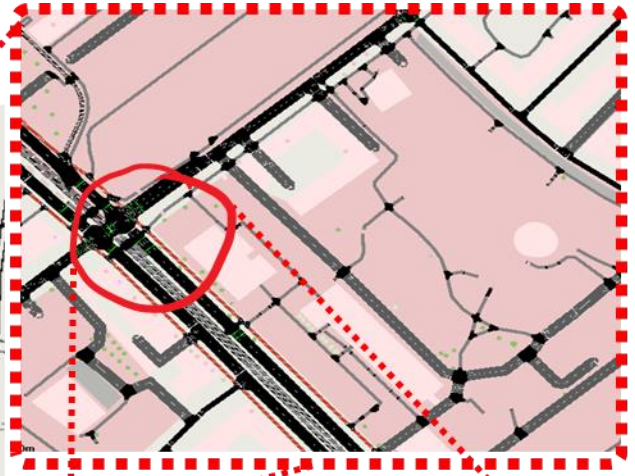
Functions



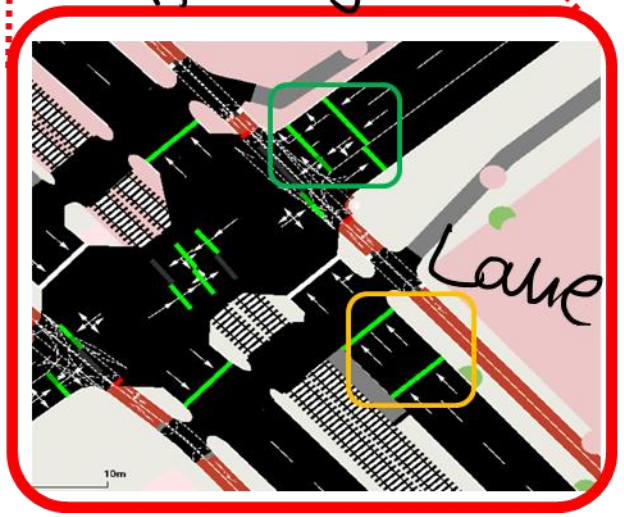
Road Traffic Monitoring and Analysis

Road Traffic Monitoring and Analysis

Demo

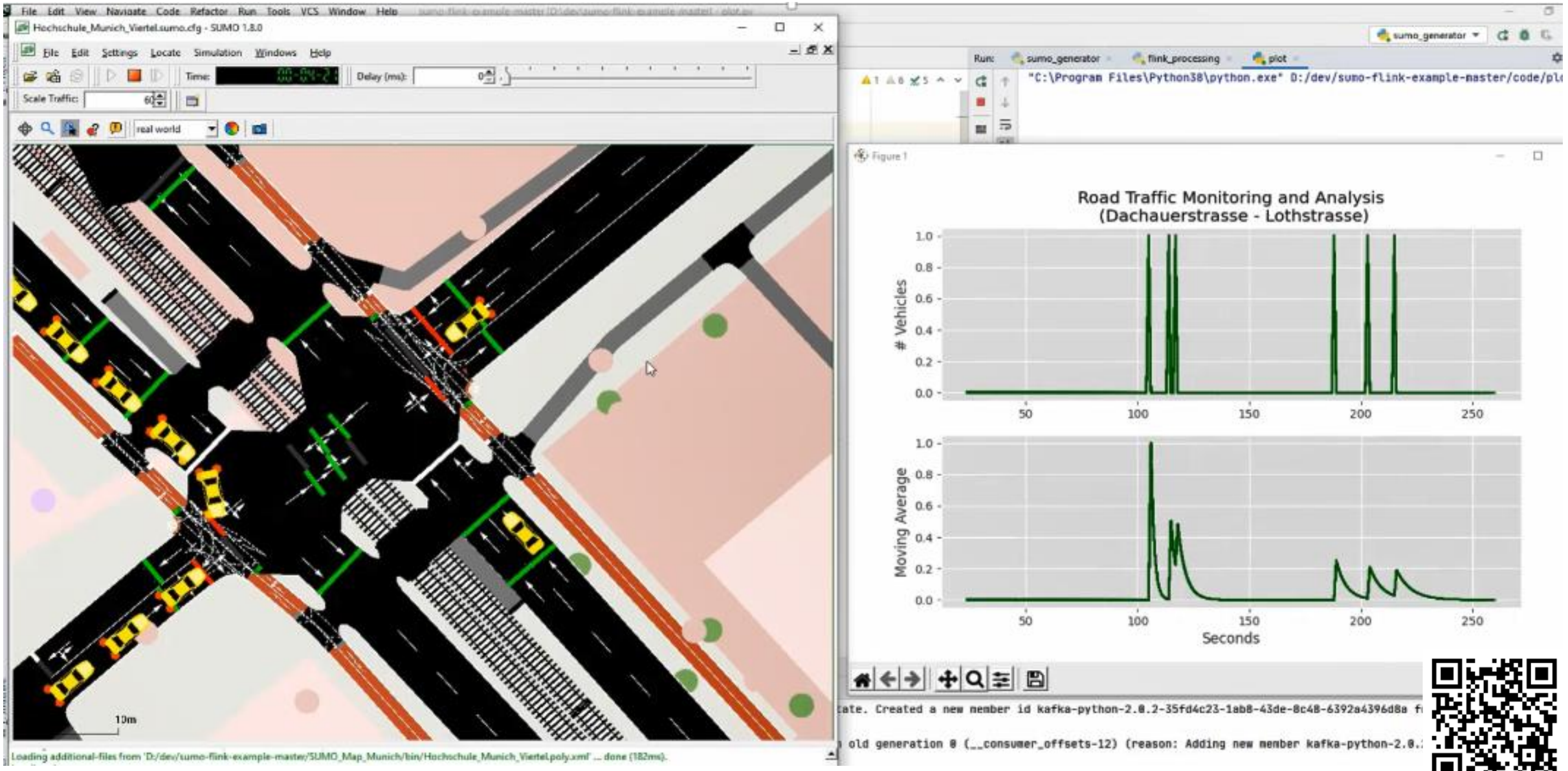


Traffic Lights



induction loops

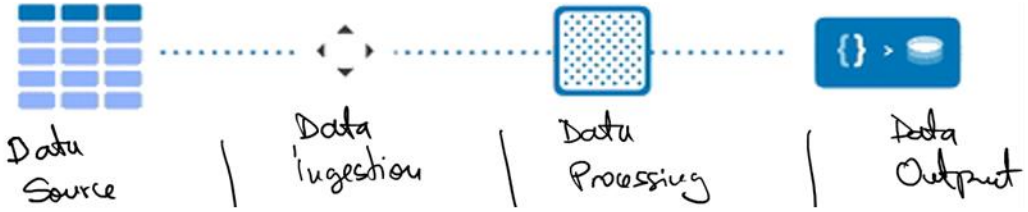
Road Traffic Monitoring and Analysis



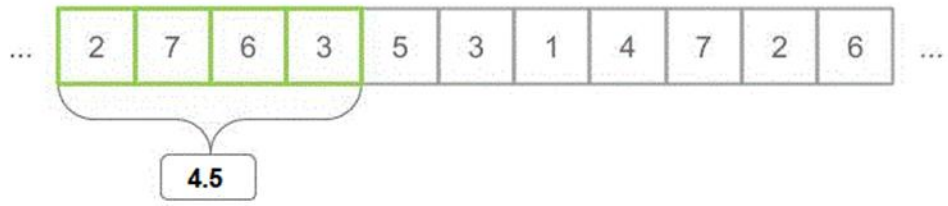
Road Traffic Monitoring and Analysis

Applied concepts

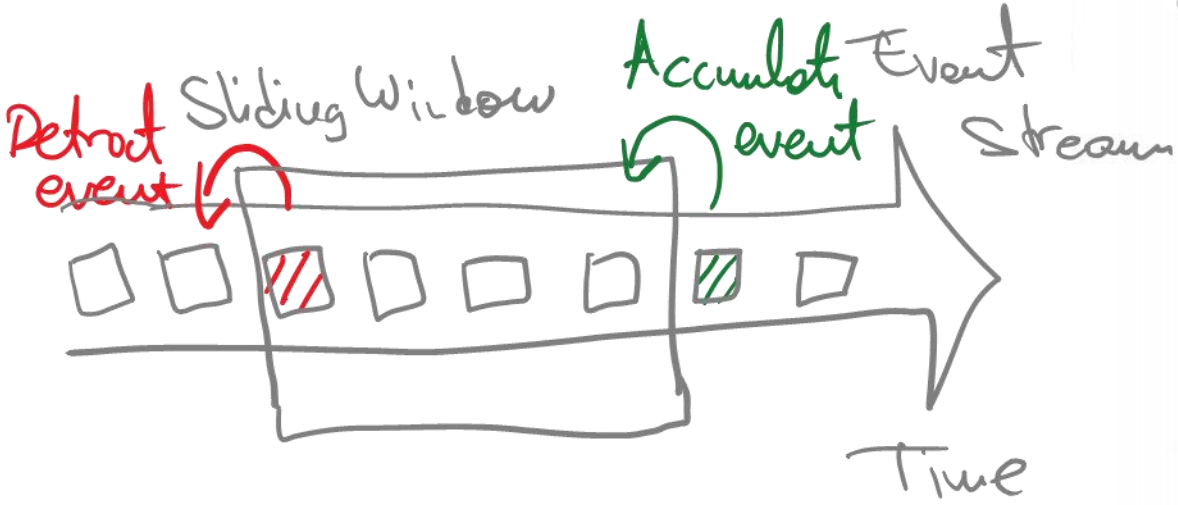
Sliding window



Processing



Principle



Consider a real valued sample $x_1, \dots, x_t, \dots x_{t//<}$ for all $i \in \{1, \dots t, \dots\}$ drawn from the distribution of the random variable X.

The sample mean of the sample of size t is $\bar{x}_t$

$$\bar{x}_t = \frac{1}{t} \cdot \sum_{i=1}^{t} x_i \quad \text{for incremental version}$$

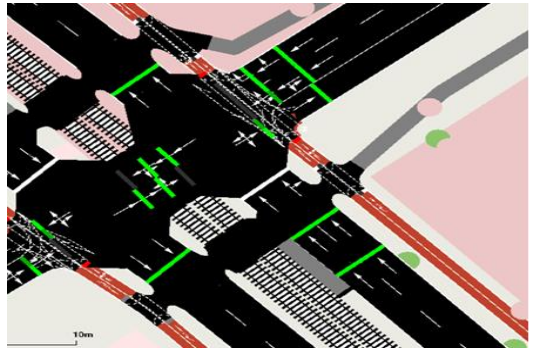
$$\bar{x}_t = \frac{1}{t}\left(x_t + \sum_{i=1}^{t-1} x_i\right) = \frac{1}{t}\left(x_t + (t-1)\bar{x}_{t-1}\right)$$

$$\boxed{\bar{x}_t = \bar{x}_{t-1} + \frac{1}{t}\left(x_t - \bar{x}_{t-1}\right)}$$

① mean

Road Traffic Monitoring and Analysis

Technology



Data Acquisition Component

Real-time Data analysis Component

Data-Cache Visualization

kafka

Flink

matplotlib

Data generation

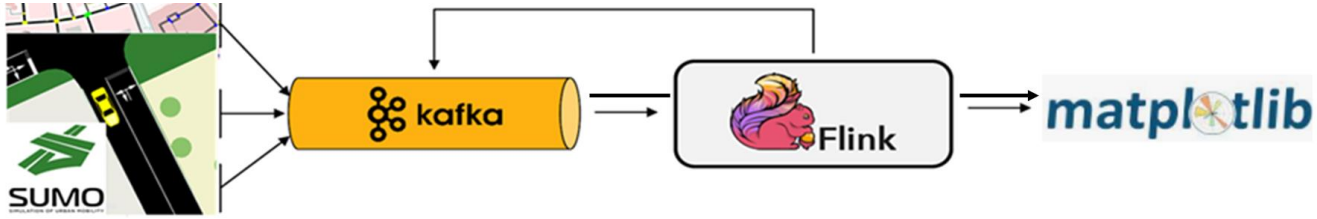
Data acquisition

Data analysis

Data visualization

Road Traffic Monitoring and Analysis

Technology




```
from pyflink.datastream import StreamExecutionEnvironment
from pyflink.table import StreamTableEnvironment, EnvironmentSettings

# Setup the Flink s
def flink_processing():
    env = StreamExecutionEnvironment.get_execution_environment()
    env.set_parallelism(1)
    env_settings = EnvironmentSettings.Builder().use_blink_planner().build()
    t_env = StreamTableEnvironment.create(stream_execution_environment=env,
                                          environment_settings=env_settings)

    t_env.get_config().get_configuration().set_string(
        "pipeline.jars",
        "file:///D:/temp/kafka_2.12-2.7.0/flink-connector-kafka_2.11-1.12.0.jar;"
        "file:///D:/temp/kafka_2.12-2.7.0/flink-sql-connector-kafka_2.11-1.12.0.jar"
    )
    source_ddl = """
                CREATE TABLE source_num(
                  `ts` TIMESTAMP(3) METADATA FROM 'timestamp',
                  `step` FLOAT,
                  `edge_id` STRING,
                  `vehicle_num` INT
                ) WITH (
                  'connector' = 'kafka',
                  'topic' = 'source_num',
                  'properties.bootstrap.servers' = 'localhost:9092',
                  'properties.group.id' = 'new_group2',
                  'format' = 'json'
                )
                """

    sink_ddl = """
                CREATE TABLE sink_table_num(
                    `ts` TIMESTAMP(3) METADATA FROM 'timestamp',
```

```
"C:\Program Files\Python38\python.exe" D:/dev/sumo-flink-example-master/code/plot
```

```
[2021-02-02 13:40:22,334] INFO [GroupCoordinator 0]: Dynamic Member with unknown member id joins group my-group in Empty state. Created a new member id kafka-python-2.0.2-35fd4c23-1ab8-43de-8c48-6392a4396d8a for this member and add
to the group. (kafka.coordinator.group.GroupCoordinator)
```

Code available on Github

Conlcusions from practice

Offenbach

Shenzen

In a **real application**
(China, Shenzen, Bantian Neighborhood,
8 intersections, 28 traffic light controlled street lanes):

- **Data collected by cameras and induction loops** every second (**280 Kafka messages/s**).
- **Processing steps** (SELECT - FROM - WHERE – ML) **sequentially**.
- **Data is processed immediately** and **incrementally** (e.g. counting the number of cars passing, queue length, flow, congestion index).
- Processing with **low-latency** (**0.03s/traffic light**) and **high throughput** (**28 traffic light control signals/s**).

Streaming Data Engineering

Feature Extraction from Road Traffic Data