PROCESSAMENTO DE STREAMS

COMPLEX EVENT PROCESSING

FACULDADE DE CIÊNCIAS E TECNOLOGIA DA UNIVERSIDADE NOVA DE LISBOA

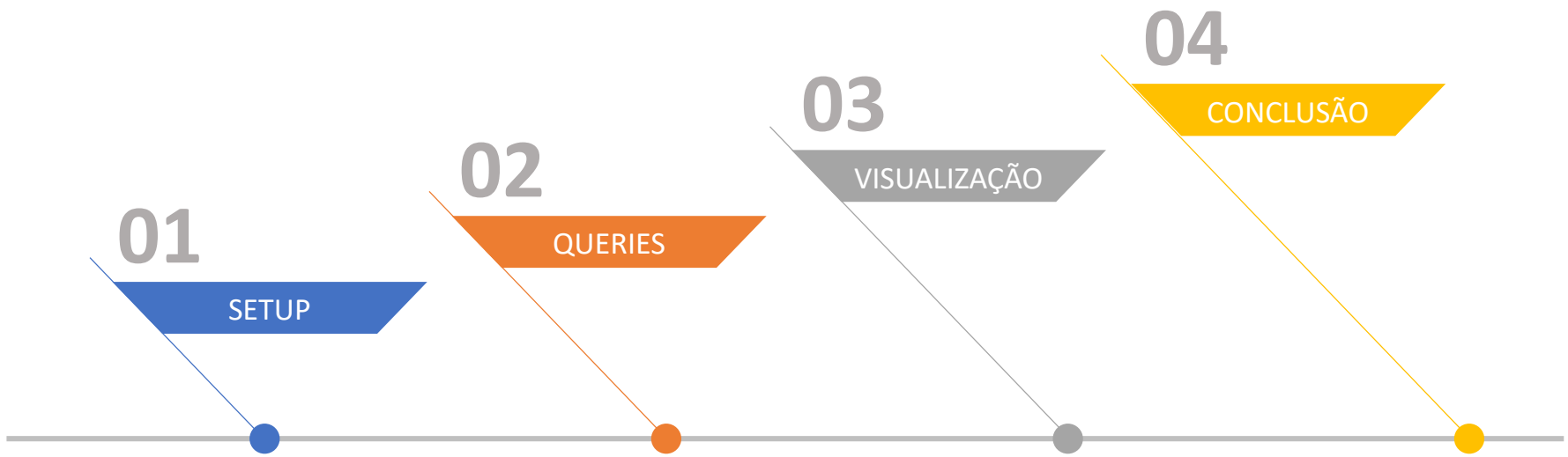
DEPARTAMENTO DE INFORMÁTICA

ANDRÉ LOPES – 45617 | NELSON COQUENIM - 45694

2017-2018

INTRODUÇÃO

INTRODUÇÃO



SETUP

Ilustração do setup aplicado.

QUERIES

Implementação das *queries* do *DEBS 2015 Challenge;*
Implementação de mais 3 queries adicionais descritas no enunciado do projecto.

VISUALIZAÇÃO

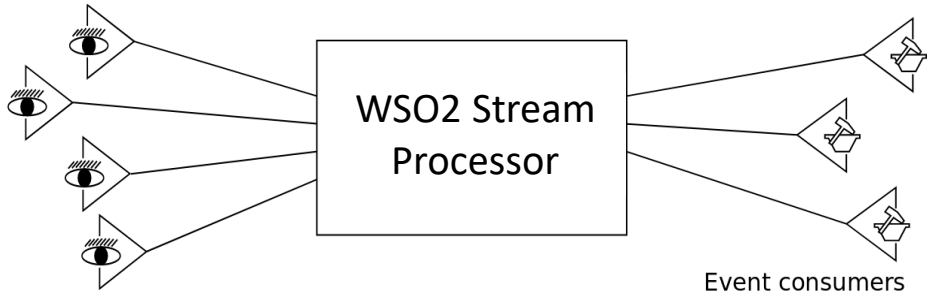
Visualização da *query Profitable Areas* através de um jupyter notebook com o uso da biblioteca gmaps para python.

CONCLUSÃO

Algumas considerações finais sobre as tecnologias utilizadas no decorrer do projecto.

SETUP

SETUP

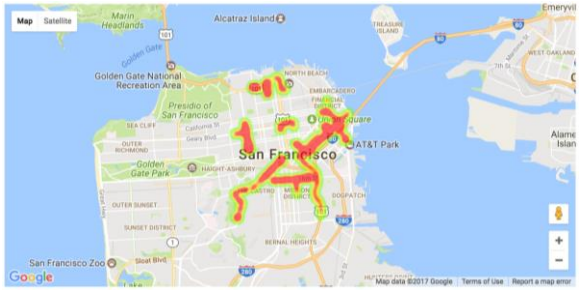


Event observers (sources)

Kafka

Event consumers (sinks)

Kafka



QUERIES

FREQUENT ROUTES

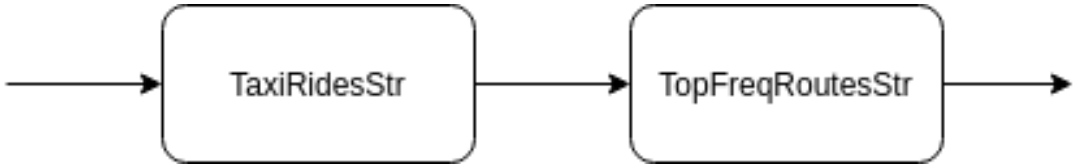


Fig.1. Diagrama de fluxo da *query Frequent Routes*.

FROM TaxiRidesStr#window.time(30 sec)

SELECT pickup_gridID, dropoff_gridID, count(*) as frequency

GROUP BY pickup_gridID, dropoff_gridID

ORDER BY frequency **DESC**

LIMIT 10

INSERT INTO TopFreqRoutesStr;

PROFITABLES AREAS

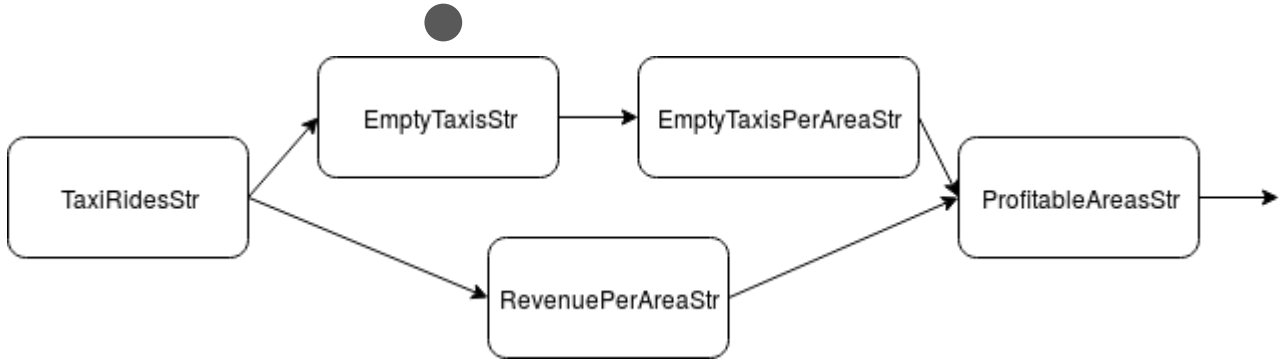


Fig.2. Diagrama de fluxo da *query Profitables Areas*.

PARTITION WITH (medallion of TaxiRidesStr)

BEGIN

FROM e1 = TaxiRidesStr -> **NOT** TaxiRidesStr[medallion == e1.medallion] for 30 sec

SELECT e1.medallion, e1.dropoff_gridID

INSERT INTO EmptyTaxisStr;

END;

PROFITABLES AREAS

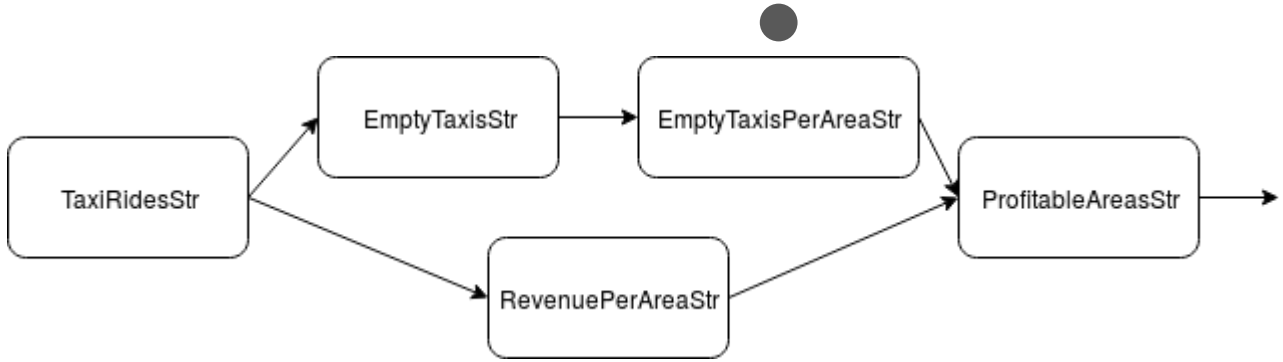


Fig.2. Diagrama de fluxo da *query Profitables Areas*.

FROM EmptyTaxisStr#window.time(30 sec)

SELECT dropoff_gridID as areaID, count(*) as emptyTaxis

GROUP BY dropoff_gridID

INSERT INTO EmptyTaxisPerAreaStr;

PROFITABLES AREAS

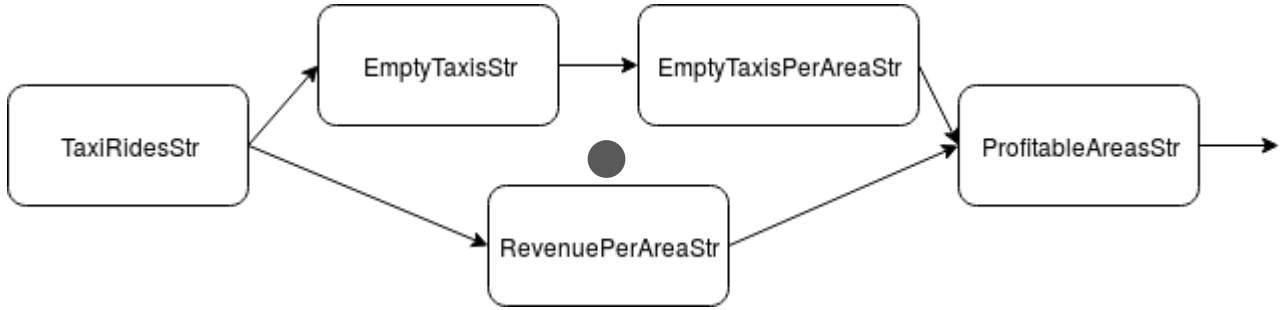


Fig.2. Diagrama de fluxo da *query Profitables Areas*.

FROM TaxiRidesStr#window.time(15 sec)

SELECT pickup_gridID as areaID, **AVG**(fare_amount + tip_amount) as revenue

GROUP BY pickup_gridID

INSERT INTO RevenuePerAreaStr;

PROFITABLES AREAS

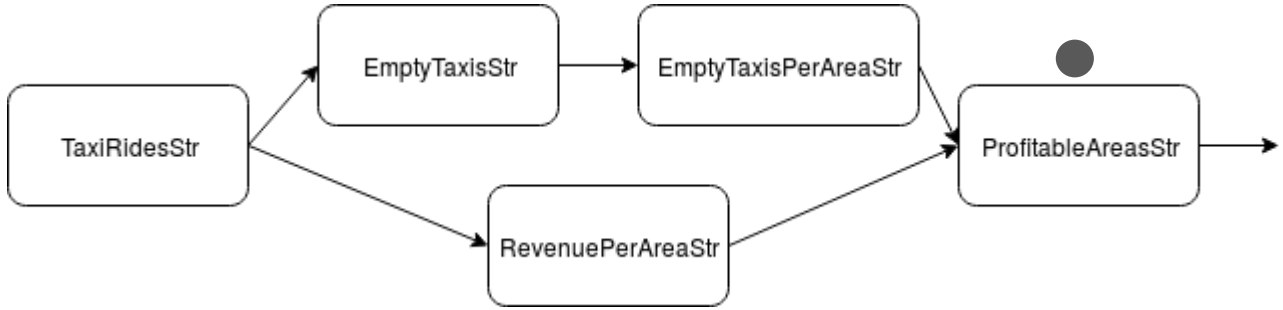


Fig.2. Diagrama de fluxo da *query Profitables Areas*.

FROM RevenuePerAreaStr#window.time(15s) as A **JOIN** EmptyTaxisPerAreaStr#window.time(15s) as B **ON** A.areaID == B.areaID

SELECT A.areaID, revenue/emptyTaxis as profit

GROUP BY A.areaIDorder by profit DESClimit 10

INSERT INTO ProfitableAreasStr;

IDLE TAXIS

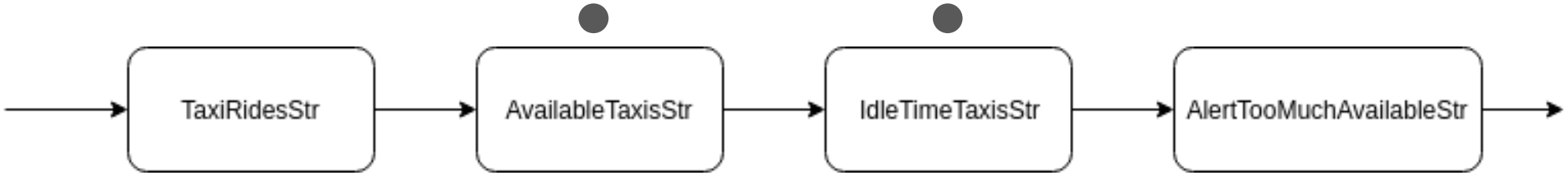


Fig.3. Diagrama de fluxo da *query Idle Taxis*.

FROM TaxiRidesStr#window.time(1 hour)

SELECT * **INSERT INTO** AvailableTaxisStr;

FROM e1 = AvailableTaxisStr -> e2 = AvailableTaxisStr[medallion == e1.medallion]

SELECT e1.medallion as taxi, (e2.p_time-e1.d_time) as idle_time

INSERT INTO IdleTimeTaxisStr;

IDLE TAXIS

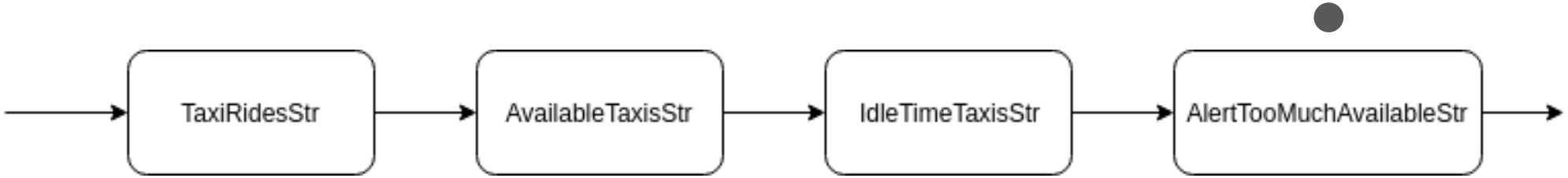


Fig.3. Diagrama de fluxo da *query Idle Taxis*.

FROM IdleTimeTaxisStr

SELECT avg(idle_time) as avg_idle_time

HAVING avg_idle_time > 10 * 60

INSERT INTO AlertTooMuchAvailableStr;

CONGESTED AREAS

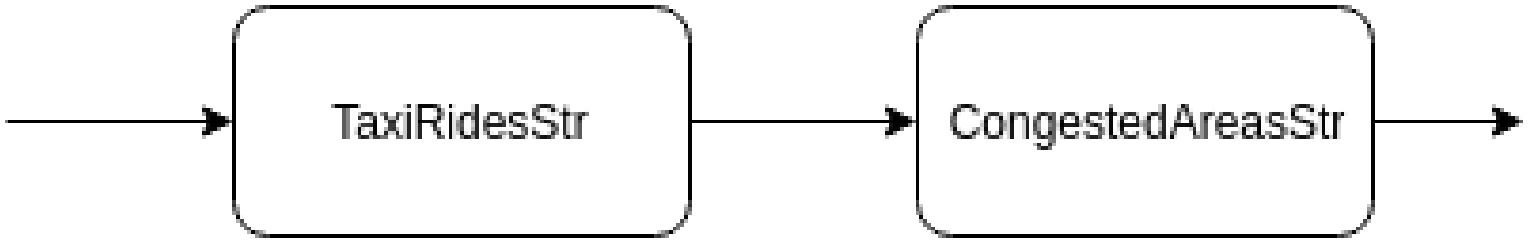


Fig.4. Diagrama de fluxo da *query Congested Areas*.

```
FROM EVERY e1 = TaxiRidesStr->
  e2=TaxiRidesStr [medallion == e1.medallion AND ride_duration > e1.ride_duration] ->
  e3=TaxiRidesStr [medallion == e2.medallion AND ride_duration < e2.ride_duration] ->
  e4=TaxiRidesStr [medallion == e3.medallion AND ride_duration > e3.ride_duration] ->
  e5=TaxiRidesStr [medallion == e4.medallion AND ride_duration > e4.ride_duration] ->
  e6=TaxiRidesStr [medallion == e5.medallion AND ride_duration > e5.ride_duration]
SELECT e2.pickup_gridID as areaID, e2.ride_duration as peak_duration
INSERT INTO CongestedAreasStr;
```

MOST PLEASANT TAXI DRIVERS

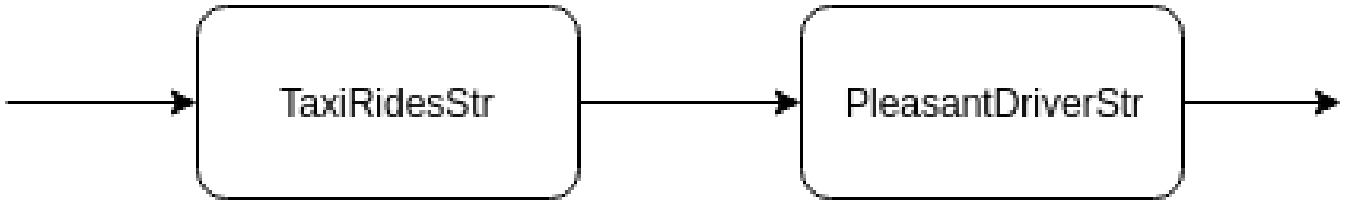


Fig.5. Diagrama de fluxo da *query Most Pleasant Taxi Drivers*.

FROM TaxiSecStr#window.timeBatch(24 min)

SELECT hack_license, sum(tip_amount) as tips_total

GROUP BY hack_license

ORDER BY tips_total **DESC**

LIMIT 1

INSERT INTO PleasantDriverStr;

CONCLUSÃO

CONCLUSÃO

SiddhiQL comparativamente ao SparkStreaming:

- + Manutenção devido a simplicidade do código;
- + Flexibilidade e Expressividade na descrição das *queries* devido à deteção de padrões de eventos de alto nível;
- - Qualidade da documentação disponível inferior, e.g. sources/sinks kafka;

- - *Editor* com alguns *bugs* na deteção de erros de sintaxe;
- - Erros de sintaxe pouco informativos;

OBRIGADO PELA ATENÇÃO

VISUALIZAÇÃO

DEMO