

38. Proudové (streamové) zpracování dat- Spark Streaming a Structured Streaming- popis, rozdíly, výhody a nevýhody.

- komponenta Sparku určená pro analýzu streamovaných dat
 - škálovatelná s vysokou propustností a odolností vůči chybám
 - checkpointy jsou pravidelně ukládány na disk
 - konstantní přísun dat z různých zdrojů
 - např. zpracování logů z webu nebo serveru
- data jsou agregována a analyzována v určitý daný interval
- data mohou pocházet z různých zdrojů
 - Amazon Kinesis, HDFS, Kafka, Flume, sockety
- opět existují dvě podobná API
 - původní založené na RDD's nazývané Spark Streaming
 - nové založené na DataFramech nazývané Structured Streaming



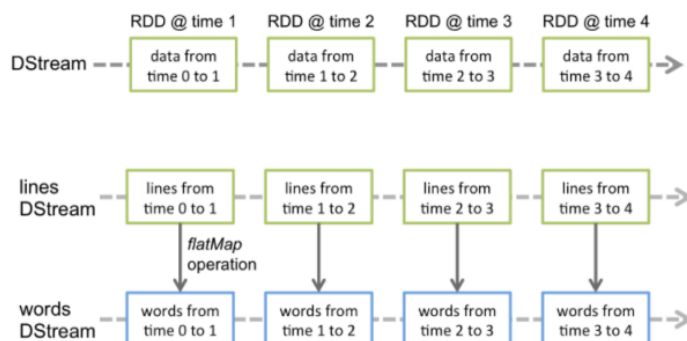
Spark streaming

jak funguje Spark Streaming?

- přijímá data ze streamů a dělí je na dávky
- dávky jsou zpracovány Spark engine
- výsledky jsou generovány ve formě streamu dávek



- nejedná se tedy o úplně real-time zpracování, ale micro-batching
- Spark poskytuje vysokoúrovňovou abstrakci DStream (discretized stream)
 - reprezentuje konzistentní tok dat
 - interně je reprezentován jako sekvence RDD's
- reálná ukázka: počet slov textu na vstupu
 - připojení k lokálnímu streamu dat přes TCP socket

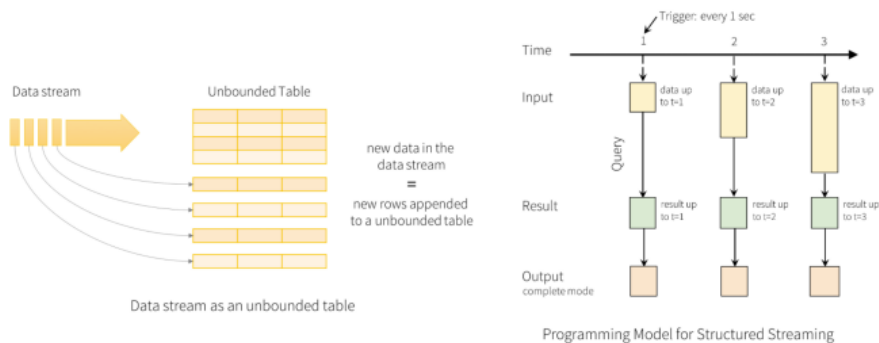


Spark structured

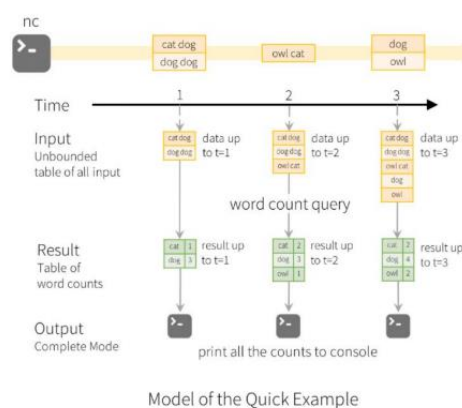
jak funguje Structured Streaming?

- streamovaná data jsou modelována jako DataFramy, které jsou rozšiřovány
 - umožňuje zpracování streamů stejným způsobem jako statických dat
- data jsou udržována ve vstupní tabulce
- nová data jsou jen připnuta do vstupní tabulky
- vstupní tabulka je zpracovávána, čímž je aktualizována výsledková tabulka
- po aktualizaci výsledkové tabulky dochází k zápisu ven
 - podpora různých módů
 - complete zapíše celou tabulku
 - append připiše jen nové řádky
 - update zapíše jen zaktualizované řádky
- možnost volby mezi micro-batch a continuous zpracováním
 - použití jen přepnutím módu s možností snížit latenci

- jak funguje Structured Streaming?



- reálná ukázka: počet slov textu na vstupu
 - připojení k lokálnímu streamu dat přes TCP socket



- podporované operace
 - většina funkcionality je podporována
 - selekce, projekce, agregace
 - windowed operace
 - join operace
 - se statickými i streamovanými DataFramy
- nepodporované operace
 - vícenásobné streamované agregace
 - limit na prvních N řádků
 - distinct
 - některé outer joiny
 - řazení **jen** po agregaci a v complete módu

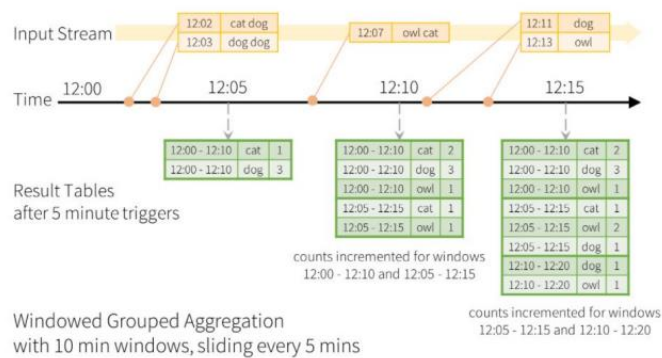
Porovnání

- | | |
|--|--|
| <ul style="list-style-type: none"> • Spark Streaming <ul style="list-style-type: none"> • původní API od verze 0.7 • založené na DStreams <ul style="list-style-type: none"> • interně využívá RDD's • optimalizace ze strany vývojáře • micro-batch zpracování • pouze timestamp doručení dat <ul style="list-style-type: none"> • ne vygenerování • opožděná data mohou být problém • stále k dispozici a používané <ul style="list-style-type: none"> • časem pravděpodobně deprecated • obtížnější propojení | <ul style="list-style-type: none"> • Structured Streaming <ul style="list-style-type: none"> • nové API od verze 2 • založené na DataFramach <ul style="list-style-type: none"> • případně DataSotech • optimalizace již zahrnuta • micro-batch i continuous • event-time koncept <ul style="list-style-type: none"> • zpracování i na základě času události • řeší problém opožděných dat • aktuální a více podporované <ul style="list-style-type: none"> • do budoucna hlavní řešení • jednodušší propojení |
|--|--|

Okna a práce s nimi.

STRUCTURED STREAMING

- veškeré operace je také možné aplikovat na okna
- možno aplikovat i na úvodní příklad čtení řádků z konzole



- veškeré operace je také možné aplikovat na okna
- k dispozici jsou různé typy oken
- statická (tumbling)
 - pevný interval
 - neexistují překryvy
- posuvná (sliding)
 - pevný interval
 - existují překryvy
 - např. 5 minut
- událostní (session)
 - různě dlouhé
 - existují pauzy

