

# Промышленное машинное обучение на Spark



# Содержание курса

- 01      Оборачиваем модель в сервис. Первая часть: Docker. Flask
- 02      Оборачиваем модель в сервис. Вторая часть: Requests. REST API
- 03      Распределенные вычисления. HDFS. MapReduce. Spark DataFrame
- 04      Погружение в среду Spark. RDD, SQL, Pandas API.
- 05      Генерация признаков. Spark feature engineering
- 06      Распределенное обучение моделей. Spark ML
- 07      Обработка и хранение текстовых данных и картинок.  
Spark image processing. Spark NLP.
- 08      Обработка потоковых данных. Spark Streaming.

# 08. Обработка потоковых данных. Structured Streaming и интеграция с spark.ml.

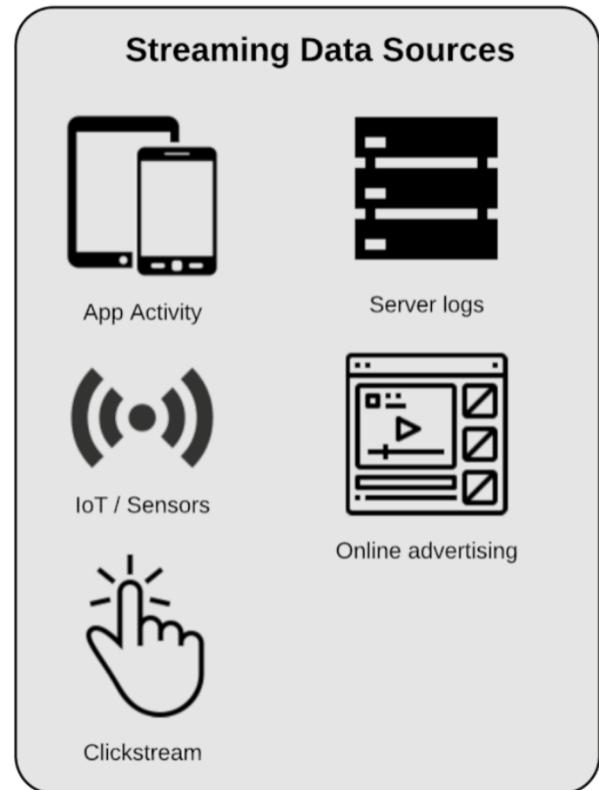
План:

1. Потоковые данные, где их искать.
2. Сценарии отказа. Обработка сценариев потери
3. Лямбда-архитектура.
4. Spark Streaming & DStream. Structured Streaming.
5. Интеграция с spark.ml.

Потоковые данные где их искать.

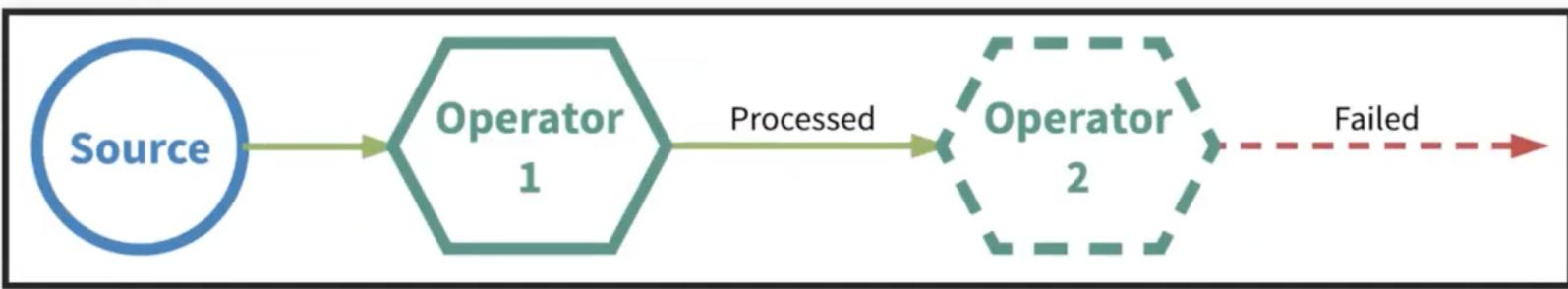
# Потоковые данные где их искать.

- Датчики IoT
- Журналы серверов и логи безопасности
- Реклама в режиме реального времени
- Передача данных из приложений и веб-сайтов по кликам

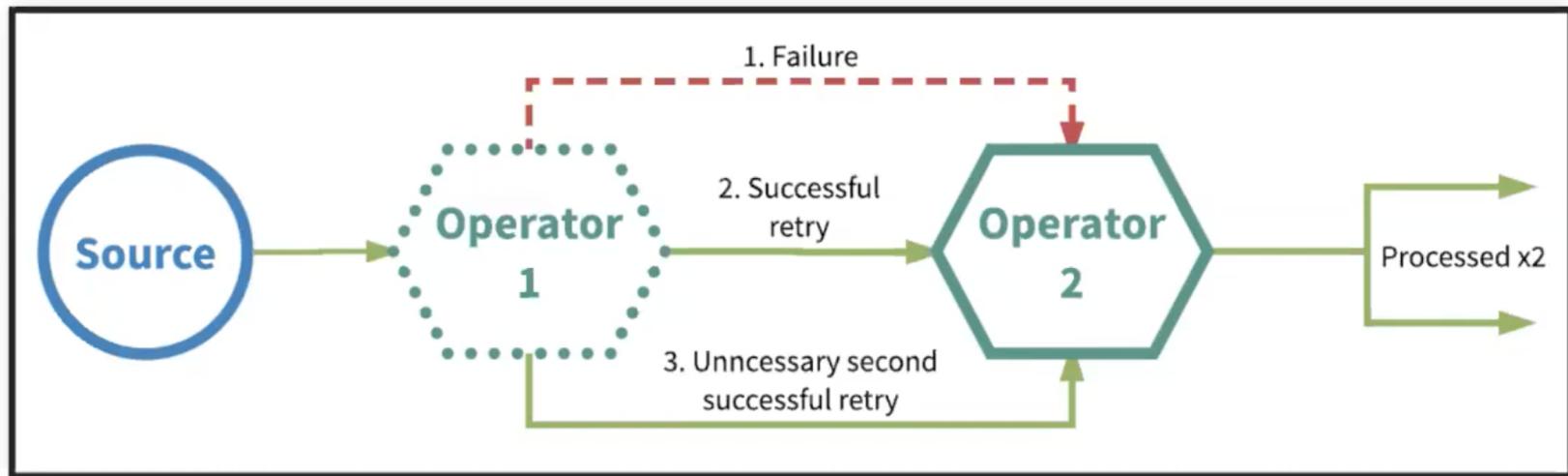


Сценарии отказа.

# Отказ во время обработки.



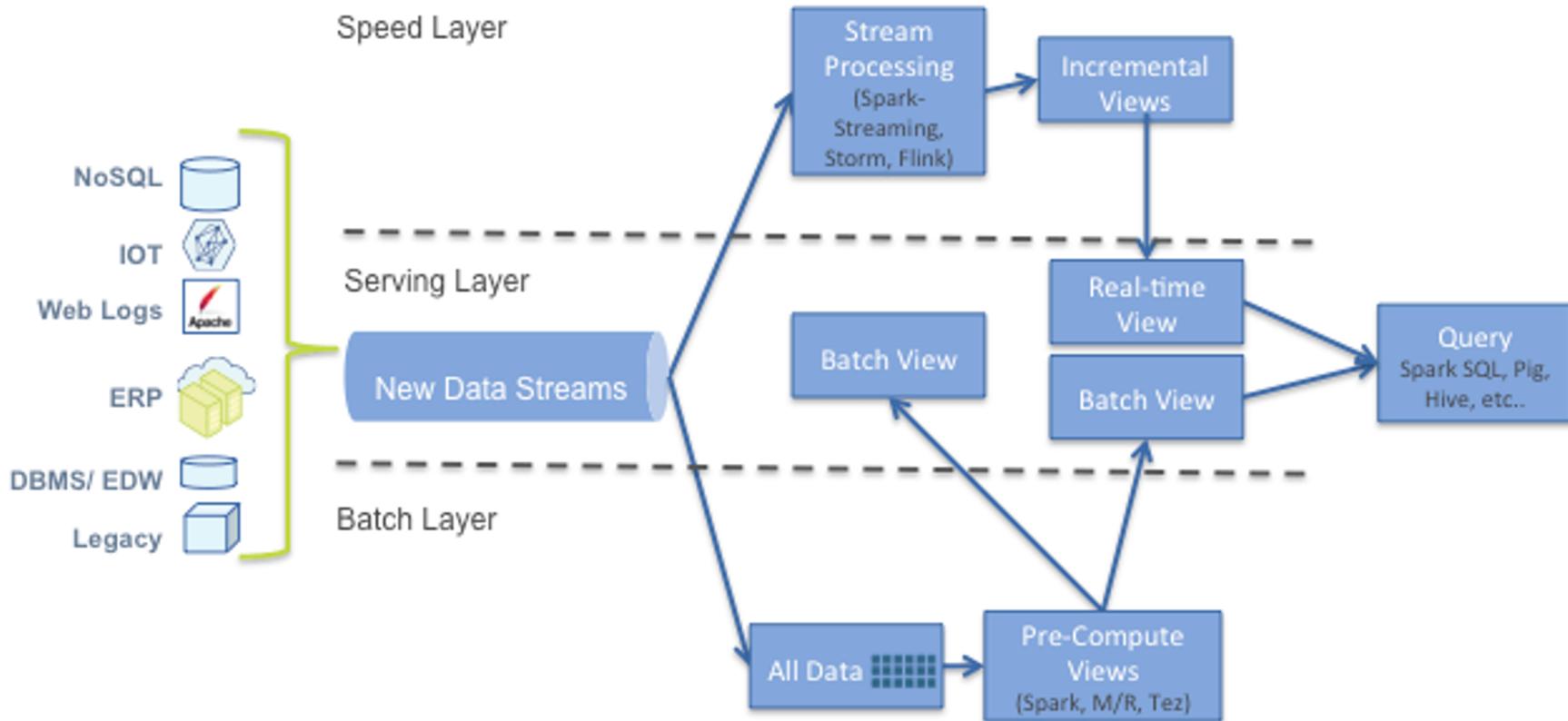
# Отказ во время отправки ответа о результатах обработки.



# Лямбда архитектура

# Лямбда-архитектура

Архитектура обработки данных, предназначенная для обработки больших объемов данных с использованием как пакетных, так и потоковых методов обработки. Этот подход к архитектуре пытается сбалансировать задержку, пропускную способность и отказоустойчивость с помощью пакетной обработки для обеспечения всестороннего и точного представления пакетных данных, одновременно используя потоковую обработку в реальном времени для обеспечения представления онлайновых данных.



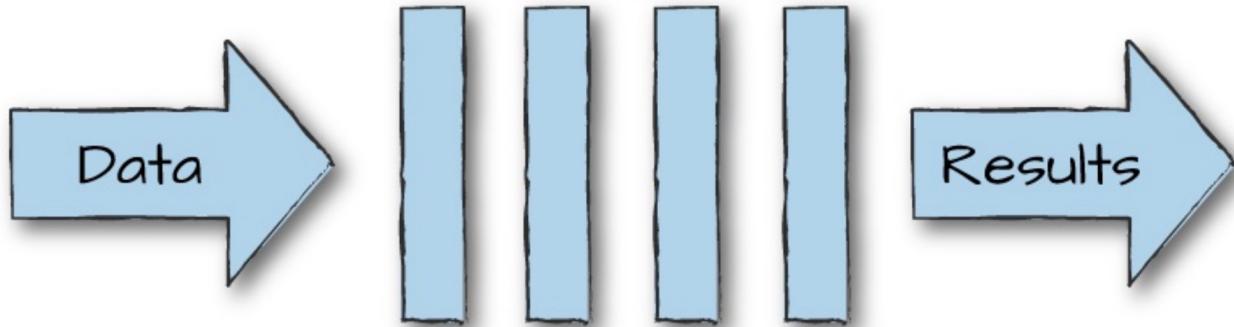
# Spark Streaming & DStream

# Spark Streaming

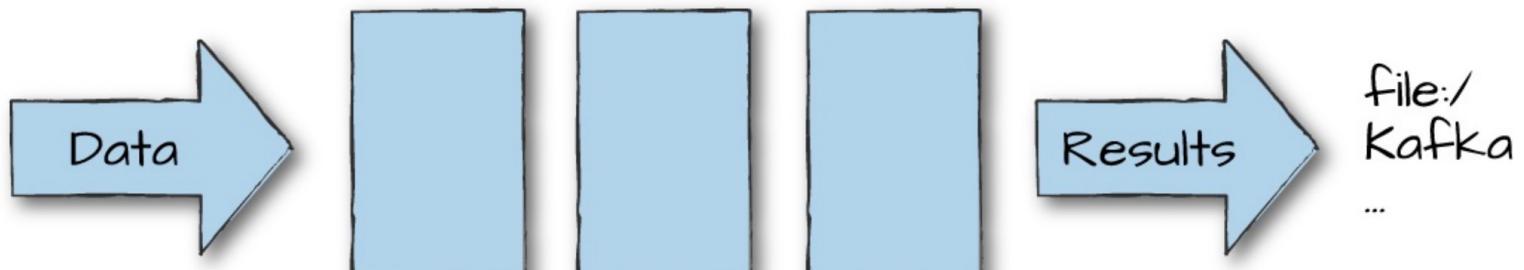
**Spark Streaming** - это расширение основного API Spark, которое обеспечивает масштабируемую, высокопроизводительную и отказоустойчивую потоковую обработку потоков данных в реальном времени. Данные могут быть получены из многих источников, таких как Kafka, Kinesis или TCP-сокеты, и могут быть обработаны с помощью сложных алгоритмов, выраженных с помощью высокоуровневых функций, таких как map, reduce, join и window. Наконец, обработанные данные могут быть перенесены в файловые системы, базы данных и живые информационные панели. Фактически, вы можете применить алгоритмы машинного обучения Spark и обработки графов к потокам данных.

# Spark Streaming





One record at a time



Microbatches of DataFrames

# Spark Streaming



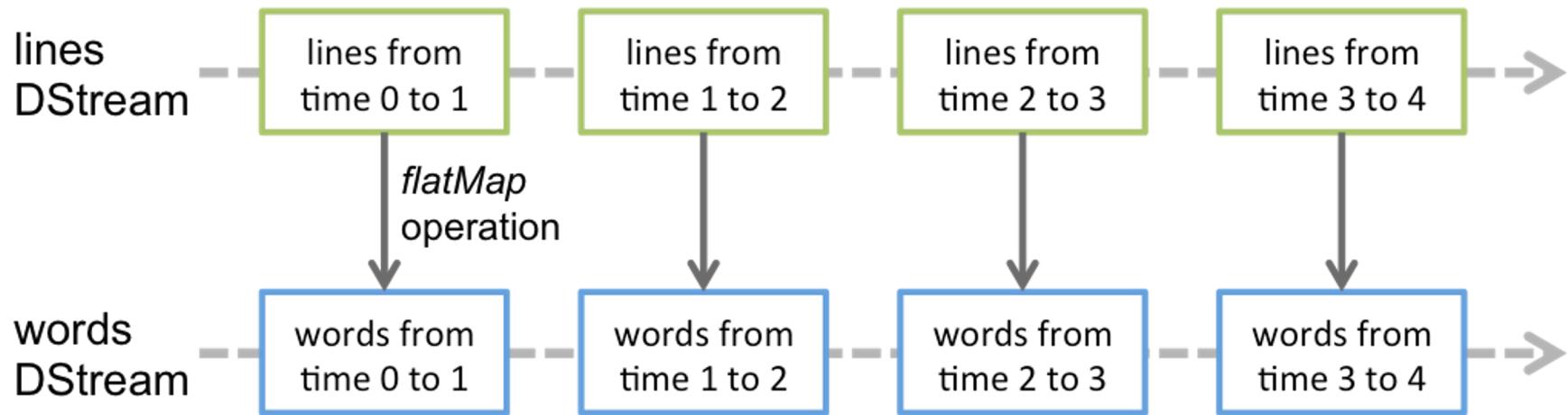
# DStream

Дискретизированный поток или **DStream**-это базовая абстракция, предоставляемая Spark Streaming. Он представляет собой непрерывный поток данных, либо входной поток данных, полученный из источника, либо обработанный поток данных, созданный путем преобразования входного потока. Внутренне DStream представлен непрерывной серией RDDS, которая является абстракцией Spark неизменяемого распределенного набора данных (подробнее см. Руководство по программированию Spark). Каждый RDD в потоке содержит данные с определенного интервала, как показано на следующем рисунке.

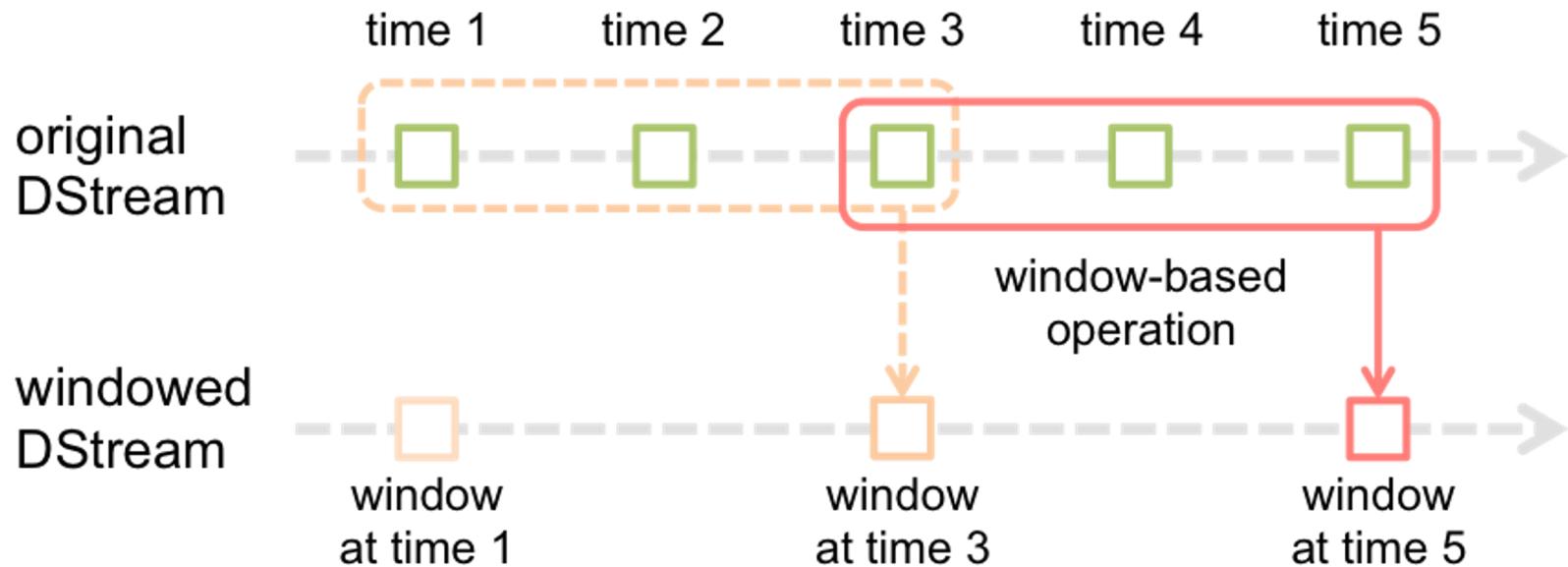
# DStream



# DStream



# DStream



# ML & DStream

Вы также можете легко использовать алгоритмы машинного обучения, предоставляемые **MLlib**. Прежде всего, существуют алгоритмы потокового машинного обучения (например, потоковая линейная регрессия, Потоковые KMeans и т. Д.), которые могут одновременно учиться на потоковых данных, а также применять модель к потоковым данным. Помимо этого, для гораздо более широкого класса алгоритмов машинного обучения вы можете изучить модель обучения в автономном режиме (то есть с использованием исторических данных), а затем применить модель онлайн к потоковым данным.

# Structured streaming & Dataframe

# Structured streaming

Ключевая идея структурированной потоковой передачи состоит в том, чтобы рассматривать поток живых данных как таблицу, которая постоянно добавляется. Это приводит к новой модели потоковой обработки, которая очень похожа на модель пакетной обработки. Вы будете выражать свои потоковые вычисления в виде стандартного пакетного запроса, как в статической таблице, а Spark запускает его как инкрементный запрос в неограниченной входной таблице.

Data stream



Unbounded Table

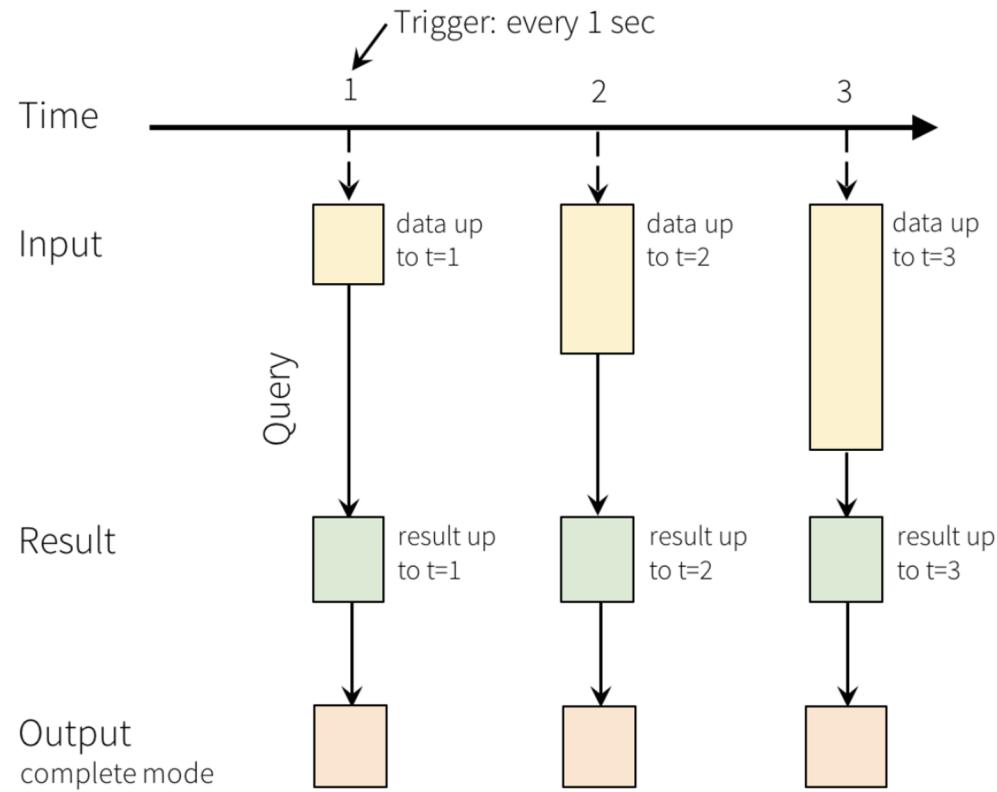
new data in the  
data stream

=

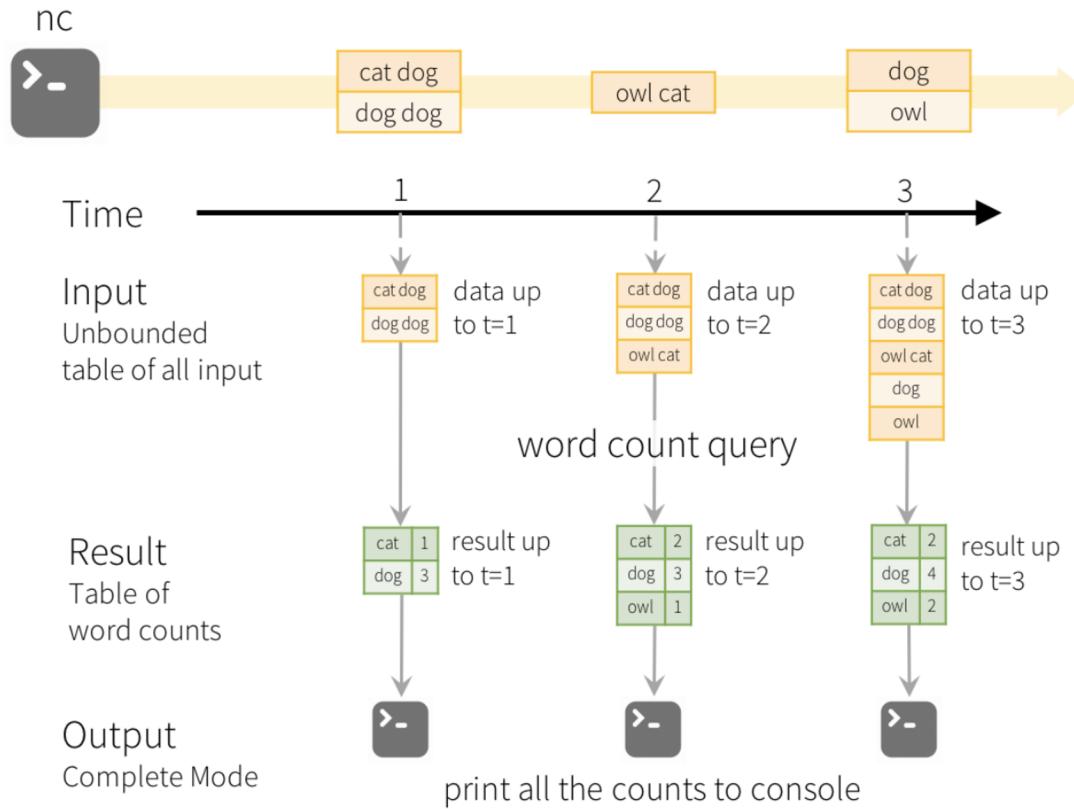
new rows appended  
to a unbounded table

Stream as a Table

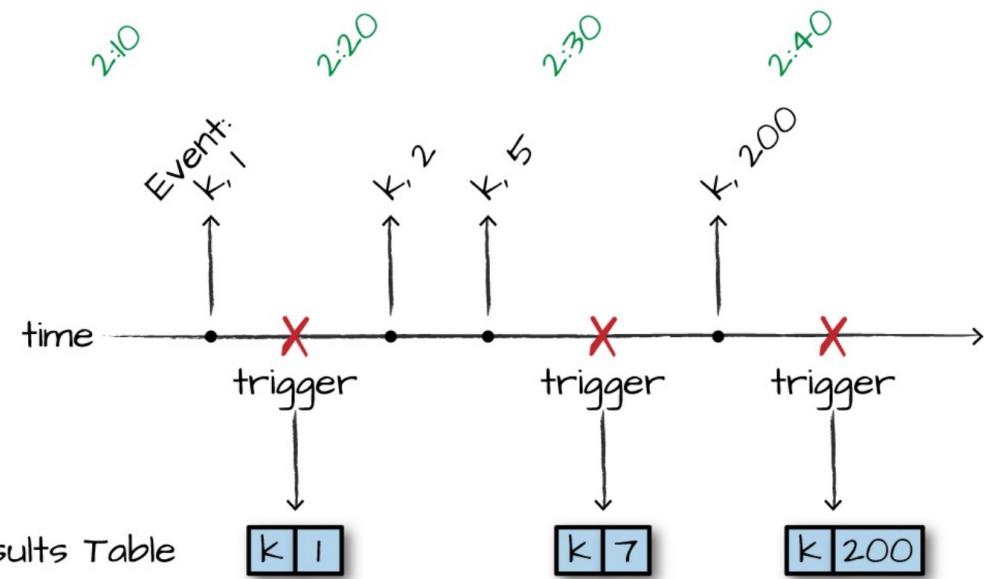
Data stream as an unbounded table

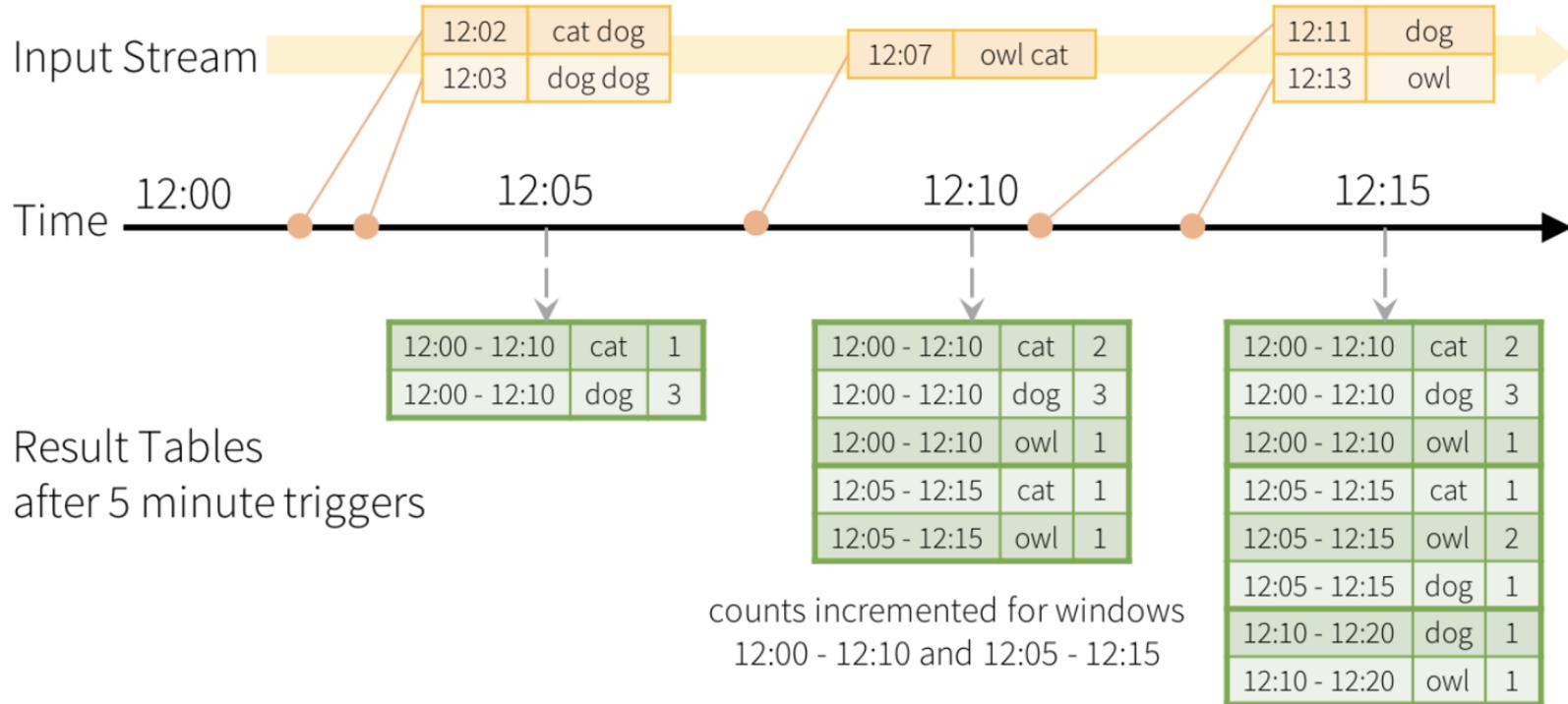


Programming Model for Structured Streaming

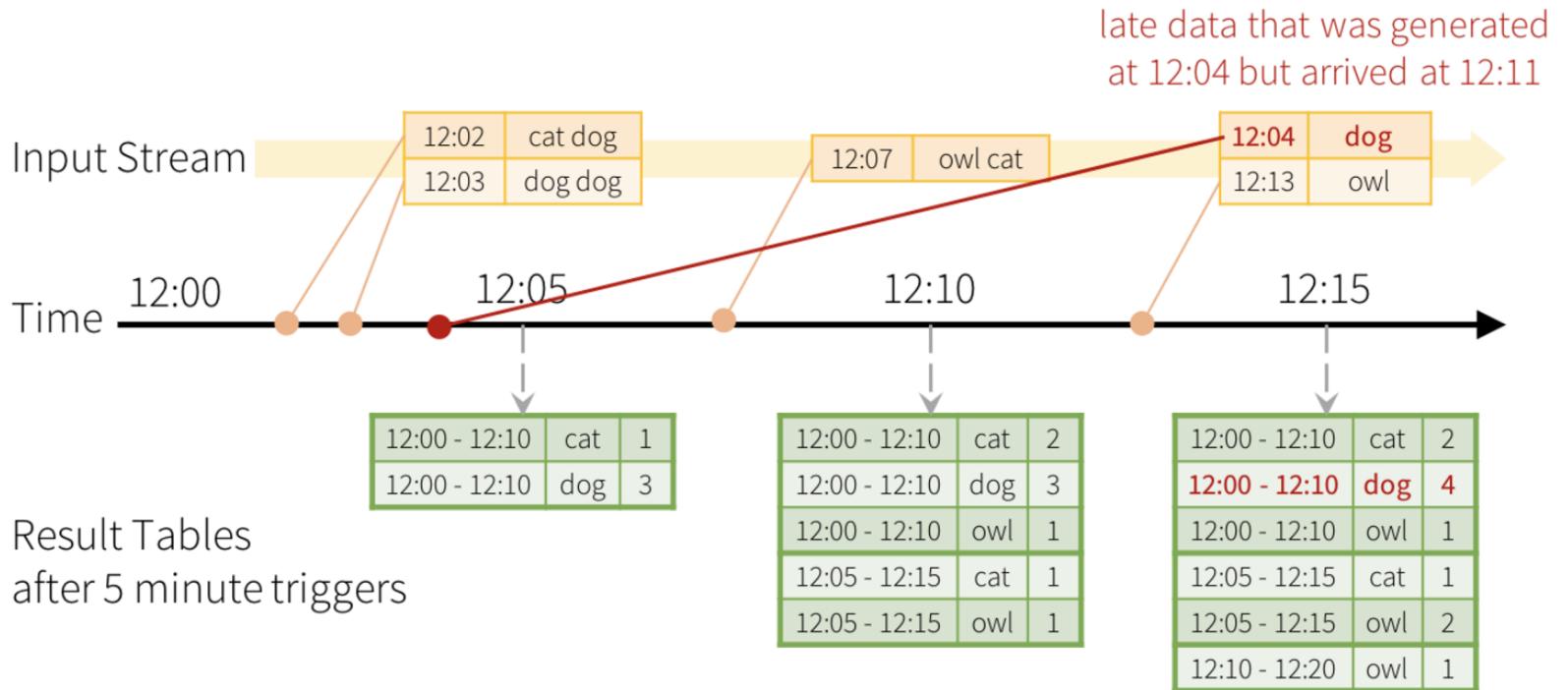


Model of the Quick Example





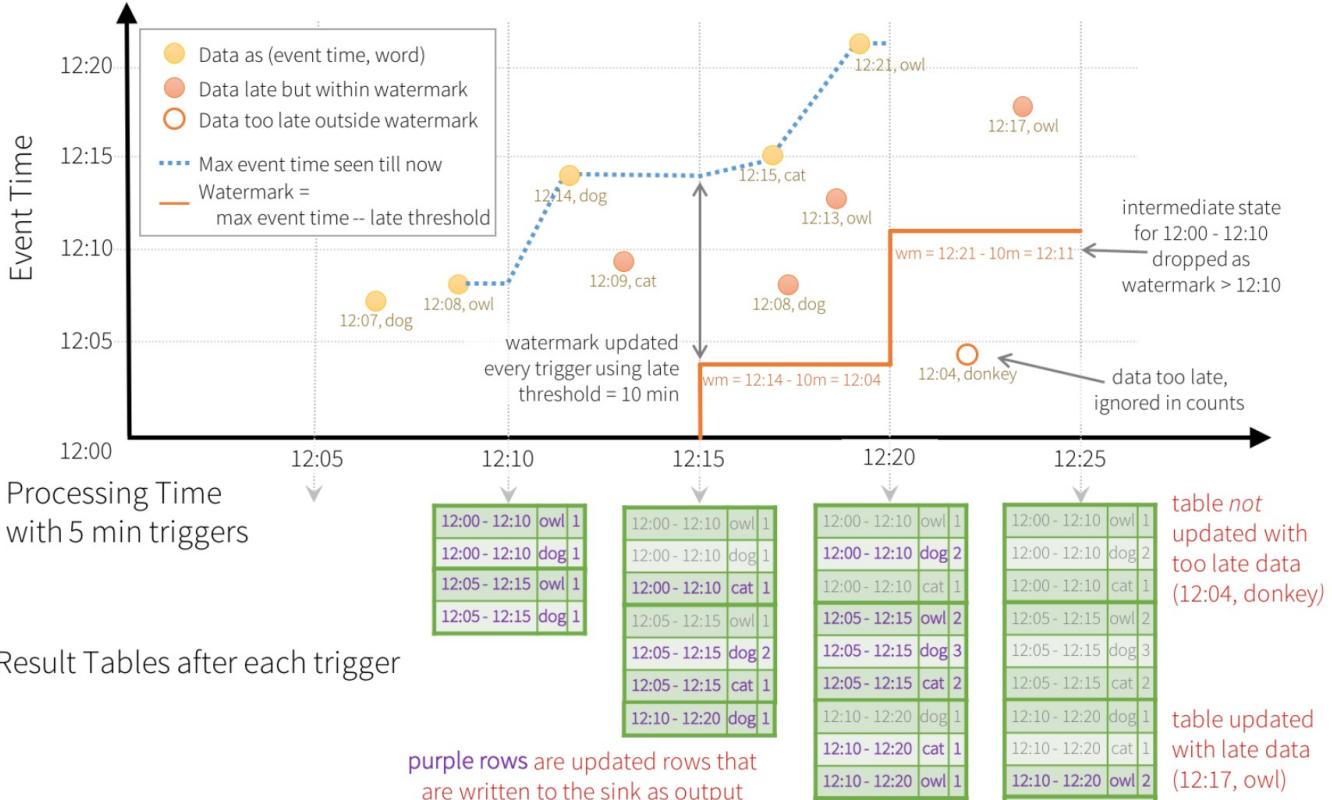
Windowed Grouped Aggregation  
with 10 min windows, sliding every 5 mins



Late data handling in  
Windowed Grouped Aggregation

late data that was generated  
at 12:04 but arrived at 12:11

counts incremented only for  
window 12:00 - 12:10



Watermarking in Windowed  
Grouped Aggregation with Update Mode