Ecossistema HADOOP

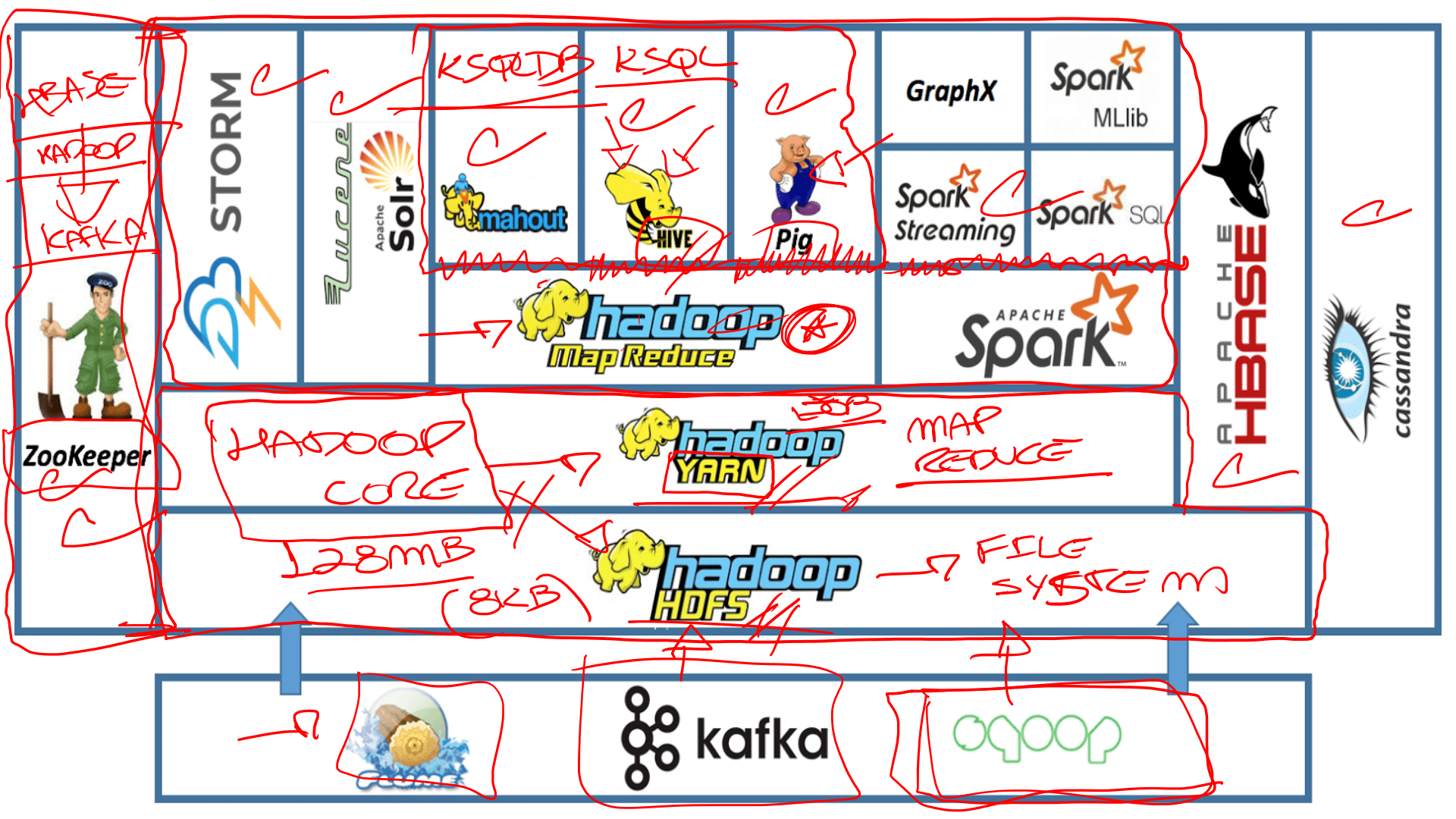
Júlio Alcântara Tavares, MSc
Instrutor

AI Data School


BIG
DATA
hadoop

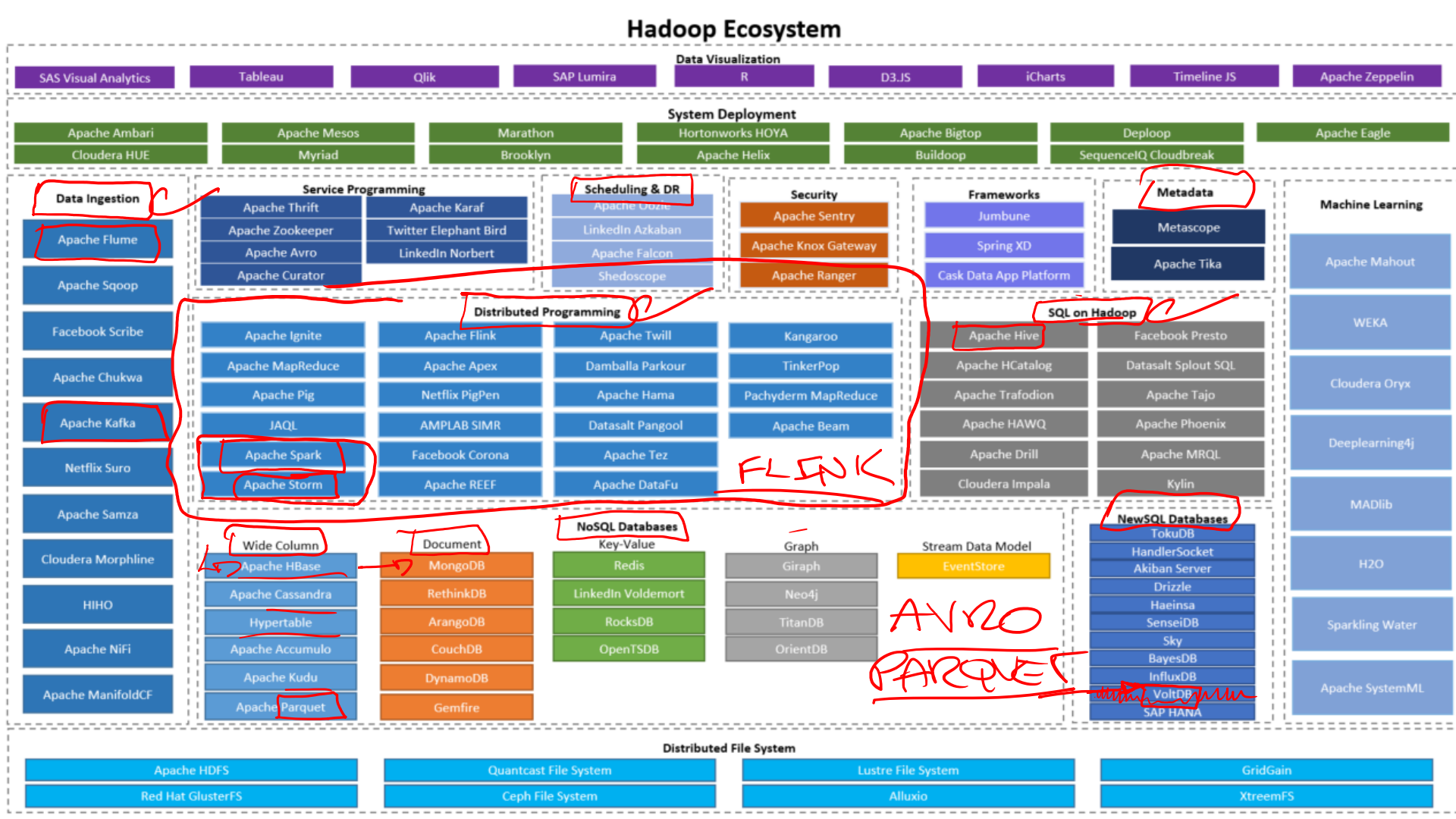
Big Data e o Processamento De Grandes Volumes De Dados

Arquitetura e Principais Módulos do Ecossistema HADOOP: Visão Geral


STORM
Apache Solr
mahout
HIVE
Pig
GraphX
Spark MLlib
Spark Streaming
Spark SQL
hadoop Map Reduce
APACHE Spark
APACHE HBASE
cassandra
ZooKeeper
hadoop YARN
hadoop HDFS
kafka

Arquitetura e Principais Módulos do Ecossistema HADOOP: Visão Detalhada

Hadoop Ecosystem



Detalhamento dos Módulos

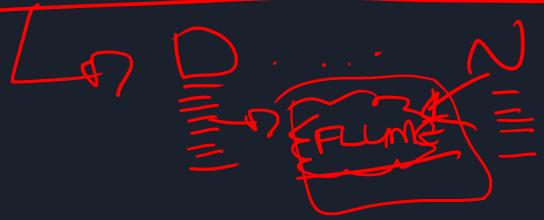
Apache ZOOKEEPER

Apache ZooKeeper is an effort to develop and maintain an open-source server which enables highly reliable distributed coordination.

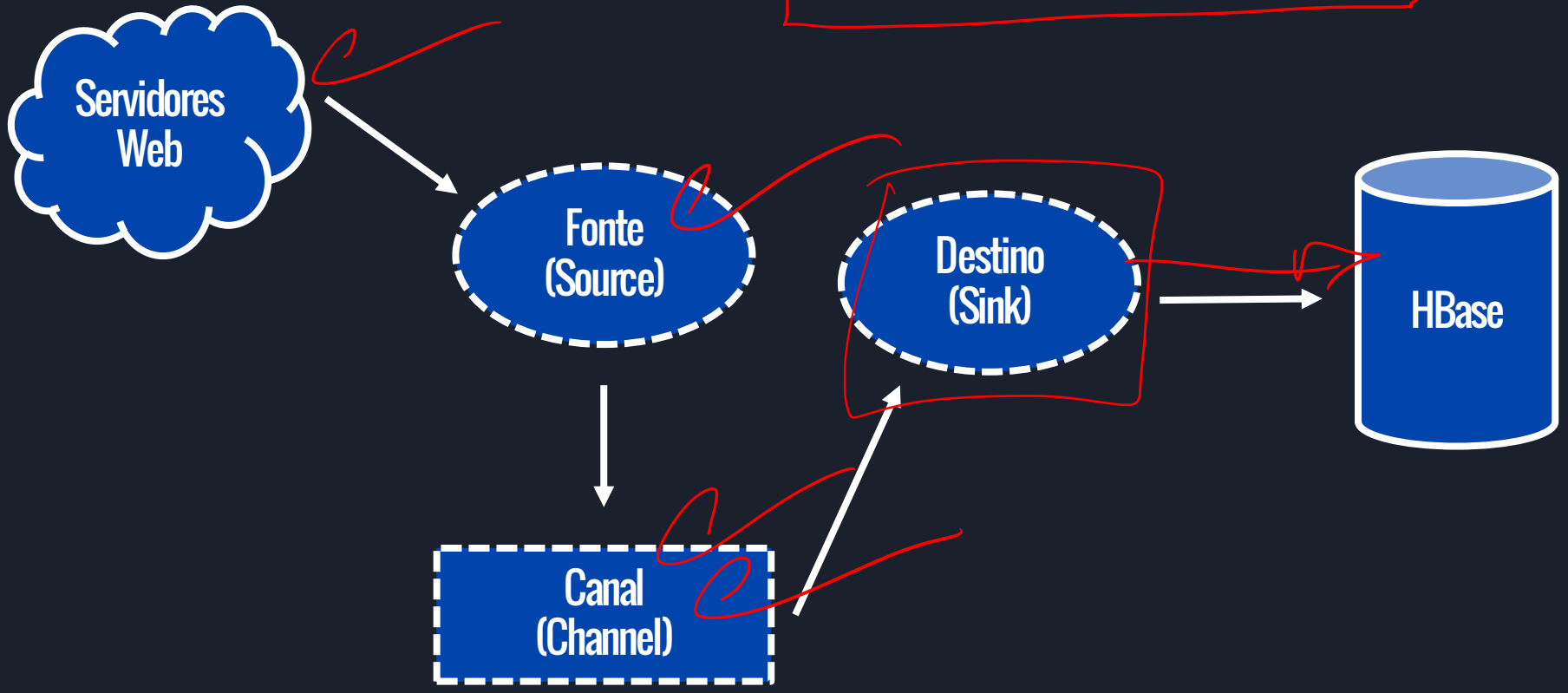
Apache FLUME

Apache Flume

- Uma outra forma de realizar o stream de dados para dentro do cluster hadoop.
- Implementado desde o início com o HADOOP em mente
  - Possui SINKs para HDFS e Hbase.
- Originalmente pensado para trabalhar com o problema da agregação de LOGs.



Anatomia de um Agente no Flume



Apache Flume

- Componentes de um Agente
  - Source (Fonte)
    - De onde os dados estão surgindo
    - Opcionalmente, é possível ter "Channel Selectors" e "Interceptors" (possibilidade de modificar/eliminar eventos on-the-fly).
  - Channel (Canal)
    - Forma como os dados são transferidos
      - Memória
      - Arquivo

Apache Flume

- Componentes de um Agente
  - Sink (Destino)
    - Para aonde os dados estão indo
    - Podem ser organizados em Sink Groups
    - Um Sink pode se conectar exclusivamente com um canal
      - O canal é notificado para deletar a mensagem logo que um “SINK” processar este dado.

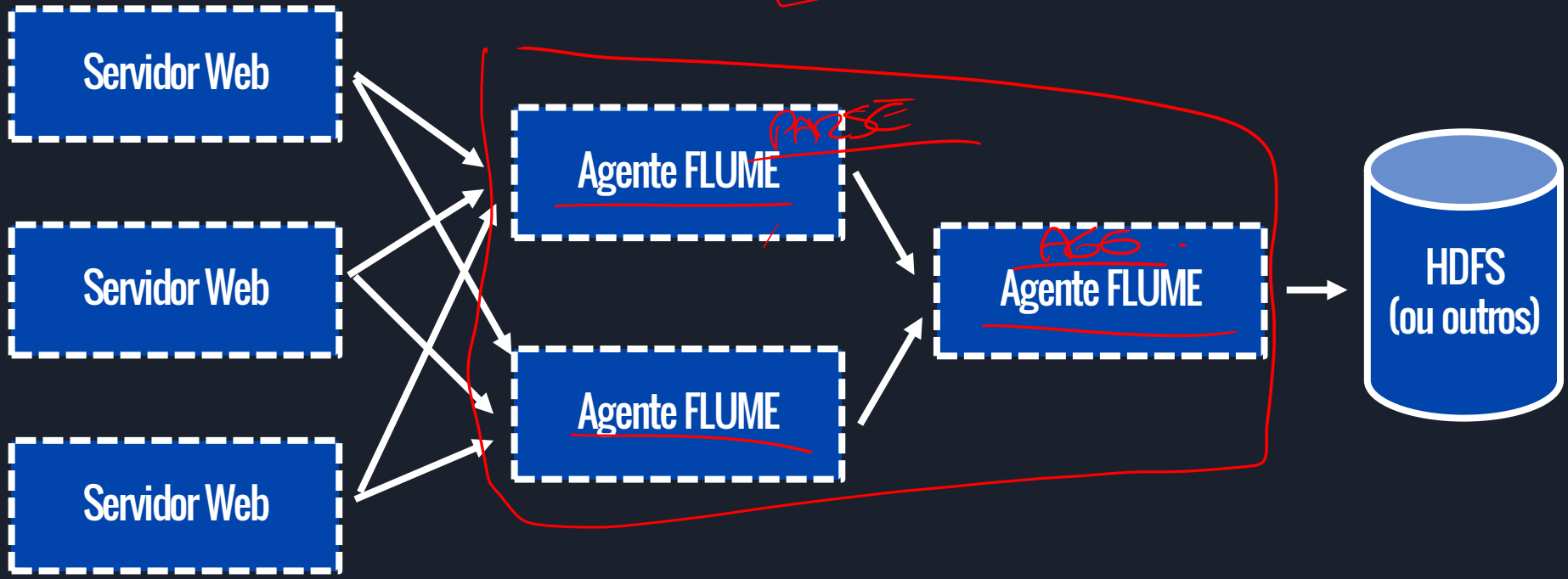
Apache Flume

- Tipos de SOURCE (Fontes Built-IN)
  - Spooling directory
  - Avro
  - Kafka
  - Exec
  - Thrift
  - Netcat
  - HTTP
  - Custom
  - E vários outros!

Apache Flume

- Tipos de SINK (Built-IN)
  - HDFS
  - Hive
  - HBase
  - Avro
  - Thrift
  - Elasticsearch
  - Kafka
  - Custom
  - Dentre vários outros!

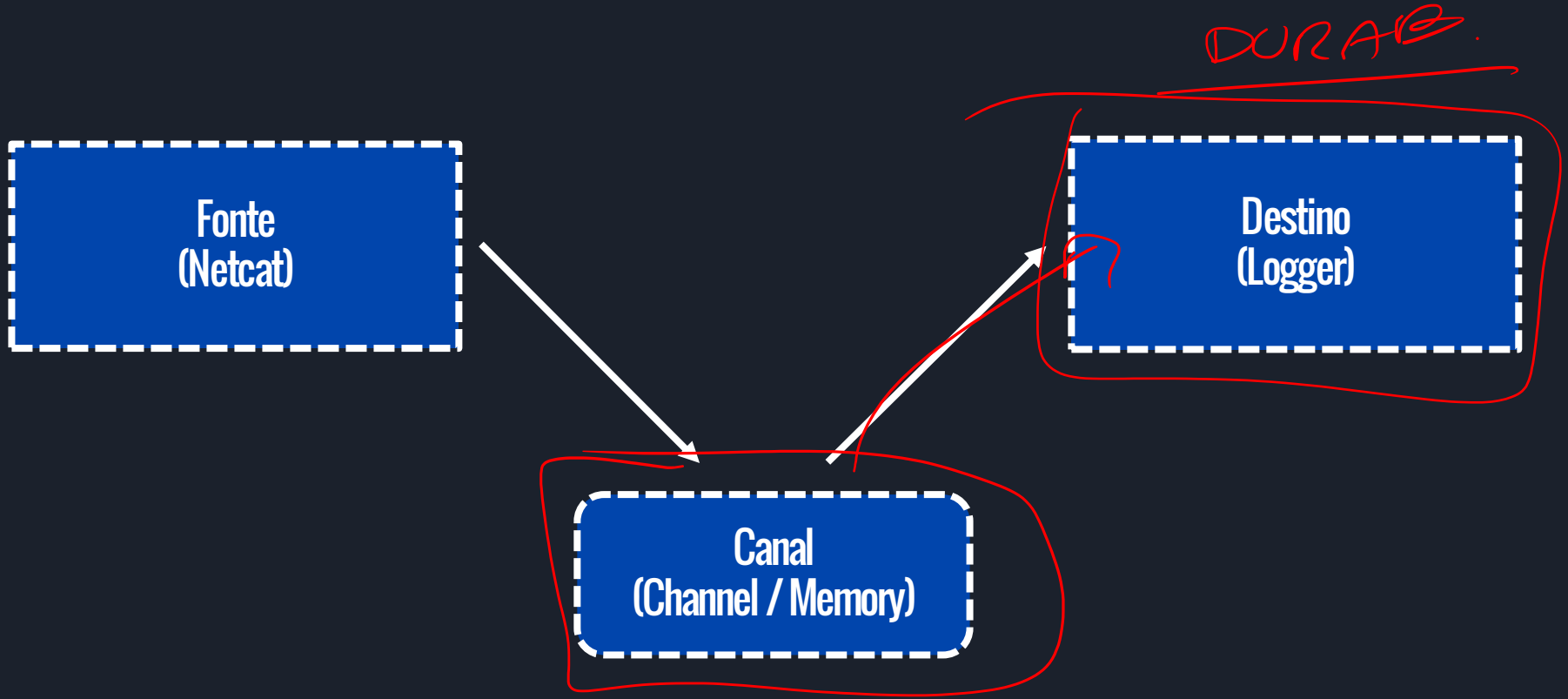
Usando AVRO, os agentes podem trocar informações entre si. (Observar topologia)



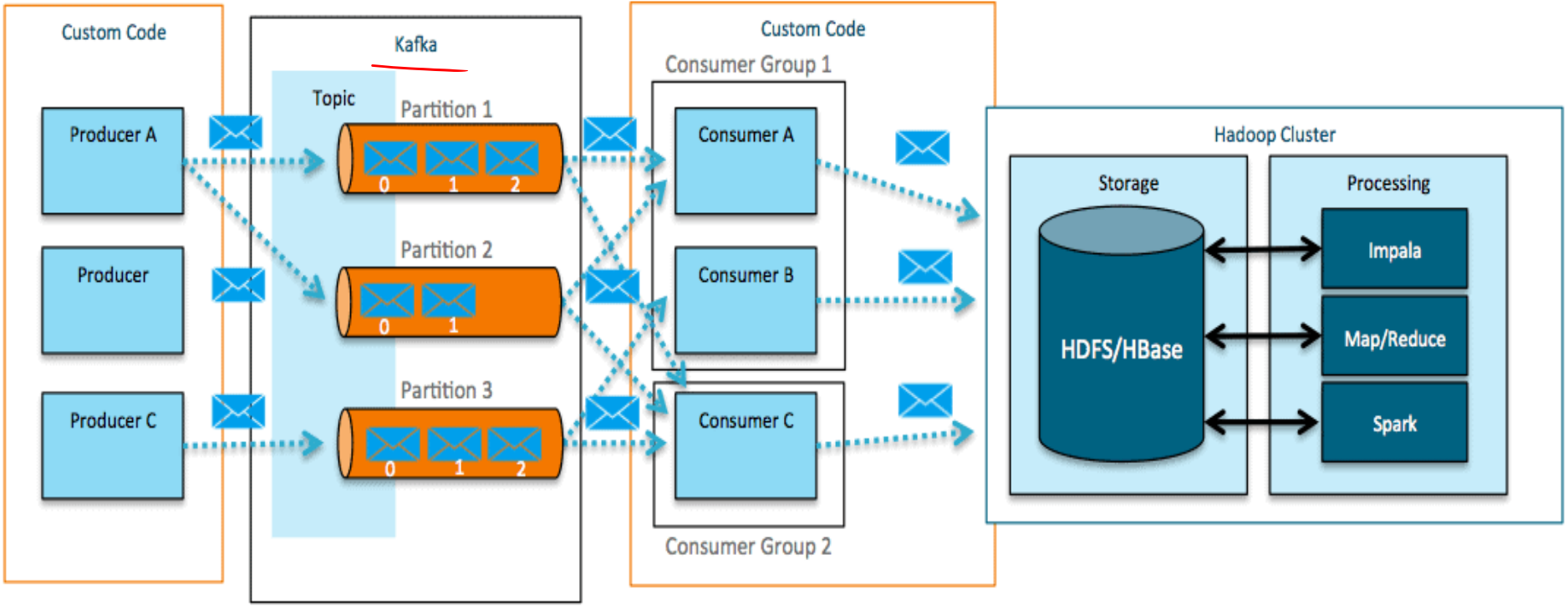
Apache Flume

Pode ser “pensado” como uma camada de buffer entre os dados e o cluster.

Exemplos de Workflow



Exemplos de Workflow



Raw Data

Apache STORM


“Real-time stream processing”

Apache Storm

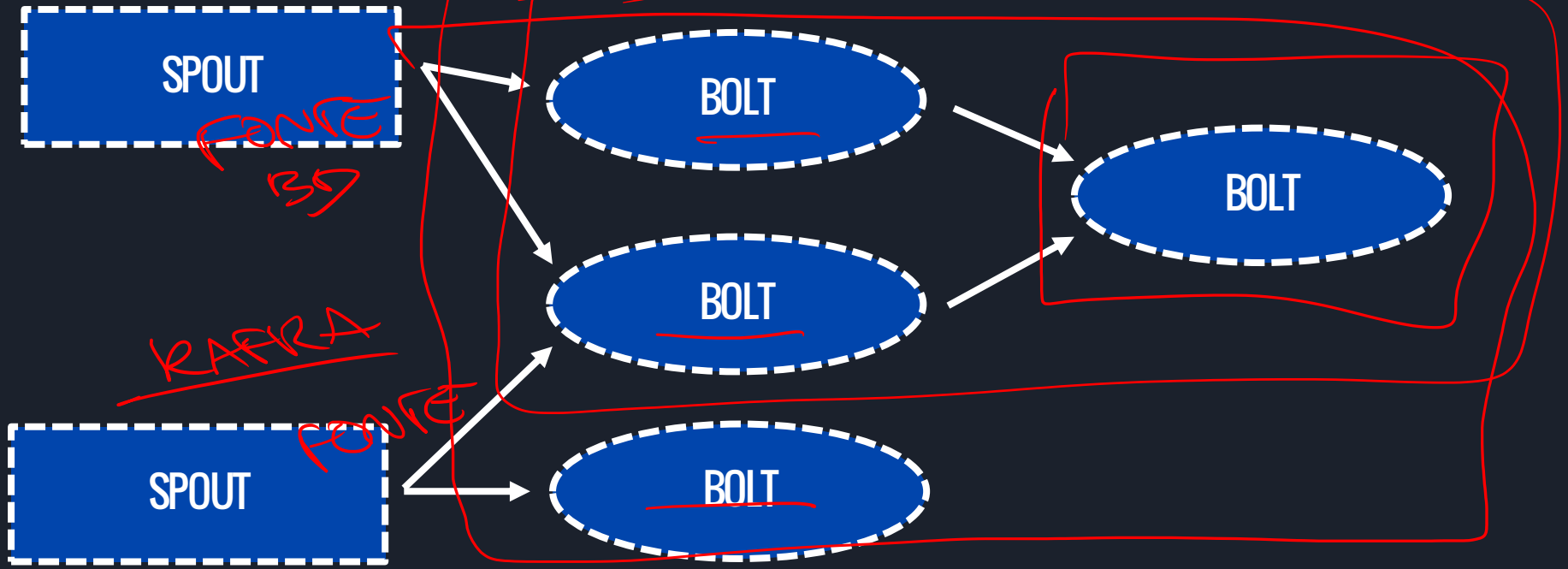
- Outro framework para processar stream contínuos de dados em um cluster, de forma distribuída
- Trabalha em eventos individuais e não em "micro-batches" (como é o caso do Spark), ou seja, trabalha bem próximo de real time
- Ideal se você precisa de uma latência a nível de "sub-seconds"

Apache Storm

- Arquitetura
  - Um stream é composto por tuplas fluem através de:
    - Spouts (São fontes de stream: Kafka, Twitter, etc)
    - Bolts (Processam dados a medida que são recebidos)
      - Transformam, agregam, gravam no BD e em HDFS

Apache Storm

- Uma topologia é um grafo de SPOUTS e BOLTS que processam seu stream



Apache Storm

- Uma topologia é um grafo de SPOUTS e BOLTS que processam seu stream

Apache Storm

- Storm vs Spark Streaming
  - Spark Streaming: possui mais opções de “extensão” e bibliotecas
  - Storm: Ideal para processamento realmente em tempo real (a nível de evento, com latência próxima de zero)
  - Storm: oferece o recurso “Tumbling Windows” (muito mais preciso) ao invés da “Sliding Windows” do Spark
  - Kafka + Storm e Kafka + Spark Streaming: ambos são combinações populares

Apache FLINK

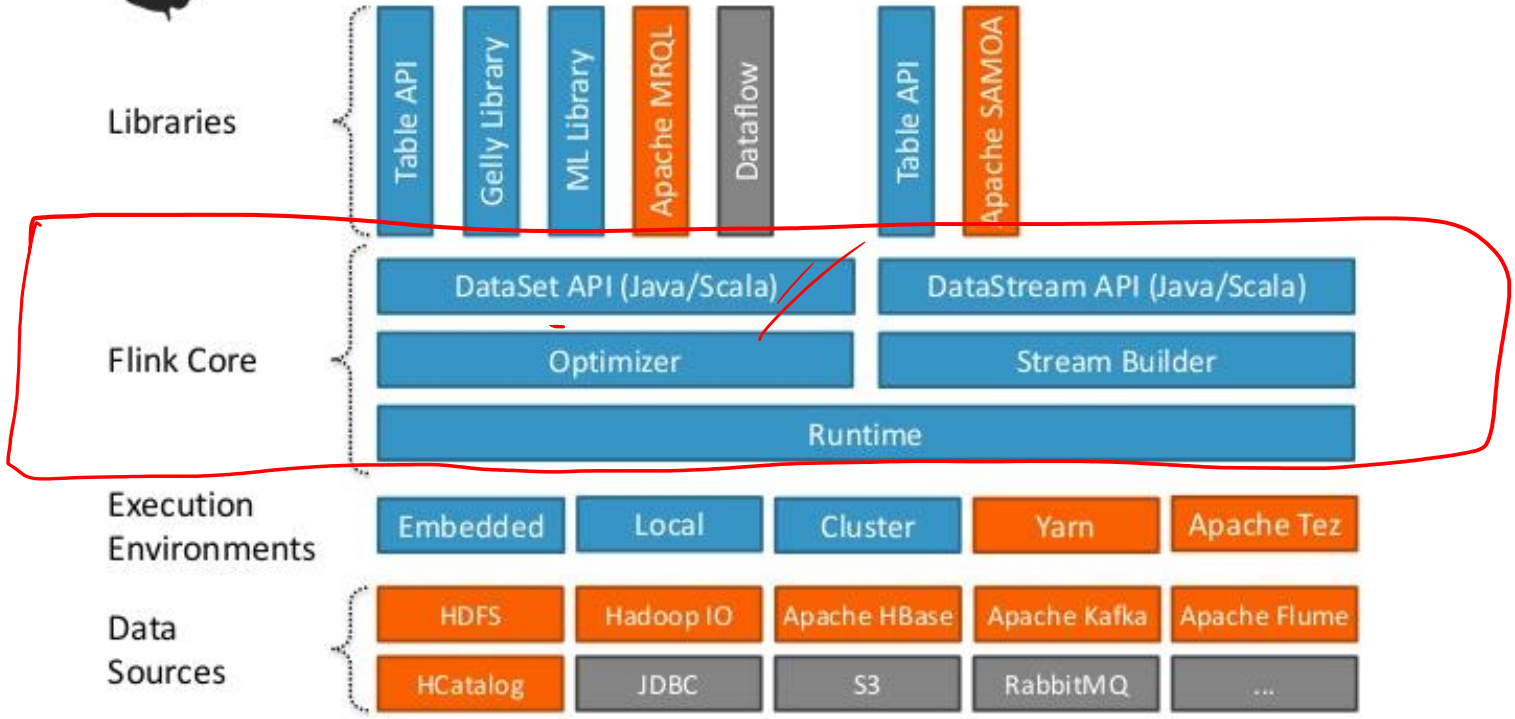
Apache Flink

- Mais uma opção para Stream de Dados!
  - Em relação ao objetivo, é semelhante ao STORM
- Pode ser executado de forma STANDALONE ou em clusters com YARN e MESOS
- Altamente Escalável (1000's de Nós)
- Tolerante a Falhas
  - Mesmo em falhas garante a semântica de "exactly-one processing"
  - Usa mecanismos de LOG para isso
- Muito mais rápido que o STORM!!!

Apache Flink

- Flink vs Spark Streaming vs Storm
  - Flink é muito mais rápido que o STORM, pois sua arquitetura implementa uma camada muito leve quando comparado com o STORM
  - Flink possui seu próprio ecossistema (assim como o SPARK)
    - Entretanto, o SPARK é muito mais evoluído neste quesito
  - Flink pode processar os eventos baseado no seu TIMESTAMP (e não na ordem em que os eventos foram recebidos no cluster)
  - Flink: Dentre todos, é a tecnologia mais nova (ainda em evolução, mas já bastante usado)

Flink in the Hadoop Ecosystem



Apache Flink

- Possui diversos conectores:
  - HDFS
  - Cassandra
  - Kafka
  - Elasticsearch, NiFi, Redis, RabbitMQ
  - E Muitos Outros!


Dúvidas?


Thank
you