TELECOM ParisTech

INSTITUT Mines-Télécom

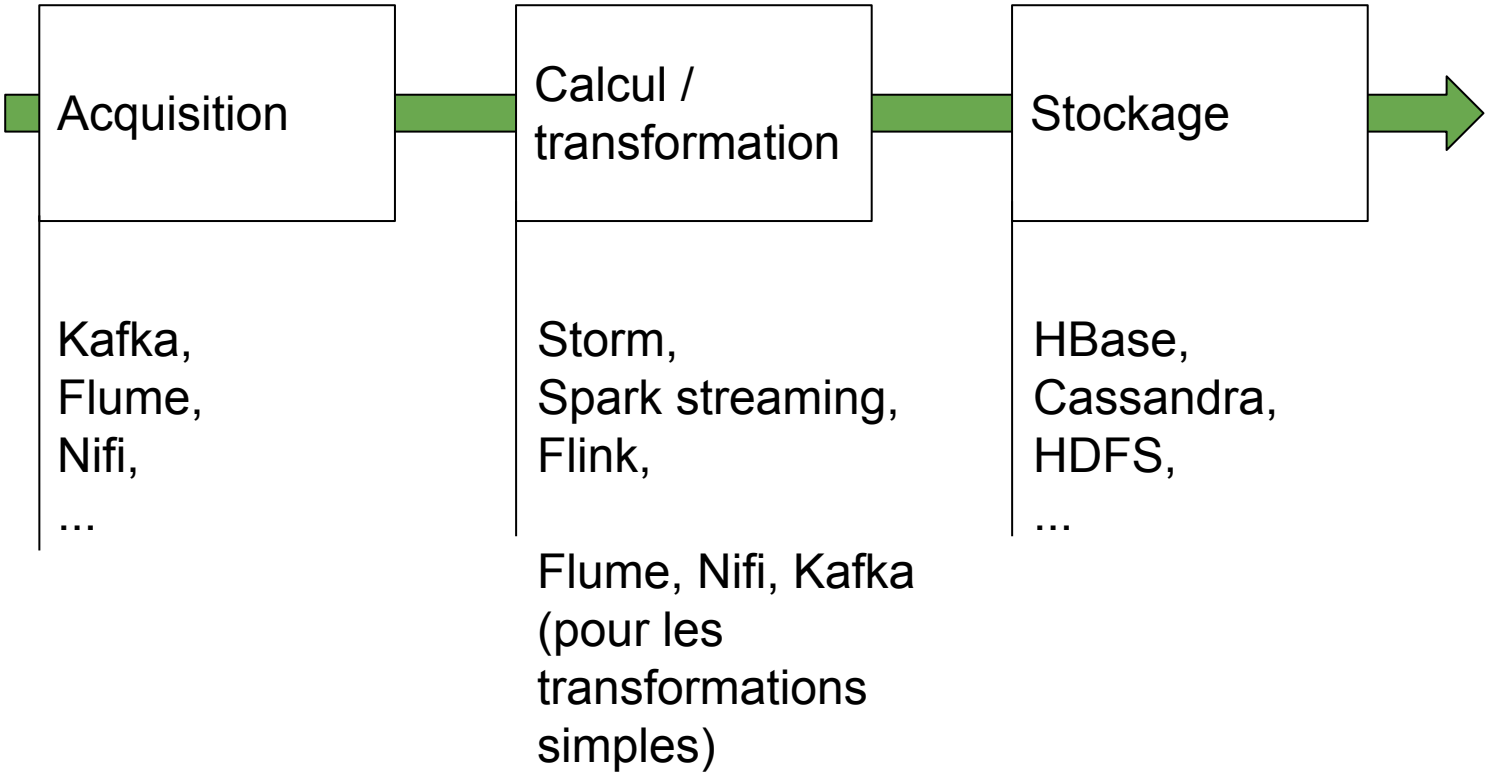
2018/2019

STREAM PROCESSING

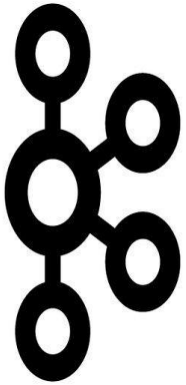
MOTIVATION STREAM PROCESSING DANS HADOOP

- Permet l'acquisition et le traitement au fil de l'eau de l'information
- Exemple de projet stream processing dans Hadoop:
  - Suivi des informations envoyées par des IOT au moment où on les reçoit
  - projet de détection d'attaques informatiques à partir de logs machines
  - Maintenance prédictive

OUTILS DE GESTION DE FLUX



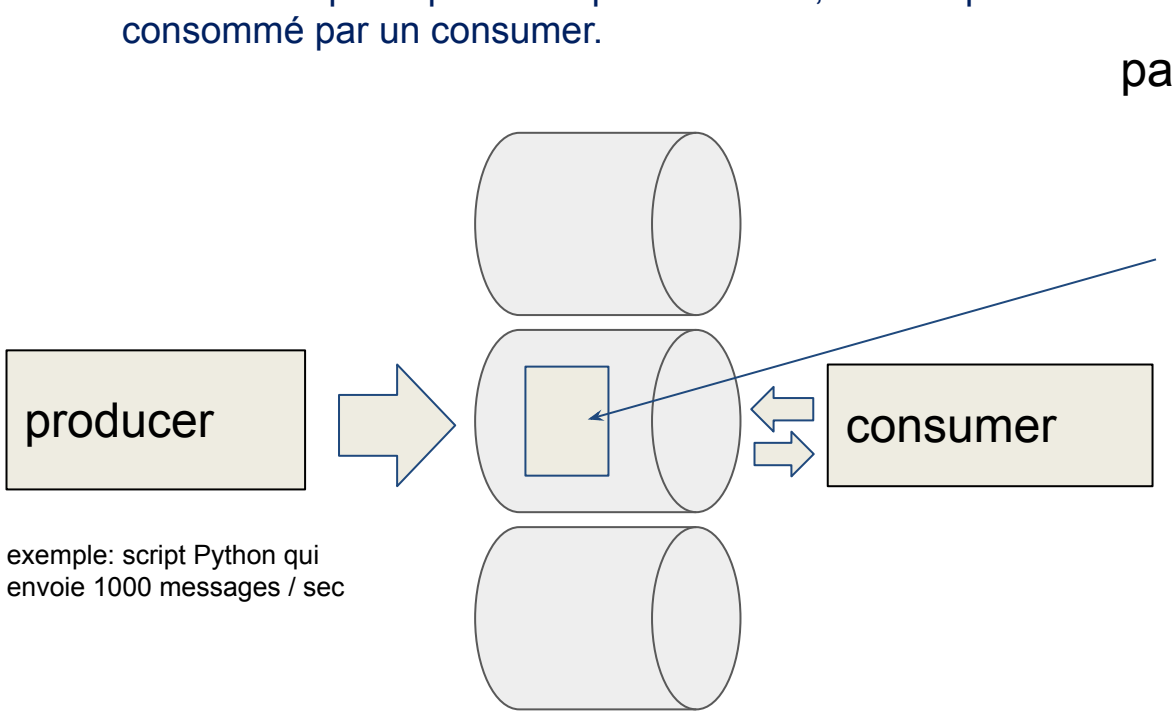
KAFKA



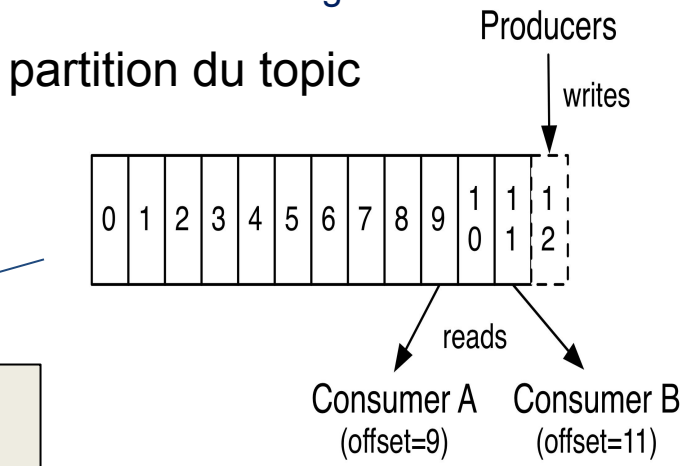
- Message Oriented Middleware, utilisé comme base de données bas niveau en entrée des systèmes consommateurs. C'est un système de stockage de flux de messages.
- On y définit des topics dans lesquels on peut produire des messages (publish) sur une ou plusieurs partitions.
- On peut s'abonner (subscribe) à un topic afin de lire les messages.
- Kafka s'exécute sur un cluster. Chaque noeud du cluster est appelé broker
- Architecture fault tolerant, possibilité de choisir le nombre de réplicas (sur différents brokers).
- Les messages sont constitués d'une clé, valeur et timestamp
- Kafka API :
    - **producer** : l'application qui envoie les messages dans Kafka
    - **consumer** : l'application qui lit les messages dans Kafka
    - **stream** : permet de réaliser une transformation sur la donnée dans Kafka
    - **connector** : permet d'utiliser des connecteurs pour s'interfacer avec d'autres systèmes informatiques (base de données,...)

KAFKA

- Cluster de 3 brokers, sur lequel on crée le topic “fr.telecom.kafkatp”
- Kafka va créer un fichier sur un des brokers appelé “partition”, spécifique au topic. C’est dans ce fichier que les producers publiant dans ce topic vont écrire et que les consumers abonnés au topic vont lire
- La partition est créée sur un des brokers
- L’offset est spécifique à chaque consumer, il correspond à l’id du dernier message consommé par un consumer.

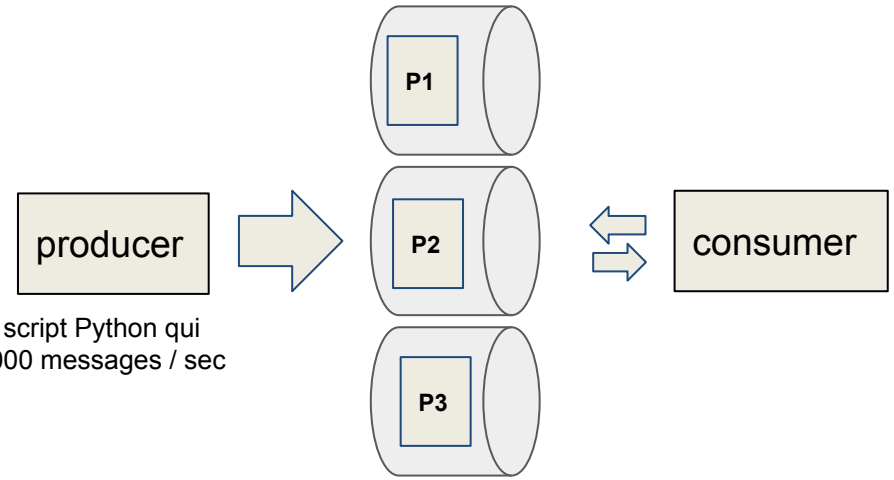


exemple: script Python qui envoie 1000 messages / sec

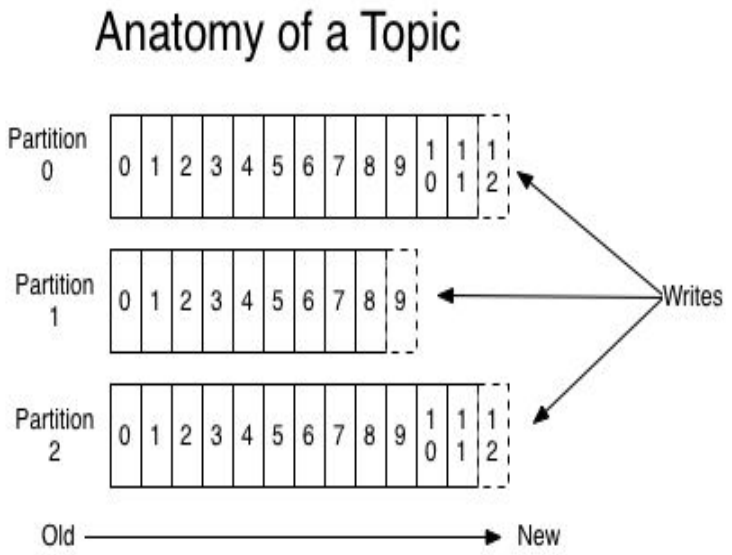


KAFKA

- Cluster de 3 brokers, sur lequel on crée le topic “fr.telecom.kafkatp” avec trois partitions
- Afin d'augmenter les performances en écriture/lecture, on peut créer des topics avec plusieurs partitions.



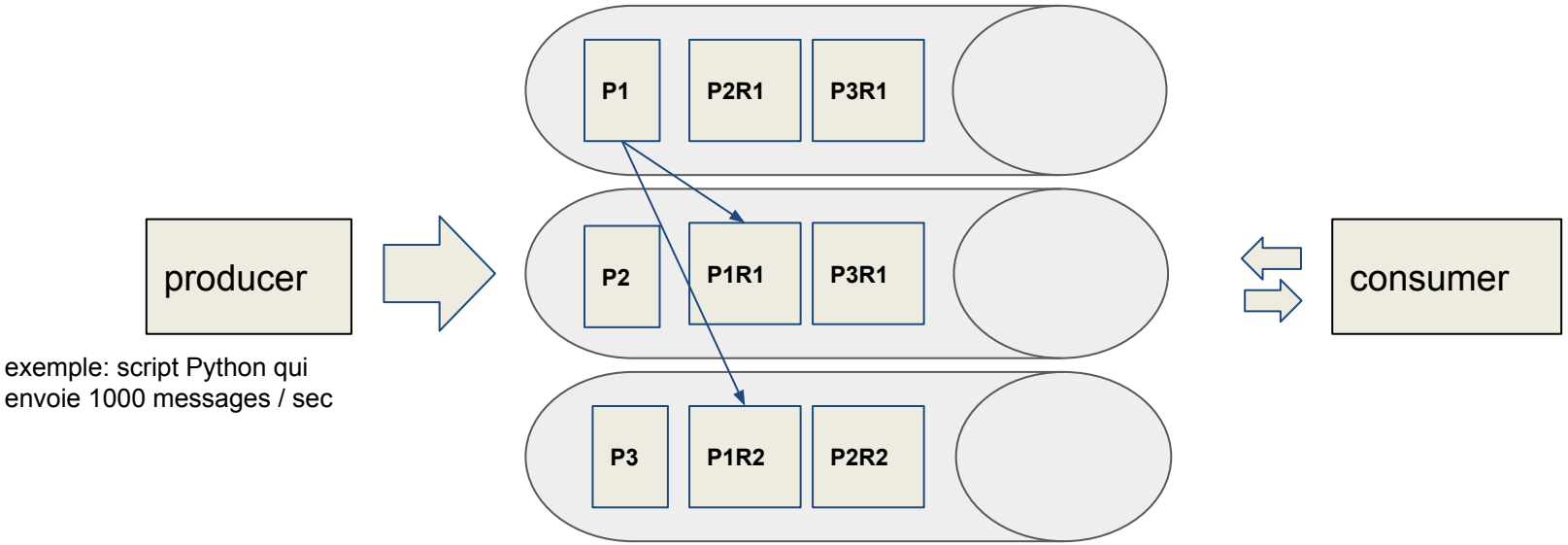
exemple: script Python qui envoie 1000 messages / sec



- Que se passe-t-il si un broker tombe ?

KAFKA

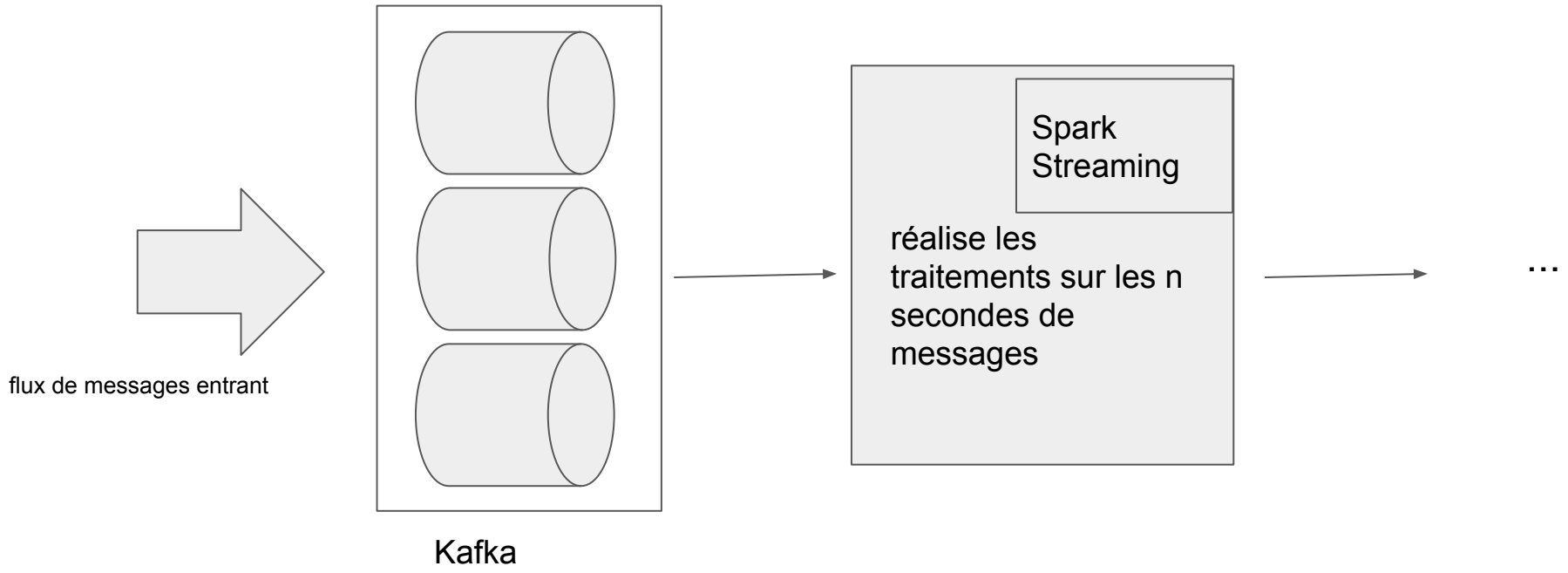
- Cluster de 3 brokers, sur lequel on crée le topic “fr.telecom.kafkatp” avec 3 partitions et un replication factor de 3
- Un topic peut être créé avec un facteur de réplication, il permet de rendre le topic “fault tolerant”
- Avec un facteur de réplication de 3 (best practice), chaque partition sera disponible sur 3 noeuds.
- Une partition est leader, si le broker la contenant tombe, un autre réplica de la partition est choisi pour être leader.



exemple: script Python qui envoie 1000 messages / sec

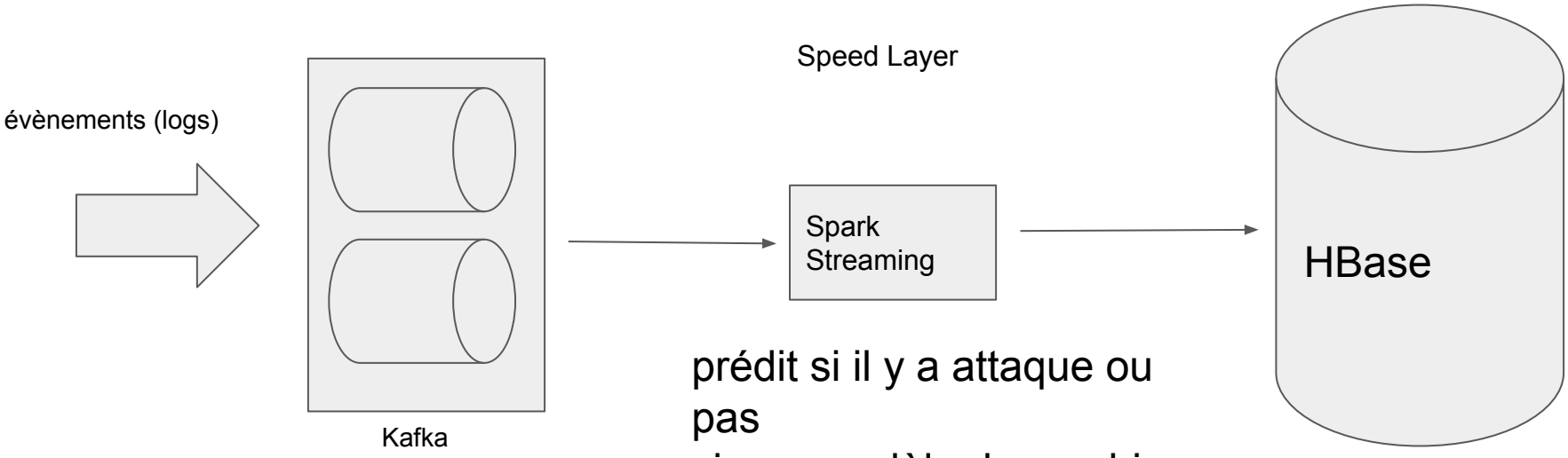
KAFKA - SPARK STREAMING

- Spark : framework de calcul distribué, utilisation BATCH
- Spark streaming : Spark lancé toutes les n secondes avec les N données récupérées. MICROBATCH
- Spark Streaming peut être utilisé pour consommer des messages dans Kafka. On peut tuner les performances en jouant sur le nombre d'executors et le nombre de partitions
- Dans une architecture spark streaming Kafka, on récupère tous les n temps un dataframe Spark de N messages de Kafka.



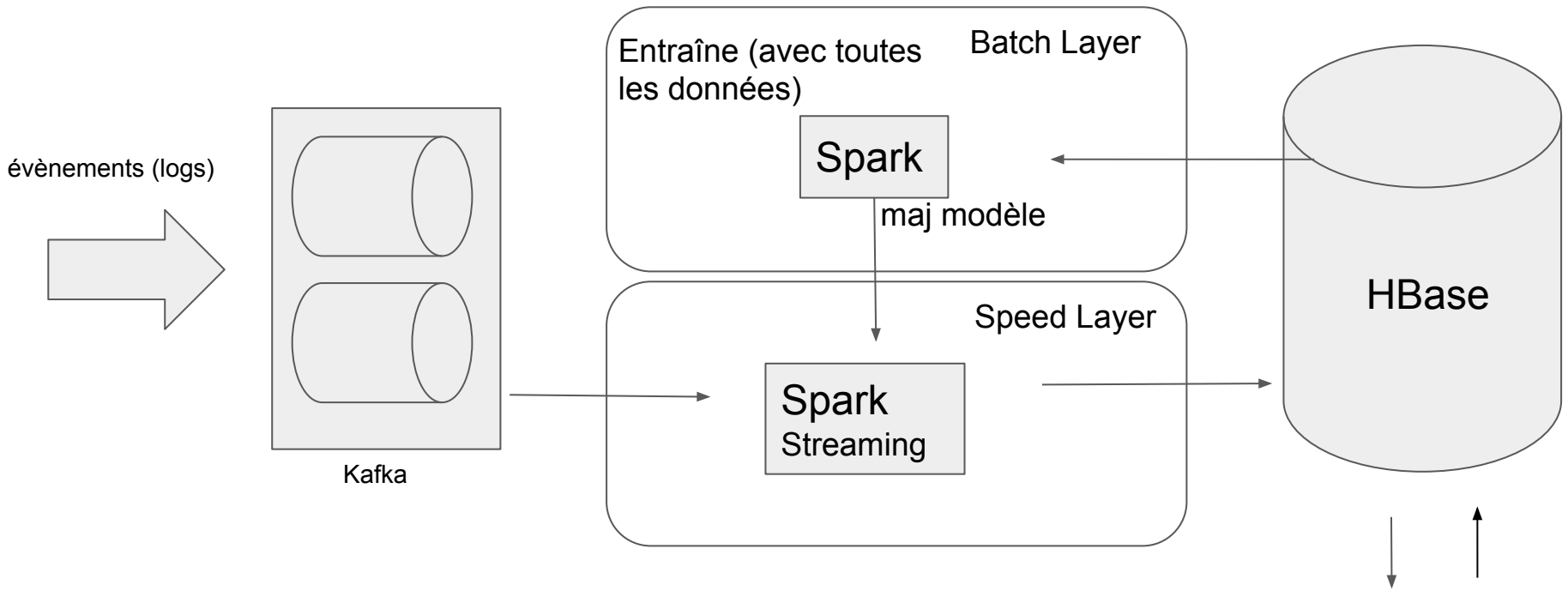
EXEMPLE DE GESTION DE FLUX

EXEMPLE : DÉTECTION D'ATTAQUES



prédit si il y a attaque ou pas
via un modèle de machine learning

DÉTECTION D'ATTAQUES : MISE À JOUR DU MODÈLE VIA LA LAMBDA ARCHITECTURE



présenté par Thalès au Hadoop Summit

APACHE NIFI

- Outil graphique de création de flux de données.

- Permet de se connecter à des sources de données diverses (APIs http, FS classiques, HDFS, BDD, etc…)
- S'interface avec des outils de l'écosystème hadoop, mais pas seulement. (HDFS, Hbase, Hive, mongodb, Amazon S3, Google Storage, etc)
- Peut être distribué sur un cluster (Pas a la même échelle que Hadoop)
- Permet de visualiser et contrôler un flux en temps réel.

APACHE NIFI : CONCEPTS CLÉS

- Flowfile:
  - Représentation d'une unité de donnée. Contient les données et des métadonnées associées, appelées attributs.

- Processor:
  - Représente une action, que ce soit de routage, de transformation, de lecture ou écriture. Agit sur les flowfiles.

- Connexions:
  - Représente les transitions entre les processors. Un processor peut avoir des connections en entrée et produit de connections en sortie. Chaque connexion en sortie doit être reliée à un autre processor, ou auto-terminées.

APACHE NIFI

- Exemple de flux