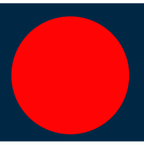
Batch : extraction depuis mariaDB

pour ingestion par
ElasticSearch

Tâches	**Statut**	Priorité
Insérer des données dans MariaDB		
Choix de la solution	en cours	
Connexion à Mariadb		
Liste des bases de données		
Liste des tables dans chaque bdd		
Accéder au schéma relationnel	Manuel	
Unifier les tables d’une base en un document	en cours	
Convertir les données récupérer en ndjson		

Problèmes décelés :

Prise en compte des relations présentes dans mariadb

Faire un transfert des tables indépendamment les unes des autres n’est pas suffisant et inutile

Besoin de définir un moyen de synchroniser le batch et le streaming

Pas de colonne de mise à jour donc il faudra que l’on récupère la clé primaire avant la mise en place du streaming

Solutions envisagées :

Extraire chaque table en dataframe, join avec pandas
Pb : la clause pour filtrer la donnée à extraire (nécessite un timestamp ou un champ à valeur unique et croissant) ;

Pour automatiser les pour join avec pandas les champs identiques doivent avoir le même nom

Synchroniser ElasticSearch avec logstash ou Kafka connect
Pb : susceptible de soumettre mariadb à des requêtes bien trop longues

Est ce que les clés primaires sont toujours des timestamp ou des à valeurs croissant
ie : il n’y a pas de clé primaires dont les valeurs sont générée aléatoirement

Est ce que au lancement du streaming il est possible d’accéder à la derniere clé en date

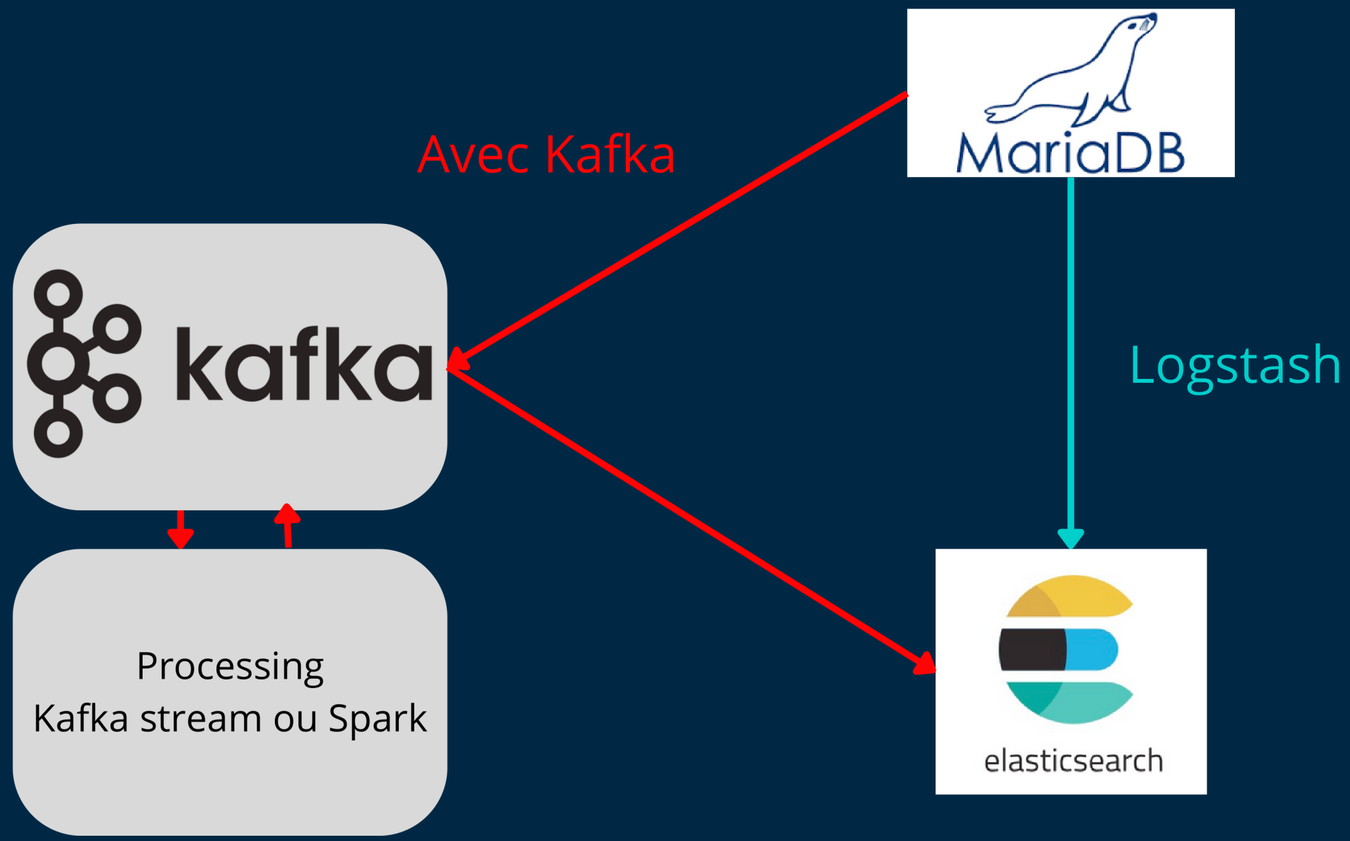
Passage par Logstash ou Kafka Connect

Spécifier la clé dans la clause WHERE

Lancer le streaming, puis récupérer manuellement la 1ère clé

Connecteurs JDBC

Problèmes \ Solutions	Script python	Logstash	Kafka Connect
Complexité configuration	+	++	+++
Synchronisation	**Contraintes sur les clés**	???	Error
Unifier les tables	manuel ou semi-automatisé	Surcharger mariadb	Surcharger mariadb ou Kafka Stream | Spark
Complexité finale	++	++	+++


Avec Kafka
MariaDB
kafka
Logstash
Processing
Kafka stream ou Spark
elasticsearch