Plan de Gouvernance de la Plateforme de Streaming Temps Réel

1. Introduction

Contexte du Projet

La plateforme de streaming temps réel repose sur une architecture distribuée comprenant Kafka pour la gestion des messages, PySpark pour le traitement des données, MongoDB et Cassandra pour le stockage, et Streamlit pour la visualisation.

Objectifs de la Gouvernance

L'objectif de ce plan est de garantir :

- La qualité, la sécurité et la disponibilité des données.
- Une gestion efficace des accès et des rôles.
- La scalabilité et la performance des composants.

---

2. Gestion des Données

Qualité des Données

- Validation des schémas à l'entrée des topics Kafka.
- Suppression des doublons et nettoyage des données avec PySpark.

Conservation des Données

- Kafka : Rétention configurable par topic (ex. 7 jours).
- MongoDB : Archivage automatique des documents obsolètes.
- Cassandra : Gestion TTL pour les lignes de données.

Sécurité des Données

- Kafka : Chiffrement TLS, authentification SASL.
- MongoDB : ACL et authentification par mot de passe.
- Cassandra : Chiffrement des données au repos.

---

3. Gestion des Accès et des Rôles

Contrôle des Accès

- Kafka : ACL pour restreindre la production et la consommation des messages.
- MongoDB : Rôles utilisateur pour limiter les droits.
- Streamlit : Authentification des utilisateurs via JWT.

Rôles et Responsabilités

- **Administrateur Kafka** : Gestion des clusters et des topics.
- **Data Engineer** : Maintenance et optimisation des jobs PySpark.
- **Analyste** : Visualisation des données via Streamlit.

---

4. Monitoring et Observabilité

Surveillance des Composants

- Kafka, Spark, MongoDB, Cassandra : Monitoring avec Prometheus et Grafana.
- Collecte des métriques de latence, débit, erreurs.

Alertes et Logs

- Kafka : Alerte sur les décalages de consommateur.
- Spark : Journalisation des échecs de tâches.

---

5. Scalabilité et Performance

Stratégies de Scalabilité

- Kafka : Augmentation du nombre de partitions.
- Spark : Ajustement dynamique des ressources.
- Cassandra : Ajout de nœuds pour la répartition de charge.

Optimisation des Performances

- Kafka : Compression des messages.
- MongoDB : Indexation des champs critiques.
- Cassandra : Choix optimisé des clés de partition.

---

6. Gestion des Changements

Pipeline CI/CD

- Déploiement automatisé des composants avec GitLab CI.
- Tests unitaires et d'intégration avant mise en production.

Documentation des Changements

- Maintien d'un journal des modifications.
- Validation des évolutions via des comités techniques.

---

7. Conformité et Audit

Audit des Accès

- Kafka, MongoDB, Cassandra : Logs d'accès et d'activité.
- Streamlit : Traçabilité des connexions.

Conformité RGPD

- Anonymisation des données sensibles.
- Possibilité de suppression des données utilisateur sur demande.

---

8. Conclusion

Ce plan de gouvernance établit les principes essentiels pour assurer la fiabilité, la sécurité et la performance de la plateforme de streaming temps réel. Grâce à des mécanismes rigoureux de gestion des accès, de surveillance et d'évolution, la plateforme peut évoluer sereinement tout en respectant les exigences métiers et réglementaires.