APACHE KAFKA

PLAN DE VEILLE TECHNOLOGIQUE

Thème : Informatique
Sous-thème : Middleware
Sous-sous-thème : Data Streaming
Technologie : Apache Kafka

TABLE DES MATIERES

I - STRATEGIE DE VEILLE

PRESENTATION DU CADRE

Le Data streaming est un pattern d'architecture visant à rendre disponible un flux de données **en temps réel** – soit au moment de leur publication – à un **cluster** de services consommateurs – et/ou producteurs – de cette donnée. Il répond à des problématiques architecturales de domaines variés – Cloud, Big Data, IOT, microservices…

Apache Kafka fut développé par Linkedin, qui confronté à la problématique de la **scalabilité** dans leurs systèmes d'échanges de données ont développé une solution interne rendue open source en 2011. En 2012 il est intégré par **l'Apache Fondation** et maintenu et développé depuis 2014 par **Confluent**.

Ses capacités de scalabilité verticale et horizontale, de garantie de transmission, de tolérance aux pannes, sa rejouabilité le rendant particulièrement polyvalent.

OBJECTIFS DE LA VEILLE

La veille technologique répond à ces objectifs stratégiques :

- Etablir un suivi de **documentation** génériques et spécifiques – à des versions – du produit
- Relever bugs, limites, failles, mauvais usages et autres points d'attention
- Gérer et mettre à jour une documentation des techniques **implémentales** de connexion à kafka, de consumer et producer en java Spring
- Gérer et mettre à jour une documentation de ses techniques de déploiement via **docker-compose** et **Kubernetes**.
- Tracer les recommandations d'environnement logiciel

OUTILS UTILISES

La veille automatisée se fait à l'aide de ces outils :

Feedly, **Google alertes** : application de suivi de flux RSS : permet l'automatisation du suivi et de la récolte
Awesome rss : extension de navigateur permettant le suivi de flux RSS
Zotero est utilisé en tant qu'outil de gestion et de partage de sources.

La recherche se fait manuellement via le moteur **Google**. Autres réseaux sociaux à surveiller : **Reddit**, **Twitter**

Les sites à surveiller particulièrement via ces outils dépendent du sujet de la recherche :

- Implémentation, mise en prod, bug : Stack Overflow, Github
- Rapports de bugs : Stack Overflow, Git Hub
- Documentation, suivi releases prioritaires : kafka.apache.org, confluent.io, doc.confluent.io
- Sites de presse : JDN, 01.net, Next Inpact

MOTS CLES DE RECHERCHE

Dans le cadre de la recherche par moteur, on privilégie la langue anglaise fournissant davantage de résultats. La liste ci-dessous est en anglais, les équivalents français sont également pertinents.

Mots clés génériques : Kafka, Confluent, Apache, "Data Streaming", "Event Streaming"

Type de source : Doc, News, Topic, Forum, Video

Type de contenu : Release, Download, Implementation, Consumer, Producer, Tools, Environnement, Issue, Bug, Report, Examples, Tutorial

Technologies associées : Docker, Kubernetes, Java, Zookeeper, KsqlDB, API, Asynchronous, Monitoring, Stream processing, ETL

II - WEBOGRAPHIE

DOCUMENTATION FONCTIONNELLE

- Lien vers la documentation officielle par version

https://kafka.apache.org/documentation.html

- Lien des téléchargement utile au suivi de release

https://kafka.apache.org/downloads

DOCUMENTATION TUTORALE

- Cours et exercices tutoraux produits par confluent sous forme de challenges pour appréhender kafka – utilise confluent cloud

https://www.100daysofcode.com/

- Cours et tutoriaux sur apache kafka

https://www.tutorialspoint.com/apache_kafka/index.htm

- Documentation plus avancée et complète

https://docs.confluent.io/home/overview.html

BUGS – FAILLES

- Confluent tend à supprimer la couche de dépendance de Kafka à Zookeeper, source de complexité de maintenance et produisant des problèmes de scalabilité.

https://www.confluent.io/blog/removing-zookeeper-dependency-in-kafka/

- Liste de bugs rapportés à Apache

https://issues.apache.org/jira/projects/KAFKA/issues/KAFKA-10410?filter=allopenissues

- Listing des failles de sécurité détectées – dont la faille log4j – des versions 1.X

https://kafka.apache.org/cve-list

DEVELOPPEMENT CONSUMER / PRODUCER

multi-langage :

- Exemples d'implémentations basiques de producer/consumer dans les principaux langages. Certains (Java, Python…) supportés par confluent, d'autres par la communauté (PHP, Perl…)

https://developer.confluent.io/get-started/java

Java Spring :

- Baeldung : respectivement :
  - comment configurer une connexion à Kafka
  - comment générer une connexion avec authentification ssl
  - Une comparaison entre deux modèles d'utilisation kafka : la séparation consumer/producer et ses deux APIs respectives, et leur abstraction via l'API Kafka Streams
  - Liste de doc d'implémentation

https://www.baeldung.com/kafka-docker-connection

https://www.baeldung.com/spring-boot-kafka-ssl

https://www.baeldung.com/java-kafka-streams-vs-kafka-consumer

https://www.baeldung.com/?s=kafka

DEPLOIEMENT

- Recommandations au déploiement

https://docs.confluent.io/3.1.1/kafka/deployment.html

Docker compose

- Kafka Zookeeperless (Kraft): version instable, déconseillée en prod

https://hellokube.dev/posts/three-ways-zookeepeerless-kafka/

- Environnements de tests et de démonstration

https://docs.confluent.io/platform/current/tutorials/build-your-own-demos.html

Déploiement Kubernetes :

- Déploiement avec Kubernetes, marche à suivre : Synergie entre Kafka et Kubernetes, deux outils scalables

https://www.magalix.com/blog/kafka-on-kubernetes-and-deploying-best-practice

- Déploiement Kafka – Cassandra : Synergie entre Kafka et Cassandra, BDD NoSQL axée performance

https://portworx.com/blog/how-to-combine-kafka-and-cassandra-on-kubernetes/

- Kafka + Kubernetes : intéressant pour les clusters Kafka petite et moyenne taille et faisable dans des environnements dépendants de Kubernetes mais source de complexité.

https://johanngyger.medium.com/kafka-on-kubernetes-a-good-fit-95251da55837

https://www.confluent.io/blog/apache-kafka-kubernetes-could-you-should-you/

Confluent cloud

- Solution Kafka native dans le cloud confluent

https://www.confluent.io/fr-fr/confluent-cloud/

ENVIRONNEMENT LOGICIEL

- Une documentation tutorale sur le monitoring Kafka

https://www.datadoghq.com/blog/monitoring-kafka-performance-metrics/#zookeeper-metrics

- Outils Kafka pour développeurs

https://www.confluent.io/blog/best-kafka-tools-that-boost-developer-productivity/

- Outils de testing

https://developer.confluent.io/learn/testing-kafka/

- Documentation KsqlDB, outil de stream processing

https://docs.ksqldb.io/en/latest/

- Documentation Kafka Connect

https://docs.confluent.io/platform/current/connect/index.html

EVÉNEMENTS EN LIGNE

Workshop / Online talks

Sur demande d'entrepreneur ou sur inscription, confluent organise des événements en ligne réguliers sur différents thèmes parcourus.

https://events.confluent.io/

https://www.confluent.io/fr-fr/resources/?assetType=online-talk&language=francais