

Document d'Architecture

Introduction

Ce document décrit l'architecture d'un système de collecte et d'analyse de données de crypto-monnaies en temps réel utilisant des technologies de conteneurisation, de streaming d'événements, et de traitement distribué. Le système se compose principalement des composants suivants : *Docker*, *Scrappy*, *Kafka*, *Apache Spark*, et une base de données.

Composants et Flux de Données

Flux

Le flux de données externe alimente le système avec des événements initiaux qui sont capturés par le Web Scraper.

Web Scraper “Producteur”

Description : Ce composant scrappe des sites web à des intervalles réguliers à l'aide de la librairie python *Scrappy* et envoie les données brutes (informations sur les crypto-monnaies) via des événements *Kafka*.

Fonctionnalités :

- Collecte de données à partir de sites web cibles.
- Envoi des données collectées sous forme d'événements Kafka.

Kafka

Description : Kafka agit comme un middleware de streaming d'événements, assurant la transmission des données du Web Scraper aux composants de traitement.

Fonctionnalités :

- Réception des événements de données du Web Scraper.
- Mise en file d'attente des événements pour traitement ultérieur par Apache Spark.

Apache Spark “Consommateur”

Description : Spark est utilisé pour le traitement distribué et l'analyse des données reçues de Kafka.

Composants :

- Master - Coordonne les tâches de traitement et distribue les charges de travail aux workers.
- Workers - Effectuent les tâches de traitement des données et renvoient les résultats au Master.

Fonctionnalités :

- Écoute des événements Kafka.
- Traitement des données en temps réel.
- Construction d'analyses à partir des données brutes.

Base de Données (PostgreSQL)

Description : La base de données stocke les données traitées, les informations chiffrées et les erreurs et logs.

Fonctionnalités :

- Stockage des résultats du traitement des données.
- Stockage des données chiffrées.
- Enregistrement des erreurs pour la surveillance et le dépannage.

Affichage

Description : Interface utilisateur pour visualiser les données traitées.

Fonctionnalités :

- Affichage en temps réel des données analysées.
- Accès aux données stockées dans la base de données.

TODO

Docker

Tous les composants du système, y compris **Scrappy**, **Kafka**, **Apache Spark**, et la base de données, sont déployés dans des conteneurs Docker pour une meilleure gestion des ressources et une isolation des processus.

Diagramme de l'Architecture

