Scalabilité horizontale dans les pipelines de données

Stage M2 en co-encadrement entreprise/laboratoire de recherche

- Date de début : février / mars 2022
- Durée : 6 mois
- Lieu : SAP France (Levallois-Perret) et LIP6 (Paris)
- Financement : environ 1400 euro/mois
- Co-encadrants : Hubert Naacke (hubert.naacke@lip6.fr) et Eric Simon (eric.simon@sap.com)

Contexte et Motivation

Aujourd'hui des entreprises gèrent leurs données avec une grande variété d'applications développées indépendamment. Or ces applications n'ont pas été conçues pour communiquer entre elles et il n'est pas envisagé de les migrer vers un système commun. Néanmoins, le besoin est fort de concevoir des nouveaux services de gestion et d'analyse de données qui valoriseront la donnée présente. Cela pose le problème de faire coopérer efficacement des applications, en particulier celles qui gèrent des grands volumes de données. Ainsi, pour faciliter la circulation de données entre les applications et définir des nouveaux services intégrant des données massives, des pipelines de données sont conçus.

Un *pipeline de données* [JHM04] est une séquence ou un graphe d'opérations sur des données. Une opération peut simplement déplacer des données ou effectuer des traitements complexes incluant la collecte de données de plusieurs sources, leur transformation, la génération de modèles par apprentissage et le stockage dans plusieurs destinations. En pratique, un pipeline peut contenir des centaines d'opérations et il peut évoluer à plusieurs reprises en étant complété avec des nouvelles opérations ou de nouvelles données. Ainsi, face au nombre croissant de pipelines à concevoir et déployer, il est crucial de disposer :

1. d'un langage de haut niveau pour définir des pipelines,
2. d'outils automatiques pour déployer et contrôler l'exécution d'un pipeline.

Les avancées technologiques récentes en matière de virtualisation et de conteneurisation telles que Kubernetes [Pou21] permettent de configurer, en langage Yaml, le déploiement d'un ensemble de tâches afin d'automatiser leur déploiement Toutefois, Yaml décrit les objets déployés (services, pod, conteneur) mais manque d'abstraction pour décrire des pipelines de données de manière suffisamment déclarative et extensible. C'est pourquoi, la société SAP a conçu un nouveau langage de définition de pipeline qui décrit l'enchainement des opérations (tâches) en spécifiant les échanges de données et la configuration de l'environnement d'exécution (techniques de virtualisation et conteneurisation).

Objectif et défis

L'objectif de ce stage est de proposer **une méthode pour instancier et déployer automatiquement et efficacement des pipelines de données**. Cela soulève plusieurs défis scientifiques et techniques :

Déploiement automatique : chaque opération du pipeline correspond à un programme (Python, node.JS, ...) ou à un appel vers une API externe (par exemple, job Spark) qui est déployé en utilisant une image/containeur Docker [Ber14] adaptée. A partir de la description d'un pipeline, il s'agit de le déployer sur une plateforme Kubernetes dans le cloud (par exemple Google Kubernetes Engine ou Elastic Kubernetes Service d'Amazon).

La parallélisation des opérations dans plusieurs pods permet d'augmenter la scalabilité horizontale du pipeline, mais nécessite également la définition d'*opérateurs de partitionnement de données* (par clé ou fenêtrage) pour répartir les traitements sur des partitions indépendantes.

Le regroupement de plusieurs opérations dans le même pod: ceci permet réduire les échanges de données entre pods qui sont remplacées par des communications moins couteuses entre les threads dans le même pod.

Travail à réaliser

Le ou la stagiaire abordera en priorité le défi de la parallélisation et traitera les points suivants :

1. Prise en main de l'environnement d'exécution. Etat de l'art sur les services Kubernetes, le déploiement automatique de pods et la génération d'images docker.
2. Parallélisation d'une opération : Implanter différents opérateurs de partitionnement adaptés à la distribution de données ordonnées (séquences, flux) et non-ordonnées (ensembles).
3. Traduire la spécification d'un pipeline, contenant des opérations parallélisées, en un déploiement yaml qui doit répliquer les pods s'exécutant en parallèle et contenir des nouveaux pods dédiés au partitionnement des données.
4. Définir des cas d'usage et conduire des expérimentations pour mesurer les performances obtenues.
5. Selon le temps disponible le ou la stagiaire pourra étudier l'optimisation du pipeline en combinant les stratégies de regroupement d'opérations dans un même pod et de parallélisation d'un pod. En particulier, ces stratégies sont limitées par un certain nombre de contraintes liées aux types d'images dockers disponibles et au protocole de communication entre pods. Il s'agit de proposer une méthode pour déterminer quelles sont les opérations à regrouper et quelles sont celles à paralléliser afin de maximiser les performances d'un pipeline.

Une perspective intéressante est de répondre aux besoins d'élasticité dans des scénarios sur des flux de données dynamiques (avec des changements de fréquence et des "bursts"). Il s'agit d'étudier l'implantation de *stratégies dynamiques* qui prennent un compte ces changements pour adapter le nombre de pods déployés aux besoins. Ceci est particulièrement important dans les déploiements sur des services cloud qui impliquent des coûts financiers.

Compétences attendues :

- bonne expérience en programmation (Python, Java)
- optimisation dans les bases de données, parallélisme de données, map-reduce
- connaissances techniques sur Docker/Kubernetes

Equipes d'accueil

- SAP France (Levallois-Perret)
- Equipe Bases de Données du LIP6 (Paris): `http://www-bd.lip6.fr/`

References

[Ber14] David Bernstein. Containers and cloud: From lxc to docker to kubernetes. *IEEE Cloud Computing*, 1(3):81–84, 2014.

[JHM04] Wesley M Johnston, JR Paul Hanna, and Richard J Millar. Advances in dataflow programming languages. *ACM computing surveys (CSUR)*, 36(1):1–34, 2004.

[Pou21] Nigel Poulton. *The Kubernetes Book*. Amazon, 2021.