Architectures des lacs de données et gestion des métadonnées

Pegdwendé N. Sawadogo, Jérôme Darmont

Université de Lyon, Lyon 2, UR ERIC
5 avenue Pierre Mendès France, F69676 Bron Cedex, France
{pegdwende.sawadogo,jerome.darmont}@univ-lyon2.fr

RÉSUMÉ. *Bien que le concept de lac de données ait émergé il y a une dizaine d'années, il n'existe pas à ce jour d'architecture ni de système de métadonnées standards. Dans cet article, nous dressons donc une synthèse des différentes approches de conception et de mise en œuvre d'un lac de données (Sawadogo, Darmont, 2021).*

ABSTRACT. *Although the data lake concept emerged about a decade ago, there are nowadays no standard architecture nor metadata system. Thus, in this paper, we propose a summary of the various approaches for designing and implementing data lakes (Sawadogo, Darmont, 2021).*

MOTS-CLÉS : *Lacs de données, architectures de lacs de données, gestion de métadonnées*

KEYWORDS: *Data lakes, data lake architectures, metadata management*

Au cours des dernières décennies, nous avons assisté à une croissance exponentielle de la quantité de données produite dans le monde. Ces mégadonnées, communément appelées *big data*, sont caractérisées par des problématiques de volume, de vélocité et d'hétérogénéité qui surpassent les capacités des systèmes traditionnels pour leur stockage et leur analyse. Pour y remédier, Dixon (2010) a introduit le concept de lac de données (LdD). Cependant, ce dernier demeure parfois flou, notamment en ce qui concerne sa mise en œuvre. C'est pourquoi nous proposons dans cet article une analyse des principales approches de conception d'un LdD. Nous nous intéressons en particulier aux architectures de LdD et à la gestion des métadonnées, qui constituent des composantes fondamentales dans la conception d'un LdD.

Nous définissons un LdD comme un système évolutif de stockage et d'analyse de données de tous types, dans leur format natif, utilisé *principalement* par des spécialistes des données (*data scientists*, *data analysts*) pour l'extraction de connaissances (Sawadogo *et al*., 2019). Bien qu'étant générique, cette définition n'est toutefois pas la seule dans la littérature. En effet, le concept de LdD est toujours en maturation et il existe différentes façons de le percevoir et de le concevoir.

Ainsi, du point de vue architectural, nous proposons une typologie mettant en exergue deux principales approches de conception. D'une part, les architectures fonctionnelles subdivisent le LdD en couches qui reflètent des fonctions de base : couche

d'ingestion, couche de stockage, couche de traitement, couche d'accès, etc. L'architecture Lambda fait partie de cette catégorie. Elle propose en effet une organisation suivant des couches de stockage et d'accès uniques, ainsi qu'une double couche de traitements, selon la nature des données (en flux ou massives). D'autre part, les architectures basées sur la maturité sont organisées suivant le niveau de raffinement des données. Dans ces architectures, les couches sont communément appelées "zones" et peuvent inclure, par exemple : une zone de transit, une zone de stockage de données brutes, une zone de stockage de données raffinées, une zone de stockage de données fiables, une zone de consommation des données, etc. C'est typiquement le cas du LdD de LaPlante et Sharma (2016). Les approches fonctionnelle et basée sur la maturité proposent chacune une vision pertinente, mais partielle de l'architecture du LdD. Une architecture hybride, spécifiant à la fois les composants fonctionnels et les étapes de maturation des données paraît donc nécessaire pour une meilleure conception du LdD.

À l'image des architectures, plusieurs approches d'organisation des métadonnées des LdD coexistent, principalement basées sur des graphes. Une première approche par graphes se focalise sur le traçage de la provenance des données. Les données sont représentées par des nœuds et les traitements par des arêtes. De cette façon, il est possible de conserver des informations telle que l'utilisateur ou même le script ayant participé à la création de nouvelles données dans le LdD. On peut ainsi assurer la reproductibilité des traitements dans le LdD. Une autre approche d'organisation des métadonnées trace les relations entre données. Par exemple, de telles relations peuvent être des mesures de similarité pour des documents textuels ou des liens de type clé primaire-clé étrangère entre deux tables relationnelles. Elles servent plus tard à la réalisation de tâches telles que la recommandation de données connexes. La modélisation par graphes est particulièrement adaptée au contexte des LdD, car elle offre la flexibilité indispensable à la gestion de données hétérogènes. Cependant, le choix d'un modèle par graphes plutôt qu'un autre n'est pas aisé. Pour guider le choix de l'approche la plus appropriée à chaque cas d'usage, nous proposons un ensemble de six fonctionnalités permettant de comparer les modèles de métadonnées entre eux. Nous montrons ainsi qu'aucune des approches de la littérature n'intègre l'ensemble des métadonnées possibles dans un LdD. De nouveaux modèles de métadonnées plus complets sont donc nécessaires.

Bibliographie

Dixon J. (2010). *Pentaho, Hadoop, and Data Lakes.* Consulté sur https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/

LaPlante A., Sharma B. (2016). *Architecting Data Lakes*. O'Reilly.

Sawadogo P. N., Darmont J. (2021). On data lake architectures and metadata management. *Journal of Intelligent Information Systems*, vol. 56, n° 1, p. 97-120.

Sawadogo P. N., Scholly E., Favre C., Ferey É., Loudcher S., Darmont J. (2019). Metadata Systems for Data Lakes: Models and Features. In *1st International Workshop on BI and Big Data Applications (BBIGAP@ADBIS 2019), Bled, Slovenia. Communications in Computer and Information Science*, vol. 1064, p. 440-451. Springer.