Construisez vos pipelines de données avec Databricks

Prenez en main votre espace de travail Databricks

Correction

Commençons par l'environnement. Vous disposez de la capture annotée de la section « Repérez les grandes zones de l'interface » : si les cinq zones que vous avez listées correspondent aux fonctions annotées, vous êtes au bon endroit. Un environnement opérationnel se reconnaît à trois signes : le warehouse passe en état démarré quand vous lancez une requête ; la cellule du notebook renvoie un résultat ; l'éditeur SQL renvoie le même résultat. Trois signes, pas un de plus.

Si l'un des trois manque, voici les trois causes que je vois le plus souvent :

1. **La page d'inscription vous a demandé un fournisseur cloud ou un paiement.** Vérifiez que vous avez bien choisi la Free Edition et non l'essai gratuit. Revenez à la section « Créez votre espace de travail en Free Edition » et reprenez à l'étape 1.
2. **La requête semble ne jamais finir.** Le warehouse serverless démarre à la première requête, et ce démarrage prend quelques dizaines de secondes. Ouvrez la zone SQL Warehouses, regardez son état, et attendez qu'il soit démarré avant de conclure à une panne.
3. **Le notebook renvoie une erreur alors que l'éditeur SQL fonctionne.** Le notebook n'a pas de ressource de calcul sélectionnée. Dans le sélecteur en haut du notebook, choisissez le compute serverless proposé par défaut, puis relancez la cellule.

Passons aux trois situations. Ce qui compte ici, c'est le raisonnement.

Situation A : éditeur SQL. Le résultat ne vous est pas destiné, il part dans un dashboard. La requête du top 10 doit donc être enregistrée, pour être branchée telle quelle au chapitre P2C1. Choisir le notebook n'est pas faux au sens strict, mais vous recopierez la requête à la main le jour venu, et vous introduirez une divergence entre ce que vous avez validé et ce que le dashboard exécute.

Situation B : notebook. Vous ne savez pas encore quelle question poser au fichier : vous cherchez. Les essais de jointures, les surprises, les notes pour votre successeur, tout cela est une démarche à documenter, et le notebook est fait pour ça. Dans l'éditeur SQL, il vous resterait une poignée de requêtes sans le raisonnement qui les relie.

Situation C : les deux se défendent. C'est un contrôle de table, ce qui plaide pour l'éditeur SQL : trois requêtes enregistrées, relancées chaque lundi, résultat en tableau. Mais vous voulez garder la trace de ce que vous avez vérifié et pourquoi, ce qui plaide pour le notebook. Les deux réponses sont acceptées si elles sont justifiées. Le critère qui fait pencher : qui lit le résultat ? Si c'est vous, un lundi sur deux, avec un commentaire à côté de chaque chiffre, le notebook l'emporte. Si ces contrôles ont vocation à alimenter une alerte ou un indicateur de qualité dans un dashboard, l'éditeur SQL l'emporte, parce que la requête enregistrée est réutilisable et le notebook ne l'est pas. Un bon rendu nomme ce critère, quelle que soit la réponse retenue.