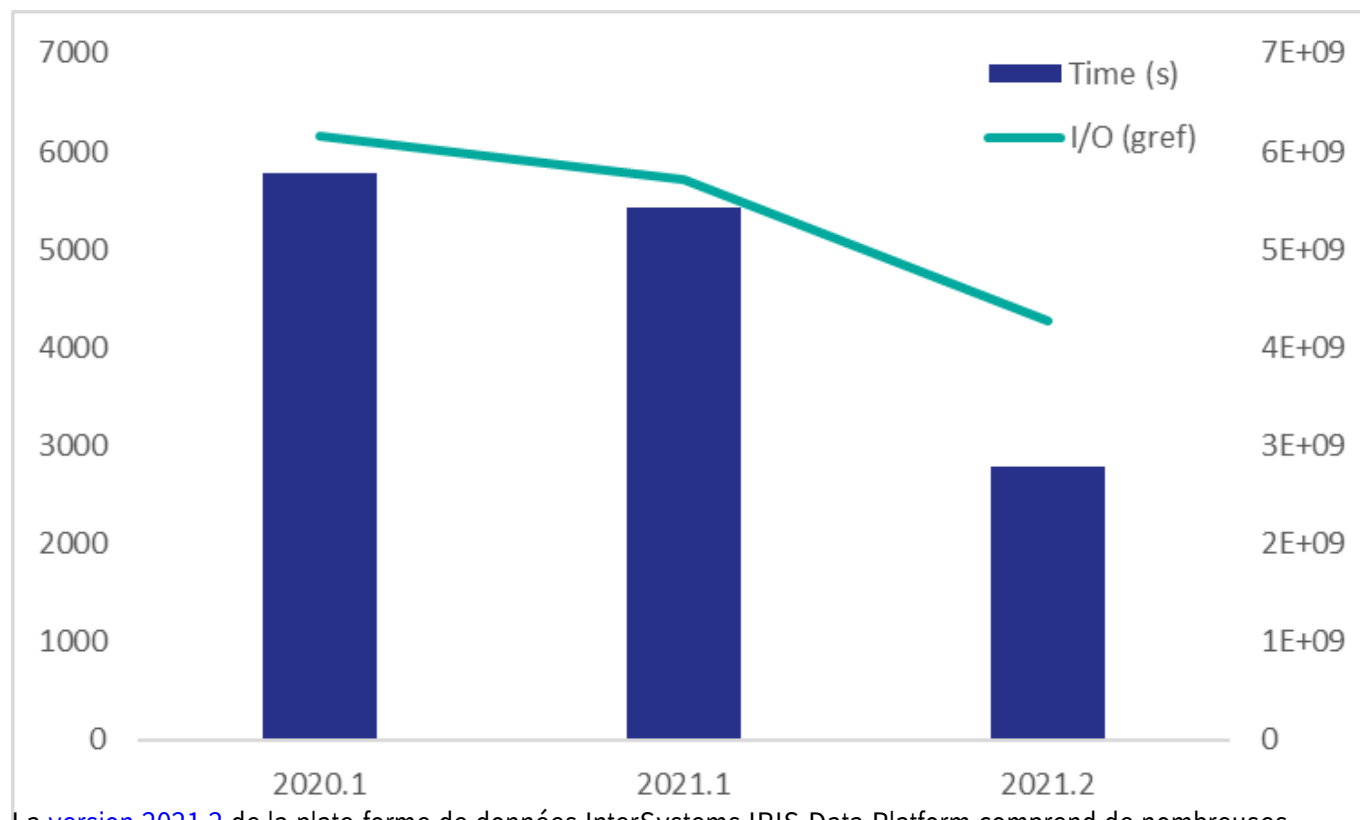


Article

[Guillaume Rongier](#) · Mai 2, 2022 4m de lecture

## 2021.2 Fonctionnalité SQL en vedette - Choix du plan d'exécution



La [version 2021.2](#) de la plate-forme de données InterSystems IRIS Data Platform comprend de nombreuses nouvelles fonctionnalités intéressantes pour le développement rapide, flexible et sécurisé de vos applications critiques. Embedded Python est certainement la vedette (et pour une bonne raison !), mais en SQL, nous avons également fait un grand pas en avant vers un moteur plus adaptatif qui recueille des informations statistiques détaillées sur les données de votre tableau et les exploite pour fournir les meilleurs plans de requête. Dans cette brève série d'articles, nous allons examiner de plus près trois éléments qui sont nouveaux dans 2021.2 et qui travaillent ensemble vers cet objectif, en commençant par **Run Time Plan Choice**.

Il est difficile de trouver le bon ordre pour en parler (vous ne pouvez pas imaginer le nombre de fois où je les ai remaniés en rédigeant cet article), car ils s'emboîtent si bien les uns dans les autres. Vous pouvez donc les lire dans un ordre aléatoire 😊

### Au sujet du traitement des requêtes IRIS

Lorsque vous soumettez une instruction au moteur IRIS SQL, celui-ci l'analyse dans une forme normalisée, en substituant tous les littéraux (paramètres de la requête), puis examine la structure de votre tableau, les indices et les statistiques sur les valeurs des champs afin de déterminer la stratégie d'exécution la plus efficace pour la requête normalisée. Cela permet au moteur de réutiliser le même plan et le même code généré lorsque vous souhaitez exécuter à nouveau la requête, éventuellement en utilisant des valeurs de paramètres de requête différentes. Par exemple, prenez la requête suivante :

```
SELECT * FROM Customer WHERE CountryCode = 'US' AND ChannelCode = 'DIRECT'
```

Cette requête sera normalisée sous une forme analogue à celle-ci :

```
SELECT * FROM Customer WHERE CountryCode = ? AND ChannelCode = ?
```

afin que les invocations ultérieures pour différentes combinaisons de pays et de canaux puissent immédiatement récupérer la même classe de requête mise en cache, ce qui permet d'éviter le travail de planification lourd en termes de calcul. Supposons que nous vendions des chaussures de course par l'intermédiaire d'une demi-douzaine de canaux et que nos produits soient vendus dans le monde entier ; en d'autres termes, les données sont réparties uniformément entre les valeurs possibles des champs CountryCode et ChannelCode. Si nous disposons d'un index régulier sur ces deux champs, le plan le plus efficace pour cette requête normalisée commencera par la condition la plus sélective (CountryCode), en utilisant l'indice correspondant pour accéder aux lignes correspondantes de la carte principale, puis vérifiera l'autre condition (ChannelCode) pour chaque ligne de la carte principale.

### Valeurs aberrantes

Supposons que nous ne soyons pas un fabricant d'équipements sportifs, mais un vendeur spécialisé dans les moules pour chocolats belges (d'où est-ce que je sors ça ? ).

Dans ce cas, supposons que la majorité de mes clients (disons 60 %) se trouvent en Belgique, ce qui signifie que "BE" devient une valeur "aberrante" pour le champ CountryCode, représentant un grand nombre de lignes de mon tableau. Soudain, l'indice sur le code pays a une autre utilité : si le paramètre de requête pour CountryCode que nous obtenons au moment de l'exécution est 'BE', l'utilisation de cet index en premier signifierait que je devrais lire la majorité de ma carte principale, et qu'il serait préférable de commencer par l'index sur ChannelCode. Cependant, si la valeur du paramètre de requête pour CountryCode est une autre valeur, cela rendrait l'indice sur CountryCode beaucoup plus intéressant, car tous les autres pays se partagent les 40% de clients non belges restants.

C'est un exemple où vous voudriez choisir un plan différent au moment de l'exécution ; ou, en reformulant ces mots : **\*\*Run Time Plan Choice\*\***. Le RTPC est un mécanisme qui ajoute un petit crochet dans la logique classique de substitution littérale et de recherche de requête en mémoire cache pour repérer les valeurs aberrantes telles que la valeur "BE" pour notre colonne CountryCode. Si vous souhaitez obtenir un aperçu plus détaillé du traitement des requêtes IRIS SQL, veuillez consulter [ce vidéo VS2020](<https://learning.intersystems.com/course/view.php?id=1598>).

Dans le passé, IRIS SQL supportait une version opt-in très rudimentaire de ce contrôle, mais la version 2021.2 introduit une toute nouvelle infrastructure RTPC, beaucoup plus légère et capable d'intervenir pour une plus grande variété de conditions de prédicat. Ayant établi que les frais généraux de cette vérification d'exécution sont effectivement minimes, nous avons décidé de l'activer par défaut, de sorte que vous n'avez rien à faire pour en bénéficier (à part [unfreeze vos plans de requête après la mise à niveau](<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls...>) comme d'habitude).

Nous avons souvent constaté à quel point les valeurs aberrantes sont fréquentes dans les ensembles de données du monde réel (et à quel point une distribution strictement uniforme est rare) et les tests sur un benchmark partenaire ont montré une amélioration spectaculaire des performances et des I/O, comme vous pouvez le voir dans le graphique ci-dessous. Nous avons également inclus les résultats du benchmark pour la version 2020.1, afin que vous puissiez apprécier nos efforts continus (et les résultats !) pour améliorer les performances au fil des versions.

Le temps de débit varie en fonction de la quantité de valeurs aberrantes dans votre ensemble de données et de la disponibilité des indices, mais nous sommes très enthousiastes quant au potentiel de ce changement et nous sommes très curieux de connaître vos expériences.

[#SQL](#) [#Tables relationnelles](#) [#InterSystems IRIS](#)

## 2021.2 Fonctionnalité SQL en vedette - Choix du plan d'exécution

Published on InterSystems Developer Community (<https://community.intersystems.com>)

---

**source:**<https://fr.community.intersystems.com/post/20212-fonctionnalit%C3%A9-sql-en-vedette-choix-du-plan-dex%C3%A9cution>