記事
Toshihiko Minamoto · 2022年2月24日 5m read

2021.2 SQL 機能スポットライト - ランタイムプランの選択

InterSystems IRIS Data Platform の 2021.2 リリース
には、ミッションクリティカルなアプリケーションを高速で柔軟性に優れ、セキュアに開発するための刺激的な新機能が多数含まれています。 Embedded Python は間違いなく脚光を浴びています（正当な理由で！）が、SQL の分野でも、テーブルデータに関する詳細な統計情報を収集し、それを最適なクエリプランに提供する、より適応性の高いエンジンに向けて大きな一歩を踏み出しました。 この短い連載記事では、2021.2 で新しく追加された 3 つの要素について詳しく説明し、**ランタイムプランの選択(RTPC)**を手始めに、この目標に向かって進みます。

これらについて適切な順序で話していくのは困難です（この記事を書く上で、私がどれだけ順序を入れ替えたか想像できないほどです！） というのも、これらが相互に非常にうまく機能するためです。
そのため、ご自由に順序を変えてお読みください 。

IRIS クエリ処理について

IRIS SQL エンジンにステートメントを送信すると、エンジンはリテラル（クエリパラメーター）を置き換えてそのステートメントを正規化された形式に解析し、テーブル構造、インデックス、およびフィールド値に関する統計を見て、正規化されたクエリにとって最も効率的な実行ストラテジーをはじき出します。 これにより、おそらく異なるクエリパラメーター値を使用してクエリをもう一度実行したい場合に、エンジンは同じプランと生成されたコードを再利用することができます。 たとえば、以下のクエリを例にします。

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

このクエリは以下のような形式に正規化されます。

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

そのため、国とチャンネルの様々な組み合わせに対する後続の呼び出しは、即座にキャッシュされた同じクエリクラスを取得するため、計算量の多いプランニング作業を省略することができます。 6 つのチャンネルを通じて世界中にランニングシューズを販売すると仮定しましょう。言い換えると、データは CountryCode と ChannelCode の可能な値全体で均等に分散されます。 これらの両方のフィールドに通常のインデックスがある場合、この正規化されたクエリで最も効率的なプランは、対応するインデックスを使用してマスターマップから一致する行にアクセスすることで最も選択的な条件（CountryCode）から始め、その後にマスターマップのすべての行に対し、別の条件（ChannelCode）をチェックします。

外れ値

では、スポーツ用品メーカーではなく、ベルギーチョコレートの型を販売する専門ベンダーだとしましょう（どこから出てきた例でしょうかね ）。 この場合、顧客の大半（たとえば 60%）がベルギーにいるとします。つまり、「BE」が CountryCode フィールドの*外れ値*であり、テーブルの多数の行を表します。 すると突然、CountryCode でのインデックスに他の使用方法が現れました。ランタイムで取得する CountryCode のクエリパラメーターが「BE」*であるとした場合*、そのインデックスを最初に使用すると、マスターマップの大半を読み取ることになるため、ChannelCode

でのインデックスから始める方がよくなります。 しかし、CountryCode
のクエリパラメーター値が別の値で*あるとした場合*、他のすべての国が残りの 40%
のベルギー以外の顧客を分割するため、CountryCode でのインデックスの方がはるかに高い価値が出てきます。

これは、ランタイム時に
異なるプランを選択する場合の例です。または、言い換えると、**ランタイムプランの選択**（RTPC）と言えます。
RTPC は、従来のリテラル置換とキャッシュ済みのクエリルックアッププロジックに小さなフックを追加して、Cou
nryCode 列の「BE」値などの外れ値を見つけるメカニズムです。 IRIS SQL
クエリ処理のさらに詳細な概要に興味のある方は、こちらの VS2020 動画をご覧ください。

IRIS SQL は過去に、非常に大雑把なオプトインバージョンをサポートしていましたが、2021.2
で、大幅に軽量化され、より広範な述語条件に対応できる、まったく新しい RTPC
インフラストラクチャを導入しました。 このランタイムチェックのオーバーヘッドは確かに最小限であるため、
ユーザーが何も行わなくてもこのメリットを得られるように、これをデフォルトでオンにしました（いつものよう
に、アップグレード後にクエリプランを解凍する必要はありません）。

これまで頻繁に、実世界のデータセットでは外れ値がどれほど一般的であるか（また、厳格に一様な分布がどれほ
ど稀であるか）を見てきましし、パートナーのベンチマークでのテストによって、以下のグラフからわかるように
、パフォーマンスと I/O で目を見開くほどの改善が得られることがわかりました。 また、2020.1 のベンチマーク
結果も含めているため、リリースのたびにパフォーマンスを改善できるように継続的に取り組んでいること（と結
果）を見ていただけるでしょう。

改善のマイレージは、データセットの外れ値の量とインデックスの可用性によって異なりますが、この変更の可能
性には非常に興奮しており、皆さんの体験をお聞かせいただければ幸いです。

#SQL #リレーショナルテーブル #InterSystems IRIS

ソースURL:
https://jp.community.intersystems.com/post/20212-sql-%E6%A9%9F%E8%83%BD%E3%82%B9%E3%83%9D%E3%83%83%E3%83%88%E3%83%A9%E3%82%A4%E3%83%88-%E3%83%A9%E3%83%B3%E3%82%BF%E3%82%A4%E3%83%A0%E3%83%97%E3%83%A9%E3%83%B3%E3%81%AE%E9%81%B8%E6%8A%9E