

MySQLクエリチ ューニング入門

2017/01/12 オプト社内勉強会

自己紹介

渋谷 充宏 @m4buya

- なんでも屋
- Ruby / Scala / インフラ(New!)
- <https://github.com/mshibuya>
- RailsAdmin/CarrierWave
committer

ねらい

MySQLのクエリチューニングがなんとなくできるようになる

- 前回の「EXPLAINの読み方入門」も見てね
 - https://opttechnologies.slack.com/files/U09ULLWRF/F6J6BB068/explain_explained.pdf

アジェンダ

- クエリチューニングにあたって
- 見るデータ量の最適化
- ケーススタディ
- まとめ

クエリチューニングにあたって

まずクエリしないで済ませられないか考える！

- 例：ページネーション

クエリチューニングにあたって

- 基本方針は、扱うデータ量を減らすこと

見るデータ量の最適化

- データ量によって全く異なる振る舞いになるので、チューニングは必ず本番相当のデータセットで行う！

見るデータ量の最適化

- 読み取るデータ量
- ソートするデータ量
- 返すデータ量

読み取るデータ量を減らす

- 減らすには...MySQLの気持ちになる

EXPLAINをじっくり眺めて、どんな流れでデータを読みに行っているのか理解しよう

読み取るデータ量を減らす

- indexはカーディナリティが命
選択性の高くないカラムにイン
デックスを貼っても、あんまり
意味がない。。

カーディナリティ

- 選択性。ユニークさみたいな？
- 高いほど絞り込みが効きやすいということ
- 例：氏名 vs 性別

読み取るデータ量を減らす

- indexはなるべくORDERに使う
 - たいていのWebアプリにはページネーションがある。
 - LIMIT句を活用することで、読み取りを途中で止められるように

読み取るデータ量を減らす

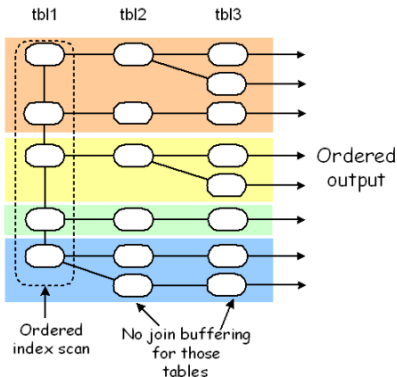
- 相関サブクエリは遅い傾向がある

ソートするデータ量を減らす

EXPLAINの時のこれを思い出して！

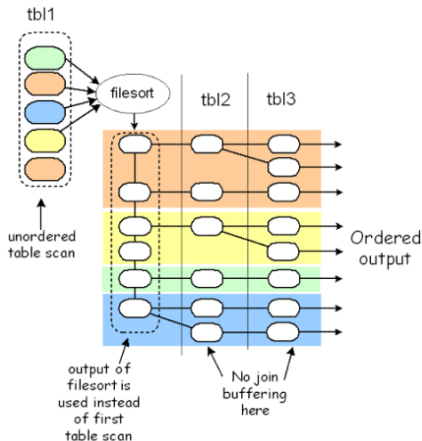
ソートするデータ量を減らす

Using temporary/Using filesortがどちらも出ない時



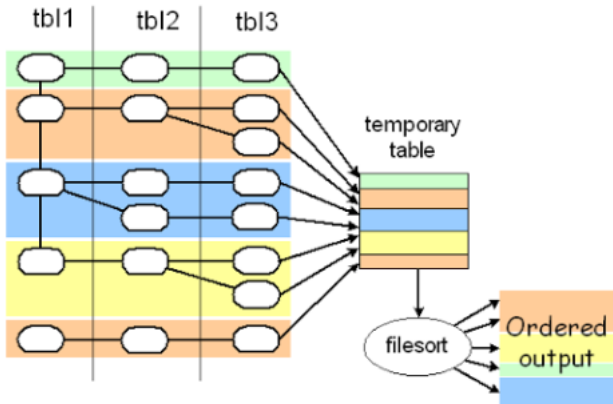
ソートするデータ量を減らす

Using filesortの時



ソートするデータ量を減らす

Using temporary; Using filesortの時



ソートするデータ量を減らす

ORDER句の列が複数テーブルにまたがっている等、クエリチューニングだけではUsing temporaryを回避できない局面がある。その場合は非正規化も検討に入ってくる

返すデータ量を減らす

でかいと当然時間がかかるので、
普通に減らそう

- ネットワーク転送にかかる時間
- アプリ側でデータを処理する時間

不要な列を減らすのが効果的なことも

ケーススタディ

GitHub opt-techからクエリチューニングっぽい事例をあつめてみました

不要なカラムを取得していたパターン

[https://github.com/opt-tech/
report_factory/pull/52/files](https://github.com/opt-tech/report_factory/pull/52/files)

ソートするデータ量を減らす パターン

<https://github.com/opt-tech/v7-apps/pull/4050/files>

サブクエリが遅いパターン

<https://github.com/opt-tech/new-adpoke-src/pull/150/files>

不要なJOIN/GROUP BYをしていたパターン

[https://github.com/opt-tech/new-adpoke-src/
commit/6a2ebf3e6bc3eb49c62317
ba91842db8bb231286](https://github.com/opt-tech/new-adpoke-src/commit/6a2ebf3e6bc3eb49c62317ba91842db8bb231286)

サブクエリが遅いパターン2

[https://github.com/opt-tech/v7-apps/
commit/0306772e448a294a5c5dce
85b268f8a07390838d](https://github.com/opt-tech/v7-apps/commit/0306772e448a294a5c5dce85b268f8a07390838d)

まとめ

- 扱うデータ量を減らせば速くなる
 - そのためには、まずクエリ結果を作るために何をしているのか把握する
 - 見るデータ / ソートするデータ / 返すデータ
- クエリチューニングはたのしいので、がんばってね