

1. Normales Select: Successfully run. Total query runtime: 197 msec. 599 rows affected.
2. CREAT INDEX select: CREATE INDEX Query returned successfully in 43 msec.
3. Normales Select danach: Successfully run. Total query runtime: 239 msec. 599 rows affected.

Möglicherweise entscheidet PostgreSQL einen Index auf den „last_name“ für Abfrageoptimierung zu ignorieren. Das ist abhängig von spezifischen Eigenschaften der Datenbank und der physischen Organisation der Daten.

last_name ist hier als erstes indiziert.

Bei nochmaliger Pagila-Datenbank Ausführung ignoriert PostgreSQL diesen Index. Der Optimierer von PostgreSQL erkennt das Verwendung des Indexes sinnlos ist, weil:

1. Datenmengen in Pagila-Datenbank als klein bedachtet werden, sind sequential scans „günstig“. Es müssen also nicht viele Daten gelesen werden und die Operationen haben vom Rechenaufwand am RAM-Platz.
2. Der hier aufgezwungenen „index only scans“ ist aufgrund spezifischer Eigenschaften der Datenbank und der physischen Organisation der Daten mit Vorsicht zu genießen. Das effiziente Lesen von Daten nur über den Index nicht immer vorteilhaft ist. Es kann dazu kommen, dass trotz des Indexes doch die eigentlichen Tabellendaten gelesen werden müssen. Davor muss PostgreSQL in der sogenannten "visibility map" nachsehen ob mit diesem zusätzliche (sinnlose) Operation notwendig sind.