|  |
| --- |
| Санкт-петербургский Академический университет |
| Лабораторная работа №1 |
|  |
|  |
| **Бандурин Д.В. Группа 504 (SE)** |
| **28.11.2011** |

|  |
| --- |
|  |

# Запросы

В лабораторной работе выполнялись следующие запросы:

-- 7.4 Give the player number and the address of each player resident in Stratford

**SELECT** PlayerNo**,** Street **+** ' ' **+** HouseNo **AS** Address

**FROM** Players

**WHERE** Town **=** 'Stratford'**;**

-- 8.1 Get the player number and the league number of each player resident in Stratford; order the result by league number.

**SELECT** PlayerNo**,** LeagueNo

**FROM** Players

**WHERE** Town **=** 'Stratford'

**ORDER** **BY** LeagueNo**;**

-- 10.2 Give the player number, the year of birth and the year of joining the club for each player who joined 17 years after he or she was born.

**SELECT** PlayerNo**,** BirthYear

**FROM** Players

**WHERE** BirthYear **IN** **(**1962**,** 1963**,** 1970**);**

-- 10.18 Get the player number and the name of each player who has at least one match for the first team.

**SELECT** PlayerNo**,** Name

**FROM** Players

**WHERE** PlayerNo **IN**

**(SELECT** PlayerNo

**FROM** Matches

**WHERE** TeamNo **=** 1**);**

-- 10.23 Get the player numbers, names and years of birth of players who are not amongst the oldest.

**SELECT** PlayerNo**,** Name**,** BirthYear

**FROM** Players

**WHERE** BirthYear **>** **ANY**

**(SELECT** BirthYear

**FROM** Players**);**

-- 15.3 Get the player number of each player who has incurred more than one penalty.

**SELECT** **DISTINCT** PlayerNo

**FROM** Penalties PN

**WHERE** **EXISTS**

**(SELECT** **\***

**FROM** Penalties

**WHERE** PN**.**PlayerNo **=** PlayerNo

**AND** PaymentNo **<>** PN**.**PaymentNo **);**

-- 15.9 Give the player number of each player who has played for the same teams as player 57.

**SELECT** PlayerNo

**FROM** Players P

**WHERE** **NOT** **EXISTS**

**(SELECT** **\***

**FROM** Matches M1

**WHERE** PlayerNo **=** 57

**AND** **NOT** **EXISTS**

**(SELECT** **\***

**FROM** Matches M2

**WHERE** M1**.**TeamNo **=** M2**.**TeamNo

**AND** P**.**PlayerNo **=** M2**.**PlayerNo**))**

**AND** PlayerNo **NOT** **IN**

**(SELECT** PlayerNo

**FROM** Matches

**WHERE** TeamNo **IN**

**(SELECT** TeamNo

**FROM** Teams

**WHERE** TeamNo **NOT** **IN**

**(SELECT** TeamNo

**FROM** Matches

**WHERE** PlayerNo **=** 57**)));**

# Физические операции

Сводная таблица по физическим операциям

|  |  |
| --- | --- |
| SQL № | Физические операции |
| 7.4 | Project,Filter,TableScan |
| 8.1 | Sort,Project,Filter,TableScan |
| 10.12 | Project,Filter,TableScan |
| 10.18 | Project,Filter,TableScan |
| 10.23 | Project,Filter,TableScan |
| 15.3 | Distinct,Sort,Project,Filter,TableScan |
| 15.9 | Project,Filter,TableScan |

# Время выполнения запроса

Сводные данные по времени выполнения запросов в зависимости от размера базы данных.

“-“ - получить время выполнения не удалось

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SQL № | Время выполнения. | | | |
| 1 | 2 | 3 | 4 |
| 7.4 | 21 | 83 | 410 | 1740 |
| 8.1 | 55 | 90 | 402 | 1627 |
| 10.12 | 44 | 24 | 392 | 1578 |
| 10.18 | 48 | 72 | 12919 | - |
| 10.23 | 61 | 122 | - | - |
| 15.3 | 94 | 641 | - | - |
| 15.9 | 200 | 627 | - | - |

На основе данных строим график

# Время генерации плана

Сводные данные по времени генерации плана в зависимости от размера плана и количества подпланов.

|  |  |  |  |
| --- | --- | --- | --- |
| SQL № | Размер плана | Количество подпланов | Среднее время генерации (мс) |
| 7.4 | 3 | 1 | 109,25 |
| 8.1 | 4 | 1 | 145,25 |
| 10.12 | 3 | 1 | 91 |
| 10.18 | 4 | 2 | 206 |
| 10.23 | 4 | 2 | 99 |
| 15.3 | 6 | 2 | 346 |
| 15.9 | 8 | 6 | 486 |

# Выводы и рекомендации

Из графиков видно, что время генерации плана напрямую зависит от количества подпланов и его длины. Но слабо зависит от размера базы данных, хотя наблюдается тенденция к увеличению времени работы оптимизатора. При маленьком количестве данных в БД время генерации сопоставимо со временем выполнения плана запроса. При увеличении количества данных время рост времени генерации по сравнению с ростом времени выполнения плана становится не существенным.

Для оптимизации данных запросов необходимо минимизировать количество физических операций TableScan. Это можно добиться путем переписывания запросов с исключением из них подзапросов, либо, если это не возможно, то добавлением индексов на поля, которые используются при фильтрации результатов (Это заменит операцию TableScan на IndexScan).