Trabalho Prático 2 - Banco de Dados II

Prof. Glauber Dias Gonçalves

Tema: Indexação.

1. Informações preliminares:

(a) Modelo relacional da base utilizada:

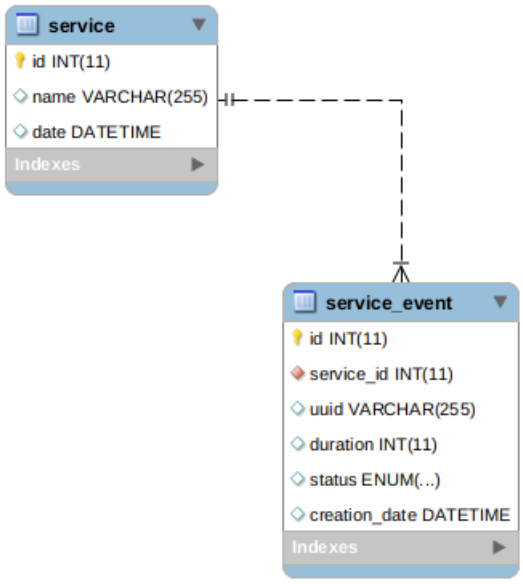

Figura 1: Modelo de relacional do esquema DBII-index.

Basicamente, nós temos uma tabela que possui alguns serviços cadastrados e outra tabela que possui eventos relacionados a estes serviços. O objetivo é melhorar o desempenho do banco de dados utilizado em uma aplicação responsável por avaliar semanalmente eventos. Isso é feito listando os serviços que apresentaram erro no campo *status* ou serviços lentos, isto é, possuem valores maiores do que 1000ms no campo *duration*.

(b) Requisitos: Mysql 5.7.25. Dica: utilize o MySQL Workbench para facilitar as medições que serão necessárias.

(c) Confira se há um total de 42 serviços:

```
SELECT COUNT(*) FROM DBII_index.service;
```

(d) Confira se há um total de 781479 eventos:

```
SELECT COUNT(*) FROM DBII_index.service_event;
```

(e) Todas as medições deverão ser realizadas pelo menos 10 vezes, sendo reportada a média acompanhada do desvio padrão. Dica essencial: Não use nenhuma outra aplicação durante os experimentos, pois elas podem influenciar nas medições. Use o desvio padrão para observar se há uma regularidade nos resultados.

2. Consultas fundamentais para a aplicação do BD (25% nota)

(a) Quantos eventos tiveram erro na semana anterior a '2018-06-29 12:00:00'? Reporte o tempo médio e o desvio padrão da referida consulta.

```
SELECT service_id, COUNT(*)

FROM service_event

WHERE status = "error"
AND creation_date <= DATE("2018-06-29 12:00:00") - INTERVAL 1 WEEK

GROUP BY service_id;
```

(b) Quantos serviços de eventos tiveram duração maior do que 1000ms na última semana referente à '2018-06-29 12:00:00'? Reporte o tempo médio e o desvio padrão da referida consulta.

```
SELECT service_id, COUNT(*)

FROM service_event

WHERE duration > 1000
AND creation_date <= DATE("2018-06-29 12:00:00") - INTERVAL 1 WEEK

GROUP BY service_id;
```

(c) Insira o comando `EXPLAIN`[1] antes do comando `SELECT` em ambas as consultas executadas anteriormente. Em seguida, execute tais consultas, individualmente, uma única vez. De acordo com os tempos de execução obtidos e informações reportas pelos campos *possible_keys*, *keys* e *rows* do comando `EXPLAIN`, o que se pode concluir acerca das duas consultas anteriores?

3. Indexando nossas tabelas

Nas consultas acima, as cláusulas `WHERE` usam 2 colunas cada (3 colunas diferentes no total). Essas colunas são: *status*, *duration* e *creation_date*. Então, vamos adicionar apenas índices básicos com base em um único um campo para essas colunas e, em seguida, executar nossas consultas novamente para ver como o tempo de execução e a saída do comando `EXPLAIN` são alterados:

(a) Adicione um índice para cada uma das colunas mencionadas. Em seguida, realize novamente a análise proposta em 2a), 2b) e 2c) reportando a média e o desvio padrão. O tempo das consultas melhorou? O que se pode concluir em relação a cada uma das consultas?

```
CREATE INDEX service_event_status_index ON service_event (status);

CREATE INDEX service_event_duration_index ON service_event (duration);

CREATE INDEX service_event_creation_date_index ON service_event (creation_date);
```

[1] Estude o comando caso necessário: `EXPLAIN` https://dev.mysql.com/doc/refman/5.7/en/using-explain.html.

(b) Vamos agora, introduzir dois índices compostos: (*status*, *creation_ date*) e (*duration*, *creation_ date*). Em seguida, refaça a análise proposta em 2a), 2b) e 2c). O que se pode concluir? Você notou algo diferente na consulta 2b) de acordo com os campos retornados pelo `EXPLAIN`?

```
CREATE INDEX service_event_status_creation_date_index
ON service_event (status, creation_date);

CREATE INDEX service_event_duration_creation_date_index
ON service_event (duration, creation_date);
```

4. Forçando o uso de um determinado índice (25% nota)

(a) Quando as consultas são estáticas, isto é, não serão alteradas, uma dica é utilizar os comandos `USE INDEX` e `FORCE INDEX`. Primeiro, pesquise sobre tais comandos na documentação do Mysql e em seguida execute as seguintes consultas, reportando o tempo médio com o desvio padrão. Finalmente, faça uma análise inserindo a cláusula `EXPLAIN` e compare os resultados com a questão 3b.

```
SELECT service_id, COUNT(*)

FROM service_event USE INDEX (service_event_duration_creation_date_index)

WHERE duration > 1000
AND creation_date <= DATE("2018-06-29 12:00:00") - INTERVAL 1 WEEK

GROUP BY service_id;

SELECT service_id, COUNT(*)

FROM service_event FORCE INDEX (service_event_duration_creation_date_index)

WHERE duration > 1000
AND creation_date <= DATE("2018-06-29 12:00:00") - INTERVAL 1 WEEK

GROUP BY service_id;
```

5. Criação de arquivos indexados (50% nota)

Exporte a tabela 'service_event' para um arquivo csv 'service_event.csv' (use colunas separadas por tabulação preferencialmente). Faça um programa com a sua linguagem de programação preferida que realiza os procedimentos a seguir a partir de um menu interativo com as seguintes opções: *(1) dividir arquivo em csv em blocos; (2) consulta sem indexação; e (3) consulta com indexação.*

As opções do menu devem realizar os seguintes procedimentos respectivamente:

1. Ordena o arquivo 'service_event.csv' pelo campo 'duração' e o divide em 100 arquivos de tamanhos iguais nomeados como 'service_event_001.csv' até 'service_event_100.csv'. A seguir, remove (apagar) o arquivo 'service_event.csv' e cria um arquivo de índice para o campo duração utilizando uma das técnicas de indexação mostradas em sala de aula. Note que o ponteiro do arquivo de índice corresponde ao nome de um dos arquivos csv.

2. Realiza a seguinte consulta **sem** utilizar arquivo de índice: mostra na tela quantos eventos em cada serviço tiveram duração maior do que 1000ms e data menor ou igual a '2018-06-29 12:00:00' e tempo de execução da consulta.

3. Realiza a seguinte consulta **com** arquivo de índice: mostra na tela quantos eventos em cada serviço tiveram duração maior do que 1000ms e data menor ou igual a '2018-06-29 12:00:00' e tempo de execução da consulta.

Quais as conclusões podem ser obtidas sobre o tempo de execução da consulta sem e com indexação? A indexação implementada ajudou na rapidez da consulta? Explique por que a indexação ajudou (ou não) na consulta.

6. Entrega

A entrega deverá ser realizada pelo *Sigaa* até o dia **30/06** no formato pdf contendo as repostas e as medições para todas as perguntas. As medições deverão ser acompanhadas de média e desvio padrão obtidas de pelo menos 10 execuções das consultas.

Observação: 1,0 ponto extra na 1a. unidade para quem construir o índice com árvore B+.