



RENAN HENRIQUE GOMES DAMAZIO ASSUNÇÃO RA 21038114  
ALEX ARANTES GONÇALVES RA 21011214

## **RELATÓRIO 2**

Tópicos Emergentes em Bancos de Dados

Santo André – SP

2019

## QUESTÃO 1

Uma relação contendo 600000 tuplas e 3 atributos foi criada seguindo as seguintes especificações: o primeiro atributo *att1* do tipo *int*, não contendo valores duplicados ou nulos. Os demais atributos também *int* aleatoriamente escolhidos entre valores 1 e 600000. Um índice do tipo *B+Tree* foi criado apenas para o primeiro atributo

Os scripts utilizados para esta questão encontram-se no arquivo *Q1.txt*. Para todos os testes desta questão foi removido o processamento paralelo os tempos apresentados ignorarem o tempo de planejamento e serem expressos em milissegundos.

Uma query do formato *SELECT \* FROM Rel where att1 between ... and ...;* foi executada aumentando-se progressivamente a quantidade de tuplas em ordem de 10% do total das 600000 tuplas presentes na relação.

As mesmas queries foram executadas sem que a indexação fosse utilizada.

Cada uma das Queries foi executada três vezes, seus tempos de execução gravados e uma média ponderada foi calculada para elaboração de um gráfico comparativo dos tempos de execuções nas duas condições apresentadas.

A tabela 1 mostra os resultados obtidos utilizando-se indexação *B+ Tree*

	Execução utilizando Indexação B+ Tree			
Qtd de tuplas	Tempos de execução (ms)			Médias(ms)
1	0	0	0	0
60000	9	8	8	8.333333333
120000	16	17	18	17
180000	26	26	26	26
240000	39	33	33	35
300000	41	41	41	41
360000	56	71	57	61.33333333
420000	60	60	60	60
480000	62	61	60	61
540000	64	65	62	63.66666667
600000	66	65	67	66

**Tabela 1** - Execução das queries utilizando indexação *B+ Tree*

A Tabela 2 apresenta os resultados obtidos para as execuções sem indexação

	Execução sem utilização de Indexação			
Qtd de tuplas	Tempos de execução (ms)			Médias(ms)
1	44	44	44	44
60000	46	46	45	45.66666667
120000	47	48	48	47.66666667
180000	50	51	50	50.33333333
240000	93	52	54	66.33333333
300000	58	54	54	55.33333333
360000	57	58	59	58
420000	58	59	84	67
480000	60	60	62	60.66666667
540000	65	62	62	63
600000	65	65	64	64.66666667

**Tabela 2** - Execução das queries sem utilização de indexação

As médias dos valores do experimento foram utilizadas para elaboração do gráfico comparativo - Gráfico 1



**Gráfico 1** - Execução das queries com indexação x sem indexação

Por meio da análise do gráfico é possível fazer algumas conclusões:

- I. A quantidade das tuplas afeta o desempenho das execuções indexadas, de modo que o otimizador optou por realizar sequential scan para uma quantidade de tuplas maior ou igual a 360.000.
- II. Apesar do desempenho do sequential scan ser inferior para quantidade de tuplas menores que 360000, se comparado ao da indexação *B+Tree*, o tempo de execução sofreu uma menor variação no decorrer das execuções.

## QUESTÃO 2

Inicialmente decidimo buscar os índices que operavam sobre as relações *depto*, *proj*, *emp* onde foi encontrado um único índice *idx\_depto* do tipo *btree* que trabalha sobre o campo *coddepto* da tabela *depto*. Todos os scripts utilizados e resultados obtidos seguem no arquivo *Q2.txt*.

As transações apresentadas foram avaliadas levando em conta o tempo de execução, e o plano de execução, gerando os seguintes resultados:

- **T1:** Teve tempo médio de execução de 35ms utilizando sequential scan e não usando o índice *idx\_depto*. Como a consulta retorna em tempo satisfatório e a relação é modificada com muita frequência decidimos não criar nenhum índice para não afetar o tempo das operações de modificação da relação e também porque os benefícios seriam muito baixos nesse momento.
- **T2:** Apresentou tempo médio de execução de 375ms e assim como a anterior fez um sequential scan e não utilizou o índice encontrado. Como das três transações essa foi a que apresente o maior tempo médio, tentamos a criação de índices. O primeiro índice criado foi do tipo *btree* e o segundo *BRIN*, porém em ambos os casos os índices não foram utilizados e não houve alteração no tempo médio. Por esse motivo decidimos não criar índices sobre essa relação.
- **T3:** Para realizar teste mais profundos com o uso do *betwee* primeiro procuramos quantas tuplas haviam na relação *depto*, obtendo o valor de 500. Em seguida executamos a relação para 5 faixas de *coddepto* distintas, de 1-100, 1-200, 1-300, 1-400 e 1-500. Apenas a primeira faixa utilizou o índice *idx\_depto* obtendo tempo de execução de 15ms, as demais utilizaram

sequential scan obtendo tempos crescentes de 15ms a 22ms. Devido ao tempo satisfatório de busca e alto índice de atualizações escolhemos não criar nenhum índice para aperfeiçoar as transações nesta relação.

A análise dos resultados sobre as transações nos leva as seguintes conclusões:

- I. Os testes executados mostraram que a criação de índices não ajudariam nas transações avaliadas, por isso decidiu-se não criar nenhum índice
- II. Foi encontrado um único índice que poderia contribuir com as transações analisadas, o *idx\_depto* do tipo btree que trabalha sobre o campo *coddepto* da tabela *depto*, porém não contribui com as transações T1 e T2 e tem baixo impacto em T3. Como ele pode contribuir com outras transações ainda não mencionadas decidimos não alterá-lo