

Disciplina Tecnologias de Bancos de Dados	Curso Sistemas de informação	Turno Noite
Professor Wladimir Cardoso Brandao	Tipo do Documento AS03: Proposição e otimização de índices e consultas	Período 1º / 2019

Alunos: Alex Junio Rodrigues, Leandro David César Pereira

- Não foi necessário a criação de index pois o mysql cria automaticamente os índices nas chaves primárias das tabelas. Segue abaixo as consultas, o plano de execução, a consulta otimizada e o plano de execução após otimização da consulta.

1- Projetar publicações do feed de um usuário

```
SELECT id_publicacao
FROM
publicacao p JOIN usuario_seguidor s ON (p.id_usuario = s.id_usuario_segue)
JOIN usuario u ON (s.id_usuario = u.ID)
WHERE u.ID = 1
ORDER BY p.`Timestamp` DESC;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index; Using filesort
1	SIMPLE	p	NULL	ALL	fk_id_user_publication	NULL	NULL	NULL	10	100	NULL
1	SIMPLE	s	NULL	eq_ref	PRIMARY, fk_follower	PRIMARY	8	const, instagram.p.id_usuario	1	100	Using index

```
SELECT id_publicacao
FROM publicacao p, usuario_seguidor s, usuario u
WHERE
s.id_usuario_segue = p.id_usuario
u.ID = s.id_usuario
u.ID = 1
ORDER BY p.`Timestamp` DESC;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index; Using filesort
1	SIMPLE	p	NULL	ALL	fk_id_user_publication	NULL	NULL	NULL	10	100	NULL
1	SIMPLE	s	NULL	eq_ref	PRIMARY, fk_follower	PRIMARY	8	const, instagram.p.id_usuario	1	100	Using index

2- Projetar stories por usuário

Observação: 86400000 equivale ao valor 24 horas em milissegundos.

```
SELECT id_story
FROM
stories p JOIN usuario_seguidor s ON (p.id_usuario = s.id_usuario_segue)
JOIN usuario u ON (s.id_usuario = u.ID)
WHERE u.ID = 1
AND now() - p.`timestamp` < 86400000
ORDER BY p.`Timestamp` DESC;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index; Using filesort
1	SIMPLE	p	NULL	ALL	fk_id_user_story	NULL	NULL	NULL	10	100	Using where
1	SIMPLE	s	NULL	eq_ref	PRIMARY, fk_follower	PRIMARY	8	const, instagram.p.id_usuario	1	100	Using index

```
SELECT s.id_story
FROM usuario_seguidor us, usuario u, stories s
WHERE
us.id_usuario_segue = s.id_usuario AND
u.ID = us.id_usuario AND
u.ID = 1 AND
now() - s.`timestamp` < 86400000
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index
1	SIMPLE	s	NULL	index	fk_id_user_story	fk_id_user_story	9	NULL	10	100	Using where; Using index
1	SIMPLE	us	NULL	eq_ref	PRIMARY, fk_follower	PRIMARY	8	const, instagram.s.id_usuario	1	100	Using index

3- Projetar nome de usuários que um determinado usuário segue

```
SELECT u.NomeUsuario
FROM usuario u JOIN usuario_seguidor s ON s.id_usuario_segue = u.ID
WHERE s.id_usuario = 1;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	s	NULL	ref	PRIMARY, fk_follower	PRIMARY	4	const	29	100	Using index
1	SIMPLE	u	NULL	eq_ref	PRIMARY	PRIMARY	4	instagram.s.id_usuario_segue	1	100	NULL

```
SELECT u.NomeUsuario
FROM usuario_seguidor s, usuario u
WHERE
s.id_usuario_segue = u.ID and
s.id_usuario = 1;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	s	NULL	ref	PRIMARY, fk_follower	PRIMARY	4	const	29	100	Using index
1	SIMPLE	u	NULL	eq_ref	PRIMARY	PRIMARY	4	instagram.s.id_usuario_segue	1	100	NULL

4- Projetar historico de um usuário

```
SELECT h.`timestamp`, h.interacao
```

FROM usuario u JOIN historico h ON h.id_usuario = u.ID
WHERE u.ID = 64;

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index
1	SIMPLE	h	NULL	ref	fk_user_historic	fk_user_historic	4	const	1	100	NULL

EXPLAIN SELECT h.`timestamp`,h.interacao
FROM usuario u, historico h
WHERE
h.id_usuario = u.ID and
u.ID = 64;

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index
1	SIMPLE	h	NULL	ref	fk_user_historic	fk_user_historic	4	const	1	100	NULL

5- Projetar nome de colecao de um usuário

SELECT c.Nome
FROM usuario u JOIN colecao c ON u.ID = c.id_usuario
WHERE u.ID = 49;

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index
1	SIMPLE	c	NULL	ref	fk_user_collection	fk_user_collection	4	const	1	100	NULL

SELECT c.Nome
FROM usuario u ,colecao c
WHERE
u.ID = c.id_usuario and
u.ID = 49;

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	u	NULL	const	PRIMARY	PRIMARY	4	const	1	100	Using index
1	SIMPLE	c	NULL	ref	fk_user_collection	fk_user_collection	4	const	1	100	NULL