



Databázové systémy

Dokumentácia k projektu

Alexandra Slezáková (xsleza20)
Matej Soroka (xsorok02)

Obsah

1	Trigger	2
1.1	map_scale	2
1.2	Player_PK	2
2	Procedúry	2
2.1	wrong_class	2
2.2	who_is_rich	2
3	Explain plan a index	2
3.1	Před zavedením indexu	3
3.2	Po zavedení indexu	3

1 Trigger

1.1 map_scale

Trigger spustený pred vkladáním alebo upravovaním riadkov tabuľky MAP.

Kontroluje správny formát mierky mapy v stĺpci `scale` pomocou regulérneho výrazu: `(\d)+ : (\d)+`

1.2 Player_PK

Trigger spustený pred vkladáním riadka do tabuľky `PLAYER` v prípade, že nie je zadaná hodnota stĺpca `player_id` obsahujúca primárny kľúč tabuľky. Spolu s triggrom je vytvorená aj sekvencia, ktorá zaručuje unikátnosť vygenerovaného primárneho kľúča.

2 Procedúry

2.1 wrong_class

Procedúra pre overenie správnosti povolaní vytvorených postáv v tabuľke `CHARACTER`. Procedúra preiteruje celú tabuľku pomocou premennej `characters%ROWTYPE`, ktorá v každej iterácii obsahuje iterovaný riadok z tabuľky. Pri každej iterácii porovnáva reťazec uložený v stĺpci `class`. Ak sa hodnota v stĺpci nerovná jednej zo všeobecne známych povolaní, procedúra vypíše užívateľa.

2.2 who_is_rich

Procedúra pomocou agregáčnej funkcie `avg` vypočíta aritmetický priemer hodnoty reprezentujúcej priemerný počet zlata stĺpca `gold` v tabuľke `PLAYER`, ktorý hráč vlastní. Hodnotu uloží do premennej. Následne procedúra preiteruje tabuľku `PLAYER` a porovnáva priemernú hodnotu s hodnotou v stĺpci `gold`, v prípade že je hodnota v stĺpci väčšia ako hodnota v premennej obsahujúcej priemer, hráč sa označí ako objektívne bohatší k väčšine a vypíše.

3 Explain plan a index

Select nám vyhľadáva počet postáv vlastnených administrátormi, keďže administrátor je podmnožina hráča, tak jeho informácie sú uložené v tabuľke `PLAYER` s jediným rozdielom, a tým je hodnota v stĺpci `role`, kde hodnota reprezentuje rolu daného hráča. Keďže rola je reťazec, tak je vhodné z neho vytvoriť index.

3.1 Pred zavedením indexu

Plan hash value: 3809253535

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		2	160	7 (43)	00:00:01
1	SORT ORDER BY		2	160	7 (43)	00:00:01
2	HASH GROUP BY		2	160	7 (43)	00:00:01
3	NESTED LOOPS		2	160	5 (20)	00:00:01
4	NESTED LOOPS		3	160	5 (20)	00:00:01
5	VIEW	VW_GBF_7	3	180	4 (25)	00:00:01
6	HASH GROUP BY		3	141	4 (25)	00:00:01
7	TABLE ACCESS FULL	CHARACTER	3	141	3 (0)	00:00:01
* 8	INDEX UNIQUE SCAN	PK_PLAYER	1		0 (0)	00:00:01
* 9	TABLE ACCESS BY INDEX ROWID	PLAYER	1	20	1 (0)	00:00:01

Predicate Information (identified by operation id):

8 - access("P"."PLAYER_ID"="ITEM_1")
9 - filter("P"."ROLE"='admin')

3.2 Po zavedení indexu

Pri použití indexu nám klesla výpočetná náročnosť a množstvo alokovanej pamäte.

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		2	126	6 (17)	00:00:01
1	SORT ORDER BY		2	126	6 (17)	00:00:01
2	HASH GROUP BY		2	126	6 (17)	00:00:01
* 3	HASH JOIN		2	126	5 (0)	00:00:01
4	VIEW	VW_GBF_7	1	16	2 (0)	00:00:01
5	TABLE ACCESS BY INDEX ROWID BATCHED	PLAYER	1	20	2 (0)	00:00:01
* 6	INDEX RANGE SCAN	ADMIN_INDEX	1		1 (0)	00:00:01
7	TABLE ACCESS FULL	CHARACTER	3	141	3 (0)	00:00:01

Predicate Information (identified by operation id):

3 - access("ITEM_1"="C"."PLAYER_ID")
6 - access("P"."ROLE"='admin')