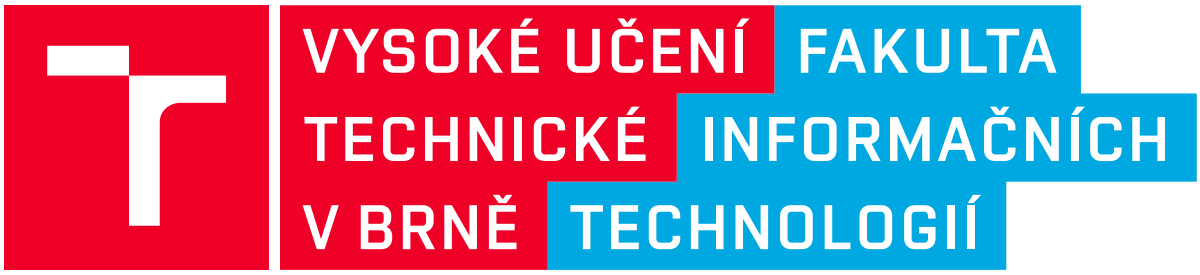

47. Svet mágie

Dokumentácia

Martin Koči (xkocim05)
Magdaléna Ondrušková (xondru16)

10.04.2020 Brno

Obsah

1 Zadanie

47. Svět Magie
Kúzelnícky svet vytvára informačný systém pre evidenciu kúziel a kúzelníkov. Mágia v kúzelníckom svete je členená podľa elementov (napr. voda, oheň, vzduch...), ktoré majú rôzne špecializácie (obrana, útok, podpora...) a rôzne, ale pevne dané, farby mágie (napr. ohnivá mágia je pomarančovo-oranžová). Každé kúzlo má potom jeden hlavný element a môže mať niekoľko vedľajších elementov (napr. voda a ľad), pričom každý kúzelník má pozitívnu synergiu s určitými elementami. U kúzelníkov takisto evidujeme veľkosť many, dosiahnutú úroveň v kúzlení (predpokladáme klasickú stupnicu E, D, C, B, A, S, SS,...). U jednotlivých kúziel potom ich zložitosť zoslania, typ (útočné, obranné) a silu. Kúzla nemôžu byť samovoľne zoslané, ale len s využitím kúzelníckych kníh tzv. grimoárov. Grimoáry v sebe zoskupujú viacero pripravených kúziel a ukladáme si celú históriu ich vlastníctva. Grimoáry môžu obsahovať kúzla rôznych elementov, ale jeden z elementov je pre ne primárny, pričom môžu obsahovať približne 10 - 15 kúziel. S postupom času však grimoáry strácajú nabitú mágiu (približne po mesiaci stratia celú mágiu) a je potrebné ich znovu dobiť, ale len na dedikovaných miestach, kde presakuje mágia (miera presakovania daného miesta je evidovaná) určitého typu elementu (predpokladajte, že na danom meste presakuje práve jeden typ). Toto nabitie však nemusí byť prevedené vlastníkom ale aj iným kúzelníkom. V prípade blížiaceho sa vypršania mágie grimoáru, systém pošle upozornenie vlastníkovi. Alternatívnym spôsobom zoslania mágie je potom s využitím zvitku, ktorý obsahuje práve jedno kúzlo a po jeho použití sa rozpadne.

2 Implementácia

2.1 TRIGGERY

2.1.1 `element_sequence` + `element_gen_id`

Sekvencia slúži na generovanie `id_element` pre tabuľku `element`. Sekvencia začína na čísle 1, pri každom ďalšom zavolaní sa zvyšuje o 1. Následne nasleduje trigger, ktorý pred pridaním novej položky do tabuľky `element` zvýši `id_element` na základe sekvencie o 1. Vrámci nového spúšťania je pomocou príkazu `DROP` vymazaná aj aktuálna sekvencia (začína sa znovu od 1).

2.1.2 `grimoar_history`

Tento trigger slúži na ukladanie histórie vlastníctva grimoáru. Zaoberáme sa tu dvoma situáciami - grimoár ešte neexistuje a vkladáme teda nový grimoár alebo grimoár už existuje ale zmenil vlastníka.
Ak grimoár ešte neexistuje, vložíme pomocou triggeru do tabuľky `history_grimoar` nový grimoár, login kúzelníka ktorý vlastní daný grimoár a dátum, kedy začal vlastniť grimoár nastavíme na aktuálny dátum a čas.
Ak meníme vlastníctvo grimoáru, ktorý už v tabuľke existuje, tak nastavíme aktuálnemu grimoáru a kúzelníkovi dátum, kedy prestal vlastniť grimoár a vložíme nový údaj (rovnakým spôsobom ako keby sme vkladali nový grimoár). Prípadne, ak nenastavujeme nového vlastníka, tak k aktuálnemu vlastníkovi daného grimoáru nastavíme dátum kedy prestal vlastniť daný grimoár na aktuálny dátum a čas.

2.2 PROCEDÚRY

2.2.1 `win_rate`

Táto procedúra slúži na vyrátanie úspešnosti daného kúzelníka (predaného v parametri) v súbojoch. Ak sa v tabuľkách nachádza viacero kúzelníkov s rovnakým menom, nájdu sa všeci títo kúzelníci a vyráta sa to pre každého kúzelníka zvlášť. Pri výpise sa rozlišujú pomocou `login`, ktorý je unikátny a slúži aj ako primárny kľúč pre tabuľku `magician`. Ak sa daný kúzelník nezúčastnil žiadneho súboju, vypíše sa táto informácia.
Pre predvedenie funkcionality, je procedúra volaná pre kúzelníka `Hermiona`.

2.2.2 `spells_in_grimoar`

Procedúra `spells_in_grimoar` slúži na zobrazenie všetkých kúziel, ktoré sa nachádzajú v grimoári, predaného pomocou `id_grimoars` do tejto procedúry. Takisto zobrazuje počet, koľkokrát sa tam jednotlivé kúzla nachádzajú v danom grimoáry. Ak je zadané neplatné `id_grimoar` tak sa táto informácie tiež vypíše. Pre predvedenie funkcionality, je procedúra volaná pre grimoár s ID `1`

2.3 PRÍSTUPOVÉ PRÁVA

Sú pridelené prístupové práve pre druhého člena týmu - v tomto prípade vedúci týmu `xkocim05` prideluje práve druhému členovi týmu `xondru16`.

2.4 MATERIALIZOVANÝ POHĽAD

2.4.1 spells_with_primary_element_air

Materializovaný pohľad zobrazí tabuľku kúziel (`spell`), ktoré majú ako primárny element vzduch (`air`). Následne na predvedenie funkcionality je do tabuľky `spell` pridané nové kúzlo, ktoré obsahuje element `air`. Potom je znovu zavolaná tabuľka tohto materializovaného pohľadu a môžeme pozorovať, že sa tam novo-pridané kúzlo nenachádza.

2.5 EXPLAIN PLAN

Zaujímali nás výsledky Explain plan pre dotaz, kde sa pýtame na to, ktorý kúzelník vlastnil koľko grimoár (vlastní v súčastnosti ale vlastnil aj niekedy v minulosti). Tento dotaz sme teda spojili pomocou agregačnej funkcii `COUNT`, ktorá zráta počet grimoárov. Následne sme výsledky zoskupili pomocou funkcie `GROUP BY`:

```
SELECT magician.name AS name_of_magician,
COUNT(id_history_grimoar) AS number_of_grimoars
FROM history_grimoar
LEFT JOIN magician ON history_grimoar.login_history_magician = magician.login
GROUP BY magician.name
ORDER BY COUNT(login_history_magician) DESC;
```

2.6 Bez použitia indexu

Na obrázku, ktorý obsahuje výsledky vygenerované Explain planom je možné vidieť, že dvakrát (`HISTORY_GRIMOAR` a `MAGICIAN`) pri zadanom dotaze pristupujeme k dátam celej tabuľky `TABLE ACCESS FULL`- musí sa prechádzať celá tabuľka (všetky riadky a stĺpce). Na obrázku tiež vidíme aj koľko nás táto tabuľka stojí vrámci procesorového času.

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		14	5502	7 (15)	00:00:01
1	SORT ORDER BY		14	5502	7 (15)	00:00:01
2	HASH GROUP BY		14	5502	7 (15)	00:00:01
* 3	HASH JOIN OUTER		14	5502	6 (0)	00:00:01
4	TABLE ACCESS FULL	HISTORY_GRIMOAR	9	1161	3 (0)	00:00:01
5	VIEW	VW_GBF_7	9	2376	3 (0)	00:00:01
6	TABLE ACCESS FULL	MAGICIAN	9	2322	3 (0)	00:00:01

Obr. 1: Výsledky Explain plan bez použitia indexu

2.7 S použitím indexu

Tentokrát sme sa snažili danú operáciu urýchliť a ušetriť čo najviac procesorového času. To sme skúšali pomocou vytvorenia dvoch indexov a to konkrétne:

```
CREATE INDEX magician_ind ON magician(name, login);
CREATE INDEX history_grim_ind ON history_grimoar(login_history_magician);
```

Obrázok nižšie obsahuje výsledky, ako sa nám znížila cena procesorového času - a teda môžeme konštatovať, že cena sa znížila zhruba o **64%** a teda využitie indexov nám pomohlo danú operáciu urýchliť.

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		14	5502	3 (34)	00:00:01
1	SORT ORDER BY		14	5502	3 (34)	00:00:01
2	HASH GROUP BY		14	5502	3 (34)	00:00:01
* 3	HASH JOIN OUTER		14	5502	2 (0)	00:00:01
4	INDEX FULL SCAN	HISTORY_GRIM_IND	9	1161	1 (0)	00:00:01
5	VIEW	VW_GBF_7	9	2376	1 (0)	00:00:01
6	INDEX FULL SCAN	MAGICIAN_IND	9	2322	1 (0)	00:00:01

Obr. 2: Výsledky Explain plan s použitím indexu