VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta informačních technologií

DATABÁZOVÉ SYSTÉMY

2018/2019

Projektová dokumentace

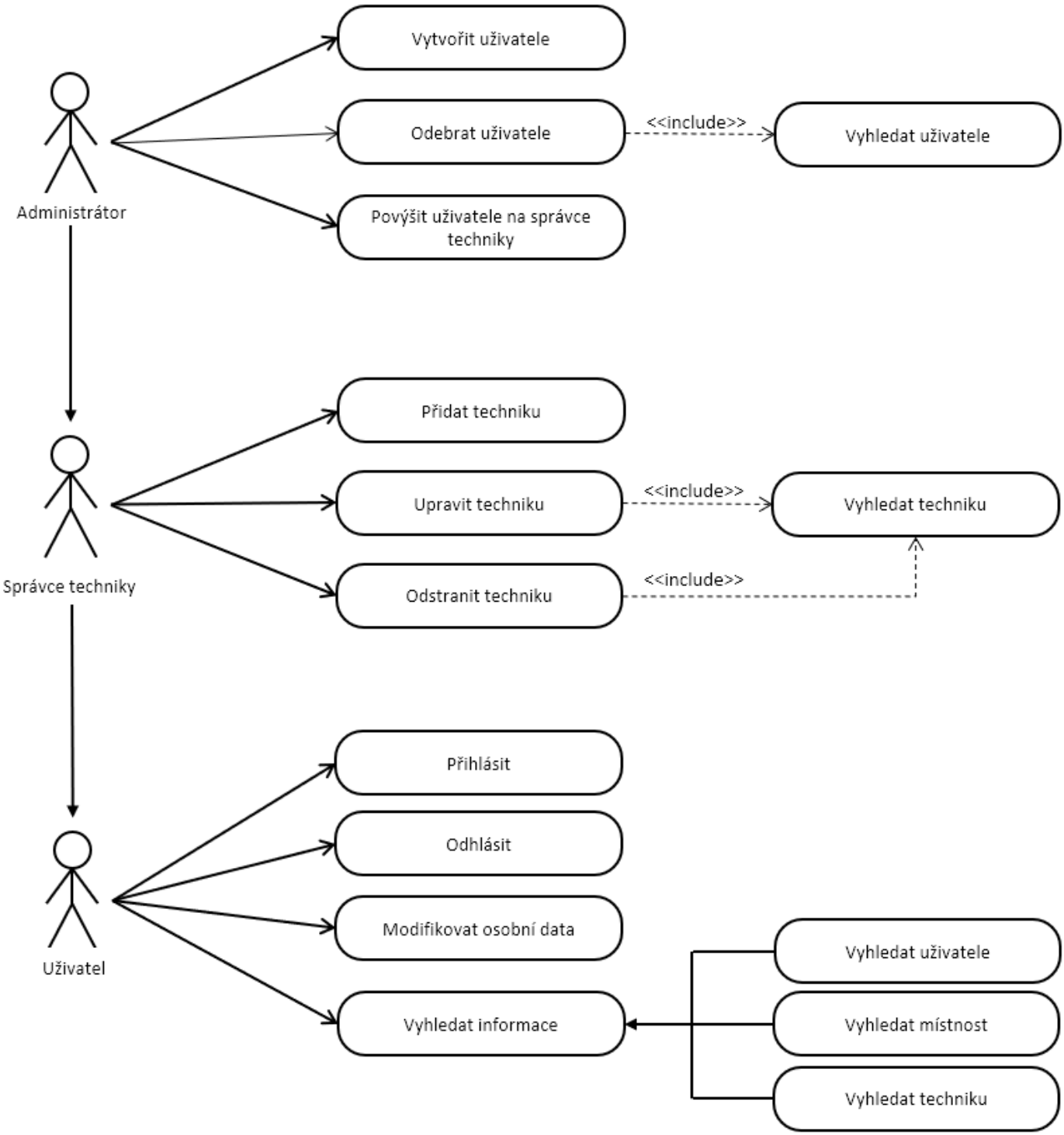
Evidence výpočetní techniky

Petr Kovář (xkovar82)

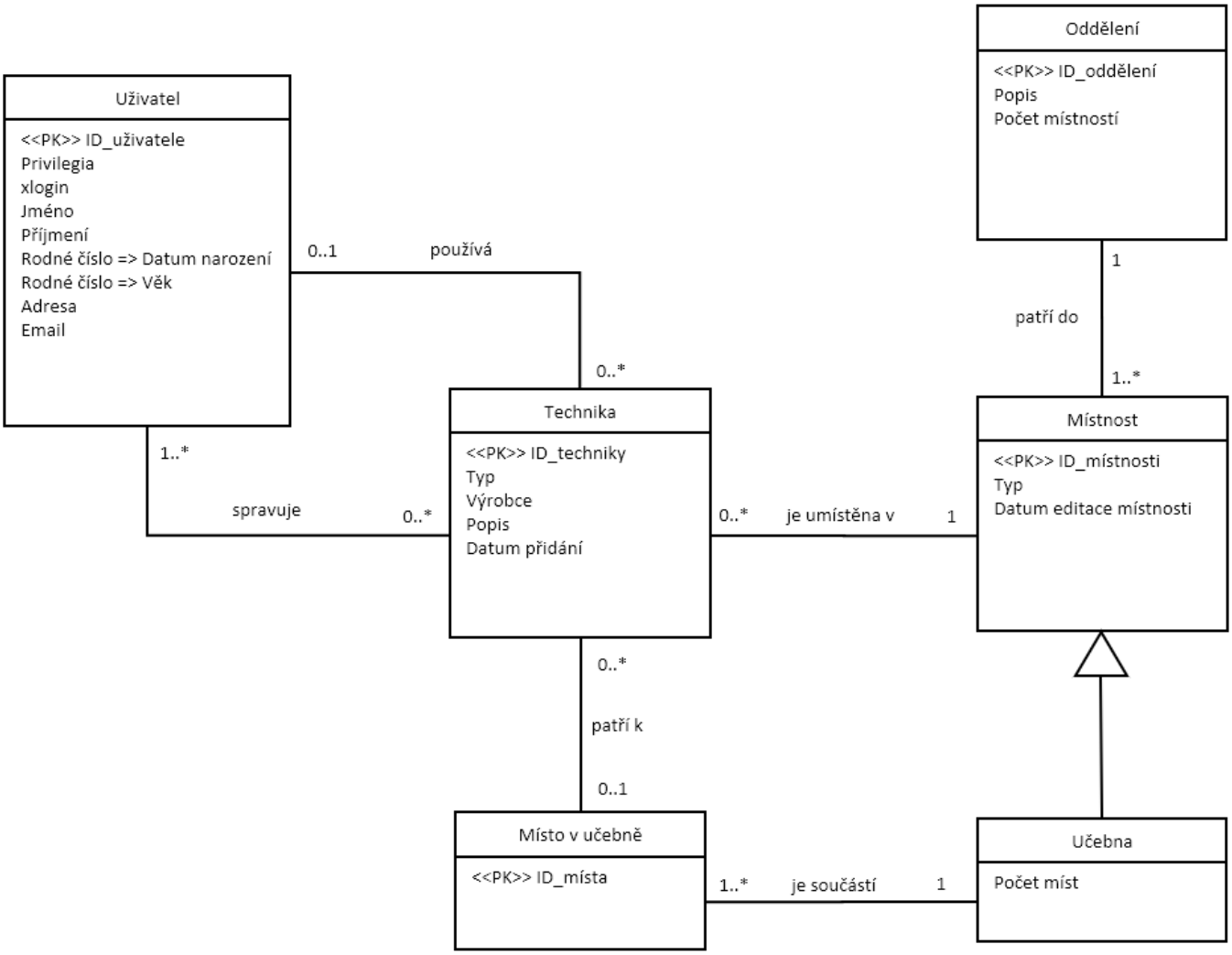
Pavel Kocurek (xkocur02)

Brno, 29. dubna 2019

Model případů užití



ER diagram


Oddělení
<<PK>> ID_oddělení
Popis
Počet místností
Uživatel
<<PK>> ID_uživatele
Privilegia
xlogin
Jméno
Příjmení
Rodné číslo => Datum narození
Rodné číslo => Věk
Adresa
Email
0..1
používá
0..*
1
patří do
1..*
Technika
<<PK>> ID_techniky
Typ
Výrobce
Popis
Datum přidání
Místnost
<<PK>> ID_místnosti
Typ
Datum editace místnosti
1..*
spravuje
0..*
0..*
je umístěna v
1
0..*
patří k
0..1
Místo v učebně
<<PK>> ID_místa
Učebna
Počet míst
1..*
je součástí
1

Implementace

Ze všeho nejdříve skript zahodí všechny objekty které mohly být vytvořené předešlou manipulací s databází, jedná se o tabulky, sequence, indexy nebo materializovaný pohled. Následně vytvoří tabulky a přidá potřebné vztahy.

Generalizace/Specializace

V případě naší implementace je UCEBNA specializací tabulky MISTNOST. Každá učebna je místnost, ale místnost nemusí být učebnou. V tabulce učebna je cizí klíč primárním klíčem tabulky místnost.

Triggery

Nejprve jsme naimplementovali triggery pro automatickou inkrementaci jednotlivých unikátních ID. Následně jsme vytvořili trigger, který kontroluje správnost formátu xloginu a emailové adresy.

Procedury

Jako první proceduru jsme vytvořili *check_psc*, která s pomocí kurzoru zkontroluje zadané psč jednotlivých uživatelů a pomocí *dbms_output* vypíše uživatele, jejichž poštovní směrovací číslo neodpovídá požadovanému formátu. V případě nečekané chyby je vyvolána exception *-2005: chyba procedury*.

Výstupem procedury *tech_perc* je procentuální soupis zastoupení jednotlivých výrobců v rámci inventáře techniky. Pomocí dvou kurzorů pro výběr z tabulky Technika (výběr sloupce "výrobce" a identický výběr pouze s odstraněnými duplicitami pomocí DISTINCT) probíhá porovnání. Využito je přitom několika proměnných: pro celkový počet techniky, počet kusů techniky od jednoho výrobce a proměnná pro uložení procentuálního zastoupení. Porovnání probíhá za pomoci dvou vnořených cyklů. Vnější cyklus získá název značky (kterou hodlá porovnávat) pomocí příkazu FETCH. Vnitřní cyklus prochází pomocí stejného příkazu tabulku se všemi hodnotami (včetně duplicit), ty následně porovná a v případě shody inkrementuje proměnnou s počtem výskytů. Dle vzorce: (pocet_vyskytu * 100) / pocet_zar je dále spočítán procentuální výskyt a společně s názvem výrobce je vypsán.

Přístupové práva

V případě přístupových práv jsme se rozhodli využít use case, kdy předáváme práva ke správě databáze administrátorovi databáze, tudíž jsme mu přenechali všechna práva na manipulaci s databází bez jakýchkoliv omezení.

Materializovaný pohled

Pro materializovaný pohled jsme využili *CACHE* které postupně optimalizuje čtení z pohledu, dále *BUILD IMMEDIATE,* které naplní pohled ihned po jeho vytvoření a *REFRESH ON COMMIT AS*, které aktualizuje pohled po commitu tabulek. Funkčnost jsme zobrazena na

případu, kdy přidáme data pomocí insert, ale na materializovaném pohledu se tato změna projeví až po provedení příkazu *COMMIT*.

Explain plan s vytvořením indexu

Pro demonstraci Explain plánu jsme využili jednoduchý SELECT.

Explain před použití indexu:

```
Plan hash value: 1284473290

-----------------------------------------------------------------------------------------------
| Id  | Operation                     | Name          | Rows  | Bytes | Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT              |               |     1 |    43 |     4  (25)| 00:00:01 |
|   1 |  HASH GROUP BY                |               |     1 |    43 |     4  (25)| 00:00:01 |
|   2 |   NESTED LOOPS                |               |     1 |    43 |     3   (0)| 00:00:01 |
|   3 |    NESTED LOOPS               |               |     6 |    43 |     3   (0)| 00:00:01 |
|   4 |     TABLE ACCESS FULL         | TECHNIKA      |     6 |    78 |     3   (0)| 00:00:01 |
|*  5 |     INDEX UNIQUE SCAN         | SYS_C00247232 |     1 |       |     0   (0)| 00:00:01 |
|   6 |    TABLE ACCESS BY INDEX ROWID| UZIVATEL      |     1 |    30 |     0   (0)| 00:00:01 |
-----------------------------------------------------------------------------------------------

Predicate Information (identified by operation id):
---------------------------------------------------

   5 - access("U"."USERID"="T"."USERID")

Note
-----
   - dynamic statistics used: dynamic sampling (level=2)
```

Explain plan po použití indexu:

```
Plan hash value: 827408779

-----------------------------------------------------------------------------------------------------
| Id  | Operation                     | Name                | Rows  | Bytes | Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT              |                     |     1 |    43 |     2  (50)| 00:00:01 |
|   1 |  HASH GROUP BY                |                     |     1 |    43 |     2  (50)| 00:00:01 |
|   2 |   NESTED LOOPS                |                     |     1 |    43 |     1   (0)| 00:00:01 |
|   3 |    NESTED LOOPS               |                     |     6 |    43 |     1   (0)| 00:00:01 |
|   4 |     INDEX FULL SCAN           | INDEX_POCET_TECHNIKY |     6 |    78 |     1   (0)| 00:00:01 |
|*  5 |     INDEX UNIQUE SCAN         | SYS_C00247232       |     1 |       |     0   (0)| 00:00:01 |
|   6 |    TABLE ACCESS BY INDEX ROWID| UZIVATEL            |     1 |    30 |     0   (0)| 00:00:01 |
-----------------------------------------------------------------------------------------------------

Predicate Information (identified by operation id):
---------------------------------------------------

   5 - access("U"."USERID"="T"."USERID")

Note
-----
   - dynamic statistics used: dynamic sampling (level=2)
```

Jak můžeme vidět, po použití indexu se sice zvýšilo využití procesoru, ale naopak se cost CPU, tedy počet přístupů na disk.

Závěr

Pro testování našeho skriptu jsme většinou využívali školní servery Oracle, ale občas nebyly servery dostupné, tak jsme byli nuceni využít dostupných online nástrojů, což nebylo tak příjemné. Při tvorbě nám pomohly znalosti získané z přednášek a demonstračních cvičení.