

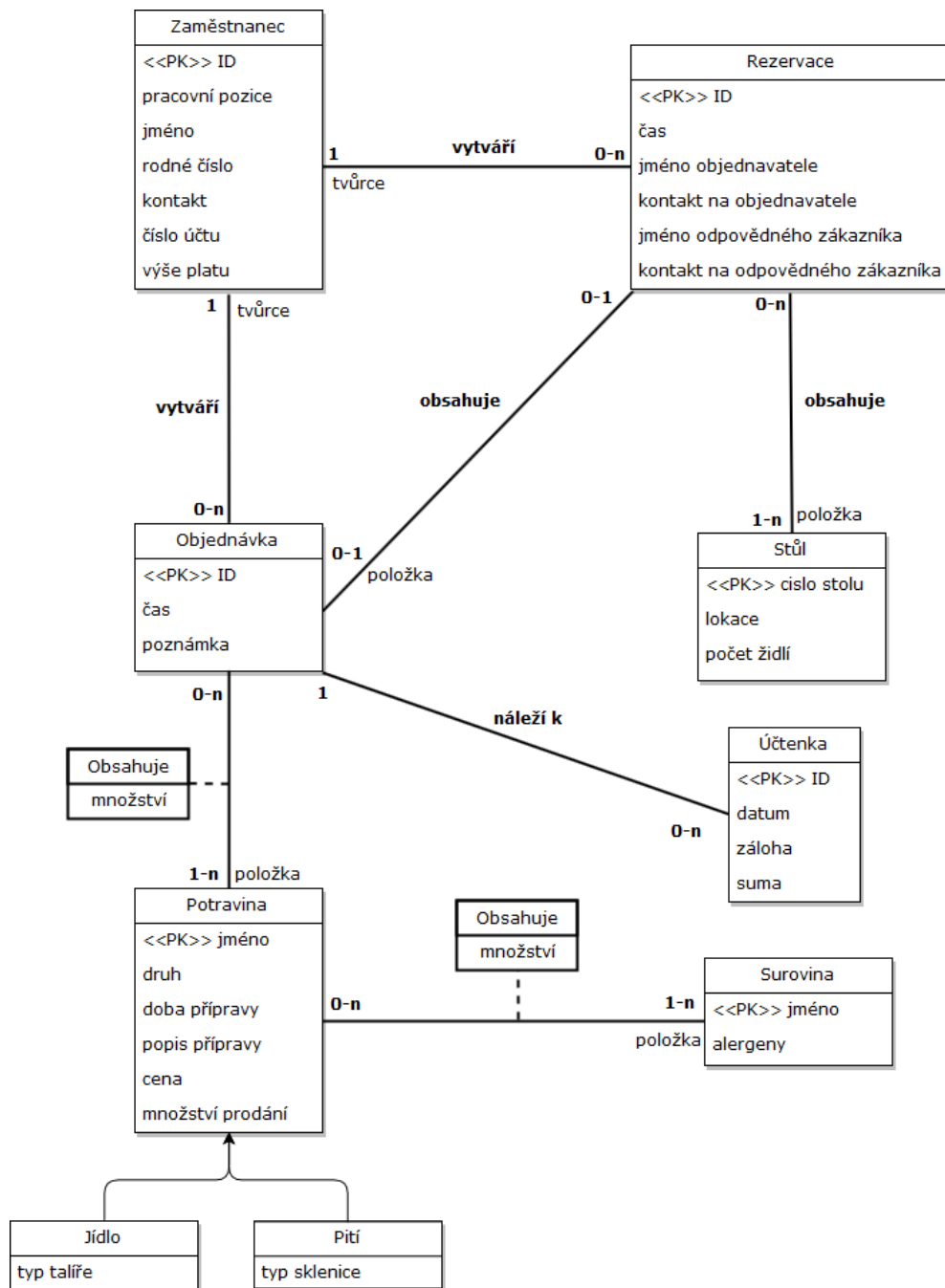
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

IDS - Databázové systémy
Dokumentace k projektu
zadání projektu
Restaurace

1 Zadání

Vytvořte IS pro restaurační zařízení, který napomůže k zjednodušení a zpřehlednění jeho provozu. Restaurace je členěna do více místností a má přední a zadní zahrádku a poskytuje běžné stravovací služby veřejnosti. Od informačního systému se požaduje, aby, krom jiného, umožnil správu rezervací a objednávek. Rezervovat je možné jeden nebo více stolů v místnostech či na zahrádkách, anebo celé místnosti, případně i celou restauraci pro různé společenské akce. Součástí rezervace také může být objednávka nápojů a jídel. Systém musí umožňovat zaměstnancům restaurace vkládat objednávky spolu s informacemi, který zaměstnanec danou objednávku vložil a pro koho je určena. Když se zákazníci rozhodnou zaplatit, musí jim systém vystavit účtenku. Po zaplacení pak příslušný zaměstnanec vloží záznam o platbě do systému. Systém by měl také poskytovat podrobný přehled tržeb za vybrané období. Přístup k této funkci bude mít pouze majitel. V neposlední řadě musí systém evidovat veškeré prodávané jídlo a pití (včetně složení), přičemž majitel a odpovědný vedoucí mají možnost měnit ceny jídla a pití nebo přidávat a odebírat položky.

2 Konečný ER diagram



3 Triggery

Oba vytvořené triggery pracují s tabulkou zaměstnanců. Jeden trigger generuje ID, druhý kontroluje platnost rodného čísla.

4 Procedury

Obě ze dvou procedur používají CURSOR a EXCEPTION. Procedura *procento_vytvorených_rezervací* má jako parametr ID zaměstnance a vypíše jeho jméno a kolik rezervací vytvořil (v procentech). Pracuje tedy se dvěma tabulkami: *Rezervace* a *Zamestnanec*. Druhá procedura *pocet_zidli* má jako parametr lokaci a vypíše lokaci a počet židlí, které se v lokaci nacházejí. Pracuje tedy s tabulkou *Stul*.

5 Přidělení práv uživateli xmisov00

Uživatel xmisov00 získal veškerá práva k procedurám a tabulkám, má tudíž oprávnění administrátora (např. majitel restaurace). Kuchař by měl, práva přístupu k tabulkám *Surovina*, *Potravina* a *PObsahujeS*.

6 Materializovaný pohled

Spolu s materializovaným pohledem byly vytvořeny i logy pro něj (využití FAST REFRESH ON COMMIT - nemusí se spouštět celý dotaz materializovaného pohledu). Mimo REFRESH FAST ON COMMIT jsme do pohledu nastavili i CACHE, pro lepší načítání z pohledu, BUILD IMMEDIATE pro okamžité naplnění pohledu po vytvoření a ENABLE QUERY REWRITE pro zapnutí optimalizace.

7 INDEX a EXPLAIN PLAIN

EXPLAIN PLAIN byl vytvořen pro spojení 3 tabulek a to *Objednavka*, *Potravina* a *OObsahujeP*. Z těchto tabulek byla vypočítána celková cena objednávek. Index byl vytvořen nad parametry tabulky *OObsahujeP* a to nad ID objednávky a jménem potraviny. Po použití INDEX se procesorové zatížení značně zmenšilo (viz. další strana).

Tabulka 1: Před použitím INDEX

ID	Operation	Name	Rows	Bytes	Cost (% CPU)
0	SELECT STATEMENT		5	855	6 (17)
1	—HASH GROUP BY		5	855	6 (17)
2	——NESTED LOOPS		5	855	6 (17)
3	——NESTED LOOPS		5	855	6 (17)
4	——VIEWS	VW_DBF_5	5	460	4 (25)
5	———HASH GROUP BY		5	395	4 (25)
6	———TABLE ACCES FULL	OBSAHUJEP	5	395	3 (0)
7	——INDEX UNIQUE SCAN	SYS_C001274290	1		0 (0)
8	——TABLE ACCES BY INDEX ROWID	POTRAVINA	1	79	1 (0)

Tabulka 2: Po použití INDEX

ID	Operation	Name	Rows	Bytes	Cost (% CPU)
0	SELECT STATEMENT		5	855	3 (0)
1	—HASH GROUP BY		5	855	3 (0)
2	——NESTED LOOPS		5	855	3 (0)
3	——NESTED LOOPS		5	855	3 (0)
4	——VIEWS	VW_DBF_5	5	460	1 (0)
5	———HASH GROUP BY		5	395	1 (0)
6	———INDEX FULL SCAN	INDEX_OBSAHUJEP	5	395	1 (0)
7	——INDEX UNIQUE SCAN	SYS_C001274290	1		0 (0)
8	——TABLE ACCES BY INDEX ROWID	POTRAVINA	1	79	1 (0)

7.1 Vysvětlení EXPLAIN PLAIN

Z tabulek je vidět, že se snížilo zatížení procesoru a též byl menší počet přístupů na disk. SELECT STATEMENT značí provedení samotného SELECT příkazu, HASH GROUP BY znamená, že se položky seskupí podle hashovacího klíče. Dva vnořené cykly ukazují průchod tabulkami. TABLE ACCES FULL značí průchod celou tabulkou bez použití indexu, INDEX FULL SCAN značí zase použití INDEXU. INDEX UNIQUE SCAN je přístup k tabulkám přes B-strom a nakonec TABLE ACCES BY INDEX ROWID je přístup do tabulky přes konkrétní řádek.

8 Závěr

Díky projektu jsme se seznámily s jazykem SQL a naučili se navrhovat databázové systémy. Pro vypracování projektu jsme používali nejen přednášky, ale hlavně externí zdroje, zejména fórum StackOverflow. Projekt jsme programovali v prostředí Oracle Developer, které nám bylo nabídnuto ve formě obrazu pro virtuální stroj. Dokumentaci jsme tvořili v pomoci \LaTeX u.