

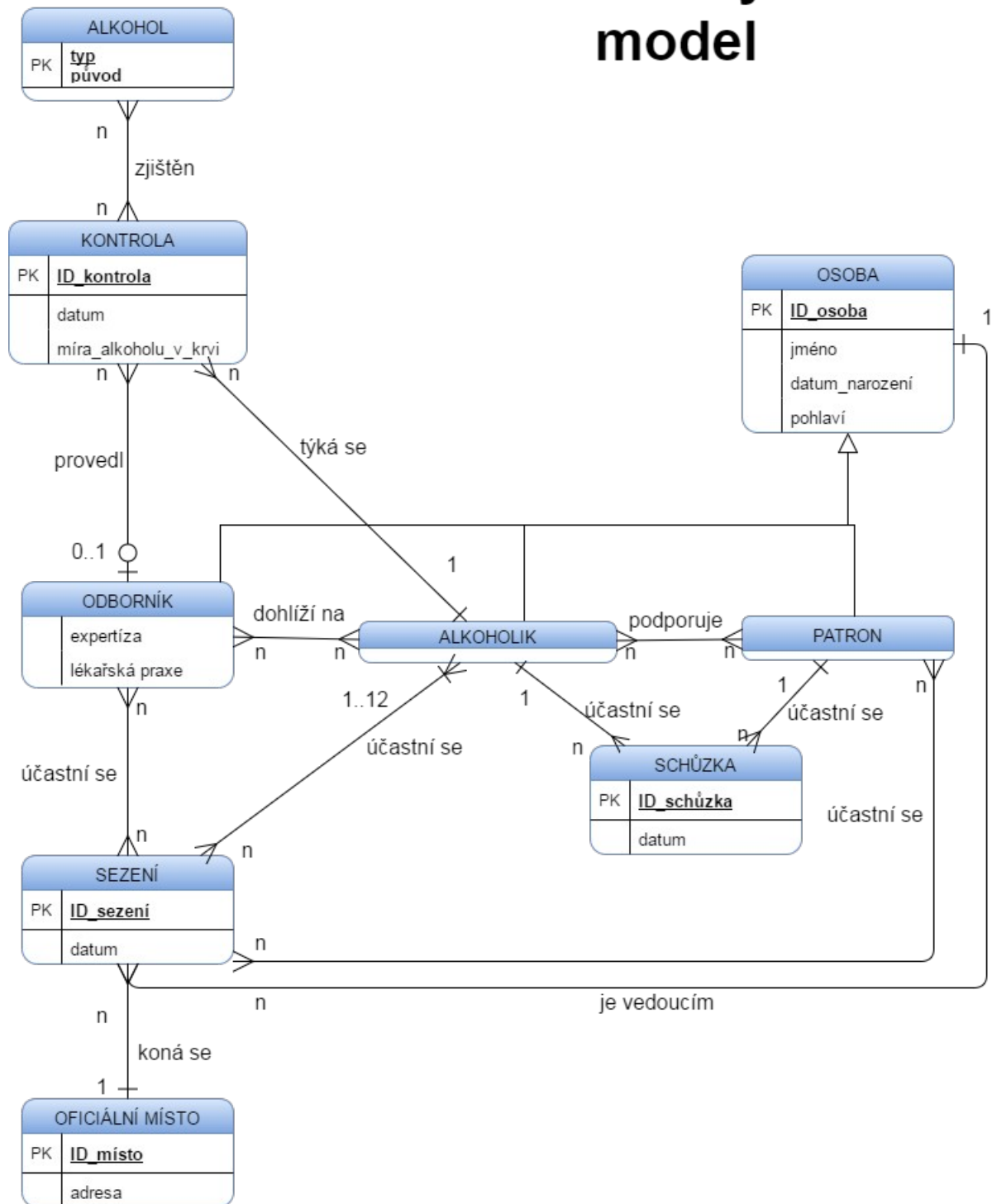
IDS

Projekt: Klub anonymních alkoholiků

Roman Andriushchenko, xandri03

Tomáš Bíl, xbilto00

Datový model



Popis pokročilých objektů schématu databáze

Triggery

- **osoba_unique**: používá sekvenci *osoba_seq* pro generování hodnot primárního klíče při vkládání do tabulky *Osoba*;
- **sezeni_misto_null**: zajistí, že při změně hodnoty primárního klíče v tabulce *Místo* dojde i ke změně v odkazující se tabulce *Sezeni*; když odkazovaný řádek bude odstraněn, příslušný cizí klíč bude nastaven na hodnotu *null*;
- **sezeni_max_alkoholiku**: zajistí, že k danému sezení lze přihlásit maximálně 3 alkoholiky.

Uložené procedury

- **alkoholick_check**: ověří, zda daný alkoholik navštívil alespoň jedno sezení v daném roce, v opačném případě simuluje zasílání upozornění;
- **osoba_zmen_jmeno**: používá se pro změnu jména v tabulce *Osoba*; vyvolává výjimku v případě, kdy hledaná osoba neexistuje.

Urychlení zpracování dotazu pomocí indexu

Uvažujme následující dotaz:

```
SELECT A.typ, A.puvod, SUM(mira) FROM Kontrola K, Alkohol_zjisten A
WHERE K.id_kontrola = A.kontrola AND K.mira >= 1.0
GROUP BY A.typ, A.puvod;
```

který spočítá celkový počet alkoholu (vzhledem k jeho typu a původu), který byl zjištěn při kontrolách, přičemž uvažujeme pouze seriózní případy kdy bylo zjištěno více než 1%. Příkaz EXPLAIN PLAN vypíše následující tabulku:

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		4	332	7 (15)	00:00:01
1	HASH GROUP BY		4	332	7 (15)	00:00:01
* 2	HASH JOIN		4	332	6 (0)	00:00:01
* 3	TABLE ACCESS FULL	KONTROLA	4	104	3 (0)	00:00:01
4	TABLE ACCESS FULL	ALKOHOL_ZJISTEN	9	513	3 (0)	00:00:01

Predicate Information (identified by operation id):

2	-	access("K"."ID_KONTROLA"="A"."KONTROLA")
3	-	filter("K"."MIRA">=1.0)

tj. použije se hash join přes všechny řádky obou tabulek pomocí sloupců *K.id_kontrola* a *A.kontrola*. Protože nás zajímají pouze kontroly s určitým rozmezím hodnot ve sloupci *mira*, můžeme daný dotaz zrychlit využitím indexu

nad tímto sloupcem. Výsledný plán vypadá následujícím způsobem:

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		4	332	6 (17)	00:00:01
1	HASH GROUP BY		4	332	6 (17)	00:00:01
* 2	HASH JOIN		4	332	5 (0)	00:00:01
3	TABLE ACCESS BY INDEX ROWID BATCHED	KONTROLA	4	104	2 (0)	00:00:01
* 4	INDEX RANGE SCAN	KONTROLA_MIRA	4		1 (0)	00:00:01
5	TABLE ACCESS FULL	ALKOHOL_ZJISTEN	9	513	3 (0)	00:00:01

Predicate Information (identified by operation id):

2 - access("K"."ID_KONTROLA"="A"."KONTROLA")

4 - access("K"."MIRA">=1.0)

teď neprocházíme přes všechny řádky, ale pouze ty které splňují podmínku ≥ 1.0 .

Materializovaný pohled

Pohled ***alkoholici_muži*** patříci druhému členovi týmu zpřístupňuje pouze alkoholiky mužského pohlaví.