



IDS – DATABÁZOVÉ SYSTÉMY

4. část

2014/2015

Zvolené zadání

Navrhněte IS malé nemocnice, který by poskytoval základní údaje o lékařích, sestrách či pacientech, kteří jsou a byli hospitalizováni v nemocnici. IS uchovává informace o všech těchto hospitalizacích, přičemž pacient může být v jeden čas hospitalizován pouze na jednom oddělení nemocnice. Při každé hospitalizaci je mu určen jeho ošetřující lékař.

Lékaři mohou pracovat na více odděleních zároveň. Na každém oddělení má lékař určitý úvazek, telefon atd., zatímco sestry pracují pouze na jednom oddělení. V rámci pobytu v nemocnici může pacient podstoupit různá vyšetření, která byla provedena na určitém oddělení ve stanoveném čase a provedl je určitý lékař, který také zapisuje výsledky vyšetření do IS. Dále mu mohou být podávány různé léky, každé podávání léku má určité detaily (kdy se podává, kolikrát apod.). V systému jsou uloženy i všeobecné informace o lécích (název, účinná látka, síla léku, kontraindikace atd.), aby si lékař mohl zkontrolovat správnost naordinovaného dávkování.

Generalizace/specializace

Vztah generalizace/specializace je v projektu použit u tabulek *Personál*, *Lékař* a *Zdravotní sestra*, přičemž tabulka *Personál* je generalizací lékaře a zdravotní sestry. Zahrnuje tedy všechny společné atributy pro tyto dvě tabulky. Tabulka *Lékař* navíc obsahuje atribut *Specializace* a tabulka *Zdravotní sestra* atribut *Kvalifikace*.

Databázové triggerry

Trigger definuje činnost, která se má spustit v případě, že nastane určitá situace. Zpravidla se jedná o různé vkládání, upravování či mazání dat v tabulkách.

V našem případě se přímo nabízí jejich použití v situacích, kdy vkládáme (přijímáme) lékaře na nové oddělení. Ze zadání nám totiž plyne povinnost umožnit lékaři pracovat na více odděleních současně a zároveň evidovat jeho úvazek, a to jak na každém oddělení zvlášť, tak celkově. Proto jsme vytvořili triggerry:

- **lekar_na_novem_odd**
- **lekar_odchazi_z_odd**

Ty jsou spuštěny v případě vkládání či odebírání záznamu z tabulky **Lekar_pracuje_na_odd** a volají procedury, které buď zvýší nebo sníží lékařův úvazek.

Dále jsme definovali trigger který hlídá, zda-li má každé oddělení vrchní sestru. A to z důvodu cyklické závislosti tabulek **Oddělení** a **Zdravotní_Sestra**. Oddělení totiž pro vytvoření potřebuje mít definovanou vrchní sestru a zároveň sestra musí mít dané oddělení na kterém pracuje. Toto jsme vytvořili tak, že oddělení ke svému vytvoření zdravotní sestru nepotřebuje, ale jakmile je mu přiřazena první zdravotní sestra (tzn. oddělení žádnou nemá), stává se z ní automaticky vrchní. Toto chování zabezpečuje trigger **vrchni_sestra**.

Dále jsme definovali trigger **pacient_insert** pro automatické generování primárních klíčů pro pacienty (ID_Pacienta) na základě sekvence **pacient_seq**.

Procedury

Použité procedury nám slouží k manipulaci s daty při použití triggerů, jsou to tyto procedury:

- **update_uvazek()** (zvýšení úvazku při přidávání lékaře k oddělení)
- **lower_uvazek()** (snížení úvazku při odebírání lékaře od oddělení)
- **zkontroluj_oddeleni()** (kontrola zda-li lékař už na oddělení nepracuje)
- **pridat_vrchni_sestru()** (pokud oddělení nemá vrchní sestru, je mu právě přiřazovaná sestra jako vrchní přidělena)

Dále máme definovanou proceduru **hospitalizovani_pacienti()**, která vypisuje právě hospitalizované pacienty, za využití kurzoru a rowtype. Nejprve se do kurzoru vloží dotaz, který vybere právě hospitalizované pacienty, a následně se v cyklu vybírají jednotlivé řádky a vypisují se do tabulky ve formátu "jméno příjmení, rč. rodné číslo".

Explain plan, index

Explain plan popisuje provádění dotazu, jehož výstupem je ID pacienta a počet jeho hospitalizací. Na prvním výpisu je vidět explain plan před zavedením indexování hospitalizací.

1	Plan hash value: 3190143351							
2								
3	-----							
4	Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	
5	-----							
6	0	SELECT STATEMENT		6	18	4 (25)	00:00:01	
7	1	HASH GROUP BY		6	18	4 (25)	00:00:01	
8	2	TABLE ACCESS FULL	HOSPITALIZACE	6	18	3 (0)	00:00:01	
9	-----							

Na druhém výpisu je stav po zavedení indexování. Je tedy zřejmé, že došlo k zjednodušení provádění dotazu, a tím také k zrychlení.

1	Plan hash value: 1499767034							
2								
3	-----							
4	Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	
5	-----							
6	0	SELECT STATEMENT		6	18	1 (0)	00:00:01	
7	1	SORT GROUP BY NOSORT		6	18	1 (0)	00:00:01	
8	2	INDEX FULL SCAN	HOSPINDEX	6	18	1 (0)	00:00:01	
9	-----							

Přístupová práva

Definice přístupových práv spočívá v použití příkazu *grant* nad všemi tabulkami, sekvencemi a materializovanými pohledy databáze, které chceme zpřístupnit pro daného uživatele. Oprávněný uživatel následně přistupuje k tabulkám pomocí zápisu `select * from login_zpřístupnitele.tabulka`.

Materializovaný pohled

Materializovaný pohled uchovává výsledky SQL dotazu. V našem případě se jedná o dotaz, jehož výsledkem je katalogové označení a název všech evidovaných léků. V našem systému se dá očekávat, že bude tento dotaz častý, proto je použití materializovaného pohledu pro tento dotaz vhodné, jelikož se dotaz při zavolání nebude provádět znovu, ale jeho výsledek bude načítán z paměti.