Aktivní databáze

- **pasívní databáze:** vytvoření databáze, vyhledávání, přidání/rušení záznamů podle požadavků uživatele
- **aktivní databáze:** navíc automatická reakce na událost nebo na změnu podmínky

Příklad: integritní omezení

v pasivním systému - velmi omezené : klíče, referenční integrita

aktivní prvky - **trigger** *(ne)platí podmínka -> vykonej akci*

implementace v pasívním systému

- při každé změně databáze spusť (uživatelský) program; nebo
- (uživatelský) proces, který periodicky testuje integritu

Alternativa : **systém je zodpovědný**

integritní podmínka porušena -> vykonej akci

Aktivní pravidla

on event **if** condition **then** action (ECA pravidla)

3 (nové) komponenty (aktivní) databáze

1. monitorování událostí

2. báze pravidel

3. výkonný modul (exekuční model) - jak použít pravidla

2 + 3 = sémantika aktivních pravidel **Příklad**

on insert(R,a) if a $>$ 100 then delete(R,a), insert(S,a)

- insert(S,200)
- insert(R,99) : pravidlo aktivizováno, podmínka vyhodnocena
- insert(R,101) : pravidlo aktivizováno, podmínka splněna, delete(R,101), insert(S,101)

K exekučním modelům

transakce $t = c_1; c_2; ..; c_n$ c_i atomický příkaz

$I_1; I_2; ..; I_n$ stavy databáze, $I_{i+1} = c(I_i)$

Kdy a jak aktivovat pravidlo: *immediate/deferred/concurrent* firing

každé aktivní pravidlo má svůj mód

někdy rozšíření o módy *událost-podmínka* a *podmínka-akce*

Obecné schéma exekuce pravidel

while nastala událost v pravidlech **do**

vyber 1 pravidlo

vyhodnoť podmínku

if podmínka splněna **then** vykonej akci

otázky, které musí být zodpovězeny

kdy vykonat pravidlo

které pravidlo vybrat

jak provést několik pravidel

jak sladit exekuci pravidel s vlastní aplikací

dosud neexistuje universální jazyk pro popis exekučních modelů