



UNIVERSIDADE FEDERAL DO MARANHÃO - UFMA

Banco de Dados II

Integridade

Carlos Eduardo **Portela** Serra de Castro

*



Integridade

- Introdução
- Restrições de Integridade
 - de Domínio
 - de Relação
- Asserções
- Procedimentos-gatilho
- Considerações Finais

Introdução

Objetivo:

Assegurar que mudanças feitas no banco de dados por usuários autorizadas não resultem na perda de consistência dos dados.

Causas:

Erros de entrada de Dados, erros de Operação ou Programação, falhas do sistema.
(danos acidentais <> falsificação deliberada -> segurança)

Restrições de Integridade

Domínio: especificam o conjunto de valores possíveis que podem ser associados a um atributo.

Relação: asseguram que um valor que aparece em uma relação para um dado conjunto de atributos também aparece em um certo conjunto de atributos de outra relação (integridade referencial).

Domínio

- **Valores nulos (nulls)**
- **Tipos de dados (boolean)**
- **Domínios compostos (DATA = DIA + MES + ANO)**
- **Ordenação (data-e < data)**
- **Procedimentos escritos pelo usuário**
- **Comparação entre domínios (Km -> milha – regras de conversão)**

Tipos de Domínio

SQL padrão

Cadeia de caracteres (1)

Numero em ponto fixo (1)

Inteiro(2)

Inteiro pequeno (smallint)(2)

Ponto flutuante (1)

Ponto flutuante com dupla precisão (2)

(1 – especificado pelo usuário; 2 – dependente da maquina)

Cláusula **check**

- A cláusula **check** em **SQL-92** permite restringir domínios:
- Exemplo:
Assegurar que um domínio de data do nascimento só permita valores posteriores a 1880
create domain data_nascimento integer(4)
constraint valor-teste check (valor >= 1880)

Integridade de Relação

- Assegura que um valor que aparece em uma relação (***tabela***) para um determinado conjunto de atributos apareça em outro conjunto de atributos em outra relação (***tabela***).

Relação e Tupla

- Como as tabelas em essência são relações, utiliza-se os termos matemáticos ***relação e tupla, no lugar de tabela e linhas.***

Relação = Tabela

Tupla = Linha

- Como uma ***relação é um conjunto de tuplas, podemos*** usar a notação matemática $t \in r$ para denotar que a ***tupla t*** está na ***relação r .***

Integridade Referencial

SQL

- **PRIMARY KEY** – chave primaria
- **UNIQUE KEY** – chave candidata
- **FOREYGN KEY** – chave estrangeira

Integridade Referencial

Definição Formal:

- Sejam as relações $r_1(R_1)$ e $r_2(R_2)$ com chaves primárias K_1 e K_2 , respectivamente.
- O subconjunto α de R_2 é uma **chave estrangeira referenciando K_1** na relação r_1 , se para toda relação t_2 em r_2 existir uma tupla t_1 em r_1 tal que $t_1[K_1]=t_2[\alpha]$.
- Restrições de integridade podem ser descritas :

$$\Pi_{\alpha}(r_2) \subseteq \Pi_{K_1}(r_1)$$

Integridade Referencial no Modelo ER

- Considere o conjunto de relacionamentos R entre as entidades $E1$ e $E2$. O esquema relacional para R inclui as chaves primárias $K1$ de $E1$ e $K2$ de $E2$.
- Então $K1$ e $K2$ formam chaves estrangeiras no esquema relacional de $E1$ e $E2$, respectivamente.
- O esquema de relação para uma entidade fraca deve incluir a chave primária da entidade da qual ela depende.

Modificações no Banco de Dados

- As operações de Inserção, Remoção e Atualização no Banco de Dados devem ser verificadas de maneira a preservar a seguinte restrição de integridade referencial:

$$\Pi\alpha(\mathbf{r2}) \subseteq \Pi\mathbf{K}(\mathbf{r1})$$

Modificações no Banco de Dados

- **Insert.** Se uma tupla (*linha*) $t2$ é inserida em $r2$ (*tabela*), o sistema precisa assegurar que existe uma tupla (*linha*) $t1$ em $r1$ (*na tabela*) tal que $t1[K]=t2[\alpha]$.
- *Isto é:*
$$t2[\alpha] \in \Pi K(r1)$$

Modificações no Banco de Dados

- **Delete.** Se uma tupla (*linha*) $t1$ é removida de $r1$ (*tabela*) o sistema precisa computar o conjunto de tuplas (*linhas*) em $r2$ (*na tabela*) que referencia $t1$:

$$\sigma_{\alpha} = t1[K] (r2)$$

Modificações no Banco de Dados

- **Update - Existem dois casos:**
 - Se uma tupla t_2 é atualizada na relação r_2 e a atualização modifica valores para a chave estrangeira α , então é feito um teste similar ao caso de inserção.
- Seja t_2' denotando o novo valor da tupla t_2 . O sistema deve assegurar que:

$$t_2'[\alpha] \in \Pi K(r_1)$$

Modificações no Banco de Dados

- Se uma tupla $t1$ é atualizada em $r1$, e a atualização modifica valores da chave primária (K), então um teste similar ao caso de **delete deve ser feito. O sistema** precisa computar :

$$\sigma (\alpha = t1) [K] (r2)$$

usando o antigo valor de $t1$ (o valor anterior à aplicação da atualização). Se este conjunto não é vazio, a atualização é rejeitada com um erro, ou a atualização é cascadeada nas tuplas do conjunto, ou ou as tuplas do conjunto podem ser removidas.

Chaves

- Chaves *primárias, candidatas e chaves estrangeiras* podem ser especificadas como parte da declaração **create table do SQL:**
- A cláusula **primary key da declaração create table inclui** uma lista de atributos que compreendem a chave primária.

Chaves

- Chaves *primárias, candidatas e chaves estrangeiras* podem ser especificadas como parte da declaração **create table do SQL:**
 - A cláusula **primary key da declaração create table inclui** uma lista de atributos que compreendem a chave primária.
 - A cláusula **unique key da declaração create table inclui** uma lista de atributos que compreendem a chave candidata.
 - A cláusula **foreign key da declaração create table inclui** uma lista de atributos que compreendem a chave estrangeira e o nome da relação referida pela chave estrangeira.

SQL

create table saldo

(numero-agencia char(6),

numero-conta char(10) **not null**,

saldo integer,

primary key (numero-conta),

foreign key (numero-agencia) references agencia)

Asserções

Uma ***asserção*** é um ***predicado expressando uma condição*** que queremos que o Banco de Dados sempre satisfaça.

- Casos especiais
 - Restrições de Domínio
 - Dependências funcionais
 - Restrições de Integridade

Assertões

Quando uma assertão é feita, o sistema testa a sua validade. Este teste pode introduzir uma quantidade significativa de sobrecarga; assim as assertões devem ser usadas com grande cuidado.

Procedimentos-gatilho

(**Gatilhos / Triggers**)

- Um **gatilho** é *um comando executado* automaticamente pelo sistema como um efeito de uma modificação no Banco de Dados.

Gatilho

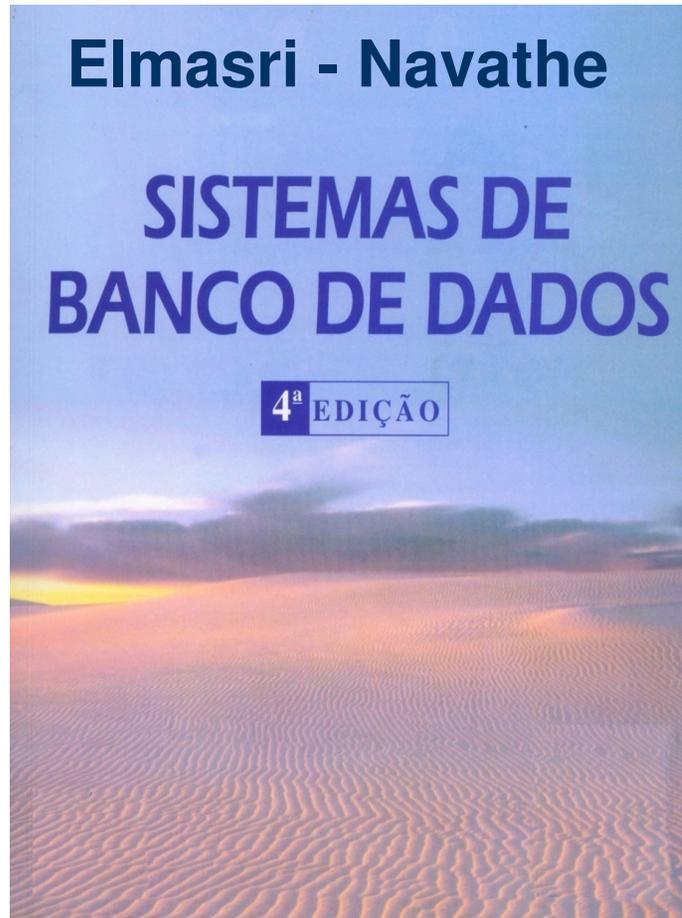
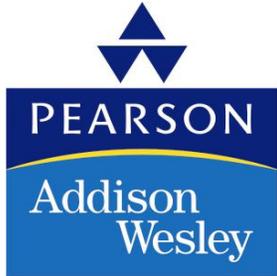
- Projeto
 - Condições de acionamento
 - Ações a executar
- Aplicações
 - Regras de Integridade
 - Outras Aplicações
 - Campos virtuais
 - Segurança
 - Medida de performance
 - Depuração de programas
 - Relatório por exceções

Gatilho

Para projetar um gatilho, é preciso:

- Especificar as **condições** sob as quais o gatilho deve ser executado.
- Especificar as **ações** a serem tomadas quando o gatilho é executado.

Considerações Finais



Atenção:

Leitura do

Capítulo 05/09:

