Fundamentos de Banco de Dados

Autorização

Márcio Silva

marcio.inacio@ufms.br

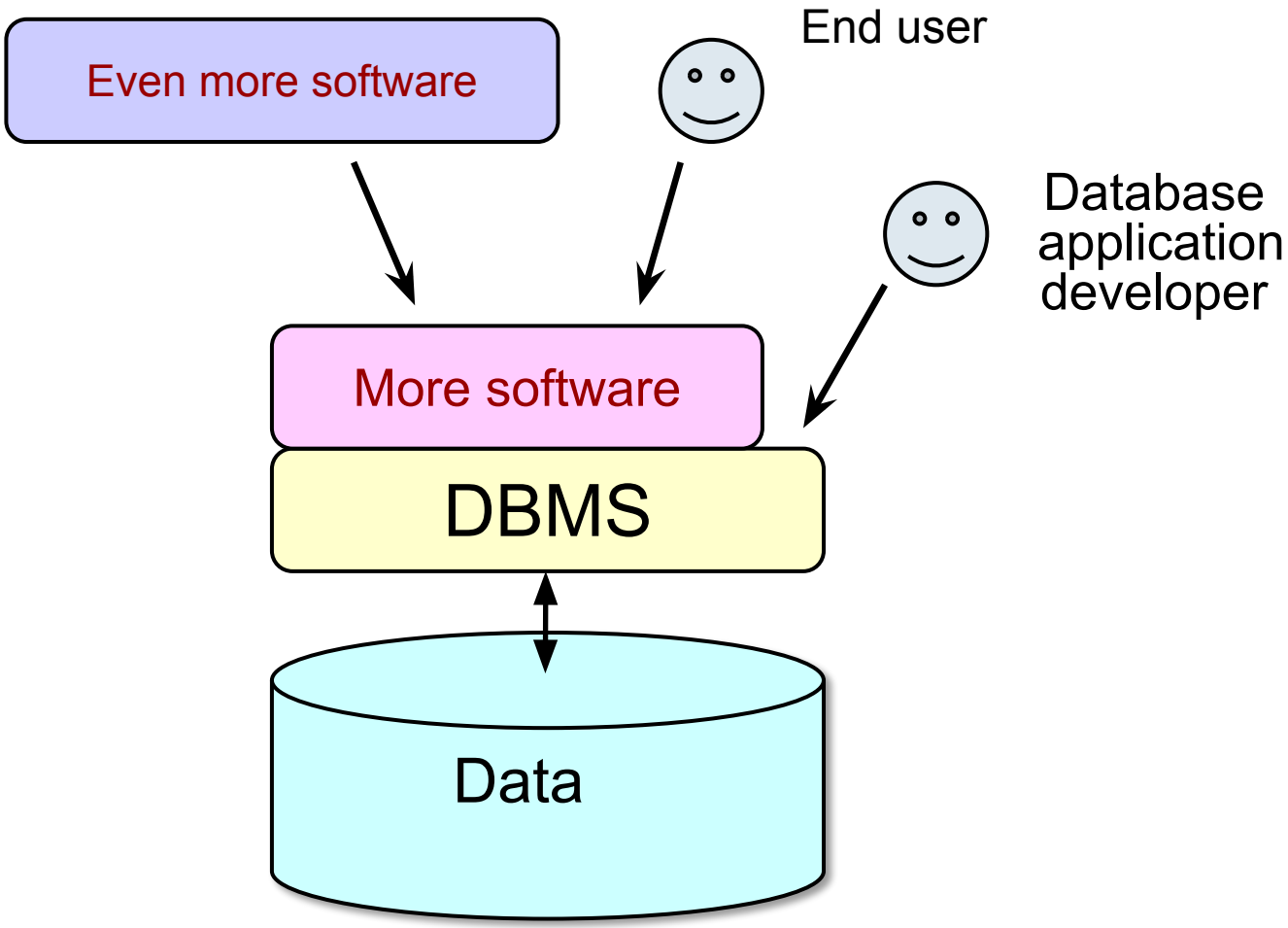
Autorização de Banco de Dados

- Garantir que usuários acessem somente o que podem realmente acessar
- Preservar o banco de dados de modificações realizadas por usuários maliciosos

Autorização de Banco de Dados

- Usuários possuem privilégios
- Somente podem operar sobre dados para os quais eles foram autorizados
  - `Select on R` or `Select(`A_1`,…,`A_n`) on R`
  - `Insert on R` or `Insert(`A_1`,…,`A_n`) on R`
  - `Update on R` or `Update(`A_1`,…,`A_n`) on R`
  - `Delete on R`
- Comandos Grant e Revoke
- Além dos privilégios a nível de tabelas
  - Use visões

Onde ficam os privilégios?



Obtenção de privilégios

- O criador da relação é o seu proprietário (owner).
- Ele tem todos os privilégios e pode conceder privilégios

```
Grant privs On R To users
  [ With Grant Option ]
```

Autorização

Formas de autorização em partes do BD:

- **Read** – permite leitura, mas não modificação dos dados.
- **Insert** – permite inserção de novos dados, mas não modificação de dados existentes.
- **Update** – permite modificação, mas não exclusão de dados.
- **Delete** – permite exclusão de dados.

Formas de autorização para modificar o esquema do BD.

- **Index** – permite a criação e exclusão de índices.
- **Resources** - permite a criação de novas relações.
- **Alteration** – permite a adição e remoção de atributos em uma relação.
- **Drop** – permite a remoção de relações.

Especificação de Autorização em SQL

- O comando **grant** é usado para dar autorização

```
grant <privilege list>
on <relation name or view name> to <user list>
```

- <user list> é:
  - um user-id
  - **public**, permite concessão de privilégios a todos os usuários válidos
  - um papel (role)
- Conceder um privilégio a uma visão não implica concessão de privilégios às relações subjacentes.
- Quem concede o privilégio já deve ter privilégio sobre o item especificado ou ser o administrador do banco de dados.

Privilégios em SQL

- **select:** permite acesso de leitura à relação ou habilidade para consultar uma visão
  - Exemplo:
    - Conceder autorização de select sobre a relação **instrutor** aos usuários U_1, U_2 and U_3:

      grant select on *instrutor* **to** U_1, U_2, U_3
- **insert**: habilidade para inserir tuplas
- **update**: habilidade para atualizar usando o comando update da linguagem SQL.
- **delete**: habilidade para deletar tuplas.
- **all privileges**: forma compacta para todos os privilégios possíveis.

No PostgreSQL

```
CREATE USER joao WITH PASSWORD 'johnwayne';
CREATE DATABASE scott;
GRANT ALL ON DATABASE scott TO joao;
ALTER USER joao PASSWORD 'tiger'

-- Usuário atual
SELECT USER;

GRANT SELECT ON mytable TO PUBLIC;
GRANT INSERT ON films TO PUBLIC;
GRANT ALL PRIVILEGES ON kinds TO manuel;
GRANT SELECT, UPDATE, INSERT ON mytable TO admin;
GRANT SELECT (col1), UPDATE (col1) ON mytable TO miriam_rw;
```

Revogação de Privilégios

```
Revoke privs On R From users
  [ Cascade | Restrict ]
```

- Cascade – também revoga privilégios concedidos a partir de privilégios sendo revogados (transitivamente), a menos que também tenham sido concedidos a partir de outra fonte.
- Restrict – desabilita a revogação através de cascade.

Revogação de Autorização em SQL

- O comando **revoke** é usado para revogar autorização.

  revoke <privilege list>

  on <relation name or view name> **from** <user list>

- Exemplo:

  revoke select on *branch* **from** U_1, U_2, U_3

- <privilege-list> pode ser **all** para revogar todos os privilégios do usuário.
- Se <user-list> incluir **public,** todos os usuários perdem o privilégio, exceto aqueles concedidos explicitamente.
- Todos os privilégios que dependem do privilégio revogado também são revogados.

Revogação no PostgreSQL

- `REVOKE INSERT ON films FROM PUBLIC;`
- `REVOKE ALL PRIVILEGES ON kinds FROM manuel;`
- `REVOKE admin FROM joe;`

Papéis (Roles)

```
create role instrutor;
grant instrutor to Amit;
```

- Privilégios podem ser concedidos a papéis:
  - `grant select on takes to instrutor;`
- Papéis podem ser concedidos a usuários, como também a outros papéis:
  - `create role monitor`
  - `grant monitor to instrutor;`
    - *Instrutor* herda todos os privilégios de monitor
- Cadeia de papéis

```
create role dean;
grant instructor to dean;
grant dean to Satoshi;
```

Papéis (Roles)

- No PostgreSQL
  - `GRANT admin TO joe;`
    - Role admin
    - User joe

Autorização em Visões

```
create view geo_instructor as
(select *
from instructor
where dept_name = 'Geology');
grant select on geo_instructor to  geo_staff
```

Outras características de Autorização

- **references** dá privilégio para criação de chave estrangeira

  grant reference (*dept_name*) **on** *department* **to** Mariano;

  - Para que pode ser usada?
- Transferência de privilégios

  grant select on *department* to Amit with grant option;

  revoke select on *department* from Amit, Satoshi cascade;

  revoke select on *department* from Amit, Satoshi restrict;
- Etc.

Referências

- Silberschatz, A., Korth, H., Sudarshan, S. “Sistema de Banco de Dados”. 5ª Edição, Editora Campus, 2006. ☐ Capítulo 4

- Elsmari, R., Navathe, Shamkant B. “Sistemas de Banco de Dados”. 6ª Edição, Pearson Brasil, 2011. ☐ Capítulo 24