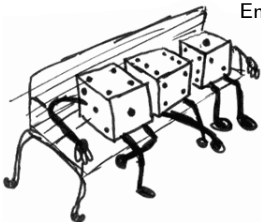


Introdução a Bancos de Dados

Aula 02 - Conceitos sobre Banco de Dados

Prof. Ramon Hugo de Souza
(ramon.hugo@ifsc.edu.br)

Instituto Federal de Santa Catarina (IFSC)
Engenharia de Telecomunicações, Campus São José
Banco de Dados (BCD029008)



1 Conceitos sobre Banco de Dados

- Conceitos Base
- Modelos de Dados
- Níveis de Abstração

2 Formalização

- Data Definition Language (DDL)
- Diagramas Entidade-Relacionamento (E-R)
- Propriedades ACID

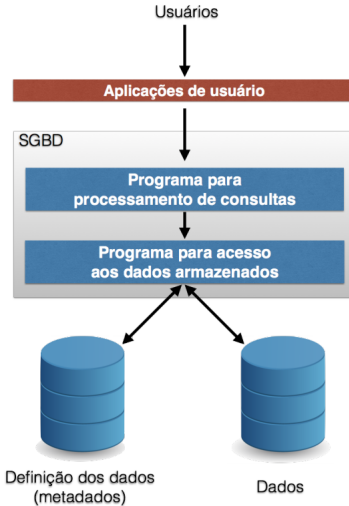
3 Big Data & NoSQL

- NoSQL
- Teorema CAP
- Modelo de Consistência Eventual (BASE)

Conceitos sobre banco de dados

- **Banco de dados** é uma coleção de dados inter-relacionados;
- **Sistema de gerenciamento de banco de dados (SGBD)** é um conjunto de programas que permitem aos usuários acessar e modificar esses dados;
- **Dicionário de dados** contém metadados (dados sobre dados).
 - Estrutura dos dados, referencial de integridade, restrições, autorização, etc.


Conceitos sobre banco de dados



Modelo de dados

- **Modelo de dados** é coleção de ferramentas para descrever os dados, seus relacionamentos, semântica e restrições de consistência.
- **Entidade-relacionamento (ER);**
 - Modelo relacional;
 - Baseado em objetos;
 - Semi-estruturado.
- Baseado na percepção do mundo real que consiste em uma coleção de objetos (entidades) e os relacionamentos entre esses objetos.

Modelo de dados

- **Modelo de dados** é coleção de ferramentas para descrever os dados, seus relacionamentos, semântica e restrições de consistência.
 - Entidade-relacionamento (ER);
 - **Modelo relacional**;
 - Baseado em objetos;
 - Semi-estruturado.
- Coleção de tabelas para representar dados e os relacionamentos entre eles;
 - Cada tabela contém registros de um mesmo tipo;
 - Cada registro define um número fixo de campos ou atributos;
 - Modelo mais usado atualmente.
- 

Modelo de dados

- **Modelo de dados** é coleção de ferramentas para descrever os dados, seus relacionamentos, semântica e restrições de consistência.
- Entidade-relacionamento (ER);
- Modelo relacional;
- Baseado em objetos;
- Semi-estruturado.
- Pode ser visto como uma extensão do modelo ER com noções de encapsulamento e identidade do objeto.

Modelo de dados

- **Modelo de dados** é coleção de ferramentas para descrever os dados, seus relacionamentos, semântica e restrições de consistência.
- Entidade-relacionamento (ER);
- Modelo relacional;
- Baseado em objetos;
- **Semi-estruturado.**
- Permite que registros do mesmo tipo possuam diferentes atributos – oposto daquilo que existe nos outros modelos;
- Exemplos: XML e JSON.

Modelo de dados

- **Modelo de dados** é coleção de ferramentas para descrever os dados, seus relacionamentos, semântica e restrições de consistência.
- Entidade-relacionamento (ER);
- Modelo relacional;
- Baseado em objetos;
- Semi-estruturado.

Esquema

Descrição de uma coleção de dados em particular com base em um modelo de dados.

Exemplo de banco de dados relacional

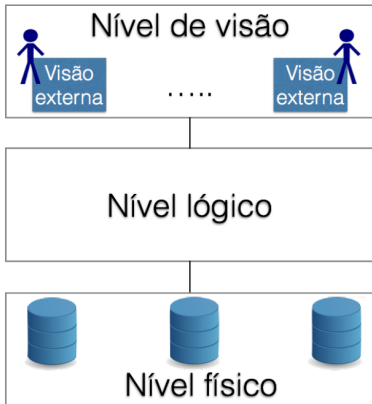
Tabela Alunos

Id	Nome	Curso
123	Juca	Telecomunicações
456	Amélia	Elétrica
789	Breno	Computação
900	Jucé	Automação

Tabela Cursos

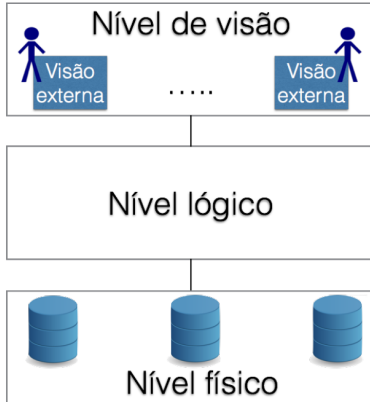
Curso	Campus	Orçamento
Telecomunicações	São José	100.000,00
Elétrica	Florianópolis	300.000,00
Computação	Lages	20.000,00
Automação	Chapecó	150.000,00

Níveis de abstração e independência de dados



- **Nível físico** – Descreve como os dados são armazenados em baixo nível.
 - Tabelas consistem em um conjunto de arquivos, cujo conteúdo não possui qualquer ordenação;
 - Índices são dados específicos que possuem uma ordenação.

Níveis de abstração e independência de dados



- **Nível de visão** – esquema externo que descreve a parte do banco de dados que um grupo de usuários pode ver.
 - Ex: O coordenador de curso só consegue ver informações sobre os alunos de seu curso.

- Conceitos Base
- Modelos de Dados
- Níveis de Asbtração

- Data Definition Language (DDL)
- Diagramas Entidade-Relacionamento (E-R)
- Propriedades ACID

- NoSQL
- Teorema CAP
- Modelo de Consistência Eventual (BASE)

Linguagem de definição de dados

Data Definition Language – DDL

- Linguagem para especificação do esquema ou estrutura do banco de dados;
- Compilador DDL gera um conjunto de tabelas em um conjunto de arquivos denominado **dicionário de dados**.

```
CREATE TABLE Alunos(  
    Id INT AUTO_INCREMENT,  
    Nome VARCHAR(60) not null,  
    Curso VARCHAR(60) null,  
    PRIMARY KEY(Id)  
);
```


Linguagem de definição de dados

Data Definition Language – DDL

- Permite acessar ou manipular dados no banco de dados;
 - Recuperação, Inserção, Exclusão ou Modificação;
- SQL (Structured Query Language) é a linguagem de consulta mais utilizada atualmente. **SQL é um padrão**

```
SELECT Nome  
FROM Alunos  
WHERE Alunos.Id = '123';
```

```
SELECT Alunos.Nome, Alunos.Curso, Cursos.Campus  
FROM Alunos, Cursos  
WHERE Alunos.Curso = Cursos.Curso  
AND Cursos.Curso = "Telecomunicacoes";
```

Algum problema com esse projeto de banco de dados?

Id	Nome	Curso	Disciplina	Professor
123	Juca	Telecomunicações	Sinais	João
456	Amélia	Elétrica	Sinais	João
789	Breno	Computação	Programação I	Martin
900	Jucé	Automação	Programação I	Martin
334	Maira	Telecomunicações	Sistemas Distribuídos	Paulo
453	Célio	Telecomunicações	Cálculo II	Luíza
112	Cícero	Computação	Cálculo II	Luíza
322	Marco	Automação	Cálculo II	Luíza
567	Alonso	Computação	Sistemas Distribuídos	Paulo
257	Luiz	Telecomunicações	Sinais	João

Algum problema com esse projeto de banco de dados?

Id	Nome	Curso	Disciplina	Professor
123	Juca	Telecomunicações	Sinais	João
456	Amélia	Elétrica	Sinais	João
789	Breno	Computação	Programação I	Martin
900	Jucé	Automação	Programação I	Martin
334	Maira	Telecomunicações	Sistemas Distribuídos	Paulo
453	Célio	Telecomunicações	Cálculo II	Luíza
112	Cícero	Computação	Cálculo II	Luíza
322	Marco	Automação	Cálculo II	Luíza
567	Alonso	Computação	Sistemas Distribuídos	Paulo
257	Luiz	Telecomunicações	Sinais	João

Normalização

Criar um conjunto de tabelas que permita armazenar informações sem redundância desnecessária

Diagramas E-R (notação baseada na UML)



Figura: Diagrama E-R

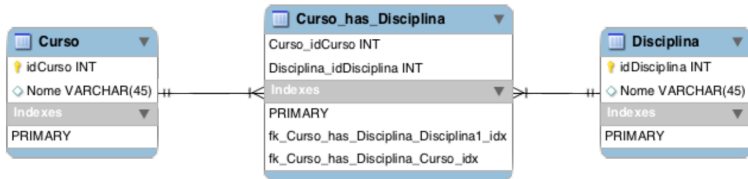


Figura: Diagrama E-R gerado pela ferramentas MySQL Workbench

Banco de dados transacional – propriedades ACID

Garante que todas operações de consulta ou de alteração são **atômicas, consistentes, isoladas e duráveis**.

- **Atomicidade:**
 - Todas operações (leitura/escrita) em uma transação são executadas com sucesso ou tudo é desfeito;
- **Consistência:**
 - A execução de uma transação leva o banco de um estado consistente para um outro estado consistente;
- **Isolamento:**
 - Transações podem acontecer de forma concorrente sem qualquer interferência;
- **Durabilidade:**
 - Ao concluir uma transação, todas modificações geradas serão persistentes.

Transação

```
read saldoContaOrigem
write (saldoContaOrigem - 500)
read saldoContaDestino
write (saldoContaDestino + 500)
```

- Antes
 - Saldo origem: 600,00
 - Saldo destino: 300,00
- Depois
 - Saldo origem: 100,00
 - Saldo destino: 800,00

Transação

```
read saldoContaOrigem
write (saldoContaOrigem - 500)
```

FALHA NO SISTEMA

```
read saldoContaDestino
write (saldoContaDestino + 500)
```

- Antes
 - Saldo origem: 600,00
 - Saldo destino: 300,00
- Depois
 - Saldo origem: 100,00
 - Saldo destino: 300,00

Transação

- SGBD garantem que a execução concorrente das transações T_1, \dots, T_n seja equivalente a uma execução serial dessas transações;
 - Antes de ler ou escrever, a transação obtém acesso a uma seção crítica e só libera depois que for concluída;
- A atomicidade e durabilidade podem ser garantidas por meio de um arquivo de registro (log);
 - Mantenha no log todas as escritas que foram concluídas;
 - Se houver alguma falha, então desfça as escritas parciais (que não foram escritas no log).

Alguns SGBDs relacionais



1 Conceitos sobre Banco de Dados

- Conceitos Base
- Modelos de Dados
- Níveis de Asbtração

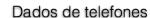
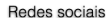
2 Formalização

- Data Definition Language (DDL)
- Diagramas Entidade-Relacionamento (E-R)
- Propriedades ACID

3 Big Data & NoSQL

- NoSQL
- Teorema CAP
- Modelo de Consistência Eventual (BASE)

- Grande **volume** de dados;
 - gerados em grande **velocidade**;
 - e com grande **variedade**;
 - que exigem novas formas de processamento para ajudar nas tomadas de decisão (alguns ainda citam **veracidade** e **valor**);
- Fundamentado sobre tecnologias de análise (Hadoop / MapReduce) e infraestrutura de armazenamento e processamento.

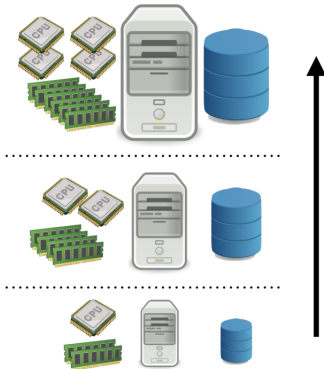


Banco de dados NoSQL

Adequado para armazenar uma vasta quantidade de dados de maneira efetiva, com baixo custo e com facilidade para implementar escalonamento horizontal (distribuição por diversos nós).

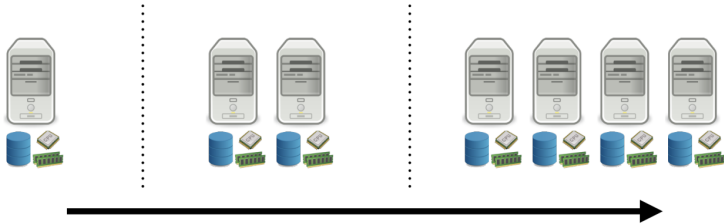
- Modelo de dados não é baseado em tabelas;
 - Em uma tabela, todas as linhas terão o mesmo número de colunas;
- Estrutura de dados em bancos NoSQL:
 - chave-valor;
 - colunas esparsas;
 - grafo;
 - orientado a documento.

Escalonamento vertical



Adiciona mais recursos (memória, cpu, disco) em um único nó para permitir que atenda o aumento da demanda.

Escalonamento horizontal

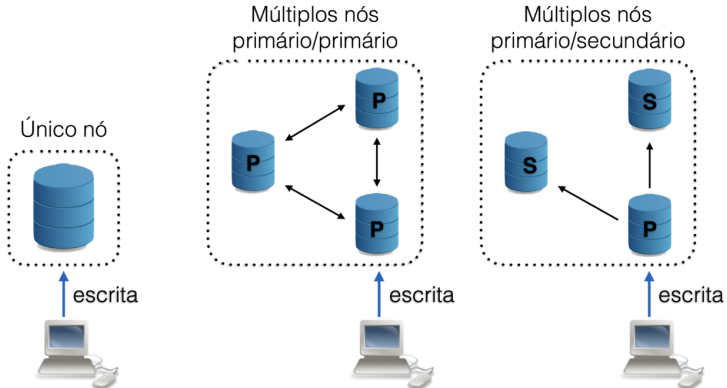


- Capacidade de processamento e armazenamento é aumentada por meio de adição de novos nós no cluster.

Escalabilidade horizontal é a opção mais barata e flexível, porém possui uma maior complexidade.

- Sistema centralizado vs sistema distribuído.

Sincronismo das réplicas



Teorema de CAP

Consistency, Availability and Partition tolerance

- **Consistência** - Toda operação de leitura em qualquer nó do cluster deve retornar a última escrita ou um erro;
- **Disponibilidade** - Todo nó não falho retornará uma resposta dentro um limite de tempo razoável, porém não há garantia que se refere a escrita mais recente;
- **Tolerância a partição** - o sistema continua a funcionar mesmo diante de mensagens perdidas/atrasadas ou falhas em alguns nós.

Em um sistema de armazenamento de dados distribuídos é impossível oferecer simultaneamente mais de duas das garantias acima.

- Diante do particionamento da rede, é necessário escolher entre consistência e disponibilidade.

- **Basic Availability**

- Todo pedido terá uma resposta, porém a resposta pode indicar uma falha na tentativa de obter o dado ou que o dado retornado está em um estado inconsistente;

- **Soft-state**

- O estado do sistema pode alterar ao longo do tempo, mesmo durante intervalo de tempo que não houve qualquer escrita;

- **Eventual consistency**

- Sistema se tornará consistente ao longo do tempo uma vez que não se tenha novas operações de escrita.

ACID vs BASE

- As propriedades do BASE são menos restritivas que as garantias do modelo ACID;
- Para o BASE pode ser mais importante garantir a disponibilidade dos dados (requisito para ambientes de larga escala) do que oferecer a garantia da consistência dos dados em todas as réplicas.

Qual modelo seria mais adequado?

- Para armazenar dados das contas bancárias de uma instituição financeira;
- Para armazenar os posts na linha do tempo em uma rede social.

Alguns bancos de dados NoSQL

APACHE
HBASE

 **Cassandra**



CouchDB
relax



riak



mongoDB

HYPERTABLE INC



Neo4j



redis

Banco de Dados

- Dúvidas?



Introdução a Bancos de Dados

Aula 02 - Conceitos sobre Banco de Dados

Prof. Ramon Hugo de Souza
(ramon.hugo@ifsc.edu.br)

Instituto Federal de Santa Catarina (IFSC)
Engenharia de Telecomunicações, Campus São José
Banco de Dados (BCD029008)

