



Instituto Federal de Educação, Ciência e Tecnologia da Bahia - IFBA  
Departamento de Informática  
Integrado / Análise e Desenvolvimento de Sistemas / Licenciatura em Computação

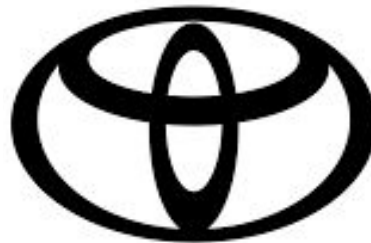
## Conceitos Básicos de Banco de Dados - PARTE 1

André L. R. Madureira <[andre.madureira@ifba.edu.br](mailto:andre.madureira@ifba.edu.br)>  
Doutorando em Ciência da Computação (UFBA)  
Mestre em Ciência da Computação (UFBA)  
Engenheiro da Computação (UFBA)

# Importância do gerenciamento de dados

---

- Sistemas necessitam de armazenamento de dados
  - Você precisa dar suporte a esses sistemas
  - Como você vai fazer isso?



# Armazenamento de dados

---

- Quase todo programa precisa armazenar dados
- Como você vai armazenar dados em seu programa?
  - **Sistemas legados:** dados são armazenados em vários arquivos separados por vários programas
  - **Sistemas DBMS:** dados armazenados usando um software DBMS que gerencia a gravação e leitura das informações



# Sistemas legados não são uma escolha razoável

---

- Sistemas legados possuem alguns problemas
  - Redundância e Inconsistência de dados
  - Integridade de dados
  - Atomicidade
  - Acesso concorrente

# Redundância e Consumo de Espaço

```
{} funcionario_deptoGERENCIA.json • {} funcionario_deptoRH.json
C: > Users > ANDRE > Desktop > {} funcionario_deptoGERENCIA.json
1  {
2    "nome": "Claudia",
3    "cpf": "111.222.333-44",
4    "endereco": "Rua ACM 5057"
5  }
```

```
{} funcionario_deptoRH.json • {} funcionario_deptoGERENCIA.json
C: > Users > ANDRE > Desktop > {} funcionario_deptoRH.json
1  {
2    "nome": "Claudia",
3    "cpf": "111.222.333-44",
4    "endereco": "Rua ACM 5057"
5  }
```

**Redundância:** Temos mais de um arquivo com mesmo conteúdo



Aumento no consumo de espaço em disco

# Redundância e Inconsistência de dados

```
{} funcionario_deptoGERENCIA.json • {} funcionario_deptoRH.json
C: > Users > ANDRE > Desktop > {} funcionario_deptoGERENCIA.json
1 {
2   "nome": "Claudia",
3   "cpf": "111.222.333-44",
4   "endereco": "Rua Apice 7484"
5 }
```

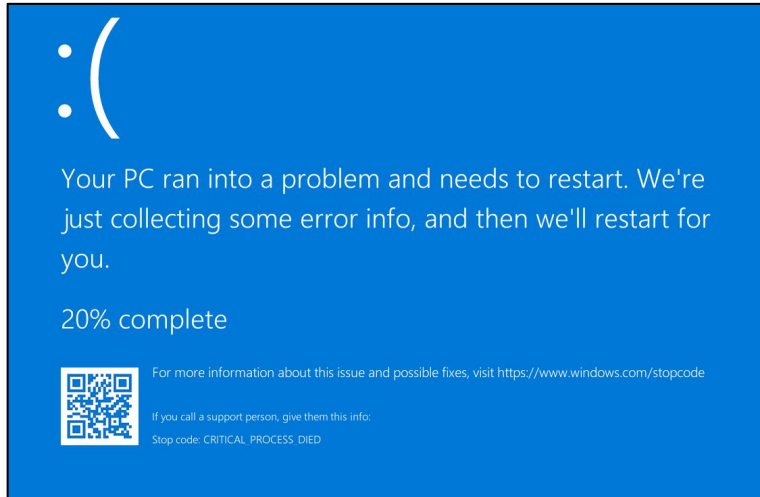
```
{} funcionario_deptoRH.json • {} funcionario_deptoGERENCIA.json
C: > Users > ANDRE > Desktop > {} funcionario_deptoRH.json
1 {
2   "nome": "Claudia",
3   "cpf": "111.222.333-44",
4   "endereco": "Rua ACM 5057"
5 }
```

**Redundância:** Temos mais de um arquivo com mesmo conteúdo



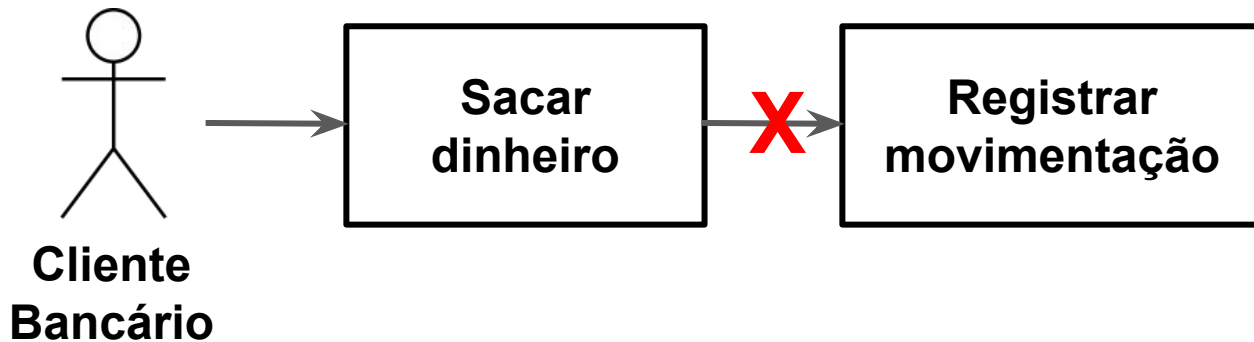
**Inconsistência de dados:** um programa pode alterar um dos arquivos sem alterar o outro

# Problemas de Integridade



Falhas durante a gravação de dados podem ocasionar perdas de dados

# Problemas de Atomicidade



## **Transação:**

Conjunto de operações que devem ser executadas juntas.

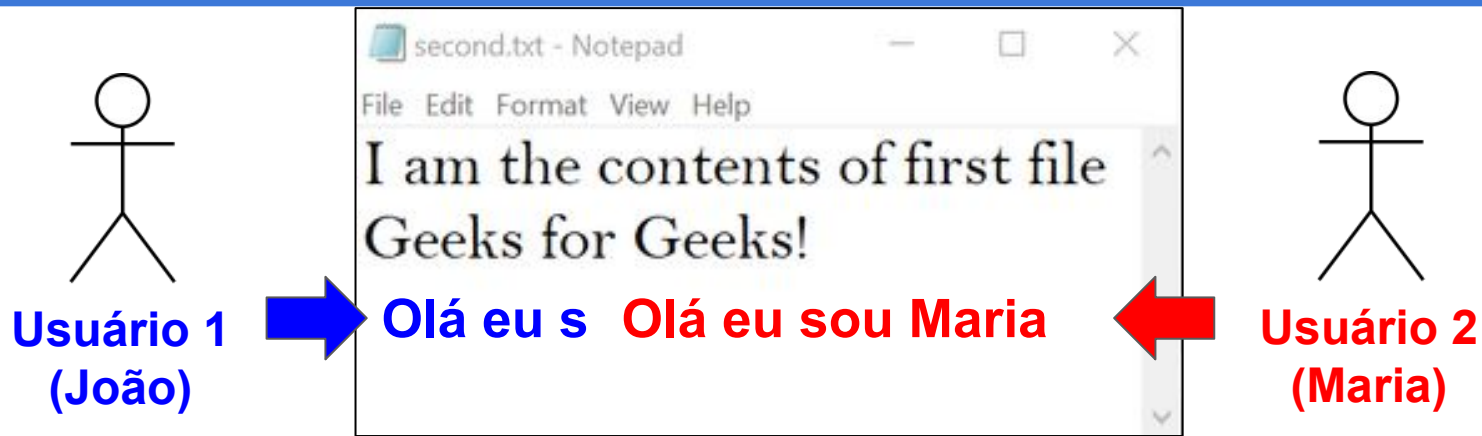
**Cada transação possui a propriedade da atomicidade.**

Isto é, ou todas as operações são realizadas com sucesso, ou nenhuma deve ser executada!

**No sistema legado não temos como executar transações!**



# Problemas com o Acesso Concorrente



**João e Maria atrapalham um  
ao outro**

O acesso simultâneo a dados  
não é suportado em sistemas  
legados!

# Sistemas de gerenciamento de bancos de dados (SGBDs ou DBMSs)

---

- Sistemas que solucionam os problemas dos sistemas legados
- SGBDs são capazes de **definir, construir, manipular, compartilhar, e proteger** informações em bancos de dados, controlando o acesso de diversos usuários e aplicações



# Definição de um banco de dados

---

- Envolve especificar os tipos, estruturas e restrições dos dados a serem armazenados
  - Essa especificação é denominada de **metadados**
- **Ex:**
  - CPF é um número inteiro (tipo de dados *INTEGER*)
  - CPF pertence a uma pessoa (é um atributo de pessoa)
  - CPF deve ser único para cada pessoa (restrição sobre o atributo)

# Construção do banco de dados

- É o processo de armazenar os dados em algum meio controlado pelo SGBD
  - **Ex:** tabelas, classes e objetos, arquivos XML ou JSON

## ALUNO

Nome	Numero_aluno	Tipo_aluno	Curso
Silva	17	1	CC
Braga	8	2	CC

```
1  {
2    "nome": "Bruno",
3    "sobrenome": null,
4    "idade": 20,
5    "ativo": true,
6    "endereco": {
7      "rua": "Rua Vinte e Um de Abril",
8      "numero": 18
9    },
10   "turmas": [
11     "Programação Orientada a Objetos",
12     "Lógica de Programação"
13   ]
14 }
```

# Manipulação do banco de dados

---

- Inclui funções como:
  - Consulta ao banco de dados para recuperar dados
  - Atualização de informações
  - Geração de relatórios
- **Ex:** Busque pelos alunos matriculados na disciplina “Geografia”
  - ***SELECT \* FROM alunos WHERE disciplina = 'Geografia'***
- **Ex:** Atualize as notas dos alunos da disciplina “Geografia” para 6,0
  - ***UPDATE alunos SET nota = 6.0 WHERE disciplina = 'Geografia'***

# Compartilhamento de dados

---

- Permite que diversos usuários e programas acessem o banco de dados simultaneamente
  - **Ex:** O banco de dados de uma grande empresa como o Banco do Brasil é acessado por diversos funcionários (e clientes) ao mesmo tempo
    - SGBD controla quando cada usuário pode acessar dados para evitar problemas de integridade
    - **Exemplo de problema de integridade:** usuários João e Maria alterando as mesmas informações ao mesmo tempo

# Proteção de dados

---

- Controla os usuários e programas que podem acessar determinadas informações do banco de dados
  - **Ex:** Gerentes de banco podem consultar o extrato das contas, mas eles não têm acesso a senha dos clientes
- Define que operações cada usuário pode realizar sobre o banco de dados
  - **Ex:** Gerente pode realizar um empréstimo, enquanto que um cliente pode realizar um saque no caixa eletrônico

# Manutenção dos dados

---

- SGBDs possuem mecanismos para recuperação do banco de dados, seja devido a uma falha ou operação indevida realizada por um usuário
  - **Ex:**
    - Logs de transação
    - Backup e Recuperação
      - Online ou Offline
      - On-site ou Off-site



# Exemplo de Banco de Dados de uma Universidade

## DISCIPLINA

Nome_ disciplina	Numero_ disciplina	Creditos	Departamento
Introd. à ciência da computação	CC1310	4	CC
Estruturas de dados	CC3320	4	CC
Matemática discreta	MAT2410	3	MAT
Banco de dados	CC3380	3	CC

## ALUNO

Nome	Numero_aluno	Tipo_aluno	Curso
Silva	17	1	CC
Braga	8	2	CC

## TURMA

Identificacao_ turma	Numero_ disciplina	Semestre	Ano	Professor
85	MAT2410	Segundo	07	Kleber
92	CC1310	Segundo	07	Anderson
102	CC3320	Primeiro	08	Carlos
112	MAT2410	Segundo	08	Chang
119	CC1310	Segundo	08	Anderson
135	CC3380	Segundo	08	Santos

## HISTORICO\_ESCOLAR

Numero_aluno	Identificacao_turma	Nota
17	112	B
17	119	C
8	85	A

# Tipos de Usuários de SGBDs

---

- SGBDs possuem diversos tipos de usuários, cada um com necessidades específicas:
  - **Administradores de bancos de dados (*DBAs*)**
  - **Projetistas de banco de dados**
  - **Usuários finais**
  - **Analistas de sistemas e Programadores de aplicações**

# Administradores de bancos de dados (*DBAs*)

---

- Supervisionam e gerenciam recursos do sistema, sendo responsáveis por:
  - Autorizar o acesso ao DB para outros usuários
  - Coordenar e monitorar o uso do DB
  - Adquirir recursos de software e hardware
  - Corrigir falhas na segurança
  - Melhorar o desempenho do sistema (tempo de resposta)



# Projetistas de bancos de dados

---

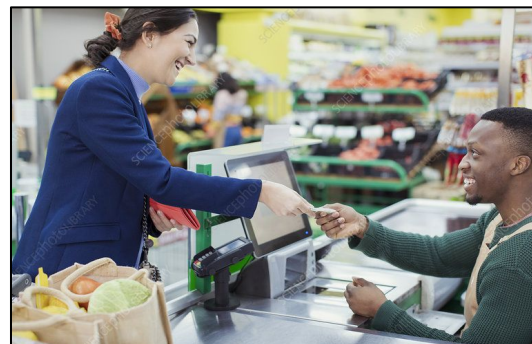
- São responsáveis por:
  - Identificar os dados a serem armazenados
  - Escolher estruturas apropriadas para representar e armazenar esses dados
- Se comunicam com os usuários finais do sistema para:
  - Entender as necessidades de todos os usuários
  - Criar um projeto de DB que atenda a todas as demandas



# Usuários finais

---

- São pessoas cujas funções exigem acesso ao banco de dados para:
  - Realizar consultas
  - Atualizar dados
  - Gerar relatórios
- **Ex:** Alunos e professores do IFBA, Clientes e gerentes de um banco, Caixas e gerentes de supermercado, etc
- Os bancos de dados (DBs) existem primariamente para atender os usuários finais !



# Usuários finais

- Podem ser classificados como:

- **Usuários finais casuais:**

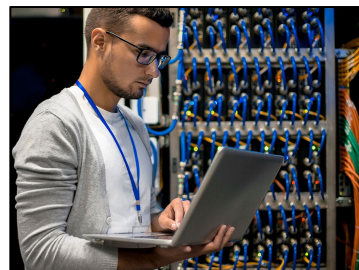
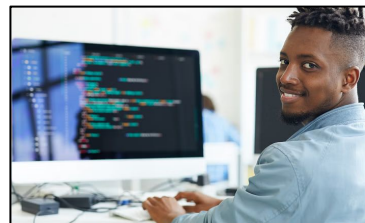
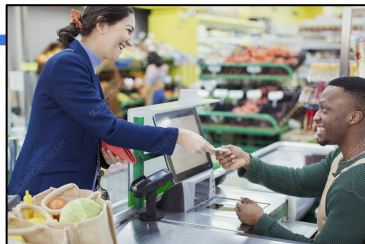
- Ocasionalmente acessam o banco de dados

- **Usuários finais iniciantes:**

- Consultam e atualizam o banco de dados constantemente

- **Usuários finais sofisticados (ou avançados):**

- Profundamente familiarizados com o SGBD
- Podem implementar as próprias aplicações



# Analistas de sistemas e Programadores de aplicações

---

- **Analistas de sistemas:**

- Identificam as necessidades dos usuários finais
- Definem as especificações do sistema



- **Programadores de aplicações:**

- Implementam essas especificações como programas
- Testam, depuram, documentam e mantêm o sistema



- **Ambos (analistas e programadores) precisam ter profundo conhecimento sobre os recursos fornecidos pelo SGBD**

# Armazenamento de informações em um BD

---

- Para armazenar dados precisamos de **estruturas de dados** de alto desempenho
  - **Estruturas de dados:** Formas de organizar e manipular dados
  - As estruturas de dados de DBMSs geralmente são complexas (difíceis de manipular)
  - Como organizar os dados que serão armazenados?



# Estruturas de dados e Alto desempenho

---

- **Problema:**

- Uso de estruturas de dados complexas dificulta o acesso aos dados
- Usuários não necessariamente precisam conhecer detalhes de implementação do DB para ter acesso aos dados

- **Solução:**

- Simplificar o uso do SGBD e o acesso às estruturas de dados

View ProximosVoos

Cia Aerea	Horario
Latam	11:10
Gol	11:15
Azul	11:20

View Empresa

Cia Aerea	Itinerarios
Latam	95
Gol	87
Azul	92

### Nível de View

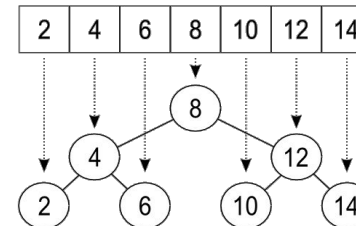
Tabela ProximosVoos

Cia Aerea	Horario	Preço
Latam	11:10	R\$ 100
Gol	11:15	R\$ 110
Azul	11:20	R\$ 95

Tabela Empresa

Cia Aerea	Funcionarios	Aviões	Itinerarios
Latam	1200	20	95
Gol	2000	35	87
Azul	1800	26	92

### Nível Lógico



### Nível Físico

# Níveis de Abstração

- **Nível de view (ou externo):**

- **Que** dados um usuário deve ter acesso

View ProximosVoos

Cia Aerea	Horario
Latam	11:10
Gol	11:15
Azul	11:20

Nível de View

- **Nível lógico (ou conceitual):**

- **Quais** dados estão armazenados no banco de dados e as suas relações

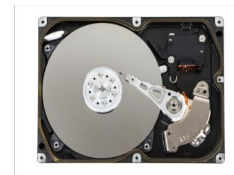
Tabela ProximosVoos

Cia Aerea	Horario	Preço
Latam	11:10	R\$ 100
Gol	11:15	R\$ 110
Azul	11:20	R\$ 95

Nível Lógico

- **Nível físico (ou interno):**

- **Como** os dados são armazenados



Nível Físico

# Instância x Esquema

**Tabela Funcionários**

Nome	Salario	Idade
Ana	1500	29
Jose	1212	21

Esquema

Instâncias

## Esquema:

Estrutura (Projeto) do banco de dados (DB)

Raramente muda

## Instância:

Informações armazenadas no DB.

Muda com frequência

# Níveis de Abstração de Esquemas

- **Esquema**

- **Esquema físico**

- Descreve o nível físico

- **Esquema lógico**

- Descreve o nível lógico

- **Esquema de view**

- Descreve o nível de view

View ProximosVoos

Cia Aerea	Horario
Latam	11:10
Gol	11:15
Azul	11:20

Nível de View

Tabela ProximosVoos

Cia Aerea	Horario	Preço
Latam	11:10	R\$ 100
Gol	11:15	R\$ 110
Azul	11:20	R\$ 95

Nível Lógico



Nível Físico

# Abstração e Independência de Dados

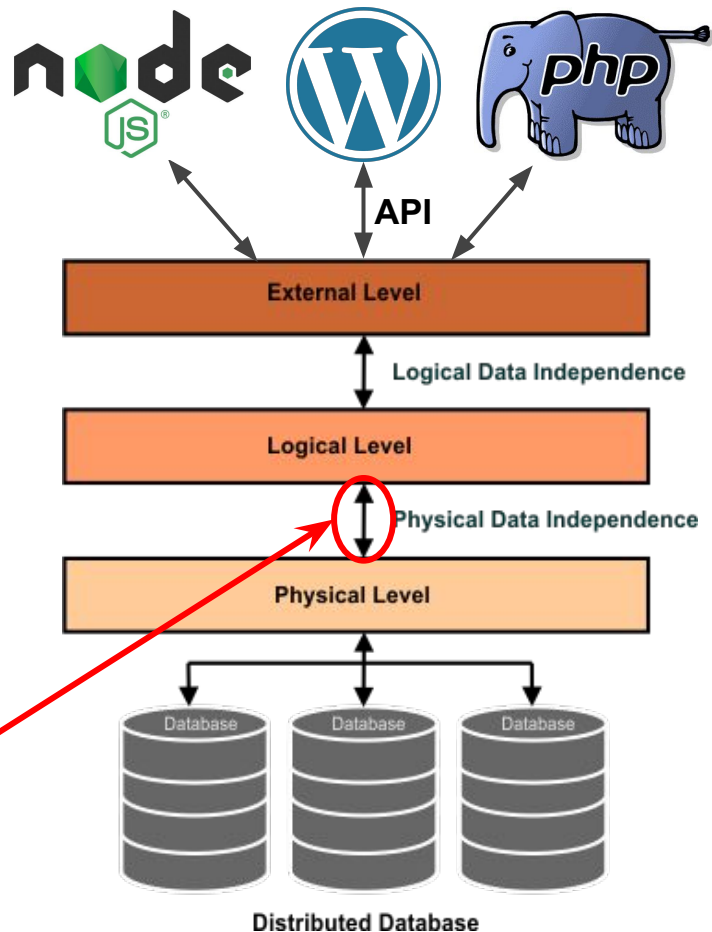
---

- Os níveis de abstração facilitam o acesso e uso do DB pelos usuários
  - Para isso precisamos que haja **independência de dados**
- Essa independência pode ser classificada em:
  - **Independência de dados física**
  - **Independência de dados lógica**

# Independência de Dados Física

- Habilidade de mudar o esquema físico sem incorrer em alterações no esquema lógico
- **Ex:** compressão de dados, criptografia, índices, hashing, etc

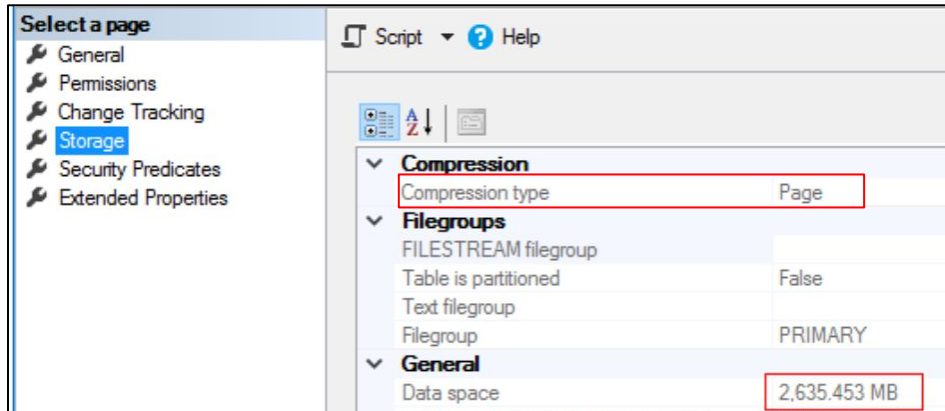
**Nível lógico não precisa mudar seu acesso aos dados**



# Exemplo de Independência de Dados Física

Gerente				
id	nome	cpf	agencia	conta_gerenciada
1	Juan	111.222.333-44	4200	51002
2	Hebert	555.666.777-88	7171	95620
3	Claudia	123.456.789-00	1952	61230

**Nível lógico**



Se eu ativar a compressão de dados no nível físico, não preciso alterar nada no nível lógico.

**Nível físico**

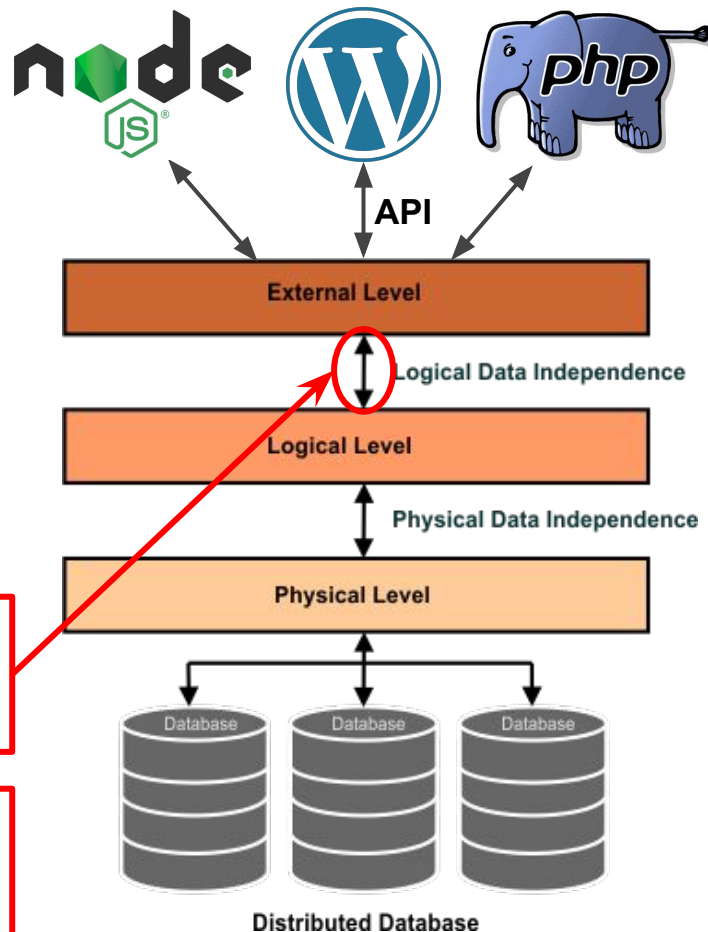


# Independência de Dados Lógica

- Habilidade de alterar o esquema lógico sem incorrer em mudanças no esquema de view
- **Ex:** adicionar campos, tabelas, etc

**Aplicação não precisa mudar  
seu acesso aos dados**

Não precisamos reescrever a aplicação quando  
o esquema de view é modificado



# Exemplo de Independência de Dados Lógica

Gerente		
id	nome	conta_gerenciada
1	Juan	51002
2	Hebert	95620
3	Claudia	61230

**Nível de view**

Gerente				
id	nome	cpf	<del>agencia</del>	conta_gerenciada
1	Juan	111.222.333-44	<del>4200</del>	51002
2	Hebert	555.666.777-88	<del>7171</del>	95620
3	Claudia	123.456.789-00	<del>1952</del>	61230

Se eu remover o atributo “agência” ou “cpf”, não preciso alterar nada no nível de view

**Nível de lógico**

# Exemplo de Independência de Dados Lógica

Gerente		
id	nome	conta_gerenciada
1	Juan	51002
2	Hebert	95620
3	Claudia	61230

**Nível de view**

Gerente					
id	nome	cpf	agencia	conta_gerenciada	supervisor
1	Juan	111.222.333-44	4200	51002	João
2	Hebert	555.666.777-88	7171	95620	Maria
3	Claudia	123.456.789-00	1952	61230	Kleber

**Nível de lógico**

Se eu adicionar um atributo como “supervisor”, não preciso alterar nada no nível de view

# Exemplo de **Ausência** de Independência de Dados Lógica

Gerente		
id	nome	conta_gerenciada
1	Juan	51002
2	Hebert	95620
3	Claudia	61230

**Nível de view**

Gerente				
id	nome	cpf	agencia	<del>conta_gerenciada</del>
1	Juan	111.222.333-44	4200	51002
2	Hebert	555.666.777-88	7171	95620
3	Claudia	123.456.789-00	1952	61230

Se eu remover o atributo  
“conta\_gerenciada” precisarei  
alterar o nível de view.  
**Logo NAO temos independência  
lógica nesse caso.**

**Nível de lógico**

# Referencial Bibliográfico

---

- KORTH, H.; SILBERSCHATZ, A.; SUDARSHAN, S. **Sistemas de bancos de dados**. 5. ed. Rio de Janeiro: Ed. Campus, 2006.
- DATE, C. J. **Introdução a sistemas de bancos de dados**. Rio de Janeiro: Ed. Campus, 2004. Tradução da 8ª edição americana.