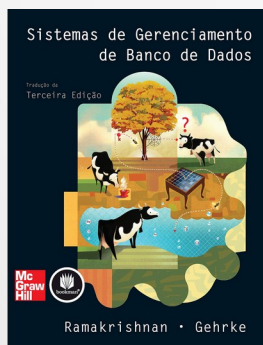


Banco de Dados I

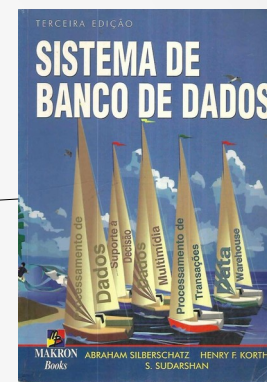


Introdução

Cap. 1 e 2 (Elmasri)

Cap. 1 (Silberschatz)

Cap. 1 (Ramakrishnan)




Denio Duarte
duarte@uffs.edu.br



Motivação

DENIO DUARTE  

 [Início](#) > [Graduação](#) > [Diário de Classe](#) > [Registrar Encontros e Faltas](#) > [Encontros](#)

Graduação

▼ Diário de Classe


Acadêmicos com Situação INC
Consultar Diário
Encerrar Diário
Registrar Avaliações e Notas
Registrar Encontros e Faltas

► Plano de Ensino

► Relatórios

► Atividades Docentes

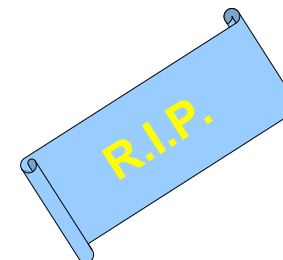
► Configurações

Campus: Chapecó
Turma: 28674 - Ciência da Computação - 4ª fase (Remoto) - Vespertino - 2020/2
Comp. Curricular: GEX090 - Banco de dados I
Frequência Mínima Exigida: 75%
Situação do Plano de Ensino: Em edição 

Professor(es)	
Nome	Créditos
DENIO DUARTE	4



Importar encontros do Plano de Ensino



Encontros

Total de períodos previstos: 72
Total de períodos cadastrados: 0
Total de períodos faltantes: 72

Novo Encontro 

Nº	Datas	Turno	Conteúdo	Períodos*	Alterar	Excluir	Faltas
				Total			

Nenhum encontro cadastrado para a turma.

Total de períodos cadastrados: 0

Legenda:

PT - Presencial teórica | PP - Presencial prática | PCCr - Prática como componente curricular | NP - Não presencial | Est - Atividade de estágio | Pes - Atividade de pesquisa | Ext - Atividade de extensão.

Turno: M - Manhã | T - Tarde | N - Noite | (-) Turno não definido.

* Cada período de aula equivale a 50 minutos.

Motivação

DENIO DUARTE

[Início](#) > [Graduação](#) > [Diário de Classe](#) > [Registrar Encontros e Faltas](#) > Encontros

Graduação

- ▼ **Diário de Classe**
 - Acadêmicos com Situação INC
 - Consultar Diário
 - Encerrar Diário
 - Registrar Avaliações e Notas
 - Registrar Encontros e Faltas
- ▶ **Plano de Ensino**
- ▶ **Relatórios**

▶ **Atividades Docentes**

▶ **Configurações**

Campus: Chapecó
Turma: 28674 - Ciência da Computação - 4ª fase (Remoto) - Vespertino - 2020/2
Comp. Curricular: GEX090 - Banco de dados I
Frequência Mínima Exigida: 75%
Situação do Plano de Ensino: Em edição ?

Professor(es)	
Nome	Créditos
DENIO DUARTE	4

Importar encontros do Plano de Ensino

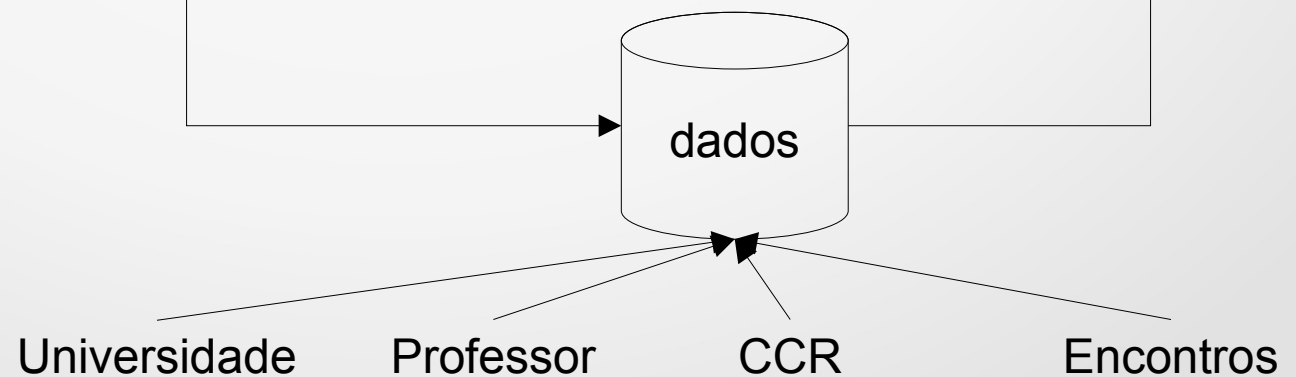
Encontros

Total de períodos previstos: 72
Total de períodos cadastrados: 0
Total de períodos faltantes: 72

Novo Encontro +

Nº	Datas	Turno	Conteúdo	Períodos*	Alterar	Excluir	Faltas
				Total			
Nenhum encontro cadastrado para a turma.							
Total de períodos cadastrados:				0			

Legenda:
PT - Presencial teórica | PP - Presencial prática | PCCR - Prática como componente curricular | NP - Não presencial | Est - Atividade de estágio | Pes - Atividade de pesquisa | Ext - Atividade de extensão.
Turno: M - Manhã | T - Tarde | N - Noite | (-) Turno não definido.
* Cada período de aula equivale a 50 minutos.



Introdução

- Dados
 - Podemos definir como a menor característica de um objeto
 - Nome, idade, cor, altura, potência, valor, dimensão, coordenada x, capacidade ...
 - Os dados têm um domínio associado:
 - Nome é representado por uma cadeia de caracteres
 - Idade é representado por um inteiro entre 0 e 110
 - Cor pode ser azul, vermelho e verde
 - ...
 - Mais adiante, chamaremos de **atributo**

Introdução

- Dados
 - Porém os dados estão "soltos" por aí – **Bando de Dados**

Gustavo Kuerten
Magda Cotrofe
Salvador Dali
Rosangela Santos

Banco de Dados I
Sistemas Operacionais
Fisiologia I
História Medieval

Ciência da Computação
Educação Física
História

2012-1
2012-2
2013-1

1,90	M	10/09/1976
1,75	F	18/01/1963
1,68	M	11/05/1904
1,65	F	20/12/1990

6,3
8,0
5,6
5,5
7

- Temos que organizá-los para serem úteis

BD R

BD S

Introdução

- Banco de Dados
 - Conjunto de dados integrados e relacionados que tem como objetivo atender uma comunidade de usuários.
 - Propriedades implícitas
 - Representa aspectos do mundo real (**minimundo** ou **universo de discurso**).
 - **Coleção** de dados **logicamente coerentes** com algum **significado** inerente.
 - Projetado, construído e povoado (**instanciado**) para aplicações específicas

Introdução

- Banco de Dados
 - Exemplos de minimundos ou universos de discurso
 - Universidade
 - **Acadêmico**: preocupado com os dados dos alunos, componentes curriculares, matrículas, professores, etc
 - **Gestão de Pessoas**: preocupado com os servidores (professores e técnicos administrativos, titulação, cargo, horas trabalhadas, férias, etc)

Introdução

- Banco de Dados (formato relacional)
 - Acadêmico

Mat Nome		Altura	Sexo	Data Nasc	Curso
11	Gustavo Kuerten	1,90	M	10/09/1976	1
22	Magda Cotrofe	1,75	F	18/01/1963	3
33	Salvador Dali	1,68	M	11/05/1904	1
44	Rosangela Santos	1,65	F	20/12/1990	2

[Ver Dados](#)

Código Nome

1	Ciência da Computação
2	Educação Física
3	História

Código	Nome	Semestre	Mat	Disciplina	Média
1	Banco de Dados I	2012-1	11	1	7,5
2	Sistemas Operacionais	2012-2	22	4	5,6
3	Fisiologia I	2013-1	33	2	8,0
4	História Medieval	2013-1	22	4	6,3

Introdução

- Banco de Dados (formato semi-estruturado)

```
{ Estudante: [ { matricula: 11,
                 nome: "Gustavo Kuerten",
                 altura: 1.9,
                 sexo: "M",
                 dtnasc: "1976-09-10",
                 curso: {codigo:1, nome: "Ciência da Computação"},
                 historico:[
                     {ccr: "Banco de Dados I", semestre: "2012-1", media: 7.5}
                 ]
               },
               { matricula: 33,
                 nome: "Salvador Dali",
                 altura: 1.68,
                 sexo: "M",
                 dtnasc: "1904-05-11",
                 curso: {codigo:1, nome: "Ciência da Computação"},
                 historico:[
                     {ccr: "Sistemas Operacionais", semestre: "2013-1", media: 8.5}
                 ]
               }
            ]
}
```

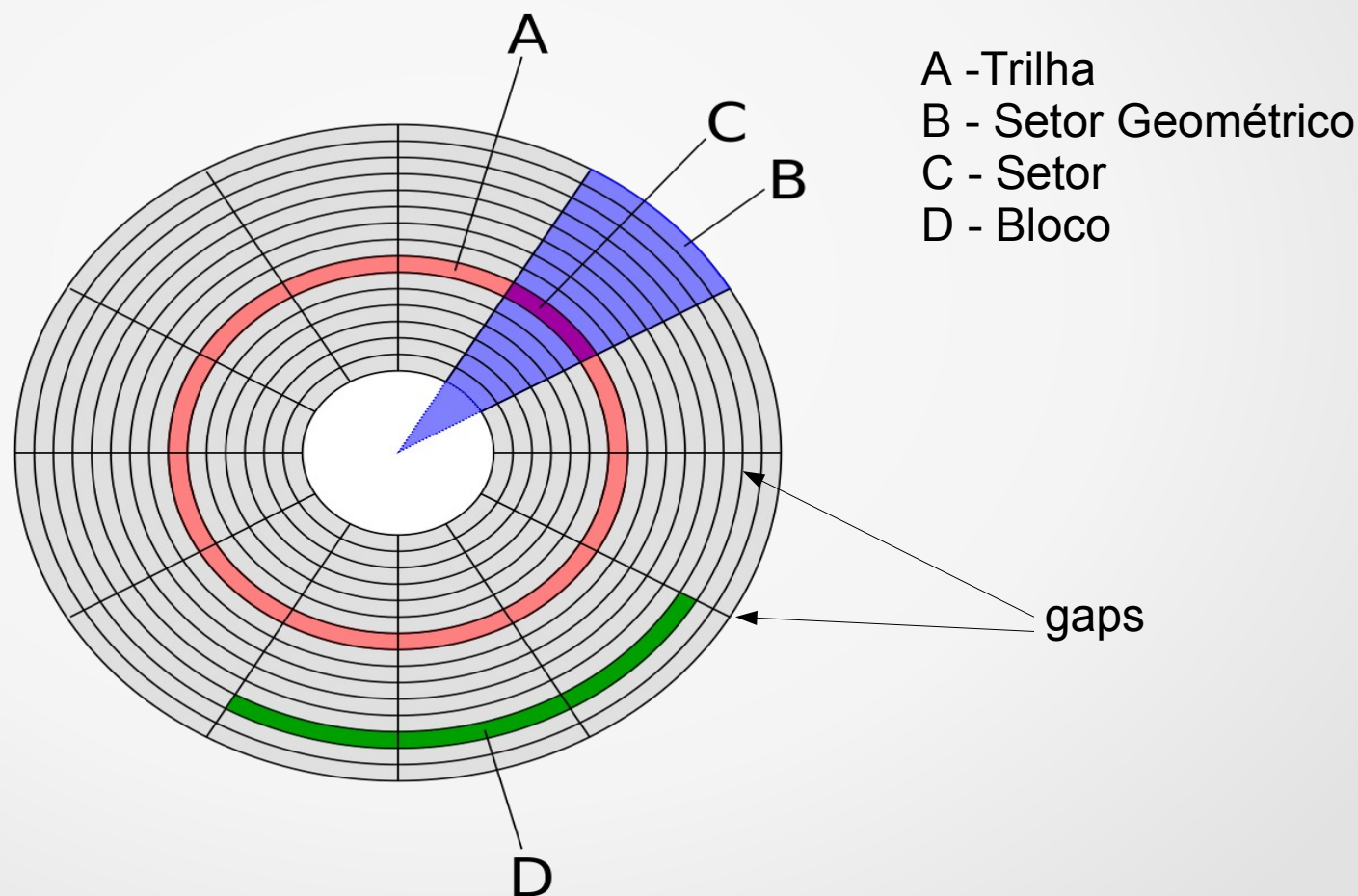
[Ver Dados](#)

Introdução

- Questões para refletir:
 - Como os dados estão organizados no disco?
 - Como acessar os dados armazenados?
 - Como estes programas "enxergam" os dados no disco?

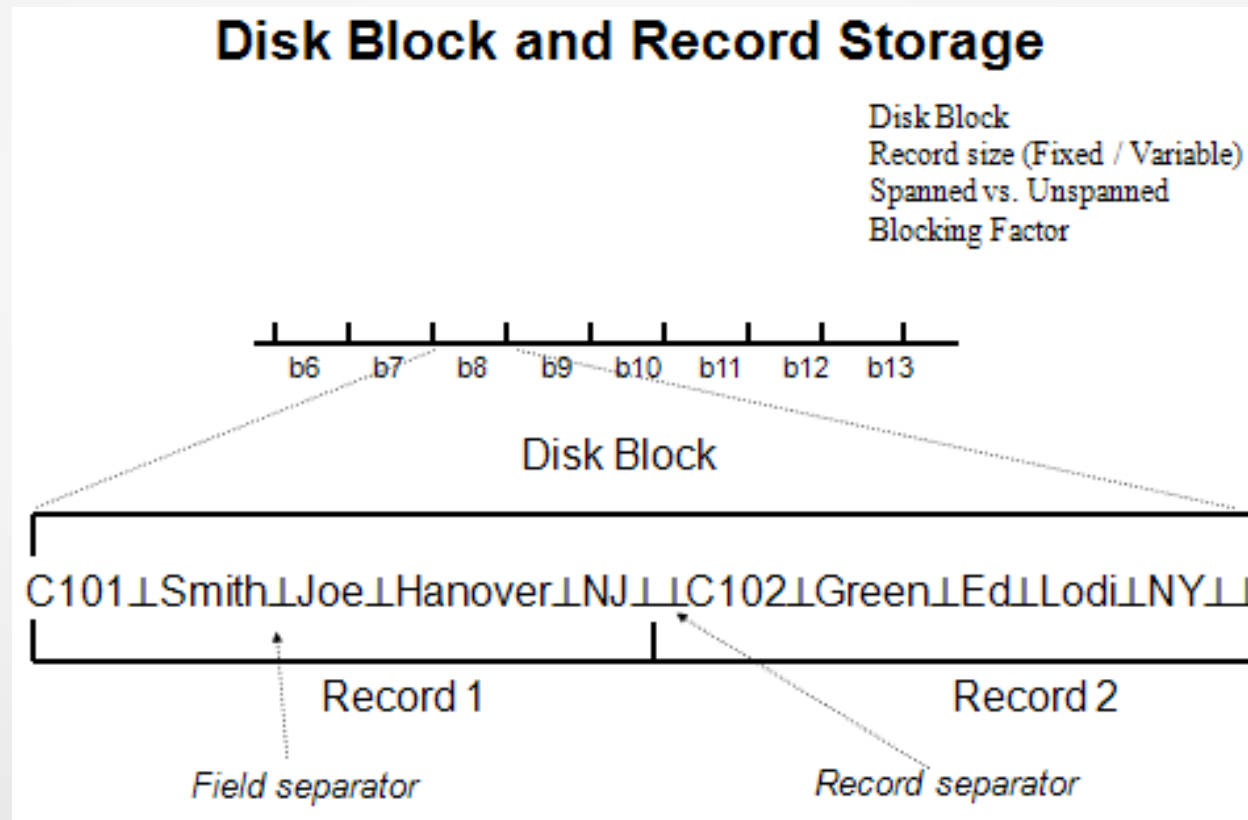
Introdução

- Como os dados estão organizados no disco?



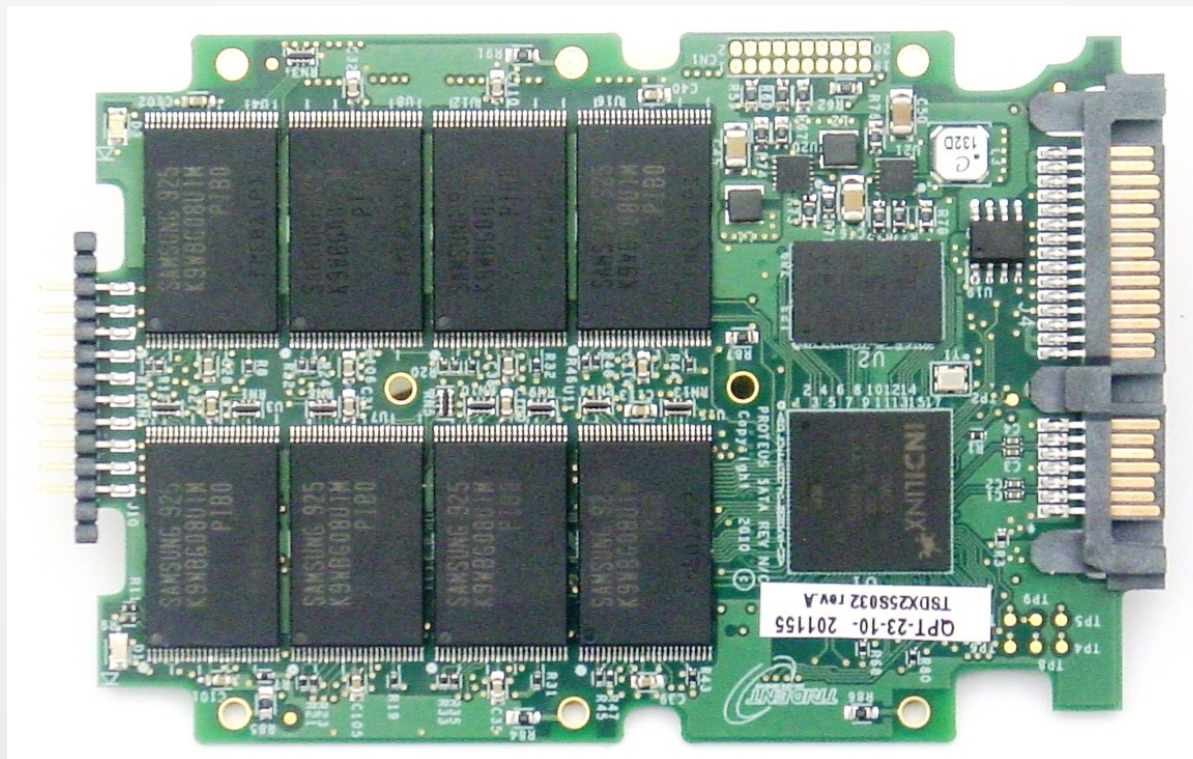
Introdução

- Como os dados estão organizados no disco?



Introdução

- SSD (Solid State Drive) começam a substituir os discos tradicionais



Introdução

- SSD Vantagens
 - Sem tempo de busca (seek time)
 - Sem latência de rotação (rotational latency)
- SSD Desvantagens
 - Custo de escrita e leitura são diferentes
 - Todos os algoritmos de acesso a disco e buffer de um SGBD devem ser adequados ao funcionamento dos SSDs.

Introdução

- Como acessar os dados armazenados?
 - Depende da interface utilizada pelo programador:
 - **Diretamente**
 - Pouco utilizada (necessário conhecer o hardware)
 - **Sistema Operacional**
 - Utiliza a camada Sistema de Arquivos
 - **Sistema Gerenciador de Banco de Dados (SGBD)**
 - Interface que separa o programa do banco de dados

Introdução

- Como estes programas "enxergam" os dados no disco?
 - Os dados estão organizados no disco como um sequência de bytes
 - Esse formato não é interessante para os programas
 - Organização na memória RAM:
 - Lista encadeada
 - Vetor
 - Árvore
 - Pilha, etc

Introdução

- Dois pontos nos interessam
 - Como acessar os dados
 - Como os dados são vistos pelos programas (ou como são organizados pela interface)
 - Chamaremos **Modelo de Dados**

Como Acessar os Dados

- Sistema de arquivos
 - Não utilizam software específico para gerenciar os dados
 - Utilizam rotinas do sistema operacional
 - Nenhuma instalação adicional para o uso
 - Problema com redundância, segurança, confiança
 - Dados são sequência de bytes
 - Volume pequeno de dados

Como Acessar os Dados

- Sistema de arquivos

- C

```
struct rec {  
    int cod;  
    char nome[20]; };  
struct rec myrec;  
FILE *f;
```

```
f=open("meuarquivo.dat","r+");  
if (f==NULL) exit(0);  
myrec.cod=1;  
strcpy(myrec.nome,"Cliente Um");  
fwrite(&myrec, sizeof(struct rec),1,f);
```

```
fseek(f,0,SEEK_SET);  
fread(&myrec,sizeof(myrec),1,f);  
printf("Codigo: %d Nome: %s",myrec.cod,myrec.nome);  
fclose(f);
```

Como Acessar os Dados

- Sistema Gerenciador de Banco de Dados SGBD
 - Grande volume de dados
 - Oferece controle de redundância, segurança e confiança
 - Necessita a instalação do SGBD para o uso
 - Linguagens de programação têm que ter bibliotecas de acesso

Como Acessar os Dados

- SGBD
 - Relacional

```
create table cliente(  
  cod integer not null primary key,  
  nome varchar(20) not null  
);  
:  
insert into cliente (cod,nome) values (1,'Cliente Um');  
:  
select * from cliente;  
:
```

Como Acessar os Dados

• SGBD vs Sistema de Arquivo

```
create table cliente(  
  cod integer not null primary key,  
  nome varchar(20) not null  
);
```

```
  :  
insert into cliente (cod,nome)  
  values (1,'Cliente Um');
```

```
  :  
select * from cliente;
```

```
  :
```

```
struct rec {  
  int cod;  
  char nome[20];}
```

```
struct rec myrec;  
FILE *f
```

```
X f=open("meuarquivo.dat","r+");  
if f==NULL exit(0);
```

```
myrec.cod=1  
strcpy(myrec.nome,"Cliente Um");  
fwrite(&myrec, sizeof(struct rec),1);
```

```
:
```

```
fseek(f,0,SEEK_SET);  
fread(&myrec,sizeof(myrec),1,f);  
printf("Codigo: %d Nome: %s",myrec.cod,myrec.nome);  
fclose(f);
```

Modelo de Dados

- Os SGBDs tratam seus dados de várias formas
 - Essas formas são chamadas de modelo de dados (como os dados são organizados para os programas acessarem)
 - Hierárquico
 - Redes
 - Relacional
 - Orientado a objetos
 - Objeto-Relacional
 - Semi-estruturado
 -

Modelos de Dados (1a Geração)

- Modelo Hierárquico (IMS – IBM)
 - Baseado em estrutura de árvores

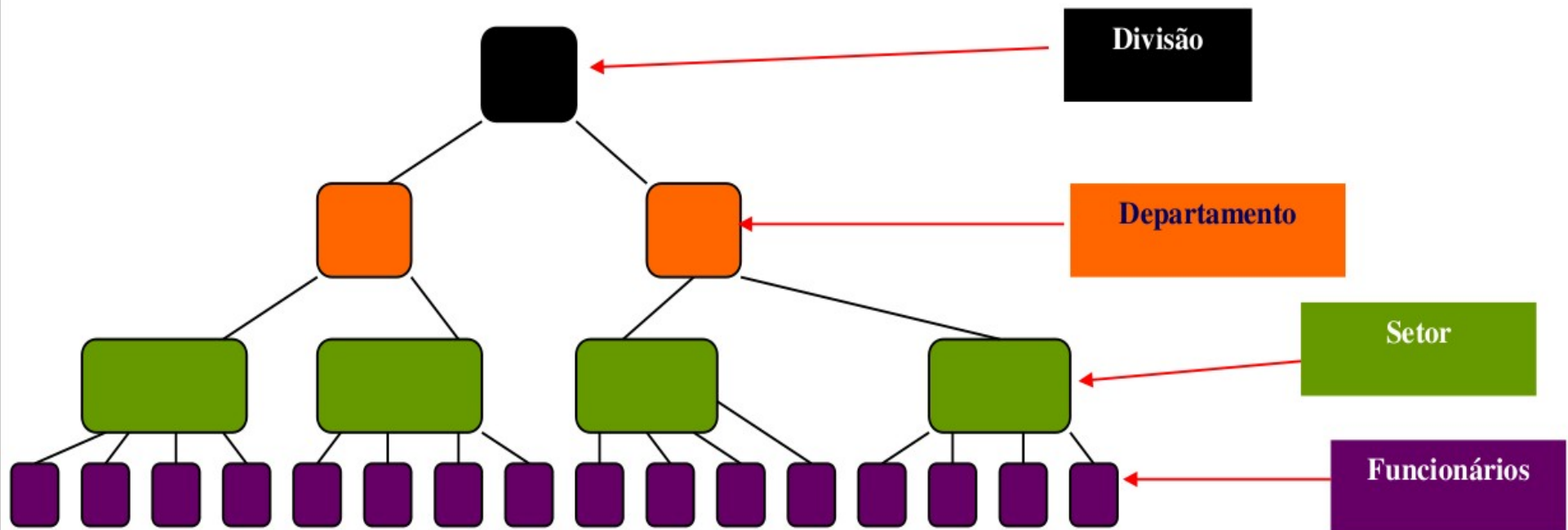
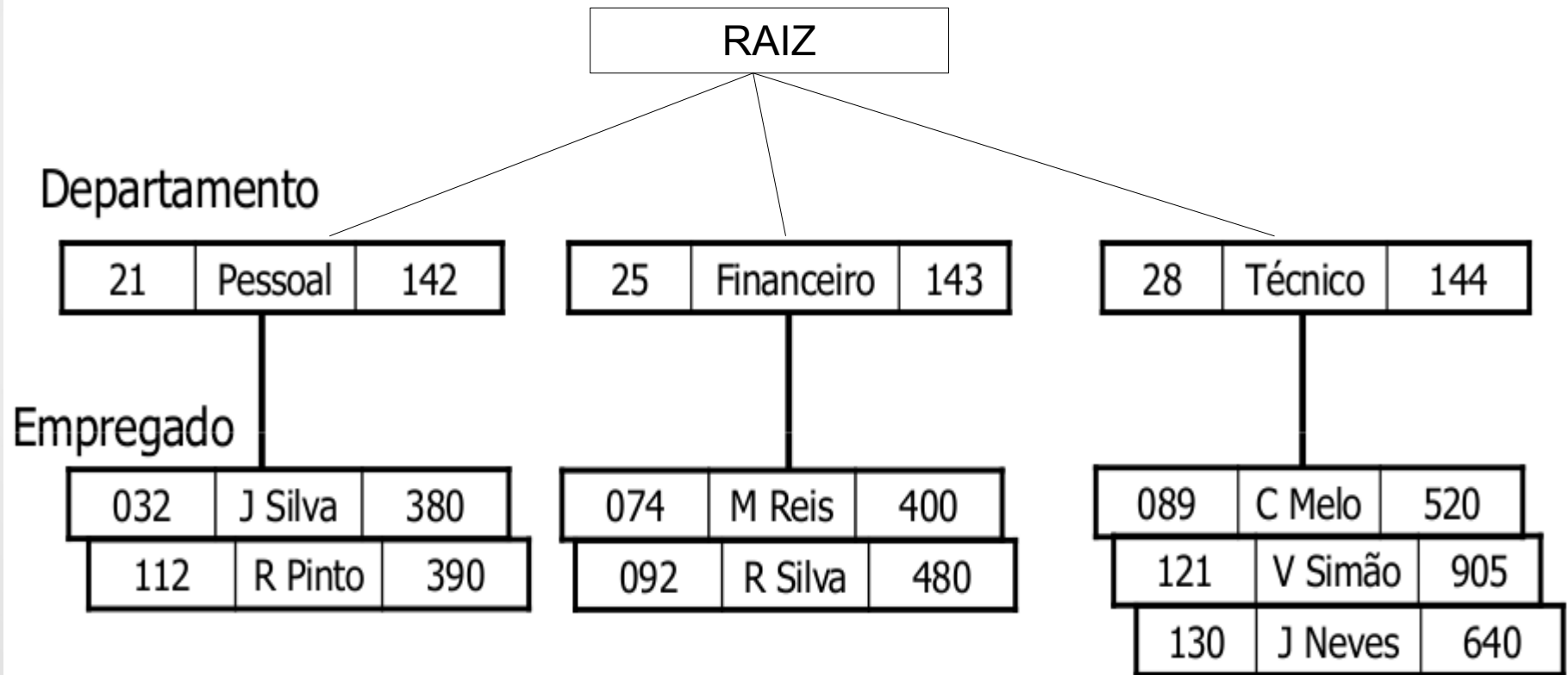


Figura 2 – Modelos Hierárquico, segundo [Fanderuff,2003]

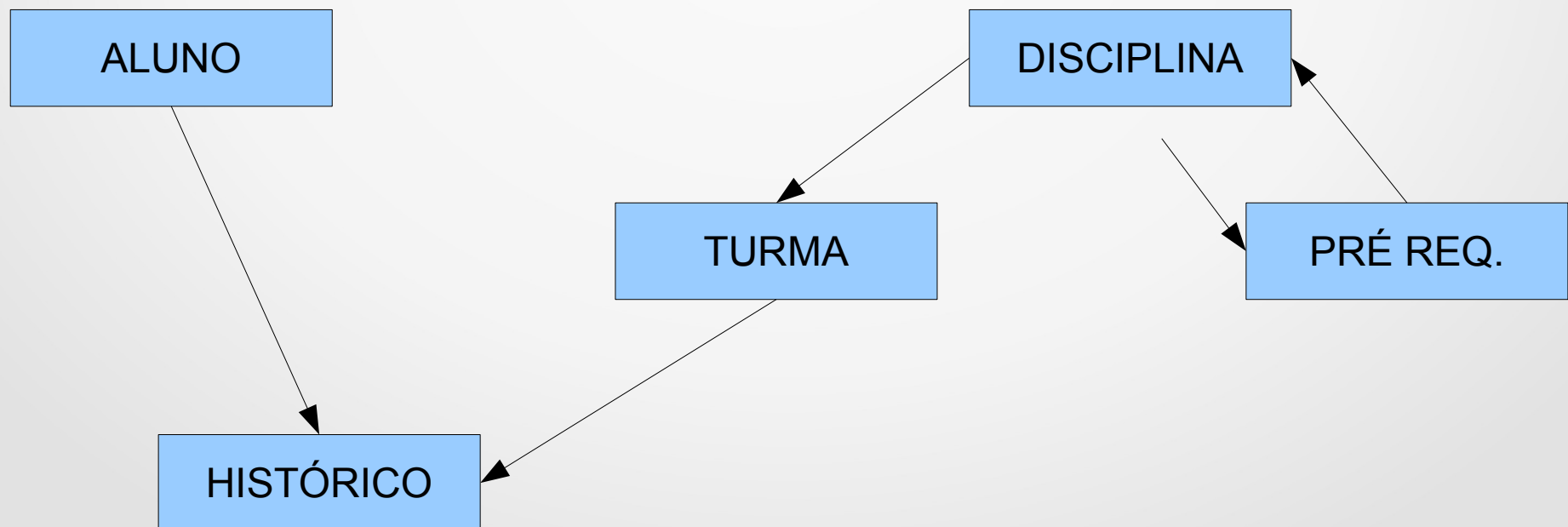
Modelos de Dados (1a Geração)

- Hierárquico



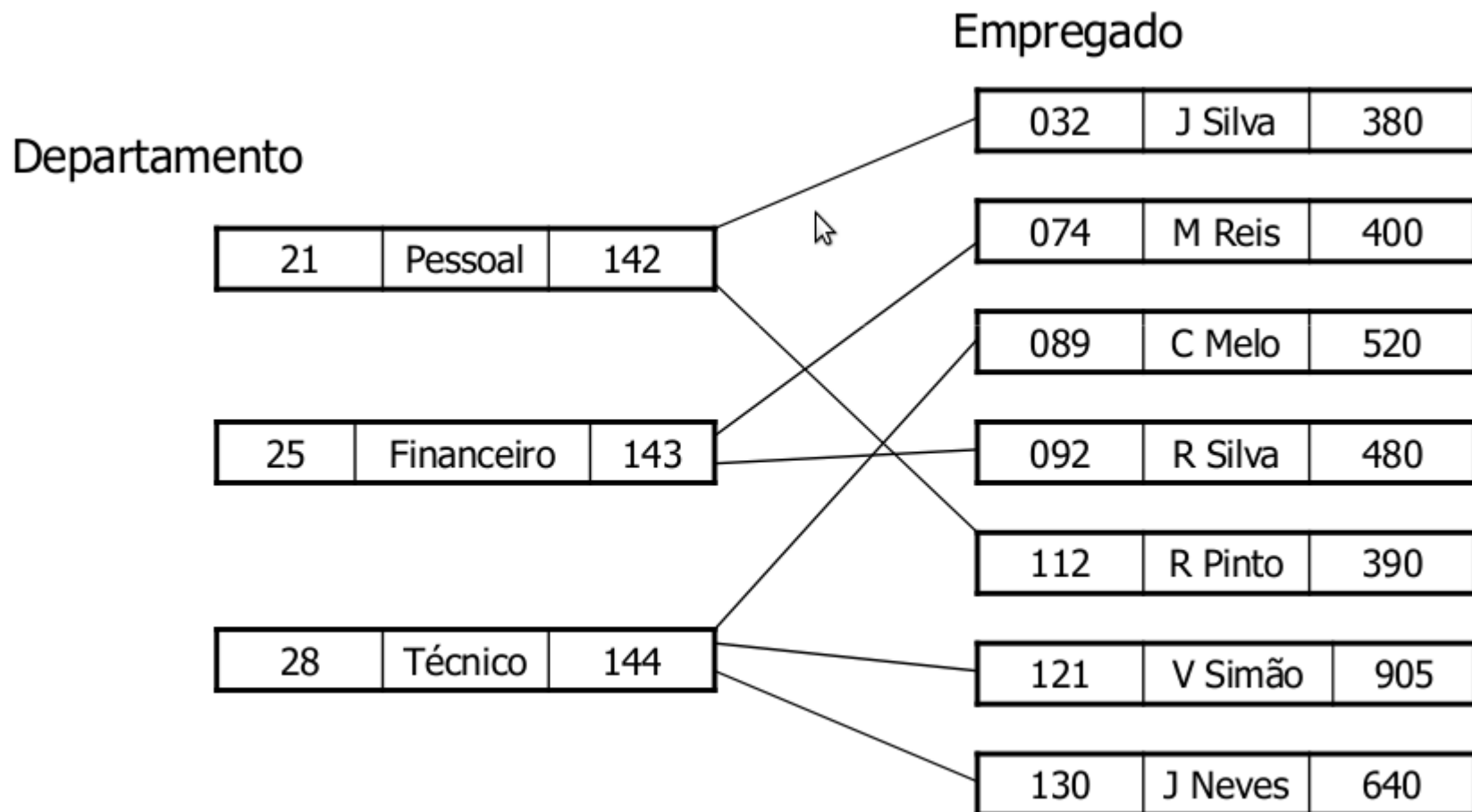
Modelos de Dados (1a Geração)

- Modelo de Rede (CODASYL)
 - Baseado em links e conexões
 - Representa dados como tipos de registros
 - Relaciona um registro com um ou vários outros através de ponteiamento



Modelos de Dados (1a Geração)

- Redes

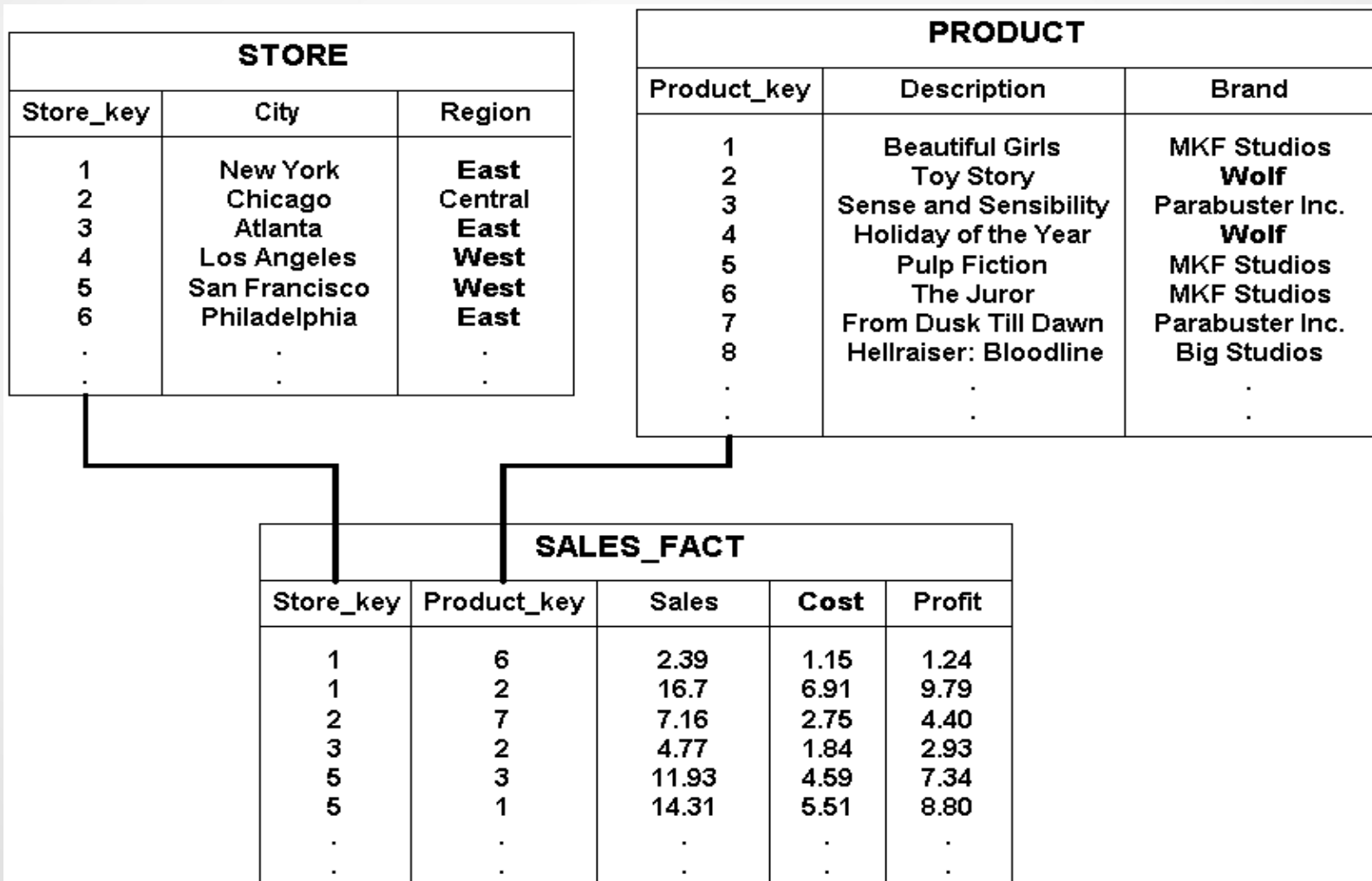


Modelos de Dados (2a Geração)

- Relacional
 - Proposto por Codd em 1970
 - Dados são vistos como relação (tabelas)
 - O relacionamento entre as relação é feito através de valores comuns entre as mesmas
 - Modelo baseado em um formalismo matemática
 - Definido sobre o mesmo a álgebra relacional
 - A base matemática permite
 - Otimizar consultas
 - Otimizar acesso
 - Otimizar armazenamento

Modelos de Dados (2a Geração)

- Relacional



Modelos de Dados (2a Geração)

- Relacional
 - Modelo de banco de dados mais utilizado por aplicações comerciais
 - Oracle
 - SQLServer
 - DB2
 - MySQL
 - PostgreSQL
 - Firebird

Modelos de Dados (3a Geração)

- Extensões do modelo relacional
- Orientado a objetos
- Objetos-relacionais
- Semiestruturados

Modelos de Dados (3a Geração)

- SGBD NoSQL (Not Only SQL)
 - Classe de SGBD para trabalhar com quantidade volumosa de dados distribuídos em diferentes nós de uma rede
 - Modelos de dados
 - Orientado a documentos (JSON, XML)
 - MongoDB, CouchBase, eXist
 - Orientado a colunas
 - MonetDB, C-store, Cassandra
 - Orientado a chave/valor
 - DynamoDB, SimpleDB, Redis, Riak
 - Orientado a grafos
 - Neo4j, GraphBase

Classificação

- Banco de dados convencionais
 - Características
 - Dados bem estruturados
 - Tipos de dados simples (inteiros, caracteres, data, reais, ...)
 - Transações simples e curtas
 - Acesso através de chaves

Classificação

- Banco de dados não convencionais
 - Características
 - Grande volume de dados (às vezes, não estruturados)
 - Tipos de dados complexos (gráficos, imagens, sons, ...)
 - Transações longas
 - Caminho de acessos não triviais
 - Controle de versões

Classificação

- Big Data
 - Conjunto de problemas e suas soluções tecnológicas em computação para tratar certos tipos de dados:
 - Volumosos, heterogêneos, fácil acesso
 - 3 V's: volumosos, variados e velozes
 - SGBDs NoSQL foram propostos para atender Big Data
 - Volume
 - Giga (G): bilhões
 - Tera (T): trilhões
 - Peta (P): mil trilhões
 - Exa (E): milhões de trilhões
 - 5E → todas as palavras ditas pelos humanos

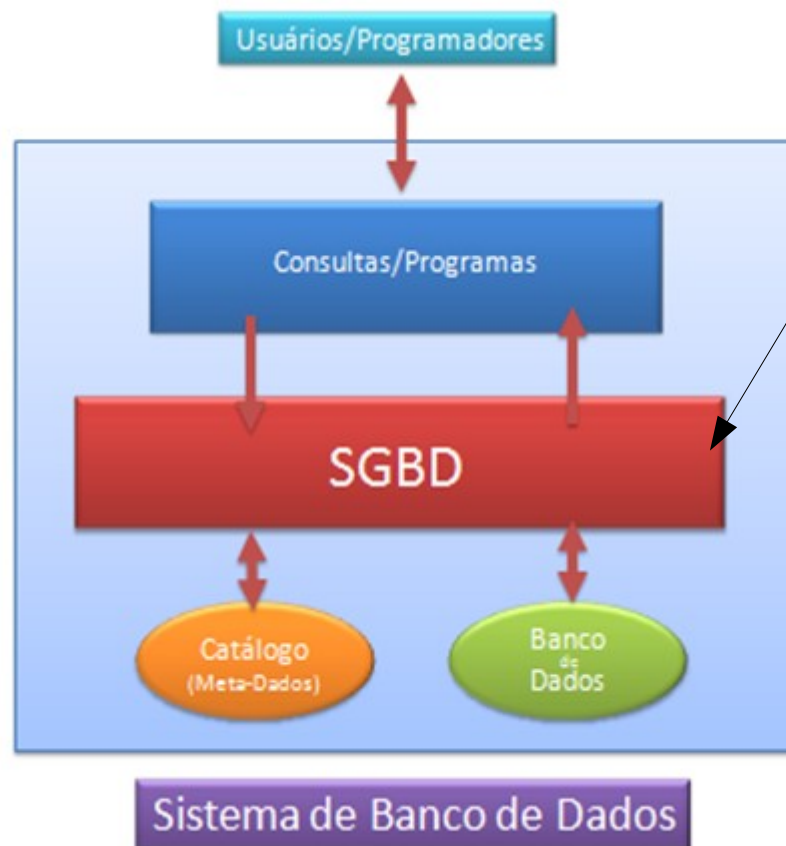
SGBD

- Sistema Gerenciando de Banco de Dados (SGBD)



SGBD

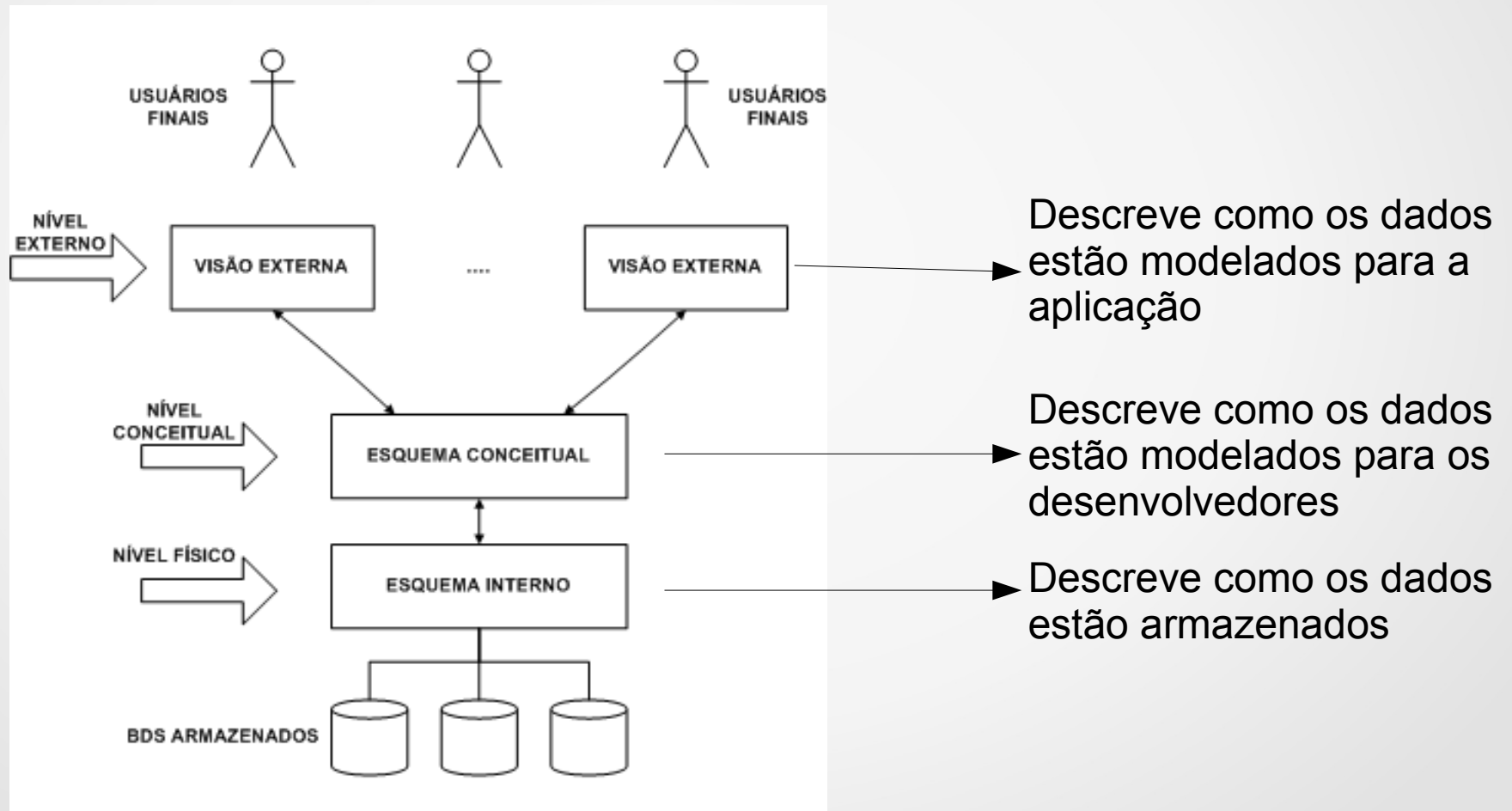
- Sistema Gerenciando de Banco de Dados (SGBD)



Conjunto de programas que permite ao usuário manter um banco de dados de forma consistente e segura

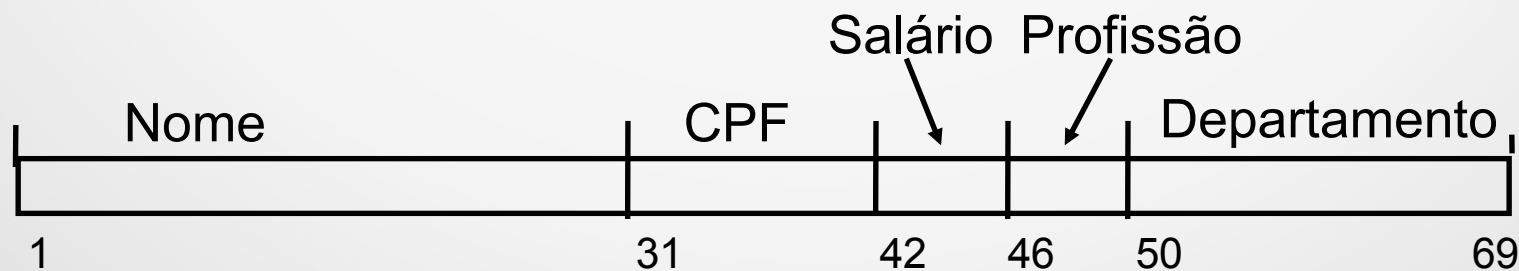
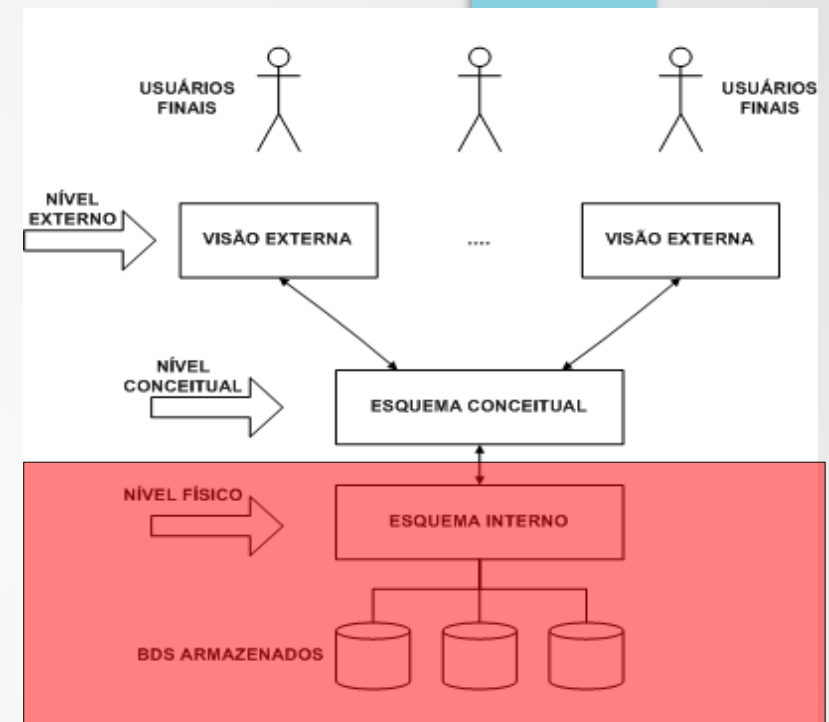
Arquitetura 3 Camadas

- Arquitetura três-camadas (ANSI/SPARC)



Arquitetura 3 Camadas

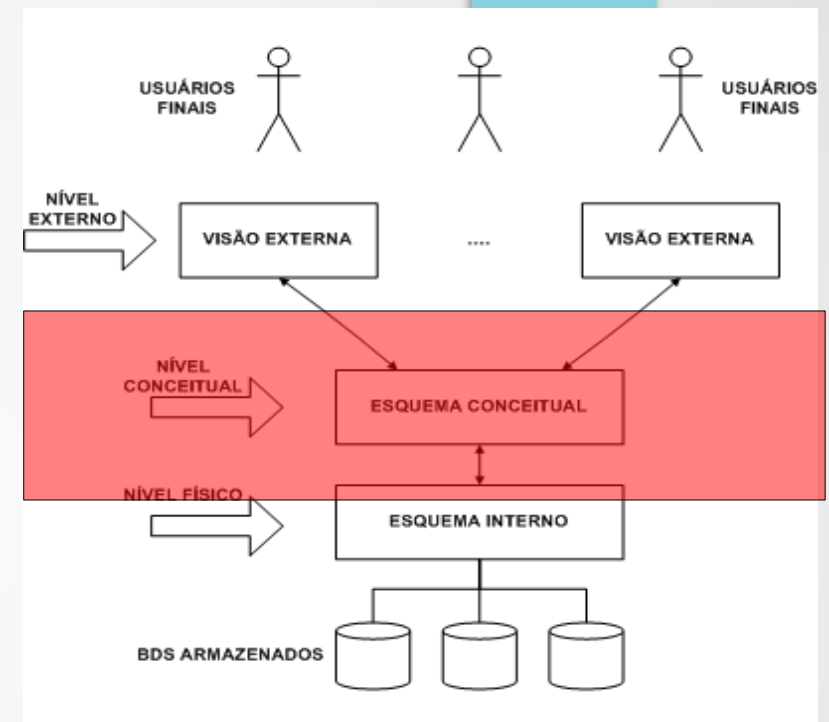
- Arquitetura três-camadas



Arquitetura 3 Camadas

- Arquitetura três-camadas

```
create table Func (  
  Nome varchar2(30),  
  CPF    number(11),  
  Sal    number(10,2),  
  CProf  number(4),  
  CDepto number(4))
```



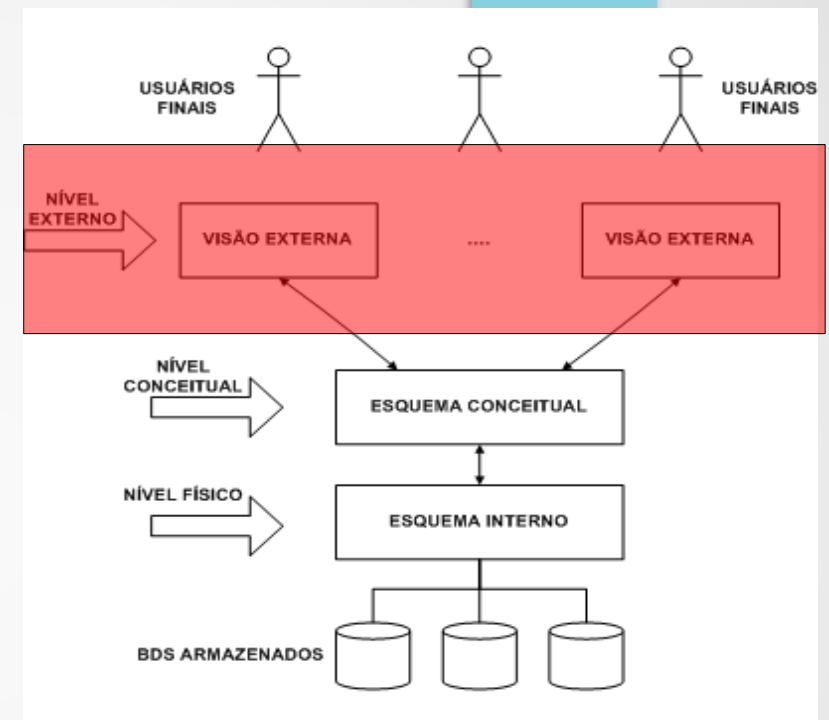
Arquitetura 3 Camadas

- Arquitetura três-camadas

Nome

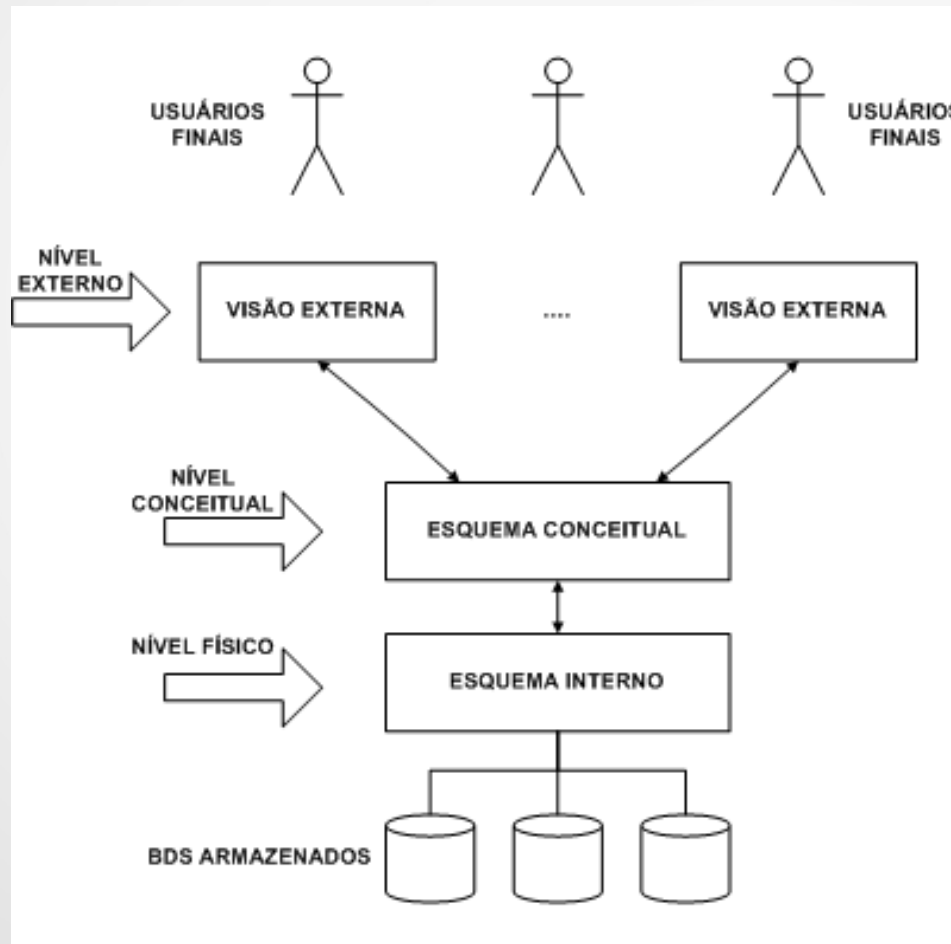
CPF

Salário



Arquitetura 3 Camadas

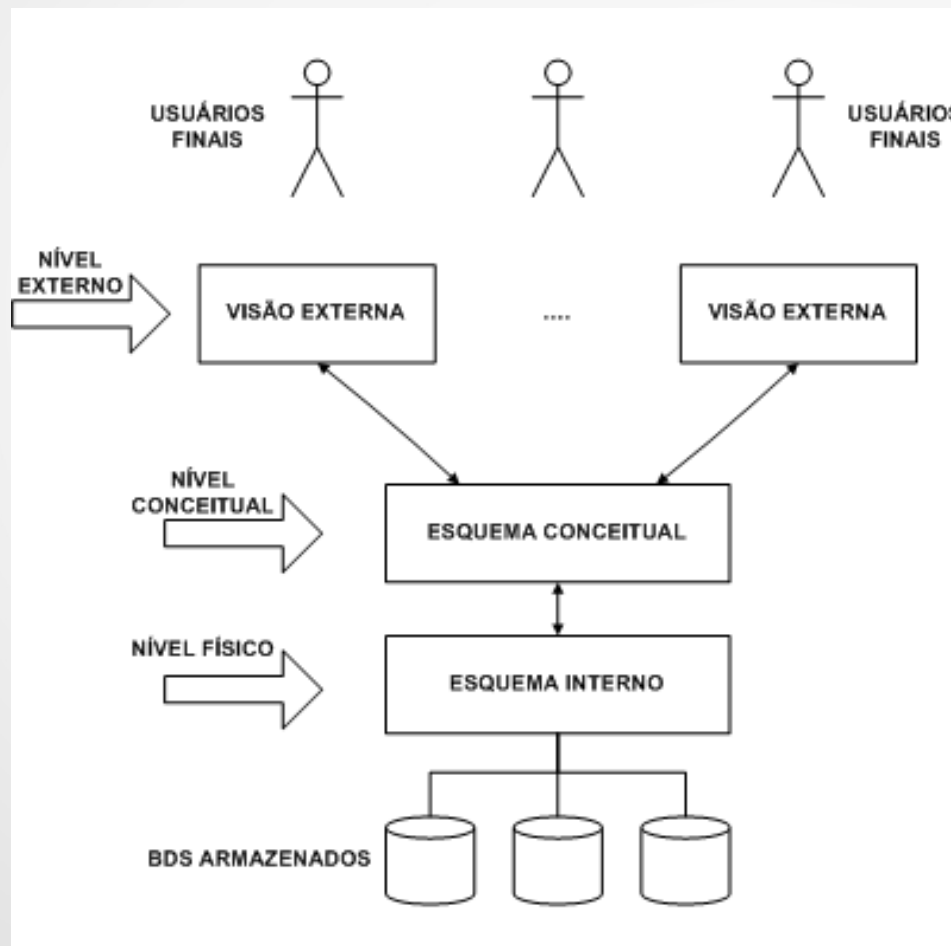
- Arquitetura três-camadas



Independência lógica dos dados:
O nível conceitual pode ser alterado sem afetar o nível externo.

Arquitetura 3 Camadas

- Arquitetura três-camadas

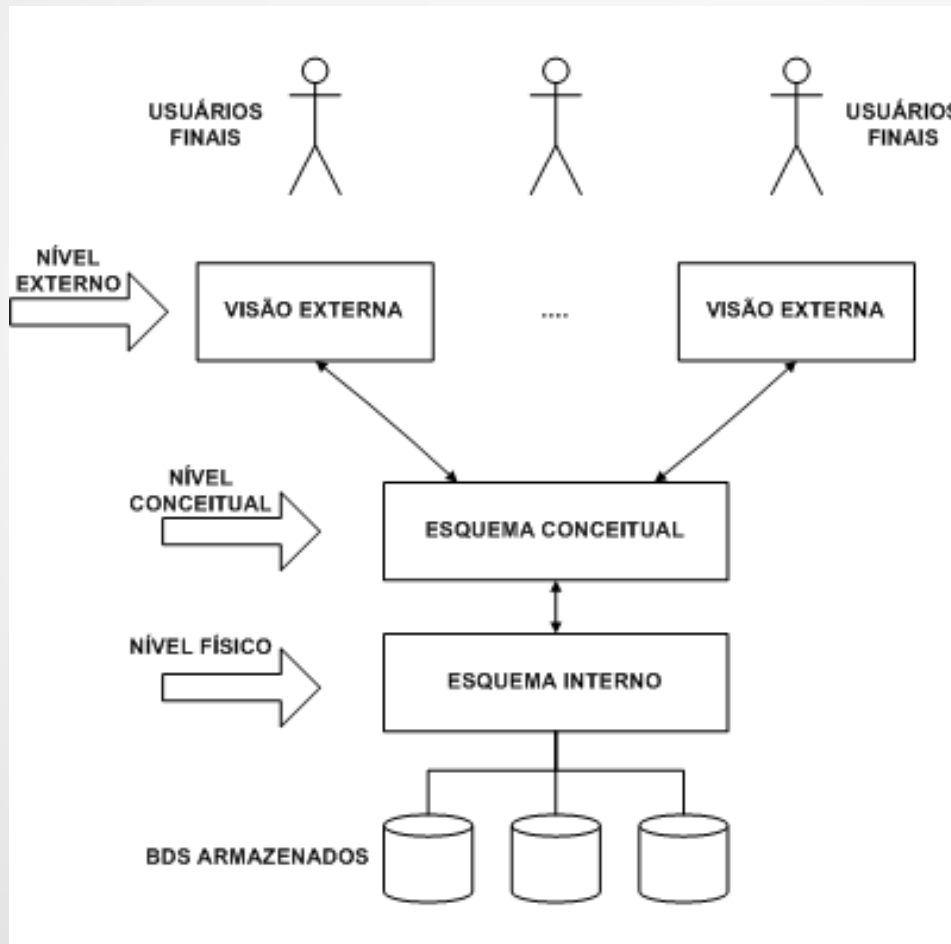


Independência lógica dos dados:
O nível conceitual pode ser alterado sem afetar o nível externo.

```
create table Func (  
    Nome varchar2(30),  
    CPF    number(11),  
    Ender  varchar2(40),  
    Sal    number(10,2),  
    CProf  number(4),  
    CDepto number(4))
```

Arquitetura 3 Camadas

- Arquitetura três-camadas



Nom	
€PF	
Salári	
o	

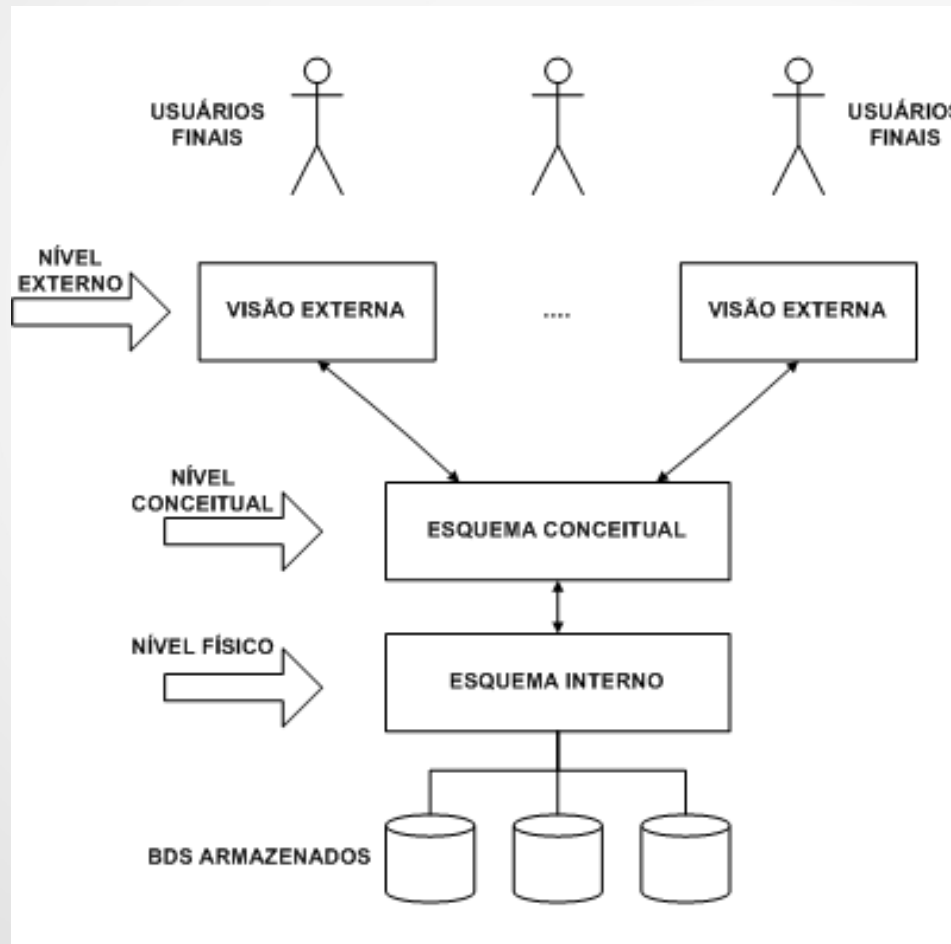
OK!

Independência lógica dos dados:
O nível conceitual pode ser alterado sem afetar o nível externo.

```
create table Func (  
  Nome varchar2(30),  
  CPF   number(11),  
  Ender varchar2(40),  
  Sal   number(10,2),  
  CProf number(4),  
  CDepto number(4))
```

Arquitetura 3 Camadas

- Arquitetura três-camadas



Independência física dos dados:
capacidade de alterar o esquema
interno sem a necessidade de
alteração do esquema conceitual.

Funções Básicas

- Integridade semântica
 - Dados corretos em relação ao domínio da aplicação
 - Tamanho de uma sequência de caracteres
 - Cardinalidade entre tabelas 1:N, 1:1, etc
 - Regras de integridade
 - Chaves primárias/estrangeiras

Funções Básicas

- Cópia, restauração e recuperação de dados
 - Backup, restore (restauração), recovery (recuperação)
- Desempenho: mecanismos de otimização
- Segurança
 - Não permitir inconsistências nos dados
 - Segurança de acesso
 - Permissões, visões
 - Segurança contra falhas
 - Gerenciamento de transações
 - Gerenciamento de recuperação

Funções Básicas

- Concorrência
 - Permitir acessos simultâneos aos dados com garantia da consistência
- Independência dos dados
- Capacidade dos dados de um BD persistirem ao longo de diferentes execuções de programas de aplicação (persistência)



Componentes SGBD

Componentes de um SGBD

