



PELO FUTURO DO TRABALHO

Modelos de Banco de Dados

Professor: Hermano Roepke

Modelos de banco de dados

- Modelo de Dados Hierárquico
- Modelo de Dados Rede
- Modelo de Dados Relacional
- Modelo de Dados Orientado a Objetos
- Modelo de Dados Pós-Relacional

Modelo de Dados Hierárquico

Modelo de Dados Hierárquico

O que é um modelo de dados?

Um modelo de dados é a forma com que os dados estão representados, ou seja, descrição da estrutura básica de armazenamento.

E por que diferentes modelos de dados?

Podemos afirmar que a existência dos diferentes modelos é atribuída a evolução natural das tecnologias e, sobretudo, das expectativas geradas pelos desenvolvedores e usuários dos sistemas de informação ao longo dos tempos.

Modelo de Dados Hierárquico

Um **registro** é o conjunto de atributos organizados que representam dados.

Um **atributo é a menor porção do registro**, isto é, constitui uma parte do conjunto de dados armazenado.

Imagine a seguinte estrutura que mantém dados de clientes:



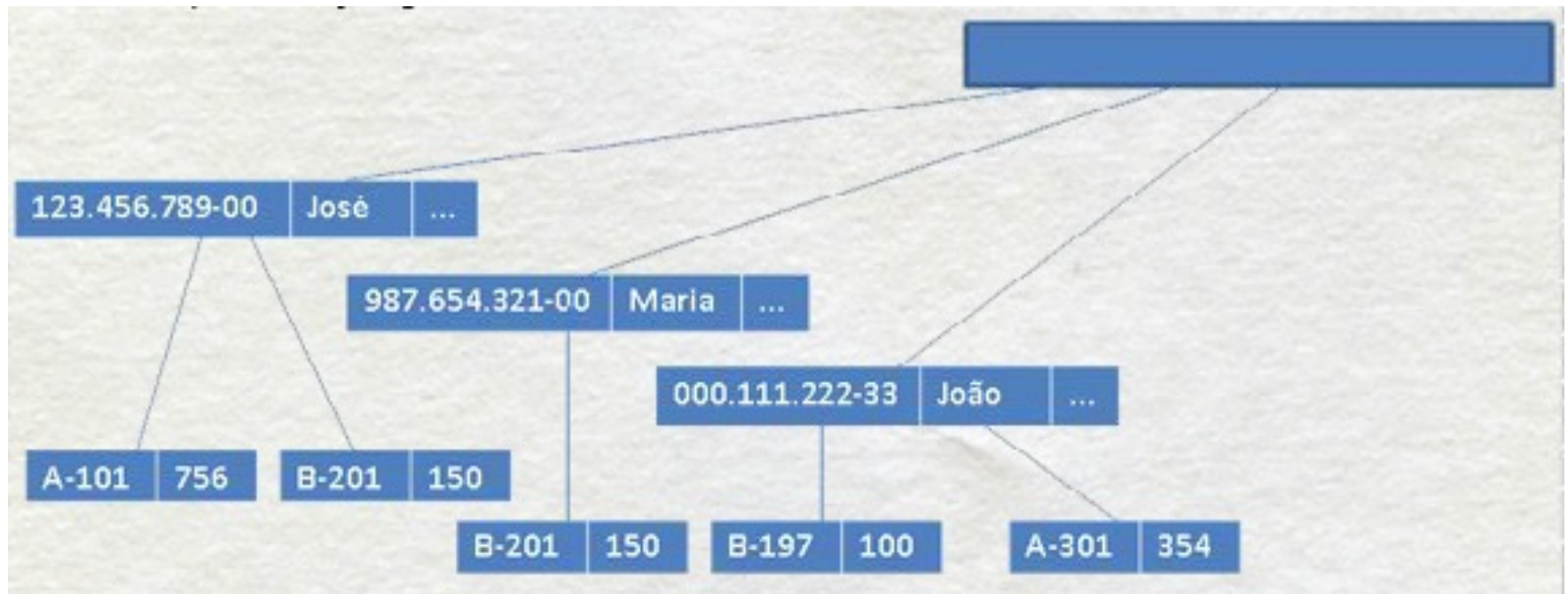
Modelo de Dados Hierárquico

Características:

- Considerando o primeiro modelo dados reconhecido;
- Dados representados na forma de registro;
- As relações entre os registros são estabelecidos através de ligações;
- Organização dos registros em estruturas de árvores;

Modelo de Dados Hierárquico

Representação gráfica:

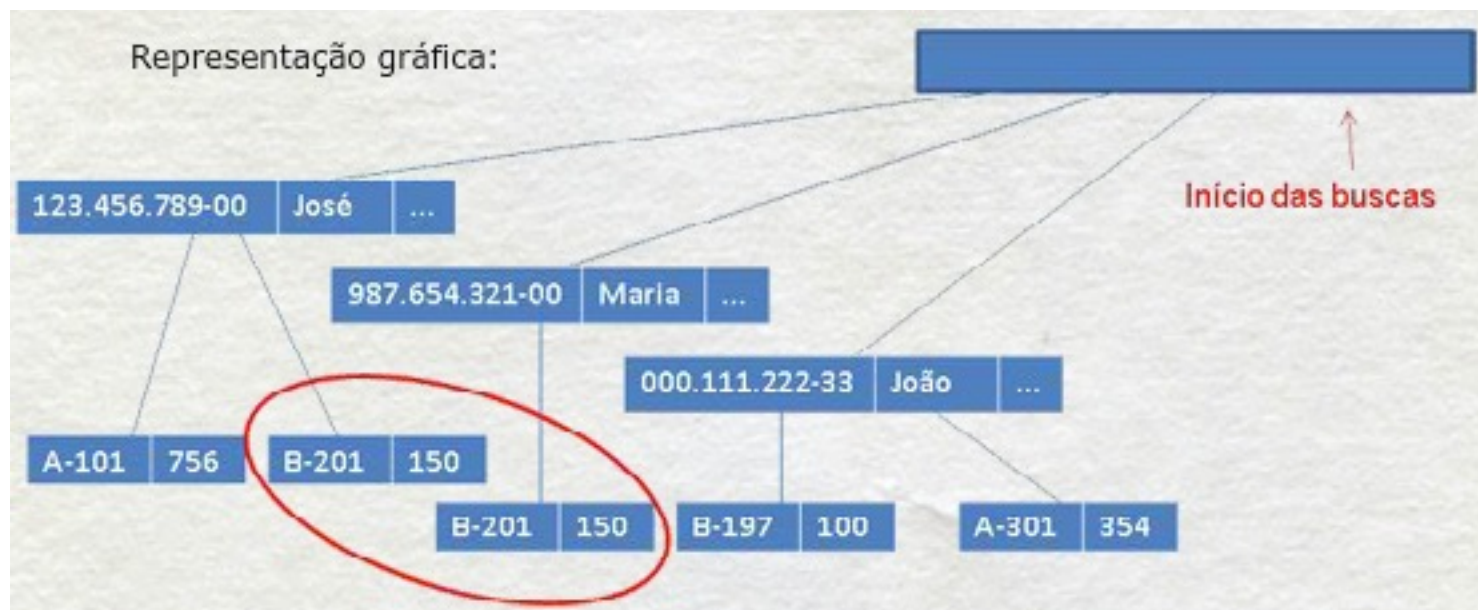


Modelo de Dados Hierárquico

Considerações:

- Replicação gera redundância e inconsistência nos dados;
- Desperdício de espaço inevitável;
- Buscas sempre partindo do registro principal;
- Forte dependência entre as estruturas e aplicação.

Modelo de Dados Hierárquico



Modelo de Dados Hierárquico

A seguir o código na linguagem DLI/IMS que define as estruturas representadas:

schema name = BANCO

Hierarchies = HIERARQUIA1

record name = CLIENTE

type = RAIZ DE HIERARQUIA1

data items = CPF integer

NOME character(30)

EMAIL character(50)

key = CPF

record name = CONTA

parent = CLIENTE

child NUMBER = 1

data items = NRO integer

SALDO float(8,2)

key = NRO

Modelo de Dados Hierárquico

Este modelo de banco de dados não contemplou um número significativo de produtos acadêmicos e nem tão pouco comerciais.

Alguns dos produtos conhecidos são:

- **Information Management System** – Empresa IBM;
- **ADABAS** – Empresa Software AG

Curiosidade: O ADABAS surgiu na década de 70 e é considerado o primeiro banco de dados comercial. Ao longo dos tempos sofreu atualizações e por isto está no mercado até hoje.

Modelo de Dados Rede

Modelo de Dados Rede

Ao estudar e conhecer o modelo de Banco de Dados Hierárquico você identificou algumas fragilidades principalmente relacionadas a estrutura e manipulação dos dados.

É nesta perspectiva que surgiu o modelo de Banco de Dados Rede o qual você terá a oportunidade de conhecer um pouco mais.

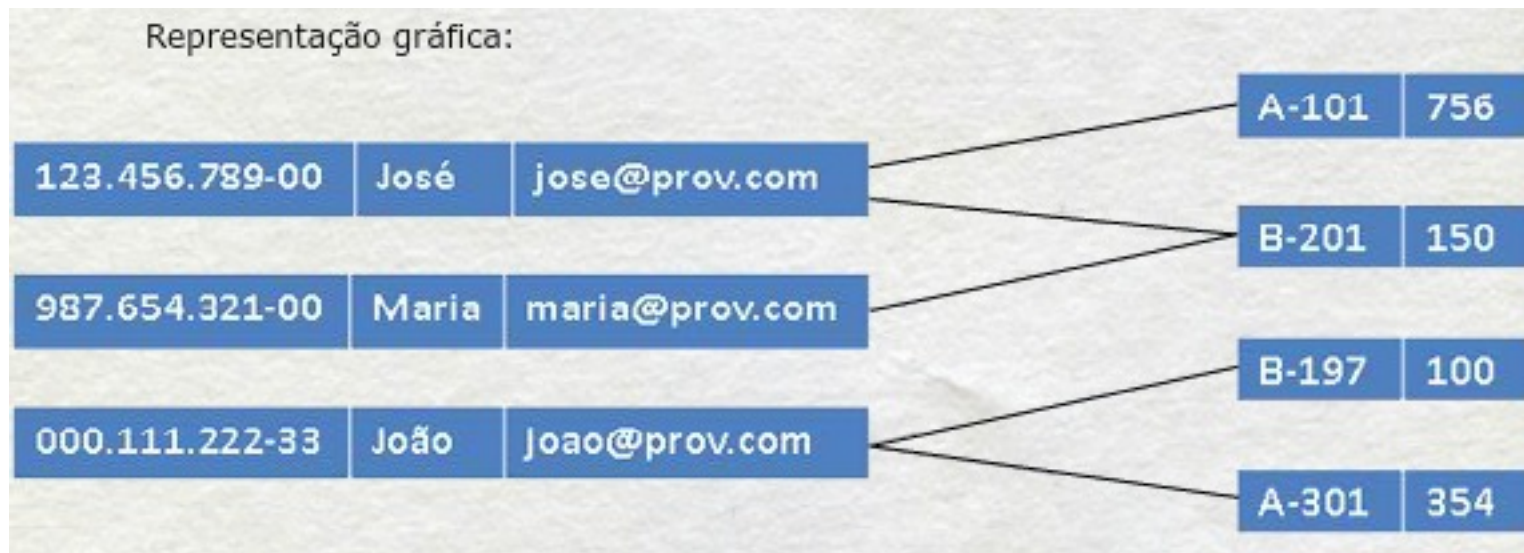
Modelo de Dados Rede

Características:

- Considerando uma extensão do modelo hierárquico;
- Dados representados na forma de registro;
- As relações entre os registros são estabelecidos através de ligações;
- Organização dos registros em estruturas de grafos;

Modelo de Dados Rede

Representação gráfica:



Modelo de Dados Rede

Considerações:

- Homologação e acompanhamento de um grupo gestor (CODASYL);
- Linguagem padrão para definição e manipulação dos dados;
- Limitada independência física dos dados;
- Implementação de mecanismos de segurança e consistência;
- Buscas podem ser iniciadas a partir de registros de referência.

Modelo de Dados Rede

A seguir o código na linguagem baseada no COBOL que define as estruturas representadas:

schema name is BANCO

record name is CLIENTE

 duplicates are not allowed for CPF

 CPF type is numeric integer

 NOME type is character(30)

 EMAIL type is character(50)

record name is CONTA

 duplicates are not allowed for NRO

 NRO type is numeric integer

 SALDO type is numeric float

Modelo de Dados Rede

Com a homologação por parte do grupo gestor (CODASY), o modelo obteve significativa representação em produtos comerciais.

Alguns dos produtos conhecidos são:

- CA-IDMS – Empresa Computer Associates;
- VAX DBMS – Empresa Digital;
- IDS – Empresa Honeywell Inc.;
- DMS110 – Empresa UNIVAC.

Curiosidade: O CA-IDMS é um dos bancos de dados comerciais mais conhecidos deste modelo e muito utilizado atualmente.

Modelo de Dados Relacional

Modelo de Dados Relacional

Até o momento você conheceu o modelo de banco de dados hierárquico e o modelo de banco em rede, cada qual com suas particularidades e funcionalidades.

Mesmo assim, ambos apresentam características comuns como a representação dos dados na forma de registros. Isto os torna modelos orientados a registros, pois qualquer acesso a base de dados para manipulação (consulta, inclusão, alteração ou exclusão) é realizado em um registro de cada vez.

Modelo de Dados Relacional

Você irá conhecer as características do modelo mais utilizado atualmente, o modelo relacional.

Por que Relacional?



Modelo de Dados Relacional

No Modelo Hierárquico, através da representação dos dados na forma de árvores, você não teve dificuldades em compreender a origem do nome.

Já no Modelo de Rede ao observar que os dados são representados na forma de grafos, logo você percebeu que tratava-se de uma rede com inúmeras ligações entre os registros o que originou a denominação do modelo.

Modelo de Dados Relacional

A origem da denominação Modelo Relacional está associada a forma como os dados, também representados na forma de registros, estão ligados uns aos outros. Neste modelo não há ponteiros de ligação (links) pois os dados “se relacionam” uns aos outros através de valores próprios. Com isto surge a nomenclatura Modelo Relacional.

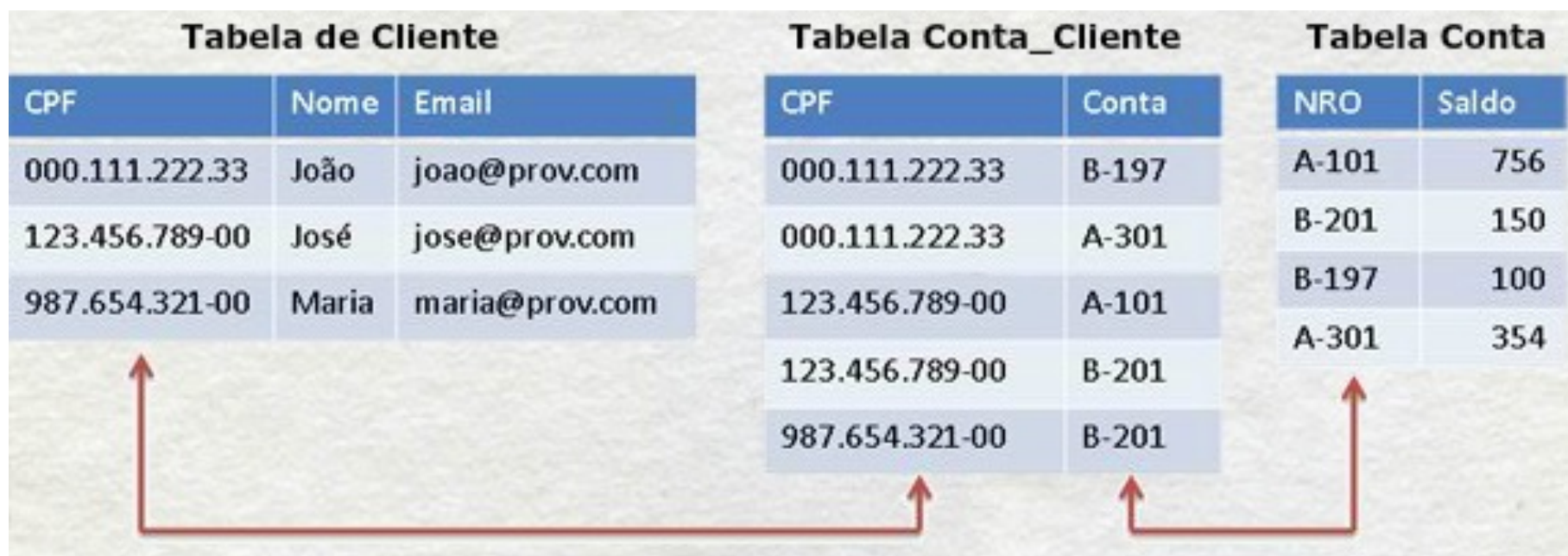
Modelo de Dados Relacional

Características:

- Independência dos dados;
- Conjunto de funções apoiadas em álgebra relacional para manipulação;
- Dados representados na forma de registro;
- Organização dos registros na forma de tabelas;
- Relação entre registros estabelecidas através de atributos (chaves);

Modelo de Dados Relacional

Representação gráfica:



Modelo de Dados Relacional

Considerações:

- Flexível para concepção e implementação de base de dados;
- Simplicidade na definição das estruturas;
- Possibilidades de alta performance na manipulação dos dados;
- Buscas a partir de qualquer tabela/registo/atributo da base de dados;
- Linguagem padrão – SQL;
- Conjunto de restrições;
- Marco referencial para o desenvolvimento do Sistema Gerenciador de Banco de Dados (SGBD).

Modelo de Dados Relacional

A seguir o código na linguagem SQL que define as estruturas representadas:

```
create database BANCO;
```

```
create table CLIENTE (  
    CPF    integer not null,  
    NOME   varchar(30),  
    EMAIL  varchar(50),  
    primary key (CPF));
```

```
Create table CONTA (  
    NRO    integer not null,  
    SALDO  float,  
    primary key(NRO));
```


```
Create table CLIENTE_CONTA (  
    CPF    integer not null,  
    NRO    integer not null,  
    primary key (CPF, NRO),  
    foreign key (CPF) references CLIENTE(CPF),  
    foreign key (NRO) references CONTA(NRO));
```

Modelo de Dados Relacional

```
create database BANCO;  
create table CLIENTE (  
    CPF    integer not null,  
    NOME   varchar(30),  
    EMAIL  varchar(50),  
    primary key (CPF));
```

```
Create table CONTA (  
    NRO    integer not null,  
    SALDO  float,  
    primary key(NRO));
```

```
Create table CLIENTE_CONTA (  
    CPF    integer not null,  
    NRO    integer not null,  
    primary key (CPF, NRO),  
    foreign key (CPF) references CLIENTE(CPF),  
    foreign key (NRO) references CONTA(NRO));
```



Para que serve estas chaves?

Modelo de Dados Relacional

O papel das chaves

Você pode observar que há uma definição de chaves primárias (primary key) e chaves estrangeiras (foreign key). Entenda o que:

- **Chave Primária:** é a combinação de um ou mais atributos que identificam unicamente a entidade. O valor deve ser único para cada registro (linha).
- **Chave estrangeira:** é a combinação de um ou mais atributos que referenciam outra tabela. Possui valor da chave primária da tabela referenciada.

Com estes conceitos é possível afirmar que a chave primária e a chave estrangeira são os elementos que estabelecem o relacionamento entre as tabelas.

Modelo de Dados Relacional

O modelo relacional apresenta inúmeros produtos comerciais e de código aberto. Alguns produtos que na década passada eram estritamente licenciados e com valor agregado significativo, atualmente disponibilizam versões livres com nenhuma ou poucas restrições de funcionalidades, porém com limitação de espaço de gerenciamento de dados.

Alguns dos produtos mais conhecidos são:

- MYSQL – Empresa Sun Microsystems;
- ORACLE – Empresa Oracle;
- MSSQL – Empresa Microsoft;
- POSTGRESQL – Projeto Open source do PostgreSQL Global Development Group;

Modelo de dados Orientado a Objetos

Modelo de dados Orientado a Objetos

Hierárquico, Rede e Relacional. O que estes modelos tem em comum? Todos são orientados a registros. É claro que esta é apenas uma das características comum entre os modelos citados.

E qual a importância desta informação neste momento? Simples, porém com mudança significativa no paradigma!

É que o modelo que você irá conhecer agora é orientado a objetos, ou seja, sua estrutura apresenta um conjunto de variáveis (similar aos atributos dos modelos que o precederam), porém com mecanismos totalmente diferenciados e específicos.

Modelo de dados Orientado a Objetos

Por que Orientado a Objetos?



Modelo de dados Orientado a Objetos

Orientação a objetos é um paradigma de análise, projeto e programação de sistemas de software baseado na composição e interação entre diversas unidades de software denominados objetos.

Porém, estas unidades de software (objetos) em geral são transientes. Objetos transientes são objetos que existem apenas enquanto a aplicação que os criou continuar executando. Após o término da aplicação eles deixam de existir.

Modelo de dados Orientado a Objetos

Para resolver este problema é necessário tornar estes objetos persistentes, ou seja, os dados referentes a estes objetos continuam existindo após o término do programa que o criou. Por isto a criação do Modelo de Banco de Dados Orientado a Objetos.

Modelo de dados Orientado a Objetos

Características:

- Ambiente acadêmico tornando-se comercial na década de 80;
- Motivação para o surgimento em função dos limites de armazenamento de dados com tipos complexos;
- Dados representados na forma de objetos;
- Padrão definido pelo Object Database Management Group e incorporador as linguagens de programação orientada a objetos.

Modelo de dados Orientado a Objetos

Representação gráfica:



Modelo de dados Orientado a Objetos

Considerações:

- Boas expectativa em relação ao modelo relacional;
- Utilizando em aplicações especializadas;
- O diagrama de classes da UML serve como esquema para o modelo;
- Objetos as mesmas variáveis e métodos são agrupados em classes;
- Mesmos conceitos da orientação de objetos para comunicação e relação entre os objetos;
- Abstração de dados;
- Object Query Language (OQL) é uma tentativa de padronização entre todas as linguagens de programação orientadas a objetos.

Modelo de dados Orientado a Objetos

O código a seguir define as estruturas representadas para a linguagem C++ com as adequações ODMG:

```
Class Cliente: Public Objeto_Persistente {  
Private:  
    String nome;  
    String email;  
Public:  
    Int cpf;  
    Set <Ref <Conta>> contas inverse Conta :: proprietarios;  
};
```


Modelo de dados Orientado a Objetos

Class Conta: Public Objeto_Persistente {

Private:

double saldo;

Public:

Int numero;

Set <Ref <Cliente>> proprietarios inverse Cliente ::
contas;

Int entrar_saldo();

Int atualizar_saldo(int valor);

};

Modelo de dados Orientado a Objetos

O Modelo de banco de dados orientado a objetos tiveram uma nova chance de crescimento em meados de 2004 quando foram lançados alguns produtos de código aberto escritos em linguagens orientadas a objetos como JAVA e C++. Com isto, acreditava-se que poderiam se consolidar no mercado. Mas isto não ocorreu. Alguns dos produtos mais conhecidos são:

- ORION – Empresa MCC;
- GemStone – Empresa GemStone Systems;
- ObjectStore – Empresa Progress Software;
- POET – Empresa Versant Corp.

Modelo de dados Objeto Relacional

Modelo de dados Objeto Relacional

O paradigma da Orientação a Objetos existe a no mínimo três décadas e vem se consolidando cada vez mais em termos análise, projeto e programação de sistemas.

Por sua vez, o modelo de banco de dados relacional também está consolidado quanto a estrutura de armazenamento e gerenciamento de dados.

E como juntar o que há de melhor da Orientação a Objetos com o potencial do Modelo Relacional?

Eis que se apresenta o Modelo de Banco de Dados Objeto-Relacional.

Modelo de dados Objeto Relacional

Certamente você já deve deduzir o porque da origem do modelo. Exatamente! O modelo Objeto-Relacional ou também conhecido como Pós-Relacional incorpora ao Modelo Relacional já consolidando os recursos cada vez mais emergentes da especificação e programação orientada a objetos.

Mas não se iluda! Há boas perspectivas para o modelo que estamos apresentando, até porque muitos dos bancos de dados relacionais largamente utilizados hoje estão adicionando características da orientação a objetos.

Modelo de dados Objeto Relacional

Características:

- Motivação para o surgimento é o de suprir dificuldades dos sistemas relacionais convencionais;
- Extensão da SQL com tipos e funções específicas da aplicação ao modelo relacional;
- Adoção do padrão SQL-3.

Modelo de dados Objeto Relacional

Representação gráfica:

Tabela Cliente						
CPF	Nome	Email	Contas			
123.456.789-00	José	jose@prov.com	A-101	756	B-201	150
987.654.321 00	Maria	maria@prov.com	B-201	150		
000.111.222-33	João	joao@prov.com	B-197	100	A-301	354

Diagram illustrating the graphical representation of the data model, showing the relationship between client information and account information.

The table is divided into two main sections:

- Informações do Cliente** (Client Information): Includes CPF, Nome, and Email.
- Informações das Contas** (Account Information): Includes Conta 1 and Conta 2.

Red arrows indicate the mapping from the table columns to these sections:

- CPF, Nome, and Email are mapped to **Informações do Cliente**.
- Conta 1 and Conta 2 are mapped to **Informações das Contas**.

Modelo de dados Objeto Relacional

Considerações:

- Apenas parte da SQL-3 está consolidada entre os produtos comerciais considerados como Objeto-relacionais, como suporte a tipos complexos, herança de tipos de dados estruturados;
- Pouca independência no uso dos recursos disponibilizados pelos fabricantes;
- Definição de objetos com tipos primitivos ou definidos pela necessidade da aplicação;
- Permite a navegação pela estrutura de objetos e a chamada de métodos diretamente em comandos de recuperação.

Modelo de dados Objeto Relacional

A seguir o código, baseado na linguagem SQL-3, que define as estruturas representadas:

```
create type ObjCONTA (  
    NRO    integer not null,  
    SALDO  float);
```

```
create type ObjCLIENTE (  
    CPF    integer not null,  
    NOME   varchar(30),  
    EMAIL  varchar(50),  
    CONTAS setof (ObjCONTA));
```

```
create table CLIENTE of type ObjCliente;
```

Modelo de dados Objeto Relacional

O modelo objeto-relacional apresenta vários produtos comerciais inclusive de código aberto.

Alguns dos produtos mais conhecidos são:

- ORACLE (a partir da versão 9i) – Empresa Oracle;
- DB2 (a partir da versão 7.1) – Empresa IBM Corp.
- POSTGRESQL (a partir da versão 6.0) – Projeto Open source do PostgreSQL Global Development Group.



PELO FUTURO DO TRABALHO

sesisc.org.br     **sc.senai.br**

0800 048 1212

Rodovia Admar Gonzaga, 2765 - Itacorubi - 88034-001 - Florianópolis, SC