

Centro
Universitário
FMU

FMU

Professor Lima

Banco de Dados



e-Learning



Research



www.professorlima.com



Banco de Dados

Banco de Dados

1 – Introdução a Banco de dados

- Conceitos básicos
- Arquitetura de um SGBD genérico.

01



2 – Modelo Entidade e Relacionamento (MER)

- Conceitos
- Diagramas.

02



3 – Modelo Entidade e Relacionamento (MER)

- Conceitos

Sistema gerenciador de banco de dados (SGBD)

- Um banco de dados computadorizado pode ser criado e mantido por um grupo de programas de aplicação escritos especificamente para essa tarefa, também conhecido como sistema gerenciador de banco de dados (SGBD database management system)
- Um sistema gerenciador de banco de dados é composto por uma coleção de programas que permitem aos usuários criar e manter um banco de dados. O SGBD é um sistema de software de uso geral que facilita o processo de definição, construção, manipulação e compartilhamento de bancos de dados entre diversos usuários e aplicações.
- Preste atenção nas quatro palavras com destaque na definição anterior (definição, construção, manipulação e compartilhamento). É claro que você as conhece. Mas, em informática, é necessário precisar os termos, pois tem significados específicos.

Sistema gerenciador de banco de dados (SGBD)

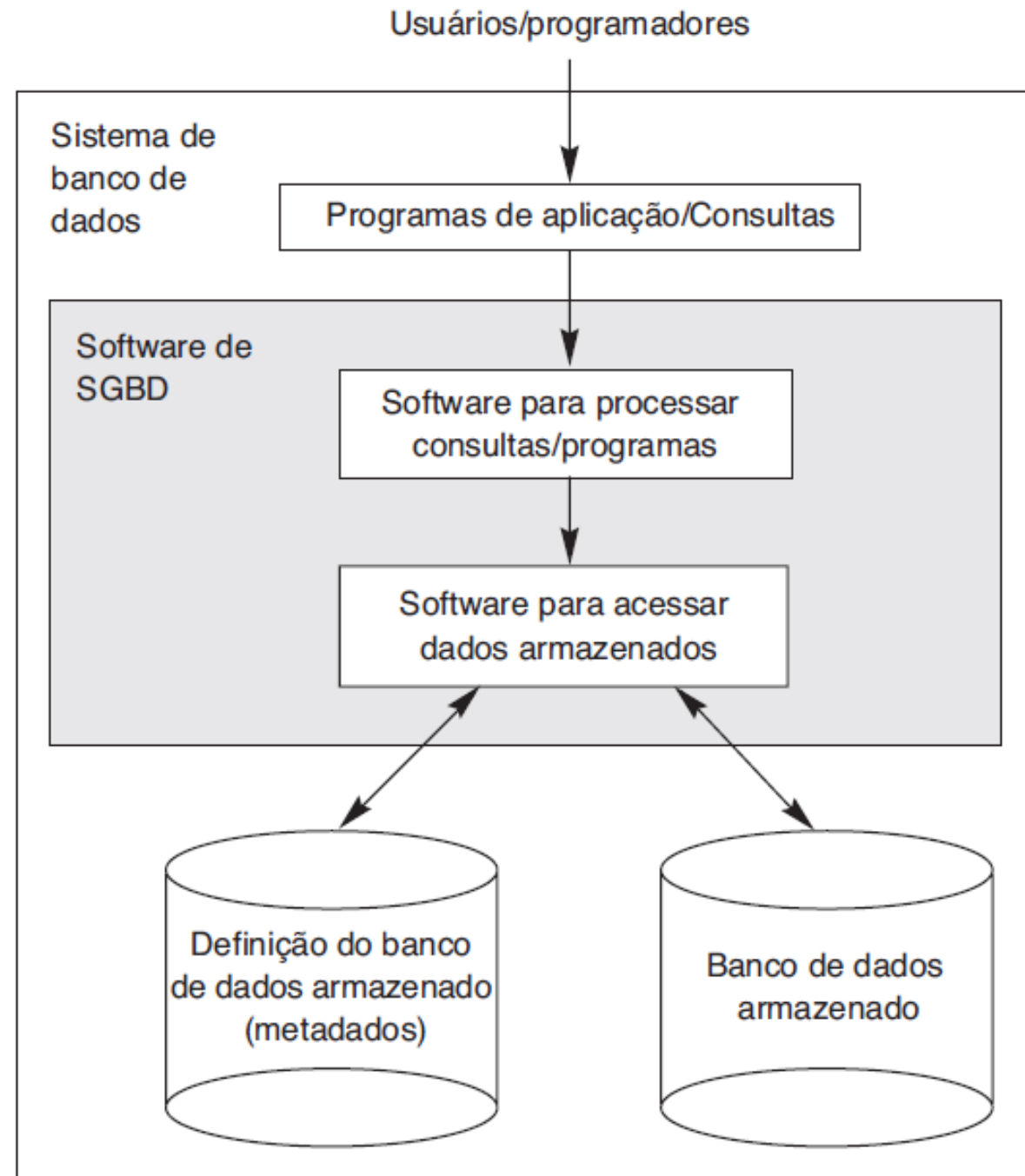
- **Definição de um banco de dados** - a definição de um banco de dados envolve especificar os tipos, as estruturas e as restrições dos dados a serem armazenados. A definição ou informação descritiva do banco de dados também é armazenada pelo SGBD na forma de catálogo ou dicionário, chamado de metadados.
- **Construção do banco de dados** - é o processo de armazenar os dados em algum meio controlado pelo SGBD.
- **Manipulação de um banco de dados** - esse processo inclui funções como consulta ao banco de dados para recuperar dados específicos, atualização para refletir mudanças no mundo e geração de relatórios com base nos dados.
- **Compartilhamento de um banco de dados** - é processo que permite que o banco de dados seja acessado simultaneamente por diversos usuários e programas.

Sistema gerenciador de banco de dados (SGBD)

Outras funções importantes fornecidas pelo SGBD são:

- manutenção do banco de dados por um longo período, permitindo que ele evolua em sintonia com as mudanças dos requisitos;
- proteção do sistema contra defeitos (ou falhas) de hardware ou software;
- proteção de segurança contra acessos não autorizados ou maliciosos.

Diagrama
simplificado de
um ambiente
de Sistema de
banco de dados



Sistema gerenciador de banco de dados (SGBD)

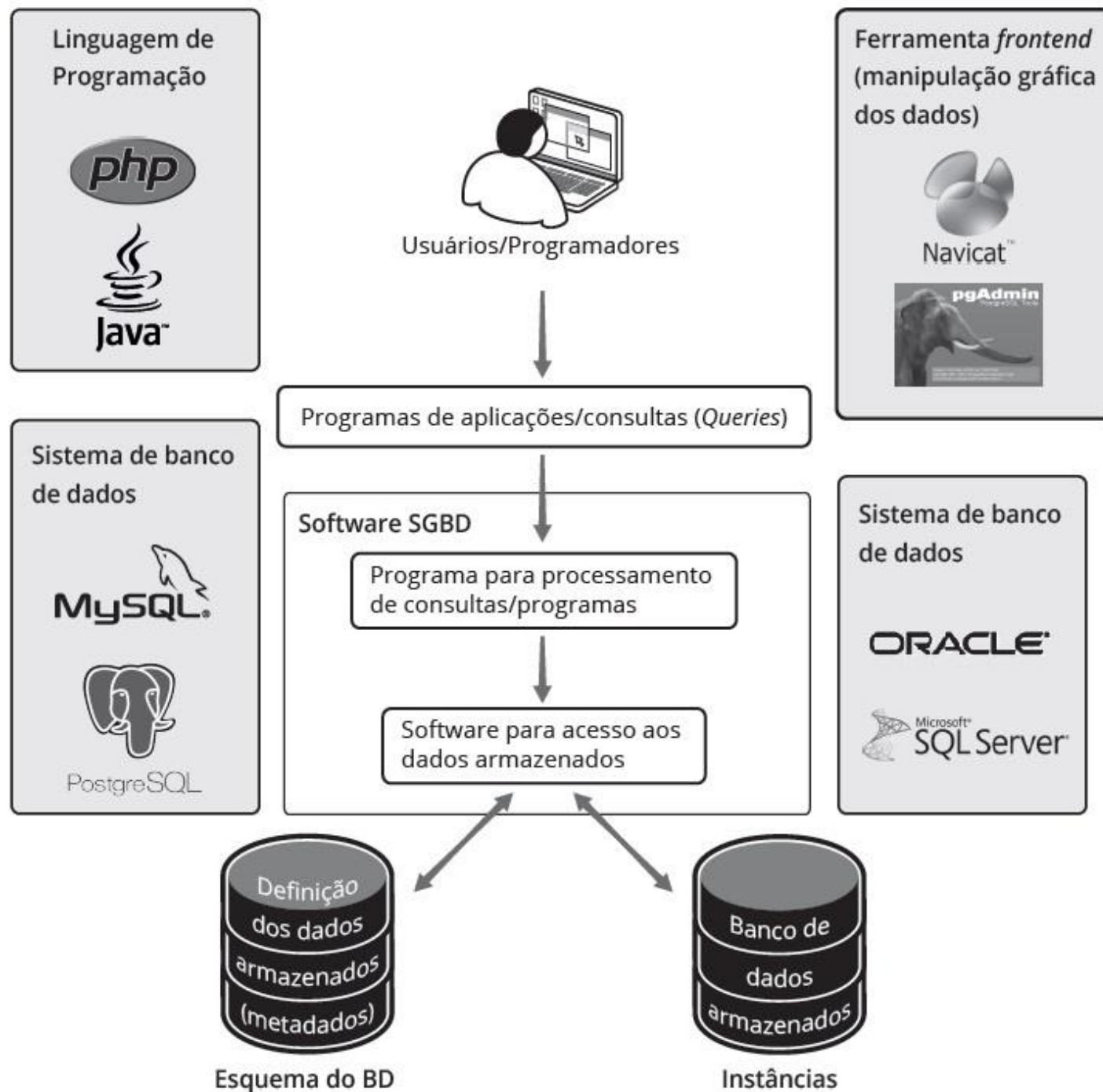
Características de um SGBD:

- Gerenciamento de grande quantidade de dados.
- Evitar redundância e inconsistência de dados.
- Concorrência de acesso.
- Facilidade de acesso.
- Segurança de dados.
- Garantia de integridade.
- Facilidade de migração.
- Suporte a Transações.
- Exemplos: PostgreSQL, MySQL, OracleDB, MS SQLServer etc.

Arquitetura de um SGBD

- De forma simplificada, um SGBD faz a interface entre a camada física de armazenamento dos dados (discos, storage, métodos de acesso, clustering de dados etc.) e a sua organização lógica (instâncias) através de um determinado modelo de organização (esquema ou subesquema).
- Linguagens de programação e ferramentas front-end visuais gráficas são algumas soluções de software que auxiliam usuários na construção, manutenção e manipulados dos dados armazenados em bancos de dados nos SGBDs. Internamente, um SGBD apresenta linguagens específicas para trabalhar com seus de dados, cuidando da sua definição (Data Definition Language – DDL), manipulação (Data Manipulation Language – DML) e consultas (Query Language).

Arquitetura de um SGBD e ferramentas complementares



Organização de SGBDs

Os principais modelos de organização de SGBDs atualmente existentes são:

- Modelo Hierárquico.
- Modelo de Rede.
- Modelo Relacional.
- Modelo Orientado a Objetos.
- Modelo Objeto-Relacional.
- Modelo NoSQL (Not only SQL).

A modelagem de dados representa a descrição formal da estrutura de um SGBD. A evolução tecnológica, tanto de hardware como de software, permitiu que diferentes alternativas para a organização da estrutura de SGBDs fossem testadas e aprimoradas.

Modelo hierárquico

- O modelo hierárquico foi o primeiro a ser reconhecido como um modelo de dados. Seu desenvolvimento somente foi possível devido à consolidação dos discos de armazenamento endereçáveis, pois esses discos possibilitaram a exploração de sua estrutura de endereçamento físico para viabilizar a representação hierárquica das informações. Nesse modelo de dados, os dados são estruturados em hierarquias ou árvores. Os nós das hierarquias contêm ocorrências de registros, onde cada registro é uma coleção de campos (atributos), cada um contendo apenas uma informação. O registro da hierarquia que precede a outros é o registro-pai, os outros são chamados de registros-filhos.
- Grande parte das restrições e consistências de dados estava contida dentro dos programas escritos para as aplicações. Era necessário escrever programas na ordem para acessar o banco de dados.

Modelo em rede

- O modelo em redes surgiu como uma extensão ao modelo hierárquico, eliminando o conceito de hierarquia e permitindo que um mesmo registro estivesse envolvido em várias associações.
- No modelo em rede, os registros são organizados em grafos, onde aparece um único tipo de associação (set) que define uma relação 1:N entre 2 tipos de registros: proprietário e membro.
- O gerenciador Data Base Task Group (DBTG), da Committee on Data Systems and Languages (CODASYL) estabeleceu uma norma para esse modelo de banco de dados, com linguagem própria para definição e manipulação de dados.

Modelo em rede

- Os dados tinham uma forma limitada de independência física. A única garantia era a de que o sistema deveria recuperar os dados para as aplicações como se eles estivessem armazenados na maneira indicada nos esquemas. Os geradores de relatórios da CODASYL também definiram sintaxes para dois aspectos-chaves dos sistemas gerenciadores de dados: concorrência e segurança. O mecanismo de segurança fornecia uma facilidade na qual parte do banco de dados (ou área) pudesse ser bloqueada para prevenir acessos simultâneos, quando necessário. A sintaxe da segurança permitia que uma senha fosse associada a cada objeto descrito no esquema.
- Ao contrário do Modelo Hierárquico, em que qualquer acesso aos dados passa pela raiz, o modelo em rede possibilita acesso a qualquer nó da rede sem passar pela raiz.

Modelo relacional

O modelo relacional apareceu devido às seguintes necessidades:

- Aumentar a independência de dados nos sistemas gerenciadores de banco de dados;
- Prover um conjunto de funções apoiadas em álgebra relacional para armazenamento e recuperação de dados;
- Permitir processamento dedicado e exclusivo.

O modelo relacional, tendo por base a teoria dos conjuntos e álgebra relacional, foi resultado de um estudo teórico realizado por um pesquisador da IBM chamado Ted Codd, que escreveu um artigo na década de 70 propondo um novo modelo para armazenamento e recuperação de dados.

Modelo relacional

- O modelo relacional revelou-se ser o mais flexível e adequado ao solucionar os vários problemas que se colocam no nível da concepção e implementação das bases de dados.
- A estrutura fundamental do modelo relacional é a relação (tabela). Uma relação é constituída por um ou mais atributos (campos) que traduzem o tipo de dados a armazenar.
- Cada instância do esquema (linha) é chamada de tupla (registro).
- O modelo relacional não tem caminhos predefinidos para se fazer acesso aos dados como nos modelos que o precederam. O modelo relacional implementa estruturas de dados organizadas em relações.

Modelo relacional

- Porém, para trabalhar com essas tabelas, algumas restrições precisaram ser impostas para evitar aspectos indesejáveis, como: repetição de informação, incapacidade de representar parte da informação e perda de informação. Essas restrições são: integridade referencial, chaves e integridade de junções de relações.

Banco de dados relacional:

- Item de dado: campo, coluna, atributo;
- Registro: linha, tupla;
- Tabela;
- Manipulação com SQL: operações **CRUD** (Create, Read, Update e Delete)

Modelo Orientado a Objetos

- Os bancos de dados Orientados a Objeto começaram a se tornar comercialmente viáveis em meados de 1980. A motivação para seu surgimento está em função dos limites de armazenamento e representação semântica impostas no modelo relacional. Alguns exemplos são os sistemas de informações geográficas (SIG), os sistemas CAD e CAM, que são mais facilmente construídos usando tipos complexos de dados. A habilidade para criar os tipos de dados necessários é uma característica das linguagens de programação orientadas a objetos.
- Contudo, esses sistemas necessitam guardar representações das estruturas de dados que utilizam no armazenamento permanente. A estrutura padrão para os bancos de dados orientados a objeto foi feita pelo Object Database Management Group (ODMG). Esse grupo é formado por representantes dos principais fabricantes de banco de dados orientados a objeto disponíveis comercialmente. Membros do grupo têm o compromisso de incorporar o padrão em seus produtos.

Modelo Orientado a Objetos

- Quando os bancos de dados orientados a objetos foram introduzidos, algumas das falhas perceptíveis do modelo relacional pareceram ter sido solucionadas com essa tecnologia e acreditava-se que tais bancos de dados ganhariam grande parcela do mercado. Hoje, porém, acredita-se que os bancos de dados Orientados a Objetos serão usados em aplicações especializadas, enquanto os sistemas relacionais continuarão a sustentar os negócios tradicionais, onde as estruturas de dados baseadas em relações são suficientes. O diagrama de classes UML serve geralmente como o esquema para o modelo de dados Orientado a Objetos.

Modelo Objeto-Relacional

- Alguns bancos de dados relacionais adicionaram a seus produtos a capacidade de incorporar objetos mais complexos, como imagem, som e vídeo, bem como alguns recursos de orientação a objetos.
- No entanto, isso não os torna sistemas puramente orientados a objetos, apesar da denominação Object-Relational Database Management System (ORDMS). Esse modelo prevê a implementação de uma camada de abstração de dados em cima dos métodos relacionais, o que torna possível a manipulação de dados mais complexos.

Modelo NoSQL

- Um dos grandes desafios atualmente na área de computação é a manipulação e processamento de grande quantidade de dados no contexto de Big Data.
- NoSQL (Not only SQL: Não só SQL) é um termo utilizado para definir um tipo de banco de dados que não segue normas de tabelas (schemas) presente no banco de dados relacional.
- A quantidade de dados gerada diariamente em vários domínios de aplicação como, por exemplo, da web, rede sociais, redes de sensores, dados de sensoramento, entre diversos outros, estão na ordem de algumas dezenas, ou centenas, de Terabytes.

Modelo NoSQL

- Uma das tendências para solucionar os diversos problemas e desafios gerados pelo contexto
- Big Data é o movimento denominado NoSQL. NoSQL promove diversas soluções inovadoras
- de armazenamento e processamento de grande volume de dados. Essas soluções
- foram inicialmente criadas para solucionar problemas gerados por aplicações, por exemplo,
- web 2.0, que na sua maioria necessitam operar com grande volume de dados, tenham uma
- arquitetura que “escale” com grande facilidade de forma horizontal, permitam fornecer
- mecanismos de inserção de novos dados de forma incremental e eficiente, além da necessidade
- de persistência dos dados em aplicações nas nuvens (cloud computing).

Bancos de dados NoSQL



Cassandra

mongoDB



membase



Sistema de Informação

- Um Sistema de Informação (SI) é um conjunto de elementos ou componentes inter-relacionados que coleta (entrada), manipula (processo), armazena e dissemina dados (saída) e informações, além de fornecer um mecanismo de realimentação (ação corretiva) para garantir a realização de um determinado objetivo.
- Informações são resultados obtidos pela seleção, sumarização e apresentação de dados de uma forma que seja útil aos interessados. Geralmente é obtida quando alguma atividade mental humana (observação e análise) é realizada com sucesso sobre dados para revelar seu significado ou sentido. Assim, vale novamente tecer algumas considerações sobre Dados x Informação.

Sistema de Informação: Dados x Informação:

- **Dados:** fatos que podem ser armazenados. Exemplo: cor dos olhos, idade, sexo etc.
- **Informação:** realidade observada sobre os dados. Exemplo: quantidade de pessoas com cor dos olhos castanho, a média de idade das pessoas do sexo masculino.
 - **Importância da Informação:** necessidade de qualidade, eficácia, informações mais confiáveis e rápidas para a tomada de decisão.
 - **Uso de recursos da TI:** envolve hardware, software, pessoas, banco de dados, redes, procedimentos

Sistema de Informação

- Os dados são constituídos de fatos crus, geralmente representando dados do mundo real, como o número de um funcionário ou total de horas trabalhadas, podendo ser de vários tipos: numéricos, alfanuméricos, imagem, vídeo, som etc. O processo de organização desses fatos crus de forma significativa os tornam uma informação.

Sistema de Informação

- Informação é um conjunto de fatos organizados de tal maneira que possuem valor adicional, além do valor dos fatos individuais (crus). Por exemplo, os gerentes de vendas de uma determinada empresa podem crer que conhecer o total de vendas de sua empresa em um determinado período seja mais importante do que o número de vendas de cada um de seus funcionários. Já o funcionário deve se preocupar com o seu total de vendas frente a uma possível meta a ser alcançada.

Desenvolvimento de SI

- O desenvolvimento de sistemas envolve criar ou modificar os sistemas de negócio existentes. As principais etapas desse processo e seus objetos incluem investigação de sistemas (ter entendimento claro de qual é o problema), análise de sistemas (definir o que o sistema deve fazer para resolver o problema), projeto de sistema (determinar exatamente como o sistema vai funcionar para atender as necessidades do negócio), implantação do sistema (criar ou adquirir os vários componentes do sistema definidos nas etapas de projeto) e manutenção e revisão do sistema (manter e depois modificar o sistema para que ele continue a atender as necessidades evolutivas do negócio).

Desenvolvimento de SI

- Para que essas etapas possam ser alcançadas, podemos partir de um plano de que propõe um ciclo de vida com vistas à organização para o processo de desenvolvimento de um SI, quais sejam:
- Desenvolvimento de um SI: informalmente, é iniciado como resultado de um “entendimento verbal” entre o contratante e o desenvolvedor.
- Plano de desenvolvimento (organização): CICLO DE VIDA.
 - Definir atividades a serem executadas;
 - Verificar consistência entre as atividades;
 - Introduzir pontos de verificação e validação.

Desenvolvimento de SI

- Nesse ponto temos de ter em mente que há uma diferença entre Sistemas de Informações e o software propriamente dito. O primeiro é formado de partes que interagem entre si, visando um objetivo comum, tais como software, hardware e recursos humanos. O segundo é um conjunto de instruções de programas desenvolvidas para resolver um problema com o uso da computação.
- Outra necessidade importante é a de entendermos o papel do DBA e do Analista de Dados no processo de desenvolvimento de um SI. DBA (DataBase Administrator) é o profissional responsável por gerenciar, instalar, configurar, atualizar e monitorar um banco de dados ou sistemas de bancos de dados, ao passo que o Analista de Dados é o profissional responsável por atuar com administração de banco de dados, desenvolver melhorias, identificar e solucionar problemas.

Desenvolvimento de software x SGBD

- Um SGBD, que é um tipo especial de software, é um componente importante de um sistema ou aplicação que fará uso dos dados nele armazenados. O SGBD em si será um componente que não precisará ele próprio ser desenvolvido, mas a organização de como os dados serão criados e armazenados dependerá diretamente dos objetivos pretendidos para o sistema ou aplicação que será desenvolvido.

Existem várias metodologias (modelos gerais, paradigmas) de desenvolvimento de software, mas em geral poderemos sempre identificar as seguintes fases:

- ✓ Especificação (dos requisitos);
- ✓ Análise e Projeto (construção dos modelos);
- ✓ Implementação (desenvolvimento/codificação);
- ✓ Testes/Homologação (implantação);
- ✓ Manutenção.

Desenvolvimento de software x SGBD

- Ainda que faça parte do escopo desse curso descrever em detalhes diferentes metodologias de desenvolvimento de software, é importante lembrar essas etapas, com destaque para a primeira etapa.
- As especificações de requisitos são objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem suas diversas propriedades. Os requisitos de software são, obviamente, aqueles entre os requisitos de sistema que dizem respeito a propriedades do software. Dessa forma a especificação de requisitos envolve as atividades de determinar os objetivos de um software e as restrições associadas a ele. Ela deve também estabelecer o relacionamento entre esses objetivos e restrições e a especificação precisa do software.

Desenvolvimento de software x SGBD

- Tradicionalmente, os requisitos de software são separados em requisitos funcionais, com a descrição das diversas funções que clientes e usuários querem ou precisam que o software ofereça, e não funcionais, com as qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras.
- Entrevista, observação in-loco e encontros são vistos como algumas das técnicas que podem ser utilizadas na fase de especificação de requisitos.

Desenvolvimento de software x SGBD

- As demais fases de análise e projeto, implementação, teste e manutenção, que complementam
- um ciclo tradicional de desenvolvimento de software, buscam traduzir o conjunto
- de requisitos levantados em uma implementação que permita atingir os objetivos propostos
- (na etapa de manutenção o ciclo se repete de modo a permitir que seja incorporadas ao
- sistema possíveis melhorias que se fizerem necessárias).
- Nosso interesse está com foco na etapa inicial, já que a modelagem do banco de dados dependerá
- diretamente do levantamento dos requisitos que o futuro sistema terá de atender.

Modelos de Sistema

- **Modelos de Sistema**

Tipos de Modelos:

- Funcional (processos).
- De Dados.
- Comportamental (tempo-dependente).

Antes de construir um SI, deve-se elaborar um modelo (planta) que seja capaz de expressar, com a máxima fidelidade e simplicidade possíveis, o ambiente no qual ele se insere visando satisfazer todos os requisitos identificados. Isso nos possibilita ter um maior domínio sobre o problema, uma vez que o modelo facilita e diminui os custos de possíveis adequações do projeto.

Modelos de Sistema

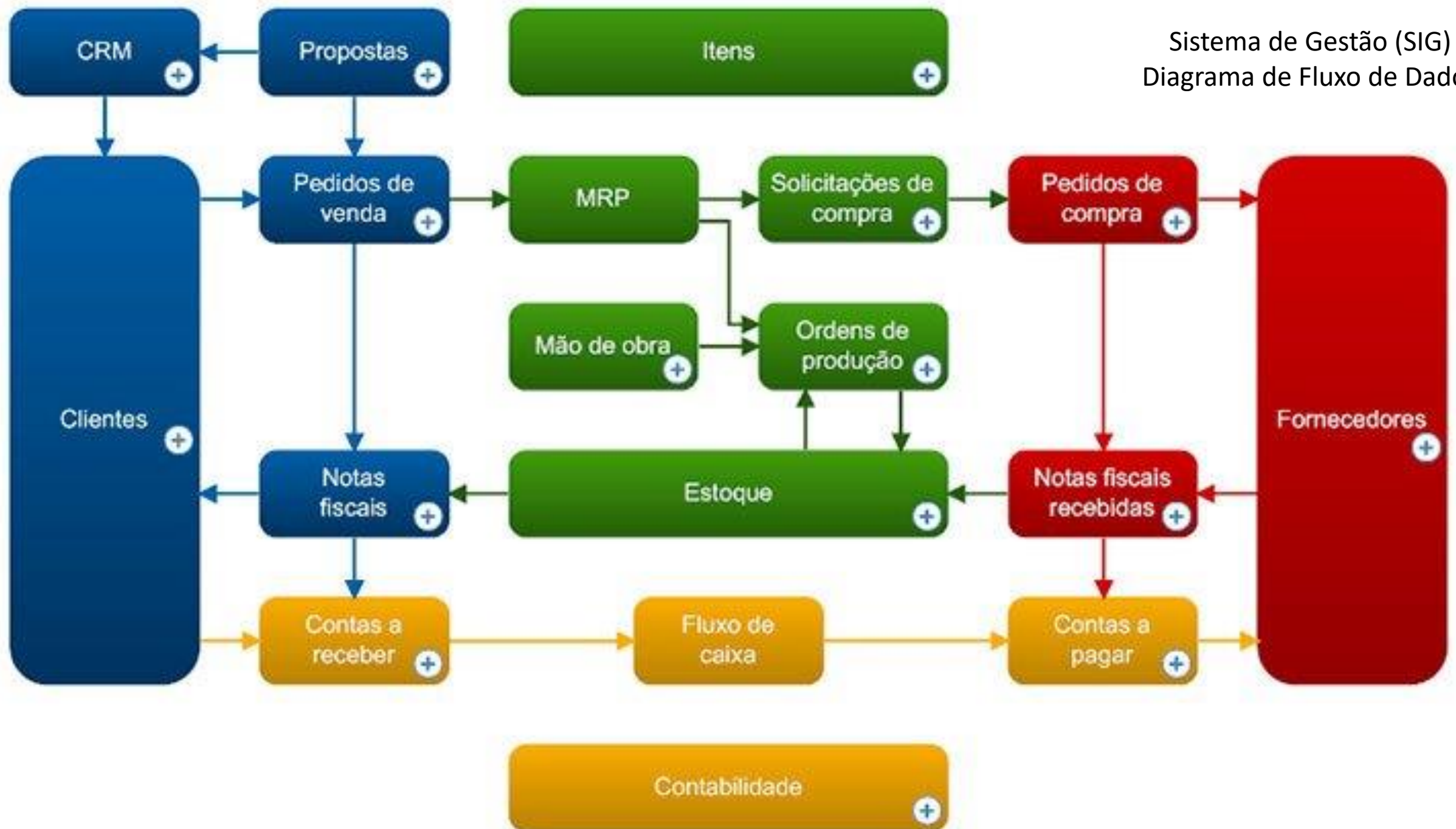
- Cada um dos modelos recém citados busca retratar diferentes aspectos ou dimensões do sistema a ser desenvolvido, sendo vistos como um mecanismo eficaz de comunicação entre técnicos e usuários. Esses modelos lidam também com a questão da complexidade através de níveis de abstração, tornando mais fáceis (e também mais baratas) as discussões em torno da adequação do sistema que será construído. A seguir detalhamos um pouco mais cada um desses modelos.

Modelos de Sistema: Funcional

- Pode ser visto como sendo formado por uma representação gráfica, acompanhada de uma descrição de cada função e das suas interfaces. Costuma ser expressa por meio de Diagrama de Fluxo de Dados (DFD).

Sistema de Gestão (SIG)

Diagrama de Fluxo de Dados



Modelos de Sistema: De dados

- Método da análise de sistemas que busca especificar, a partir de fatos relevantes que estejam associados ao domínio de conhecimento analisado, a perspectiva dos dados, permitindo organizá-los em estruturas bem definidas, estabelecendo as regras de dependência entre eles, produzindo um modelo expresso por uma representação, ao mesmo tempo descritiva e diagramática

Modelos de Sistema: Comportamental

- Descreve de que maneira o sistema, enquanto um conjunto de elementos inter-relacionados, reage, internamente, com um todo organizado, aos estímulos externos. Preocupa-se em mostrar as ações que o sistema deve executar para responder adequadamente aos eventos previstos no modelo ambiental.



Projeto de banco de dados

Projeto de banco de dados



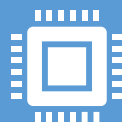
Três níveis de abstração (necessário realizar o mapeamento entre os três modelos):



Modelo Conceitual (DER).



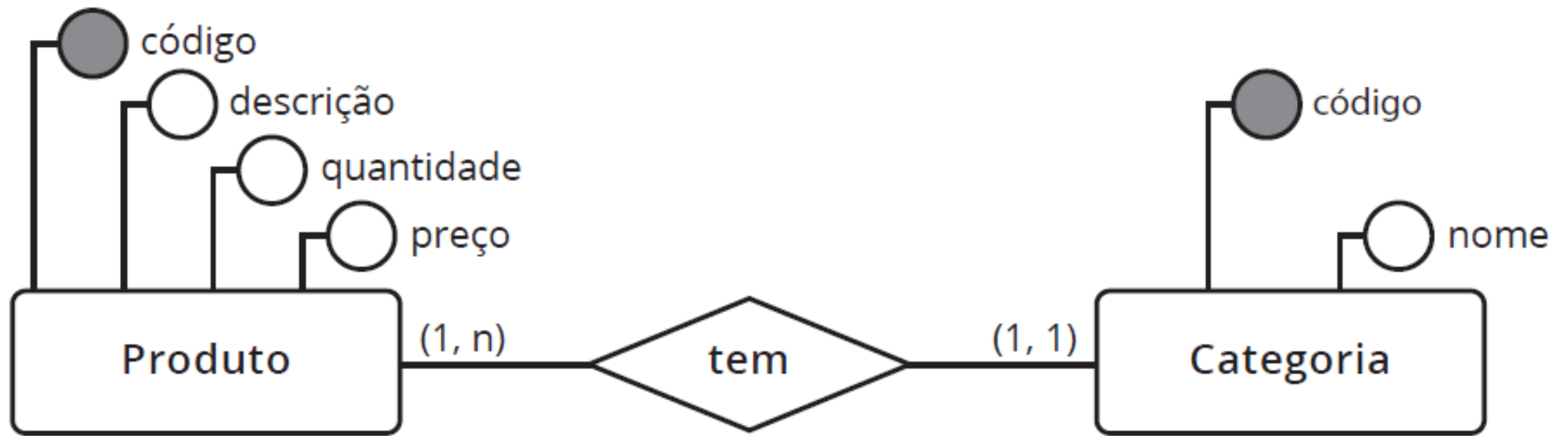
Modelo Lógico (Esquema do BD).



Modelo Físico (Script do BD em SQL).

Projeto de banco de dados

- O projeto de concepção e implementação do banco de dados que vai dar suporte ao sistema sendo desenvolvido deve, igualmente, passar por diferentes estágios de abstração. São enormes as pressões para que a equipe de informática possa responder o mais rapidamente possível às demandas para colocar em funcionamento o novo sistema.
- Assim, não é incomum tomar conhecimento de projetos de desenvolvimento onde as etapas de levantamento de requisitos e modelagem são relegadas a um segundo plano. Esse é um erro comum que costuma cobrar seu preço, em geral muito significativo, nas etapas seguintes do processo de desenvolvimento. Nossa recomendação é não negligenciar nenhuma das seguintes etapas ao projetar um banco de dados.



(DER): Diagrama
Entidade-
Relacionamento

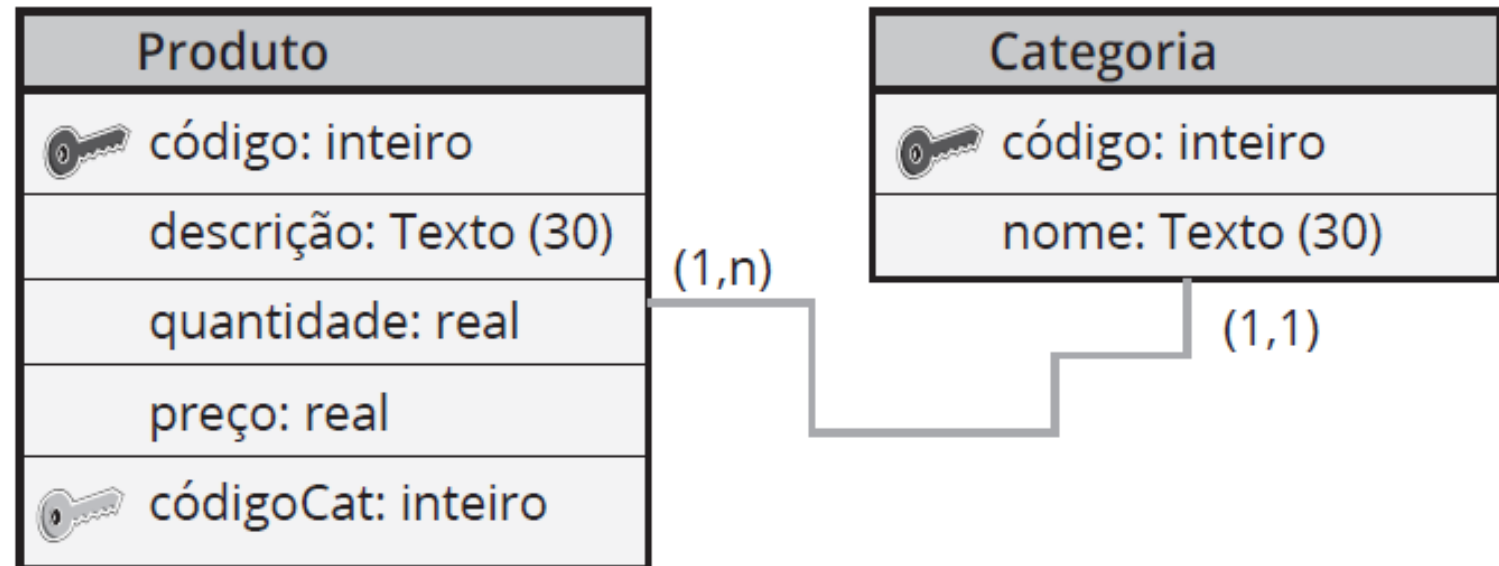
- **Modelo Conceitual (DER):** também conhecido como Diagrama Entidade-Relacionamento, é um modelo de dados abstrato que descreve a estrutura de um banco de dados independente de sua implementação;

Modelo Lógico (Esquema do BD):

- **Modelo Lógico (Esquema do BD):** tem como objetivo transformar o modelo conceitual em um modelo que define como o banco de dados será implementado em um SGBD específico.
- Deve representar relações e restrições do modelo de dados que representa a estrutura de um BD e o Esquema do Banco de Dados;

Categoria (@codCat: inteiro, nomeCat: literal[30])

Produto (@codProd: inteiro, descProd: literal[50],
quantProd: real, precoProd: real, codCatProd: inteiro)



Modelo Físico (Script do BD em SQL):

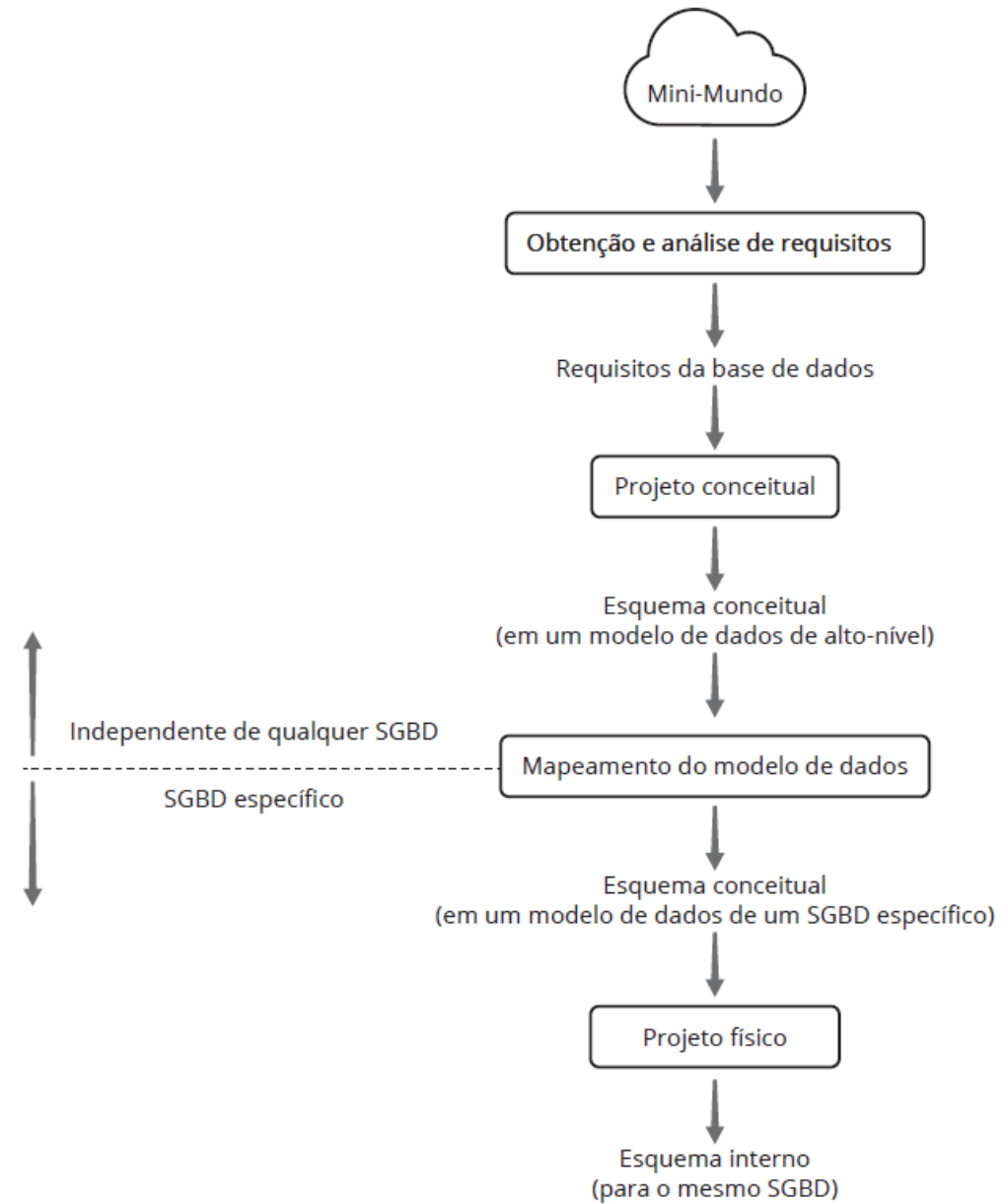


- **Modelo Físico (Script do BD em SQL):** nessa fase, o modelo do banco de dados é enriquecido com detalhes que influenciam no desempenho do banco de dados, mas não interferem na sua funcionalidade. Script do banco de dados em SQL representa os detalhes dos dados internamente ao BD (campo, tipo/domínio, restrições).
- CREATE TABLE PRODUTO (
 - codprod INTEGER PRIMARY KEY,
 - quantprod REAL,
 - precoprod REAL,
 - descrprod VARCHAR(30),
 - codcatprod INTEGER
 -);
- CREATE TABLE CATEGORIA (
 - codcat INTEGER PRIMARY KEY,
 - nomecat VARCHAR(30)
 -);
- ALTER TABLE PRODUTO ADD FOREIGN KEY(codcatprod) REFERENCES
- CATEGORIA (codcat);

Etapas da modelagem do banco de dados

- A figura a seguir ilustra o esquema geral das diferentes etapas que serão percorridas ao longo do desenvolvimento de um novo banco de dados no contexto do desenvolvimento de um novo sistema ou aplicação.

Etapas da modelagem do banco de dados



Primeira Etapa – Análise de Requisitos

Levantamento/Estudo de Viabilidade:

- Identificar usuários responsáveis e definir escopo.
- Identificar as deficiências no sistema atual (justificativa).
- Estabelecer metas e objetivos para o novo sistema.
- Determinar se é possível “informatizar” (viabilidade).
- Fazer estimativas (cálculo de Custo/Benefício).
- Preparar um cronograma.

Primeira Etapa – Análise de Requisitos

- É indispensável que o profissional ou equipe envolvidos com o projeto do banco de dados tenha um bom domínio do negócio da sua organização ou empresa. Para tanto, pode e deve fazer uso de diferentes ferramentas para a coleta de dados, tais como entrevistas, análise de procedimentos e documentos, questionário etc. O objetivo é identificar requisitos de dados (fatos do mundo real) que deverão ser observados ou atendidos.
- Essa é uma etapa pré-modelagem, onde o mais comum é elaborar a especificação formal dos dados no formato de um texto descritivo, também chamado de minimundo ou Universo de Discurso, que pode ser validado pelos futuros usuários do sistema em desenvolvimento. A seguir, apresentamos um exemplo de minimundo.

Exemplo de Minimundo: EMPRESA

- A **empresa** está organizada em departamentos. Cada **departamento** tem um nome, um número único e um empregado que gerencia o departamento. Armazena-se a data em que o empregado começou a gerenciar o departamento. Um departamento pode ter diversas localizações;
- Um departamento controla inúmeros **projetos**, sendo que cada um tem um nome, um número único e uma localização;
- Do **empregado** armazena-se o nome, o número do seguro social, endereço, salário, sexo e data de nascimento. Todo empregado é alocado em um departamento, mas pode trabalhar em diversos projetos, que não são necessariamente controlados pelo mesmo departamento. Armazena-se, também, o número de horas semanais que o empregado trabalha em cada projeto. Mantém-se, ainda, a indicação do supervisor direto de cada empregado;
- É feito um controle sobre os dependentes de cada empregado para fins de seguro.
- De cada **dependente** é registrado o nome, sexo, data de nascimento e o parentesco com o empregado.

Segunda Etapa – Análise e projeto

Projeto do BD.

- 1) Modelo Conceitual (DER).
 - 2) Modelo Lógico (Esquema do BD/Relações).
 - 3) Modelo Físico (Script do BD/Tabelas-Restrições).
-
- Com base no texto descritivo, ou minimundo, gerado na primeira etapa, inicia-se o processo de modelagem propriamente dito, indo do mais abstrato para o mais concreto.

1. Descreva com suas palavras. Qual é o principal objetivo de se utilizar o processo de modelagem de dados no processo de desenvolvimento de sistemas?

01

2. Defina os seguintes termos: dados, banco de dados, SGBD, sistema de banco de dados, catálogo de banco de dados, independência entre dados e programas, visão do usuário, DBA, usuário final, transação programada, sistema de banco de dados dedutivo, objeto persistente, metadados e aplicação para processamento de transação.

02

3. Quais os quatro tipos principais de ações que envolvem bancos de dados? Discuta cada tipo rapidamente.

Respondas as questões no DISCORD em exercícios-de-fixação.

Tudo o que temos de decidir é o que fazer com o tempo que nos é dado.

Gandalf



Obrigado!

