
Engenharia de Software

Modelo de Domínio

Luís Morgado

Instituto Superior de Engenharia de Lisboa
Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Modelo de Domínio

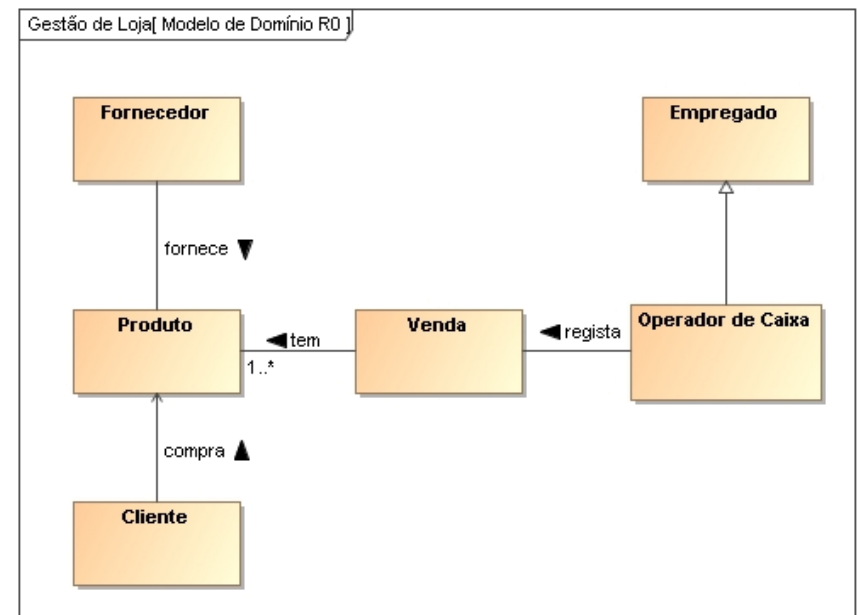
- O modelo de domínio é um modelo conceptual que representa os conceitos do domínio do problema e respectivas relações
- Não inclui pormenores de implementação técnica, sendo o foco nos conceitos do domínio e nas suas relações
 - É independente da forma de realização lógica ou física
- Contribui para clarificar os requisitos e a arquitectura a conceber, pois reflecte a compreensão dos principais conceitos e relações que caracterizam o domínio de operação do sistema
 - Um modelo de domínio ajuda a clarificar os objectivos e os requisitos do sistema a desenvolver, através da compreensão dos conceitos e relações envolvidas e respectivas responsabilidades no domínio do problema
- Fornece uma linguagem comum
 - Um modelo de domínio fornece uma linguagem comum, partilhada com outras partes interessadas, nomeadamente não técnicas, como utilizadores, para compreender e comunicar acerca do domínio do problema

Modelo de Domínio

- O modelo de domínio suporta a transposição do domínio do problema para o domínio da solução
- Os conceitos e relações do domínio do problema são representados por classes e relações conceptuais
- Organizado em diagramas de classes conceptuais



**Domínio do
problema**



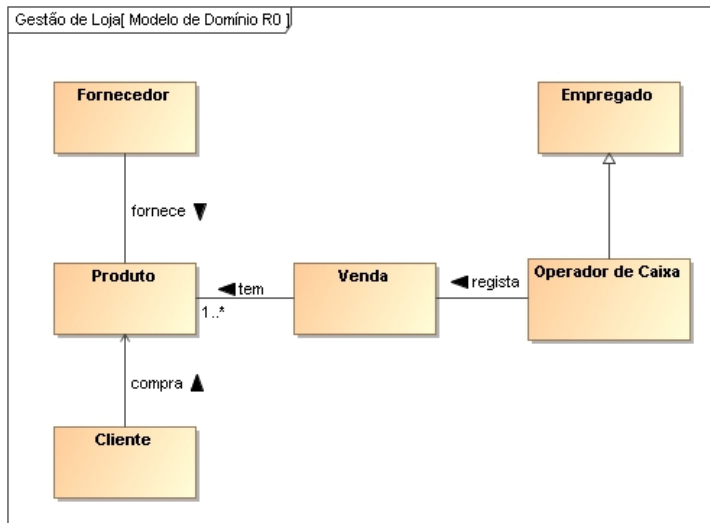
**Domínio da
solução**

Modelo de Domínio

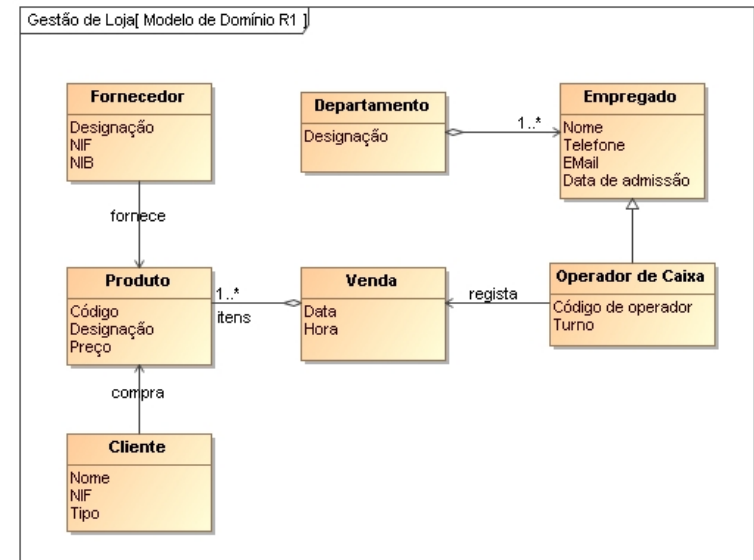
O modelo de domínio é progressivamente detalhado, quer para elaboração da arquitectura lógica de mecanismos e subsistemas, quer para elaboração da arquitectura de dados

- Inicialmente pode incluir apenas a designação dos conceitos e relações entre conceitos
- Progressivamente pode incluir atributos de entidades, domínios de valores e outra informação relevante

Versão inicial

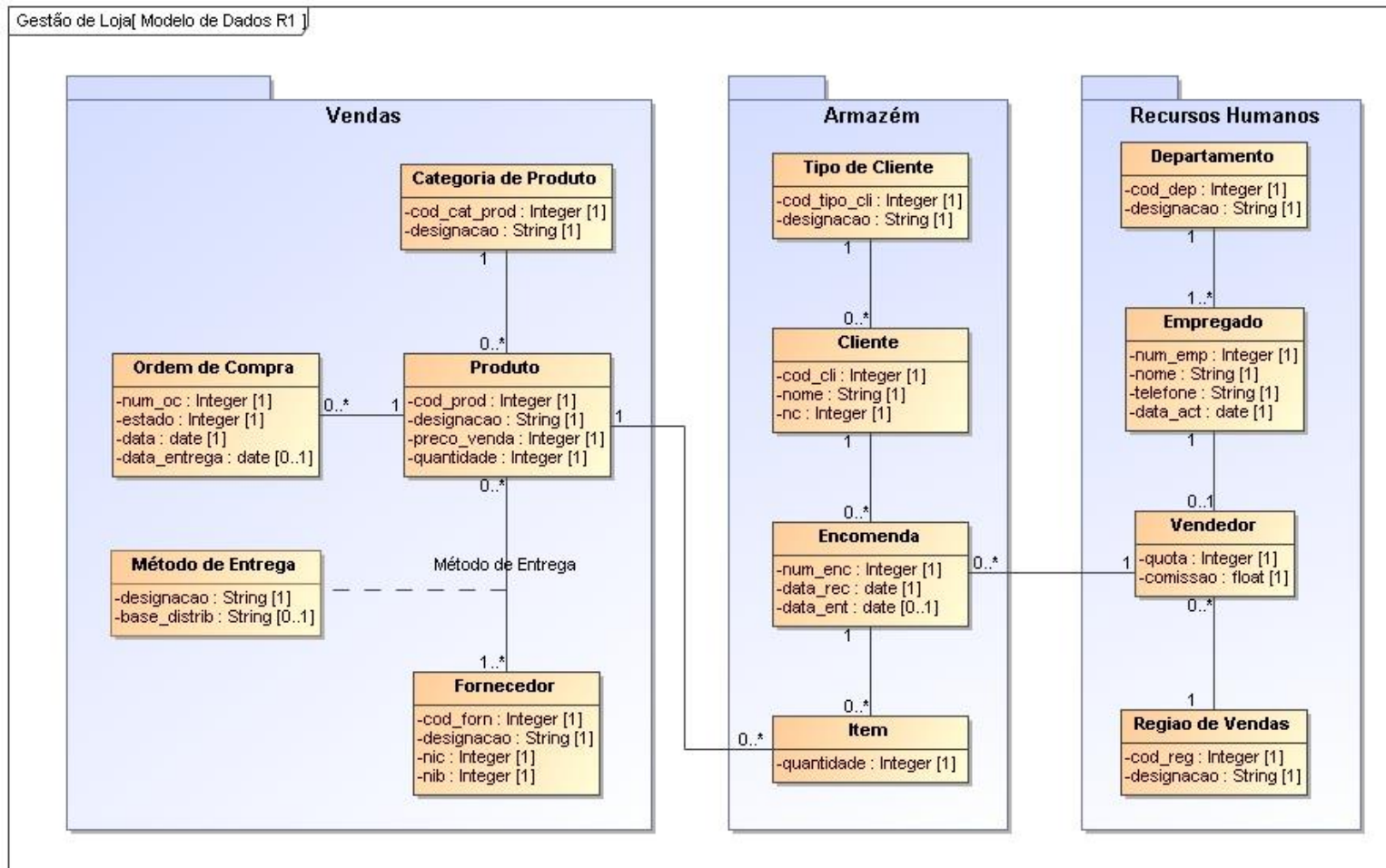


Refinamento com aumento de detalhe



Arquitetura de Dados

O modelo de domínio é a base da elaboração da *arquitetura de dados*, a qual representa os conceitos e relações conceptuais do domínio em termos de entidades e relações lógicas (arquitetura lógica) e correspondentes entidades e relações concretas dependentes da tecnologia (arquitetura física)



Modelo de Domínio

- É um aspecto importante da análise *orientada-a-objectos*
 - Descrição do **domínio do problema** em termos dos **objectos relevantes**
- Elaboração do modelo de domínio
 - Deve representar as **classes e relações conceptuais** que **descrevem as situações reais** de um determinado domínio
 - Foco nos **conceitos**
 - **Não representa classes do domínio da solução** (implementação)
 - Ponte para o domínio da solução (arquitectura e implementação)
 - Relação com o modelo de análise
 - Classes com estereótipo **Entidade** (*Entity*)

Elaboração do Modelo de Domínio

Passos principais:

- Identificar classes conceptuais
 - Identificação das classes conceptuais que existem no domínio do problema
- Elaborar um diagrama de classes com as classes conceptuais
 - Elaboração de uma representação gráfica, sob a forma de um diagrama de classes, das classes de domínio
- Identificar relações entre as classes
 - Identificação das relações entre as classes de domínio e representação dessas relações com base em associações entre as classes
- Definir atributos (propriedades) das classes
 - Definição de atributos ou propriedades das classes de domínio que constituem características dessas classes

Elementos do Modelo de Domínio

- **Entidades (*Classes de Domínio*)**
 - Representam tipos de objectos do domínio do problema
- **Atributos**
 - Representam características das entidades
- **Associações**
 - Representam relações entre entidades que descrevem as dependências entre as instâncias dessas entidades
- **Restrições**
 - Indicam requisitos adicionais representados através de anotações complementares

Entidades (*Classes de Domínio*)

- Cada entidade (*classe de domínio*) representa um tipo de objecto (representa as características comuns das instâncias)
- Exemplos:
 - **Objectos de negócio**
 - Representam coisas que são manipuladas no negócio (e.g. *Produto*, *Factura*)
 - **Objectos diversos**
 - Representam coisas que não são específicas do negócio (e.g. *Contacto*)
 - **Eventos**
 - Representam a ocorrência de condições ou situações relevantes (e.g. *Venda*)

Elaboração do Modelo de Domínio

- **Artefactos base**
 - Visão
 - Modelo de casos de utilização
 - Especificação suplementar
- **Identificação de elementos**
 - **Nomes**
 - Indicam classes, atributos e objectos
 - **Verbos**
 - Indicam operações e relações

Elaboração do Modelo de Domínio

- **Filtragem**

- **Eliminar elementos**

- Vagos
 - Ambíguos
 - Redundantes

- A informação fornecida pelos utilizadores pode ser redundante, ambígua e incompleta
- Interessa-nos
 - Informação útil
 - O que deve ser mantido

- **Manter classes que representam**

- Elementos tangíveis (objectos físicos)
 - Entidades conceptuais
 - Categorias
 - Interfaces para o exterior do sistema
 - Papéis

Modelo de Domínio

Exemplo: Rede ATM

Pretende-se implementar um sistema de gestão de uma rede bancária, que inclui caixas manuais e caixas automáticos (ATM), a ser partilhado por um consórcio de bancos. Cada banco disponibiliza um seu computador para manter contas e processar transacções sobre essas contas. Os caixas manuais pertencem aos bancos individuais e comunicam directamente com os computadores desses bancos. Os caixas manuais inserem dados de contas e de transacções. Os caixas automáticos comunicam com um computador central que coordena as transacções entre bancos. Os caixas automáticos aceitam cartões bancários, interagem com os utilizadores, comunicam com o sistema central para realizar transacções, fornecem dinheiro e imprimem recibos. O sistema necessita de mecanismos de histórico e de segurança. O sistema deve gerir correctamente acessos concorrentes à mesma conta. Os bancos disponibilizam o software dos respectivos computadores; pretende-se o software dos ATM e da rede. O custo do sistema será partilhado pelos bancos de acordo com o número de clientes com cartões bancários.

Exemplo

Identificação de classes:

sistema

rede bancária

mecanismo de histórico

mecanismo de segurança

software

caixa automático

utilizador

dados de conta

dados de transacção

dinheiro

recibo

caixa manual

ATM

consórcio

banco

funcionário

computador do banco

conta

transacção

computador central

cartão bancário

cliente

[Rumbaugh 1991]

Exemplo

Eliminação de classes não relevantes:

sistema

Vago

rede bancária

Vago

mecanismo de histórico

Vago

mecanismo de segurança

Vago

software

Vago

caixa automático

Redundante

utilizador

Redundante

dados de conta

Atributo

dados de transacção

Atributo

dinheiro

Atributo

recibo

Atributo

caixa manual

ATM

consórcio

banco

funcionário

computador do banco

conta

transacção

computador central

cartão bancário

cliente

← (nomes que descrevem objectos individuais devem ser descritos como atributos)

Exemplo

Identificação de classes:

caixa manual

ATM

consórcio

banco

funcionário

computador do banco

conta

transacção

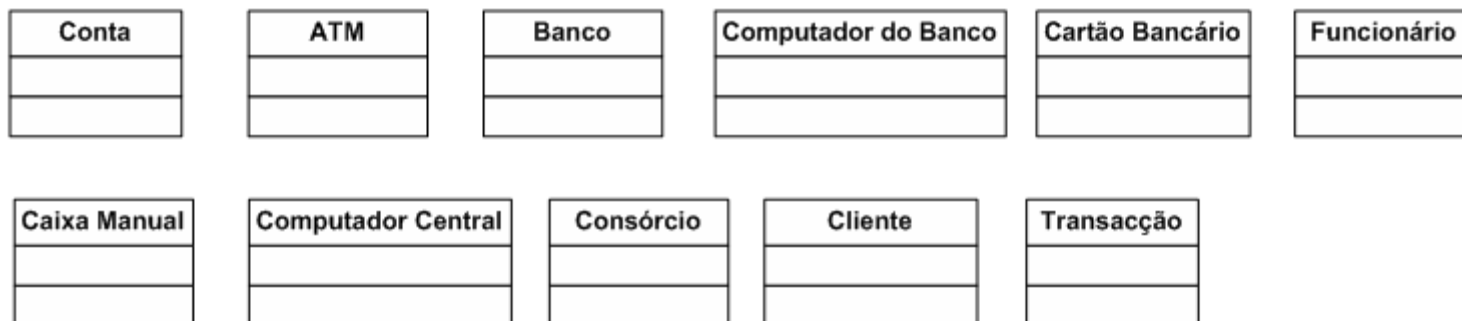
computador central

cartão bancário

cliente



Classes candidatas:



[Rumbaugh, 1991]

Exemplo

Identificação de relações:

Frases com verbos:

consórcio partilha ATMs
banco disponibiliza computador do banco
computador do banco mantém contas
computador do banco processa transacções sobre contas
caixas manuais pertencem aos bancos
caixas manuais comunicam com os computadores dos bancos
funcionários inserem transacções em contas
caixas automáticos comunicam com um computador central
computador central coordena transacções entre bancos
ATMs aceitam cartões bancários
ATMs interagem com os utilizadores
ATMs fornecem dinheiro
ATMs imprimem recibos

Frases implícitas com verbos:

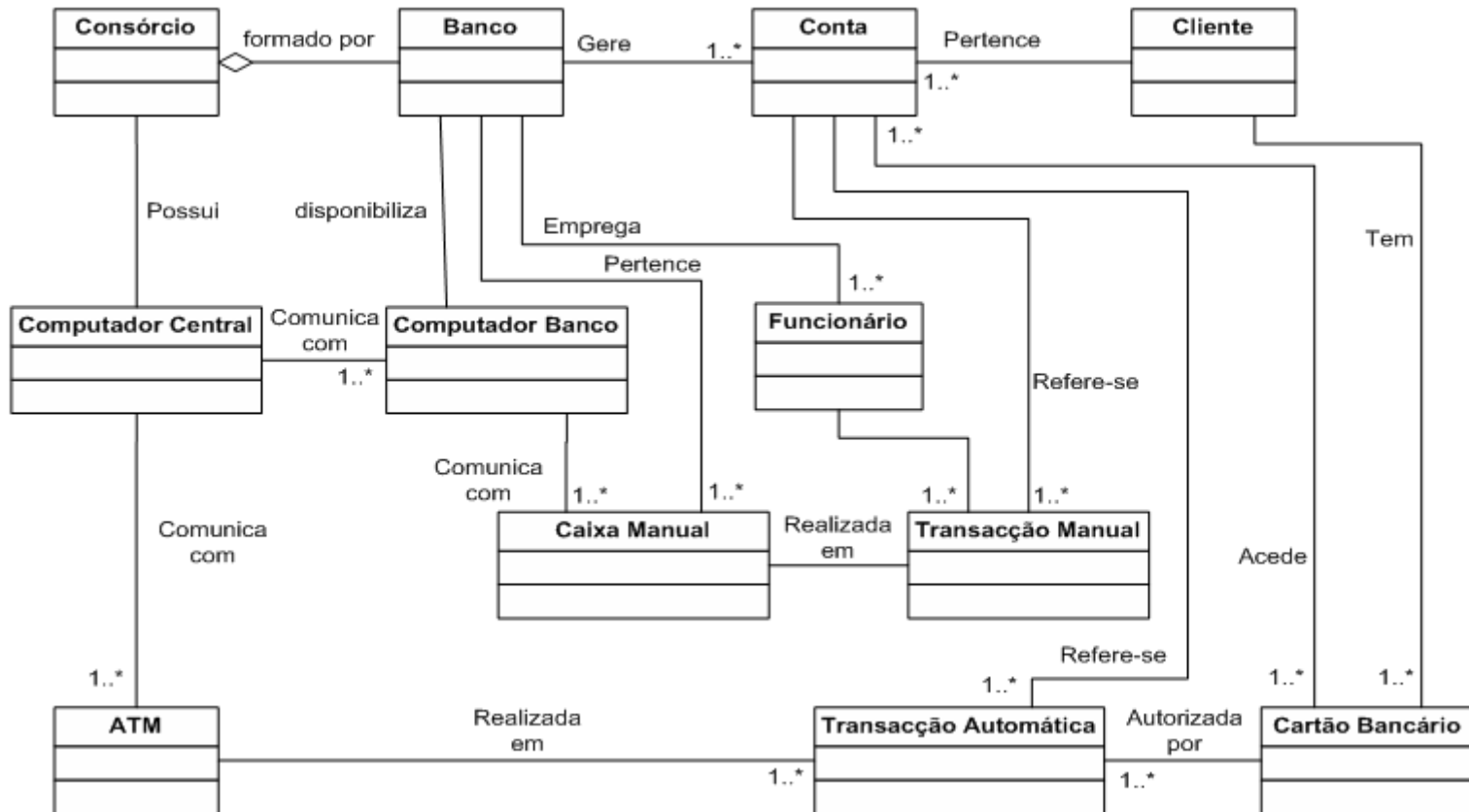
consórcio é formado por bancos
bancos gerem contas
consórcio possui um computador central
clientes possuem cartões bancários

Conhecimento do domínio do problema:

cartões bancários permitem o acesso a contas
banco emprega funcionários

Exemplo

Definição de relações:

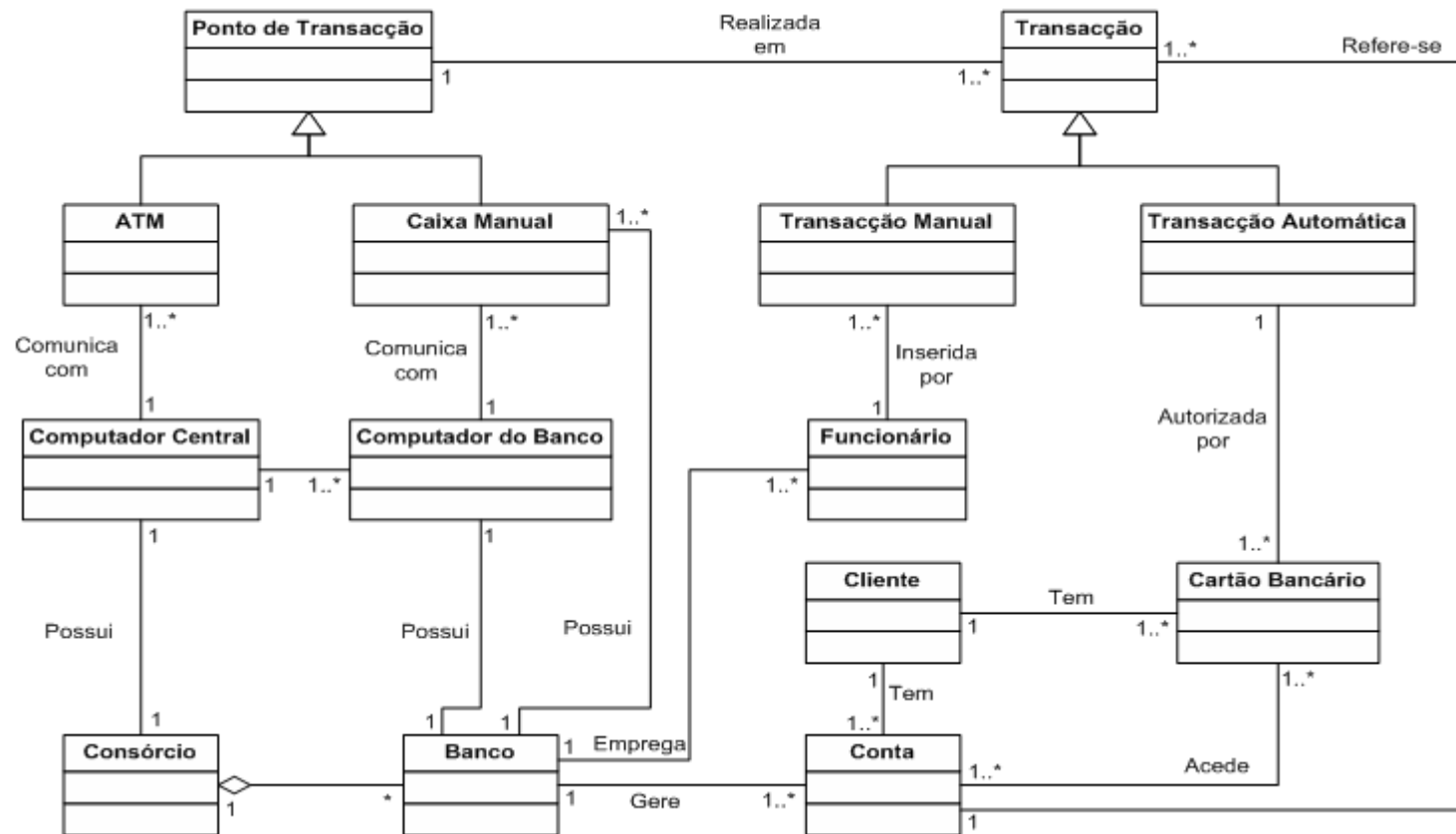


[Rumbaugh, 1991]

Exemplo

Organização e simplificação do modelo

Em particular, factorização com base em herança para eliminação de redundância



[Rumbaugh, 1991]

- **Características**

- Proporciona um enquadramento conceptual das entidades do domínio do problema
- Ajuda a perceber o problema
 - Foco na semântica do problema
- Perspectiva estrutural
 - Identificação de requisitos não funcionais, em particular requisitos de dados
- Perspectiva comportamental
 - Identificação de actividades
 - Regras de negócio
 - Fluxos de trabalho
- Suporta a concepção da arquitectura do sistema, nomeadamente a realização de casos de utilização e a arquitectura de dados

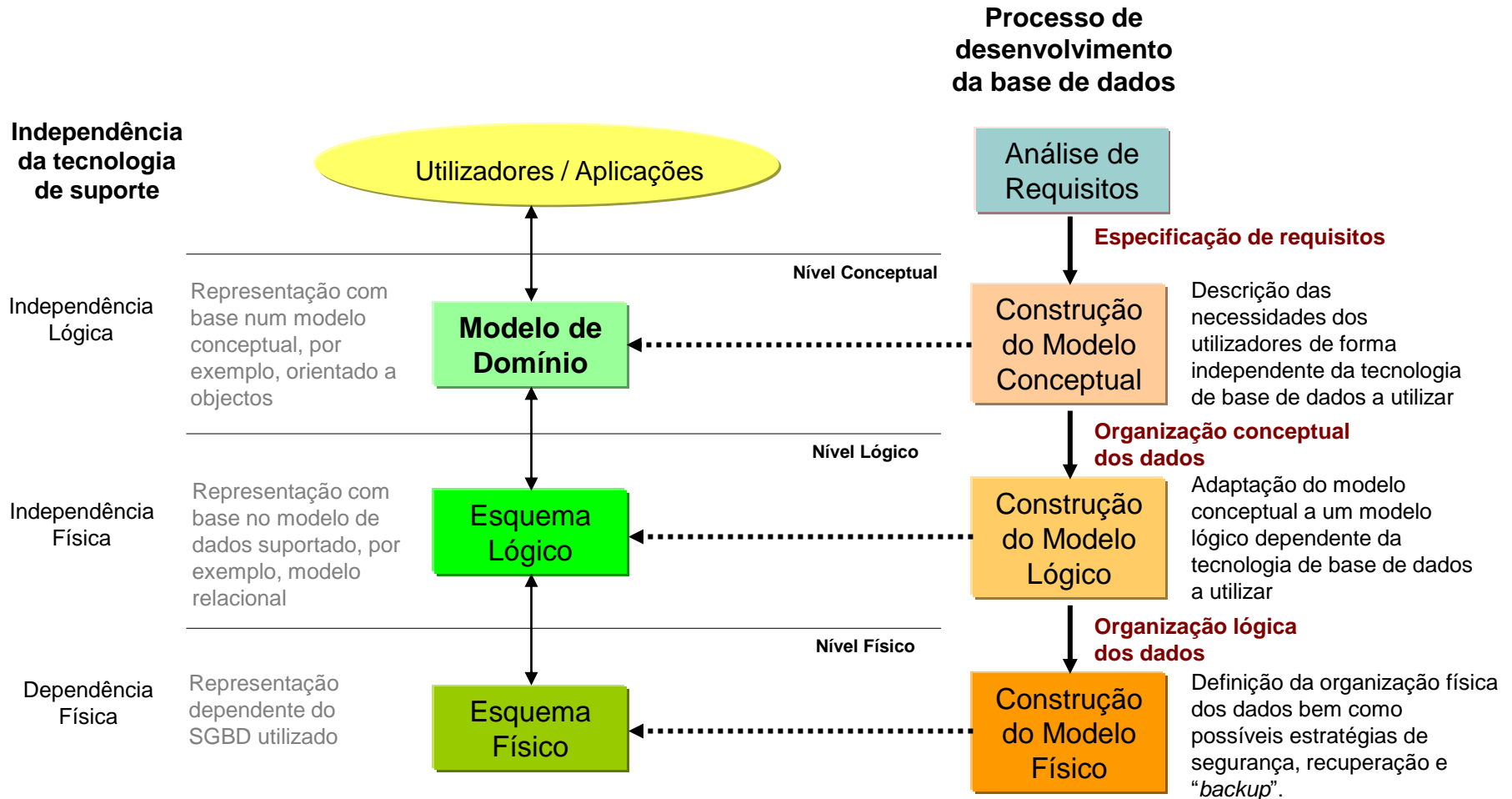
Arquitetura de Dados

A arquitetura de dados é uma representação a diferentes níveis de abstracção de dados mantidos de forma persistente por uma aplicação

- Organiza elementos de dados e define a forma como estes se relacionam entre si e representam as propriedades de entidades do domínio da aplicação
- Elaboração baseada no modelo de domínio
- Organizada em três níveis principais de representação:
 - **Modelo conceptual:** representa os conceitos e as restrições que regem os dados na perspectiva do domínio do problema, de forma independente da tecnologia de persistência de dados a utilizar
 - **Modelo lógico:** representa a estrutura e as relações dos dados de forma detalhada mas independente do suporte físico de persistência dos dados
 - **Modelo físico:** representa os aspectos concretos de armazenamento e manipulação dos dados tendo em conta o suporte específico de persistência dos dados

Arquitetura de Dados

Representação a diferentes níveis de abstracção, de dados mantidos de forma persistente por uma aplicação

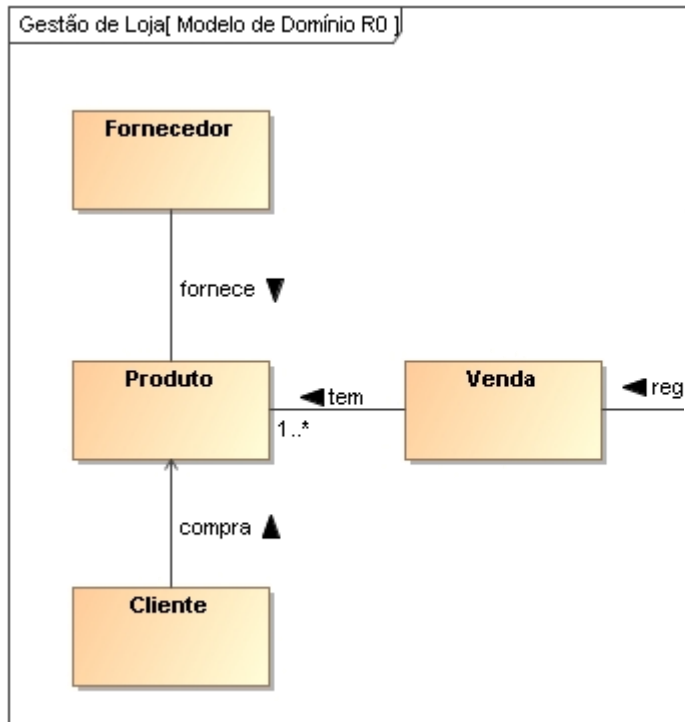


Arquitetura de Dados: Exemplo

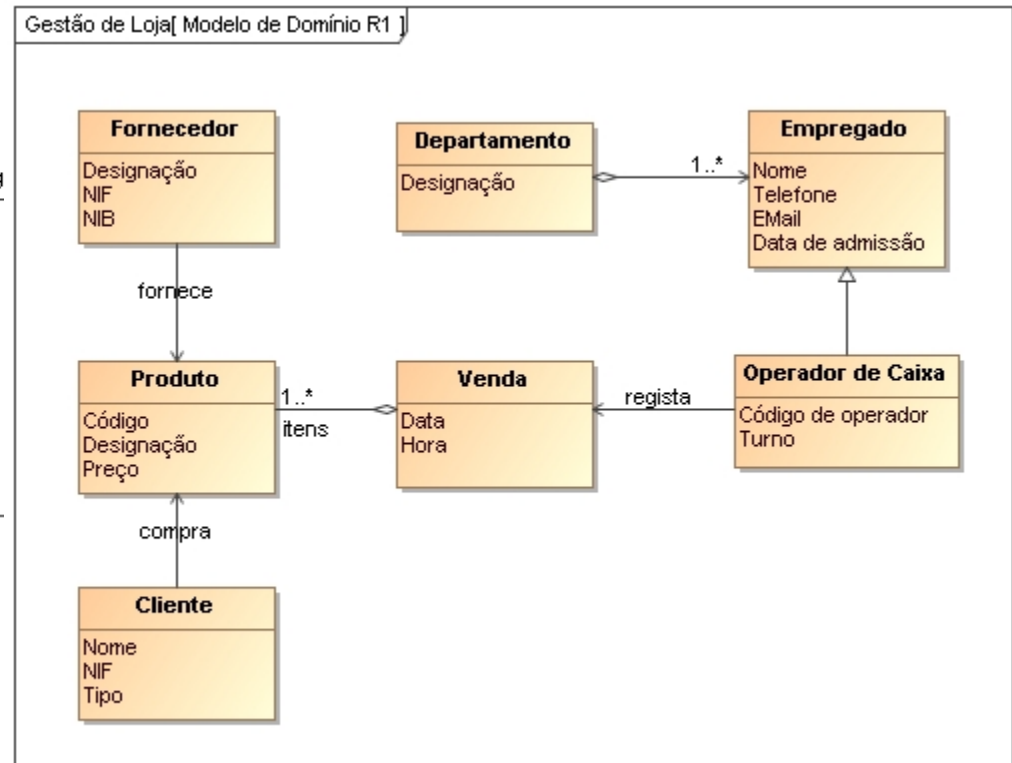
Modelo de Domínio

Base de elaboração da arquitectura de dados

Início



Refinamento



Arquitetura de Dados: Exemplo

Modelo Lógico (e.g. Modelo Relacional)

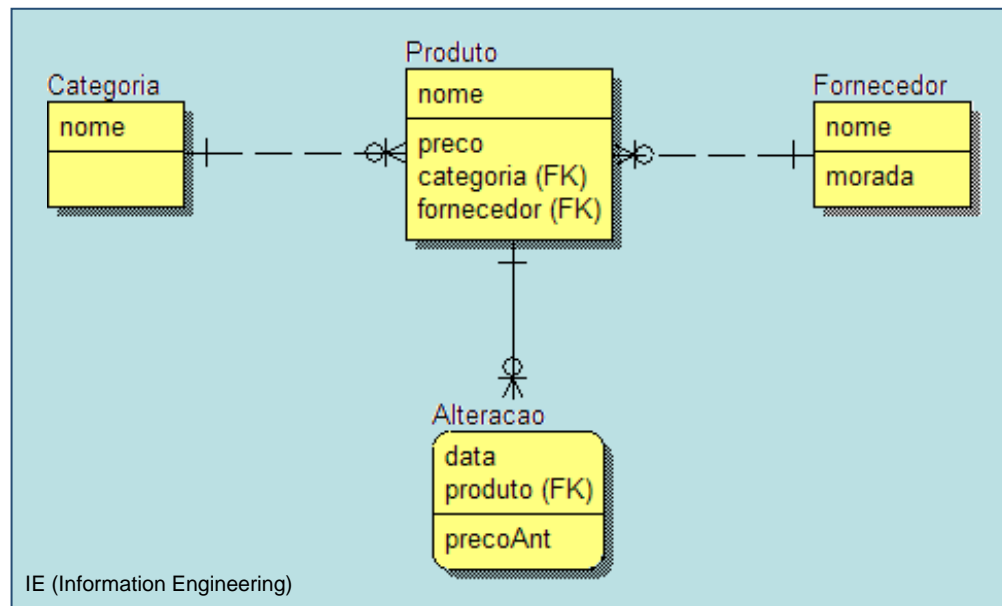
Independente da plataforma de persistência de dados

Produto(nome, preco, categoria [FK], fornecedor [FK])

Categoria(nome)

Fornecedor(nome, morada)

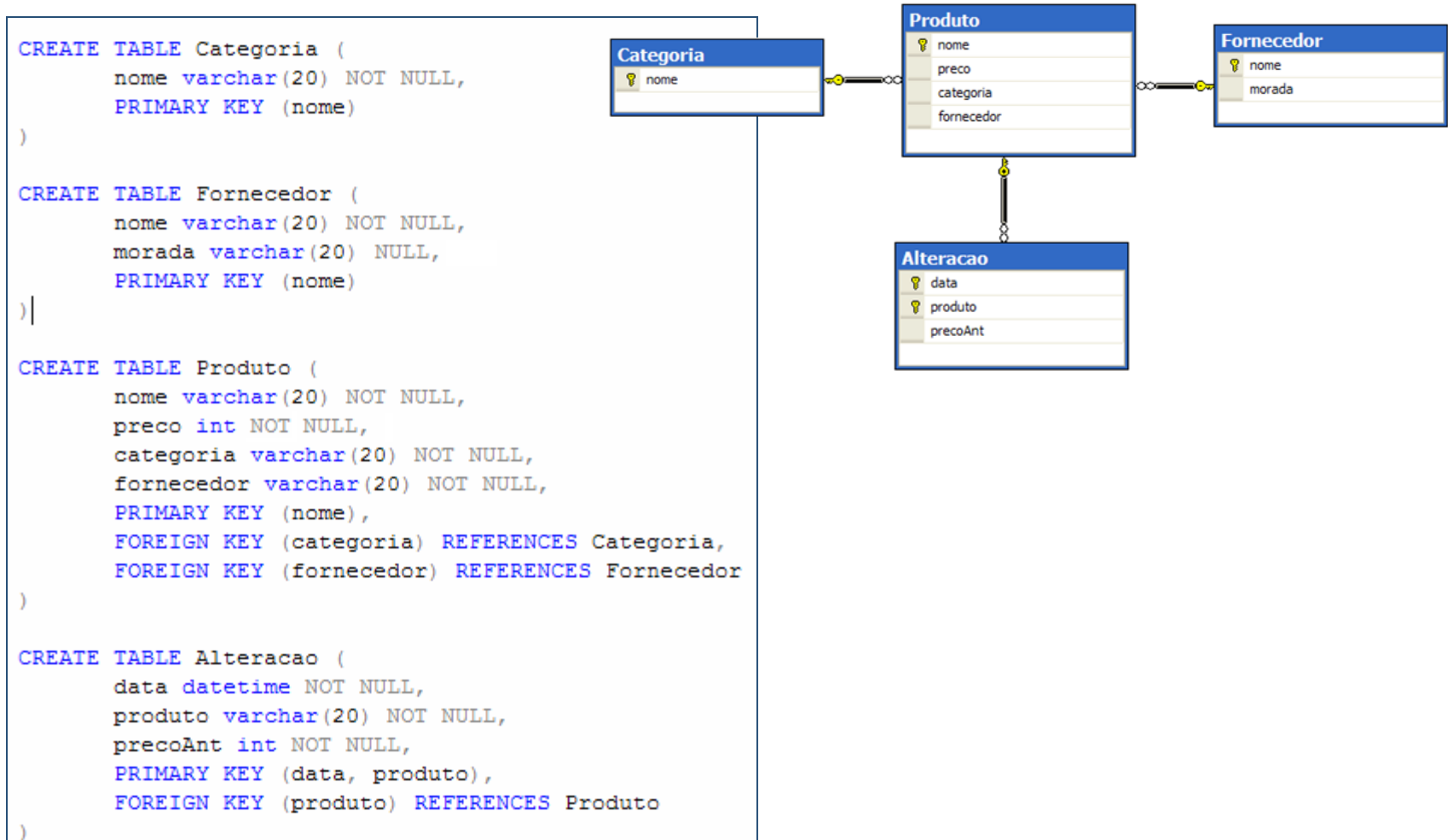
Alteração(produto [FK], data, preco)



Arquitetura de Dados: Exemplo

Modelo Físico

Dependente da plataforma de persistência de dados



Bibliografia

[Selic, 2003]

B. Selic, *Brass bubbles: An overview of UML 2.0*, Object Technology Slovakia, 2003.

[Graessle, 2005]

P. Graessle, H. Baumann, P. Baumann, *UML 2.0 in Action*, Packt Publishing, 2005.

[Eriksson et al., 2004]

H. Eriksson, M. Penker, B. Lyons, D. Fado, *UML 2 Toolkit*, Wiley, 2004.

[Rumbaugh, 1991]

J. Rumbaugh, *Object-oriented Modeling and Design*, Prentice-Hall, 1991.

[OMG, 2020]

Unified Modeling Language (Specification), OMG, 2020.