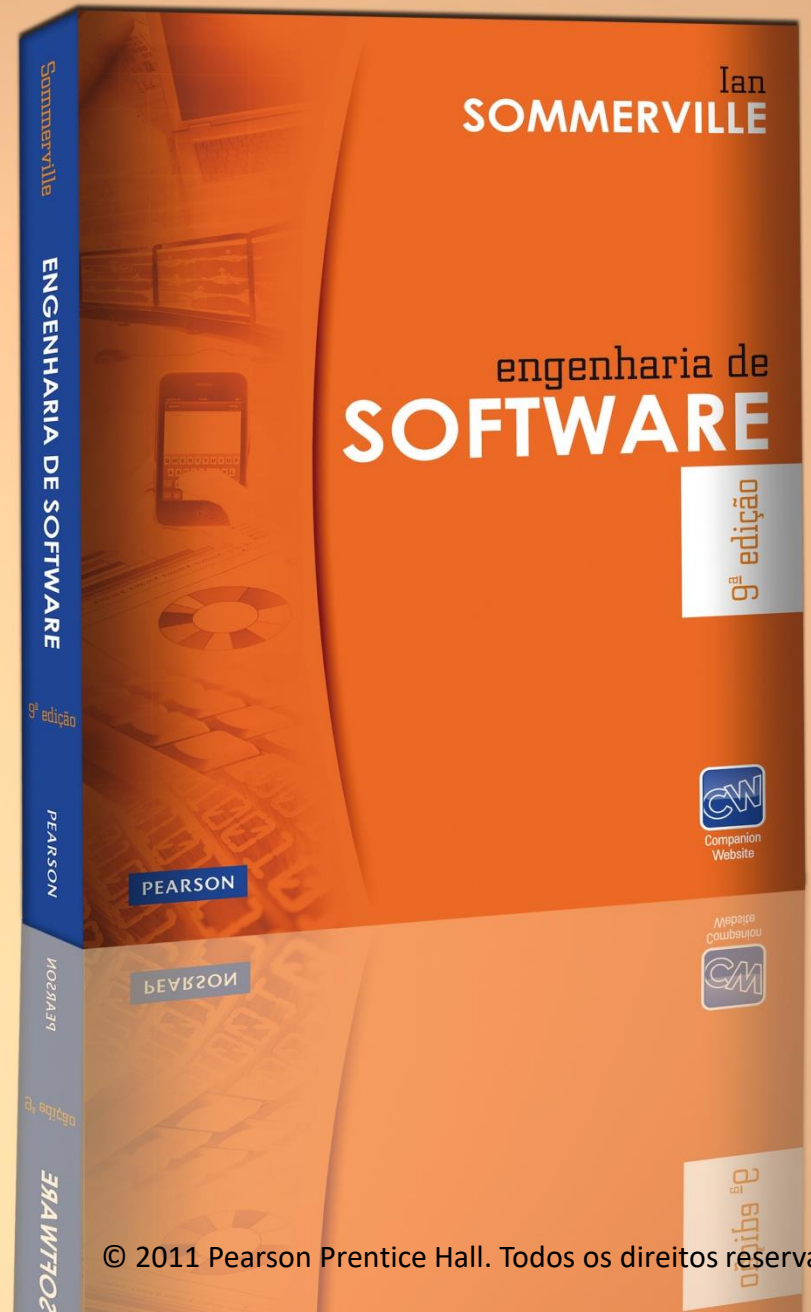


# Capítulo 5

## Modelagem de Sistemas



## Tópicos Apresentados

- Modelos de contexto
- Modelos de interação
- Modelos estruturais
- Modelos comportamentais
- Engenharia dirigida a modelos

- A modelagem de sistemas é o processo de desenvolvimento de modelos abstratos de um sistema, de maneira que cada modelo apresenta uma visão ou perspectiva diferente do sistema.
- Atualmente, a modelagem de sistemas se tornou a representação de um sistema usando algum tipo de notação gráfica, que hoje em dia quase sempre são baseadas em notações em Unified Modeling Language (UML).
- A modelagem de sistemas ajuda o analista a entender a funcionalidade do sistema e os modelos são usados para comunicação com os clientes.

## Modelos de sistemas existentes e planejados

# engenharia de SOFTWARE

- Os modelos de sistemas existentes são usados durante a engenharia de requisitos. Eles ajudam a esclarecer o que o sistema existente faz e podem ser usados como uma base para discussão dos seus pontos fortes e fracos. O que pode levar a requisitos para um novo sistema.
- Os modelos de um novo sistema são usados durante a engenharia de requisitos para ajudar a explicar os requisitos propostos para outros stakeholders do sistema.
- Os engenheiros usam esses modelos para discutir propostas de projeto e para documentar o sistema para implementação.
- Em um processo de engenharia dirigida a modelos, é possível gerar uma implementação do sistema completa ou parcial a partir do modelo do sistema.

- Uma perspectiva externa, na qual você modela o contexto ou ambiente do sistema.
- Um perspectiva de interação, em que você modela as interações entre um sistema e seu ambiente ou entre os componentes de um sistema.
- Uma perspectiva estrutural, na qual você modela a organização de um sistema ou a estrutura dos dados processados pelo sistema.
- Uma perspectiva comportamental, em que você modela o comportamento dinâmico do sistema e como ele responde a eventos.



## Tipos de diagramas UML

- Os diagramas de atividades, que mostram as atividades envolvidas em um processo ou no processamento de dados.
- Os diagramas de casos de uso, que mostram as interações entre um sistema e seu ambiente.
- Os diagramas de sequência, que mostram as interações entre os atores e o sistema e entre os componentes do sistema.
- Os diagramas de classe, que mostram as classes de objeto no sistema e as associações entre as classes.
- Os diagramas de estado, que mostram como o sistema reage a eventos internos e externos.

## Uso de modelos gráficos

- Como forma de facilitar a discussão sobre um sistema existente ou proposto
- Os modelos incompletos e incorretos podem ser usados já que servem para auxiliar a discussão.
- Como uma forma de documentar um sistema existente
- Os modelos devem ser uma representação precisa do sistema mas não precisam estar completos.
- Como uma descrição detalhada do sistema, a qual pode ser usada para gerar uma implementação do sistema
  - ✓ Os modelos precisam e estar corretos e completos.

## Modelos de contexto

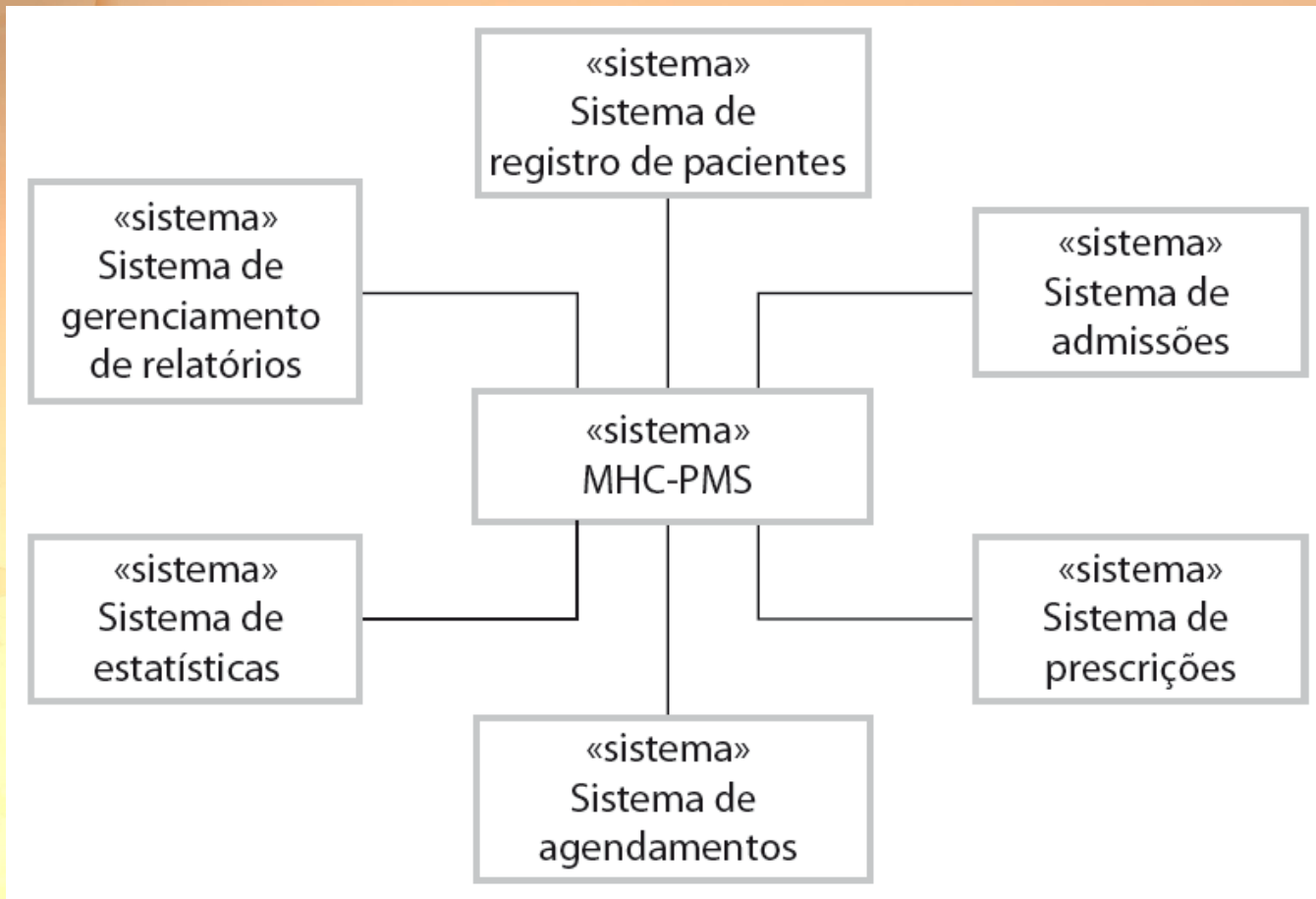
- Os modelos de contexto são usados para ilustrar o contexto operacional de um sistema – eles mostram o que existe fora dos limites do sistema.
- Interesses sociais e organizacionais podem afetar a decisão de onde posicionar os limites do sistema.
- Modelos de arquitetura mostram o sistema e seu relacionamento com outros sistemas.



## Limites do sistema

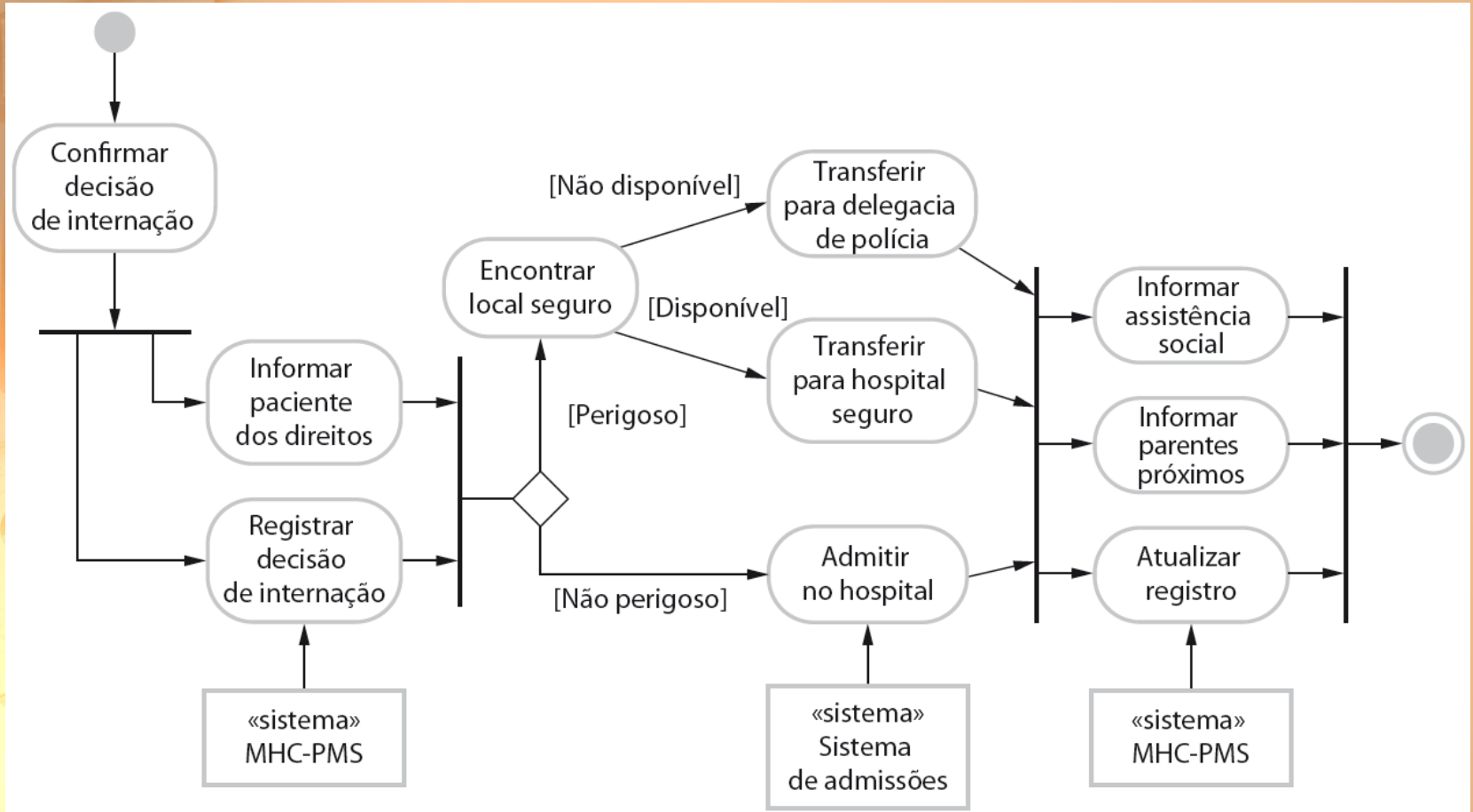
- Os limites do sistema são estabelecidos para definir o que está dentro e o que está fora do sistema.
  - ✓ Eles mostram outros sistemas que são usados ou dependem do sistema que está sendo desenvolvido.
- A posição do limite do sistema tem um efeito profundo nos requisitos do sistema.
- A definição de um limite de sistema é uma decisão política
  - ✓ Podem haver pressões para desenvolver limites de sistema que aumentem/diminuam a influência ou carga de trabalho de diferentes partes de uma organização.

## O contexto do MHC-PMS



- Os modelos de contexto mostram apenas os outros sistemas no ambiente, e não como o sistema que está sendo desenvolvido é usado nesse ambiente.
- Os modelos de processo revelam como o sistema em desenvolvimento está sendo desenvolvido e como esse é utilizado em processos de negócio mais amplos.
- Os diagramas de atividades da UML podem ser usados para definir os modelos de processo do negócio.

# Modelo de processo de internação involuntária



## Modelos de interação

- A modelagem da interação do usuário é importante por auxiliar a identificar os requisitos do usuário.
- Modelar a interação sistema-a-sistema destaca os problemas de comunicação que possam vir a surgir.
- Modelar a interação de componentes auxilia a entender se uma estrutura proposta de sistema pode fornecer um desempenho e uma confiança requeridas do sistema.
- Os diagramas de caso de uso e os diagramas de sequência podem ser usados para modelagem de interação.

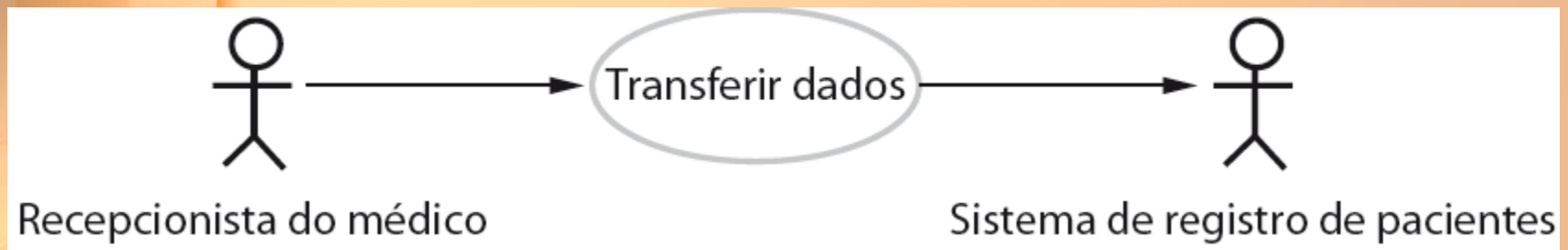


## Modelagem de caso de uso

- Os casos de uso foram desenvolvidos originalmente para auxiliar na elicitação de requisitos e atualmente, estão incorporados na UML.
- Cada caso de uso representa uma tarefa distinta que envolve interação externa com um sistema.
- Os atores em um caso de uso podem ser pessoas ou outros sistemas.
- São representados esquematicamente para fornecer uma visão geral do caso de uso e em uma forma textual mais detalhada.

## Caso de uso de transferência de dados

# engenharia de SOFTWARE

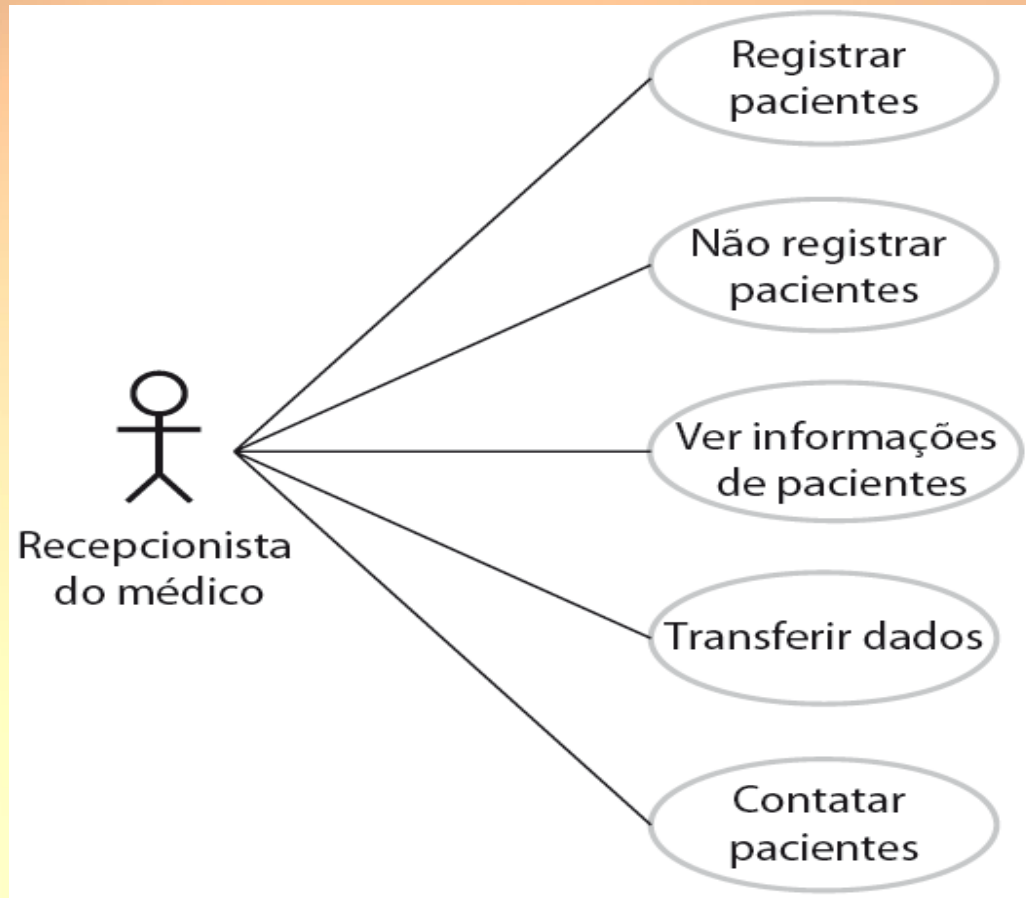


## Descrição tabular do caso de uso 'Transferir dados'

Atores	Recepcionista do médico, sistema de registros de pacientes (PRS, do inglês <i>patient records system</i> )
Descrição	Uma recepcionista pode transferir dados do MHC-PMS para um banco de dados geral de registros de pacientes mantido por uma autoridade de saúde. As informações transferidas podem ser atualizadas com as informações pessoais (endereço, telefone etc.) ou com um resumo do diagnóstico e tratamento do paciente.
Dados	Informações pessoais do paciente, resumo do tratamento.
Estímulos	Comando de usuário emitido pela recepcionista do médico.
Resposta	Confirmação de que o PRS foi atualizado.
Comentários	A recepcionista deve ter permissões de proteção adequadas para acessar as informações do paciente e o PRS.

## Casos de uso envolvendo o papel da 'Recepcionista do médico'

# engenharia de SOFTWARE

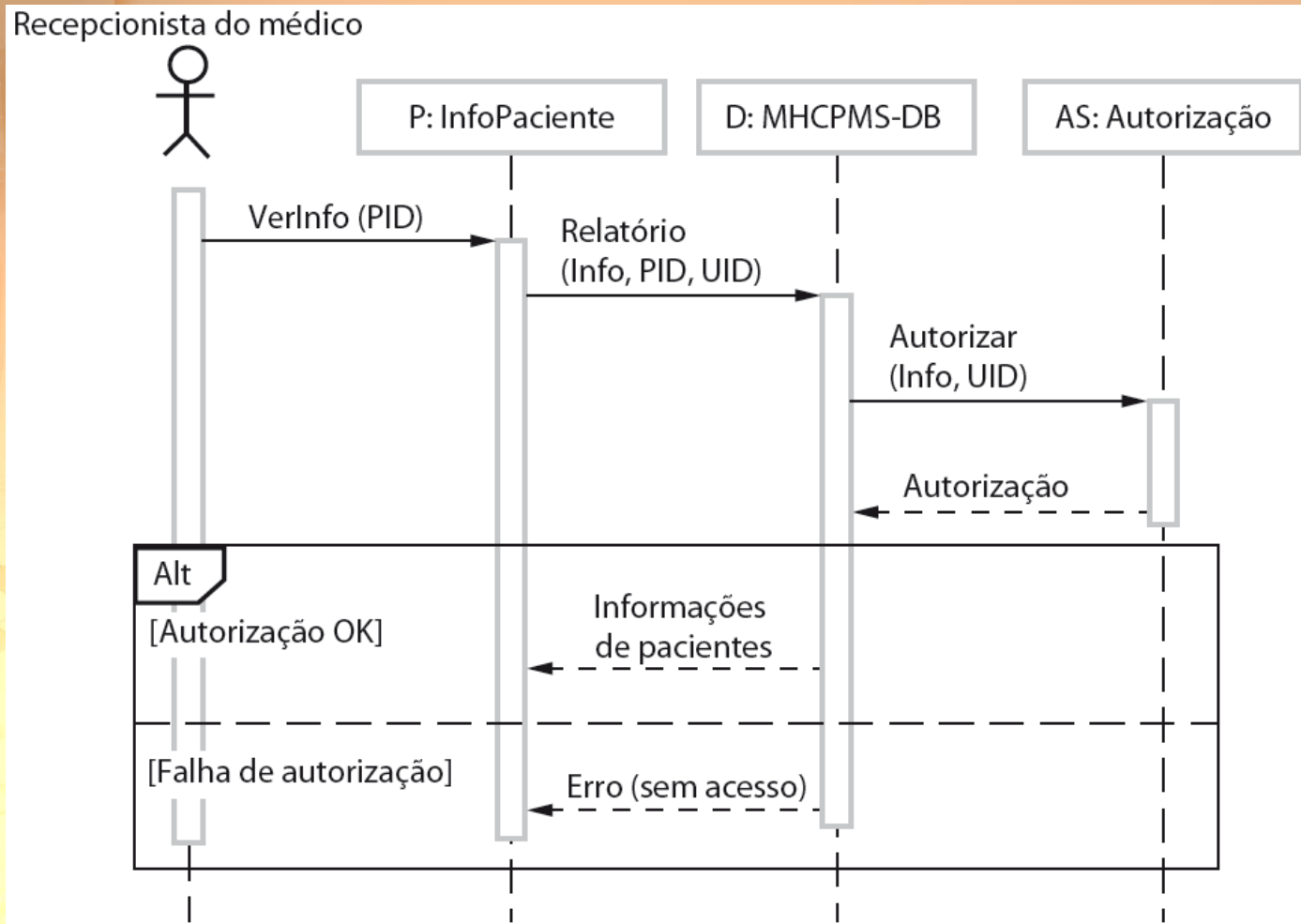


## Diagramas de sequência

- Os diagramas de sequência são parte da UML e são usados para modelar as interações entre os atores e os objetos no sistema.
- Um diagrama de sequência mostra a sequência de interações que ocorrem durante um caso de uso em particular ou uma instância de caso uso.
- Os objetos e os atores envolvidos são listados na parte superior do diagrama com uma linha pontilhada desenhada verticalmente a partir deles.
- As interações entre os objetos são indicadas por setas anotadas.

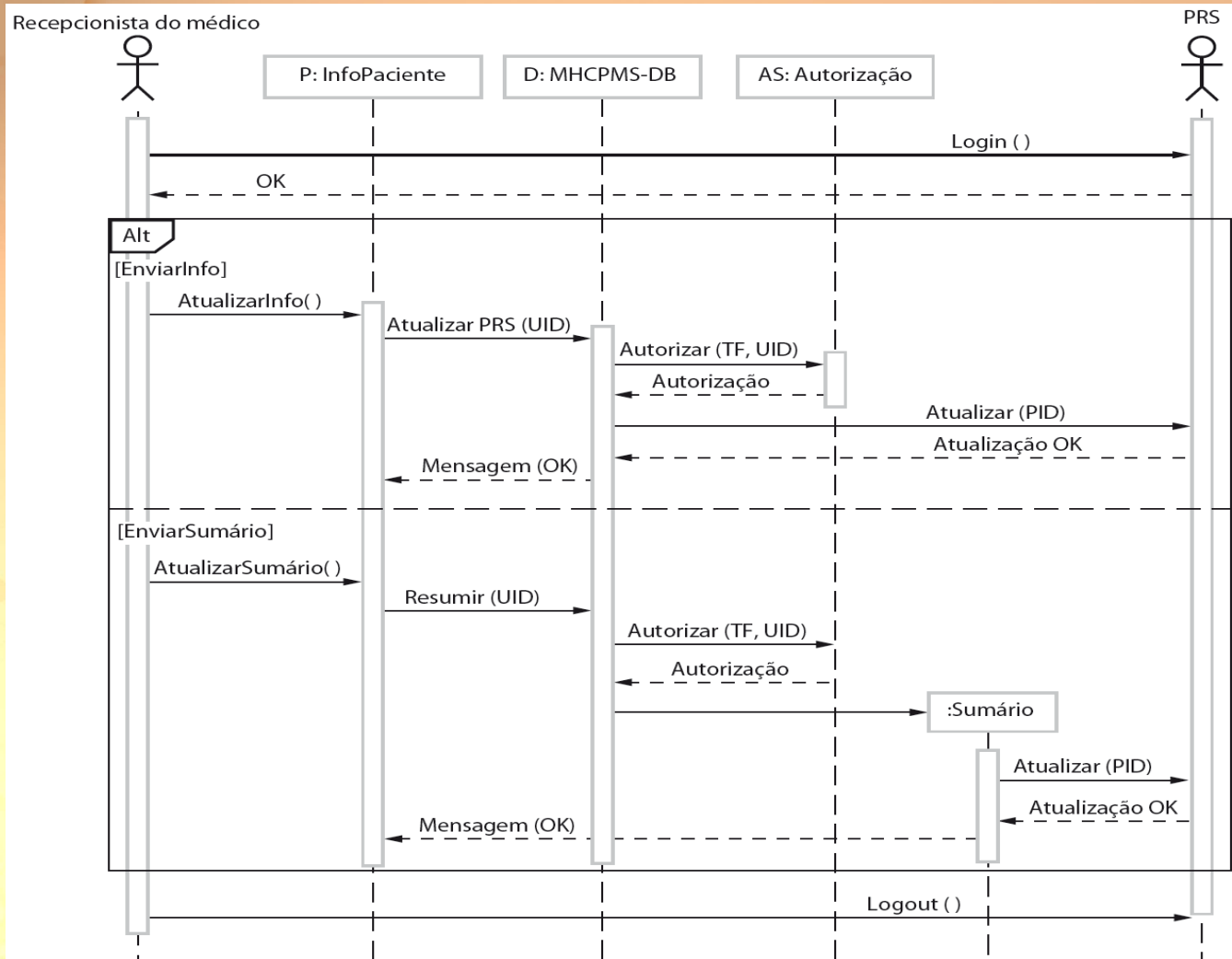


# Diagrama de sequência para 'Ver informações de pacientes'



# Diagrama de sequência para 'Transferir dados'

# engenharia de SOFTWARE



- Modelos estruturais de software mostram a organização de um sistema em termos de seus componentes que compõem este sistema e seus relacionamentos.
- Os modelos estruturais podem ser modelos estáticos, que mostram a estrutura do projeto do sistema, ou modelos dinâmicos, que mostram a organização do sistema quando ele está em execução.
- Você cria modelos estruturais de um sistema quando se está discutindo e projetando a arquitetura do sistema.

## Diagramas de classe

- Os diagramas de classe são usados quando se está desenvolvendo um modelo de sistema orientado a objetos para mostrar as classes em um sistema e as associações entre as classes.
- Uma classe de objeto pode ser considerada como uma definição geral de um tipo de objeto do sistema.
- Uma associação é uma ligação entre classes que indica que há algum relacionamento entre essas classes.
- Quando se está desenvolvendo os modelos durante os primeiros estágios do processo de engenharia de software, os objetos representam alguma coisa no mundo real como um paciente, uma receita médica, um médico, etc.

## Classes e associação em UML

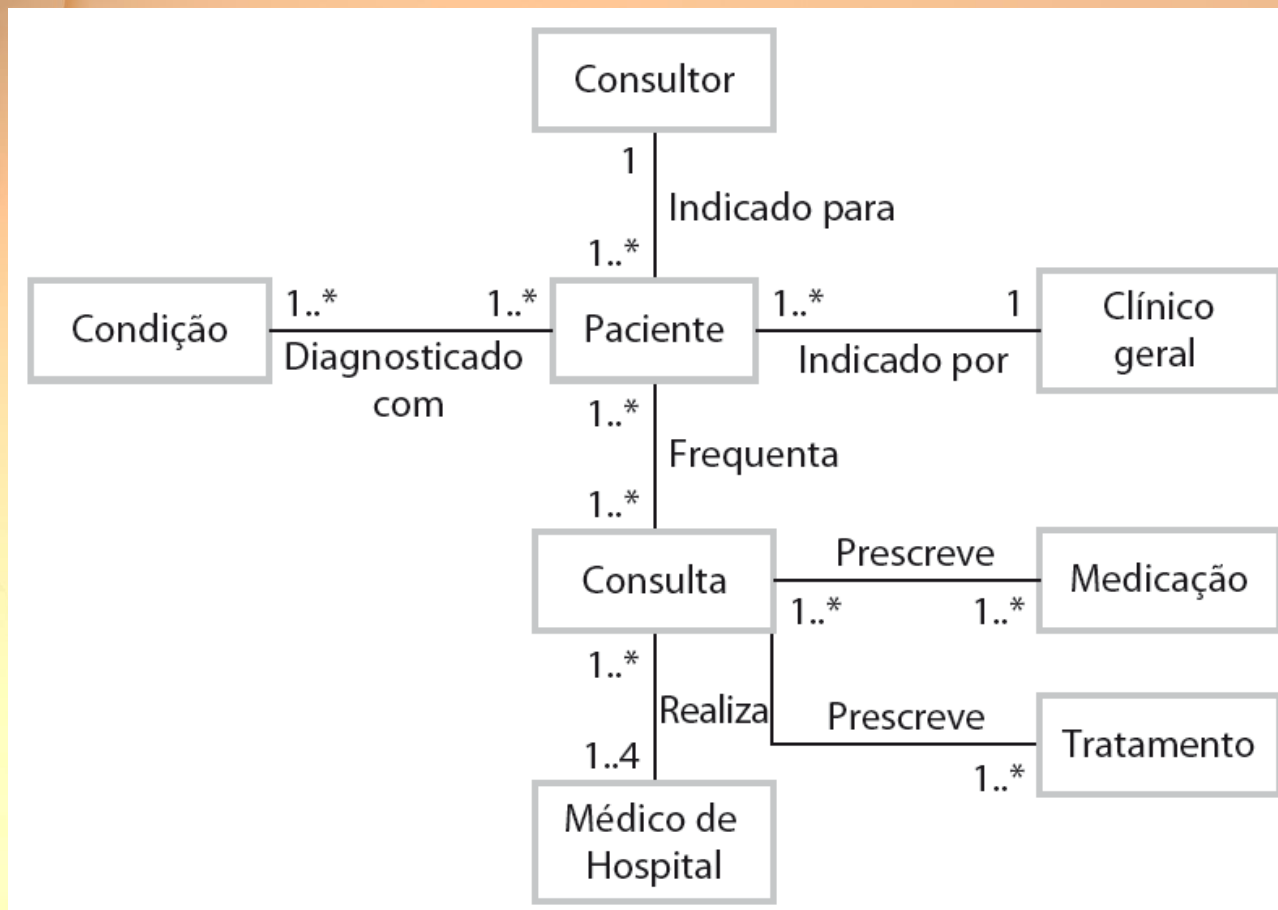
# engenharia de SOFTWARE





## Classes e associações no MHC-PMS

# engenharia de SOFTWARE



## A classe Consulta

# engenharia de SOFTWARE

Consulta
Médicos
Data
Horário
Clínica
Motivo
Medicação prescrita
Tratamento prescrito
Anotações de voz
Transcrições
...
Novo ( )
Prescrever ( )
RegistrarAnotações ( )
Transcrever ( )
...

## Pontos Importantes

- Um modelo é uma visão abstrata de um sistema que ignora alguns detalhes do sistema.
- Modelos de sistema complementares podem ser desenvolvidos para mostrar o contexto, interações, estrutura e comportamento do sistema.
- Os modelos de contexto mostram como um sistema que está sendo modelado está posicionado em um ambiente com outros sistemas e processos.
- Os diagramas de caso de uso e os diagramas de sequência são usados para descrever as interações entre o usuário do sistema que está sendo projetado e usuários ou outros sistemas.

## Pontos Importantes

- Os casos de uso descrevem as interações entre um sistema e atores externos; diagramas de sequência adicionam mais informações a esses mostrando as interações entre os objetos do sistema.
- Os modelos estruturais mostram a organização e arquitetura de um sistema.
- Os diagramas de classe são usados para definir a estrutura estática de classes em um sistema e suas associações.

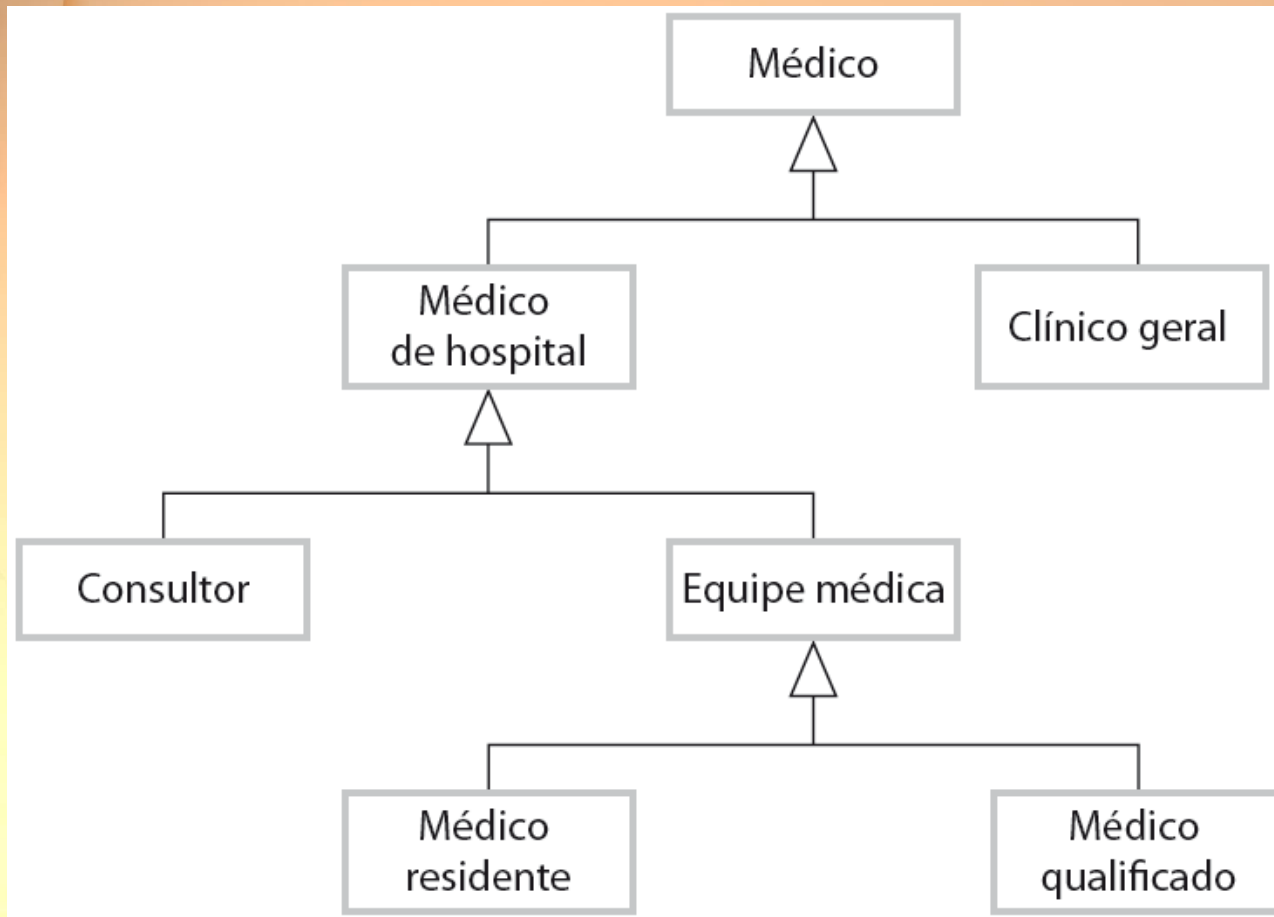
- A generalização é uma técnica de uso cotidiano para gerenciar complexidade.
- Ao invés de aprender as características detalhadas de todas as entidades envolvidas, colocamos essas entidades em classes mais gerais (animais, carros, casas, etc.) e aprendemos as características dessas classes.
- O que nos permite supor que membros diferentes dessas classes possuem algumas características comuns como, por exemplo, esquilos e ratos são roedores.



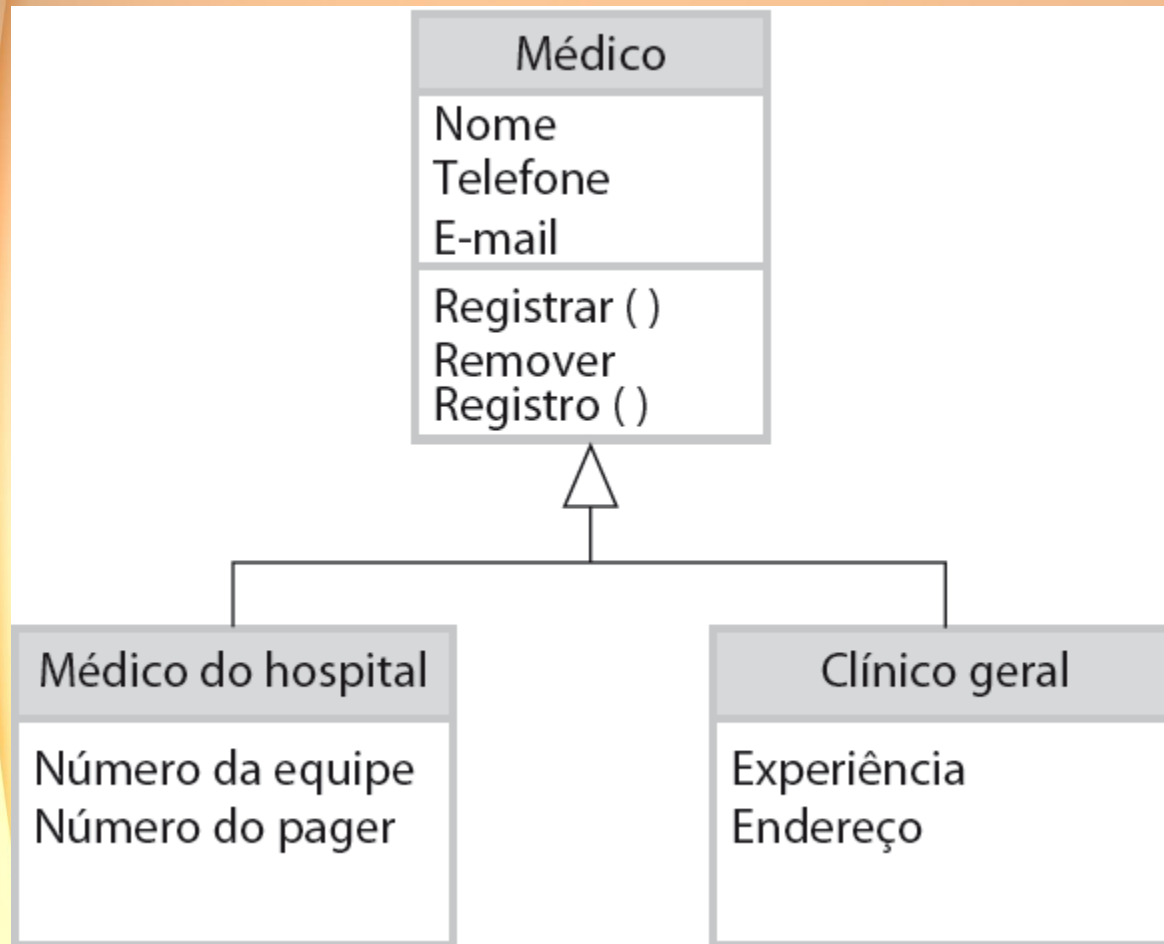
- Na modelagem de sistemas é geralmente útil examinar as classes de um sistema para ver se há escopo para generalização. Se mudanças forem propostas você não precisará examinar todas as classes no sistema para ver se são afetadas pela mudança.
- Em linguagens orientadas a objeto, como o Java, generalização é implementada usando os mecanismos de herança de classe que já existem na linguagem.
- Em uma generalização, os atributos e operações associados a classes mais altas também são associados com as classes de nível mais baixo.
- As classes de nível mais baixo são subclasses e herdam os atributos e operações de suas superclasses. Essas classes mais baixas adicionam atributos e operações mais específicas.

## Uma hierarquia de generalização

# engenharia de SOFTWARE



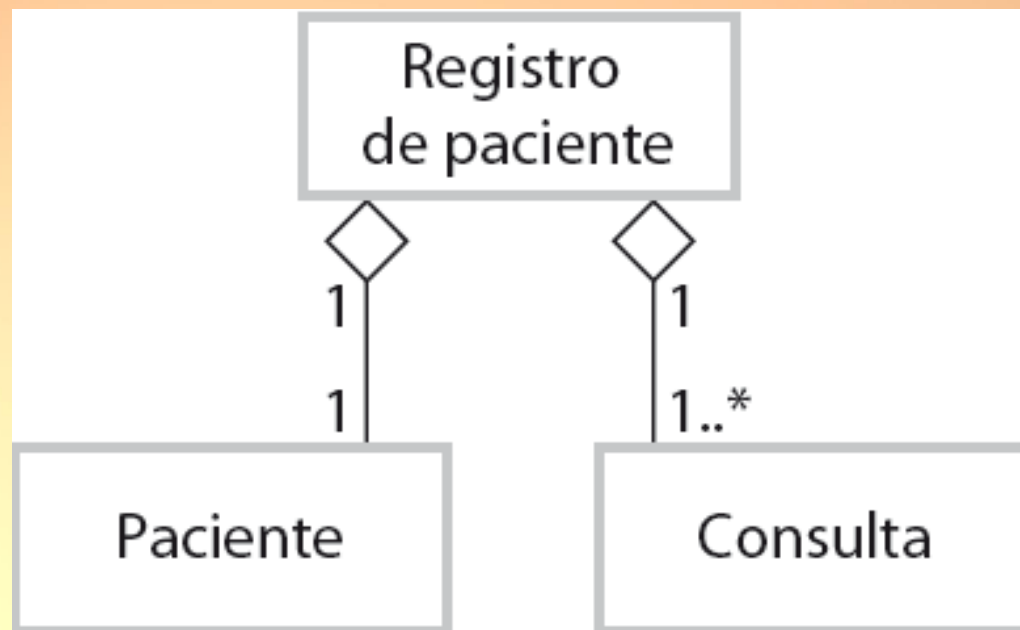
## Uma hierarquia de generalização com detalhes adicionais



# Modelos de agregação de classes de objetos

- Um modelo de agregação mostra como as classes que são coleções são compostas de outras classes.
- Os modelos de agregação são similares aos relacionamentos parte-de em modelos de dados semânticos.

## A associação por agregação



## Modelos comportamentais

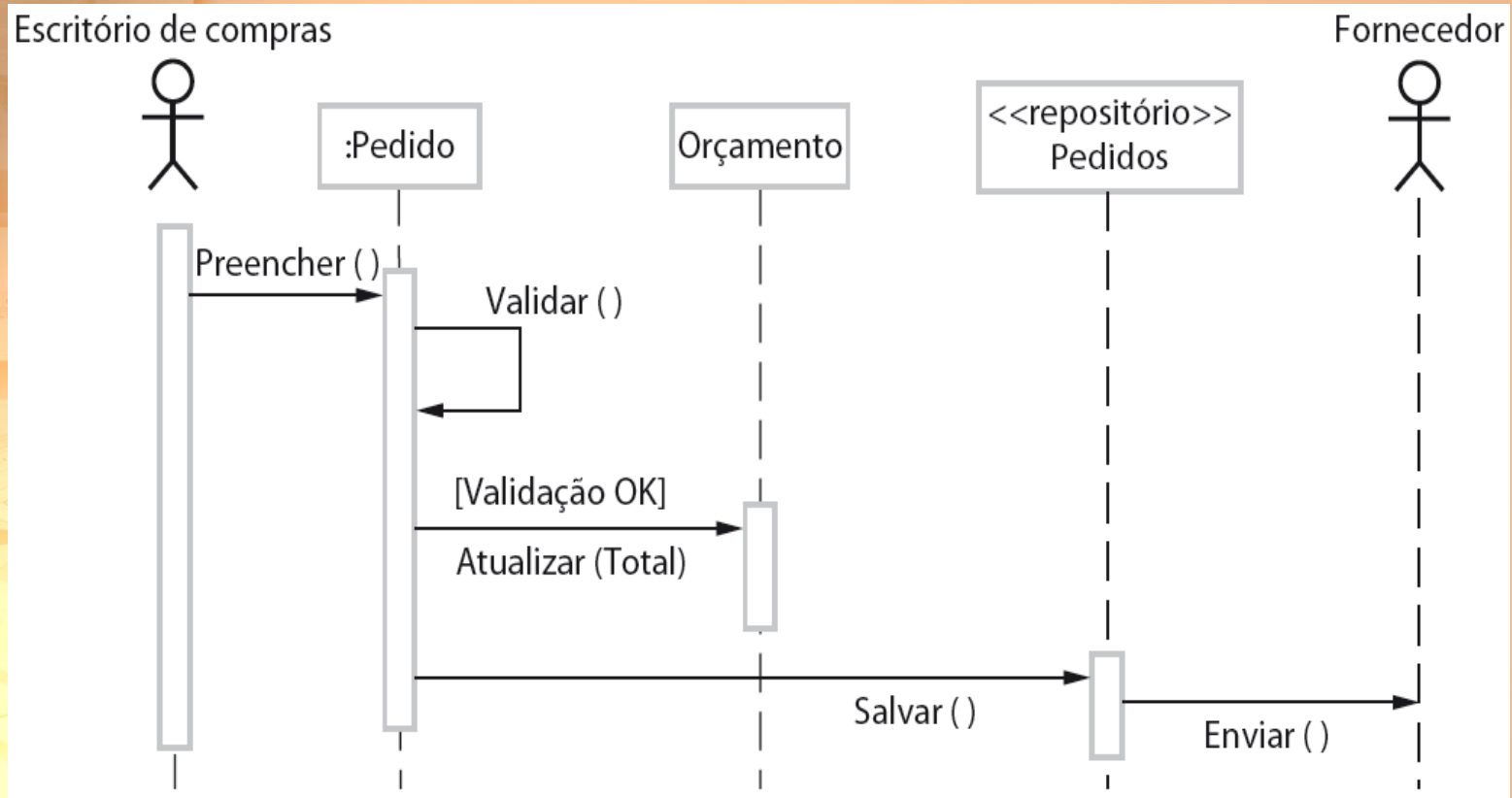
- Os modelos comportamentais são modelos do comportamento dinâmico do sistema enquanto esse está em execução. Eles mostram o que acontece ou o que deveria acontecer quando um sistema responde a um estímulo de seu ambiente.
- Podem-se considerar esses estímulos como sendo de dois tipos:
  - ✓ Dados: Alguns dados que chegam e precisam ser processados pelo sistema.
  - ✓ Eventos: Alguns eventos que acontecem disparam o processamento do sistema. Os eventos podem ter dados associados, embora isso não ocorra sempre.



## **Modelagem dirigida a dados**

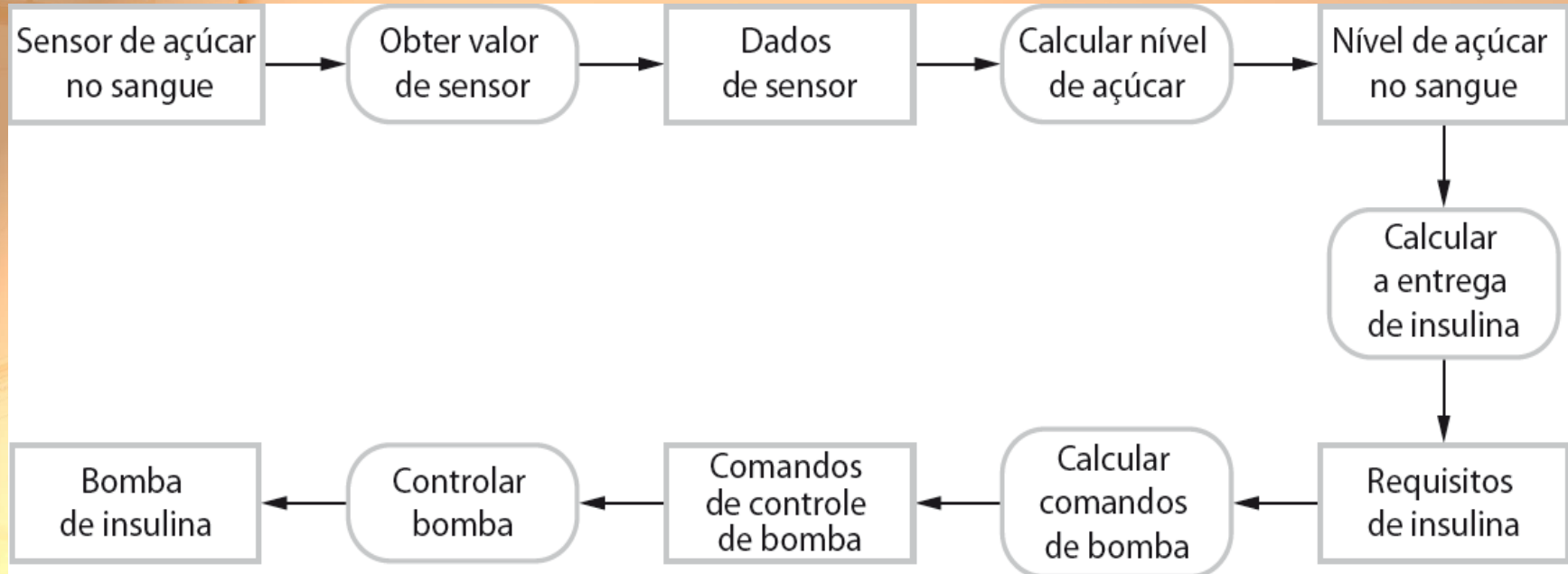
- Vários sistemas de negócio são sistemas de processamento de dados que são dirigidos principalmente a dados.
- Eles são controlados pela entrada de dados no sistema com relativamente quase nenhum processamento de eventos externos.
- Os modelos dirigidos a dados mostram a sequência de ações envolvidas no processamento dos dados entrantes e na geração de uma saída associada.
- Eles são particularmente úteis durante a análise de requisitos por poderem ser usados para mostrar o processamento fim-a-fim em um sistema.

# Um modelo de atividades de funcionamento da bomba de insulina



## Processamento de pedidos

# engenharia de SOFTWARE



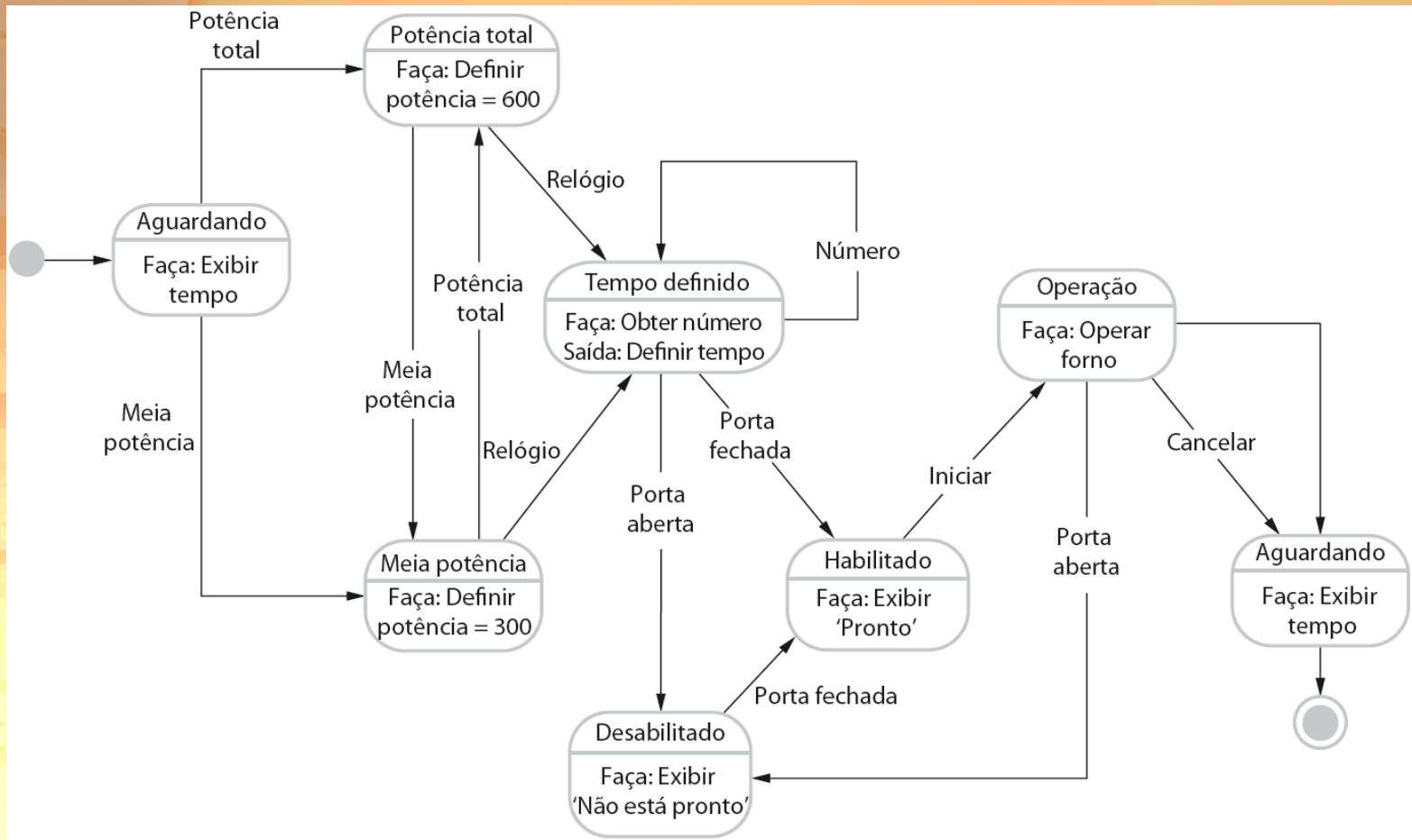
## Modelagem dirigida a eventos

- Geralmente , os sistemas em tempo real são dirigidos a eventos, com o mínimo de processamento de dados.
- Por exemplo, um sistema de comutação de linhas telefônicas responde a eventos como 'recebimento de chamada' gerando um tom de discagem.
- A modelagem dirigida a eventos mostra como um sistema responde a eventos externos e internos.
- Essa se baseia na suposição de que um sistema tem um número limitado de estados e que os eventos (estímulos) podem causar a transição de um estado para outro.

## Modelos de máquina de estados

- Esses modelam o comportamento do sistema em resposta a eventos externos e internos.
- Eles mostram as respostas do sistema a estímulos, e por isso, geralmente são usados para modelar sistemas de tempo real.
- Os modelos de máquina de estados mostram os estados do sistema, tais como nós, e os eventos desses, tais como arcos entre esses nós. Quando um evento ocorre, o sistema move-se de um estado para outro.
- Os gráficos de estado são uma parte integral da UML e são usados para representar os modelos de máquina de estados.

# Diagrama de estados de um forno de micro-ondas





## Estados e estímulos para o forno de micro-ondas (a)

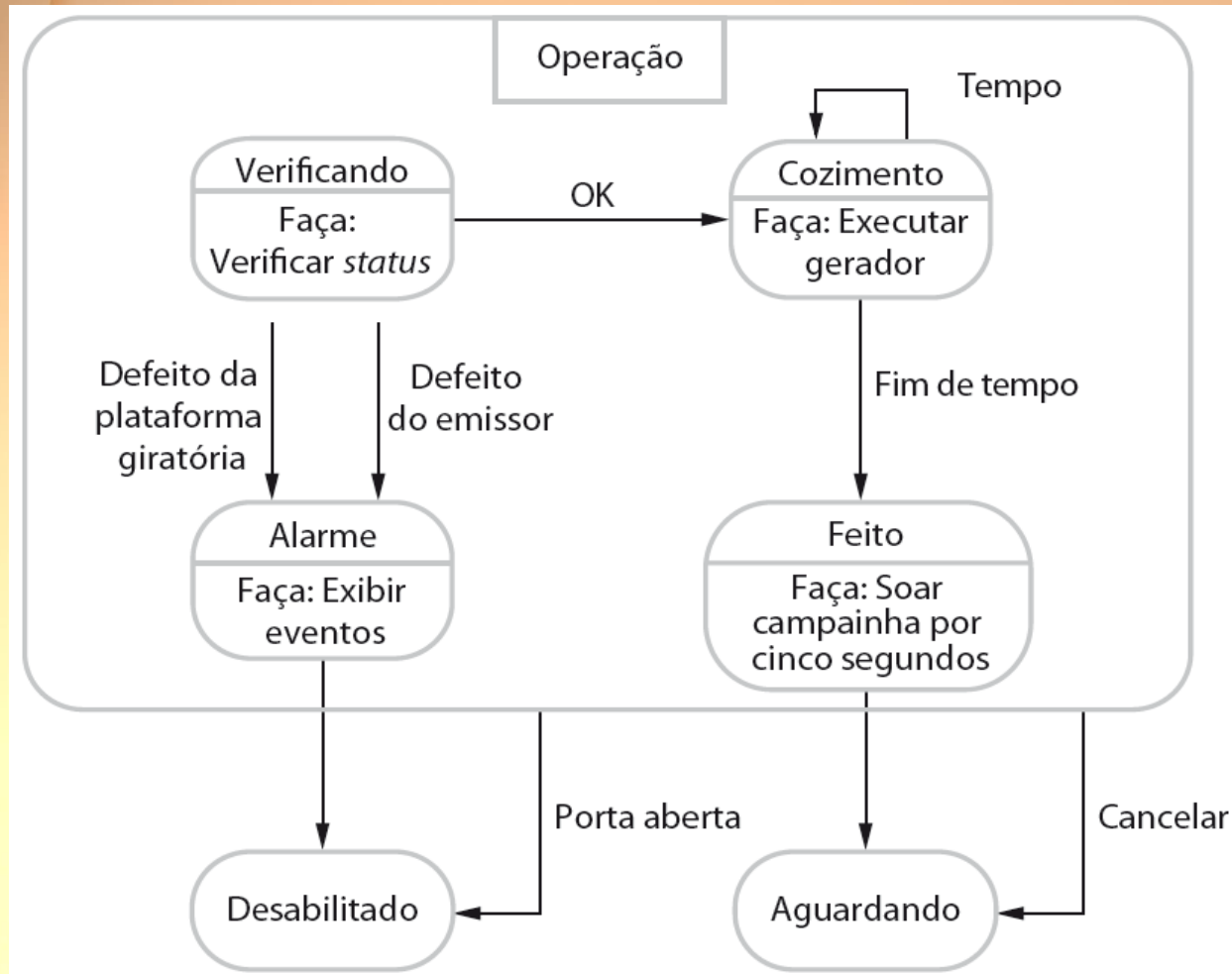
Estado	Descrição
Aguardando	O forno está aguardando uma entrada. O <i>display</i> mostra a hora atual.
Meia potência	A potência do forno é definida para 300 watts. O <i>display</i> mostra 'Meia potência'.
Potência total	A potência do forno é definida para 600 watts. O <i>display</i> mostra 'Potência total'.
Tempo definido	O tempo de cozimento é definido como valor de entrada do usuário. O <i>display</i> mostra o tempo de cozimento selecionado e é atualizado conforme o tempo definido.
Desabilitado	A operação do forno está desabilitada por questões de segurança. A iluminação interna do forno está acesa. O <i>display</i> mostra 'Não está pronto'.
Habilitado	A operação do forno está habilitada. A iluminação interna do forno está desligada. O <i>display</i> mostra 'Pronto'.
Operação	Forno em operação. A iluminação interna está acesa. O <i>display</i> mostra a contagem regressiva do relógio. No fim do cozimento, a campainha soa por cinco segundos. A luz do forno está acesa. O <i>display</i> mostra 'Cozimento completo', enquanto a campainha está soando.

## Estados e estímulos para o forno de micro-ondas (b)

Estímulos	Descrição
Meia potência	O usuário pressionou o botão de meia potência.
Potência total	O usuário pressionou o botão de potência total.
Relógio	O usuário pressionou um dos botões do relógio.
Número	O usuário pressionou uma tecla numérica.
Porta aberta	O interruptor da porta do forno não está fechado.
Porta fechada	O interruptor da porta do forno está fechado.
Iniciar	O usuário pressionou o botão Iniciar.
Cancelar	O usuário pressionou o botão Cancelar.

# Operação do forno de micro-ondas

# engenharia de SOFTWARE



- A engenharia dirigida a modelos (MDE – Model-Driven Engineering) é uma abordagem do desenvolvimento de software, na qual os modelos e não os programas são os principais produtos do processo de desenvolvimento.
- Os programas executados em uma plataforma de hardware/software são gerados automaticamente, a partir dos modelos.
- Os proponentes da MDE argumentam que isso aumenta o nível de abstração na engenharia de software, o que faz com que os engenheiros não precisem mais se preocupar com detalhes de linguagem de programação ou com as especificidades das plataformas de execução.

- A engenharia dirigida a modelos ainda está em um estágio de desenvolvimento inicial e ainda não é certo se essa terá um efeito significativo na prática da engenharia de software.
- Prós
  - ✓ Permite tratar dos sistemas em níveis mais altos de abstração
  - ✓ A geração automática do código significa menor custo de adaptação dos sistemas à novas plataformas.
- Contras
  - ✓ Os modelos são para abstração e não estão necessariamente corretos para implementação.
  - ✓ A economia criada pela geração de código pode ser contrabalanceada pelos custos de desenvolver tradutores para novas plataformas.

## Arquitetura dirigida a modelos

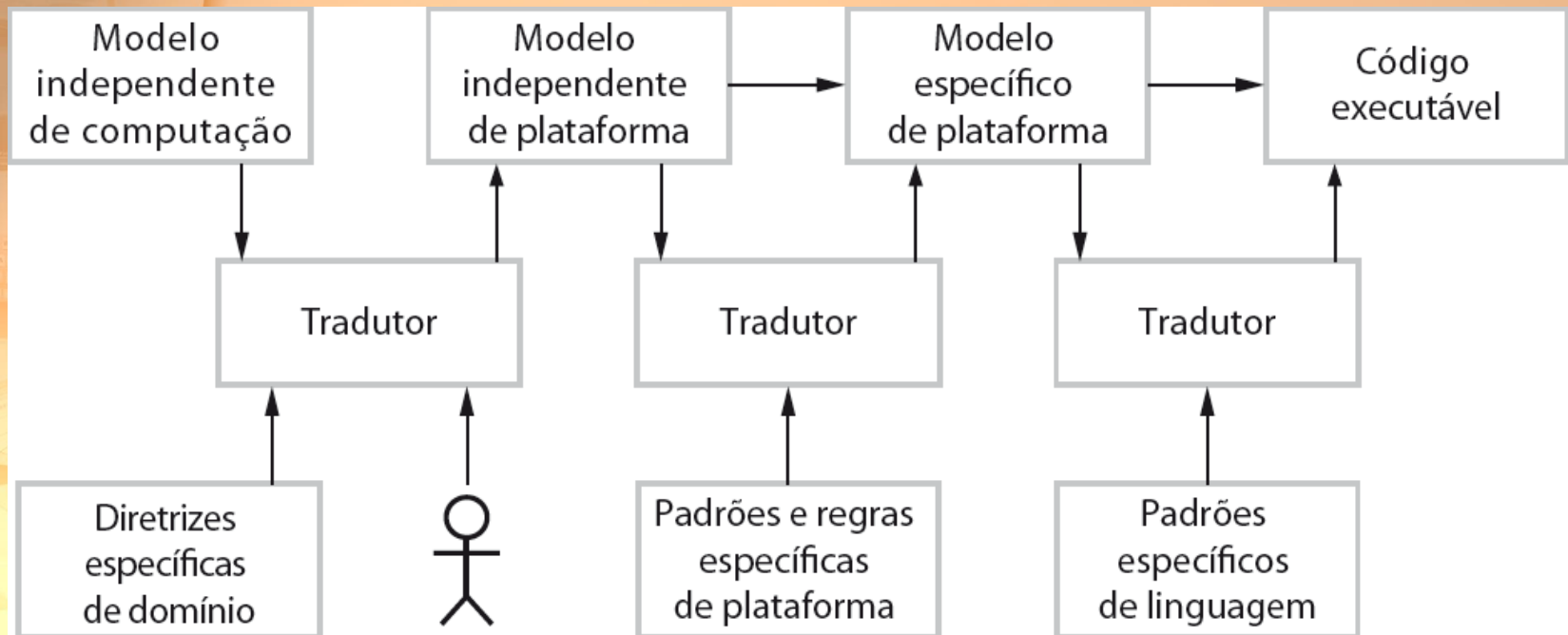
- A arquitetura dirigida a modelos (MDA – Model-Driven Architecture) foi a precursora da mais genérica engenharia dirigida a modelos.
- A MDA é uma abordagem de projeto e implementação de software centrada em modelos, que usa um subconjunto de modelos da UML para descrever o sistema.
- São criados modelos em diferentes níveis de abstração.
- Em princípio, a partir de um modelo independente de plataforma de alto nível, é possível gerar um programa funcional sem intervenção manual.



## Tipos de modelos

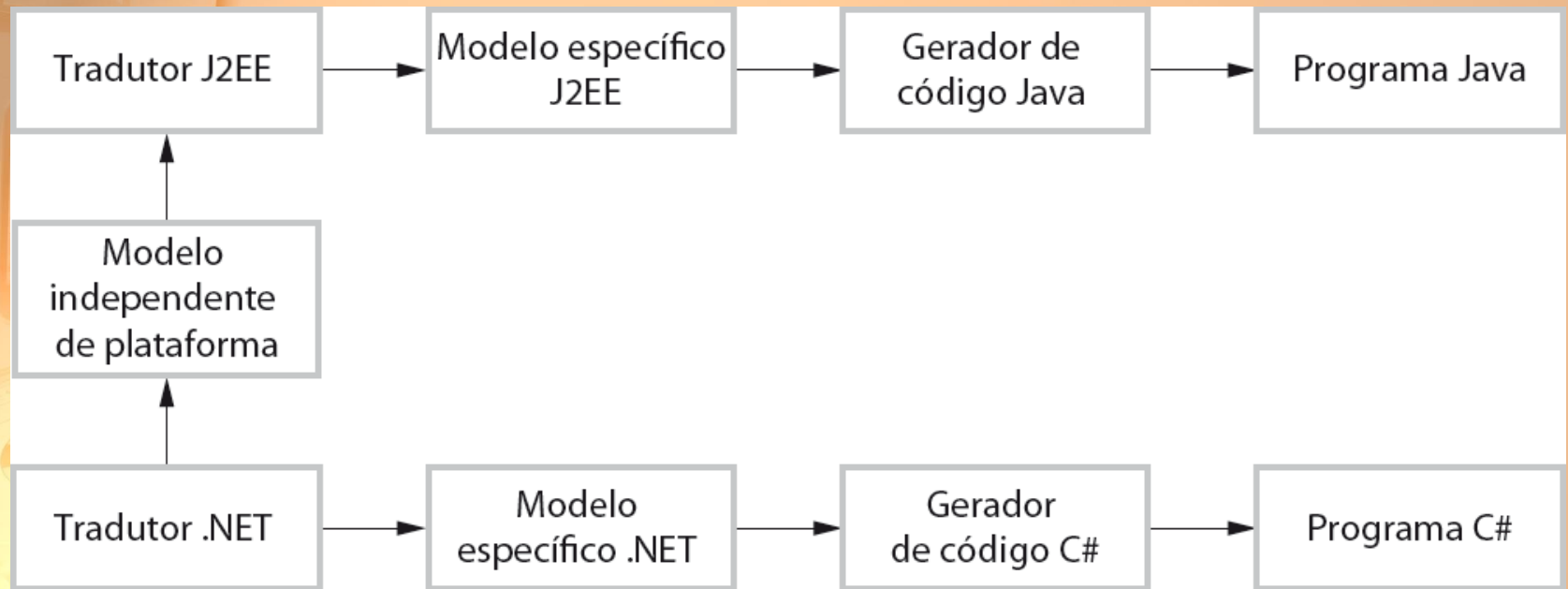
- Um modelo independente de computação (CIM – Computation Independent Model)
  - ✓ Esses modelam as importantes abstrações de domínio usadas em um sistema. Às vezes, os CIMs são chamados de modelos de domínio.
- Um modelo independente de plataforma (PIM – Platform Independent Model)
  - ✓ Esses modelam a operação do sistema sem referência para a sua implementação. Geralmente, o PIM é descrito pelo uso de modelos da UML que mostram a estrutura estática do sistema, e como esse responde a eventos externos e internos.

- Modelos específicos de plataforma (PSM – Platform Specific Models)
  - ✓ Esses são transformações do modelo independente de plataforma com um PSM separado para cada aplicativo da plataforma. A princípio, devem haver camadas de PSM, com cada camada adicionando algum detalhe específico à plataforma.



# Vários modelos específicos de plataforma

# engenharia de SOFTWARE



## Métodos ágeis e MDA

- Os desenvolvedores de MDA alegam que a intenção da MDA é auxiliar uma abordagem iterativa ao desenvolvimento e por isso, essa pode ser usada nos métodos ágeis.
- A noção da modelagem inicial extensiva contradiz as idéias fundamentais do manifesto ágil e eu suspeito que poucos desenvolvedores ágeis sentem-se confortáveis com a engenharia dirigida a modelos.
- Se as transformações puderem ser completamente automáticas e um programa completo gerado a partir de um PIM, então, em princípio, a MDA poderia ser usada em um processo de desenvolvimento ágil já que não seria necessária nenhuma codificação separada.

- A noção fundamental por trás da engenharia dirigida a modelos é que deve ser possível a transformação completamente automática dos modelos em códigos.
- O que é possível, se usarmos um subconjunto da UML 2 chamado UML Executável (Executable UML) ou xUML.



## Características da UML executável

- Para criar um subconjunto executável da UML, o número de tipos de modelos foi então reduzido dramaticamente para esses 3 tipos essenciais:
  - ✓ Os modelos de domínio que identificam os principais interesses em um sistema. Esses são definidos usando diagramas de classe da UML e incluem objetos, atributos e associações.
  - ✓ Os modelos de classe, nos quais as classes são definidas, junto com seus atributos e operações.
  - ✓ Os modelos de estado, nos quais um diagrama de estado é associado a cada classe e é usado para descrever o ciclo de vida da classe.
- O comportamento dinâmico do sistema pode ser especificado declarativamente usando a linguagem de restrição de objetos (OCL – Object Constraint Language) ou pode ser expressado usando a linguagem de ação da UML.

## Pontos Importantes

- Os modelos comportamentais são usados para descrever o comportamento dinâmico de um sistema em execução. Esse comportamento pode ser modelado da perspectiva dos dados processados pelo sistema, ou pelos eventos que estimulam respostas de um sistema.
- Os diagramas de atividades podem ser usados para modelar o processamento dos dados, no qual cada atividade representa um passo do processo.
- Os diagramas de estado são usados para modelar o comportamento de um sistema em resposta a eventos internos e externos.
- A engenharia dirigida a modelos é uma abordagem do desenvolvimento de software, na qual um sistema é representado como um conjunto de modelos que podem ser automaticamente transformados em códigos executáveis.