

# Especificação de Sistemas e SysML

Centro de Informática - Universidade Federal de Pernambuco  
Engenharia da Computação

Kiev Gama

[kiev@cin.ufpe.br](mailto:kiev@cin.ufpe.br)

*Slides elaborados pelos professores Marcio Cornélio e Kiev Gama*

O autor permite o uso e a modificação dos *slides* para fins didáticos



UNIVERSIDADE FEDERAL DE PERNAMBUCO

# Modelagem de sistema

- A modelagem de sistema auxilia o analista a **entender** a funcionalidade do sistema
  - Modelo => visão **simplificada** e **abstrata** de um sistema
  - Foco no que é **importante**
- Modelos são usados para melhorar a **comunicação** com os clientes.
- Modelos podem ser **executáveis!**
  - Exs.: Especificações formais e testes

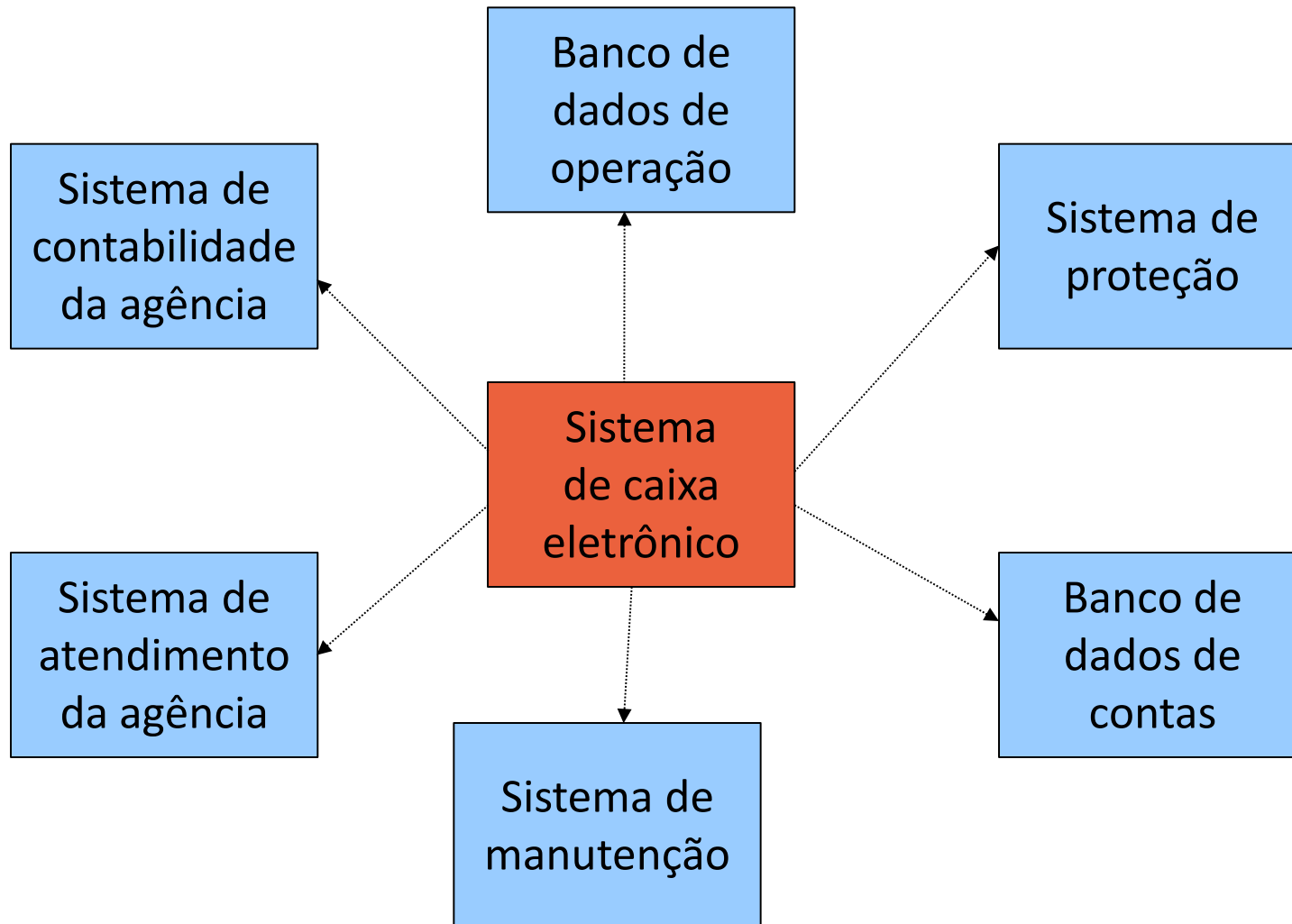
# Modelagem de sistema

- Modelos diferentes apresentam o sistema a partir de perspectivas diferentes. Ex:
  - Perspectiva externa
    - Contexto ou ambiente do sistema
  - Perspectiva de interação
    - Interações entre sistema e ambiente ou entre componentes do sistema
  - Perspectiva estrutural
    - Organização do sistema e a estrutura dos dados que ele processa
  - Perspectiva comportamental
    - Comportamento dinâmico do sistema e como ele responde a eventos

# Modelos de contexto

- Usados para ilustrar o **contexto operacional** de um sistema
- Mostram, em particular, elementos com os quais o sistema interagirá
- Pode ser difícil determinar os limites do sistema
  - Falta de compreensão
  - Requisitos bem definidos e validados podem mitigar este problema

# O contexto do sistema de um caixa eletrônico

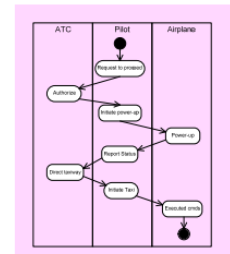


# Linguagens de Modelagem

- Demais tipos de modelos categorizados por Sommerville (Interação, Comportamental, Estrutural) são cobertos por linguagens de modelagem
- Notação padronizada
  - **UML** (Unified Modeling Language)
    - padrão de fato para modelagem de software
  - **SysML** (Systems Modeling Language)
    - derivação do UML voltada p/ sistemas

# Modelagem de sistema

- Modelos são mais úteis se escritos em uma Linguagem de Modelagem
- Engenharia de sistemas baseada em modelos
  - **Aplicação da modelagem para apoiar diversas atividades (requisitos, análise, projeto, verificação e valicação) da engenharia de sistemas**

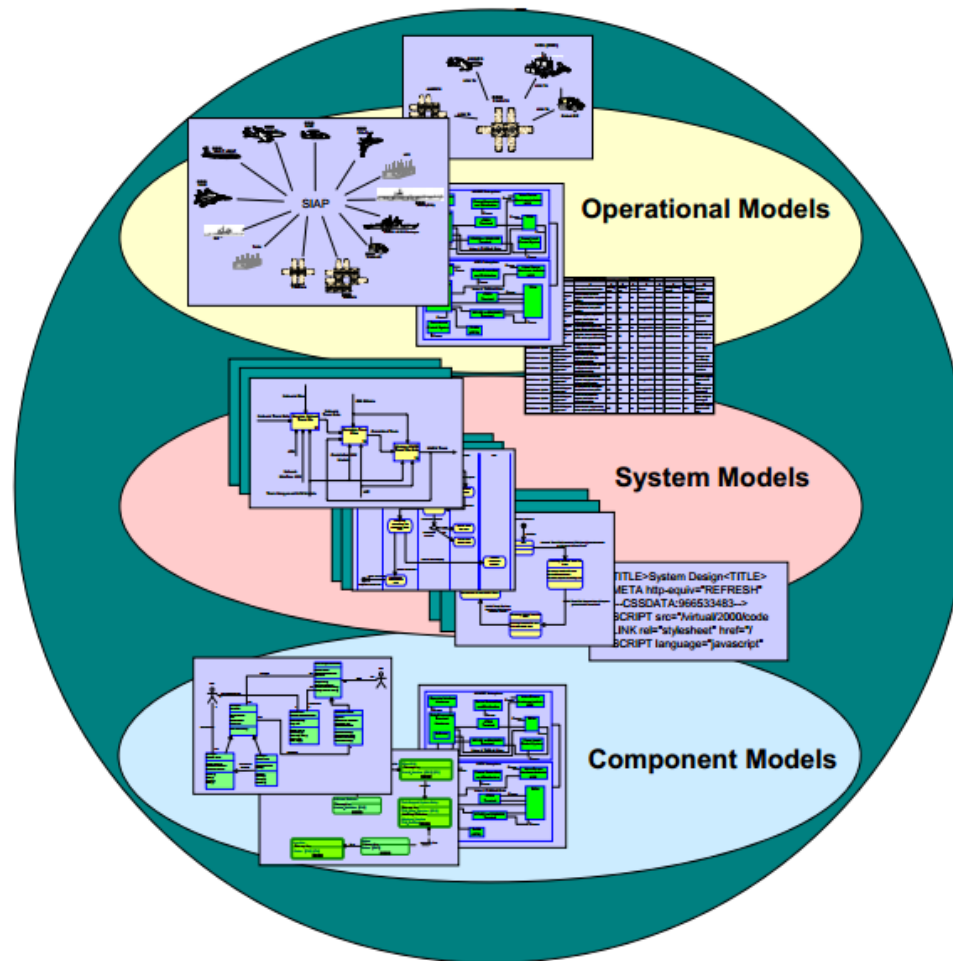


# Vantagens

- Auxilia no gerenciamento do desenvolvimento de sistemas complexos
- Conhecimento compartilhado entre equipe e stakeholders
- Reduz riscos permitindo validar requisitos e design
- Separação de preocupações através de várias visões de modelos integrados
- Permite rastreabilidade através de modelos hierárquicos
- Facilita análise de requisitos e mudanças no design do projeto
- Reduz ambiguidades
- Transferência de conhecimento através da captura do design de sistemas existentes



# Modelagem de vários níveis do sistema



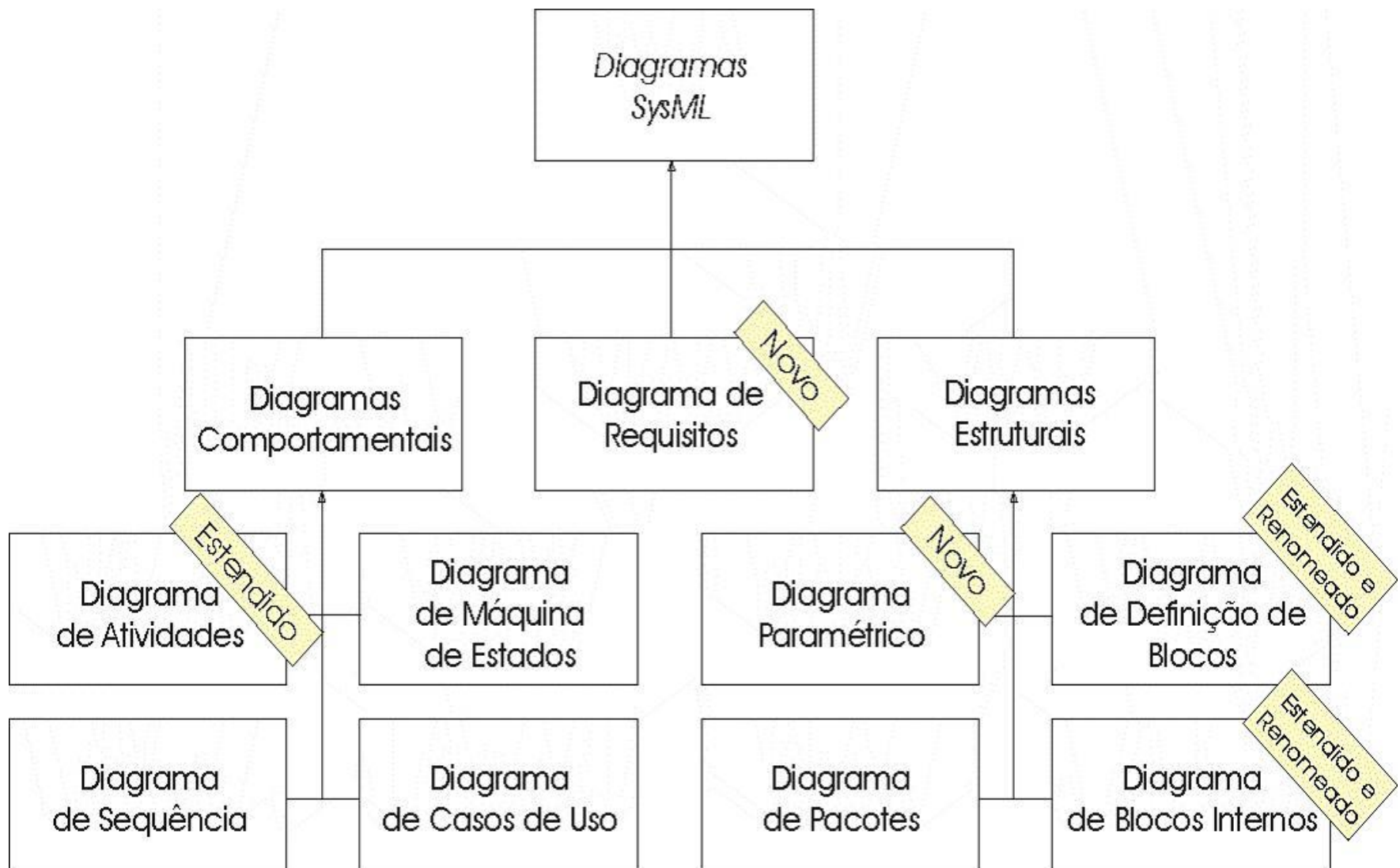
# SysML – Linguagem Unificada de Modelagem

Linguagem de modelagem de uso para aplicações de engenharia de sistemas.

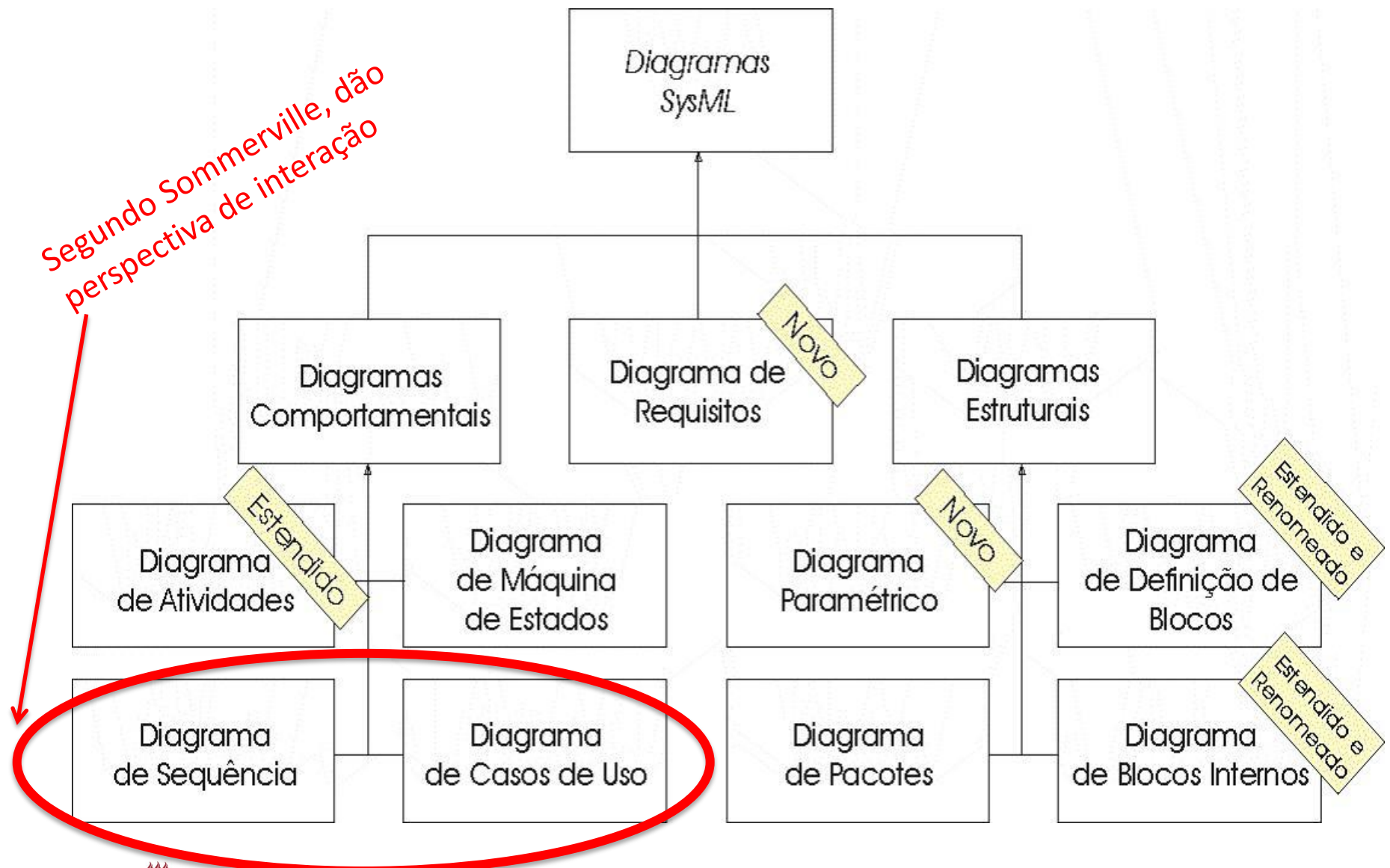
- Não explicita **como** a modelagem deve ser conduzida
- Permite a modelagem através de diferentes perspectivas

Não é uma metodologia nem uma ferramenta!

# Diagramas

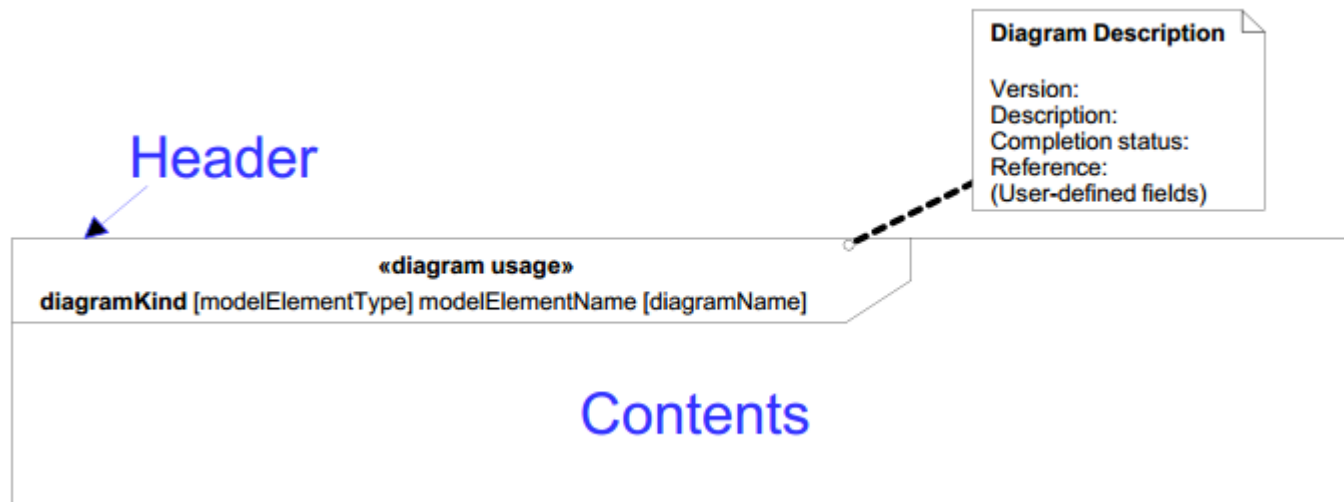


# Diagramas



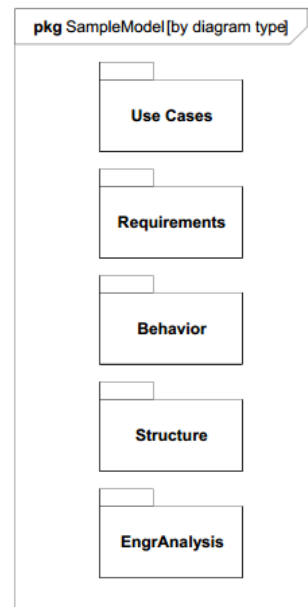
# Estrutura dos diagramas SysML

- Cada diagrama possui um frame
- O contexto do diagrama é exibido no cabeçalho
  - Tipo (act, bdd, ibd, sd)
  - Tipo do elemento
  - Nome do elemento
- Descrição do diagrama

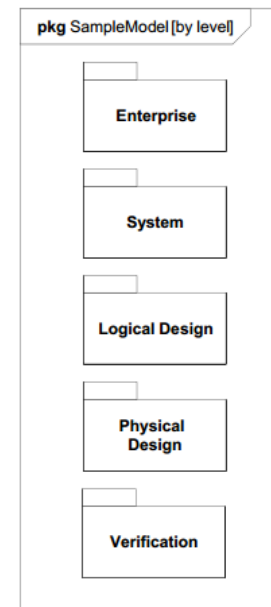


# Diagrama de Pacotes

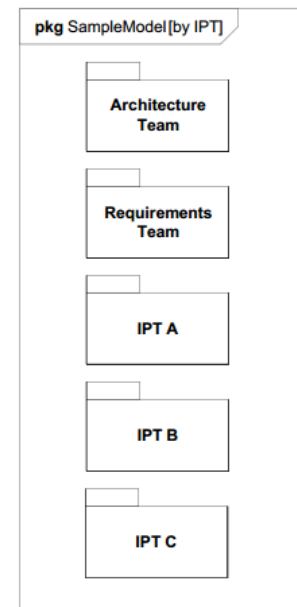
- Usado para organizar o modelo
  - Agrupa elementos em um namespace
- Modelos podem ser organizados de várias formas
  - Por hierarquia do sistema
  - Por tipo de diagrama



By Diagram Type

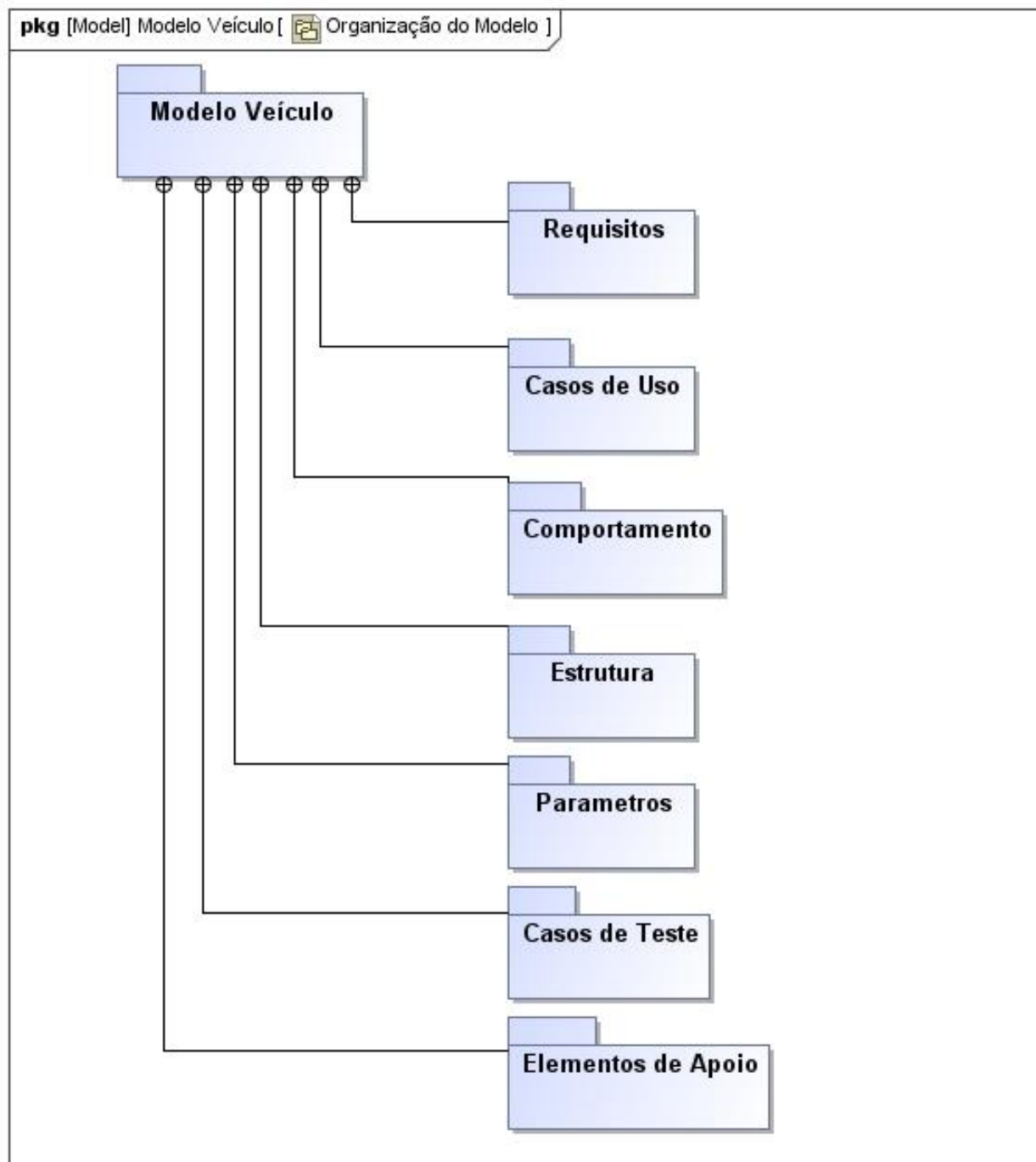


By Hierarchy



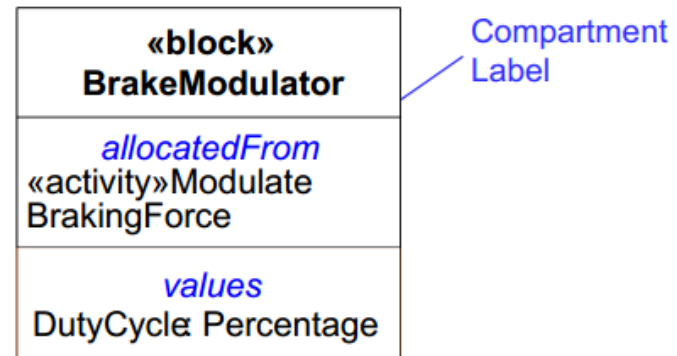
By IPT

# Pacotes



# Diagrama de definição de blocos

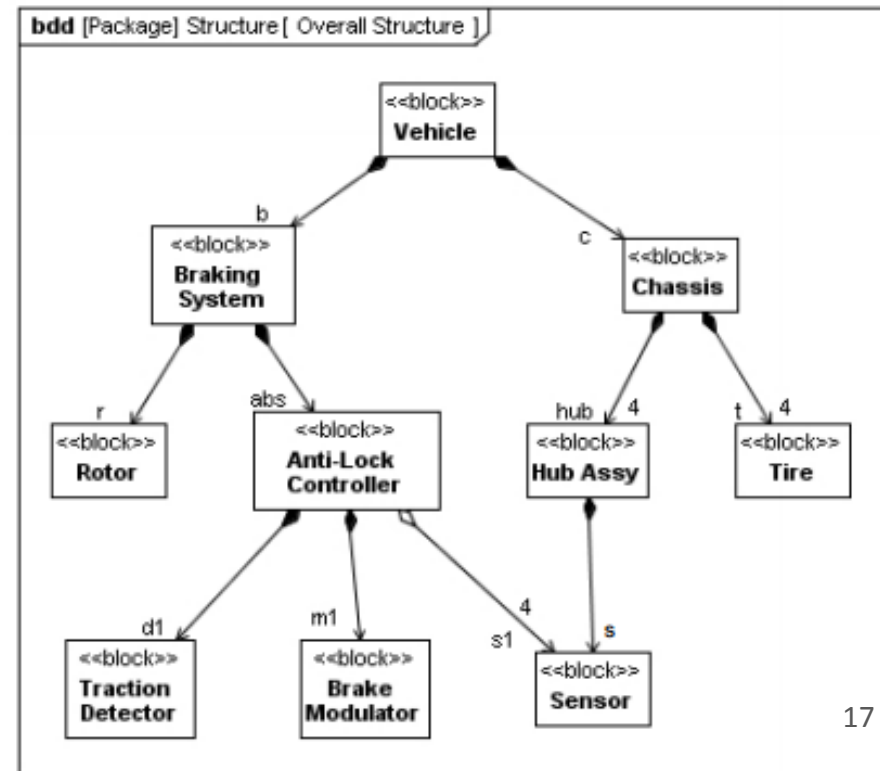
- Elementos estruturais básicos
- Conceito unificado para descrever a estrutura de um elemento do sistema
  - (Sub)sistema
  - Hardware
  - Software
  - Dado
  - Procedimento
  - Prédio
  - Pessoa
- Diferentes compartimentos para descrever características
  - Propriedades (parts, references, values, ports)
  - Operações
  - Constraints
  - Alocações de/para outros elementos do modelo
  - Requisitos
  - Compartimentos definidos pelo usuário





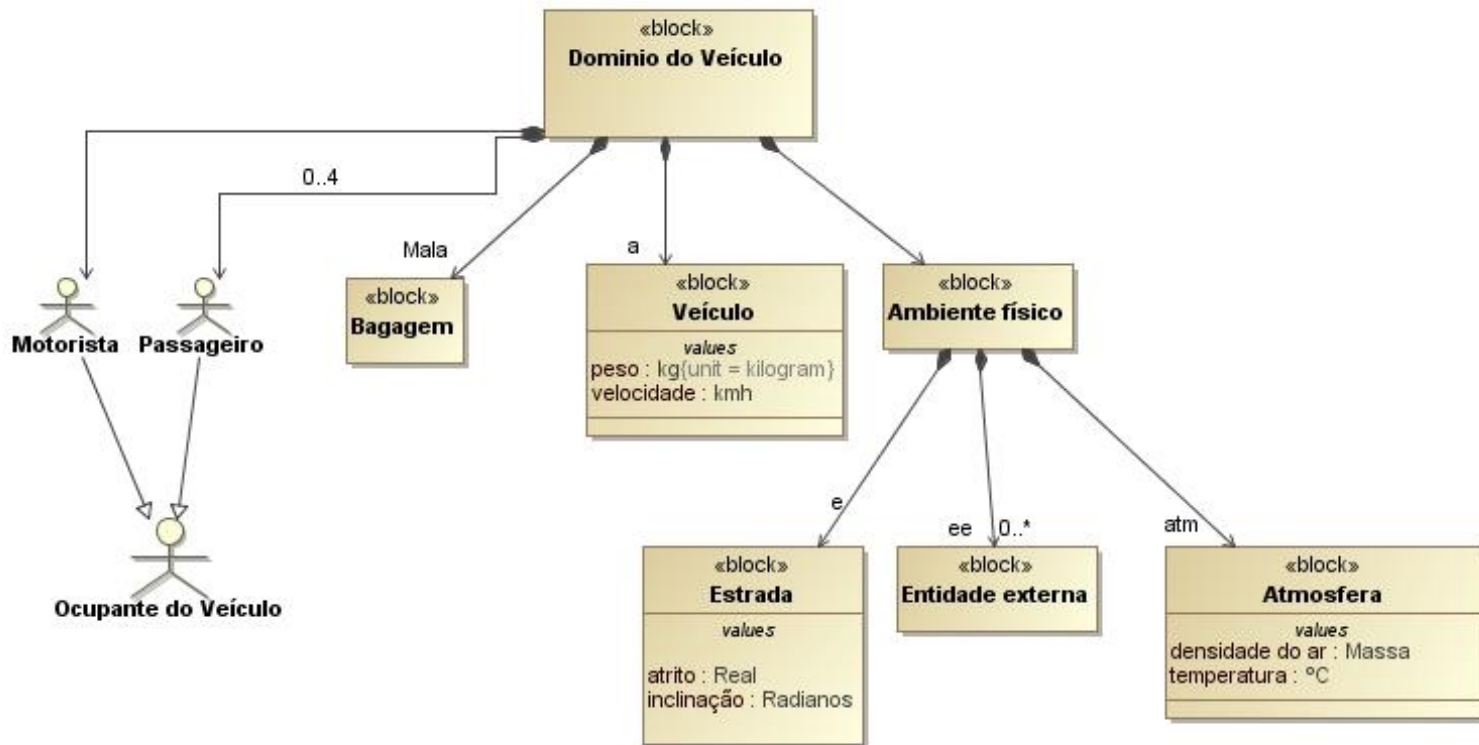
# Diagrama de Definição de Blocos

- Definem relações entre blocos
  - Composição
  - Associação
  - Especialização
- Captura propriedades
- Pode ser reusado em vários contextos



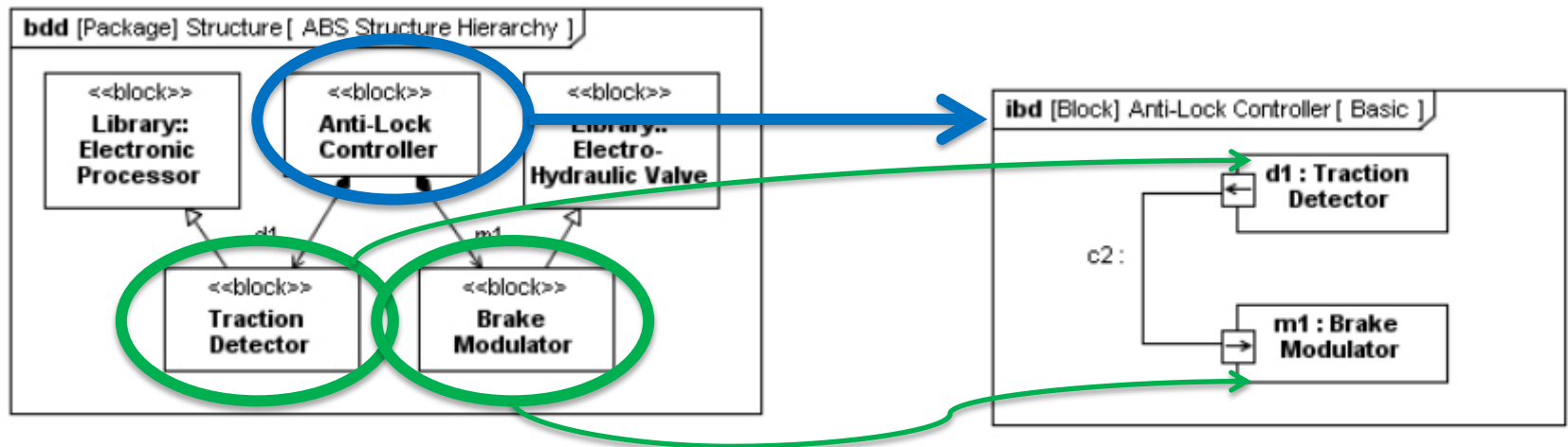
# Diagrama de Definição de Blocos

bdd [Model] Modelo Veículo [ Domínio do Veículo ]

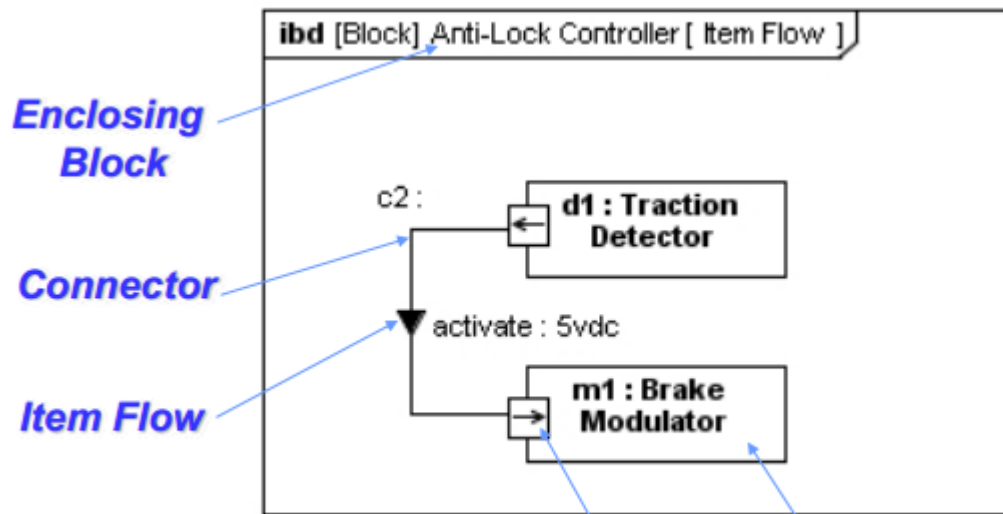


# Diagrama de bloco interno

- Descreve a estrutura interna de um bloco em termos de propriedades e conectores
- Expressa interconexão entre partes
- Enquanto o bdd representa a definição (equivalente a um “tipo de objeto”), o ibd ilustra o uso de um bloco (verde) no contexto (azul) de um bloco no qual ele faz parte



# Diagrama de bloco interno

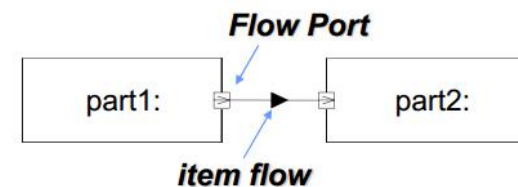
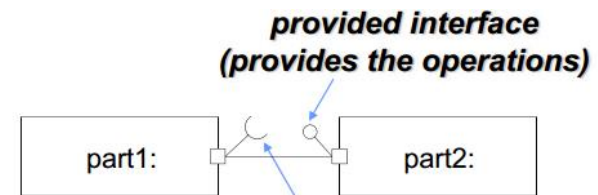


**Port**

**Part**

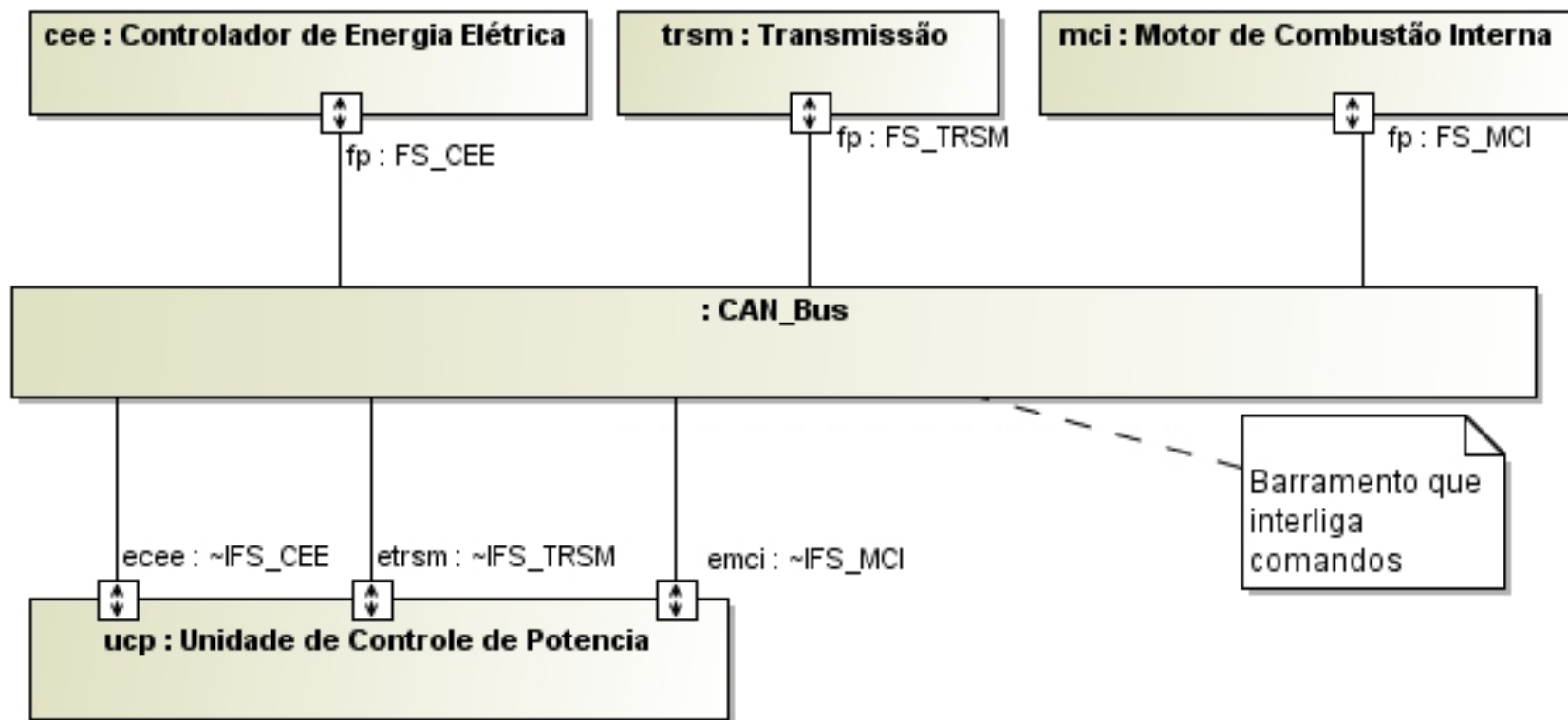
**Standard Port**

**Flow Port**



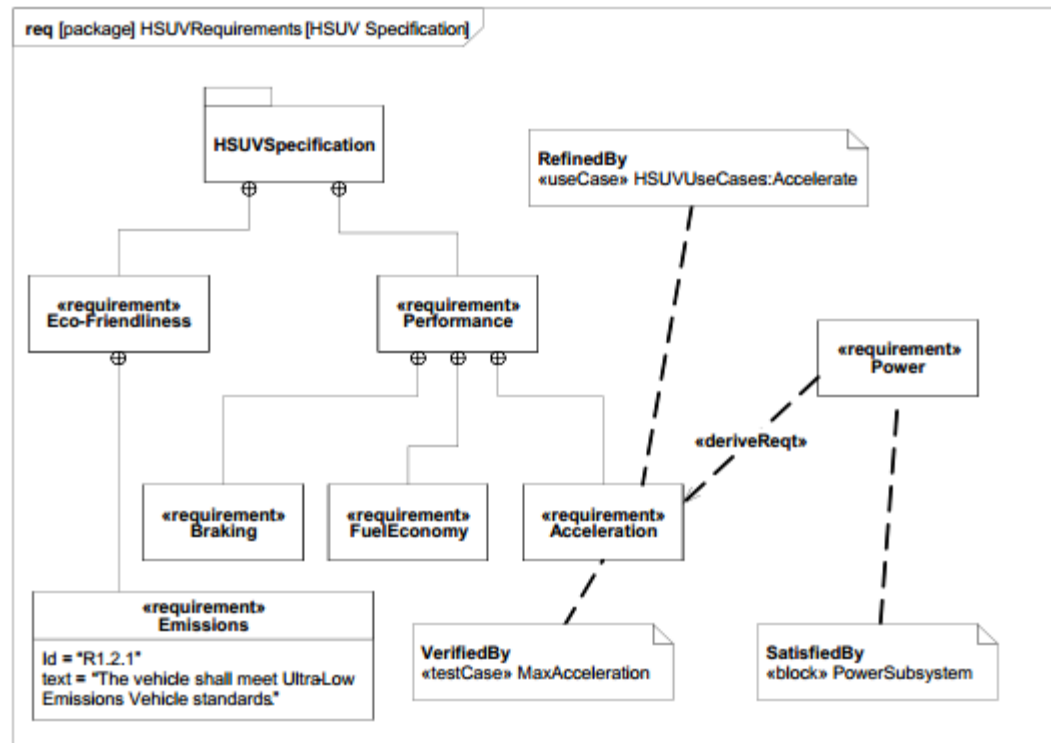
# Diagrama de Bloco Interno

**ibd** [Block] Subsistema de Energia [ Modelo Veículo TD ]



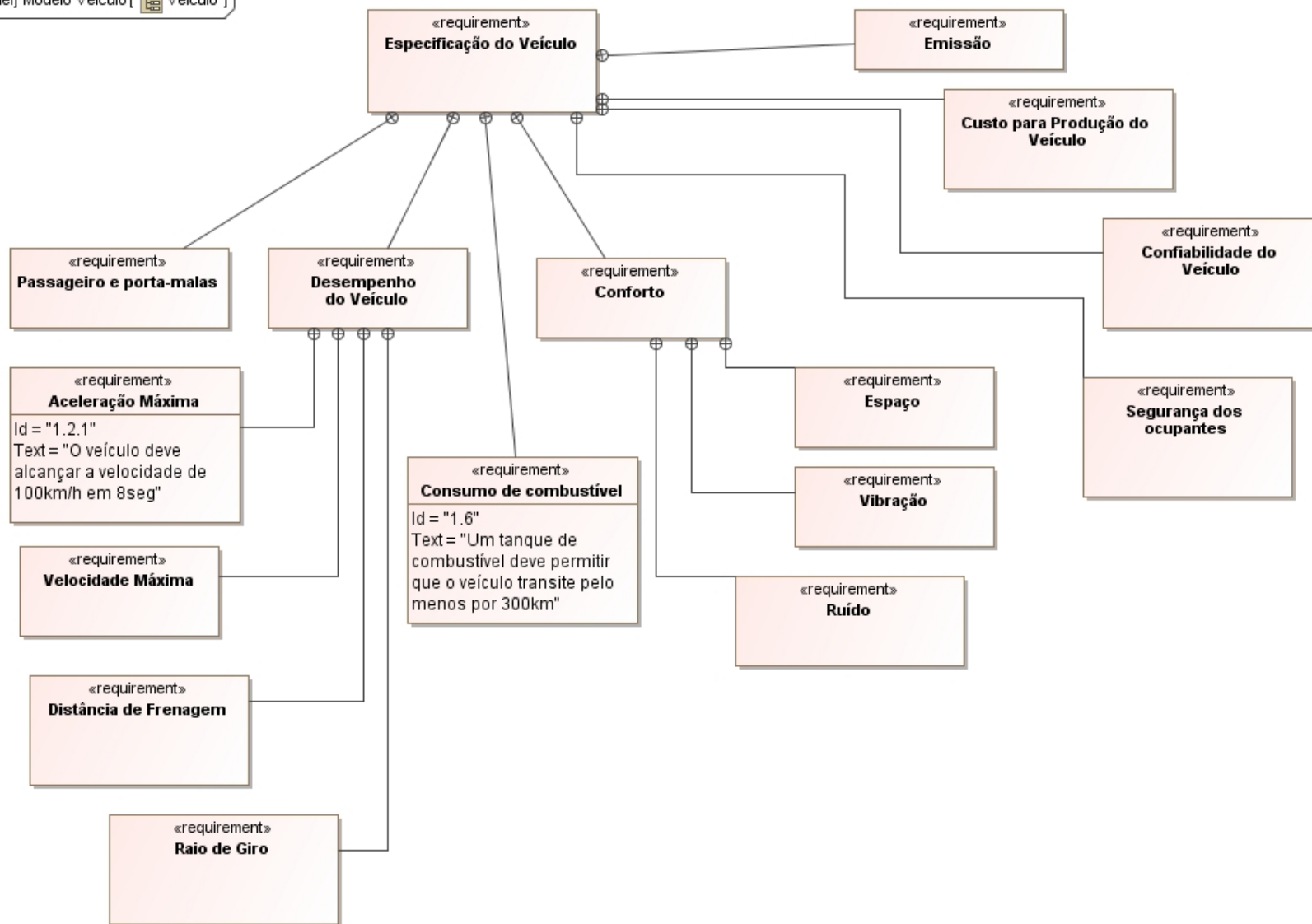
# Diagrama de Requisitos

- Representa requisitos do sistema
  - Inclui um id e um texto como propriedades
  - Pode ter propriedades definidas pelo usuário
  - Pode ter categorias definidas pelo usuário (funcional, interface, não funcional, etc)
- Pode descrever relações (verified by, satisfied by)

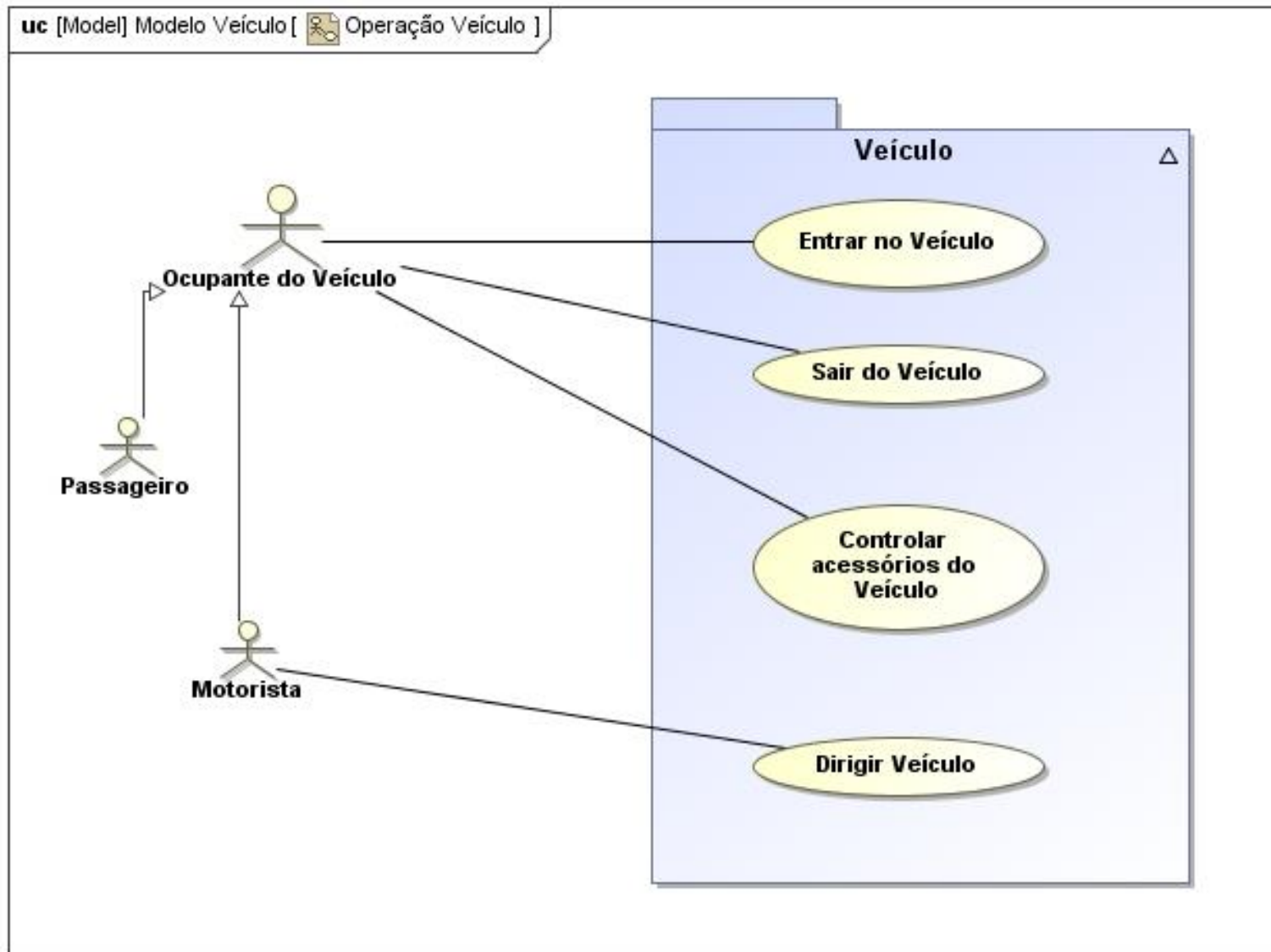


# Diagrama de Requisitos

req [Model] Modelo Veículo [ Veículo ]



# Diagrama de Casos de Uso

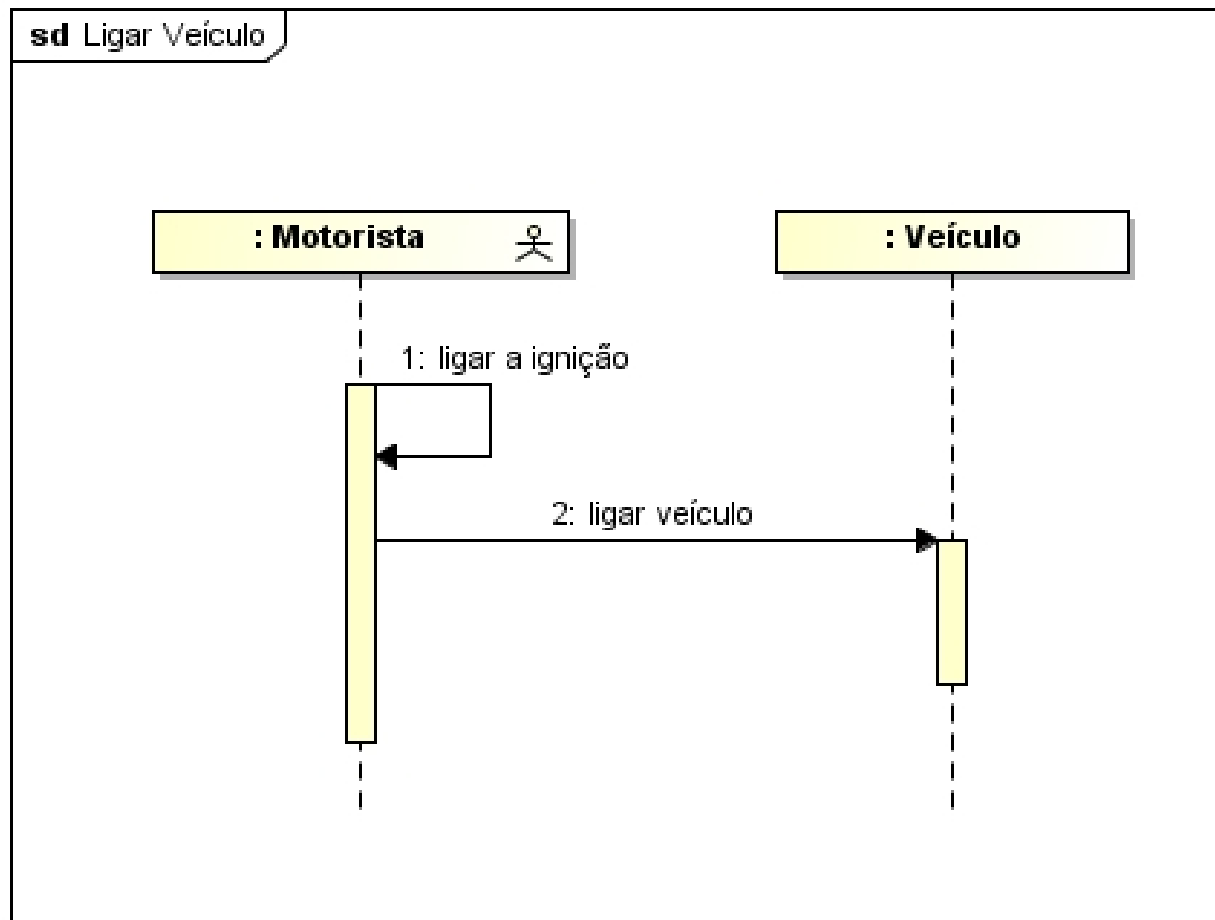




# Diagrama de Sequência

- Representa fluxo de controle
- Descreve interação entre partes
- Possui mecanismos para representar cenários mais complexos
  - opt (condicional com 1 alternativa)
  - alt (condicional com 2 ou mais alternativas)
  - loop (repetição)
  - break (aborta execução do resto do fluxo no contexto)
  - critical (execução atômica)
  - ...

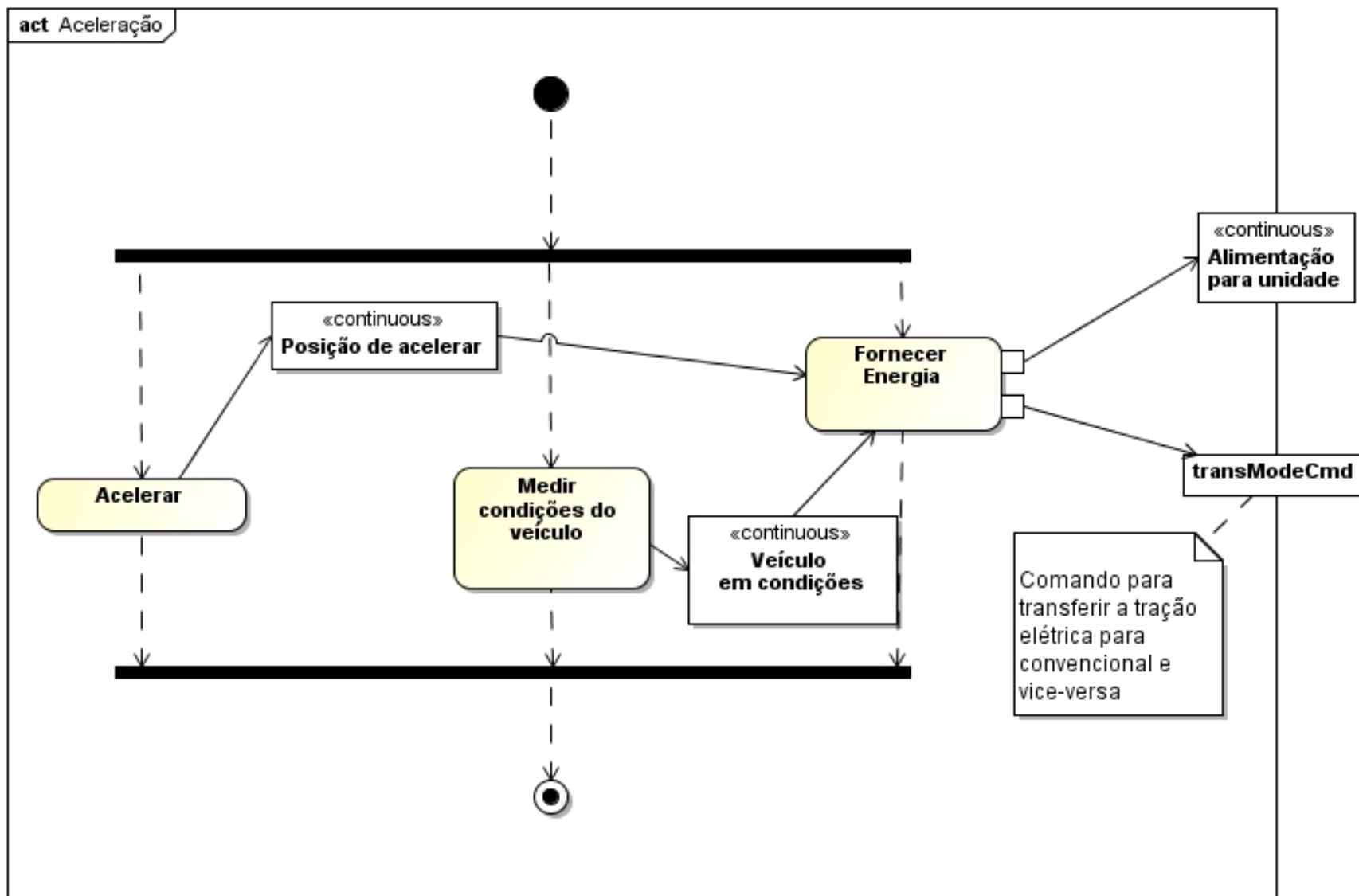
# Diagrama de Sequência



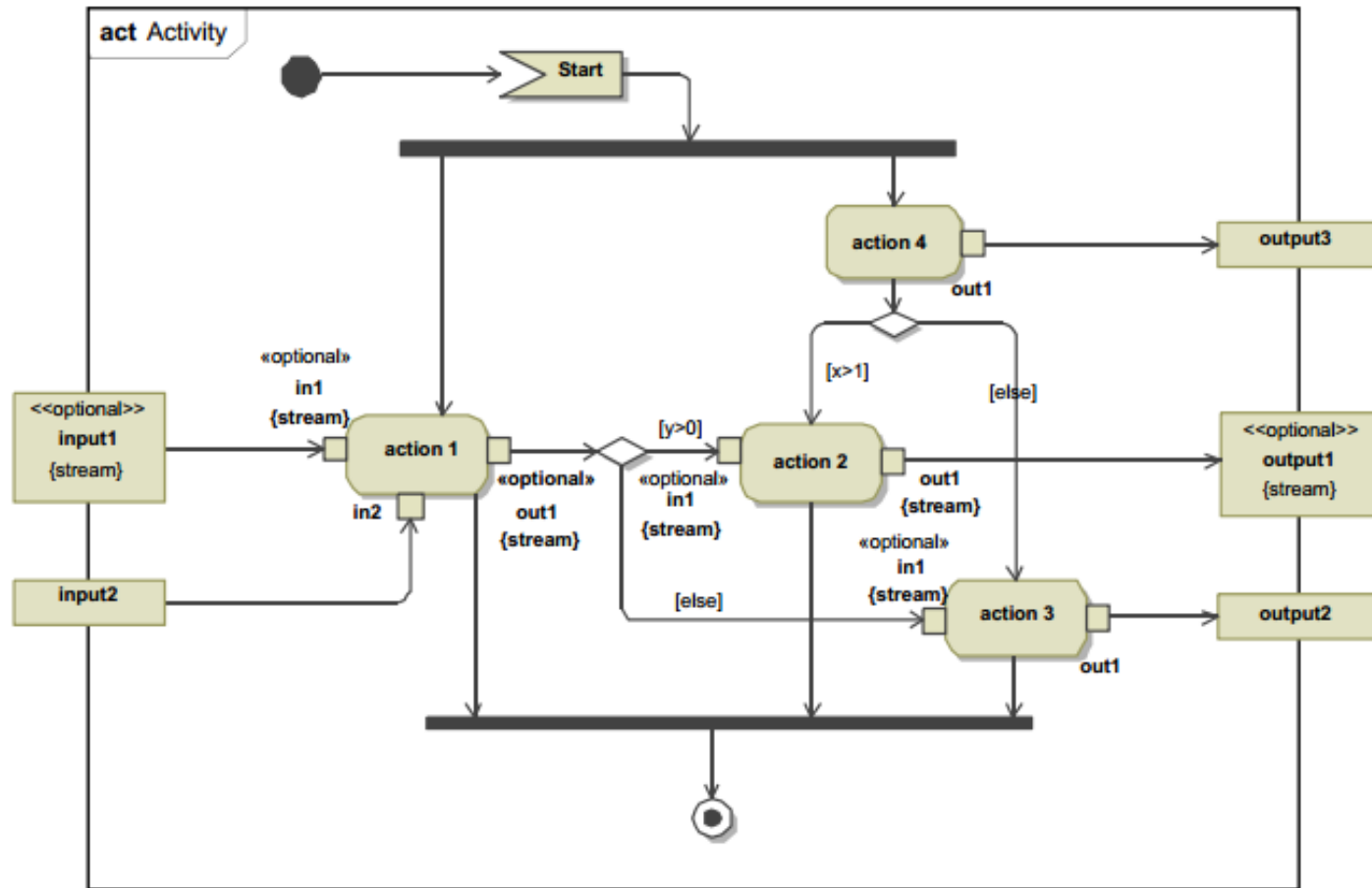
# Diagrama de Atividades

- Modelam **atividades**, a **ordem** em que são realizadas e **dependências** entre elas
  - Podem também indicar entradas e saídas das atividades
- Úteis para modelar **fluxos de trabalho**
- Exemplos:
  - Sequência de passos da descrição de um requisito
  - Processos dentro de uma empresa

# Diagrama de Atividades



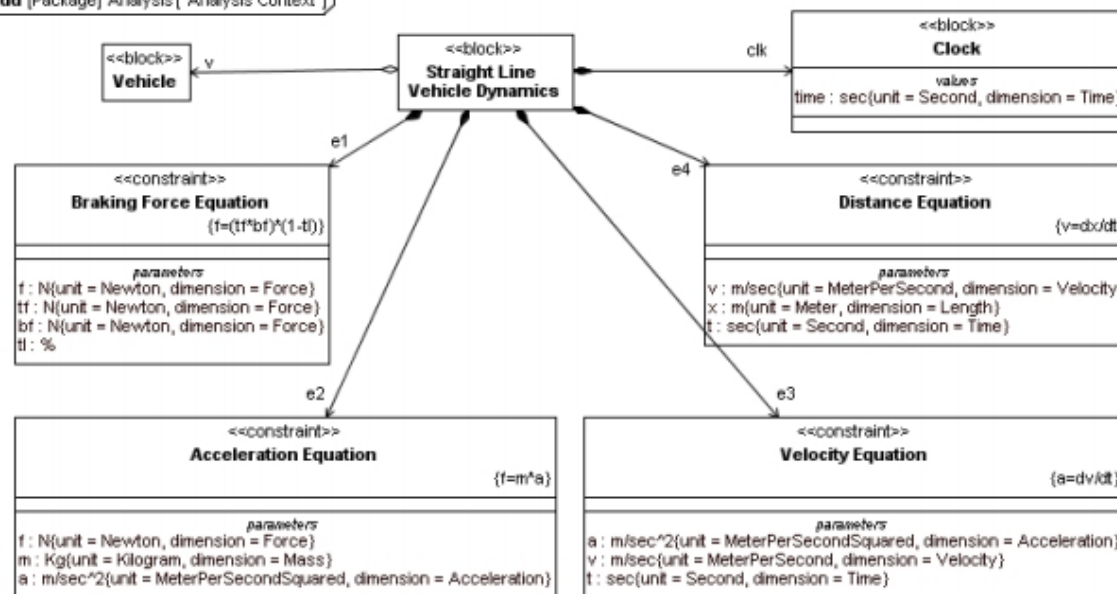
# Diagrama de Atividades



# Diagrama paramétrico

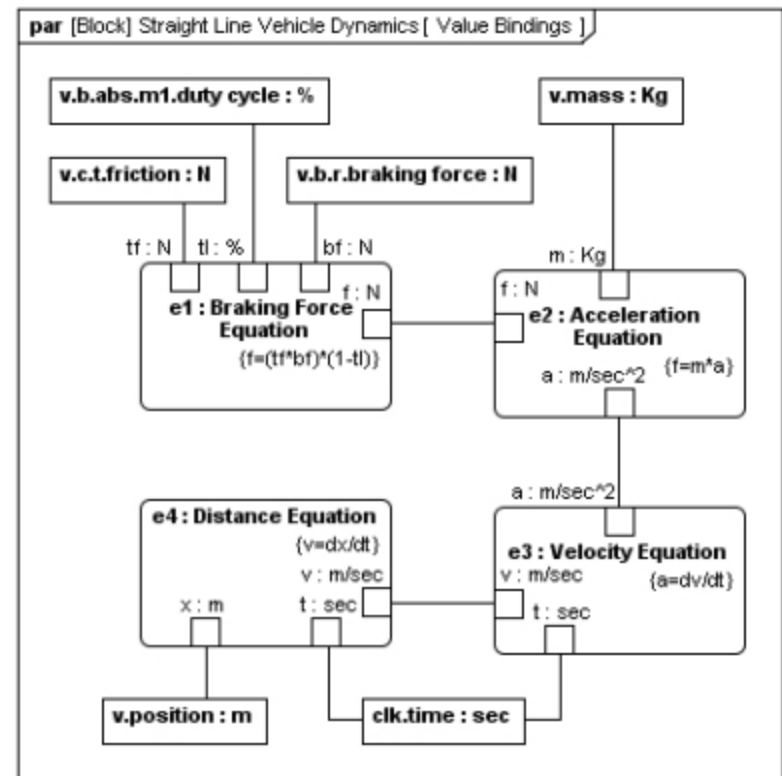
- Usado para exprimir constraints entres valores de propriedades
  - Dá suporte para análise
  - Facilita identificação de propriedades críticas de performance
- O bloco <<constraint>> captura equações
  - Eventual análise/execução da equação é feita por ferramenta e não no SysML
- Diagrama representa o uso das constraints em um contexto de análise

bdd [Package] Analysis [ Analysis Context ]



Contraints definidas em diagrama de blocos

Diagrama paramétrico ilustrando sistemas de equações interligados com as propriedades dos blocos



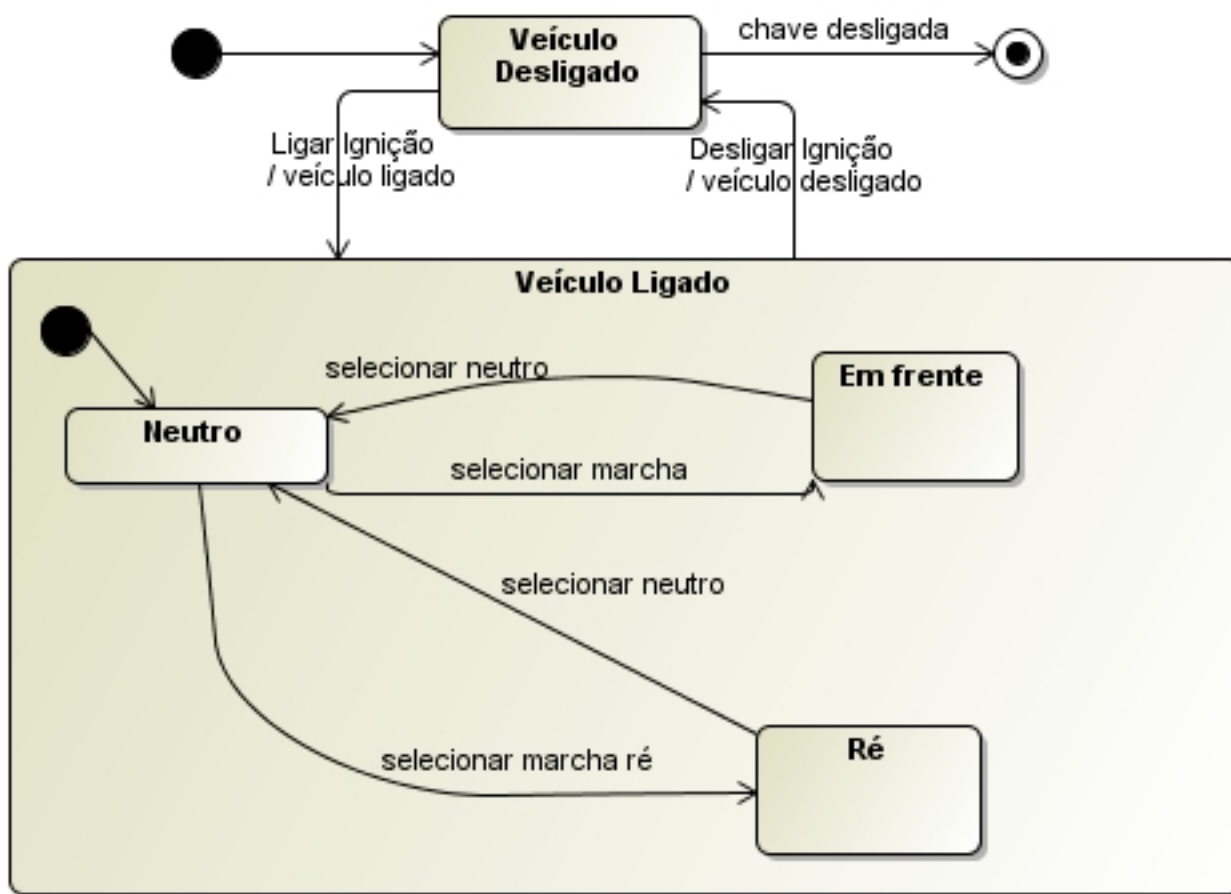
# Diagrama de Máquina de estado

- Modelam o comportamento do sistema em resposta aos **eventos externos e internos**
- Mostram os estados do sistema como **nós** e os eventos como **arestas** entre estes nós.
  - Quando um evento ocorre, o sistema muda de um estado para um outro.
- **Statecharts** são usados para representar os modelos de máquina de estados na SysML



# Diagrama de Máquina de estado

stm Máquina de Estados Ligar Veículo



# Referências complementares

SOMMERVILLE, I. Engenharia de Software. 9ª. Ed. São Paulo: Pearson Education, 2011. Capítulo 5 (modelagem de sistemas)

S. Friedenthal, A. Moore, R. Steiner. **A Practical Guide to SysML: The Systems Modeling Language. 2ed. 2011**

Martin Fowler e Kendall Scott, *UML Essencial*. 3ª Edição, Bookman Companhia Ed., 2004.

Bom livro de **referência, i.e., para tirar dúvidas sobre a notação UML**

Tutoriais SysML:

<http://www.sysmlforum.com/sysml-tutorials/>

Ferramentas:

<http://www.sysmlforum.com/sysml-tools/>