



## IV - Unified Modeling Language (UML)

António Palma dos Reis

Aristides Sousa Mendes

Filipa Pires da Silva

Winnie Picoto



# Índice

---

- O que é o UML e Respectiva Evolução
- Elementos Básicos de UML
- Diagramas de Casos de Utilização
- Diagramas de Actividades
- Diagramas de Classes
- Diagramas de Estados
- Diagramas de Sequência



# UML (Unified Modeling Language) (1/2)

---

- É uma linguagem para especificar, visualizar, construir e documentar artefactos de sistemas diversos (software, negócios ou outros);
- Permite especificar:
  - » a estrutura dos sistemas (elementos estáticos),
  - » o comportamento dos sistemas (elementos dinâmicos);
- A UML resultou da fusão de 3 métodos OO para análise e desenho:
  - » *Object Modeling Technique* (OMT) (de Rumbaugh)
  - » *Método de Booch* (de Booch)
  - » *Object Oriented Software Engineering* (OOSE) (de Jacobson).



## UML (Unified Modeling Language) (2/2)

---

- A especificação é realizada com recurso a diagramas;
- Cada diagrama apresenta uma perspectiva específica sobre o sistema;
- Mais informação em *<http://www.omg.com>*



# A Evolução do UML

Fragmentação [mead-1970/mead 1990]

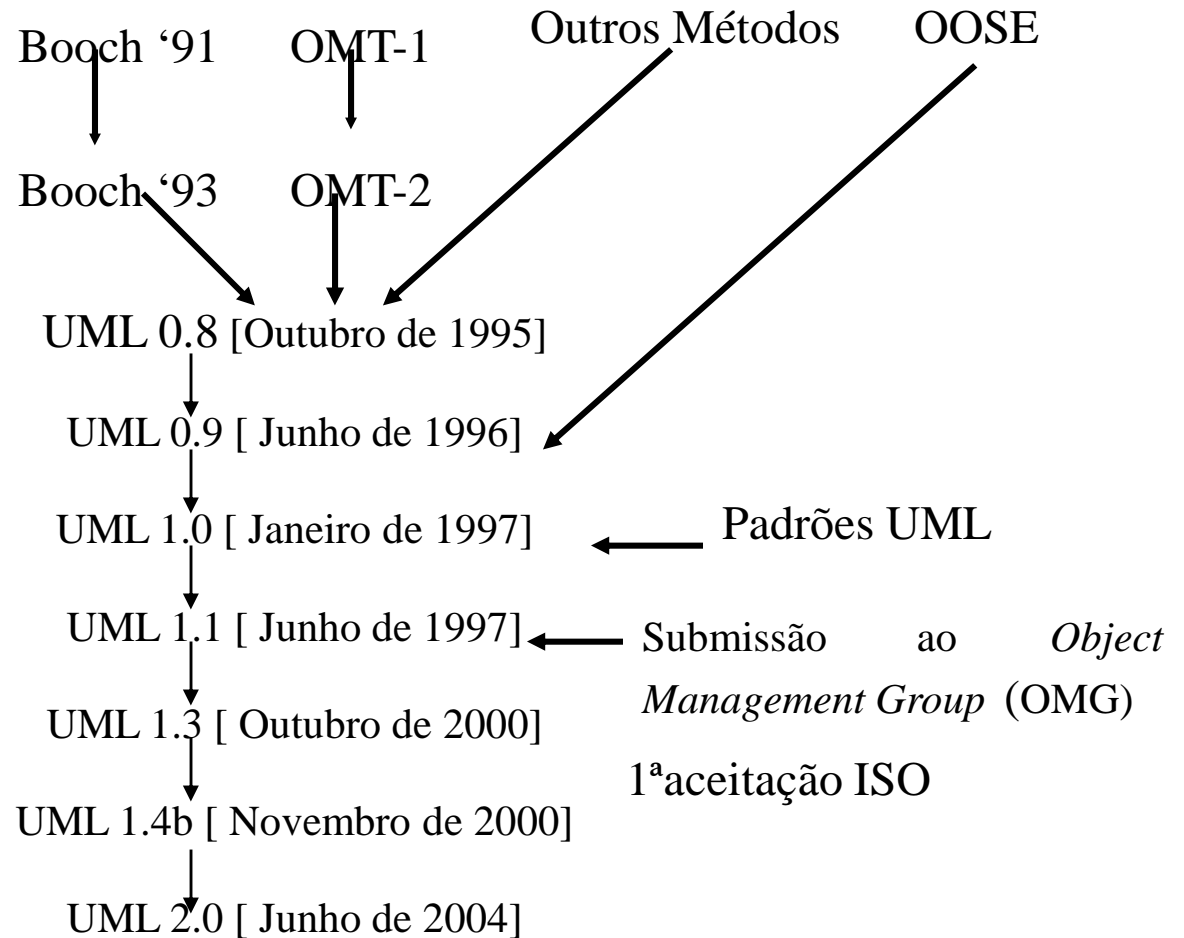
1ª Geração de Métodos e Ling. de Modelação

2ª Geração de Métodos e Ling. de Modelação

Unificação  
[Outubro de 1995]

Padronização  
[Novembro de 1997]

Industrialização  
[Outubro de 1998]





## Elementos Básicos (3/5)

---

### Tipos de Diagramas(1/3):

- Diagramas de Visão Estrutural ou Estática
  - » Diagramas de Classes – estrutura estática da informação do sistema
  - » Diagramas de Componentes – dependências entre componentes de software
  - » Diagramas de Instalação – configuração dos componentes físicos (software e hardware).



## Elementos Básicos (4/5)

---

### Tipos de Diagramas (2/3):

- Diagramas de Visão Funcional
  - » Diagramas de Casos de Utilização – permitem a visão do sistema na perspectiva dos seus utilizadores
  - » Diagramas de Actividades – representam as diversas actividades levadas a cabo pelos actores



# Elementos Básicos (5/5)

---

## Tipos de Diagramas (2/3):

- Diagramas de Visão Dinâmica
  - » Diagramas de Interação
    - Diagramas de Sequência – mensagens entre objectos ao longo do tempo
    - Diagramas de Comunicação – visão não temporal da comunicação entre os objectos
  - » Diagramas de Estados (ou Máquinas de Estados) – representação do ciclo de vida dos objectos do sistema, através dos estados por que vão passando





# Diagramas de Casos de Utilização

---

- Usados preferencialmente nas tarefas de especificação de requisitos e/ou de modelação de processos de negócio
- Permitem capturar os requisitos funcionais, segundo uma aproximação focada primordialmente na utilização do sistema
- Dão uma visão global e de alto nível do sistema
- Referem os grandes módulos do sistema pelos agentes externos ao sistema, denominados actores;



# Diagramas de Casos de Utilização

- Caso de Utilização:
  - » é uma sequência de acções que um ou mais actores realizam num sistema de modo a obterem um resultado particular
  - » corresponde a um conjunto de actividades que se traduzem por sequências de mensagens entre objectos.

## Sintaxe



Oval com uma frase que representa uma determinada acção e que deve estar escrita na voz activa, com um verbo no infinitivo

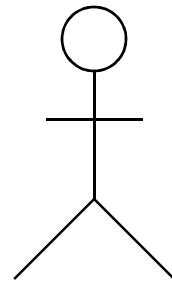


# Diagramas de Casos de Utilização

- Actor:

- » é o conceito que representa, em geral, um papel que um utilizador desempenha relativamente ao sistema em análise. Este utilizador não tem de ser um utilizador humano, podendo representar outro sistema informático, equipamento ou a simples passagem de tempo
- » apesar de os actores não fazerem parte do sistema a implementar, a sua modelação é fundamental para estabelecer os interfaces do sistema com o mundo exterior (a sua fronteira)

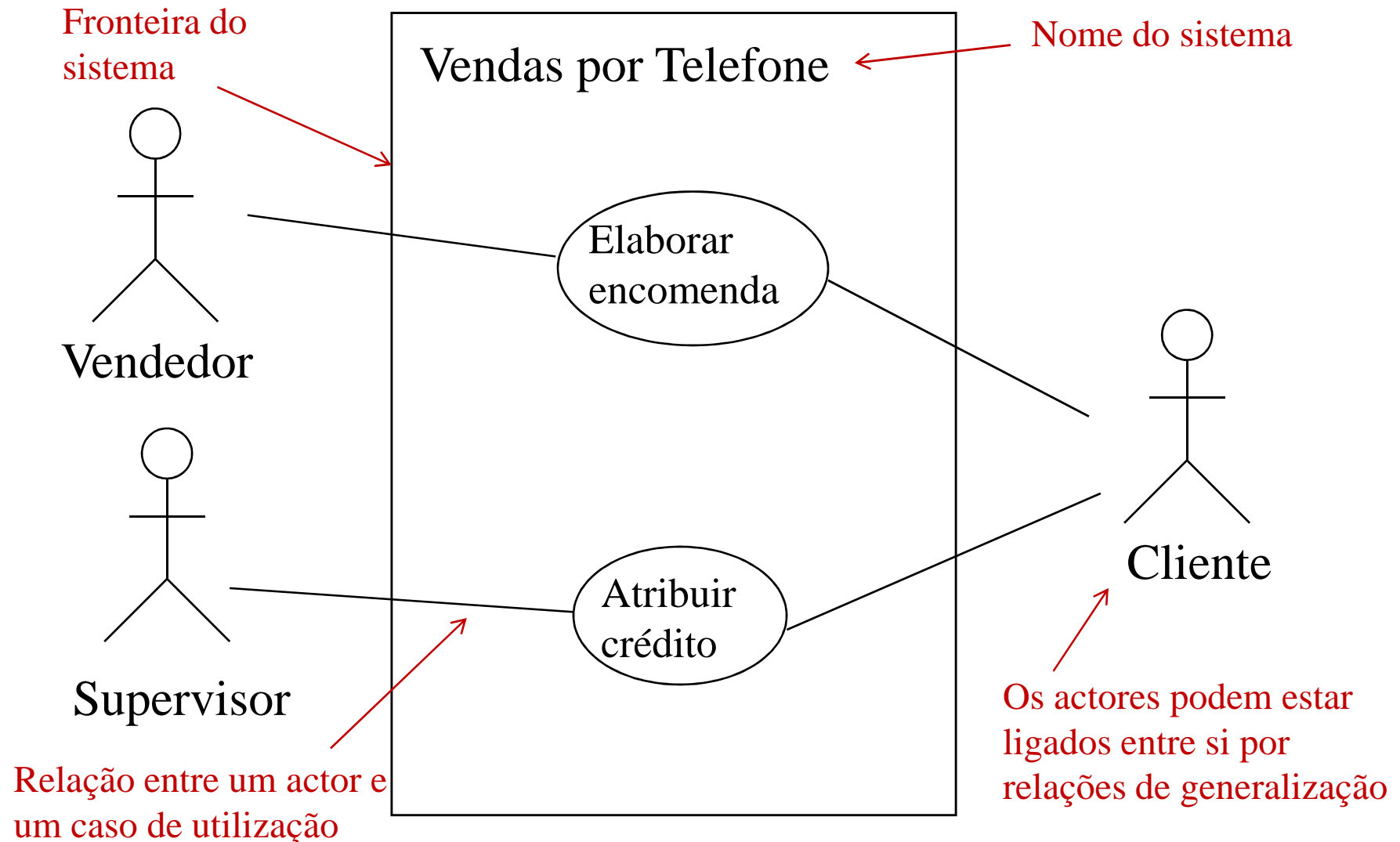
- » **Sintaxe:**



Cliente



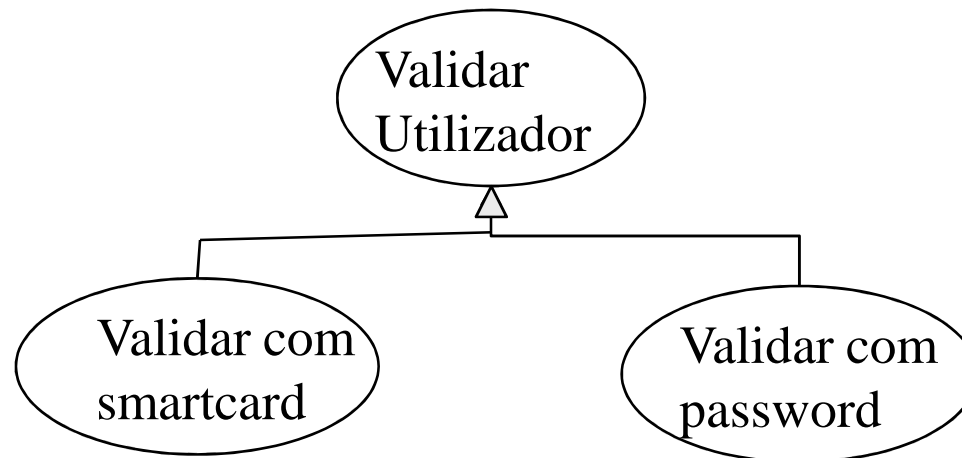
# Exemplo de Diagrama de Casos de Utilização





# Ligações entre Casos de Utilização

- Os casos de utilização podem estar relacionados por três tipos de relacionamentos:
  - Generalização
    - » Permite definir casos à custa de outros já existentes, sendo que o caso de utilização filho herda as propriedades e semântica definidas para o seu pai
    - » Deve-se evitar esta ligação adoptando alternativamente ligações de inclusão e de extensão

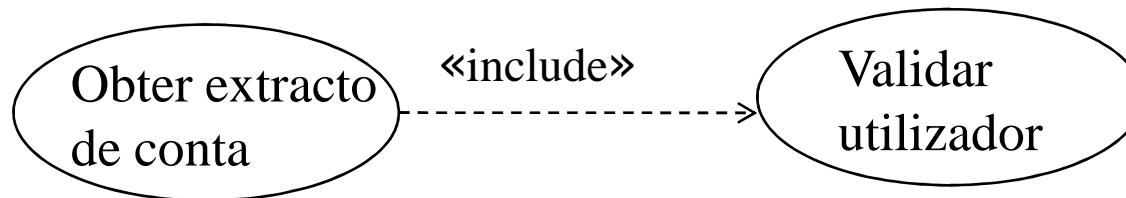




# Ligações entre Casos de Utilização

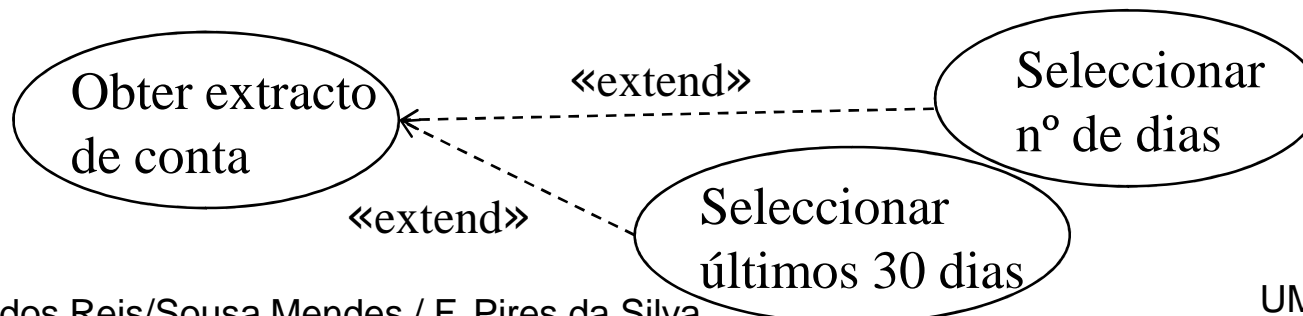
- Inclusão

- » O caso base incorpora o comportamento do outro caso relacionado



- Extensão

- » O caso destino pode ser estendido com o comportamento de outro(s) caso(s)
- » Permite representar a parte de um caso que um utilizador vê como opcional, ou como existindo várias alternativas





## Caso de Estudo “Aluguer de Equipamento”

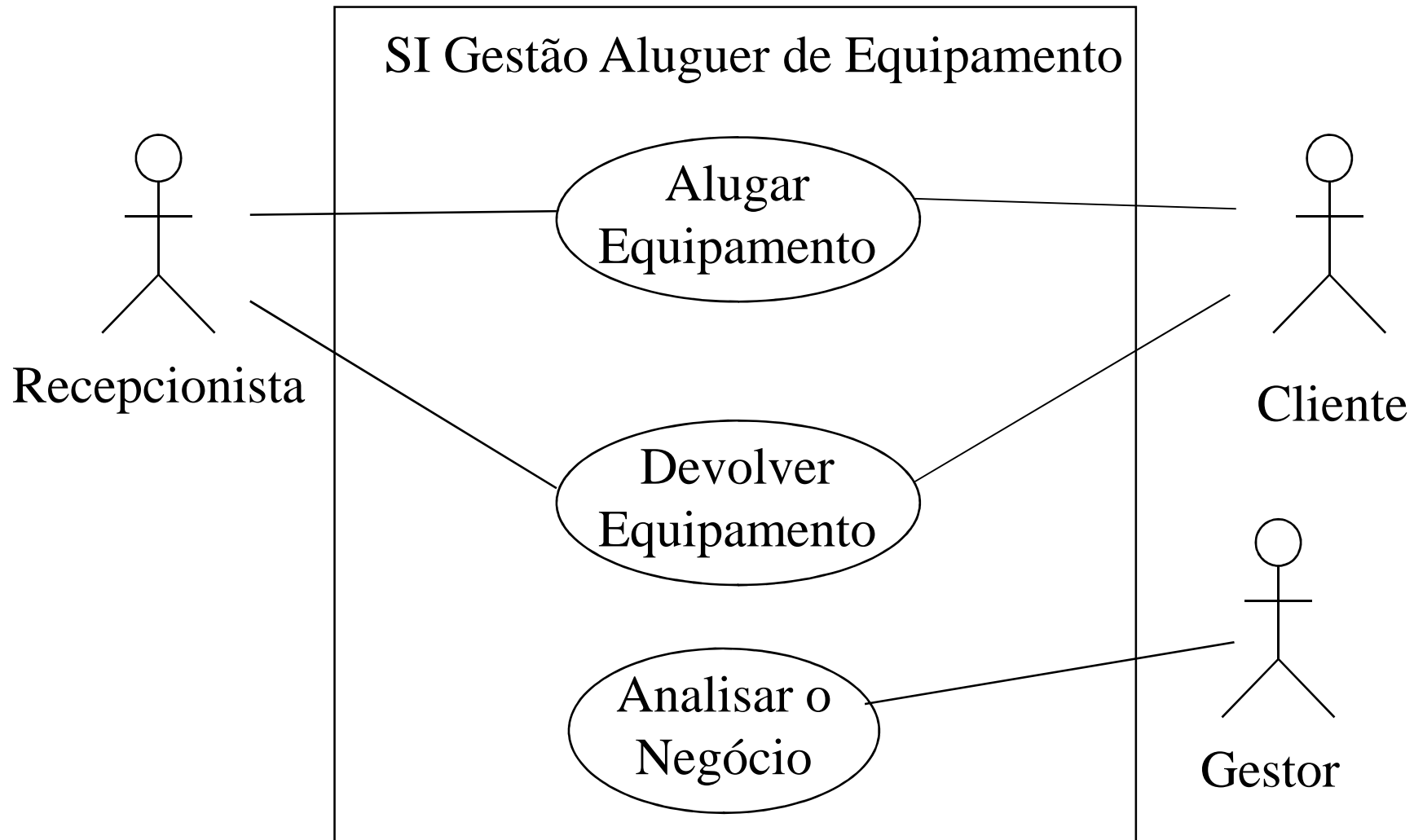
---

Pretende-se desenvolver um SI que permita gerir uma casa de aluguer de equipamento funcionando segundo os seguintes tópicos:

- o aluguer é formalizado por um contrato, de que constam, para além dos equipamentos a alugar e da data prevista de devolução, os elementos do seguro que será accionado em caso de qualquer avaria ou deterioração dos equipamentos alugados;
- se o cliente devolver os equipamentos fora do prazo definido, será sujeito a uma multa que depende do tipo de equipamento;
- no momento da devolução, o recepcionista verifica o estado dos equipamentos e o aluguer é facturado na última devolução do contrato.



# Caso de Estudo “Aluguer de Equipamento”

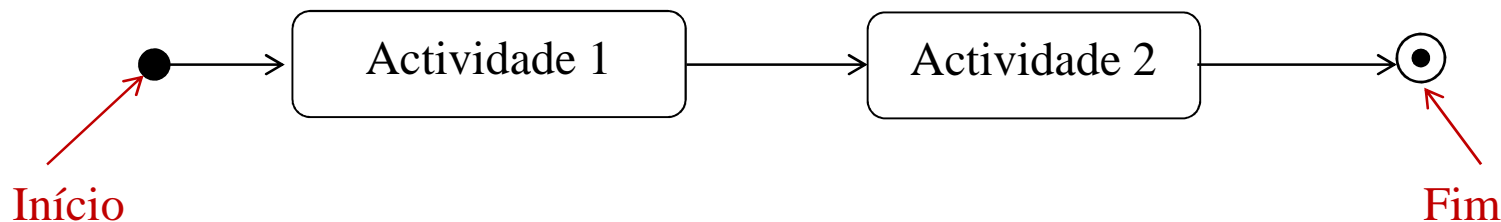






# Diagramas de Actividades

- Adequados à modelação da visão funcional de um sistema, porque permitem descrever a lógica dos seus processos ou das suas funções
- Descrevem o comportamento interno desses processos ou funções, representando uma série de acções e/ou actividades e explicitando as dependências entre as mesmas





# Diagramas de Actividades

- Actividade é um passo de um processo onde algum trabalho é realizado (cálculo, manipulação, pesquisa, etc)

- » Actividades sequenciais - uma actividade só pode ser executada quando todas as actividades que dela dependam já tiverem sido executadas.

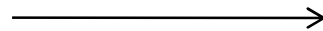
- » Actividades concorrentes – as actividades podem ser executadas em paralelo.

- » **Sintaxe:**

Inserir pedido

- Transição representa em geral que uma actividade terminou e o processo ou função deve evoluir para uma actividade seguinte ou simplesmente terminar

- » **Sintaxe:**

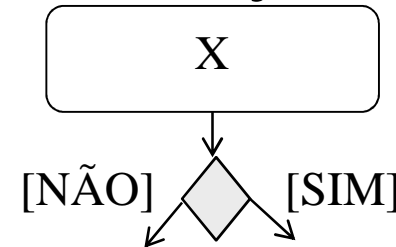




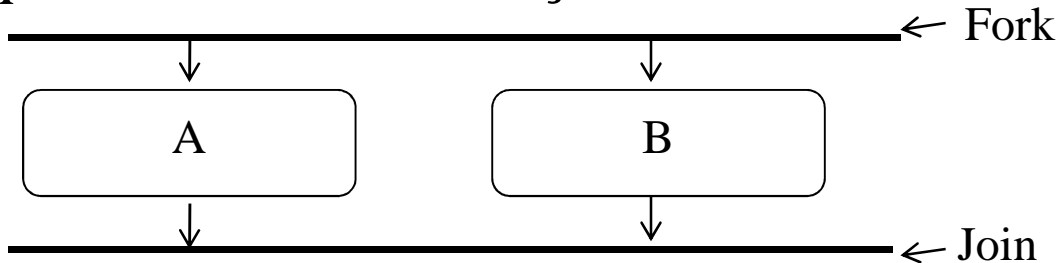
# Diagramas de Actividades

- Nos Diagramas de Actividades também podemos ter:
  - » Nós de decisão: mecanismo que consiste em especificar que actividade deve ser realizada após a execução da actividade corrente

É suportada por uma condição de guarda:  
expressão booleana limitada por parêntesis  
rectos

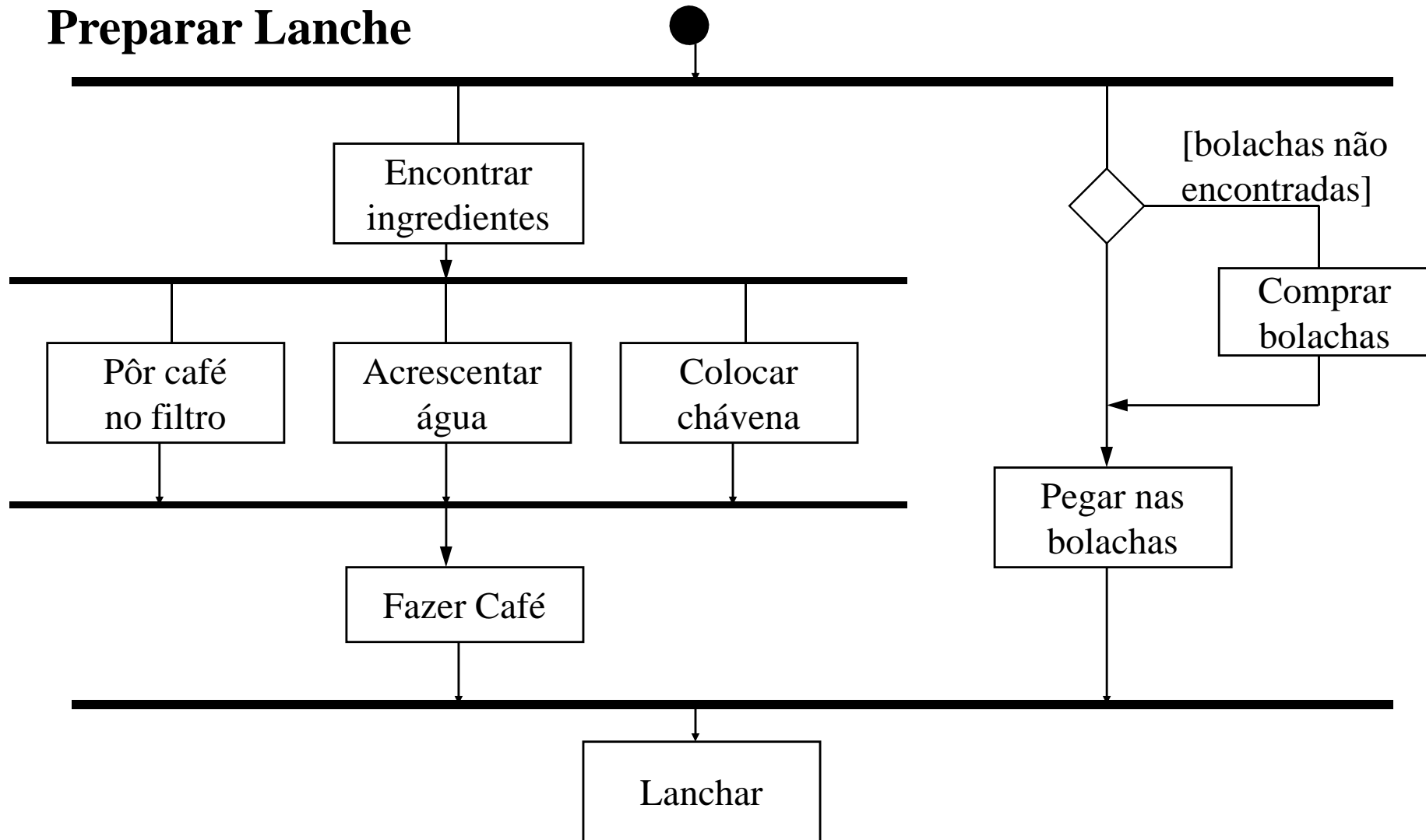


- » Nós de difusão (fork) e junção (join): descrevem um conjunto de actividades que devem ser realizadas, mas em que a ordem de execução não é relevante



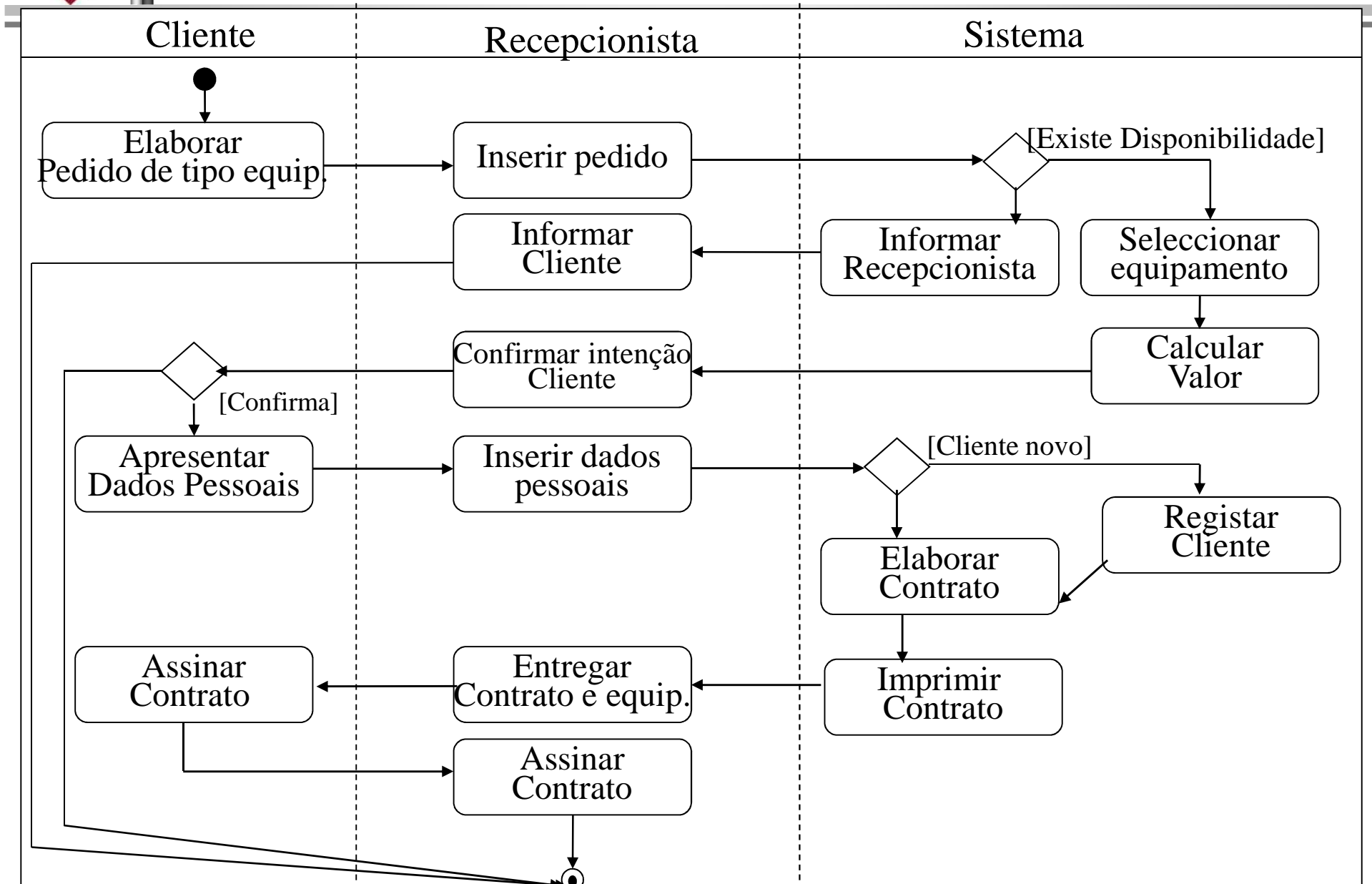


# Exemplo de Diagrama de Actividades



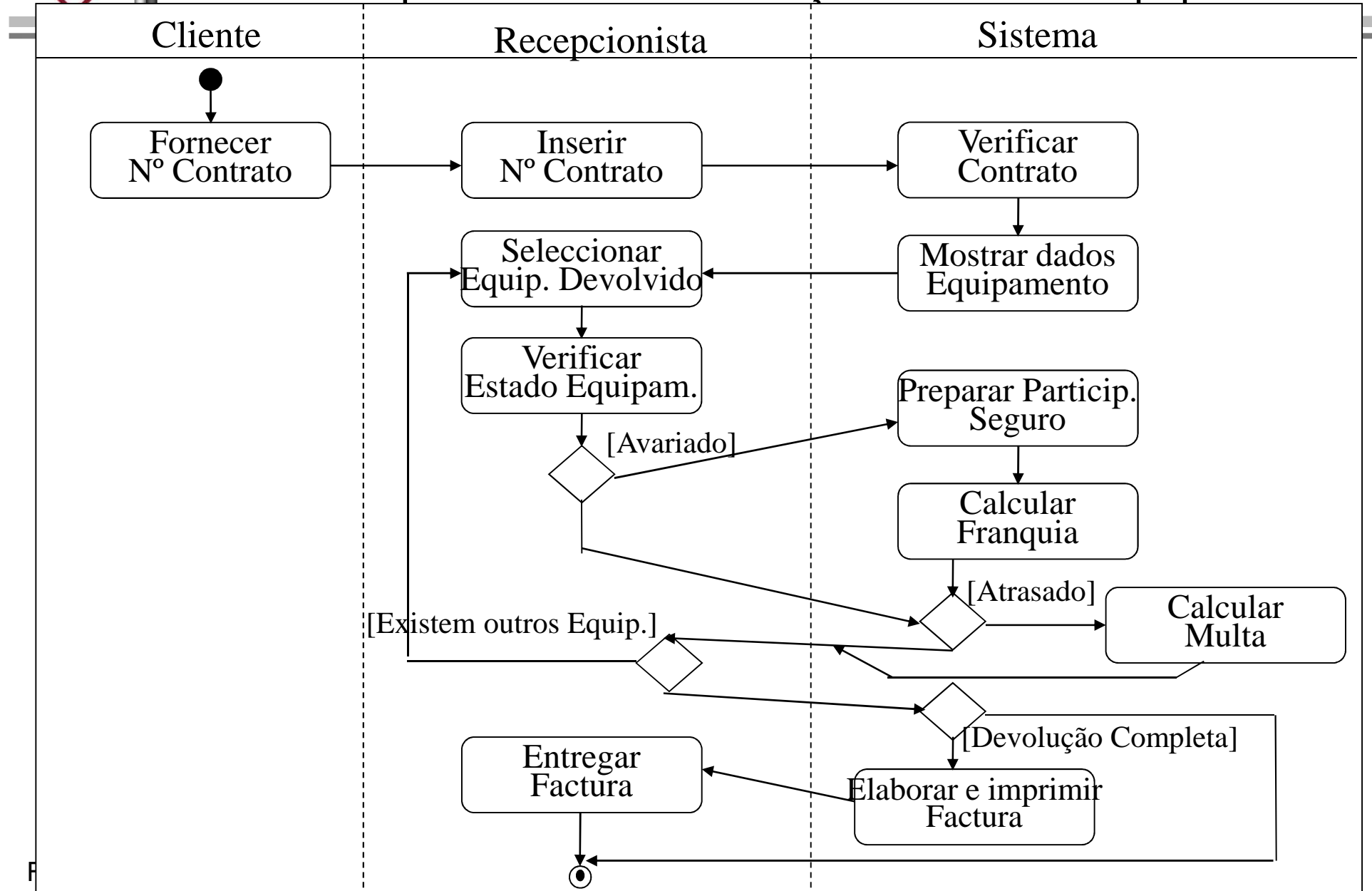


# Caso de Estudo “Aluguer de Equipamento” – Diagrama de Actividades para o Caso de Utilização: “Alugar Equipamentos”





# Caso de Estudo “Aluguer de Equipamento” – Diagrama de Actividades para o Caso de Utilização: “Devolver Equipamentos”

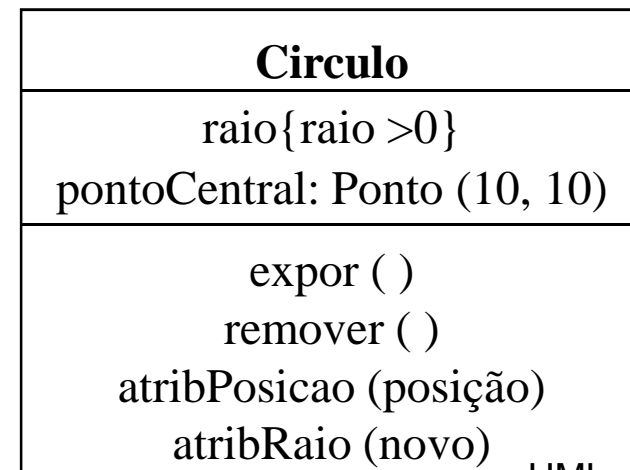




# Diagramas de Classes

- Um Diagrama de Classes é um grafo em que:
  - » os nós representam as classes,
  - » os arcos representam as relações entre as classes;
- Mostra quais as abstracções importantes do sistema e como estão relacionadas;
- É uma representação estática: inclui as classes que uma dada organização conhece e usa;

**Não descreve o  
comportamento do sistema!**

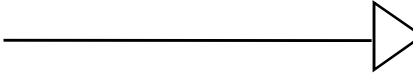


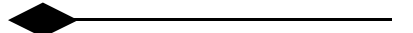




# Relacionamentos

- Um relacionamento em UML estabelece a ligação entre elementos e é representada graficamente por um determinado tipo de linha.

Os tipos mais importantes são:

- » Generalização 
- » Associação 
- » Agregação 
- » Composição 

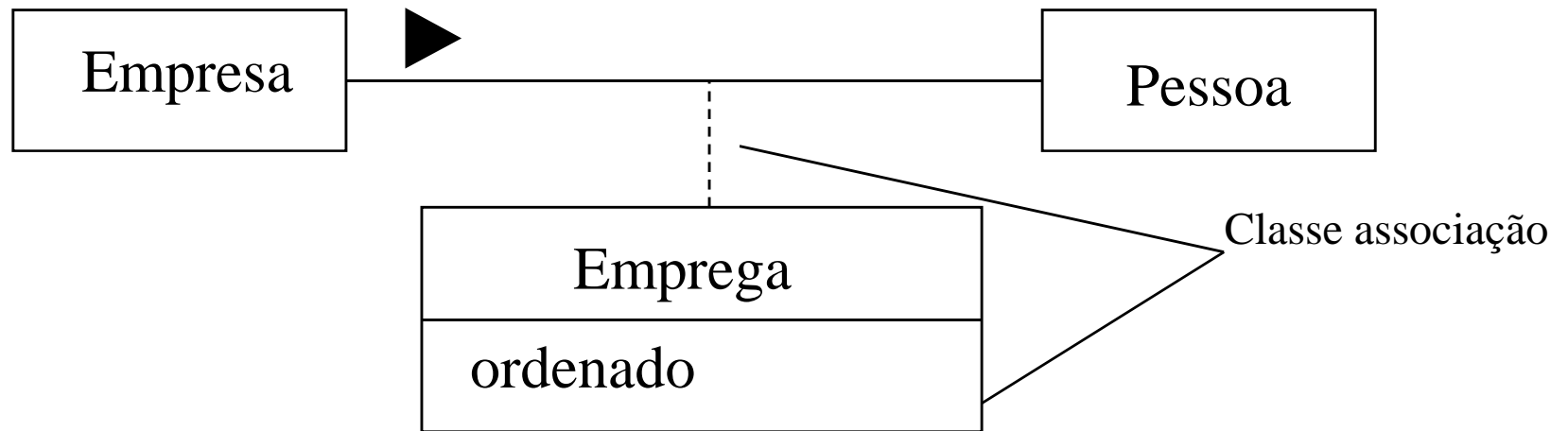




## Classe-Associação ou Associação Atributiva

- A associação pode possuir os seus próprios atributos devendo ser representada como uma classe.

### *Diagrama de Classes*





# Grau das Associações

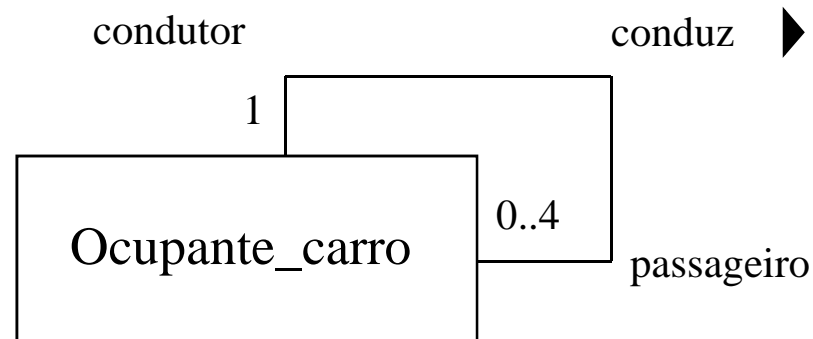
---

- Grau (ou Aridade) traduz o número de classes que intervêm numa dada associação.
- As associações podem ser
  - » Binárias – associação entre duas classes (caso mais frequente)
  - » Unárias (ou Reflexivas) – associação de uma classe consigo própria
  - » N-árias – associação entre mais de duas classes



# Associações Unárias ou Reflexivas

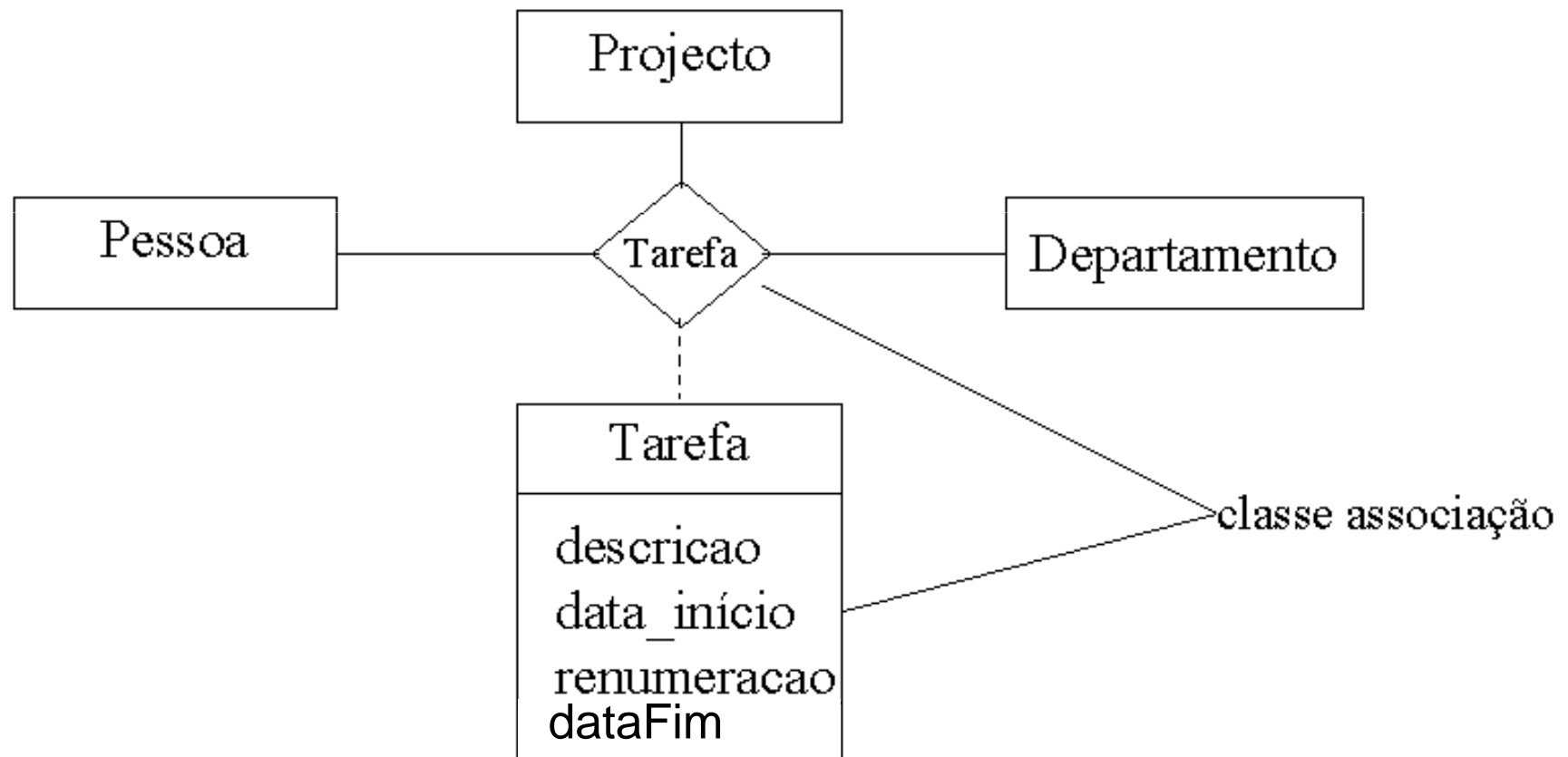
- Uma associação diz-se reflexiva quando estabelece uma relação duma classe consigo própria.
- Este tipo de associação acontece quando as ocorrências de uma classe desempenham diferentes papéis.





# Associações N-Árias ( $N \geq 3$ )

- As associações N-Árias ( $N \geq 3$ ) são relativamente pouco comuns





## Caso de Estudo “Aluguer de Equipamento”

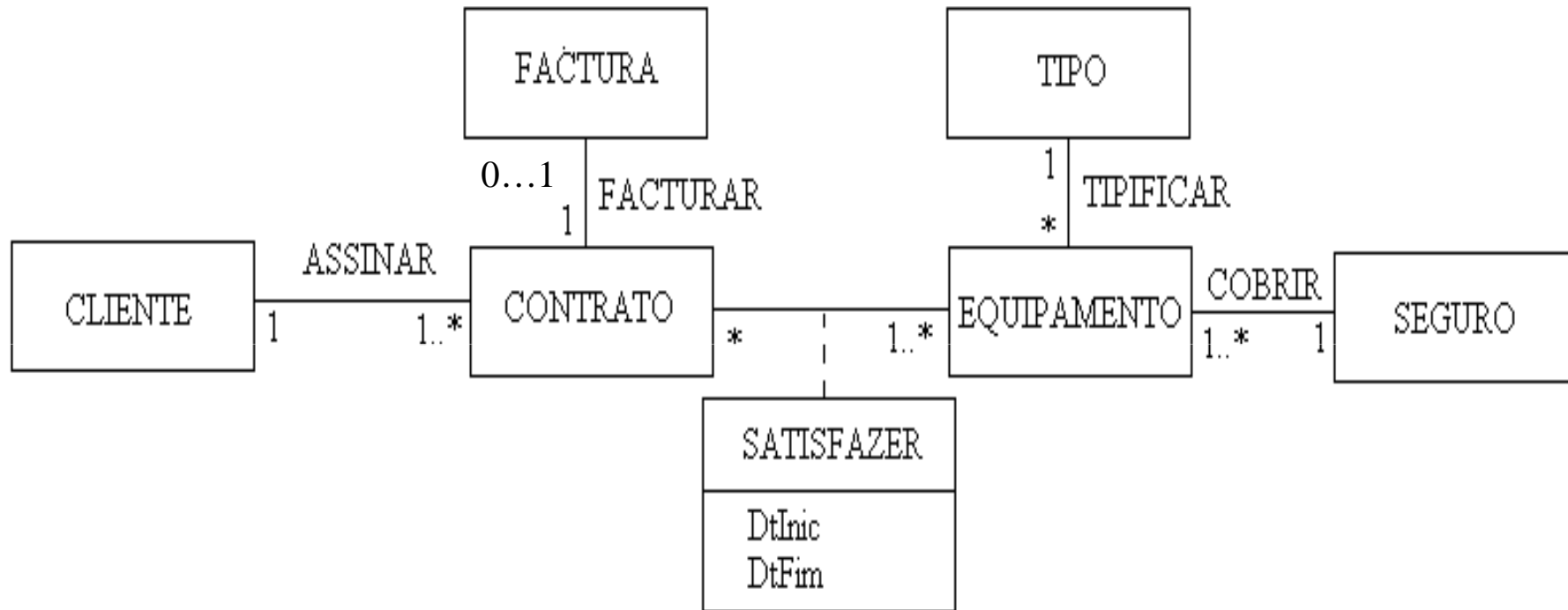
---

Pretende-se desenvolver um SI que permita gerir uma casa de aluguer de equipamento funcionando segundo os seguintes tópicos:

- o aluguer é formalizado por um contrato, de que constam, para além dos equipamentos a alugar e da data prevista de devolução, os elementos do seguro que será accionado em caso de qualquer avaria ou deterioração dos equipamentos alugados;
- se o cliente devolver os equipamentos fora do prazo definido, será sujeito a uma multa que depende do tipo de equipamento;
- no momento da devolução, o recepcionista verifica o estado dos equipamentos e o aluguer é facturado na última devolução do contrato.

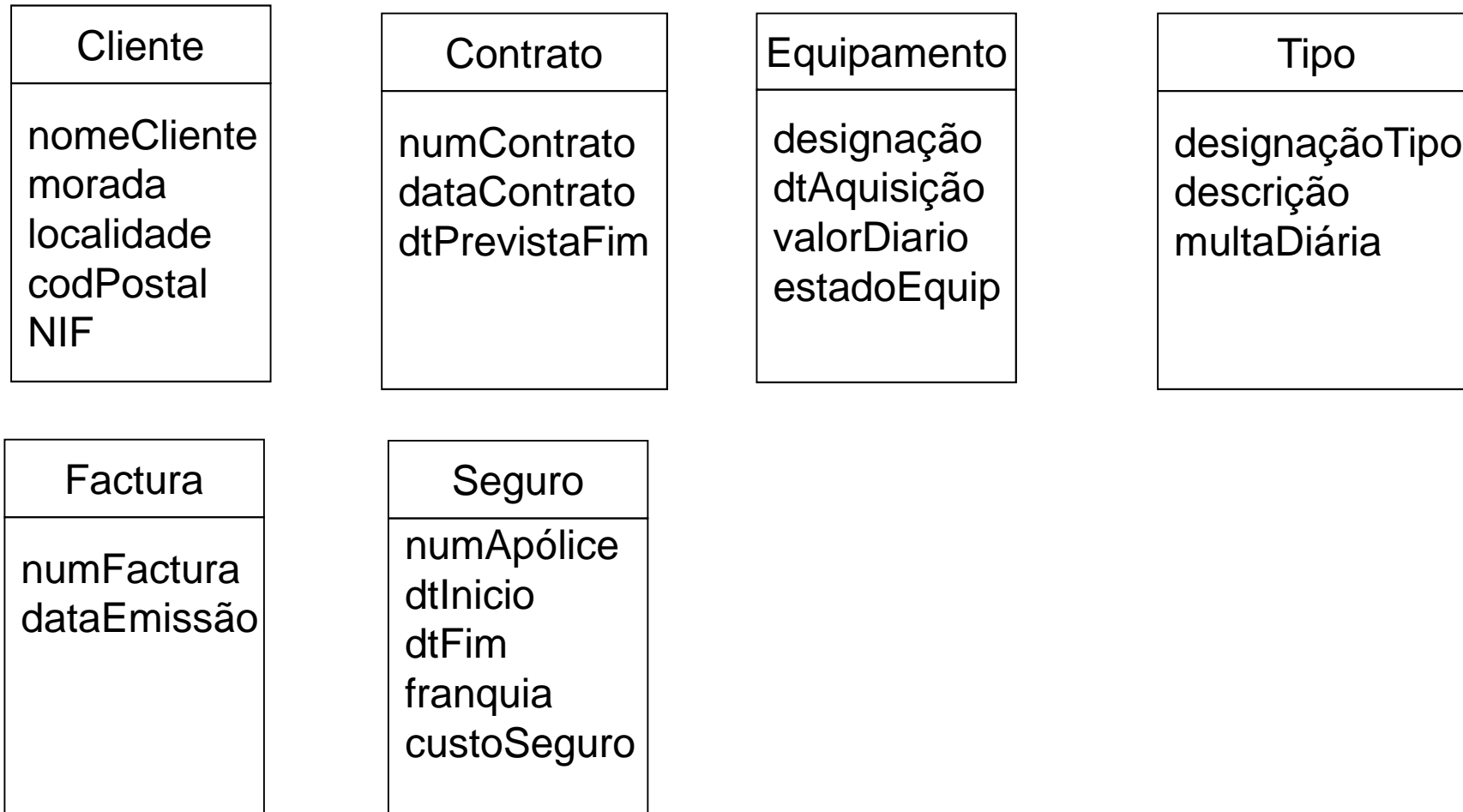


# Caso de Estudo “Aluguer de Equipamento” – Diagrama de Classes (1/2)





# Caso de Estudo “Aluguer de Equipamento” – Diagrama de Classes (2/2)





# Diagrama de Estados

---

- Permite ilustrar o comportamento interno de um determinado objecto, subsistema ou sistema global
- Representa os possíveis estados de um objecto, através de um simulador de estados (máquina de estados);
- São construídos com grafos em que os nós são estados e os arcos transições;



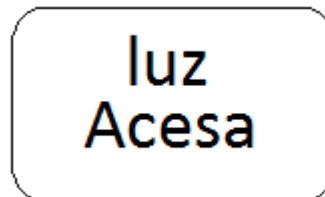


# Diagrama de Estados

---

- Estado

- » situação registada por um objecto durante o seu ciclo de vida, durante a qual uma condição é verificada
- » Cada estado pode ter sub-estados, que são estados dentro de outros estados





# Diagrama de Estados

---

- Transição

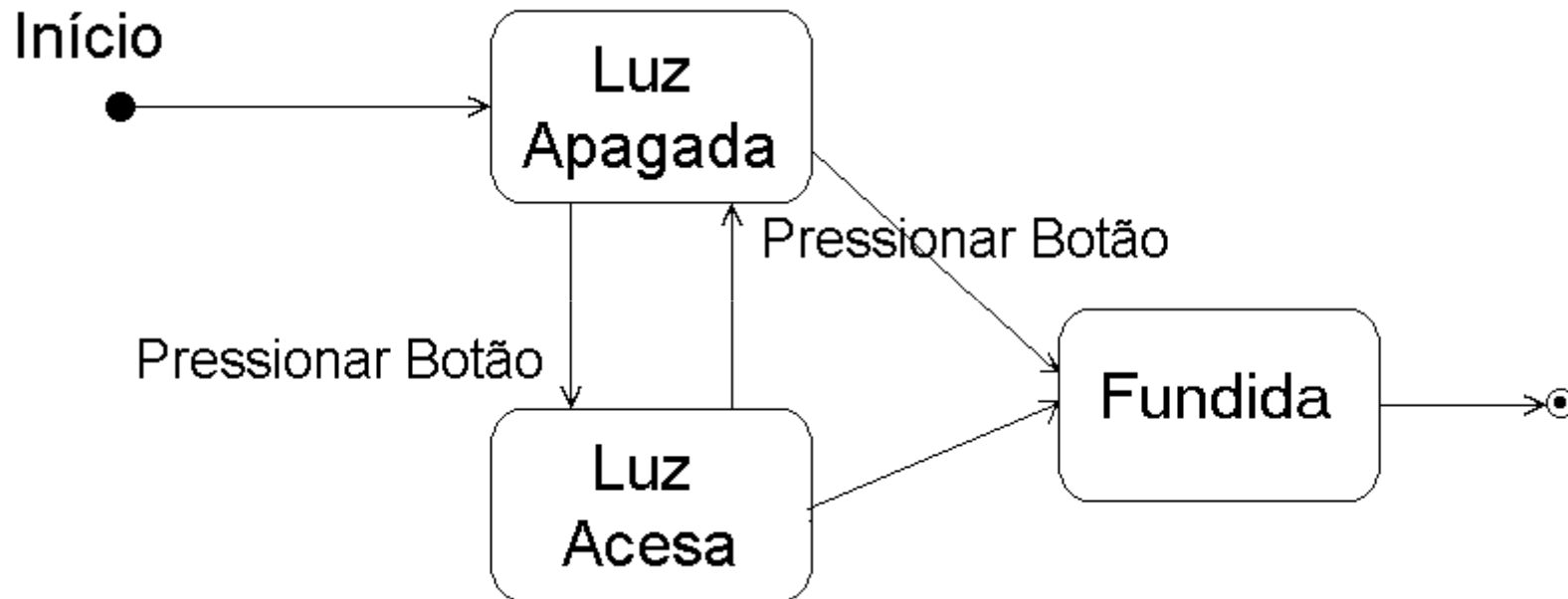
- » relação entre dois estados que especifica que um objecto que se encontre no primeiro estado, realizará um conjunto de acções e mudará para o segundo estado quando um determinado evento ocorrer e determinadas condições se verificarem

- Evento

- » Ocorrência de um estímulo que pode corresponder a uma transição de estado
- » Existem 4 tipos de eventos: sinais, invocação de mensagens, passagem de tempo e satisfação de condição

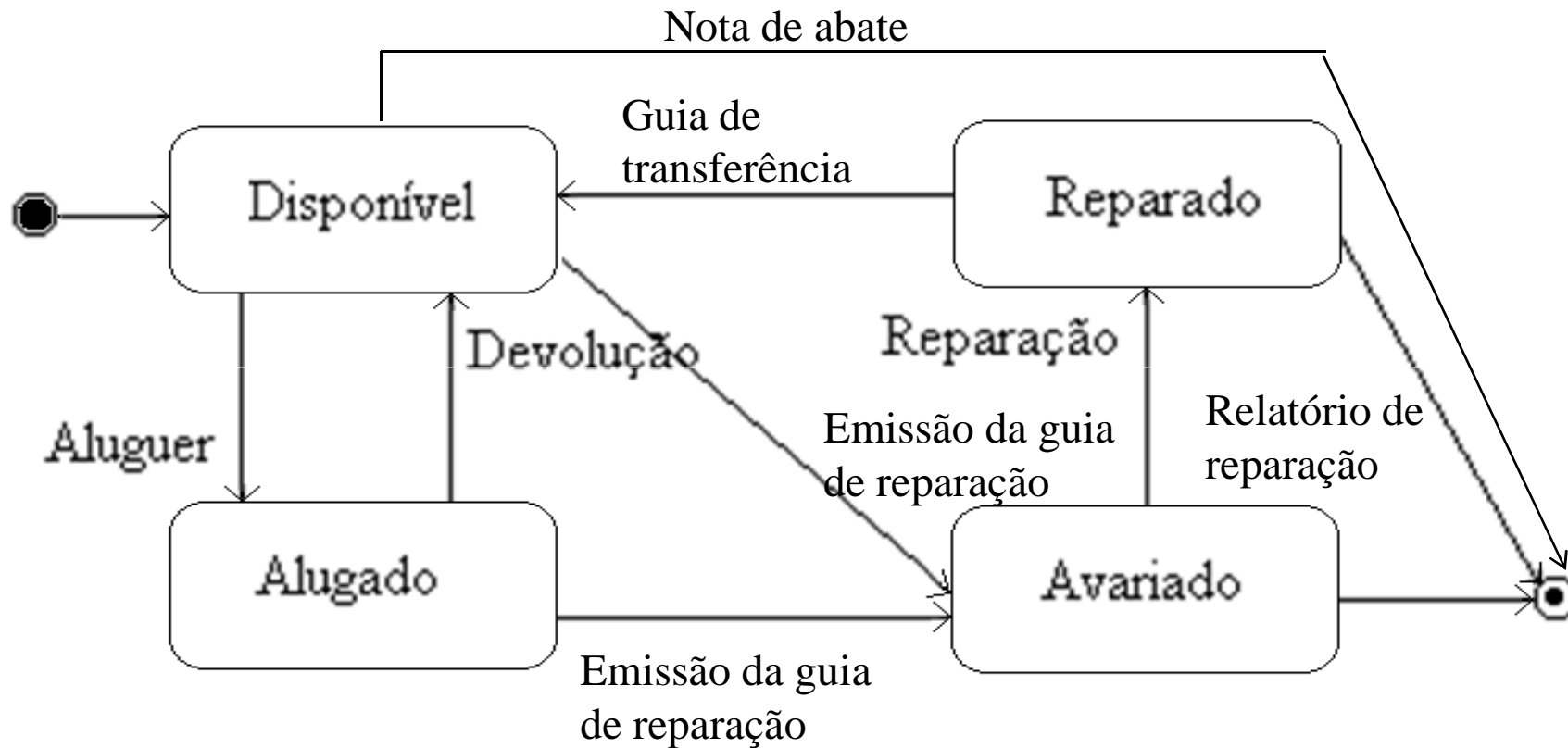


# Exemplo de Diagrama de Estados





# Caso de Estudo “Aluguer de Equipamento” – Diagrama de Estados para a Classe “Equipamento”



O que é que acontece ao diagrama se houver indicação de que um equipamento deverá permanecer no sistema por um período de 5 anos depois de “abatido”?



# Diagrama de Sequência

---

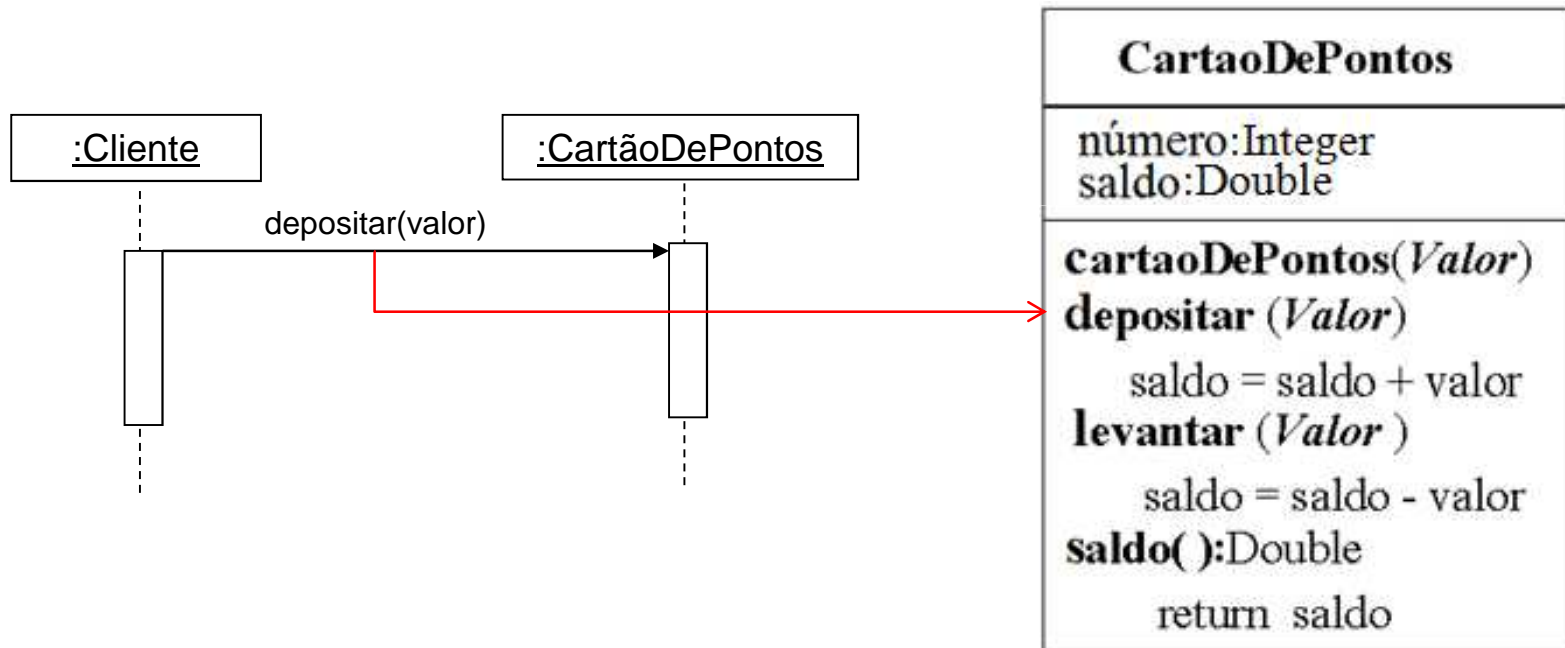
- Normalmente é desenvolvido um para cada cenário
- A representação é feita segundo dois eixos:
  - » no horizontal representam-se os objectos,
  - » no vertical o tempo;
- Mostra as interacções entre objectos ao longo do tempo;
- Os objectos representam-se do mesmo modo que as classes, mas com o nome sublinhado;

:Cliente



# Diagramas de Sequência

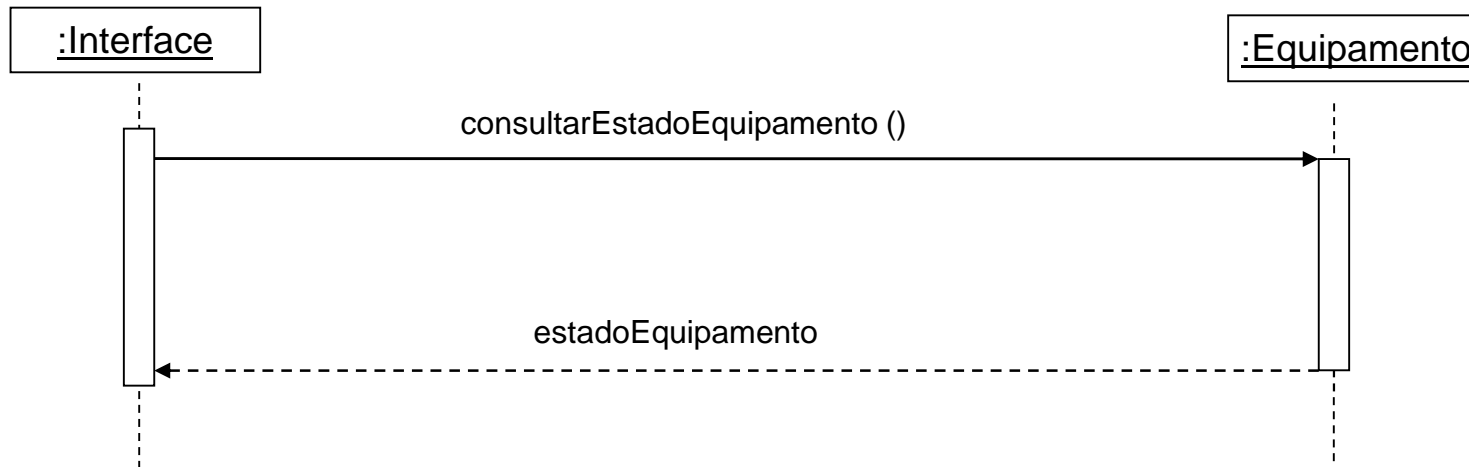
- As interações representadas correspondem a invocações de mensagens nos objectos





# Exemplo de Diagrama de Sequência

## Diagrama de Sequência para “Consultar Estado do Equipamento”





## Notas sobre a elaboração de diagramas

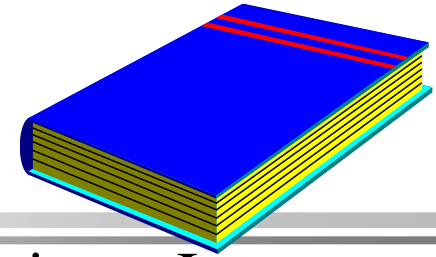
---

- Um diagrama de Casos de Utilização para a totalidade do projecto
- Um diagrama de Actividades para cada um dos casos de utilização
- Um diagrama de Classes – 1ª versão – é efectuado para identificar as classes, relacionamentos e atributos
- Diagrama de Classes final (com os atributos e operações)





# Bibliografia Utilizada



- 1 - Alhir, S., S., *UML in a Nutshell*, O'Reilly & Associates, Inc., USA., 1998.
- 2 - Oestereich, Bernd, *Developing Software With UML*, Addison Wesley, second edition, 2002.
- 3 - Silva, Alberto, Videira, Carlos *UML Metodologias e Ferramentas CASE*, Centro Atlântico, Publishing Ltd, 2004.
- 5 - Rumbaugh, J., Blaha, M., Premerlani. W., Eddy, F., Lorensen, W.; *Object-Oriented Modeling and Design*, Prentice-Hall Inc., USA., 1991.
- 6 - Stevens, P., Pley, R.; *Using UML software Engineering with Objects and Componentes*, Addison Wesley.
- 7 - Rumbaugh, J., Jacobson, I.,Booch, G.; *The UML reference manual*, Addison Wesley, 2004.