
Engenharia de Software

Projecto de
Arquitectura de Software

Luís Morgado

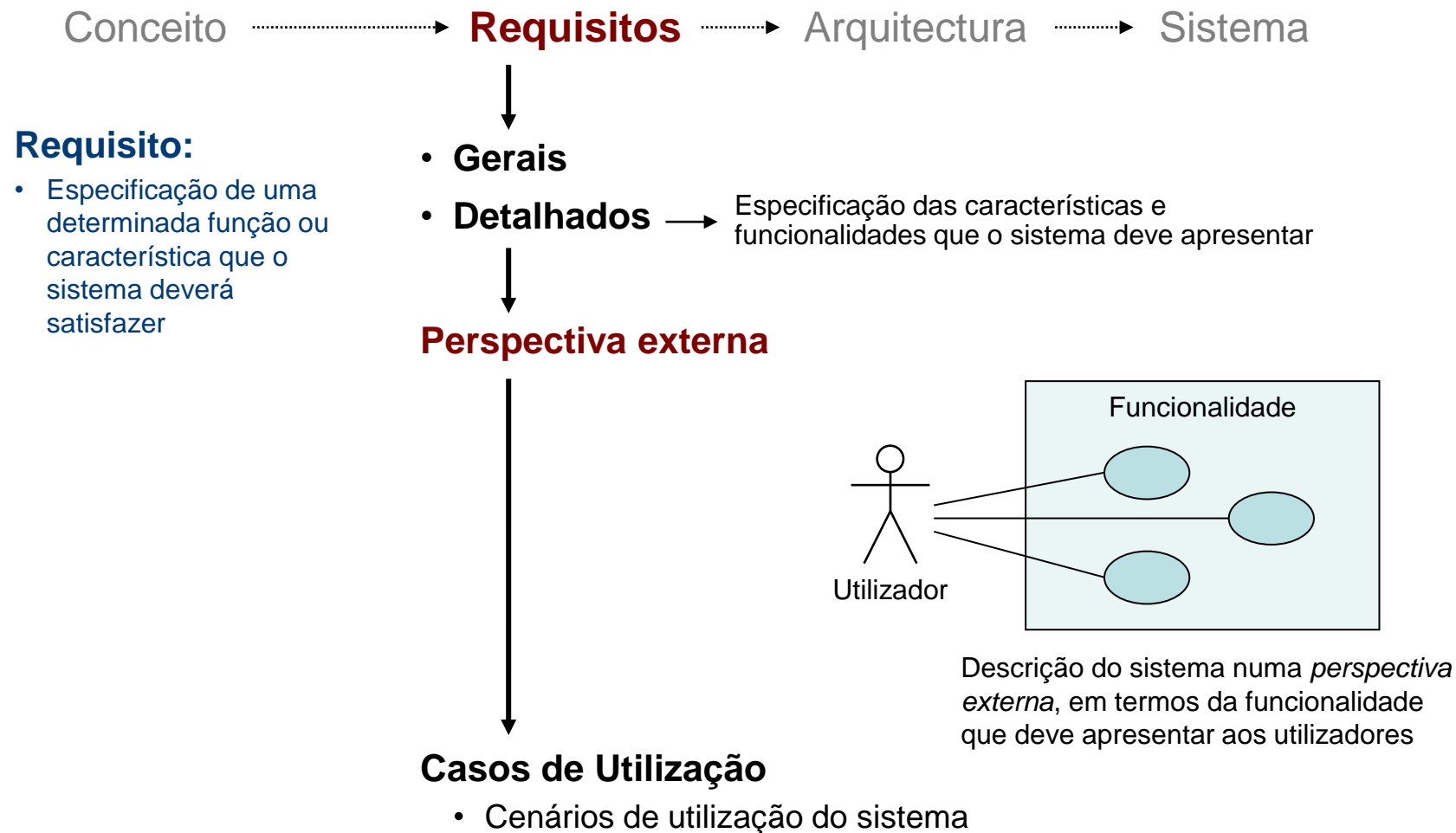
Instituto Superior de Engenharia de Lisboa
Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Processo de Desenvolvimento

- **Análise**
- **Concepção**
 - Transição análise de requisitos – projecto de arquitectura de software
 - Analisar, compreender e verificar a forma de realizar a solução proposta
 - Elaborar a arquitectura da solução
 - Conceber e especificar a arquitectura da solução
- **Construção**
- **Verificação**

Processo de Desenvolvimento

Fase: Análise



Processo de Desenvolvimento

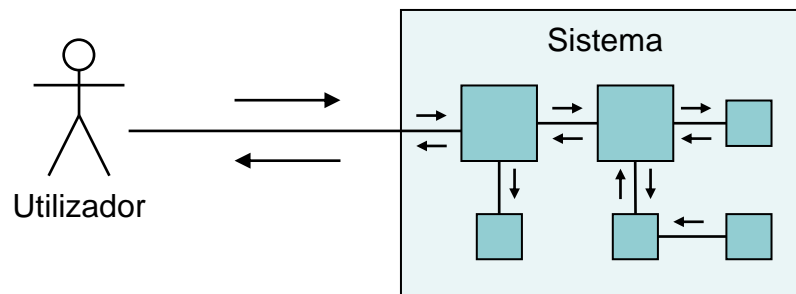
Fase: Concepção

Conceito> Requisitos> **Arquitetura**> Sistema



Perspectiva interna

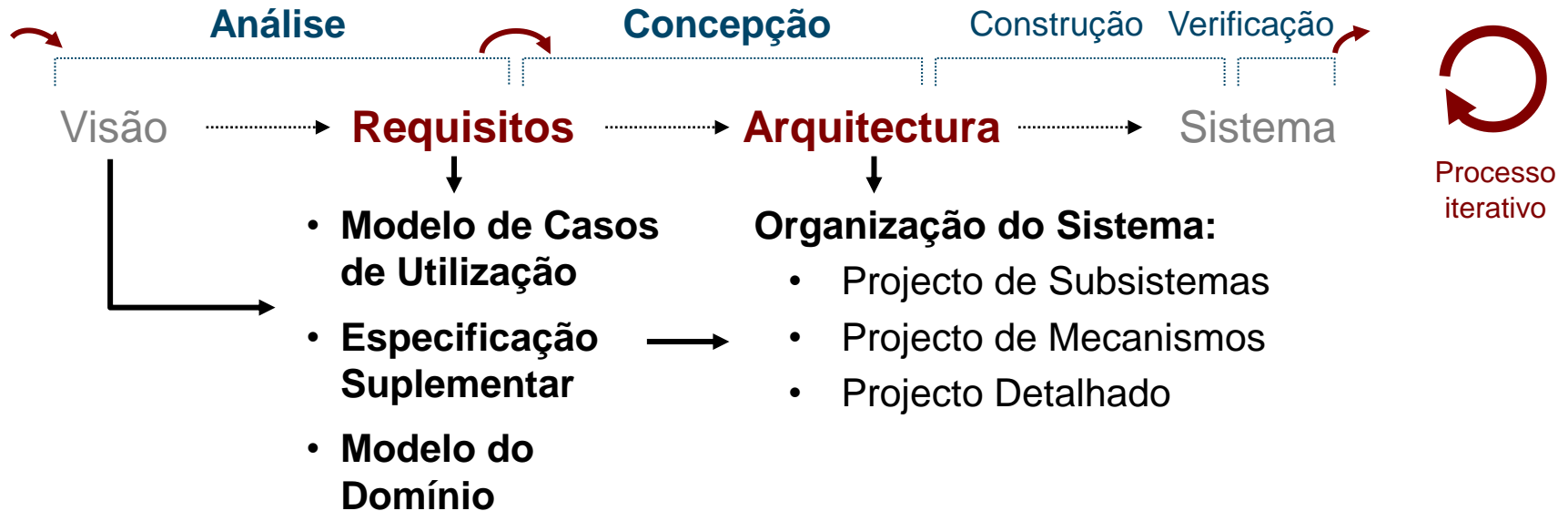
- Que partes constituem o sistema ?
- Como interagem ?



Descrição do sistema numa *perspectiva interna* em termos da organização que realiza a funcionalidade do sistema

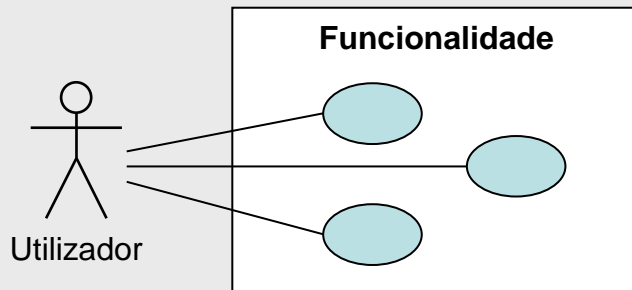
Transição Análise - Concepção

A transição entre a fase de análise e a fase de concepção é determinante para a qualidade e sucesso do desenvolvimento de software, pois realiza a transposição do domínio dos requisitos para o domínio da organização do sistema



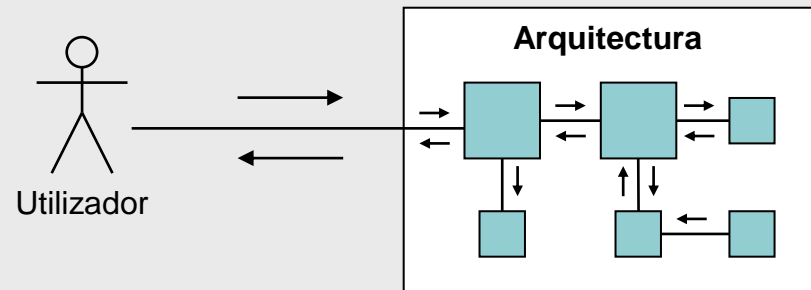
Perspectiva externa

- Especificação das características e funcionalidades que o sistema deve apresentar



Perspectiva interna:

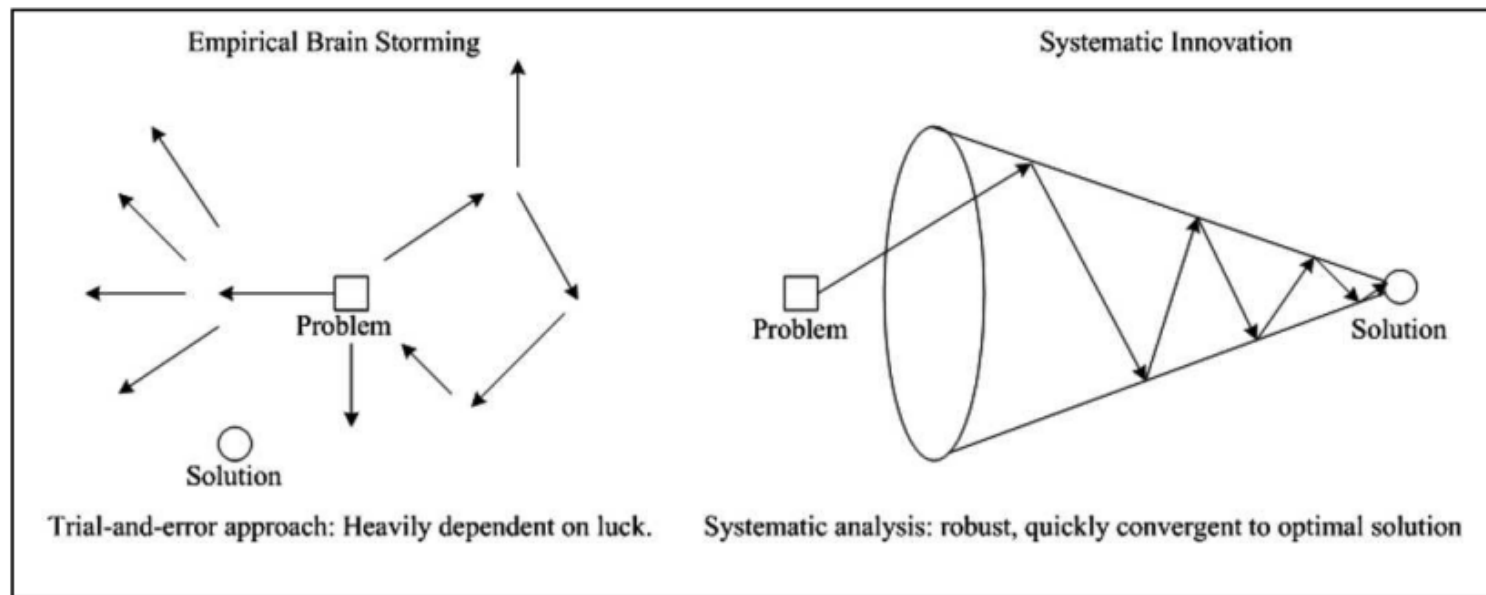
- Que partes constituem o sistema ?
- Como interagem ?



Transição Análise - Concepção

Necessidade de um processo sistemático

Sem um processo sistemático, a transição entre análise e concepção, ocorre sem rumo, sendo impossível garantir qualidade ou o atingir do que se pretende, a complexidade e a desorganização predominam, originando um elevado esforço e risco de desenvolvimento



[Matias *et al.*, 2011]

Processo baseado na Tentativa – Erro

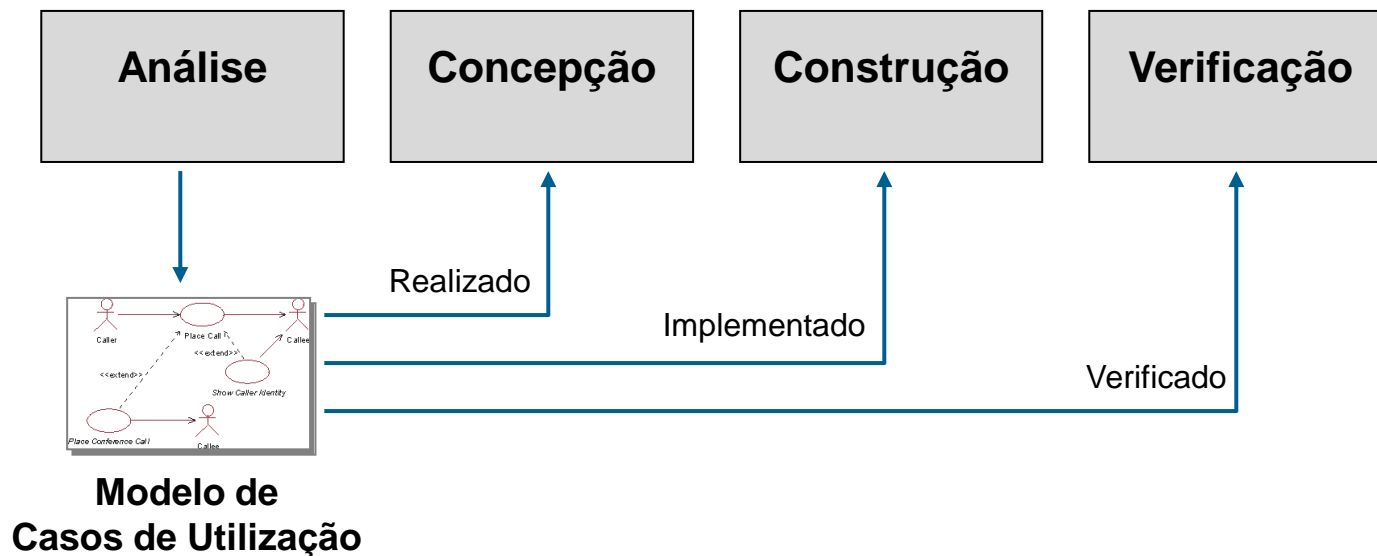
- Parte de uma ideia vaga do que é pretendido
- Ausência de método sistemático de trabalho
- Reagir, corrigir, remediar são a regra
- Caos, ineficiência

Processo sistemático

- Conceber e construir a solução de forma iterativa e incremental, tendo por base o conhecimento obtido na fase de análise e descrito na especificação de requisitos
- Convergir para a solução pretendida de forma guiada e controlada

Processo de Desenvolvimento

Processo de desenvolvimento guiado pela funcionalidade que o sistema deve apresentar aos utilizadores, descrita sob a forma de cenários de utilização, organizados em histórias de utilização ou, de forma mais abrangente, em casos de utilização



Os **casos de utilização** constituem um suporte organizado e sistemático, para a realização das diferentes fases de desenvolvimento de um sistema

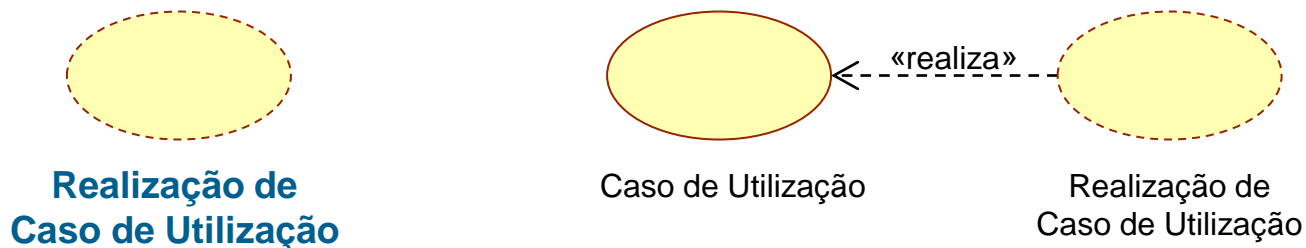
Transição Análise - Concepção

Na actividade de concepção é elaborada a arquitectura do sistema de modo a *realizar* a funcionalidade especificada no modelo de casos de utilização

- **Realização de um caso de utilização**

- Define uma forma de organizar um conjunto de partes (uma *colaboração* ou *mecanismo*) que mostra como a arquitectura do sistema suporta o comportamento definido num caso de utilização

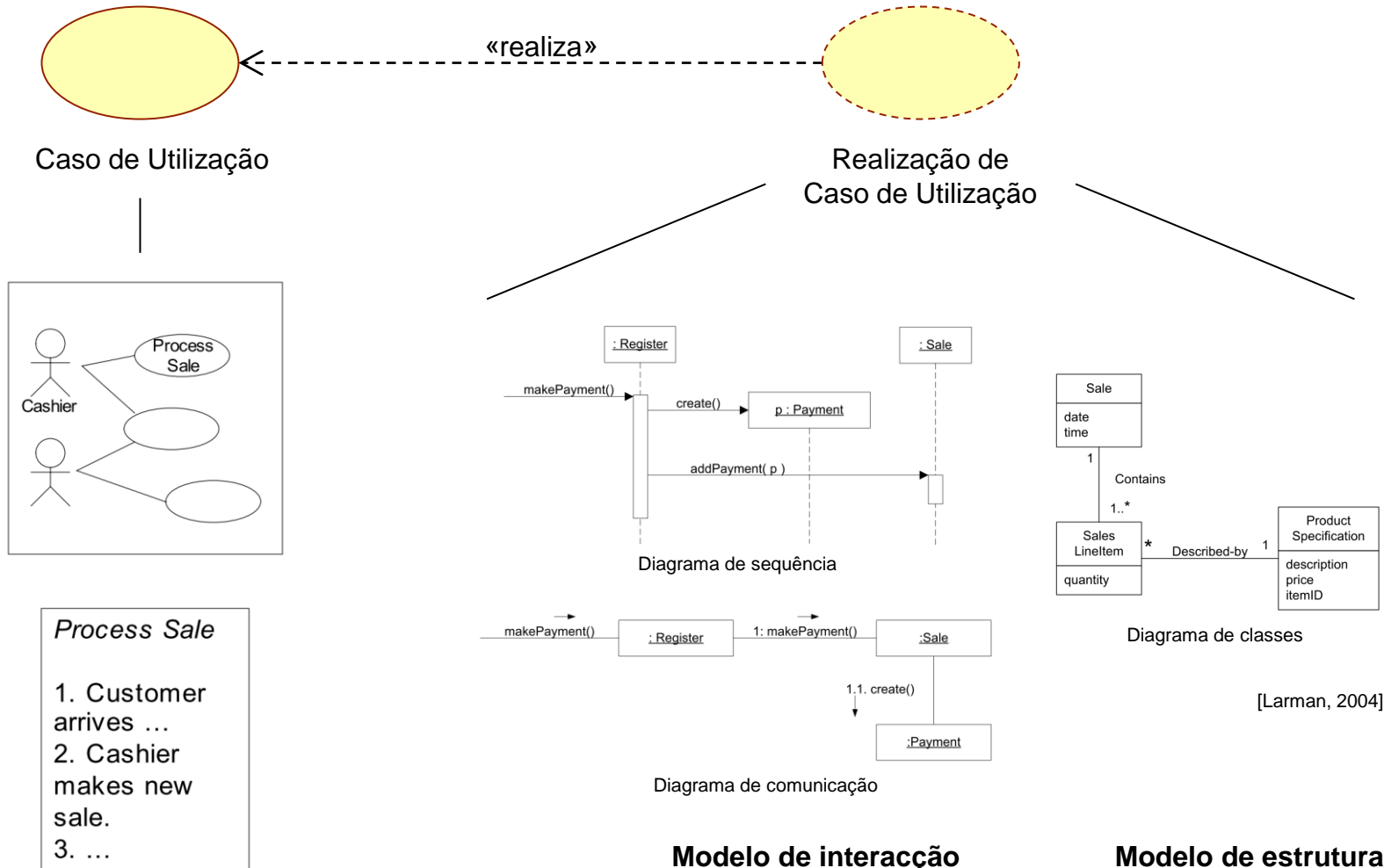
Notação UML:



Realização de Casos de Utilização

Realização de um caso de utilização

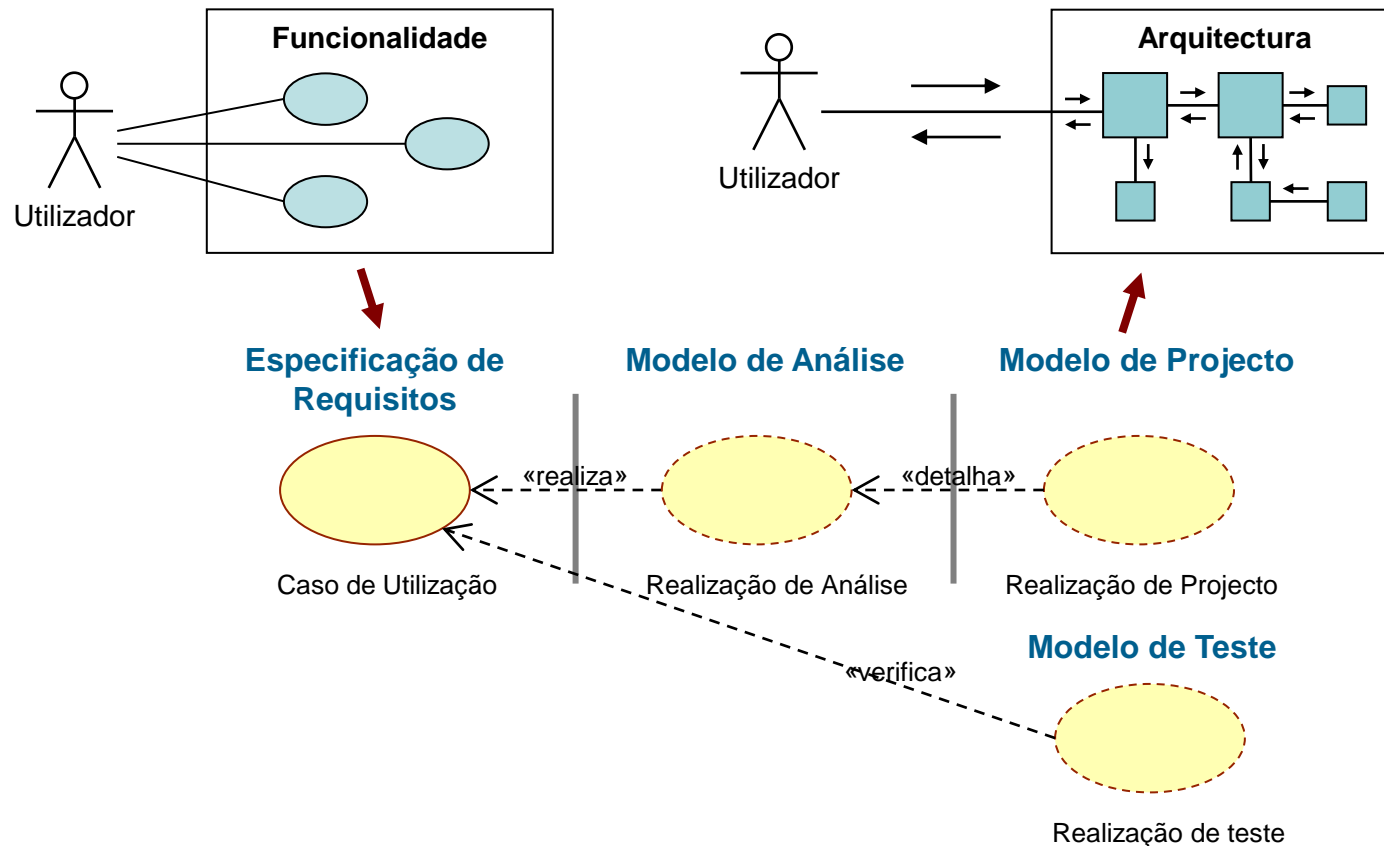
Define uma forma de organizar um conjunto de partes (uma colaboração ou mecanismo) que mostra como a arquitectura de um sistema suporta o comportamento definido num caso de utilização



Realização de Casos de Utilização

A realização de um caso de utilização pode ser definida a 3 níveis

- **Modelo de análise** (da arquitectura)
 - Definição de um mecanismo conceptual que realiza de forma abstracta a funcionalidade descrita na especificação de requisitos
- **Modelo de projecto**
 - Definição de um mecanismo concreto mas independente da tecnologia de implementação
- **Modelo de teste**
 - Definição de um mecanismo de teste da funcionalidade respectiva



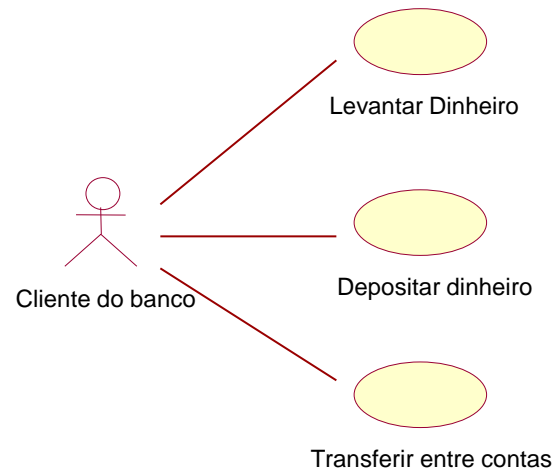
Realização de Casos de Utilização

Exemplo

Sistema ATM (levantamento automático de dinheiro)

Descrição:

O cliente do banco utiliza um sistema ATM para levantar e depositar dinheiro das contas e para transferir dinheiro entre contas.

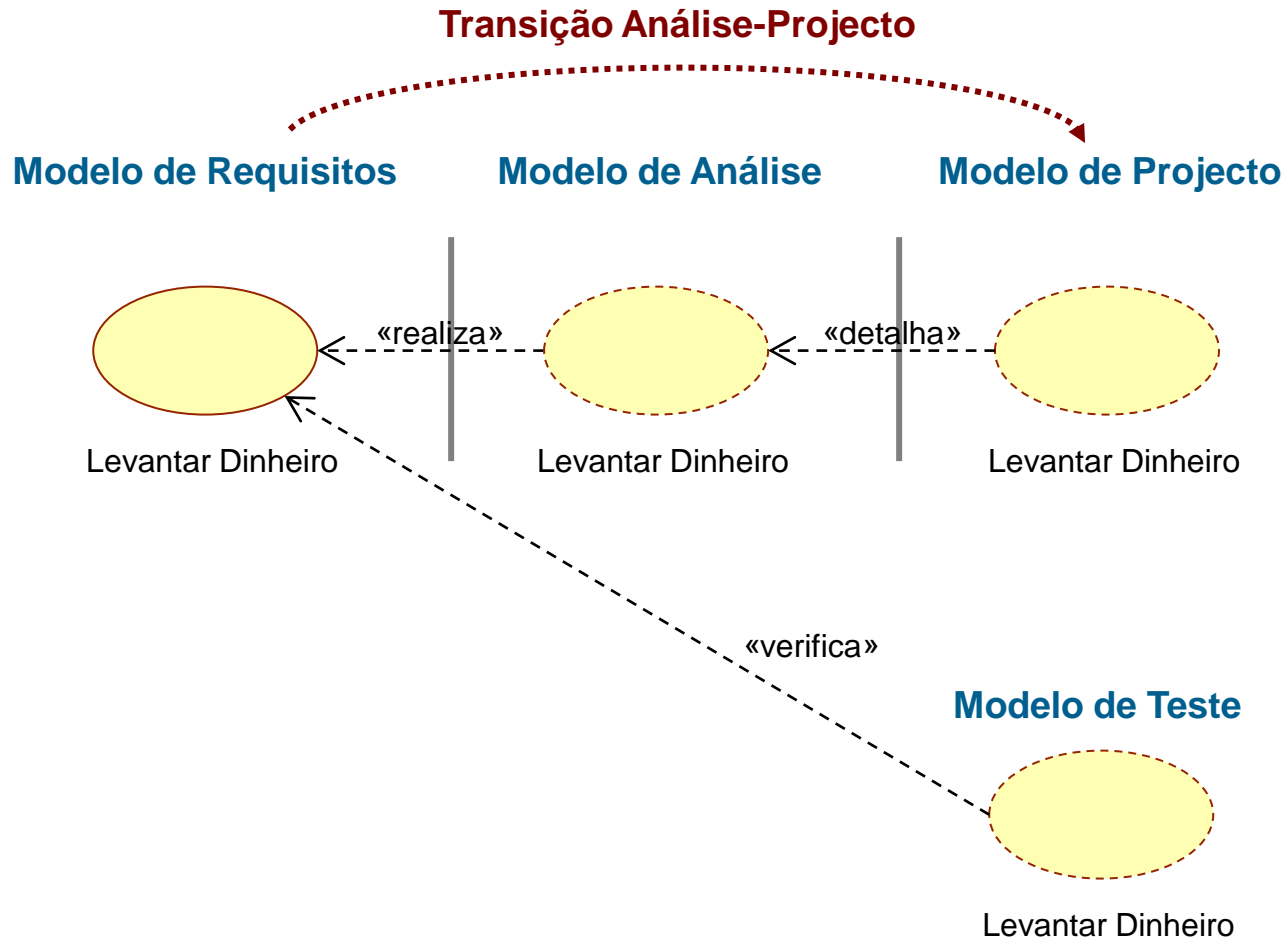


[Rational 2003]

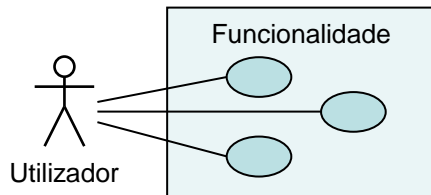
Realização de Casos de Utilização

Exemplo

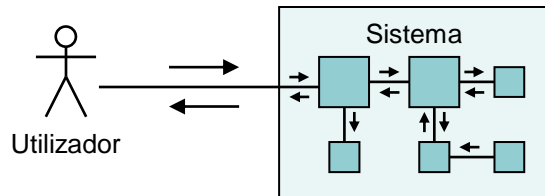
Sistema ATM (levantamento automático de dinheiro)



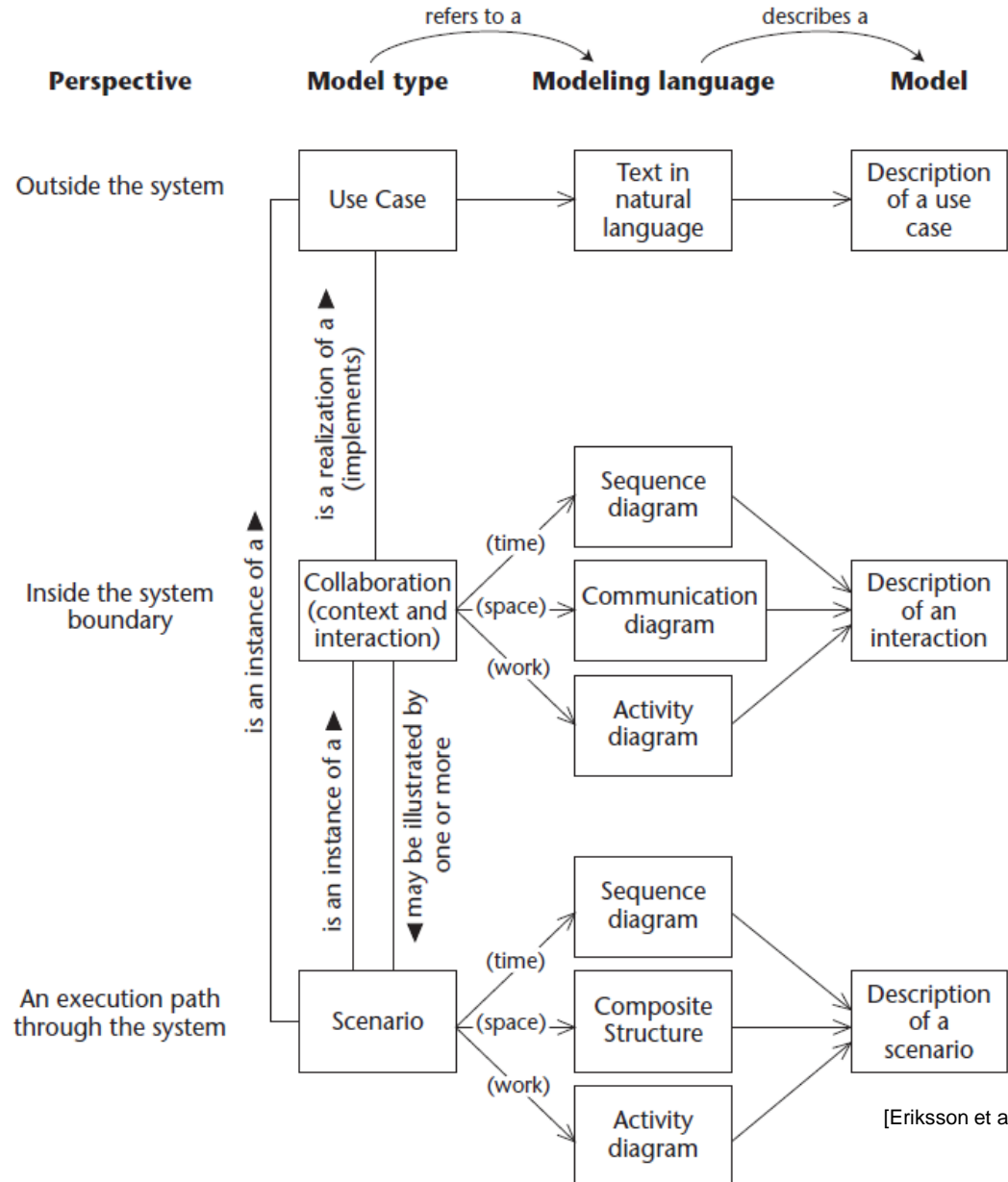
Transição Análise - Concepção



Descrição do sistema numa *perspectiva externa*, em termos da funcionalidade que deve apresentar aos utilizadores, descrita em linguagem natural sob a forma de cenários de utilização



Descrição do sistema numa *perspectiva interna* em termos da organização que realiza a funcionalidade do sistema, tendo por base as interações entre partes para realizar os cenários definidos nos casos de utilização



[Eriksson et al., 2004]

Modelo de Análise

- Modelo que **descreve a realização dos casos de utilização de forma abstracta**, tendo por base classes de análise
 - Omite detalhes das partes e da interacção entre partes
- Suporta a transição Análise – Concepção
 - Serve de base para a elaboração do modelo de projecto
 - Modelo transitório
- Após a sua finalização deve ser feita uma revisão preliminar de projecto
 - Objectivo:
 - Garantir que a especificação de requisitos e o modelo de análise são consistentes

Elaboração do Modelo de Análise

- Identificar objectos e classes
 - Modelo de domínio
- Elaborar realizações de casos de utilização
 - Modelo de interacção
- Identificar associações entre objectos
- Identificar atributos de objectos (nível abstracto)
- Organizar e simplificar modelo
 - Herança (factorização)
- Agrupar classes em módulos e subsistemas

Estereótipos de Análise

Constituem heurísticas que guiam o processo de identificação das partes envolvidas na realização da funcionalidade de um sistema

- **Fronteira (*Boundary*)**

- Utilizado para modelação da interacção entre o sistema e o ambiente operacional exterior (utilizadores e outros sistemas)



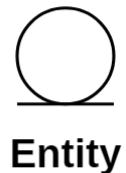
- **Controlo (*Control*)**

- Utilizado para modelação de aspectos de coordenação de acção e controlo de outras partes (e.g. sequenciação, controlo transaccional)
- Realizam a ligação entre os objectos de fronteira e entidade



- **Entidade (*Entity*)**

- Utilizado para modelação de informação persistente
- Entidades do modelo de domínio

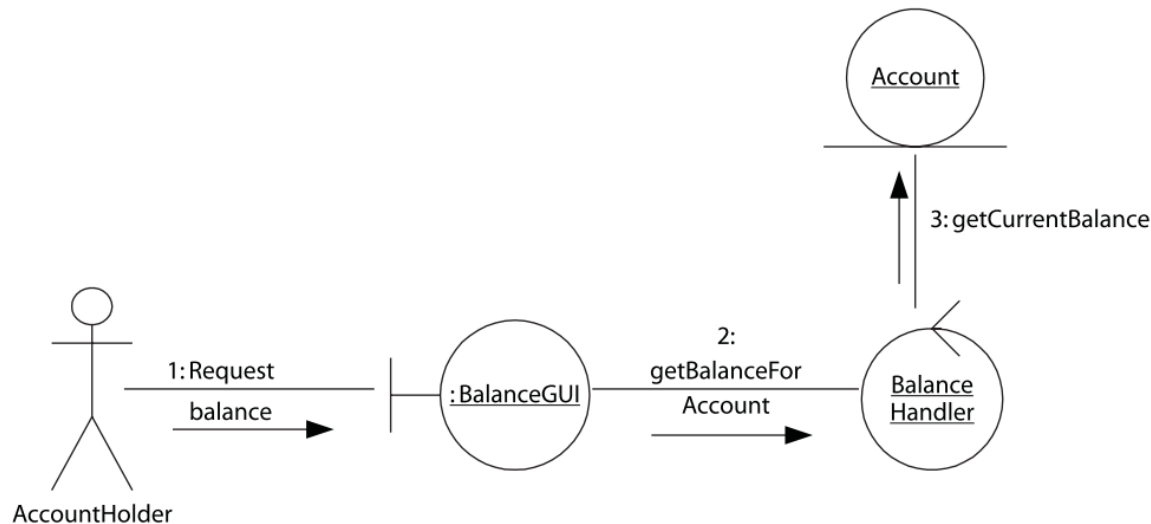


Padrão Entidade-Controllo-Fronteira

O padrão **Entidade-Controllo-Fronteira** (ECB – *Entity-Control-Boundary*) é um padrão de arquitectura de software utilizado na realização de casos de utilização, que orienta a identificação das partes necessárias de acordo com as suas responsabilidades na realização do caso de utilização, em três tipos principais: *entidade*, *controlo* e *fronteira*

Exemplo

Sistema bancário



Modelo de Análise

- A necessidade do modelo de análise varia em função do conhecimento acerca do domínio do problema e do domínio da solução
 - Forma exploratória de **aumentar a compreensão do problema** e de **definir uma visão da solução**
 - Quando o modelo de projecto evolui, o modelo de análise vai ficando desactualizado
 - A sua utilidade diminui
 - Para ser útil deve ser mantido actualizado
 - Necessário ter em conta o esforço necessário para manter o modelo de análise actualizado à medida que o projecto evolui
 - Modelo transitório

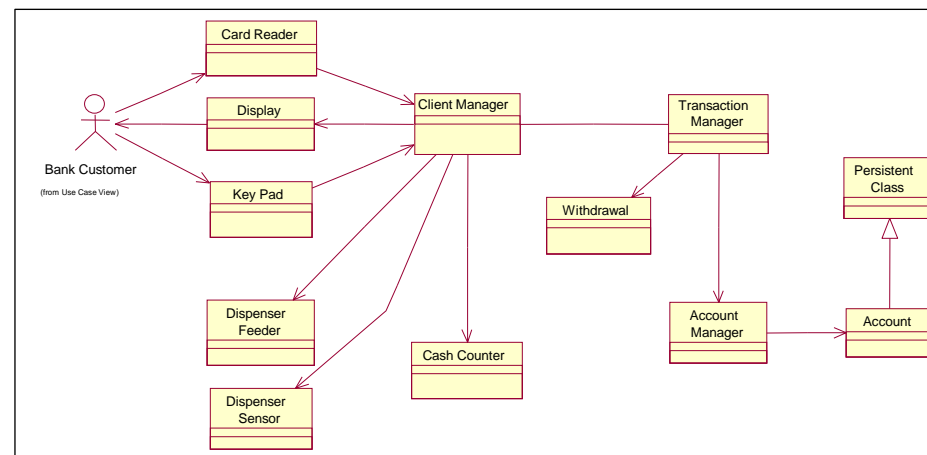
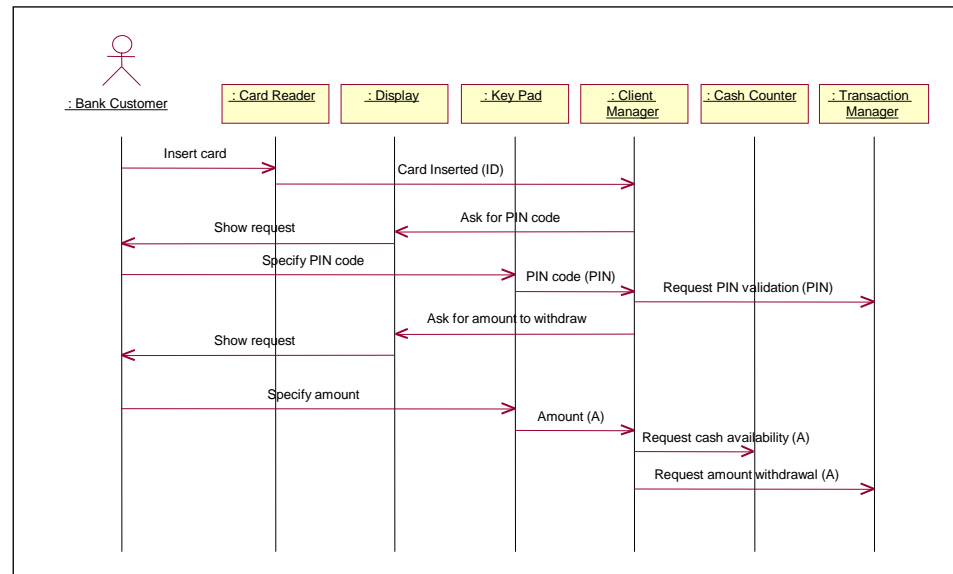
Modelo de Projecto

- Modelo que **descreve a realização dos casos de utilização em termos da arquitectura do sistema** a diferentes níveis de abstracção
- Serve de base para a definição dos modelos de implementação e de teste
- Elaborado de forma iterativa
 - Serve de base para refinar a análise
- Considera-se **correcto quando realiza a especificação de requisitos**
- Foco na arquitectura

Modelo de Projecto

No modelo de projecto os *mecanismos* que realizam os casos de utilização são detalhados com partes e interacções progressivamente mais concretas

Exemplo Sistema ATM



Modelo de Projecto

O modelo de projecto descreve a arquitectura de um sistema em diferentes níveis de arquitectura com detalhe progressivo

Níveis de arquitectura

- **Subsistemas**

- Agregados de mecanismos
- Funcionalidade global

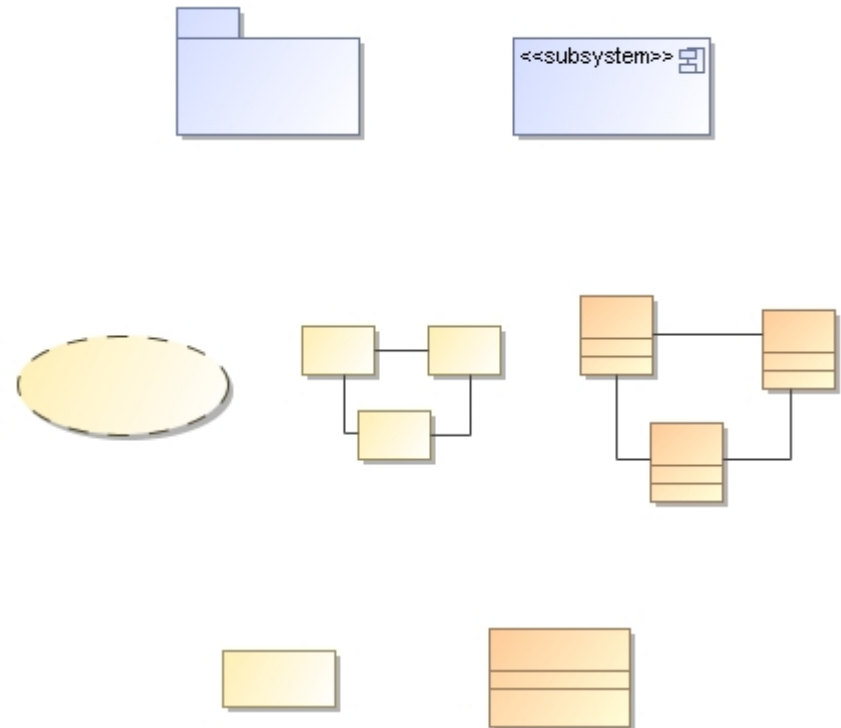
- **Mecanismos**

- Realização de casos de utilização
- Agregados de elementos
- Funcionalidade local

- **Elementos**

- Partes base
- Funcionalidade micro

UML



Níveis de Projecto de Arquitectura

- **Projecto de Subsistemas**

Âmbito: Subsistemas

O que é especificado:

- Organização do sistema como um todo em termos de subsistemas
- Padrões de arquitectura de subsistemas

Âmbito

Sistema

- **Projecto de Mecanismos**

Âmbito: Grupos de classes

O que é especificado:

- Organização de mecanismos em termos das classes que os constituem para realizar a respectiva funcionalidade
- Padrões de arquitectura de mecanismos

Inter-objecto

- **Projecto Detalhado**

Âmbito: Classes, Interfaces

O que é especificado:

- Detalhes de implementação de atributos e operações
- Definição de aspectos específicos de implementação, por exemplo, algoritmos, mecanismos concretos de persistência de dados

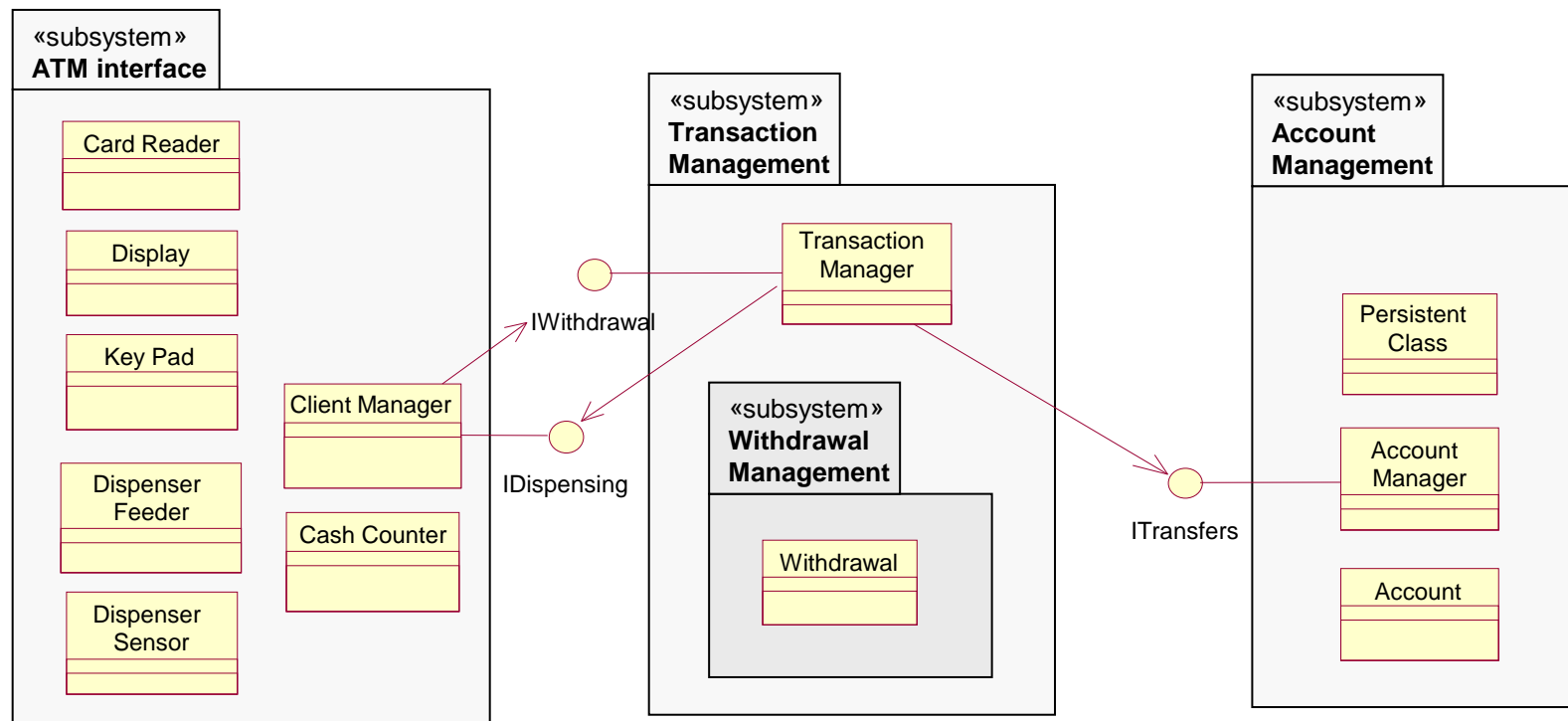
Intra-objecto

Modelo de Projecto

No modelo de projecto, as partes que constituem um sistema são organizadas em mecanismos, os vários mecanismos são organizados e encapsulados de forma modular e agregados em subsistemas coesos

Exemplo

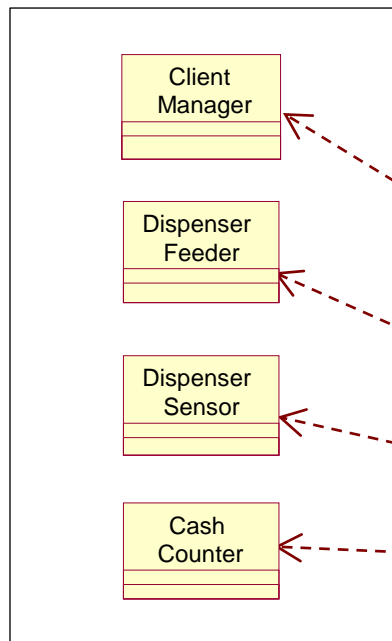
Sistema ATM



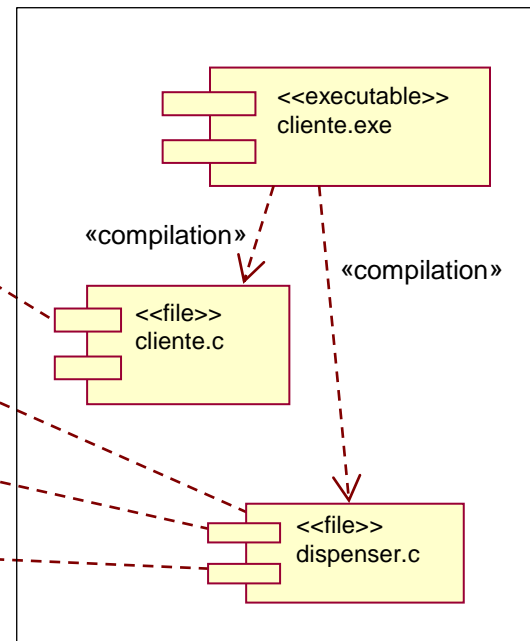
Modelo de Implementação

O modelo de implementação define a organização do sistema em termos dos vários componentes que constituem a implementação do sistema, ou seja, que implementam as várias partes lógicas do sistema, definidas no modelo de projecto

Modelo de Projecto



Modelo de Implementação



[Rational 2003]

Modelo de Domínio

- O modelo de domínio consiste numa representação das classes e relações conceptuais que descrevem aspectos principais de um determinado domínio, com foco nos conceitos do domínio e respectivas relações
- O modelo de domínio pode ser elaborado em conjunto com a especificação de requisitos, permitindo identificar conceitos relevantes para a descrição dos cenários dos casos de utilização
- É utilizado na realização de casos de utilização e no restante projecto de arquitectura do sistema para identificação de partes que constituem o sistema e para a elaboração da arquitectura de dados

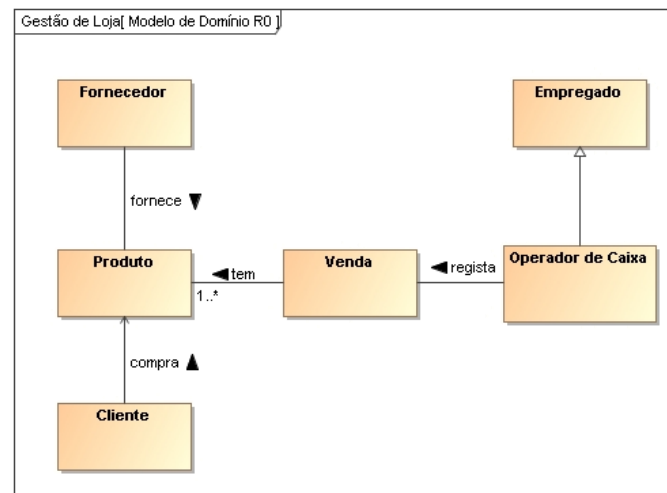
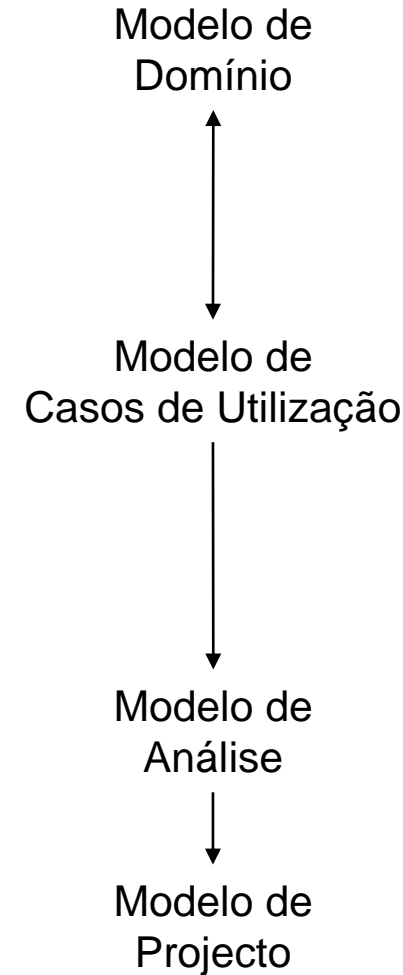
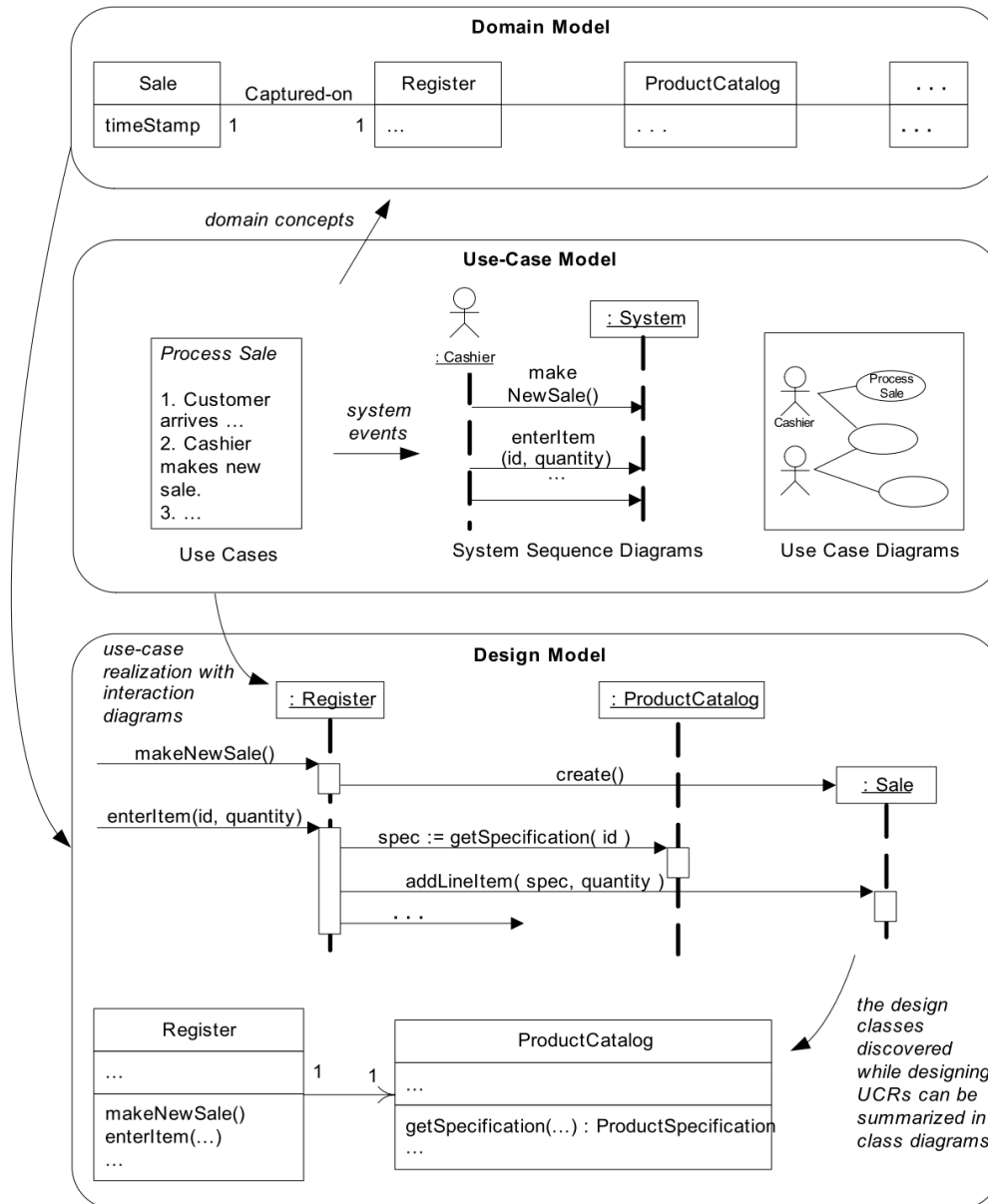


Diagrama de classes conceptual de um modelo de domínio

Projecto de Arquitectura de Software



Bibliografia

[Skyttner, 2001]

L. Skyttner, *General Systems Theory*, World Scientific, 2001.

[Pressman, 2003]

R. Pressman, *Software Engineering: a Practitioner's Approach*, McGraw-Hill, 2003.

[Shaw & Garlan, 1996]

M. Shaw, D. Garlan, *Software Architecture: Perspectives on an Emerging Discipline*, Prentice-Hall, 1996.

[Larman, 2004]

C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, Prentice Hall, 2004.

[Matias *et al.*, 2011]

Matias, João, Dias, Ana, Abreu, Antonio, Matias, J.O.C., Dias, J., *Systematic Innovation in Engineering; Triz and Non-triz Tools*, International Conference on Engineering, UBI, 2011.