



UML > VISÃO GERAL E DIAGRAMAS DE ATIVIDADES

MODELAÇÃO E ANÁLISE DE SISTEMAS

ILÍDIO OLIVEIRA ico@ua.pt
v2021-10-15

Objetivos de aprendizagem

Justificar a utilização de modelos no desenvolvimento de software (benefícios)

Justificar as vantagens dos modelos visuais. Distinguir dos modelos dinâmicos e estruturais (em UML).

Explicar a semântica dos elementos de modelação mais comuns no Diagrama de Actividades

Interpretar (ler) D. A.

Construir um D. A. para visualizar um processo de negócio enunciado.

Utilizar fluxos de controlo e de dados; utilizar eventos e partições.

Enumerar situações do SDLC em que a utilização de D.A. é adequada.



PAPEL DOS MODELOS NO DESENVOLVIMENTO DE SW

Usamos **modelos visuais para captar partes do mundo/realidade**

D Trumpet Version

Allegro Assai
from
Brandenburg Concerto #2

J. S. Bach
arranged by Mark Adler

Allegro assai
tr
mf

Allegro assai
mf



Uma linguagem comum (escrever, ler)

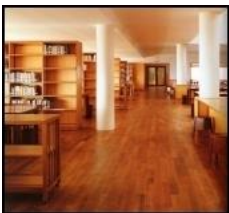
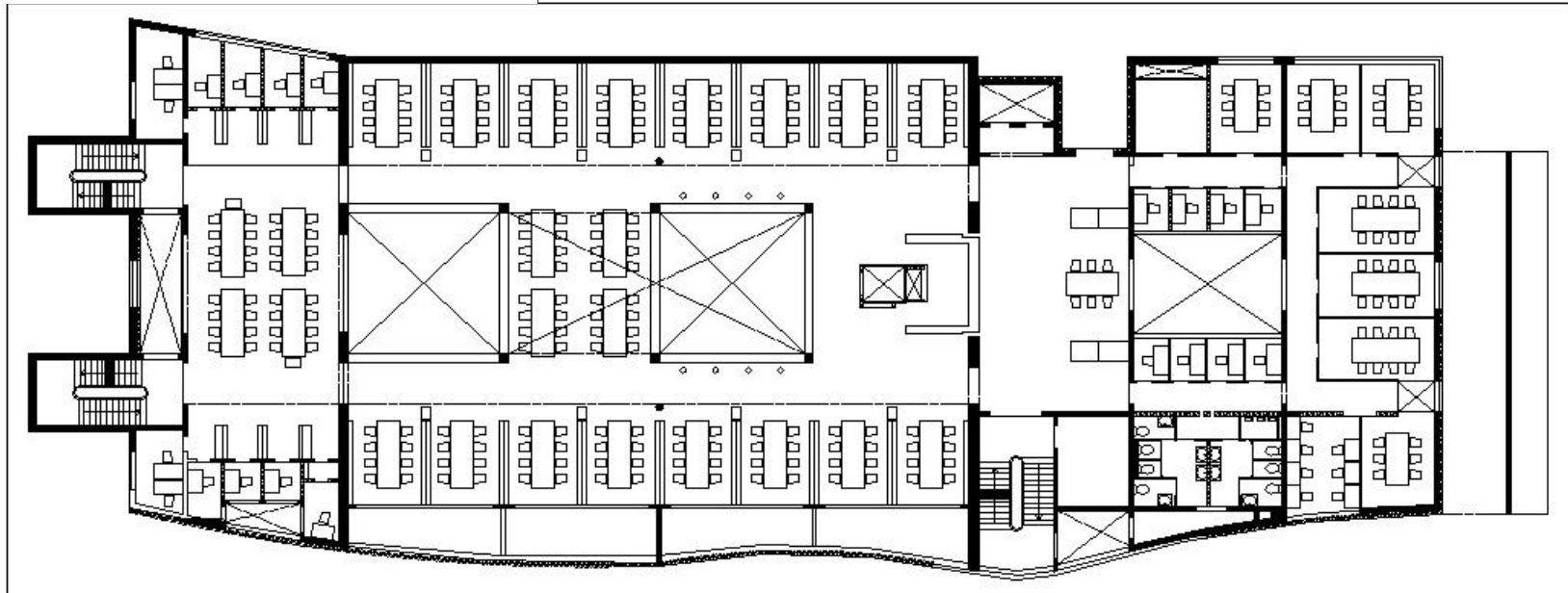
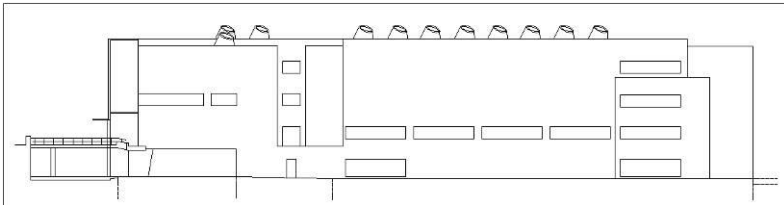
Especificações visuais são mais ricas a mostrar relações entre os conceitos

Compor: aplicar talento e disciplinas técnicas

Orquestra: a prova que os modelos funcionam!



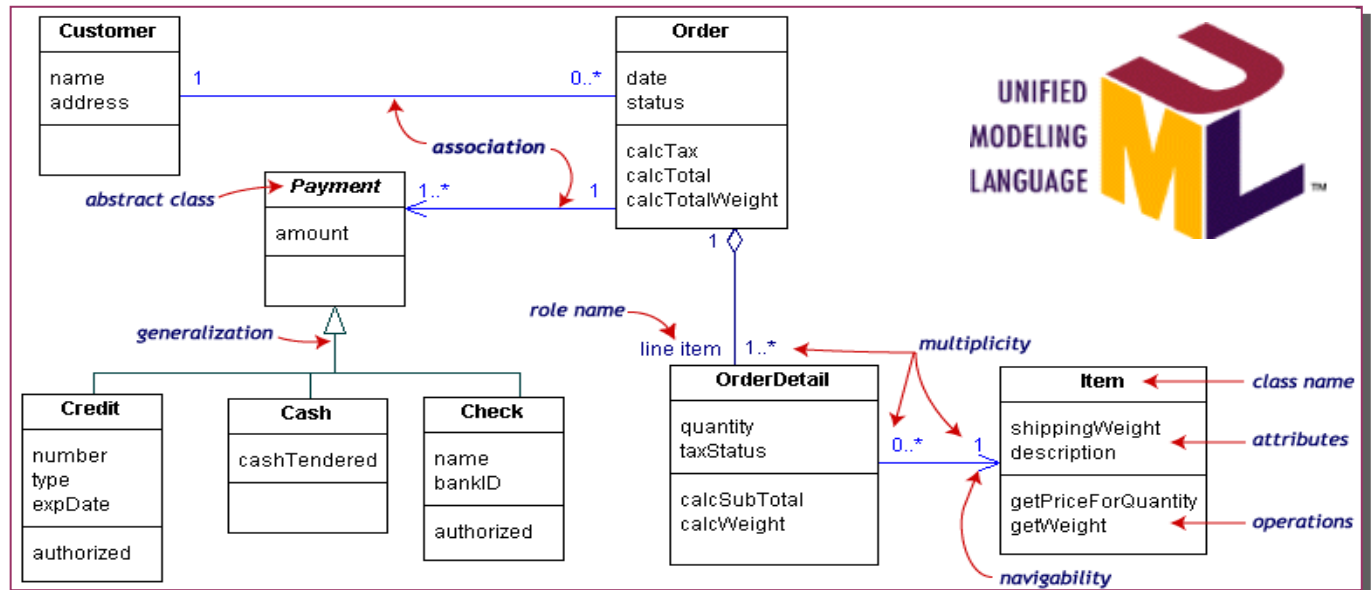
Um modelo é uma simplificação da realidade



Os modelos ajudam a **gerir a complexidade**

Grady Booch apresenta 4 razões para usar modelos:

- Ajudar a **visualizar um sistema**, como se pretende que venha a ser
- Especificar a **estrutura e o comportamento** do sistema (antes de implementar)
- Serve como **referência / orientação** para a construção ("planta")
- **Documentar** as decisões (de desenho) que foram feitas



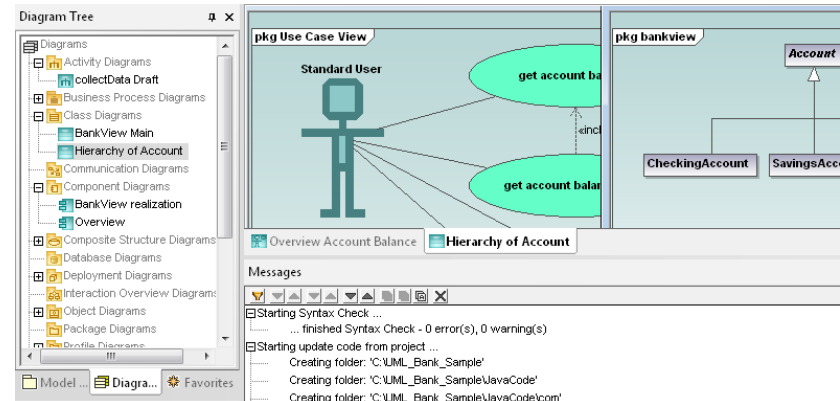
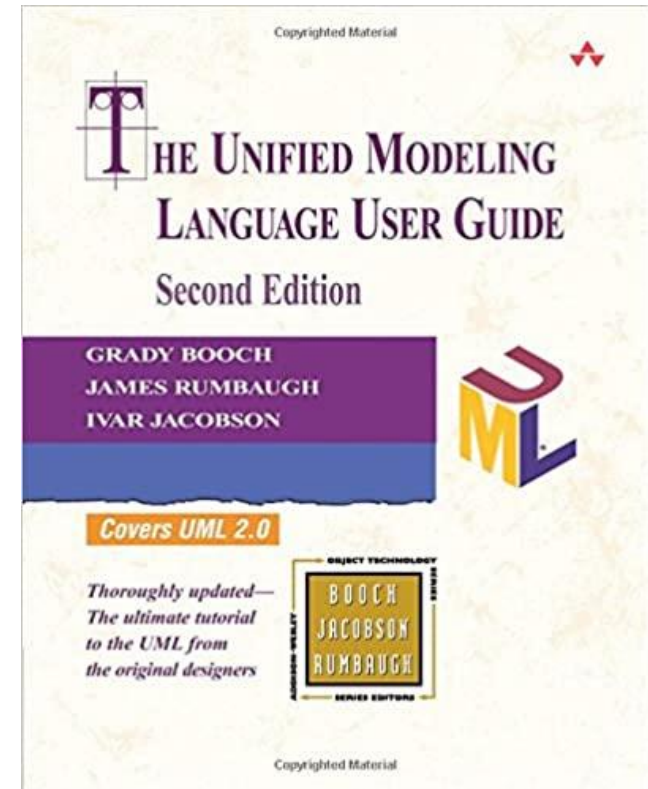
Modelação visual no desenvolvimento

UML 2: Unified Modeling Language

Linguagem de modelação normalizada, orientada para a área de eng.a de software

Benefícios

- Promover a comunicação mais clara e sucinta
- Manter o desenho (planeamento) e a implementação (construção) coerentes
- Mostrar ou esconder diferentes níveis de detalhe, conforme apropriado
- Pode suportar, em parte, processos de construção automática (gerar a solução a partir do modelo)



Também reconhecida como um standard internacional ISO



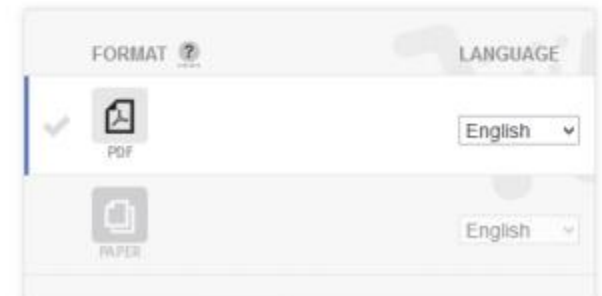
ISO/IEC 19505-1:2012[®]

Information technology -- Object Management Group Unified Modeling Language (OMG UML) -- Part 1: Infrastructure

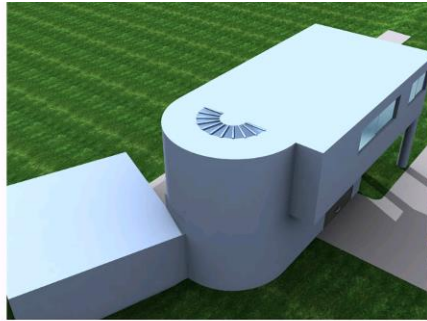
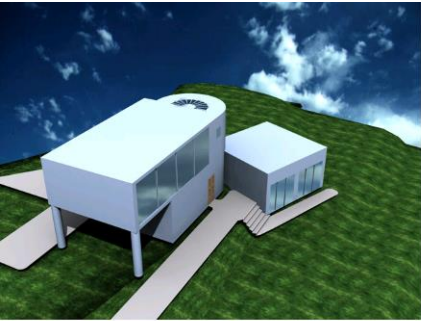
Abstract

[Preview ISO/IEC 19505-1:2012](#)

ISO/IEC 19505-1:2012 defines the Unified Modeling Language (UML), revision 2. The objective of UML is to provide system architects, software engineers, and software developers with tools for analysis, design, and implementation of software-based systems as well as for modeling business and similar processes.



Não há uma vista única, mas várias e complementares para descrever um sistema



Para que serve o sistema?

→ análise funcional

Quais são as estruturas de informação?

→ Análise das estruturas de dados

Decomposição funcional de atividades complexas

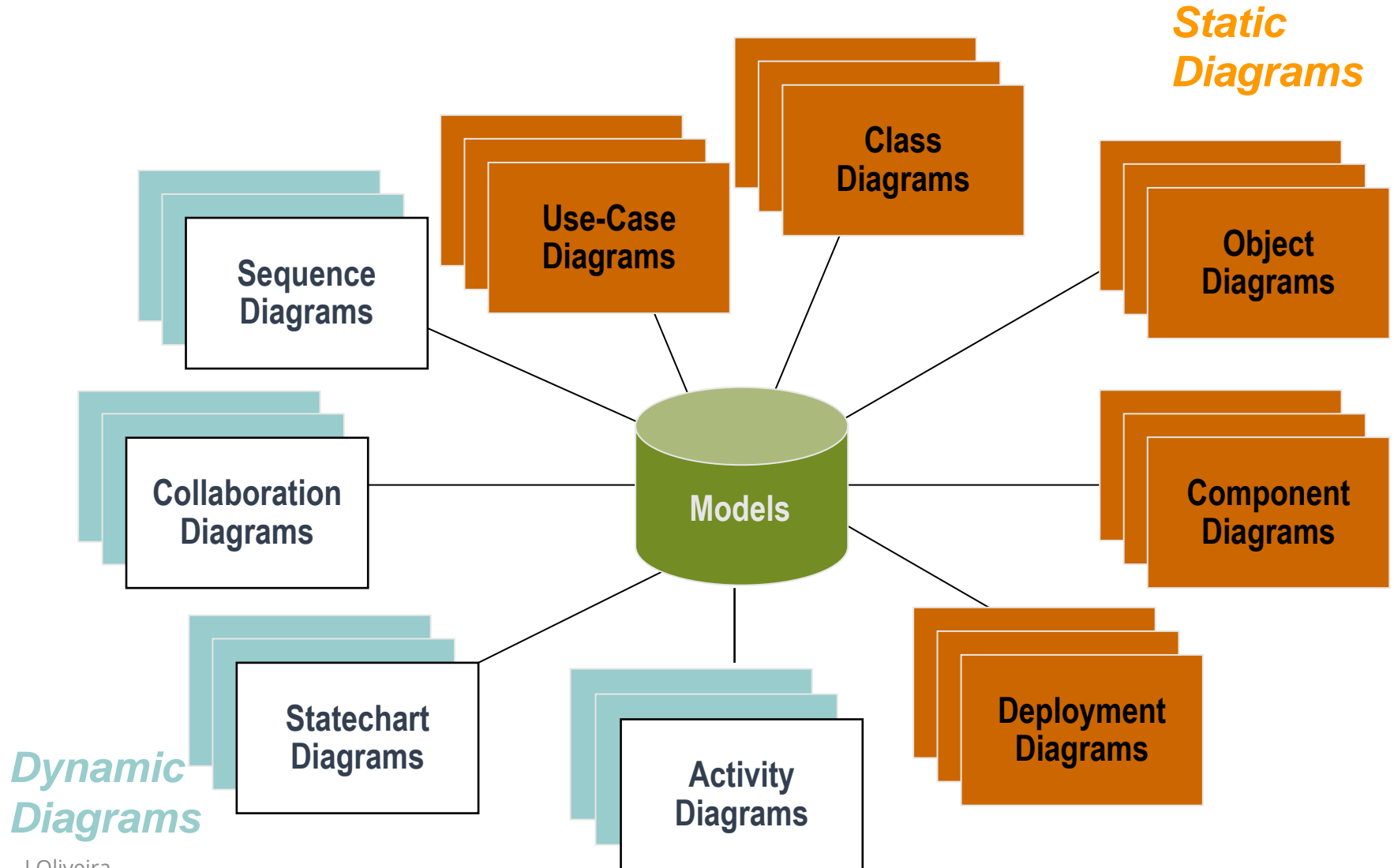
→ Colaboração entre peças/módulos

Visualizar a organização do software em partes e as suas interações

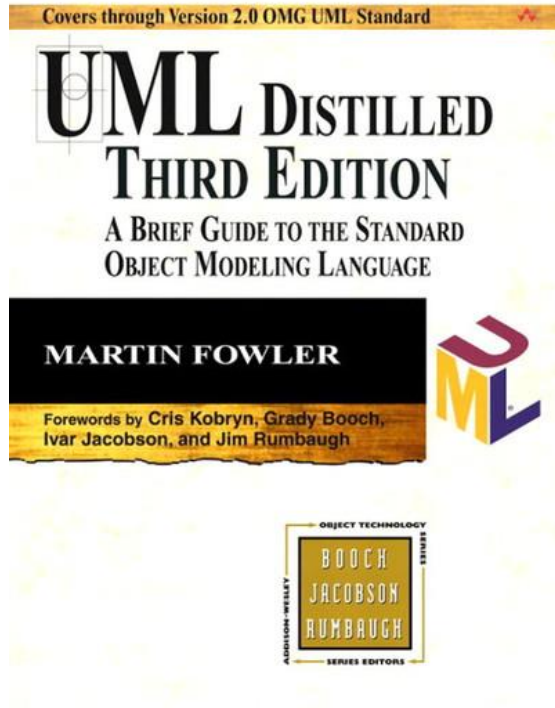
→ Explicitar arquitetura, componentes

Etc.

A UML fornece a sintaxe e a semântica para modelar diferentes aspectos de um SI

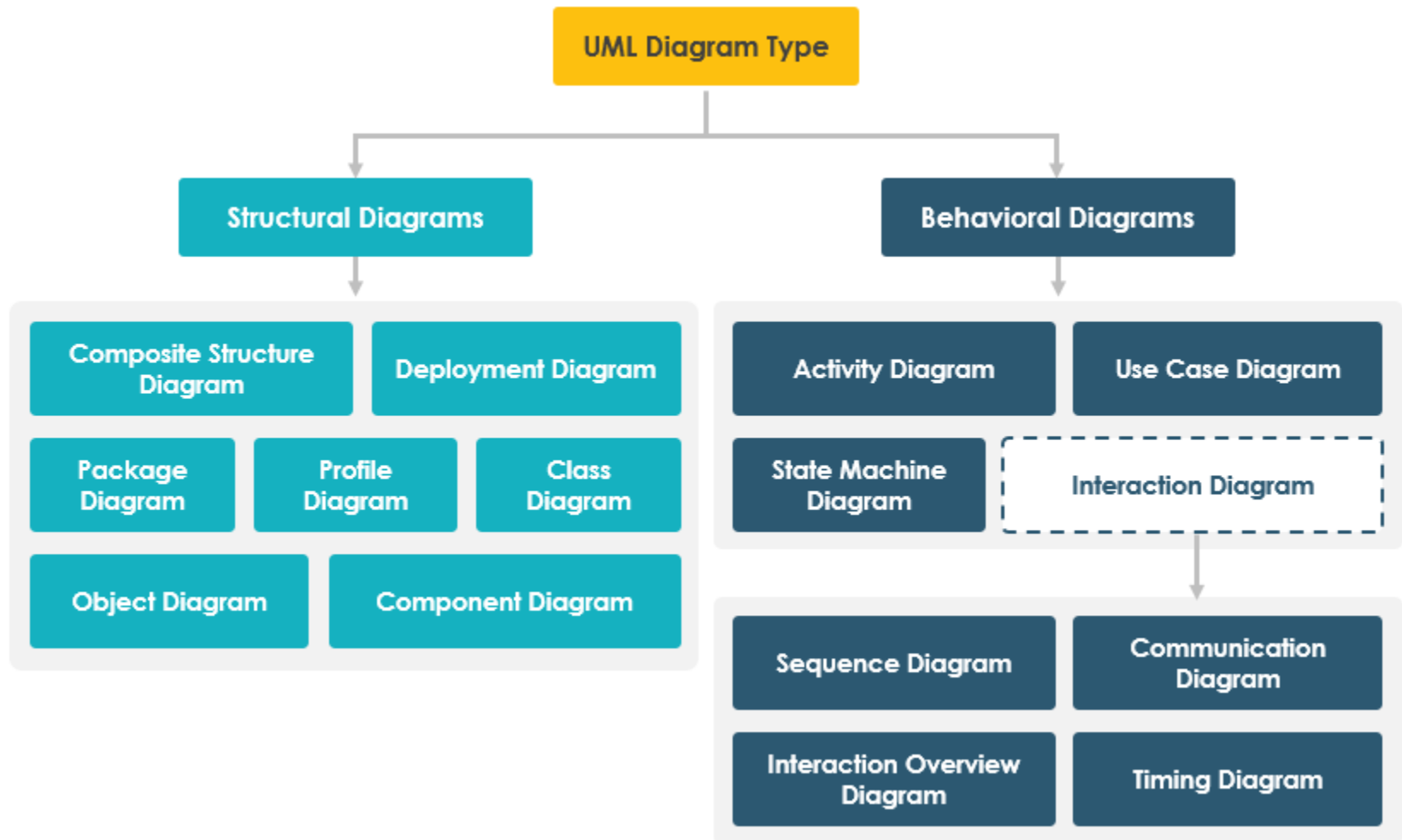


Referência rápida

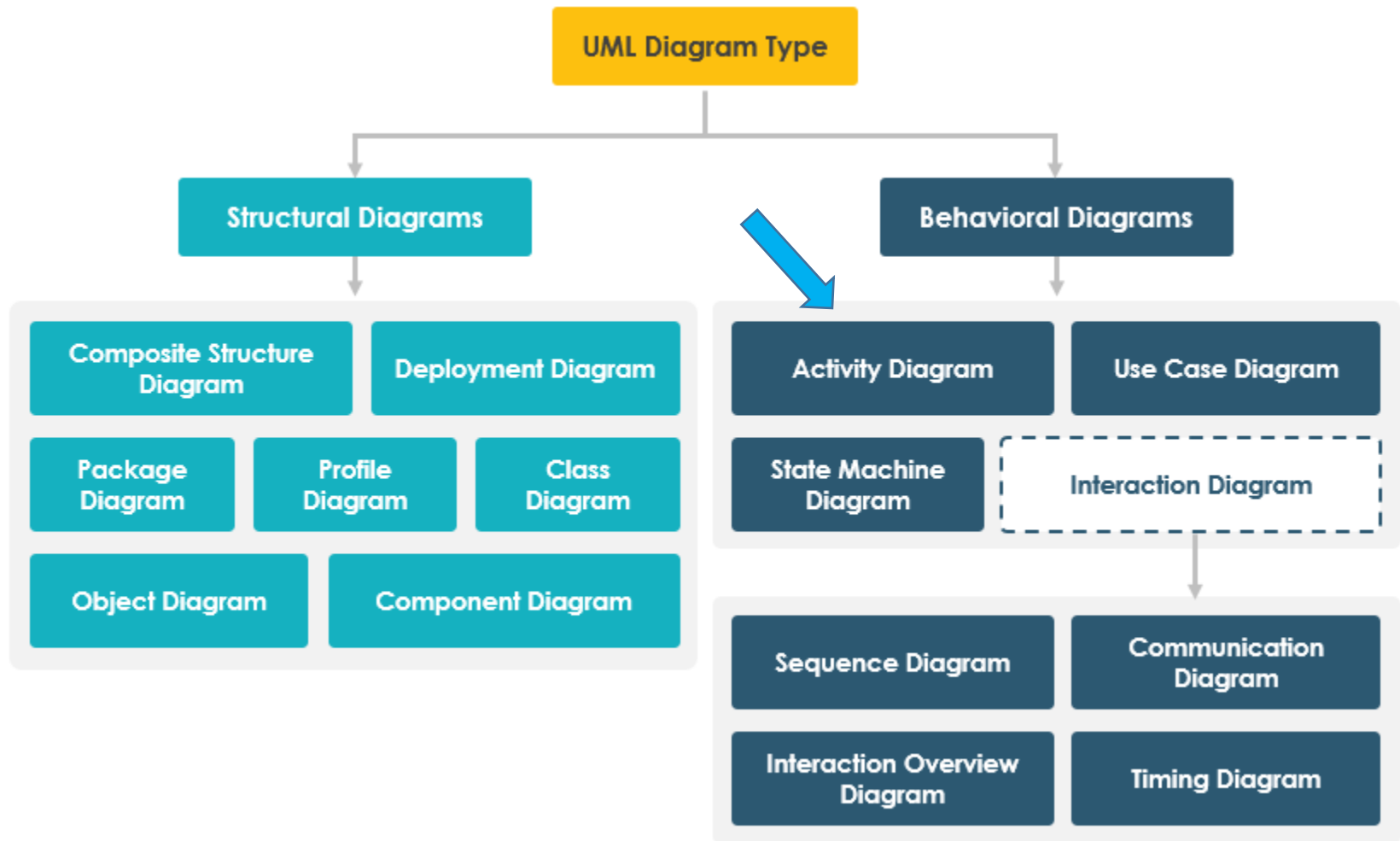


<https://learning.oreilly.com/library/view/uml-distilled-a/0321193687/>

Diagramas da UML 2.x



Diagramas da UML 2.x



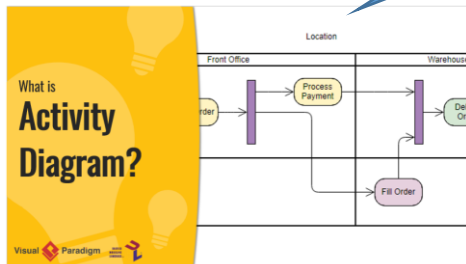
Os diagramas de atividade mostram o fluxo de ações (e de dados)

Quando aplicar?

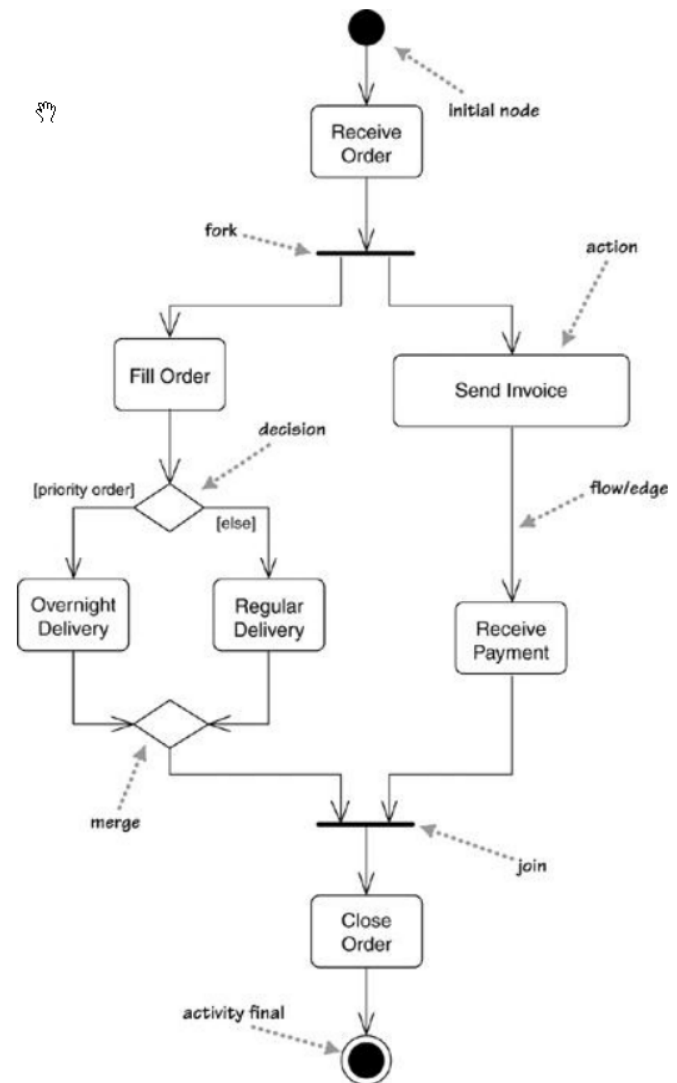
- Modelar fluxos de trabalho/processos de negócio
- Modelar processos computacionais: e.g. descrever um algoritmo complexo

Ver explicação tutorial aqui!

What is Activity Diagram?



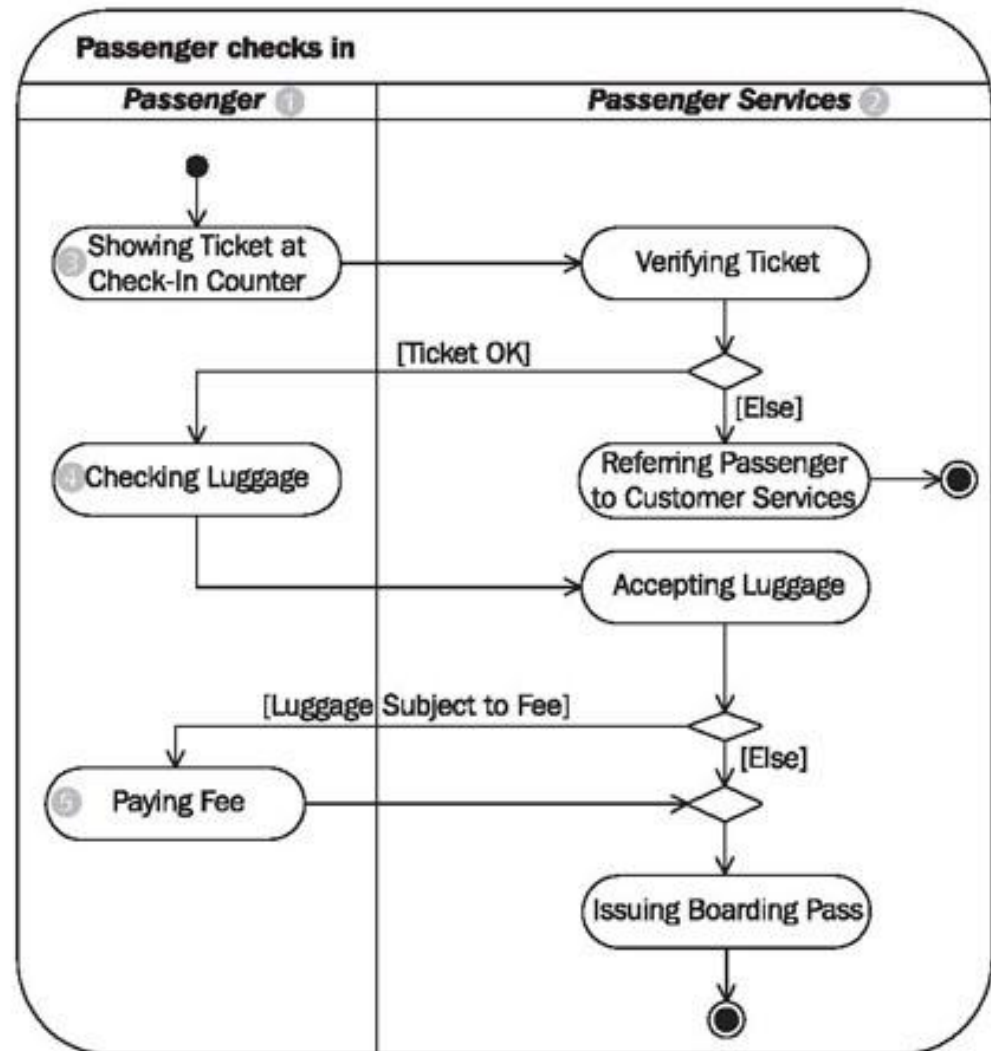
<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/>



Os DA são adequados para “business process modeling”

Pode ser usado para descrever os processos organizacionais existentes / novos

- Neutro em relação à programação
- Bom a captar papéis desempenhados
- Pode captar o fluxo de dados também



Exemplo de aplicação

Defesa de dissertações no DETI

→ <https://www.ua.pt/pt/deti/page/7976>



Diagramas para representar *workflow*

Definido na UML.

Diagram	Description	Advantages	Disadvantages
System flowchart	Earliest form for depicting sequencing of activities.	<ul style="list-style-type: none">■ Intuitive. Each type of input and output is clearly marked with its own symbol.■ Includes logic symbols.	<ul style="list-style-type: none">■ Not compliant with UML.■ Can be hard to learn (many symbols).
Activity diagram	UML tool for describing logic. Used to describe entire system, a use case, or an activity within a use case. Has two versions: <ul style="list-style-type: none">■ Activity diagram without partitions (swimlanes): Does not show who does what.■ Activity diagram with partitions: shows who does what.	<ul style="list-style-type: none">■ Part of UML standard.■ Can handle many situations in one diagram.■ Simple diagramming conventions.■ Encourages thinking about opportunities for parallel activities (more than one activity going on at the same time).	<ul style="list-style-type: none">■ Ability to handle many situations can lead to a diagram that is too complex to follow.
Business process diagram (BPD)	Business process modeling notation (BPMN) tool for describing workflow	<ul style="list-style-type: none">■ Part of BMN standard, managed by the OMG■ Rich symbol set can model complex and subtle workflow requirements better than activity diagrams.	<ul style="list-style-type: none">■ Not UML-compliant■ Difficult to understand without prior training

Referências e consulta adicional

- [Vídeo tutorial](#) com uma explicação dos Diagramas de Atividades
- Visual Paradigm tutorials: what is the [Activity Diagram](#)
- MSDN, "[Developing models for software design](#)"
- <https://www.uml-diagrams.org/activity-diagrams-examples.html> (exemplos)
- M. Fowler, "UML Distilled" (3rd ed)
- NoMagic's [UML Reference card](#)