

Introdução a Linguagem de Modelagem Unificada

UML

Sinopse

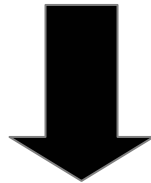
1. História da UML
2. Conceitos Gerais
3. Modelagem Visual utilizando UML
4. Apresentação dos Diagramas

História

- 60-70
 - COBOL, FORTRAN, C
 - Métodos de Análise e Projeto Estruturado
- 60: Linguagem Simula (incorporava elementos essenciais de LPOO atuais)
- 80 – início 90's
 - Smalltalk, ADA, C++, Visual Basic
- 90: Início de atração por OO
 - Java, UML, Unified Process
 - Proliferação de Métodos de DesenvolvimentoOO
 - Método: Processo + Notação

Tecnologia de Orientação a Objetos

- Com base em construções computacionais chamadas objetos, proporciona um paradigma evolucionário para:
 - criar modelos do mundo real em computador
 - usar estes modelos para simular o mundo real



- Qualquer SOO pode ser visualizado como um conjunto de objetos inter-relacionados que interagem dinâmica e colaborativamente por meio de passagem de mensagens para efetuar tarefas pertinentes a um domínio de problemas

Métodos de Desenvolvimento OO

- **CRC (Classe-Responsabilidade-Colaborador) 1989 - Wirfs-Brock**
 - *Designing Object-Oriented Software, Prentice-Hall, 1990.*
- **Modelos OOA e OOD (1991) - Coad/Yourdon**
 - *Análise Baseada em Objetos, Editora Campus, 1991*
 - *Projeto Baseado em Objetos, Editora Campus, 1991*
- **OMT (Object Modeling Technique) (1991) – James Rumbaugh**
 - *Modelagem Baseada em Objetos - Editora Campus, 1991*
- **BOOCH (1991) – Grady Booch**
 - *Object-Oriented Design with Applications, Benjamin Cummings, 1991*
- **OBJECTORY OOSE (1992) – Ivar Jacobson**
 - *Object-Oriented Software Engineering, Addison-Wesley*
- **Fusion (Booch, OMT, CRC, Métodos Formais) 1994 - Colemann**

Precursores da UML

- Grande aumento do número de métodos entre 89 e 94
- “guerra dos métodos”
- novas versões dos métodos, incorporando técnicas uns dos outros (OOSE, OMT-2, Booch’93)
- reconhecia-se que havia pontos mais fortes em cada um dos métodos
- 94: Parceria de metodologistas: Booch (GE) e Rumbaugh (OMT)
 - Busca de Unified Method (UM)
- 95: Rational compra Objective Systems de Jacobson (Objectory)
 - Rational Incorpora Objectory
 - parceria Booch/Rumbaugh estendida com Jacobson
 - Unified Modeling Language (UML) ao invés de UM)
- 98: Reengenharia de livros, métodos e cases OO para incluir UML

Por que a UML foi criada ?

- James Rumbaugh – Object Modeling Technique (OMT)
- Grady Booch – Booch Method
- Ivar Jacobson – Objectory (OOSE) Process

- Cada autor adotava idéias dos métodos dos outros, então, evoluindo juntos produziram melhorias
- Justificativa:
 - inviável estabelecer um processo universal
 - a unificação dos 3 métodos deve trazer uma estabilidade para o mercado
 - COMUNICAÇÃO: a linguagem natural é imprecisa
- UML: linguagem para modelar sistemas usando conceitos OO

Como definir UML? **NÃO É UM MÉTODO**

- Linguagem de Modelagem Unificada (*Unified Modelling Language*)
- A UML combina o melhor das metodologias em:
 - Conceitos de Modelagem de Dados (Diagramas de Entidade Relacionamento)
 - Modelagem de Negócios (workflow)
 - Modelagem de Objetos
 - Modelagem de Componentes
- Foco principal está em padronizar uma linguagem de modelagem e não em padronizar um processo
- Remove as diferenças desnecessárias entre as notações do Booch, OMT e OOSE

Características UML

- Linguagem de modelagem padronizada pela OMG (*Object Management Group*)
- Método = Linguagem de Modelagem (notação – sintaxe da linguagem de modelagem – a coisa gráfica)
- +
- Processo
- Para discutir um projeto, é fundamental compreender a linguagem de modelagem, não o processo que foi utilizado para desenvolver o projeto
- UML define uma notação e um metamodelo
- Método RUP (*Rational Unified Process*)
 - Concepção, Elaboração, Construção e Transição
 - Desenvolvimento Iterativo e Incremental

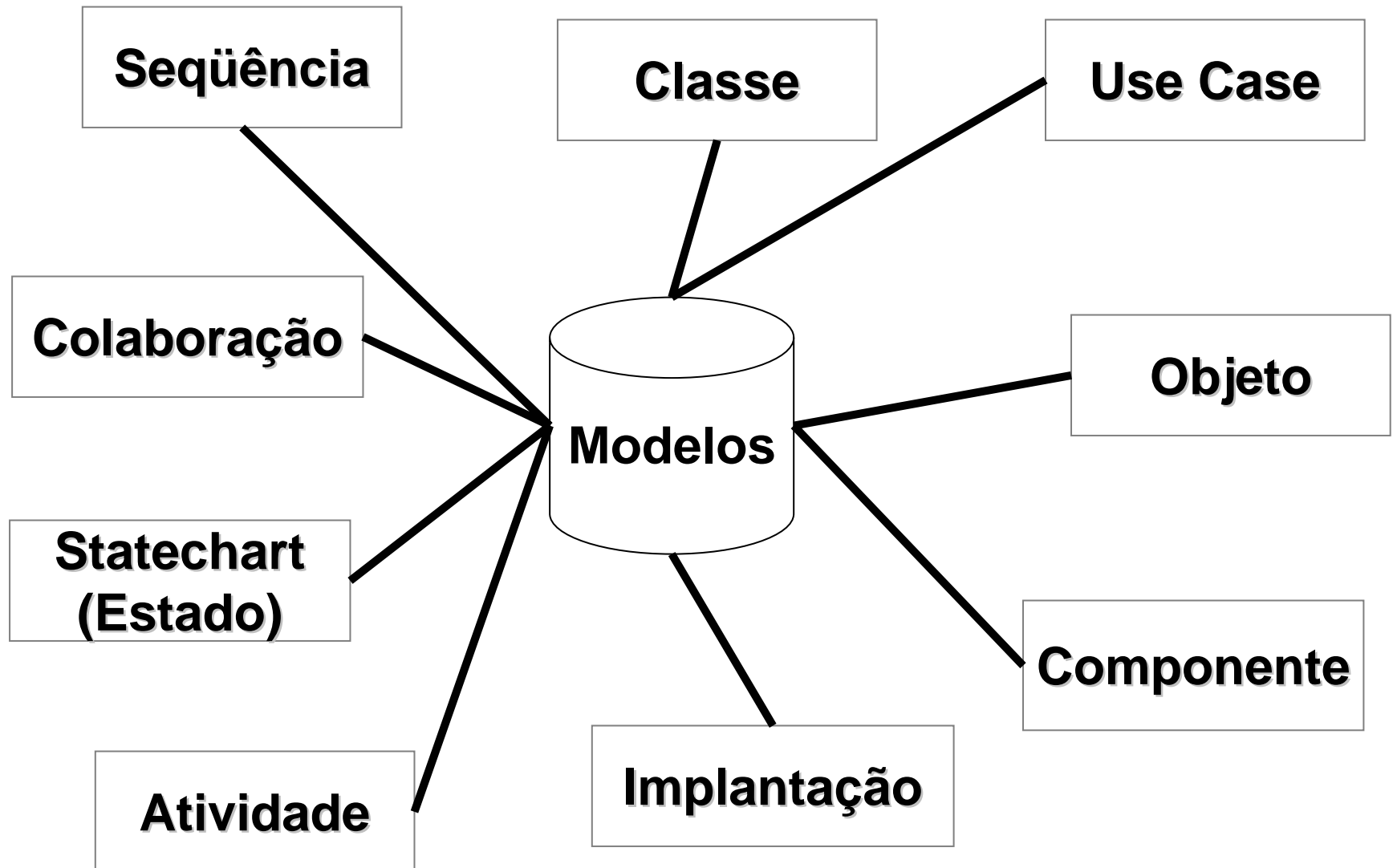
Usos de UML

- A UML é uma linguagem de modelagem para
 - Visualização
 - Especificação
 - Construção
 - Documentação
 - Comunicação



Diagramas

9 Diagramas



Diagramas Comportamentais

- 1. Diagramas de Caso de Uso
 - organiza o comportamento do sistema
- 2. Diagrama de Seqüência
 - Tem, como foco, a ordem temporal das mensagens
- 3. Diagrama de Colaboração
 - Tem, como foco, a organização estrutural de objetos que enviam e recebem mensagens
- 4. Diagramas de Estados
 - Tem, como foco, o estado de mudança de um sistema orientado por eventos
- 5. Diagrama de Atividades
 - Tem, como foco, o fluxo de controle de uma atividade para outra

Elementos Estruturais

- São as partes estáticas de um modelo, representando elementos conceituais ou físicos
 - Classe
 - Componente
 - Casos de Uso
 - Interface
 - Nó

Elementos Comportamentais

- São as partes dinâmicas dos modelo da UML
 - Interação: especifica um conjunto de mensagens trocadas entre objetos
 - Máquina de Estado: especifica seqüências de estados de um objeto

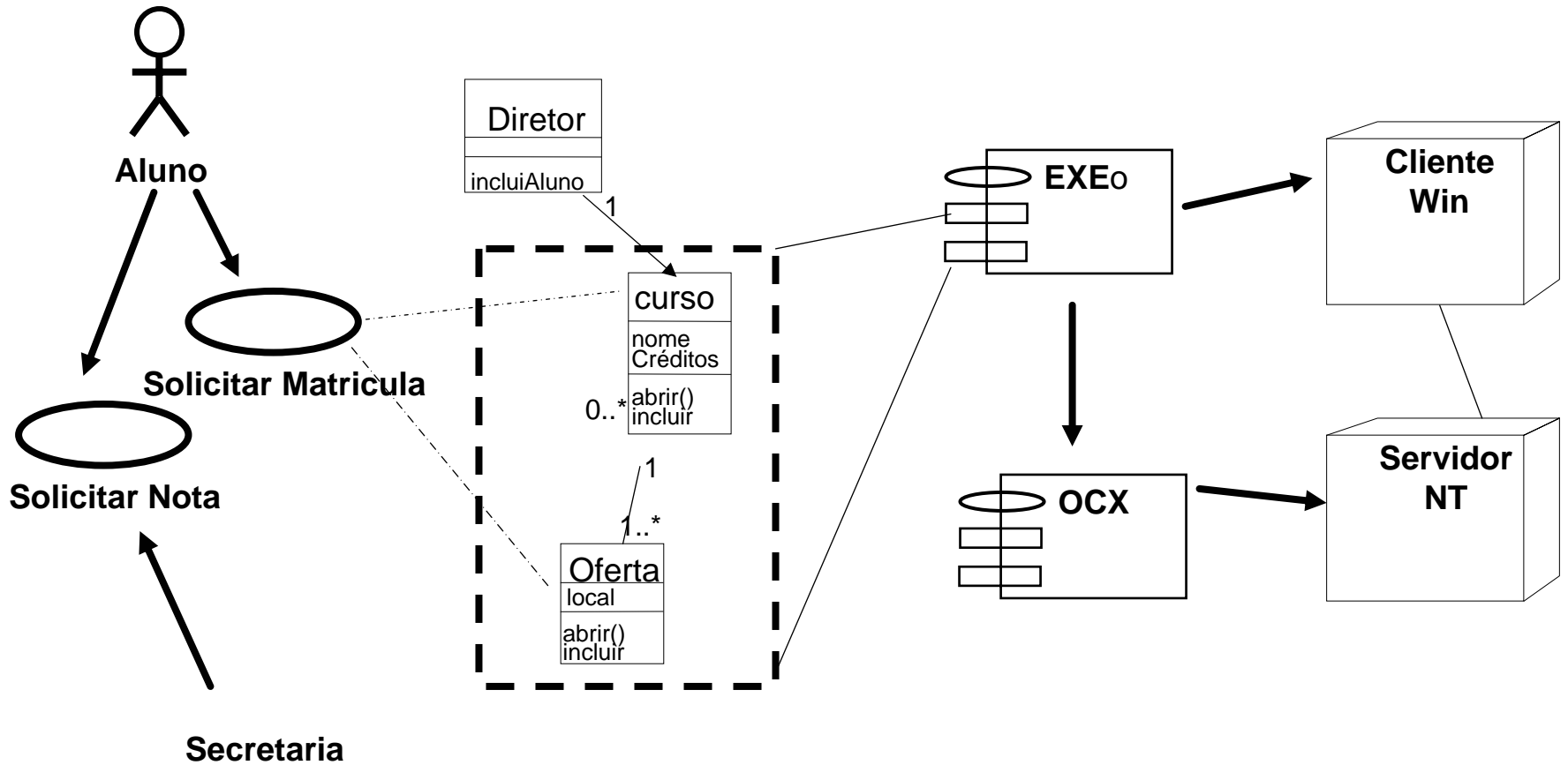
Elementos de Agrupamento

- São as partes organizacionais dos modelo da UML
 - Pacotes: mecanismo para organizar os elementos em grupos

Elementos de Anotação

- São as partes explicativas dos modelo da UML. São comentários utilizados para descrever, iluminar e remarcar elementos no modelo
 - Nota: símbolo contendo restrições ou comentários que são melhor expressados em textos

Análise usando UML



Caso de uso

Classes

Componentes

Implantação