

# EMPREGAR DIAGRAMAS DE OBJETOS E DE PACOTES PARA APOIAR A ESPECIFICAÇÃO DE UM SISTEMA

## VISÃO GERAL DO DIAGRAMA DE OBJETOS

Além do Diagrama de Classes, a UML define um segundo tipo de diagrama na perspectiva estática e estrutural do sistema: o Diagrama de Objetos.

Assim como objetos são instâncias de classes, os Diagramas de objetos podem ser vistos como instâncias de Diagramas de Classes.

Um Diagrama de Objetos UML representa uma instância específica de um Diagrama de Classes em determinado momento. Portanto, do ponto de vista gráfico, ele é bastante parecido com o Diagrama de Classes.

Saiba mais

Bezerra (2015) afirma que um Diagrama de Objetos exibe uma “fotografia” do sistema em certo momento, mostrando as associações estabelecidas entre objetos, de acordo com suas interações e com os valores que seus atributos assumem.

Um Diagrama de Objetos mostra os valores de atributos do conjunto de objetos que ele representa e como esses objetos se relacionam entre si. Um objeto, ou uma instância de uma classe, possui uma identificação única (não existem dois objetos iguais) e os seus atributos assumem determinados valores em um momento específico no tempo.

Os valores dos atributos podem sofrer alterações e, nesse caso, dizemos que o objeto sofreu uma **mudança de estado**.

### Exemplo

Por exemplo, o objeto Maria, criado a partir de uma classe Pessoa, possui como valor do atributo Endereço: Rua das Palmeiras. Quando Maria muda de endereço, o valor do atributo passa a ser o novo endereço: Avenida Vieira Souto. Assim, em um diagrama de estados, o objeto Maria seria representado com o valor do atributo Endereço quando o diagrama foi criado.

No Diagrama de Objetos, não faz sentido falar em operações, que são as mesmas para todos os objetos de uma classe. Portanto, nesse diagrama, o elemento principal, o Objeto, possui apenas o nome e os atributos com seus valores.

Em um Diagrama de Objetos, cada objeto é representado por um retângulo com dois compartimentos. O compartimento superior mostra a identificação do objeto na forma de texto sublinhado:

**:NomeClasse**

**nomeObjeto: NomeClasse**

Lembre-se de que cada objeto é único e, portanto, esse nome identifica exclusivamente um objeto. A identificação do objeto pode ser omitida.

Por exemplo:

- Marta: Professor
- :Professor

No primeiro caso, o objeto da classe professor foi identificado com o nome Marta; no segundo, o analista optou por não caracterizar o objeto em si, significando que pode estar se referindo a qualquer objeto daquela classe.

A segunda representação é frequentemente usada na etapa de projeto do processo de desenvolvimento de sistemas, quando se representa a interação entre objetos (instâncias de classes) nos diagramas de sequência ou de comunicação.

No compartimento inferior (cuja utilização é opcional), aparecem os valores que os atributos definidos na classe assumem para este objeto. Dessa forma, deve mostrar uma lista de pares da forma: nome do atributo = valor do atributo.

A figura 14 reproduz o exemplo de Diagrama de Objetos mostrado por Bezerra (2015). Nesse exemplo, uma instância de pedido é associada a três instâncias de ItemPedido. Cada item, por sua vez, está ligado a um produto. Os valores para cada atributo de cada objeto são apresentados nesse diagrama.

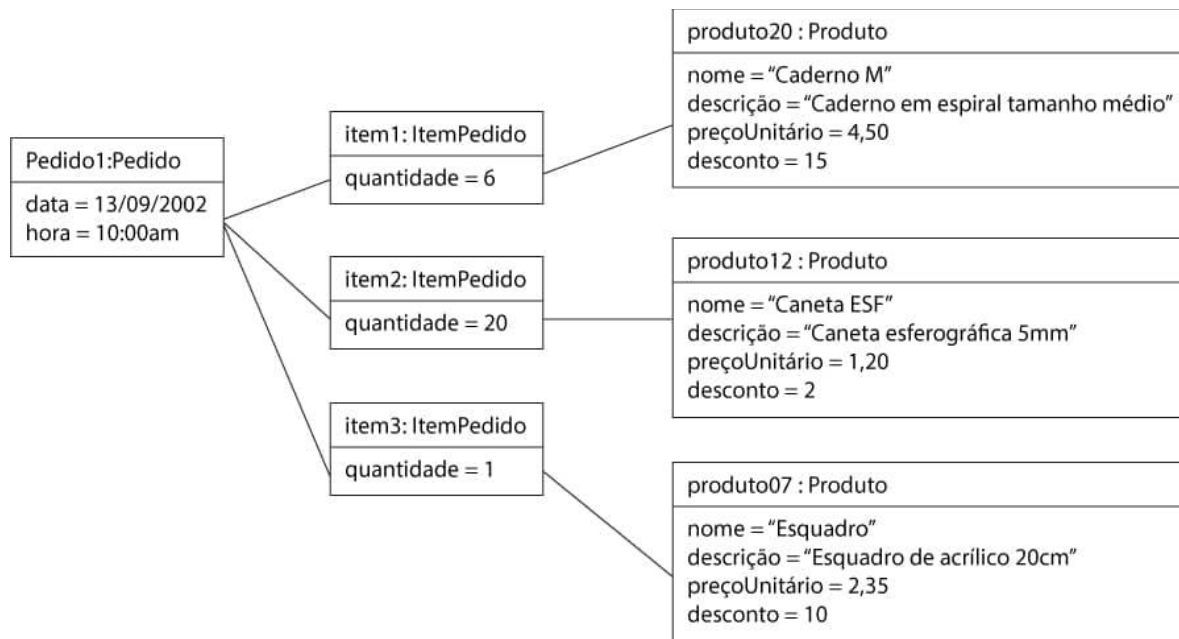


Figura 14 – Exemplo de Diagrama de Objetos (BEZERRA, 2015)

Você pode observar também que não existem detalhes nos relacionamentos, tal como no Diagrama de Classes. Os relacionamentos no Diagrama de Objetos mostram uma situação específica em que aqueles objetos se encontram, não as possíveis relações entre eles.

No diagrama da figura 14, temos um Pedido que possui 3 Itens associados a ele, e 1 Produto associado respectivamente a cada Item.

A figura 15 mostra outro exemplo de Diagrama de Objetos, em que observamos que o Cliente Pablo está associado a 2 contratos de aluguel de carros diferentes. Cada contrato tem seu identificador e atributos próprios. Esse diagrama ajuda no entendimento da situação desse cliente.

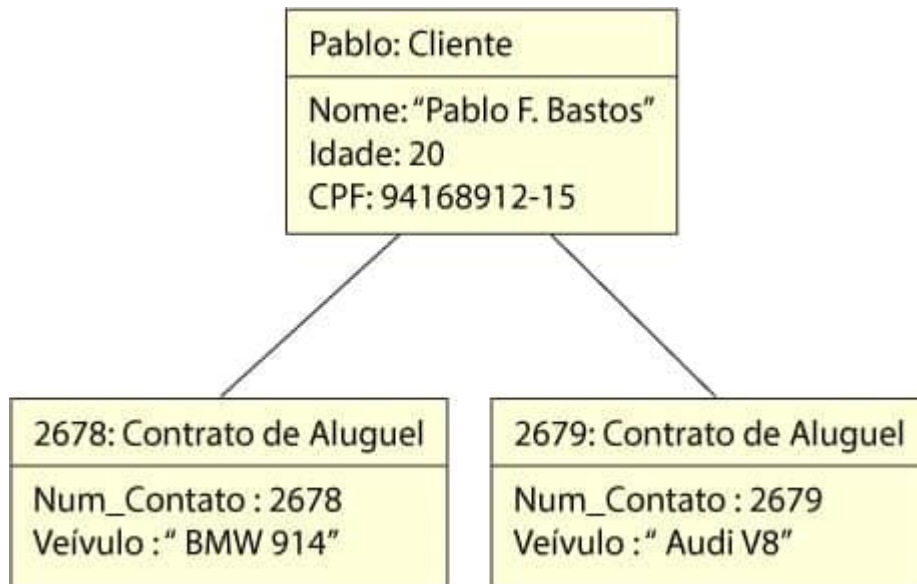


Figura 15 –

Exemplo de Diagrama de Objetos

Saiba mais

De acordo com Bezerra (2015), Diagramas de Objetos não são muito utilizados na prática. Mas eles podem ser úteis na atividade de validação no processo de desenvolvimento, pois permitem ilustrar e esclarecer certos aspectos de um Diagrama de Classes.

Mesmo antes de criar um Diagrama de Classes, você pode criar um Diagrama de Objetos para descobrir fatos sobre elementos específicos e suas associações ou para esclarecer dúvidas por meio de exemplos específicos. Além disso, ajuda na construção dos diagramas de interação (de sequência ou comunicação), pois estes utilizam a mesma notação.

Em resumo, os Diagramas de Objetos podem ter as seguintes aplicações:

Ajudar na análise de uma interação específica.

Mostrar uma visão geral de alto nível do sistema.

Validar um Diagrama de Classes.

Os Diagramas de Objetos podem ser usados também para verificar se o sistema foi desenvolvido conforme os requisitos, e até mesmo para analisar se uma regra de negócio está sendo cumprida.

#### Dica

Para criar esses diagramas, procure exibir associações entre objetos somente com uma ligação (por exemplo, uma linha única juntando dois objetos, sem setas ou outros detalhes), pois um Diagrama de Objetos tem por objetivo mostrar somente a existência de associações e não os tipos de associações, como em um Diagrama de Classes. Além disso, evite representar todos os objetos em um único Diagrama de Objetos, pois ele apresenta apenas um estado do objeto.

### VISÃO GERAL DO DIAGRAMA DE PACOTES

Muitos sistemas possuem uma quantidade bastante grande de casos de uso e classes, portanto, representá-los em um único diagrama é inviável devido à sua complexidade para o entendimento. Para resolver esse problema, usamos o elemento da UML chamado de pacote (do inglês, *package*).

Os pacotes servem para agrupar diagramas, tais como classes, casos de uso, ou mesmo outros pacotes. Normalmente, os grupos de classes são associados com módulos de programação do sistema. Uma classe só pode aparecer uma vez no mesmo pacote, mas pode estar presente em vários pacotes diferentes.

Um pacote representa um grupo de elementos que pode se relacionar com outros pacotes por meio de uma relação de dependência. É possível definir a visibilidade (conforme estudamos no Módulo 3) de pacotes e dos elementos contidos nele. Pacotes são úteis para prover uma visão geral de um conjunto grande de objetos; para organizar um modelo grande e agrupar seus elementos.

A representação gráfica de um pacote é um retângulo com uma aba no canto superior esquerdo, com o nome do pacote na aba ou no centro do retângulo. A figura 16 mostra um exemplo de pacotes que agrupam conjuntos de classes. O pacote Módulo Docente agrupa as classes relacionadas com professores, o Módulo Discente agrupa as classes relacionadas com alunos, e o Módulo Organização Curricular agrupa classes pertinentes à matriz curricular, em um sistema acadêmico.

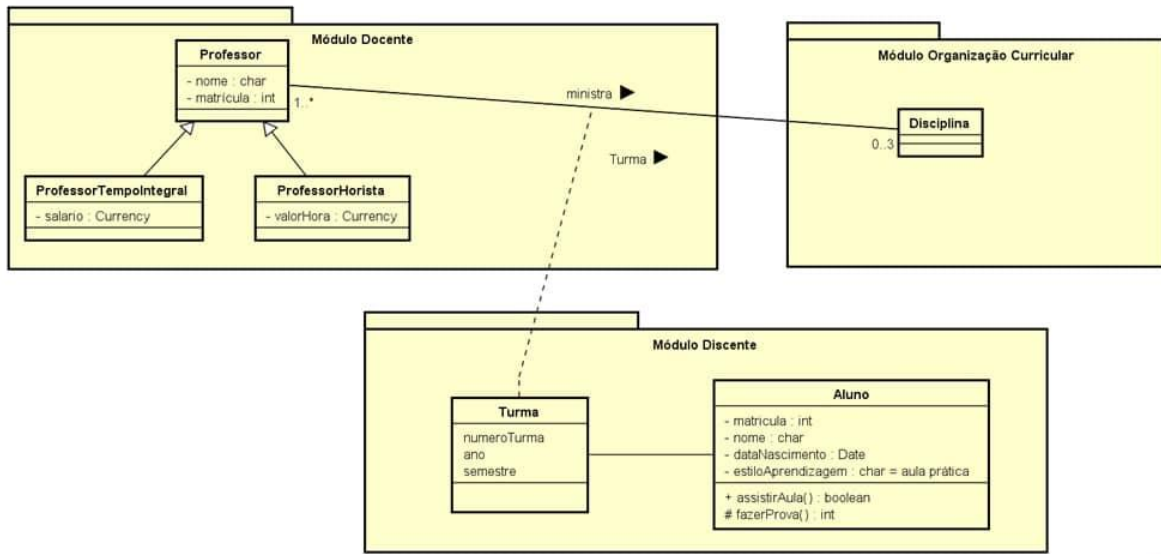


Figura 16 – Exemplo de representação de pacote, elaborado com ferramenta ASTAH (2021)

## DESDOBRAMENTOS DOS DIAGRAMAS NO PACOTE

As ligações entre pacotes são relacionamentos de dependência. Um pacote P1 depende de outro pacote P2, se algum elemento contido em P1 depende de algum elemento contido em P2.

A figura 17 mostra o Diagrama de Pacotes do exemplo do sistema acadêmico sem os relacionamentos explícitos entre as classes internas; apenas com a representação de dependências existentes entre os próprios pacotes. Ou seja:

Os pacotes se relacionam com outros pacotes por meio de um vínculo de dependência.

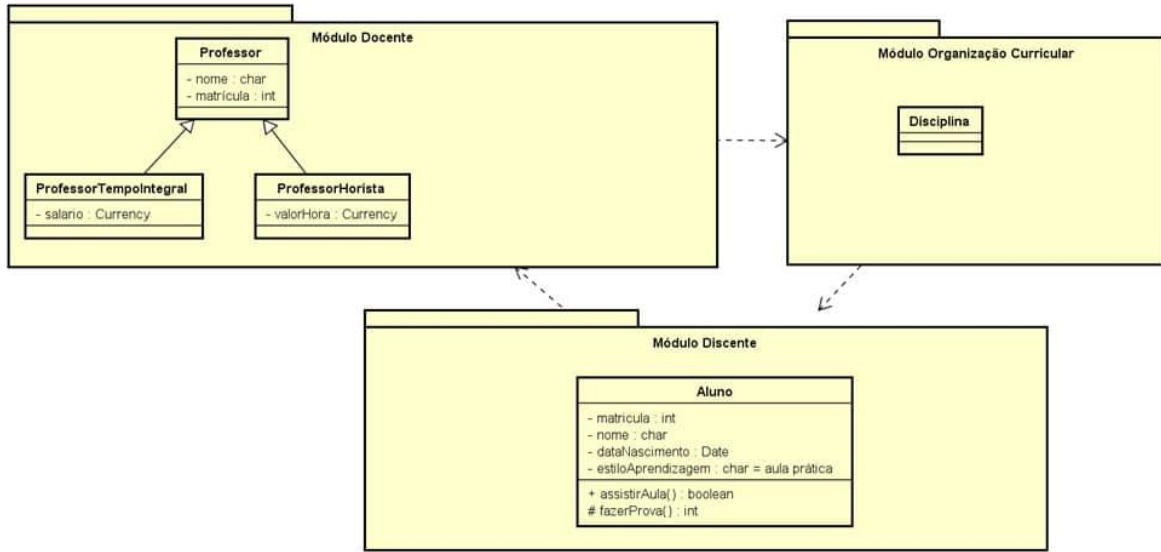


Figura 17 – Exemplo de representação de dependências entre pacote, elaborado com ferramenta ASTAH (2021)

As dependências entre pacotes mostram que um pacote depende ou influencia outro. As dependências podem ser divididas em dois tipos: acesso e importação.

Clique nas informações a seguir.

#### Dependência de acesso

Indica que um pacote requer assistência das funções de outro pacote.

#### Dependência de importação

Indica que uma funcionalidade foi importada de um pacote para outro.

Ambas são representadas por uma linha pontilhada com uma seta apontando para o pacote dependente.

Pacotes também podem formar grupos de pacotes e hierarquias. A figura 18 (Bezerra, 2015) mostra como pacotes podem ser agrupados dentro de outros pacotes, formando uma hierarquia de contenção.

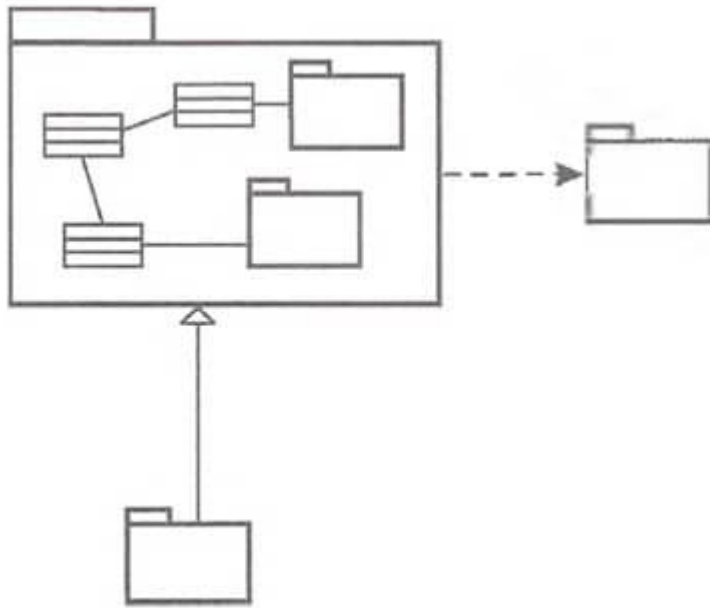


Figura 18 - Agrupamento de pacotes (BEZERRA, 2015)

Além disso, como já vimos, existem três tipos de visibilidade de elementos dentro dos pacotes:

**+ (público)**

Visível por todos que importam ou acessam o pacote.

**# (protegido)**

Visível só pelos pacotes filhos (por relação de generalização).

**- (privado)**

Visível só por outros elementos do pacote.

**Atenção**

Note que aqui não se aplica a visibilidade de pacotes (~), que é aplicável tão somente a atributo de classe que seja visível a qualquer classe pertencente ao mesmo pacote no qual está definida a classe.

No contexto da modelagem de casos de uso, os pacotes podem ser utilizados com diversos objetivos. Alguns desses objetivos são: estruturar o modelo de casos de uso com base nos grupos de usuários do sistema; definir a prioridade na qual os casos de uso serão desenvolvidos; e definir o grau de correlação entre os casos de uso.

O critério para criar os pacotes é subjetivo e depende da visão e das necessidades do projetista.

**Comentário**

Normalmente, o projetista define uma semântica e aloca elementos similares em um pacote, pois esses elementos tendem a ser alterados em conjunto no mesmo pacote.

Um pacote bem estruturado tem as seguintes características:

É coeso em relação aos elementos incluídos.

É fracamente acoplado em relação à exportação apenas dos elementos que os outros pacotes precisam enxergar.

Não contém muitos alinhamentos, ou seja, possui um conjunto equilibrado de elementos.

Ao especificar um pacote, é uma prática usar a forma simples de ícone retangular, obscurecendo o seu conteúdo. No caso da necessidade de apresentar o seu conteúdo interno (como nas figuras 16 e 17), deve-se mostrar apenas os elementos necessários para a compreensão do significado.

Esse diagrama também é muito utilizado para ilustrar a arquitetura de um sistema, mostrando o agrupamento de suas classes. A figura 19 mostra um exemplo de Diagrama de Pacotes usado com o objetivo de organizar as classes de acordo com as camadas da arquitetura do sistema. Tipicamente, um sistema pode ser organizado conforme o padrão de projeto Model (Modelo), View (Visão) e Control (Controle).

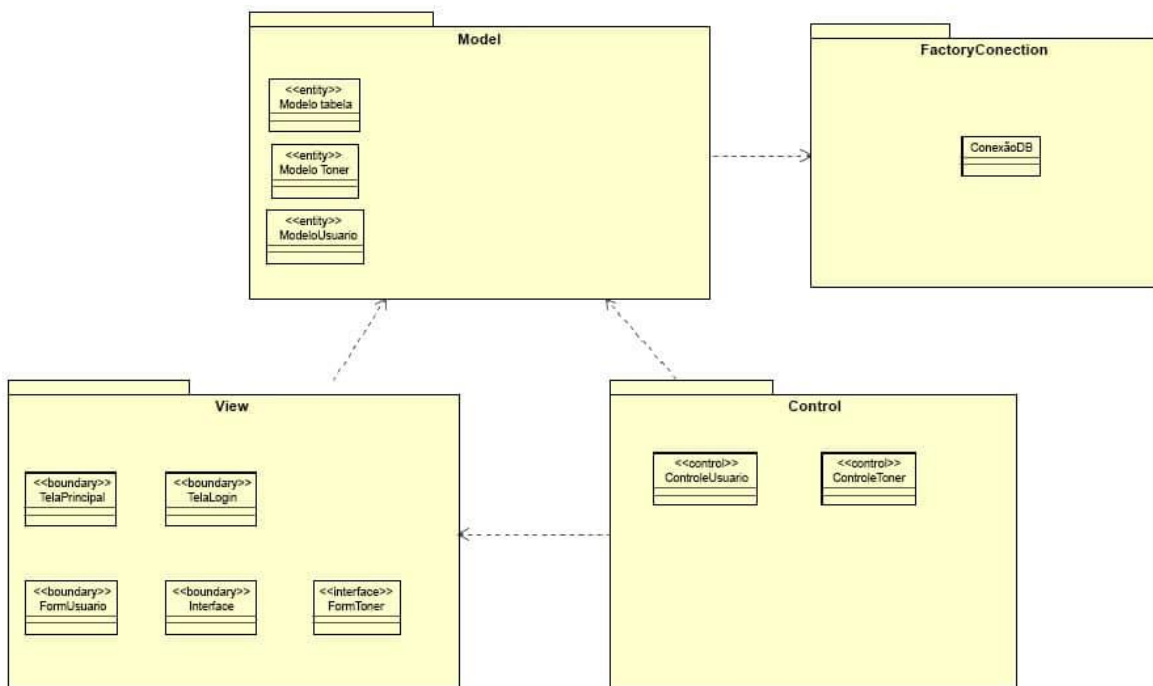


Figura 19 – Exemplo de diagrama de pacotes

O pacote tem uma grande similaridade com a agregação. O fato de um pacote ser composto de modelos de elementos cria uma espécie de agregação de composição, pois, se esse pacote for destruído, todo o seu conteúdo também será.