

Composição de Componentes

Eduardo Figueiredo

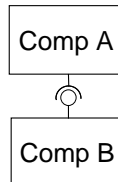
<http://www.dcc.ufmg.br/~figueiredo>

Composição de Componentes

- É o processo de integração de componentes para criar o sistema
- Tipos de composição
 - Composição hierárquica
 - Composição sequencial
 - Composição aditiva

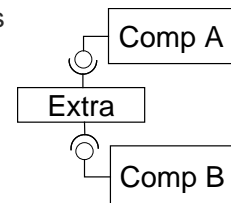
Composição Hierárquica

- Um componente chama diretamente os serviços de outro componente
- Uma interface provedora é composta diretamente a uma interface requerida



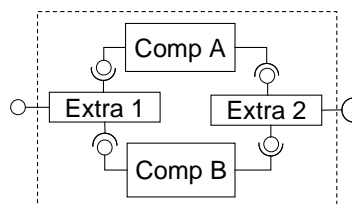
Composição Sequencial

- Serviços dos componentes constituintes são executados em sequência
- Interfaces provedoras de dois componentes são compostas
- Algum código extra é necessário na composição



Composição Aditiva

- Interfaces de dois ou mais componentes são compostas para criar um novo componente
- A remoção de operações duplicadas pode ser necessária



Incompatibilidades

- Durante a composição de componentes, podem ocorrer incompatibilidades
- Tipos comuns de incompatibilidades são
 - Incompatibilidade de operação
 - Incompatibilidade de parâmetros
 - Operações incompletas

[Incompatibilidade de Operação]

- Os nomes das operações nas interfaces provedoras e requeridas são diferentes
- Exemplo
 - Requerida:
public void debitar(double valor);
 - Provedora:
public void sacar(double valor);

[Incompatibilidade de Parâmetros]

- As operações têm o mesmo nome nas interfaces, mas os parâmetros são diferentes
- Exemplo
 - Requerida:
public void debitar(double valor);
 - Provedora:
public void debitar(int conta, double valor);

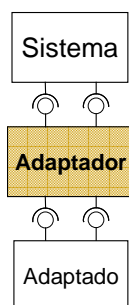
[Operações incompletas]

- As operações de uma interface provedora é um subconjunto das operações da interface requerida
 - Ou vice-versa
- Exemplo
 - Provedora: debitar e creditar
 - Requerida: debitar

[Componente Adaptador]

- Problemas de incompatibilidade são geralmente resolvidos usando um componente adaptador
- Exemplos de adaptações
 - Adicionar ou remover operações (específicas da aplicação)
 - Renomear interfaces ou elementos internos das interfaces, etc.

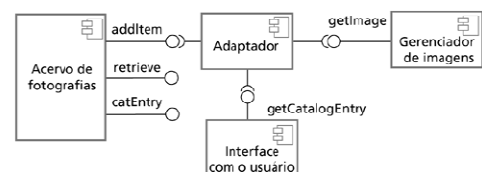
[Representação do Adaptador]



- Um componente adaptador converte uma interface em outra
 - A implementação é mínima
 - Tipicamente, componentes adaptadores não implementam novos serviços

[Tipos de Adaptadores]

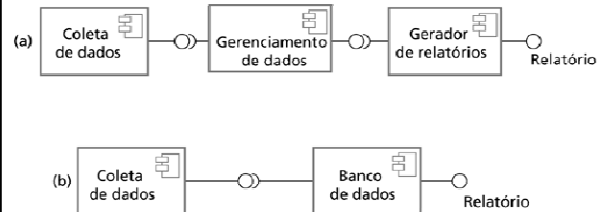
- O tipo do adaptador depende da natureza da composição
 - Hierárquica, sequencial ou aditiva



[Escolha da Solução]

- Em muitas situações, existem soluções variadas para o mesmo problema
- Exemplo
 - Considere um sistema de coleta de dados de fontes diferentes e geração de relatórios a partir destes dados

[Possíveis Soluções CBSE]



[Vantagens da Solução A]

- Os gerenciamentos dos dados e da geração de relatórios estão separados
 - Há maior flexibilidade para mudanças futuras
- Tanto do componente de Gerenciamento de Dados quanto o Gerador de Relatórios podem ser substituídos de forma independente

[Vantagens da Solução B]

- Um componente de Banco de Dados tem a geração de relatórios *built-in*
 - Semelhante ao que ocorre no MS Access
- Menos componentes são necessários
 - O sistema pode ter melhor desempenho, pois evita o *overhead* de comunicação
- Regras de integridade dos dados podem ser implementadas no mesmo componente
 - As regras são válidas tanto para os dados quanto para o relatório que apresenta os dados (menos replicação)

[Bibliografia]

- Ian Sommerville. **Engenharia de Software**, 9ª Edição. Pearson Education, 2011.
 - Cap. 17 (Seção 17.3)