

Padrões

- Um padrão de é uma maneira de documentar uma solução conhecida para um problema usualmente encontrado
- O objetivo do padrão é permitir que boas soluções sejam reutilizadas em diferentes projetos
- Um padrão de projeto possui 3 partes distintas:
 - Contexto
 - Problema recorrente neste contexto
 - Solução para o problema
- Características dos Padrões de Projeto
 - São observados através da experiência
 - São descritos de uma forma estruturada
 - Previnem contra a “reinvenção da roda”
 - Existem em diferentes níveis de abstração
 - Estão em desenvolvimento contínuo
 - São artefatos reutilizáveis
 - Transmitem melhores práticas
 - Permitem o uso de um vocabulário comum
 - Podem ser utilizados em conjunto para resolver um problema mais amplo

Classificação de Padrões

Padrões

- Os padrões de projeto podem ser classificados de acordo com a fase de desenvolvimento em que são mais adequados:
 - Padrões de Análise (Analysis patterns)
 - Seu foco é na fase de análise ou modelamento de negócio
 - Padrões ligados ao domínio do problema
 - Padrões de Arquitetura (Architectural patterns)
 - Seu foco é na arquitetura do software
 - Padrões de Projeto (Design patterns)
 - Foco no projeto de componentes do software
- Muitas das vezes os padrões podem estar muito ligados tanto ao domínio da solução, quanto do problema

Classificação de Padrões

Padrões

- Padrões de Análise (Analysis patterns)
 - Martin Fowler , 1996
- Padrões de Arquitetura (Architectural patterns)
 - Apresentado inicialmente por Frank Buschmann et al., 1996
 - Computação Distribuída - Frank Buschmann et al., 2007
- Padrões de Projeto (Design patterns)
 - GOF (Gang of Four) E. Gamma, R. Helm, R. Johnson, J. Vlissides – 1995
 - Aplicações Concorrentes e em Rede - Frank Buschmann et al. – 2000
 - Enterprise Integration Patterns – Gregor Hohpe, 2003
 - Real-time Design Patterns – Bruce Douglass, 2003
 - .Net Design Patterns - Christian Thilmany, 2003
 - J2EE Design Patterns - Deepak Alur, 2003
 - Web Services Patterns – Paul Monday, 2003
 - Ajax Design Patterns - Michael Mahemoff, 2006
 - SOA Design Patterns – Thomas Erl, 2009

Padrões de Análise (Analysis Patterns)

- Proposto por Martin Fowler, em livro publicado em 1996
- Notação do Livro não é baseada em UML
- Baseada em áreas (domínios) específicas como: manufatura; financeira e saúde
- Mesmo assim, padrões podem apresentados podem ser úteis em outros domínios
- Alguns princípios apresentados
 - Um modelo não está certo ou errado, eles podem ser mais ou menos úteis
 - Modelos conceituais estão ligados a tipos (interfaces) e não implementações (classes)
 - Padrões são o ponto de partida, não o destino
 - Sempre que possível, quando existir um tipo e um supertipo, considere colocar os recursos no supertipo, desde que isto faça sentido
 - Quando múltiplos atributos possuam um comportamento relacionado e presente em muitos tipos, combine estes atributos em um novo tipo fundamental

Padrões de Análise (Analysis Patterns)

- Exemplos de alguns padrões de projeto
 - Quantity (3.1)
 - Conversion Ratio (3.2)
 - Compound Units (3.3)
 - Measurement (3.4)
 - Observation (3.5)
 - Range (4.3)
 - Name (5.1)
 - Account (6.1)
 - Transaction (6.2)
 - Summary Account (6.3)
 - Plan (8.4)
 - Contract (9.1)
 - Product (10.3)
 - Associative Type (15.1)

Padrões de Projeto

GoF

- Trabalho proposto inicialmente por Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides (Gang of Four) em 1995
- Famílias de Padrões
 - De Criação
 - Responsáveis pela criação de objetos
 - Permitem que o sistema fique independente da forma como os objetos são criados
 - Estruturais
 - Relacionados com a forma com que classes e objetos são compostos a fim de formar estruturas maiores
 - Comportamentais
 - Relacionados com a atribuição de responsabilidades entre objetos
 - Descrevem a comunicação entre objetos