

Universidade Federal de Campina Grande - UFCG
Centro de Engenharia Elétrica e Informática - CEEI
Coordenação de Pós-Graduação em Informática - COPIN

PROJETO DE SOFTWARE ORIENTADO A OBJETO

RESUMO - Architectural Patterns Revisited - A Pattern Language
Maio de 2008

João Arthur Brunet Monteiro
Mestrado em Ciências da Computação - CEEI/UFCG
Campina Grande, Maio de 2008

1 Resumo

Este documento traz uma sumarização do artigo *Architectural patterns revisited - a pattern language* [1], bem como de minhas opiniões a respeito do assunto abordado no mesmo.

Embora o uso de padrões arquiteturais seja reconhecido como atividade de extrema importância na construção de sistemas software, não há uniformidade na aplicação destes padrões. O que na prática se observa é a aplicação *ad hoc* desses conceitos, tornando mais difícil a tarefa de arquitetos e engenheiros de *software* projetarem um sistema. Entre as diversas causas deste problema, podemos destacar entre as mais importantes duas destas causas:

- Falta de consenso na granularidade ideal dos padrões arquiteturais de projeto.
- Falta de um catálogo de padrões arquiteturais para ser usado como guia na especificação de arquiteturas de *software*.

Dois principais visões sobre o assunto são apresentadas pelos autores, entre elas, padrões arquiteturais baseiam-se na descrição de problemas e dos padrões relacionados a este padrão. Por outro lado, estilos arquiteturais não focam no problema e sua solução, mas nos componentes, comunicação, controle dos dados etc.

Com o intuito de endereçar o problema exposto acima, o autores propoem uma linguagem de padrões para agrupar e relacionar as várias categorias de padrões de projetos. Esta linguagem visa prover uma forma de relacionar os padrões arquiteturais através de visões arquiteturais.

Visão arquitetural, segundo os autores, é uma forma de representar o sistema através de um conjunto comum de interesses, formando uma instância de um pontos de vista arquitetural (*ViewPoint*). Um ponto de vista arquitetural consiste nos tipos dos elementos e seus relacionamentos. Neste contexto, padrão arquitetural consiste em uma especialização de uma visão arquitetural, uma vez que define aspectos semânticos dos elementos da arquitetura e de seus relacionamentos.

Ao todo, oito visões são apresentadas pelos autores, como constituintes da linguagem apresentada:

- Layered View: Visão que trata da decomposição do sistema em partes que interagem entre si.
- Data Flow View: Visão relacionada ao tratamento dos *streams* de dados.

- Data-centered View: Esta visão está relacionada ao uso de um repositório central acessado por vários componentes.
- Adaptation View: Visão que trata da abstração provida pelo sistema sobre a camada de infra-estrutura.
- User interaction View: Visão que trata do relacionamento entre os componentes para prover interface com o usuário.
- Component Interaction View: Visão relacionado à comunicação (troca de mensagens etc) entre os componentes do sistema.
- Distribution View: Trata da distribuição dos componentes em um ambiente distribuído.

1.1 Opinião/Crítica

Por se tratar de um artigo mais expositivo, não há como tomar um posicionamento em relação ao assunto abordado. Por isso, pretendo neste espaço apenas comentar a intensão dos autores ao escreverem o artigo.

Acredito que trata-se de um trabalho importante no que tange à busca de um senso comum na catalogação de padrões arquiteturais. A taxonomia das visões é algo a se elogiar no trabalho efetuado. De fato, é preciso saber separar os interesses para que possamos aplicar o padrão arquitetural apropriado ao contexto do problema a ser resolvido. No entanto, há de se estudar quão adotada foi a linguagem proposta pelos autores. Até onde pesquisei, a adoção não foi bem sucedida, uma vez que não há relatos de muitos projetos que a usaram na descrição de sua arquitetura. Além disso, um trabalho futuro seria refinar esta linguagem para que sua adoção seja maior tanto na academia, para que possa ser estudada mais a fundo, quanto no mercado, para que possa realmente trazer benefícios aos engenheiros de *software*.

Referências

- [1] P. Avgeriou and U. Zdun. Architectural patterns revisited—a pattern language. *10th European Conference on Pattern Languages of Programs (EuroPlop 2005)*.