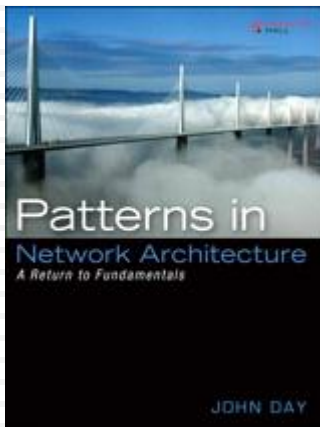


1

Fundamentos para a Arquitetura de Redes

Capítulo 1

Patterns in Network Architecture



Introdução

2

- **Motivação:** identificar um pequeno número de princípios que possam levar a uma teoria unificada de redes!
- A diferença principal entre a Ciência da Computação e outros campos científicos é que: “Nós construímos aquilo que medimos”!
- **Hipótese:**
 - ▣ Princípios são independentes das implementações

Começando do início (um pouco de filosofia)

3

- *Tractatus Logico-Philosophicus* (Ludwig Wittgenstein, 1921):
 - O mundo é tudo o que é o caso.
 - O mundo é a totalidade dos fatos, não das coisas.
 - O mundo é determinado por ser os fatos e por ser todos os fatos.

Começando do início (um pouco de filosofia)

4

- Relação entre proposições lógicas e realidade:
 - ▣ O que é o caso – um fato – é a existência do estado atual.
 - ▣ Um retrato lógico dos fatos é um pensamento.
 - ▣ Um pensamento é uma proposição com um sentido.
 - ▣ Uma proposição é uma função-verdade de proposições elementares.
 - ▣ A forma geral de uma função-verdade é $(\rho, \varepsilon N(\varepsilon))$. Esta é a forma geral de uma proposição.

Começando do início (um pouco de filosofia)

5

- Conclui com talvez a afirmação mais revolucionária e devastadora jamais feita em filosofia, matemática, ou ciência:
 - ▣ Aquilo do qual não podemos falar, devemos passar em silêncio.

Começando do início (um pouco de filosofia)

6

- Conclui com talvez a afirmação mais revolucionária e devastadora jamais feita em filosofia, matemática, ou ciência:
 - ▣ Aquilo do qual não podemos falar, devemos passar em silêncio.

Começando do início (um pouco de filosofia)

7

- O *Tractatus* foi abraçado pela Matemática e Ciências como podendo descrevê-las por um sistema lógico preciso tão completo como a Geometria de Euclides, os Princípios de Newton, ou o Tratado de Maxwell.
- Em computação foi usado como:
 - ▣ Base para a lógica e linguagens de programação
 - ▣ Base para IA e sistemas de BD
 - BD: esquema conceitual
 - IA: base de conhecimento

Começando do início (um pouco de filosofia)

8

- A mesma abordagem foi utilizada para Sistemas Distribuídos:
 - ▣ Para que dois parceiros possam se comunicar, eles devem possuir um esquema conceitual compartilhado
 - ▣ Devem ter uma linguagem comum ou protocolo e
 - ▣ Uma compreensão comum sobre o que significam os símbolos da linguagem.

Começando do início (um pouco de filosofia)

- O esquema conceitual compartilhado de máquinas de estado de protocolo são informações trocadas sobre:
 - ▣ Controle de fluxo
 - ▣ Reconhecimentos
 - ▣ Endereços,
 - ▣ Etc.
- Os dados dos usuários estavam fora do esquema conceitual e era ignorado (ou são passados para outra máquina de protocolo).

Começando do início (um pouco de filosofia)

10

- Estamos principalmente interessados naquilo que o *Tractatus* nos ensina sobre reduzir os problemas ao seu essencial.
- A penúltima afirmação do *Tractatus* é:
 - ▣ Minhas proposições servem como elucidações da seguinte forma: qualquer um que me entenda eventualmente as reconhecerá como sem sentido, quando tivê-las usado – como degraus – para subir além delas. (Ele deve, de certo modo, jogar fora a escada depois que a tiver subido).
- Procuraremos também fazer este exercício.

Níveis de Abstração

- Níveis de abstração representam uma ferramenta importante para gerenciar a complexidade de um sistema ou arquitetura.
- Não confundir com as camadas do sistema que está sendo projetado.
- Do ponto de vista do projeto, toda a implementação está no mesmo nível de abstração.
- Duas formas ortogonais de abstração são úteis em arquitetura:
 - ▣ Níveis do projeto e especificação e
 - ▣ Divisão em camadas da arquitetura.

Divisão em Camadas

- ❑ Evoluiu no final dos anos 60 e início dos anos 70 no projeto de sistemas operacionais e software em geral.
- ❑ Expansão do conceito de “caixa preta”.
- ❑ Em SO cada camada adicionava um nível de funcionalidade começando com o hardware na camada mais baixa e o usuário na camada mais alta.
- ❑ Da mesma forma, a divisão em camadas em redes, provê uma abstração sobre a tecnologia específica de hardware da rede.
- ❑ Qual é o esquema “certo” para estas “camadas”?

Projeto e Especificação da Arquitetura da Rede

13

- Níveis de abstração:
 - Modelo
 - Serviço
 - Protocolo e interface
 - Implementação.

O que é uma boa arquitetura?

14

- Definição comum de dicionário:
 - ▣ Um conjunto de regras e restrições que caracterizam um estilo particular de construção

- Houaiss (4^a. Definição):
 - ▣ Conjunto de princípios, normas, materiais e técnicas usadas para criar o espaço arquitetônico

Modelo

15

- O modelo em essência define o esquema conceitual compartilhado da comunicação.
- Define os objetos no universo do discurso, seus atributos, as operações que podem ser realizadas sobre eles, como se relacionam, a comunicação de informações entre eles, etc.

Serviço

16

- Um dos conceitos mais importantes no desenvolvimento de arquiteturas de comunicação.
- Definido aqui como uma abstração da interface entre as camadas que é independente do sistema.
- É um nível de abstração acima de protocolo e interface, mas abaixo da arquitetura ou modelo.

Serviço

17

- O serviço **NÃO** é uma interface!
 - ▣ Uma interface é local a um determinado sistema.
 - ▣ O conceito de serviço é usado para descrever os aspectos de uma interface que todas as interfaces devem ter independentemente de como as questões locais são resolvidas.
- Um serviço é definido para ser independente do protocolo.

Definição de Serviço

18

- **Consiste em:**
 - ▣ **Um conjunto de primitivas de serviço** que especifica as operações a serem executadas no serviço e um conjunto de parâmetros que são usados como argumento para as operações.
 - ▣ **Um conjunto de regras** que determinam as sequências legais nas quais as primitivas de serviço podem ser invocadas.

Definição de Serviço

- Uma definição de serviço é definida para uma camada.
- É especificada como uma máquina parcial de estados para a interação do usuário da camada e a máquina de protocolo (PM) na camada de um único sistema.
- Nunca afirma explicitamente que uma primitiva de serviço submetida a uma PM provoca o aparecimento de uma outra primitiva de serviço na PM parceira.
 - ▣ Qualquer associação deste tipo é feita pelo protocolo.

Protocolo e Interface

- Este é provavelmente o nível de abstração mais importante.
- A especificação de um *protocolo* define as regras e o comportamento requerido por qualquer entidade que participe da transferência de dados.
- Define a sequência de troca de mensagens entre os participantes.
- Nível de abstração entre o modelo e o serviço e a implementação.

Protocolo e Interface

- Um protocolo especifica a máquina de estados mínima à qual qualquer implementação deve obedecer.
- Especificações de protocolos devem ser tomadas como documentos de requisitos e não especificações de projeto.
- A especificação de um protocolo não deveria divergir muito da estratégia de implementação “normal”.
- Os testes de conformidade deveriam se restringir aos comportamentos que sejam visíveis externamente.
- APIs padrão podem ser definidas para linguagens e SO específicos em junções importantes da arquitetura.

Implementação

22

- Nível mais baixo de abstração.
- Deve endereçar todas as questões não cobertas pela especificação do protocolo:
 - ▣ Estratégias de armazenamento local
 - ▣ Interações com o sistema operacional,
 - ▣ Etc.

Especificando Protocolos

23

- Especificar protocolos e interface são passos críticos.
 - ▣ Especificações informais
 - ▣ Técnicas de descrição formais (FDTs):
 - Estelle – *Extended Finite State Machine Language*
 - LOTOS – *Language Temporal Ordering Specification*
 - SDL – *Specification and Definition Language*

FDTs

- ❑ Não deve ser mais complexa do que a linguagem de programação usada para a implementação.
- ❑ Deve ser útil como ferramenta de projeto.
- ❑ É apenas outra forma de programação.
- ❑ Devem existir ferramentas analíticas que possam analisar a descrição formal para assegurar a ausência de impasse, perda de dados, corridas e outros comportamentos patológicos.
- ❑ Não deve necessitar de mais especificidade do que necessário.
- ❑ Deve ser fácil ir da especificação informal para a formal.

FDTs: Formas das técnicas e linguagens

25

- Matemática ou baseada em linguagens.
- Máquina de estados finita
- Lógica temporal