Fatec
Araras

Engenharia de Software I

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br

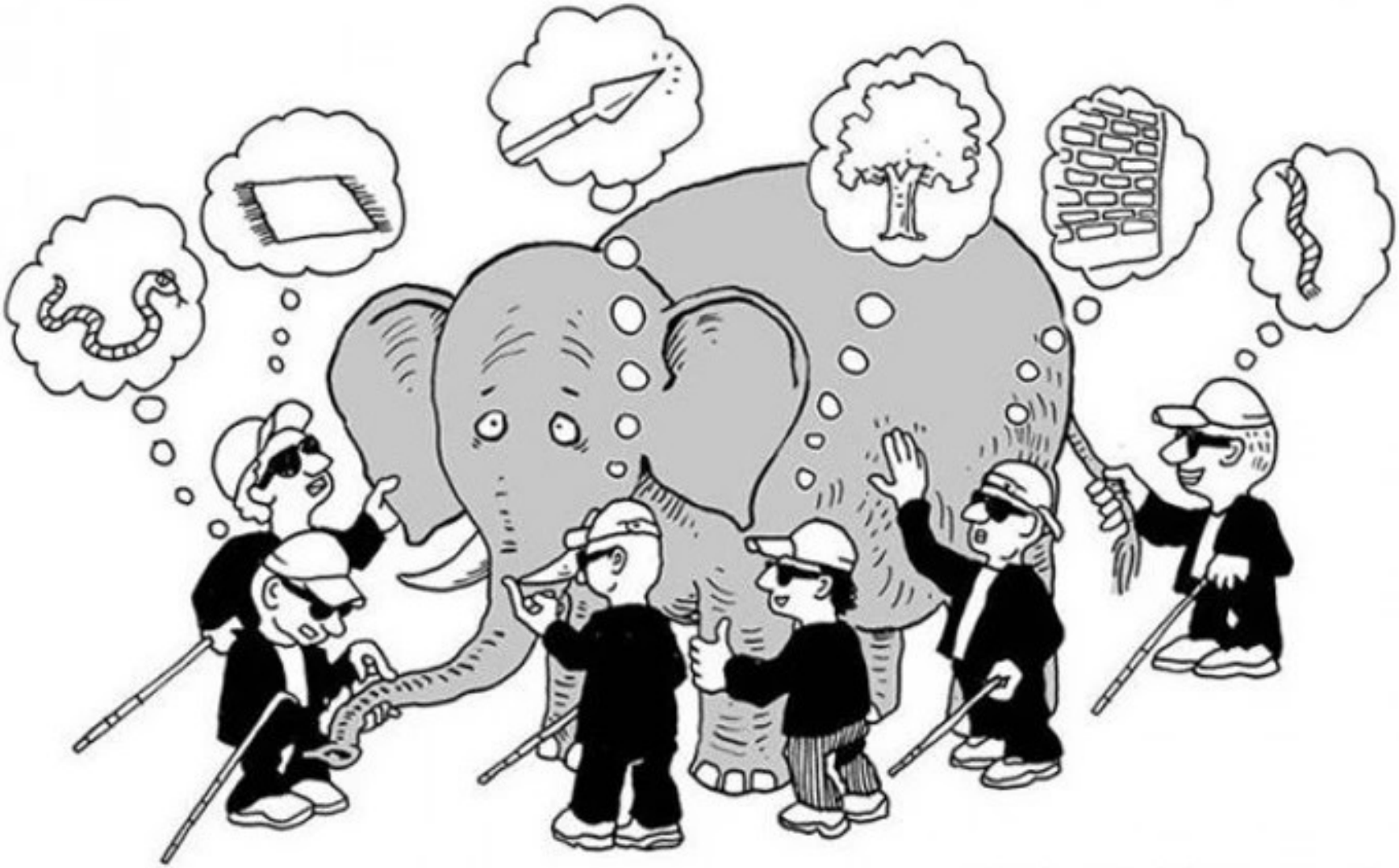


Illustration: Hans Møller, mollers.dk

Um sistema de software difícil de desenvolver provavelmente não terá uma vida útil longa e saudável.

Portanto, a arquitetura deve facilitar o desenvolvimento desse sistema pelas equipes de desenvolvedores

Robert C. Martin

* Livro “Arquitetura Limpa” página 137

Arquitetura

Considerações sobre arquiteto de software

Fatec Araras

Um arquiteto de software é e continua a ser um programador. Nunca caia na mentira de que um arquiteto deve abandonar o código para tratar de questões de nível mais alto.

Os arquitetos de software podem não escrever tanto código quanto os outros programadores, mas continuam a se envolver em tarefas de programação. Isso porque eles não podem fazer o seu trabalho de maneira adequada se não vivenciarem os problemas que criam para os demais programadores.

Arquitetura

Fatec Araras

Há muitos sistemas por aí que funcionam muito bem apesar das suas arquiteturas terríveis.

Seus problemas não estão na operação: ocorrem na sua implementação, manutenção e desenvolvimento continuado. O propósito primário da arquitetura é suportar o ciclo de vida do sistema.

Uma boa arquitetura torna o sistema fácil de entender, fácil de desenvolver, fácil de manter e fácil de implementar. O objetivo final é minimizar o custo da vida útil do sistema e não maximizar a produtividade do programador.

Arquitetura

Fatec Araras

A arquitetura de um sistema de software é a forma dada a esse sistema pelos seus criadores.

Essa forma está na divisão desse sistema em componentes, na organização desses componentes e nos modos como esses componentes se comunicam entre si.

O propósito essa forma é facilitar o desenvolvimento, implantação, operação e manutenção do sistema de software contido nela.

A estratégia por trás dessa facilitação é deixar o máximo possível de tempo de opções abertas, pelo máximo de tempo possível.

Mantendo as opções abertas

Fatec Araras

Todos os sistemas de software podem ser decomposto em dois elementos principais: política e detalhe.

O elemento político engloba todas as regras e procedimentos de negócios. A política é onde está o verdadeiro valor do sistema.

Os detalhes são itens necessários para que os seres humanos, outros sistemas e programadores se comuniquem com a política, mas não causam impacto algum sobre o comportamento da política.

Mantendo as opções abertas

O objetivo do arquiteto é criar uma forma para o sistema que reconheça a política como o elemento mais essencial do sistema e torne os detalhes irrelevantes para essa política. Isso permite que as decisões para esses detalhes sejam adiadas e diferidas.

Quanto mais tempo você deixar as opções abertas, mais experimentos poderá executar, mais coisas poderá testar e mais informações terá quando atingir o ponto no qual essas decisões não poderão ser mais adiadas.

Projeto de Arquitetura

Projeto de Arquitetura

Fatec Araras

O projeto de arquitetura está preocupado com a compreensão de como um sistema deve ser organizado e com a estrutura geral desse sistema.

Segundo *Sommerville*, no modelo do processo de desenvolvimento de software, o projeto de arquitetura é o primeiro estágio no processo de projeto de software.

Projeto de Arquitetura

Fatec Araras

É o elo crítico entre o projeto e a engenharia de requisitos, pois identifica os principais componentes estruturais de um sistema e os relacionamentos entre eles.

O resultado do processo de projeto de arquitetura é um modelo de arquitetura que descreve como o sistema está organizado em um conjunto de componentes de comunicação.

Projeto de Arquitetura

Em processos ágeis, geralmente se aceita que um estágio inicial do processo de desenvolvimento se preocupe com o estabelecimento de uma arquitetura global do sistema.

O desenvolvimento incremental de arquiteturas geralmente não é bem-sucedido, embora a refatoração de compontentes em resposta às mudanças costume ser relativamente fácil, a refatoração de uma arquitetura é geralmente cara.

Projeto de Arquitetura

Fatec Araras

Na prática, existe uma considerável sobreposição entre os processos de engenharia de requisitos e de projeto de arquitetura.

Idealmente, uma especificação de sistema não deve incluir todas as informações do projeto. Mas essa não é a realidade, exceto para sistemas muito pequenos.

A decomposição de arquitetura é normalmente necessária para estruturar e organizar a especificação.

Projeto de Arquitetura

Fatec Araras

Portanto, como parte do processo de engenharia de requisitos, você poderá propor uma arquitetura abstrata de sistema em que seja possível associar grupos de funções ou recursos do sistema aos componentes em larga escala ou subsistemas.

Você pode, então, usar essa decomposição para discutir com os *stakeholders* os requisitos e recursos do sistema.

Projeto de Arquitetura

Fatec Araras

Você pode projetar as arquiteturas de software em dois níveis de abstração:

1. A arquitetura em pequena escala está preocupada com a arquitetura de programas individuais. Nesse nível, estamos preocupados com a maneira como um programa individual é decomposto em componentes.

2. A arquitetura em grande escala preocupa-se com a arquitetura de sistemas corporativos complexos que incluem outros sistemas, programas e componentes de programas

Projeto de Arquitetura

Fatec Araras

A arquitetura de software é importante, pois afeta o desempenho e a robustez, bem como a capacidade de distribuição e de manutenibilidade de um sistema (BOSCH, 2000).

Os componentes individuais implementam os requisitos funcionais do sistema. Os requisitos não funcionais dependem da arquitetura do sistema - a forma como esses componentes estão organizados e se comunicam.

Em muitos sistemas, os requisitos não funcionais são também influenciados por componentes individuais, mas não há dúvida de que a arquitetura de sistema é a influência dominante.

Projeto de Arquitetura

Fatec Araras

Bass et al. (2003) discutem três vantagens de projetar e documentar, explicitamente, a arquitetura de software:

1. **Comunicação de *stakeholders***. A arquitetura é uma apresentação de alto nível do sistema e pode ser usada como um foco de discussão por uma série de diferentes stakeholders.

2. **Análise de sistema.** Tornar explícita a arquitetura do sistema, em um estágio inicial de seu desenvolvimento, requer alguma análise. As decisões de projeto de arquitetura têm um efeito profundo sobre a possibilidade de o sistema atender ou não aos requisitos críticos, como desempenho, confiabilidade e manutenibilidade.

3. **Reúso em larga escala.** Um modelo de uma arquitetura de sistema é uma descrição compacta e gerenciável de como um sistema é organizado e como os componentes interoperam. A arquitetura do sistema geralmente é a mesma para sistemas com requisitos semelhantes e, por isso, pode apoiar o reúso de software em grande escala.

Decisões de Projeto de Arquitetura

Fatec Araras

O projeto de arquitetura é um processo criativo no qual você projeta uma organização de sistema para satisfazer aos requisitos funcionais e não funcionais de um sistema.

Durante o processo de projeto de arquitetura, os arquitetos de sistema precisam tomar uma série de decisões estruturais que afetam profundamente o sistema e seu processo de desenvolvimento. Com base em seus conhecimentos e experiência, eles precisam considerar as seguintes questões fundamentais:

Decisões de Projeto de Arquitetura

Fatec Araras

1. Existe uma arquitetura genérica de aplicação que pode atuar como um modelo para o sistema que está sendo projetado?
2. Como o sistema será distribuído por meio de um número de núcleos ou processadores?
3. Que padrões ou estilos de arquitetura podem ser usados?
4. Qual será a abordagem fundamental para se estruturar o sistema?
5. Como os componentes estruturais do sistema serão decompostos em subcomponentes?
6. Que estratégia será usada para controlar o funcionamento dos componentes do sistema?
7. Qual a melhor organização de arquitetura para satisfazer os requisitos não funcionais do sistema?
8. Como o projeto de arquitetura será avaliado?
9. Como a arquitetura do sistema deve ser documentada?

Decisões de Projeto de Arquitetura

Fatec Araras

Embora cada sistema de software seja único, sistemas no mesmo domínio de aplicação frequentemente têm arquiteturas similares, que refletem os conceitos fundamentais desse domínio.

Ao projetar uma arquitetura de sistema, você precisa decidir o que seu sistema e as classes de uma aplicação mais gerais têm em comum, e quanto conhecimento dessas arquiteturas de aplicação você pode reusar.

Decisões de Projeto de Arquitetura

Fatec Araras

Um padrão de arquitetura é uma descrição de uma organização do sistema (GARLAN e SHAW, 1993), como uma organização cliente-servidor ou uma arquitetura em camadas.

Os padrões de arquitetura capturam a essência de uma arquitetura que tem sido usada em diferentes sistemas de software. Ao tomar decisões sobre a arquitetura de um sistema, você deve conhecer os padrões comuns, bem como saber onde eles podem ser usados e quais são seus pontos fortes e fracos.

Decisões de Projeto de Arquitetura

Fatec Araras

Devido à estreita relação entre os requisitos não funcionais e a arquitetura do software, o estilo e a estrutura da arquitetura particular que você escolhe para um sistema devem depender:

- Desempenho
- Proteção
- Segurança
- Disponibilidade
- Manutenção

Visões de Arquitetura

Fatec Araras

1. Que visões ou perspectivas são úteis ao se projetar e documentar uma arquitetura de sistema?

2. Quais notações devem ser usadas para se descrever modelos de arquitetura?

É impossível representar todas as informações relevantes sobre a arquitetura de um sistema em um único modelo de arquitetura, pois cada modelo mostra apenas uma visão ou perspectiva do sistema. Pode mostrar como um sistema é decomposto em módulos, como os processos de *run-time* interagem, ou as diferentes formas como são distribuídos os componentes do sistema através de uma rede.

Visões de Arquitetura

Fatec Araras

Na prática, as visões conceituais são, quase sempre, desenvolvidas durante o processo de projeto e são usadas para apoiar a tomada de decisões de arquitetura. Elas são uma maneira de comunicar a essência de um sistema para os diferentes stakeholders.

Vários pesquisadores têm proposto o uso de linguagens de descrição de arquitetura (ADLs, do inglês *Architectural Description Languages*) mais especializadas (BASS et al., 2003) para descrever as arquiteturas de sistema. Os elementos básicos de ADLs são componentes e conectores, que incluem regras e diretrizes para arquiteturas bem formadas. No entanto, devido a sua natureza especializada, especialistas de domínio e aplicação consideram as ADLs difíceis de entender e usar, o que torna difícil avaliar sua utilidade para engenharia de software prática.

Visões de Arquitetura

Fatec Araras

Os usuários de métodos ágeis alegam que, na maior parte do tempo, a documentação detalhada de projeto não é usada. E, portanto, desenvolvê-la seria um desperdício de tempo e dinheiro.

Você deve desenvolver visões úteis para a comunicação, e não se preocupar se sua documentação de arquitetura está completa ou não. No entanto, a exceção é quando você está desenvolvendo sistemas críticos, quando você precisa fazer uma análise detalhada da confiança do sistema.

Padrões de Arquitetura

Padrões de Arquitetura

Fatec Araras

A ideia de padrões como uma forma de apresentar, compartilhar e reusar o conhecimento sobre sistemas de software é hoje amplamente usada.

O gatilho para isso foi a publicação de um livro sobre os padrões de projeto orientado a objetos(GAMMA et al., 1995), o que levou ao desenvolvimento de outros tipos de padrões, como padrões para o projeto organizacional (COPLIEN e HARRISON, 2004), padrões de usabilidade (USABILITY GROUP, 1998), interação (MARTIN e SOMMERVILLE, 2004), gerenciamento de configuração (BERCZUK e APPLETON, 2002), e assim por diante.

Padrões de Arquitetura

Fatec Araras

Você pode pensar em um padrão de arquitetura como uma descrição abstrata, estilizada, de boas práticas experimentadas e testadas em diferentes sistemas e ambientes. Assim, um padrão de arquitetura deve descrever uma organização de sistema bem-sucedida em sistemas anteriores. Deve incluir informações de quando o uso desse padrão é adequado, e seus pontos fortes e fracos.

Padrões de Arquitetura
MVC

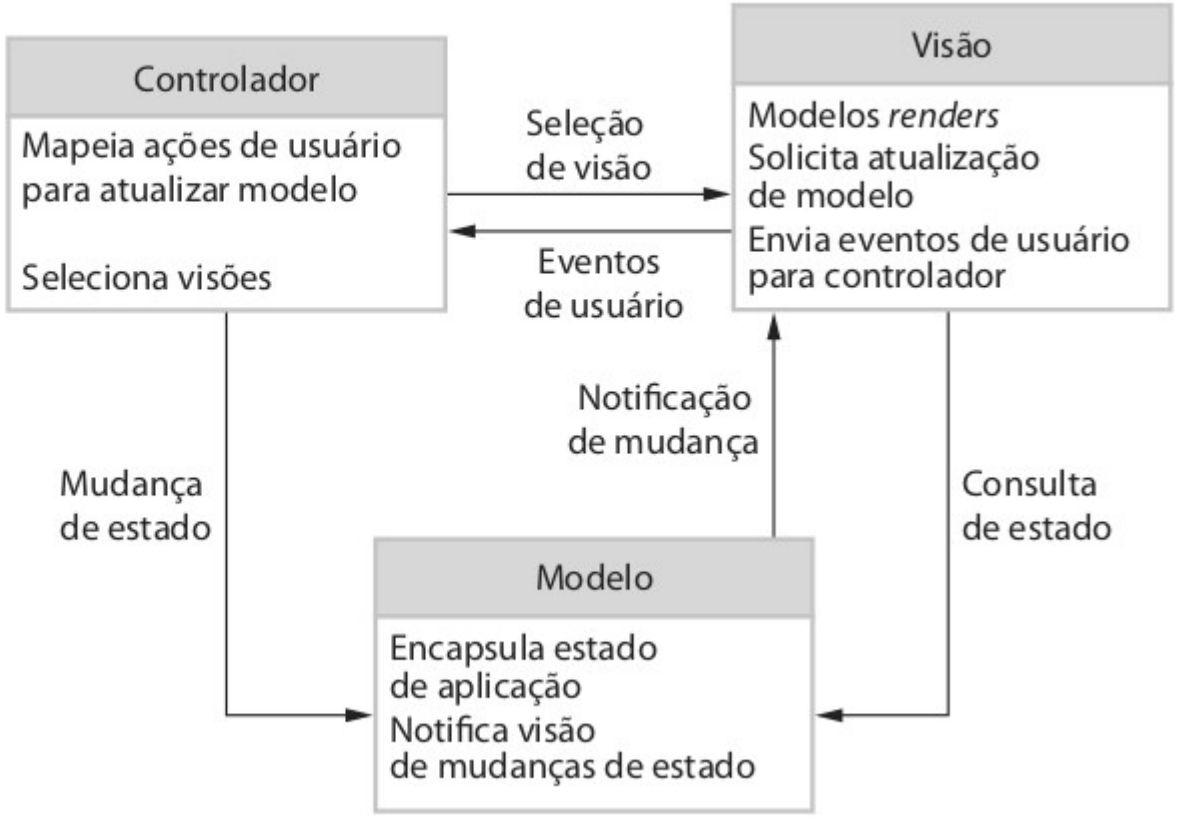
Fatec Araras

Nome	MVC (Modelo-Visão-Controlador)
Descrição	Separa a apresentação e a interação dos dados do sistema. O sistema é estruturado em três componentes lógicos que interagem entre si. O componente Modelo gerencia o sistema de dados e as operações associadas a esses dados. O componente Visão define e gerencia como os dados são apresentados ao usuário. O componente Controlador gerencia a interação do usuário (por exemplo, teclas, cliques do mouse etc.) e passa essas interações para a Visão e o Modelo. Veja a Figura 6.2.
Exemplo	A Figura 6.3 mostra a arquitetura de um sistema aplicativo baseado na Internet, organizado pelo uso do padrão MVC.
Quando é usado	É usado quando existem várias maneiras de se visualizar e interagir com dados. Também quando são desconhecidos os futuros requisitos de interação e apresentação de dados.
Vantagens	Permite que os dados sejam alterados de forma independente de sua representação, e vice-versa. Apoia a apresentação dos mesmos dados de maneiras diferentes, com as alterações feitas em uma representação aparecendo em todas elas.
Desvantagens	Quando o modelo de dados e as interações são simples, pode envolver código adicional e complexidade de código.

Padrões de Arquitetura

Fatec Araras

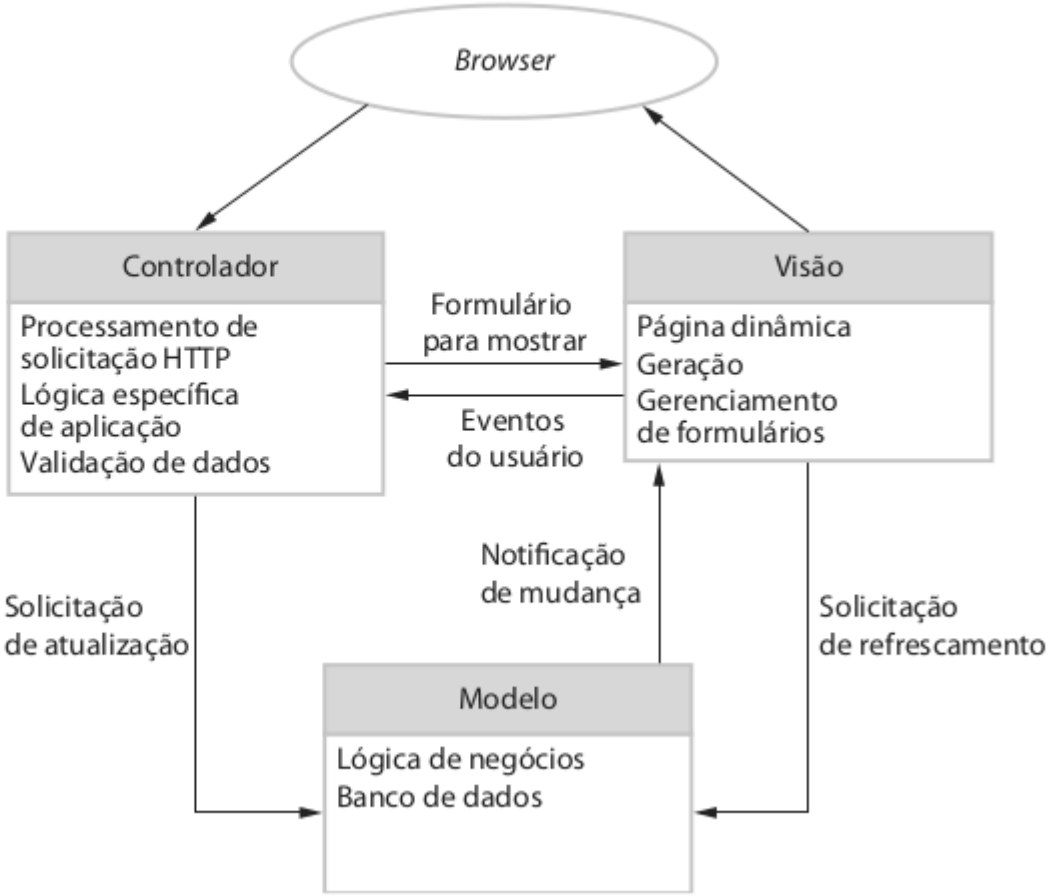
Figura 6.2 A organização do MVC



Padrões de Arquitetura

Fatec Araras

Figura 6.3 Arquitetura de aplicações Web usando o padrão MVC



Padrões de Arquitetura
Arquitetura em Camada

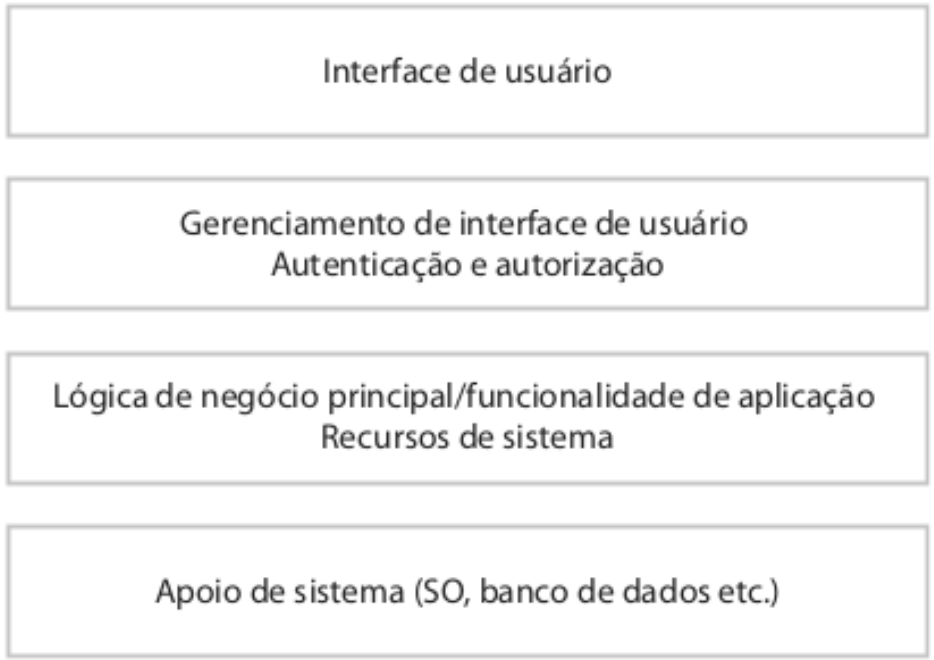
Fatec Araras

Nome	Arquitetura em camadas
Descrição	Organiza o sistema em camadas com a funcionalidade relacionada associada a cada camada. Uma camada fornece serviços à camada acima dela; assim, os níveis mais baixos de camadas representam os principais serviços suscetíveis de serem usados em todo o sistema. Veja a Figura 6.4.
Exemplo	Um modelo em camadas de um sistema para compartilhar documentos com direitos autorais, em bibliotecas diferentes, como mostrado na Figura 6.5.
Quando é usado	É usado na construção de novos recursos em cima de sistemas existentes; quando o desenvolvimento está espalhado por várias equipes, com a responsabilidade de cada equipe em uma camada de funcionalidade; quando há um requisito de proteção multinível.
Vantagens	Desde que a interface seja mantida, permite a substituição de camadas inteiras. Recursos redundantes (por exemplo, autenticação) podem ser fornecidos em cada camada para aumentar a confiança do sistema.
Desvantagens	Na prática, costuma ser difícil proporcionar uma clara separação entre as camadas, e uma camada de alto nível pode ter de interagir diretamente com camadas de baixo nível, em vez de através da camada imediatamente abaixo dela. O desempenho pode ser um problema por causa dos múltiplos níveis de interpretação de uma solicitação de serviço, uma vez que são processados em cada camada.

Padrões de Arquitetura

Fatec Araras

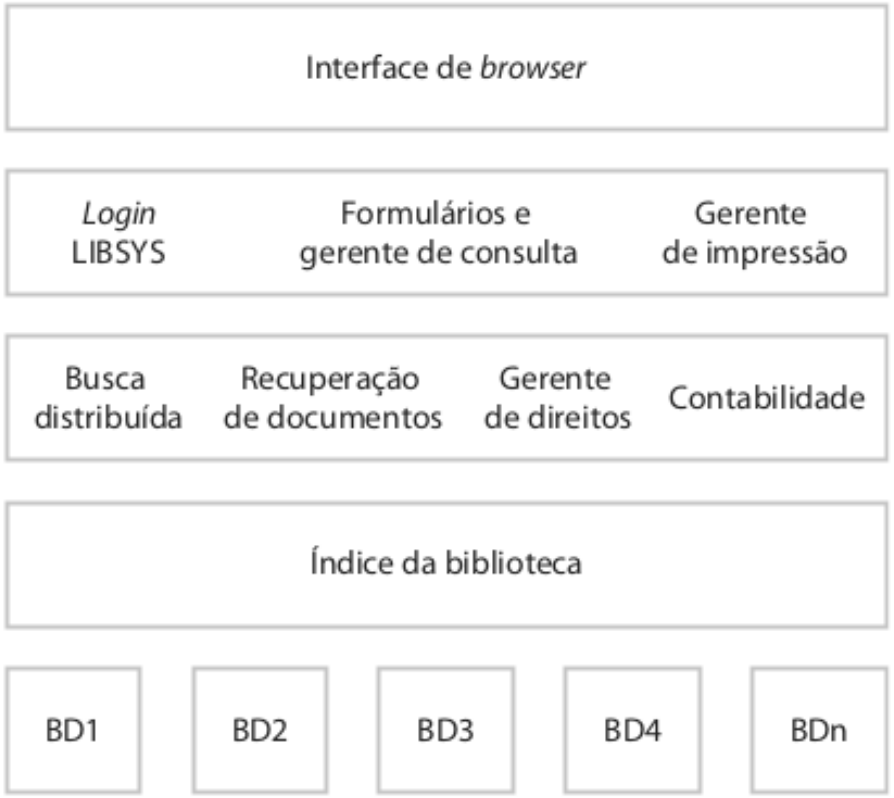
Figura 6.4 Uma arquitetura genérica em camadas



Padrões de Arquitetura

Fatec Araras

Figura 6.5 A arquitetura do sistema LIBSYS



Padrões de Arquitetura

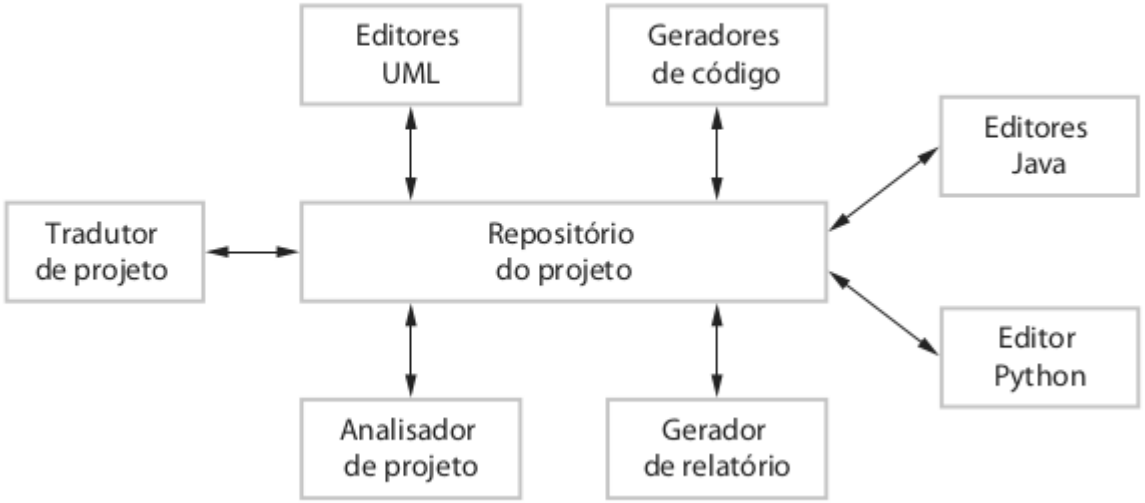
Fatec Araras

Nome	Repositório
Descrição	Todos os dados em um sistema são gerenciados em um repositório central, acessível a todos os componentes do sistema. Os componentes não interagem diretamente, apenas por meio do repositório.
Exemplo	A Figura 6.6 é um exemplo de um IDE em que os componentes usam um repositório de informações sobre projetos de sistema. Cada ferramenta de software gera informações que ficam disponíveis para uso por outras ferramentas.
Quando é usado	Você deve usar esse padrão quando tem um sistema no qual grandes volumes de informações são gerados e precisam ser armazenados por um longo tempo. Você também pode usá-lo em sistemas dirigidos a dados, nos quais a inclusão dos dados no repositório dispara uma ação ou ferramenta.
Vantagens	Os componentes podem ser independentes — eles não precisam saber da existência de outros componentes. As alterações feitas a um componente podem propagar-se para todos os outros. Todos os dados podem ser gerenciados de forma consistente (por exemplo, *backups* feitos ao mesmo tempo), pois tudo está em um só lugar.
Desvantagens	O repositório é um ponto único de falha, assim, problemas no repositório podem afetar todo o sistema. Pode haver ineficiências na organização de toda a comunicação através do repositório. Distribuir o repositório através de vários computadores pode ser difícil.

Padrões de Arquitetura

Fatec Araras

Figura 6.6 Uma arquitetura de repositório para um IDE



Padrões de Arquitetura
Cliente Servidor

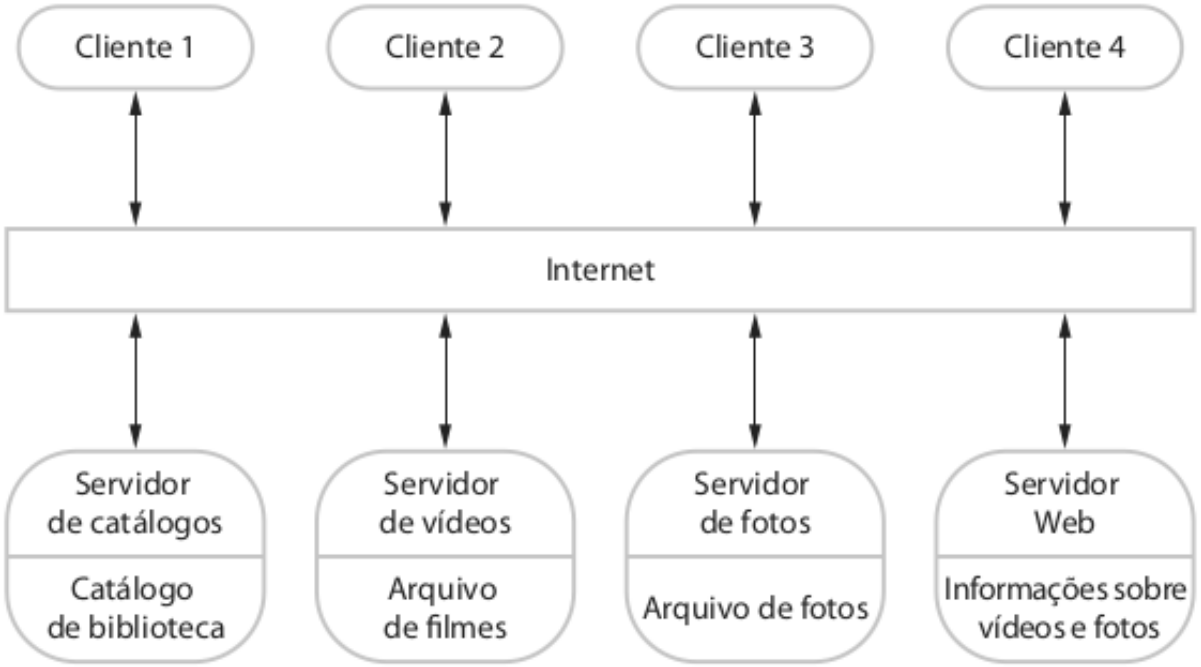
Fatec Araras

Nome	Cliente-servidor
Descrição	Em uma arquitetura cliente-servidor, a funcionalidade do sistema está organizada em serviços — cada serviço é prestado por um servidor. Os clientes são os usuários desses serviços e acessam os servidores para fazer uso deles.
Exemplo	A Figura 6.7 é um exemplo de uma biblioteca de filmes e vídeos/DVDs, organizados como um sistema cliente-servidor.
Quando é usado	É usado quando os dados em um banco de dados compartilhado precisam ser acessados a partir de uma série de locais. Como os servidores podem ser replicados, também pode ser usado quando a carga em um sistema é variável.
Vantagens	A principal vantagem desse modelo é que os servidores podem ser distribuídos através de uma rede. A funcionalidade geral (por exemplo, um serviço de impressão) pode estar disponível para todos os clientes e não precisa ser implementada por todos os serviços.
Desvantagens	Cada serviço é um ponto único de falha suscetível a ataques de negação de serviço ou de falha do servidor. O desempenho, bem como o sistema, pode ser imprevisível, pois depende da rede. Pode haver problemas de gerenciamento se os servidores forem propriedade de diferentes organizações.

Padrões de Arquitetura

Fatec Araras

Figura 6.7 Uma arquitetura cliente-servidor para uma biblioteca de filmes



Padrões de Arquitetura
Duto e filtro

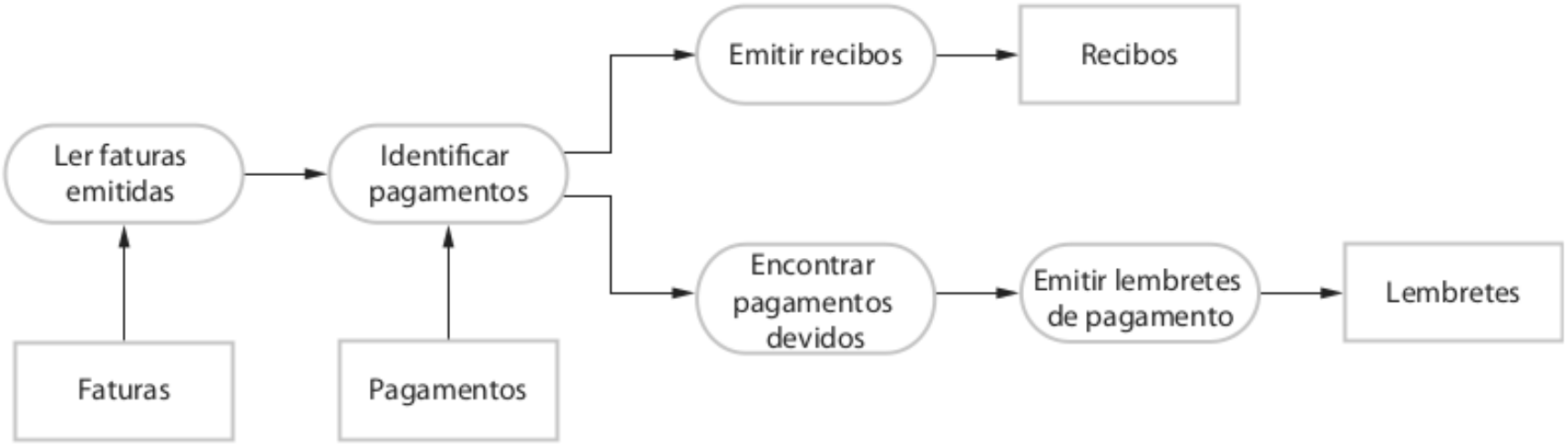
Fatec Araras

Nome	Duto e filtro
Descrição	O processamento dos dados em um sistema está organizado de modo que cada componente de processamento (filtro) seja discreto e realize um tipo de transformação de dados. Os dados fluem (como em um duto) de um componente para outro para processamento.
Exemplo	A Figura 6.8 é um exemplo de um sistema de duto e filtro usado para o processamento das faturas.
Quando é usado	Comumente, é usado em aplicações de processamento de dados (tanto as baseadas em lotes como as baseadas em transações) em que as entradas são processadas em etapas separadas para gerarem saídas relacionadas.
Vantagens	O reúso da transformação é de fácil compreensão e suporte. Estilo de *workflow* corresponde à estrutura de muitos processos de negócios. Evolução por adição de transformações é simples. Pode ser implementado tanto como um sistema sequencial quanto concorrente.
Desvantagens	O formato para transferência de dados tem de ser acordado entre as transformações de comunicação. Cada transformação deve analisar suas entradas e gerar as saídas para um formato acordado. Isso aumenta o *overhead* do sistema e pode significar a impossibilidade de reúso de transformações funcionais que usam estruturas incompatíveis de dados.

Padrões de Arquitetura

Fatec Araras

Figura 6.8 Um exemplo da arquitetura duto e filtro



Prática

Prática 1

https://en.wikipedia.org/wiki/Tanenbaum%E2%80%93Torvalds_debate

Fatec
Araras

Dúvidas

Prof. Orlando Saraiva Júnior
orlando.nascimento@fatec.sp.gov.br

Síntese

- Uma arquitetura de software é uma descrição de como um sistema de software é organizado. As propriedades de um sistema, como desempenho, proteção e disponibilidade, são influenciadas pela arquitetura adotada.
- As decisões de projeto de arquitetura incluem decisões sobre o tipo de aplicação, a distribuição do sistema, os estilos da arquitetura a serem utilizados e as formas como a arquitetura deve ser documentada e avaliada.

Síntese

- As arquiteturas podem ser documentadas a partir de diferentes perspectivas ou visões. As possíveis visões incluem uma visão conceitual, uma visão lógica, uma visão de processo, uma visão de desenvolvimento e uma visão física.
- Os padrões da arquitetura são um meio de reusar o conhecimento sobre as arquiteturas genéricas de sistemas.
- Eles descrevem a arquitetura, explicam quando elas podem ser usadas e discutem suas vantagens e desvantagens.
- Entre os padrões de arquitetura comumente usados estão: Modelo-Visão-Controlador, Arquitetura em camadas, Repositório, Cliente-servidor e Duto e filtro.