Análise e Projeto de Sistemas

Universidade Federal do Ceará – UFC
Campus de Quixadá
Curso de Sistemas de Informação
Prof. Lincoln Souza Rocha (lincolnrocha@ufc.br)

“Nada que é visto, é visto de uma vez e por completo.”

(Euclides)

ARQUITETURA DE SISTEMA

Esses slides são uma adaptação das notas de aula do professor Eduardo Bezerra autor do livro Princípios de Análise e Projeto de Sistemas com UML

Índice

- Introdução
- Arquitetura Lógica
- Arquitetura Física
- Projeto de Arquitetura do Processo I&I

INTRODUÇÃO

Introdução

- Em um SSOO, os objetos interagem entre si através do envio de mensagens com o objetivo de executar suas tarefas
  - Um SSOO também pode ser visto como um conjunto de subsistemas que o compõem

- A definição dos subsistemas de um SSOO é feita no ***projeto da arquitetura*** ou ***projeto arquitetural***

- Essa atividade define de que forma o sistema se divide em partes e quais são as interfaces entre essas partes

Introdução

- Vantagens de dividir um SSOO em subsistemas
  - Produzir unidades menores de desenvolvimento
  - Maximizar o reuso no nível de subsistemas componentes
  - Ajuda a gerenciar a complexidade no desenvolvimento

Introdução

- Questões relacionadas à arquitetura de um sistema
  - Como um sistema é decomposto em subsistemas, e como as suas classes são dispostas pelos diversos subsistemas?
  - Como subsistemas devem ser dispostos fisicamente quando o sistema tiver de ser implantado? (nós de processamento)

  "Arquitetura de software é a estrutura organizacional do software. Uma arquitetura pode ser recursivamente decomposta em partes que interagem através de interfaces."
- As decisões tomadas para a definição da arquitetura de software influenciam diretamente na forma como um SSOO irá atender a seus requisitos ***não-funcionais***

ARQUITETURA LÓGICA

Arquitetura Lógica

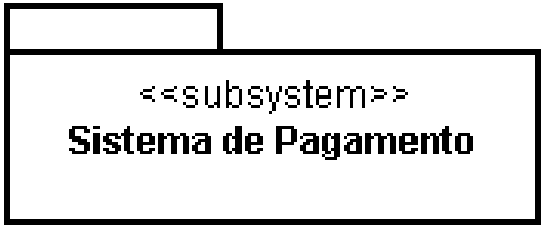
- Chamamos de arquitetura lógica à organização das classes de um SSOO em subsistemas, que correspondem a aglomerados de classes
  - Um SSOO pode ser subdividido em diversos subsistemas

- Um subsistema provê serviços para outros através de sua interface

- A interface de um subsistema corresponde ao conjunto de serviços que ele provê
  - Cada subsistema provê ou utiliza serviços de outros subsistemas

Diagrama de Subsistema

- Uma visão gráfica dos diversos componentes de um SSOO pode ser representada por um ***diagrama de subsistemas***
  - Um diagrama de susbistemas é um diagrama de pacotes, onde cada pacote representa um subsistema
  - Cada subsistema é rotulado com o estereótipo <<subsystem>>



Alocação de Classes a Subsistemas

- Durante o desenvolvimento de um SSOO, seus subsistemas devem ser identificados, juntamente com as interfaces entre eles

- Cada classe do sistema é, então, alocada aos subsistemas

- Uma vez feito isso, esses subsistemas podem ser desenvolvidos quase que de forma independente uns dos outros

Alocação de Classes a Subsistemas

- Dicas para alocação
  - Na fase de análise, considerar as classes mais importantes do modelo de classes de domínio
    - Para cada uma dessas classes, um subsistema é criado
    - Outras classes menos importantes e relacionadas a uma classe considerada importante são posicionadas no subsistema desta última
  - Na fase de projeto
    - Algumas das classes do modelo de domínio podem sofrer decomposições adicionais, o que resulta em novas classes
    - Pode ser que uma classe de domínio seja ela própria um subsistema

Alocação de Classes a Subsistemas

- Subsistemas devem ser minimamente acoplados
- Subsistemas devem ser maximamente coesivos
- Dependências cíclicas entre subsistemas devem ser evitadas
- Uma classe deve ser definida em um único subsistema

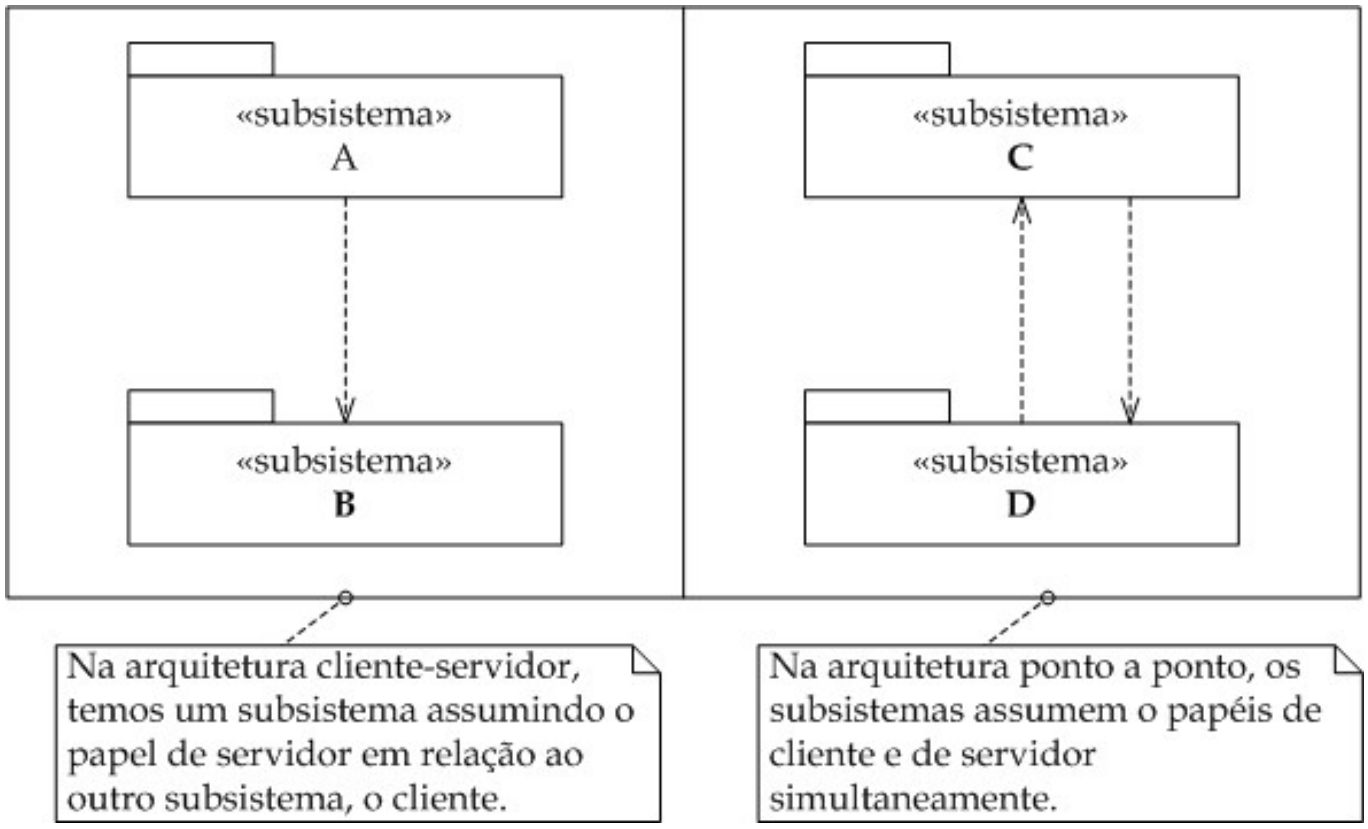
Camadas de Software

- Dizemos que dois subsistemas interagem quando um precisa dos serviços do outro

- Há basicamente duas formas de interação entre subsistemas
  - **Ponto a ponto**: na arquitetura ponto a ponto, a comunicação pode acontecer em duas vias
  - **Cliente-servidor**: há a comunicação somente em uma via entre dois subsistemas, do cliente para o servidor. Nessa arquitetura, chamamos de **camadas** os subsistemas envolvidos

- Essas formas de interação entre subsistemas influenciam a o modo pelo qual os subsistemas são distribuídos fisicamente pelos nós de processamento

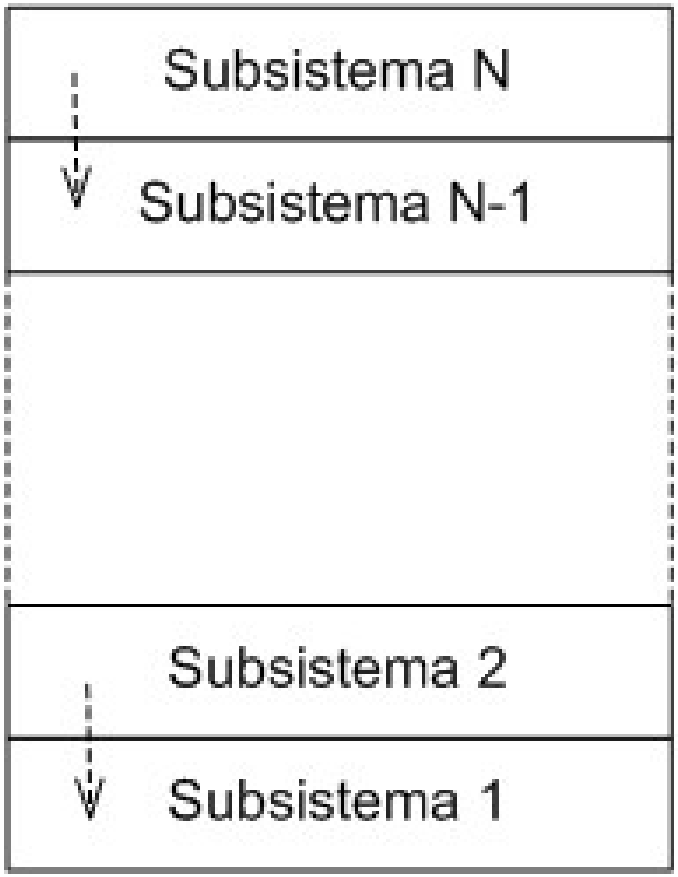
Camadas de Software

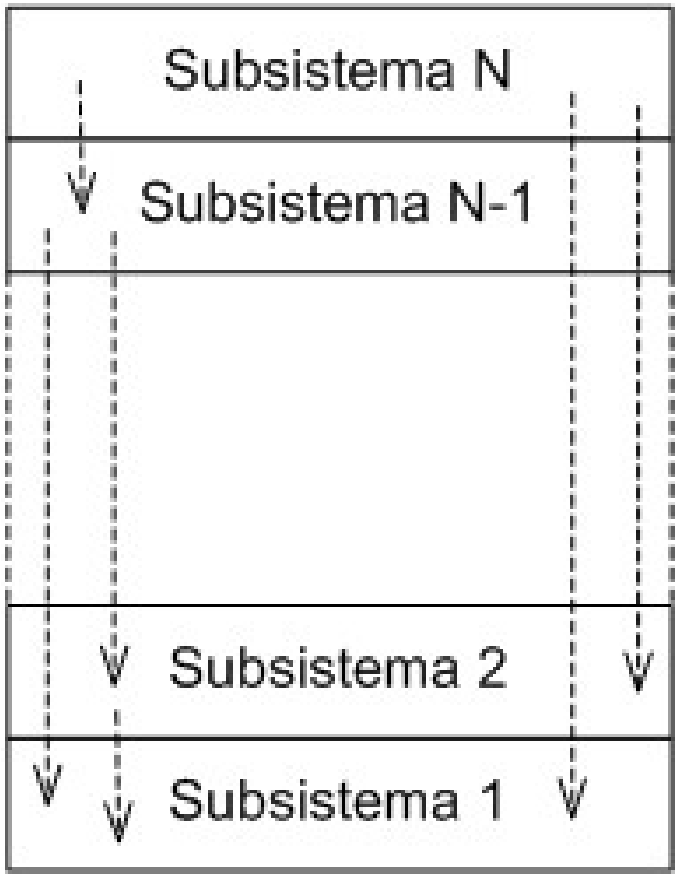


Camadas de Software

- Um SSOO projetado em camadas pode ter uma **arquitetura aberta** ou uma **arquitetura fechada**
  - Em uma arquitetura fechada, um componente de uma camada de certo nível somente pode utilizar os serviços de componentes da sua própria camada ou da imediatamente inferior
  - Em uma arquitetura aberta, uma camada em certo nível pode utilizar os serviços de qualquer camada inferior

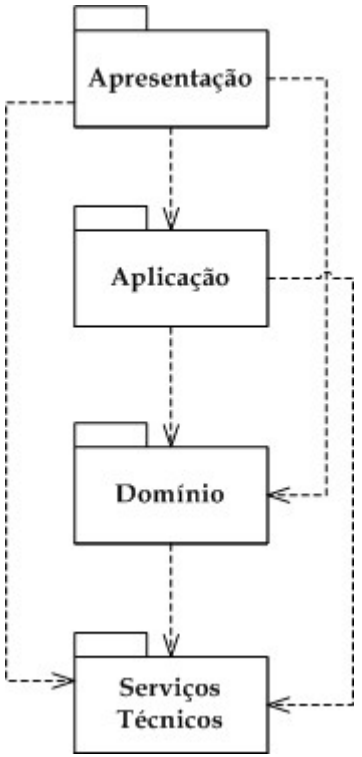
Camadas de Software





Camadas de Software

- Uma divisão tipicamente encontrada para as camadas lógicas de um SSOO é a que separa o sistema nas seguintes camadas
  - Apresentação
  - Aplicação
  - Domínio
  - Serviços técnicos



ARQUITETURA FÍSICA

Arquitetura de Implantação

- Representa a disposição física do sistema de software pelo hardware disponível

- A divisão de um sistema em camadas é independente da sua disposição física
  - As camadas de software podem estar fisicamente localizadas em uma única máquina, ou podem estar distribuídas por diversos processadores
  - Alternativamente, essas camadas podem estar distribuídas fisicamente em vários processadores

- O modelo que representa a arquitetura física é denominado ***modelo de implementação*** ou modelo ***da arquitetura física***

Arquitetura de Implantação

- A arquitetura de implantação diz respeito à disposição dos subsistemas de um SSOO pelos nós de processamento disponíveis

- Para sistemas simples, a arquitetura de implantação não tem tanta importância

- No entanto, na modelagem de sistemas complexos, é fundamental conhecer quais são os componentes físicos do sistema, quais são as interdependências entre eles e de que forma as camadas lógicas do sistema são dispostas por esses componentes

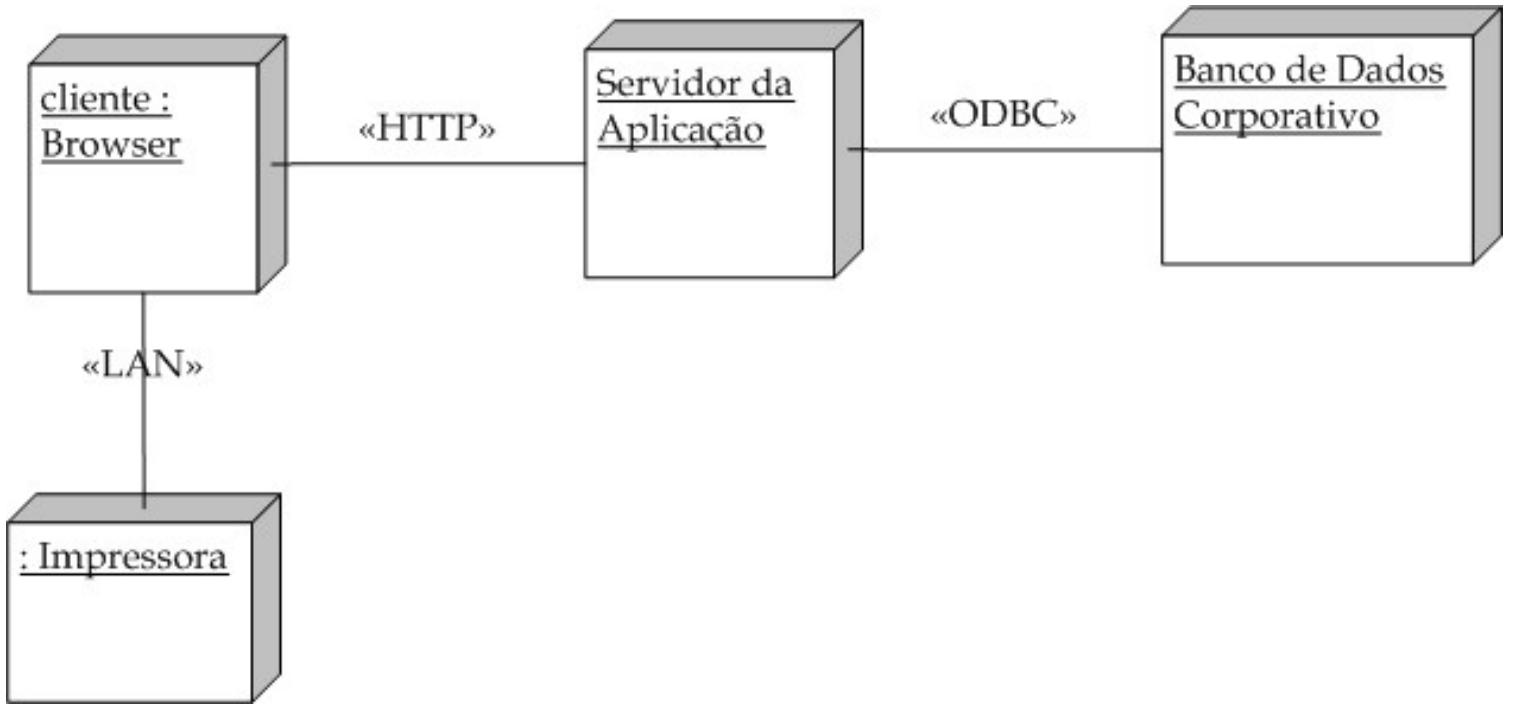
Alocação de Camadas

- Em um sistema construído segundo a arquitetura a cliente-servidor, é comum utilizar as definições das camadas lógicas como um guia para a alocação física dos subsistemas

- Sendo assim, a cada nó de processamento são alocadas uma ou mais camadas lógicas

- Note que o termo camada é normalmente utilizado com dois sentidos diferentes
  - Para significar uma camada lógica (***layer***)
  - Para significar uma camada física, esta última normalmente associada a um nó de processamento (***tier***)

Diagrama de Implantação



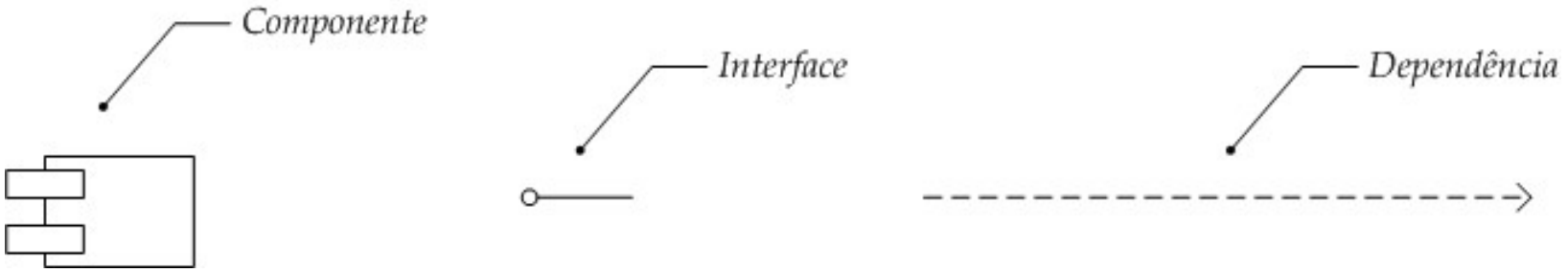
Alocação de Componentes

- Na arquitetura (alocação) física, devemos também definir quais os ***componentes de software*** de cada camada

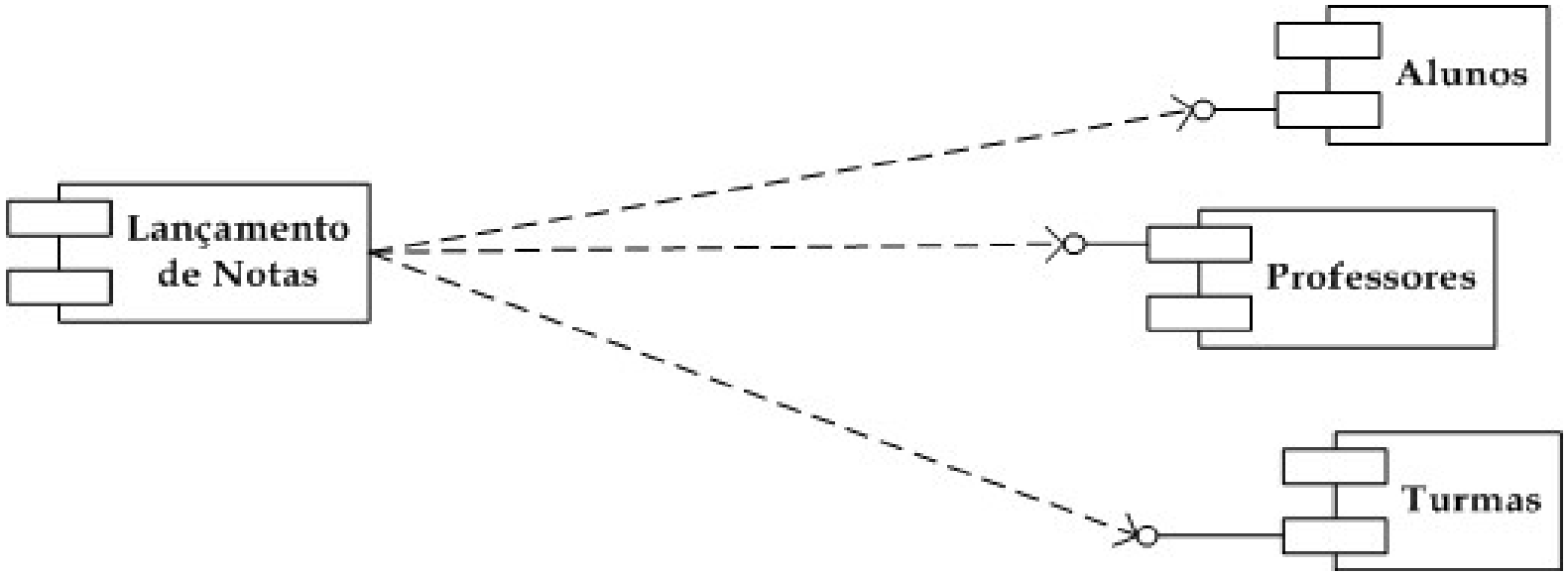
Um componente de software é uma unidade que existe em tempo de execução, que pode ser utilizada na construção de vários sistemas e que pode ser substituída por outra unidade que tenha a mesma funcionalidade.

Alocação de Componentes

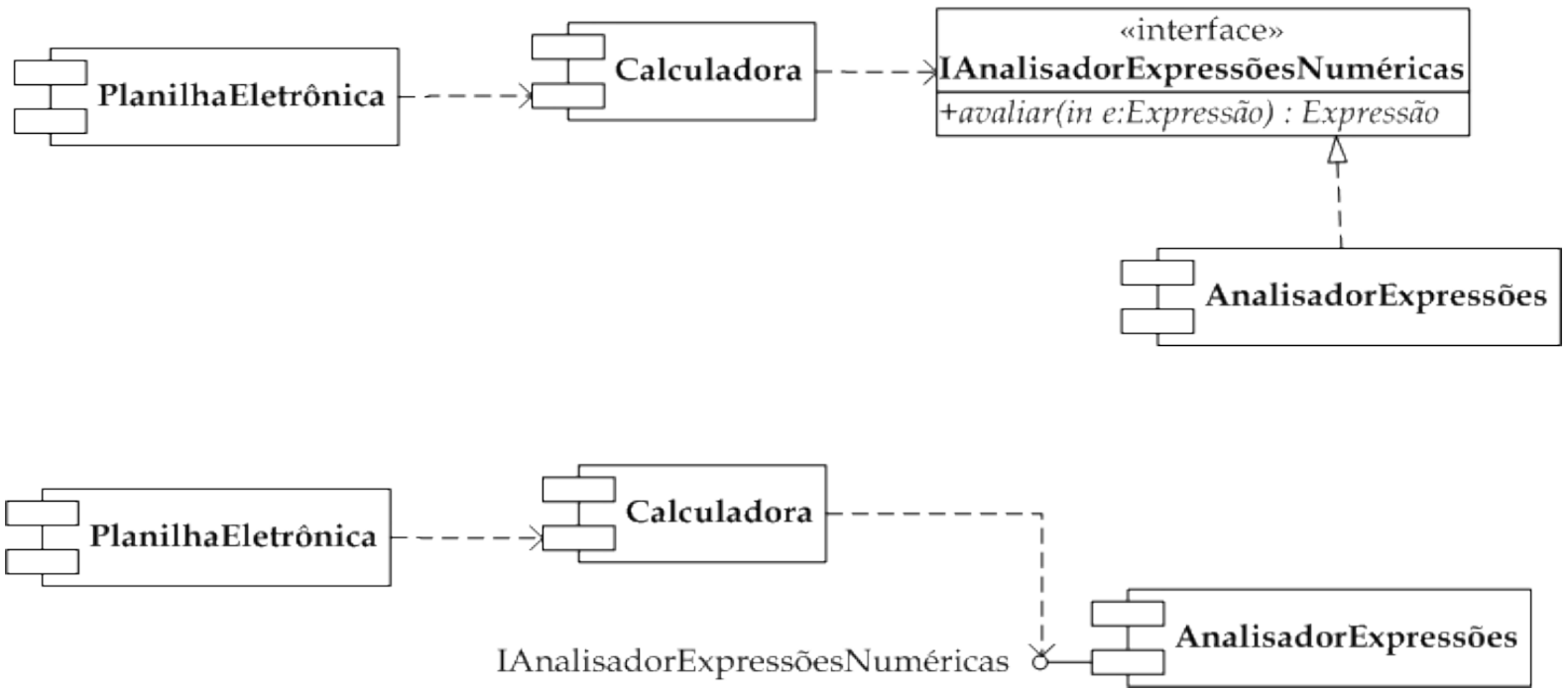
- A UML define uma forma gráfica para representar componentes visualmente, o diagrama de componentes



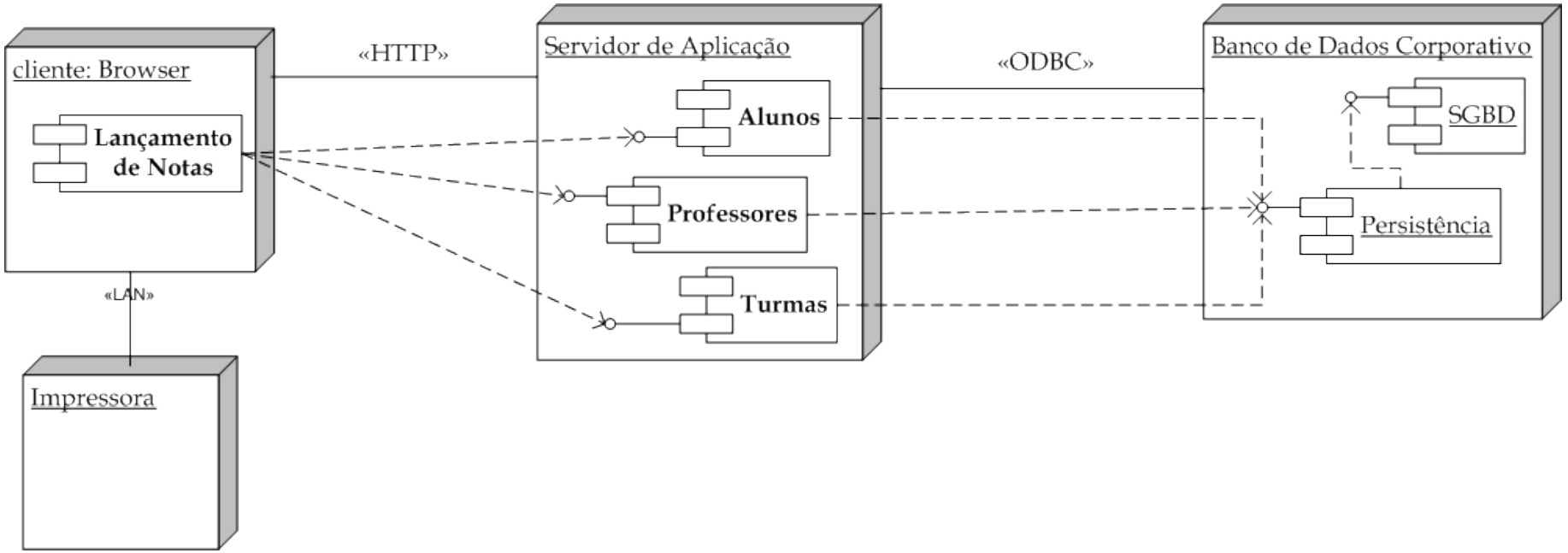
Alocação de Componentes



Alocação de Componentes



Alocação de Componentes



PROJETO DE ARQUITETURA NO PROCESSO I&I

Projeto de Arquitetura no Processo I&I

- A construção dos diagramas de componentes é iniciada na fase de elaboração (projeto da arquitetura) e refinada na fase de construção (projeto detalhado)

- A construção de diagramas de componentes se justifica para componentes de execução
  - Permite visualizar as dependências entre os componentes e a utilização de interfaces a tempo de execução do sistema
  - Sistema distribuído: representar seus componentes utilizando diagramas de implantação

Projeto de Arquitetura no Processo I&I

- Não é recomendável usar diagramas de componentes para representar dependências de compilação entre os elementos do código fonte do sistema
  - A maioria dos ambientes de desenvolvimento tem capacidade de manter as dependências entre códigos fonte, códigos objeto, códigos executáveis e páginas de ***script***
  - Só adiciona mais diagramas que terão que ser mantidos e que não terão uma real utilidade
  - Há também vários sistemas de gerenciamento de configurações e de versões de código (e.g, CVS)

Projeto de Arquitetura no Processo I&I

- Em relação ao diagrama de implantação, sua construção tem início na fase de elaboração

- Na fase de construção, os componentes são adicionados aos diversos nós
  - Cada versão do sistema corresponde a uma versão do DI, que exibe os componentes utilizados na construção daquela versão
  - O DI deve fazer parte dos manuais para instalação e operacionalização do sistema

- Nem todo sistema necessita de diagramas de componentes ou diagramas de implantação
  - Sistemas simples versus sistemas complexos

Referências

- BEZERRA, E. Princípios de Análise e Projeto de Sistemas com UML. 2ª ed. Rio de Janeiro: Elsevier, 2007.

- FOWLER, M. 3. UML Essencial. 3. ed. Porto Alegre: Bookman, 2007.