

Aula 21

Professores:

Cláudia Maria Lima Werner

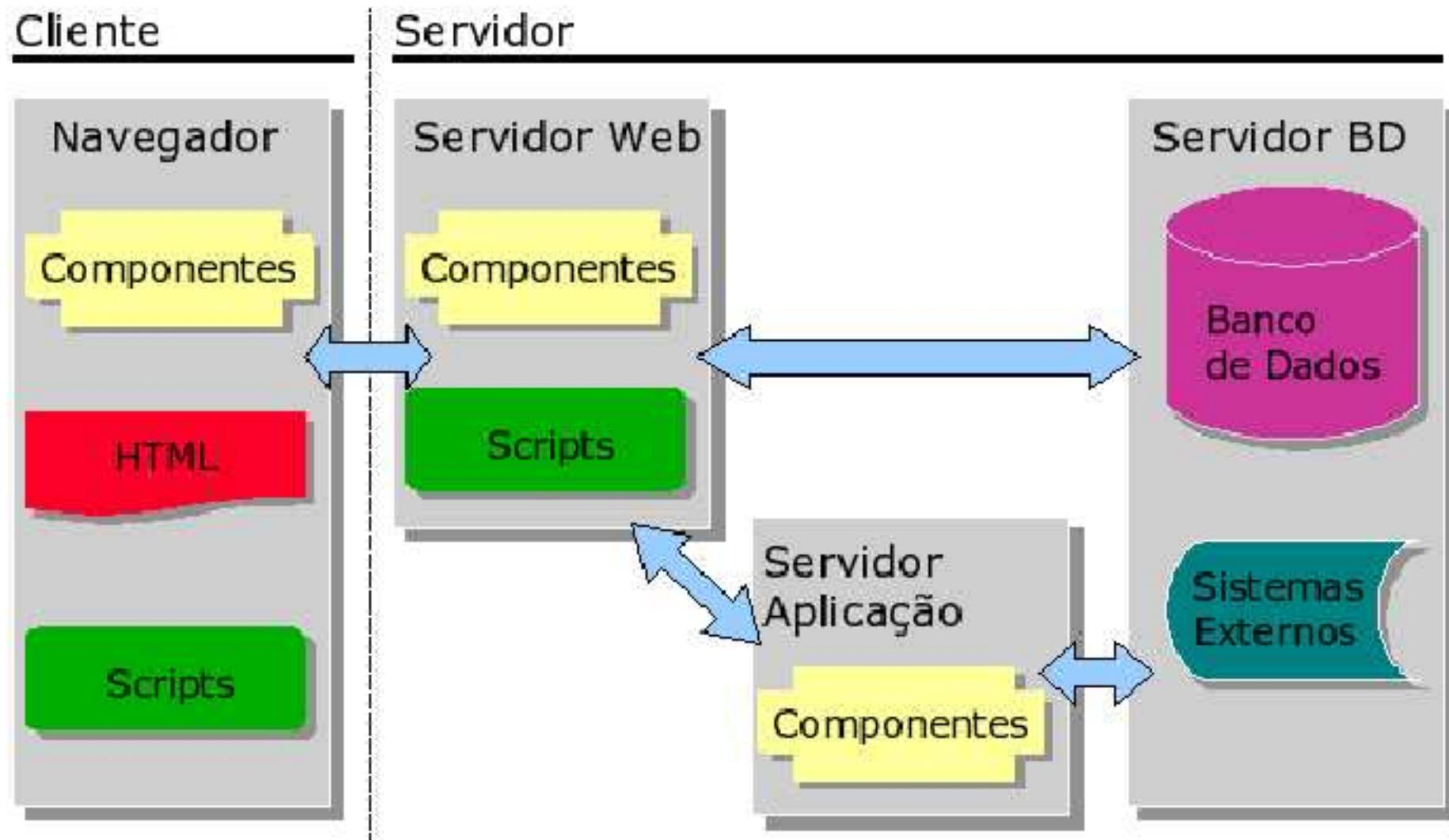
Leonardo Gresta Paulino Murtas

Estilos Arquiteturais para Web

Conteúdo:

- Arquitetura de Internet típica
- Estilo arquitetural Thin Client
- Estilo arquitetural Scripted Client
- Estilo arquitetural AJAX
- Estilo arquitetural Thick Client
- Estilo arquitetural Web Delivery
- Cenário exemplo

Arquitetura de Internet Típica



Estilos Arquiteturais



- Diversos estilos arquiteturais podem ser aplicados
 - *Thin Client*: mínima utilização dos recursos do cliente, exigindo mínima capacidade de seu navegador
 - *Scripted Client*: utilização de *scripts* de cliente apenas para a verificação de dados e dinâmica da interface com o usuário
 - AJAX: utilização de *scripts* de cliente de forma assíncrona, permitindo a carga parcial de páginas
 - *Thick Client*: distribuição da lógica de negócio entre o cliente e o servidor
 - *Web Delivery*: utilização de protocolos complementares ao HTTP, tais como RMI ou COM+

Thin Client

→ Características

- Estilo mais apropriado para aplicações Internet
- Baixo controle da configuração do navegador cliente
- Toda a lógica de negócio reside no servidor
- O servidor pode utilizar qualquer tecnologia ou recurso
- Principais aplicações: *sites* para grande público e *sites* de comércio eletrônico

Thin Client

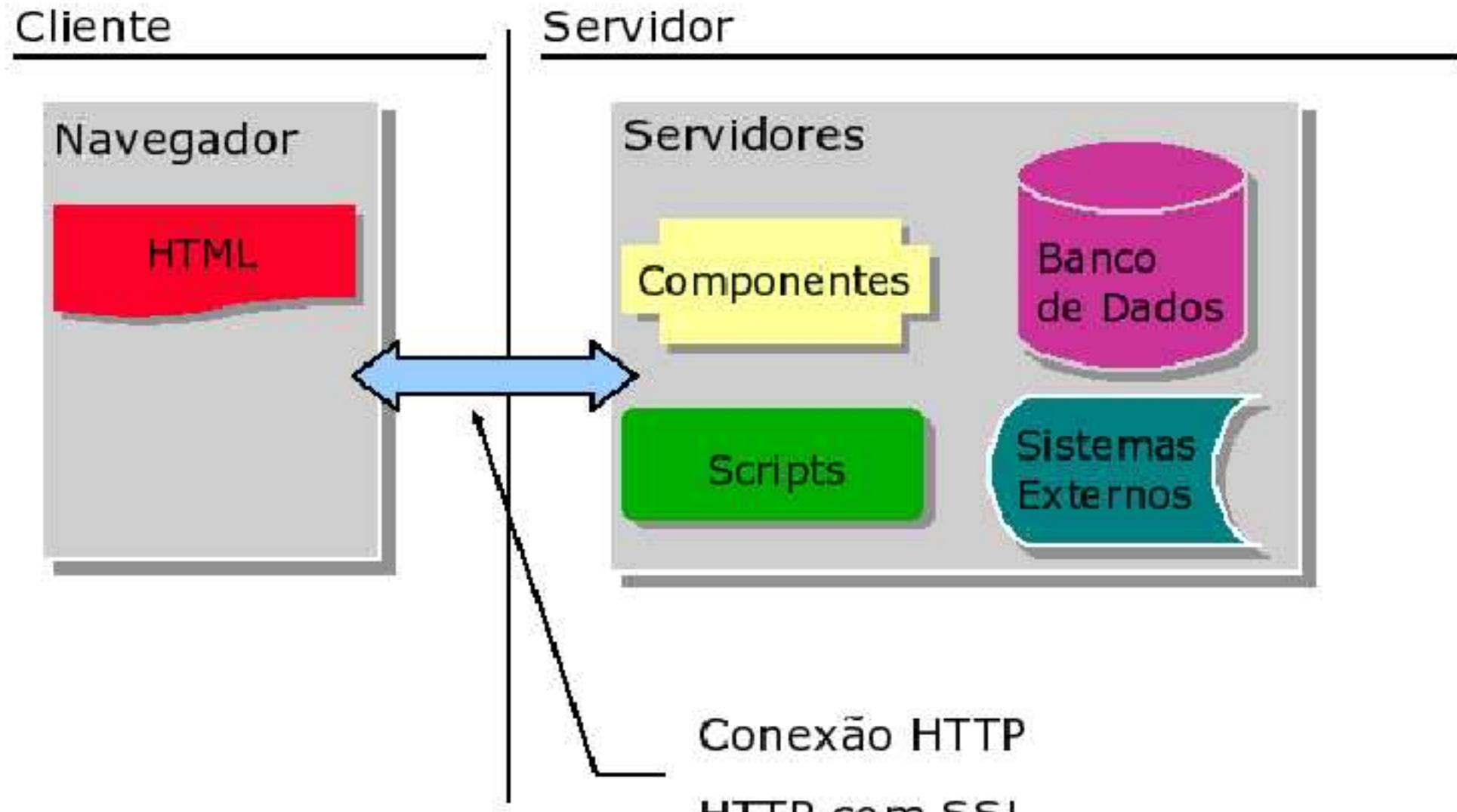
→ Vantagens

- Independência total de plataforma ou navegador
- Concentração da lógica de negócio no servidor

→ Limitações

- Interface com o usuário limitada pela linguagem HTML
- Maior número de transações com o servidor

Thin Client



Scripted Client

→ Características

- Extensão do estilo *thin client* que permite que um cliente possua scripts
- Os *scripts* devem ser utilizados apenas para a verificação dos dados de entrada e mínima dinâmica da interface
- O estilo aumenta minimamente a dependência entre a aplicação Internet e o navegador do cliente, mas oferece melhor performance

Scripted Client

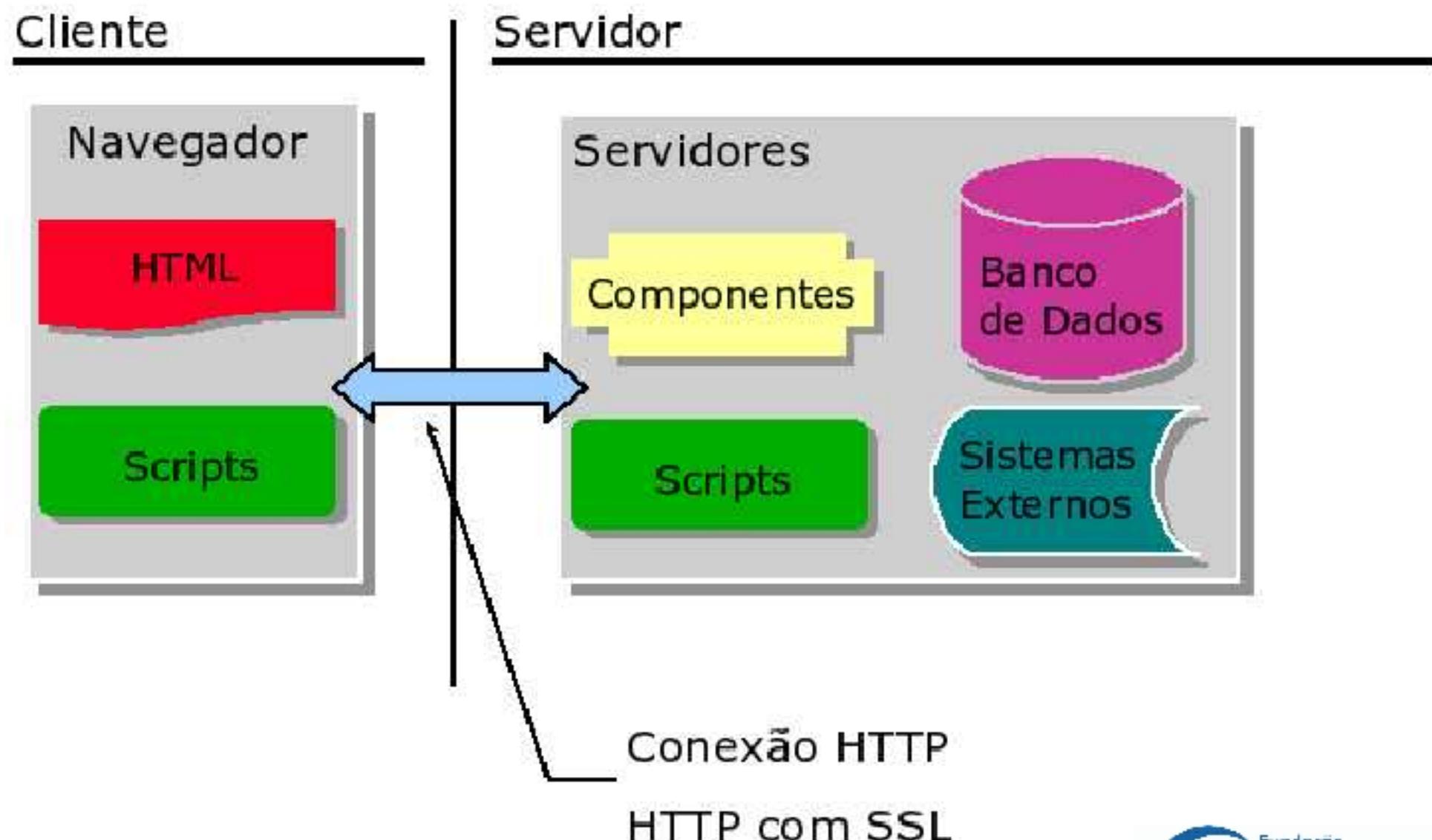
→ Vantagens

- Redução do número de transações com o servidor
- Lógica de negócio "de fato" concentrada no servidor

→ Limitações

- Dependência (limitada) com o navegador

Scripted Client



AJAX

→ Características

- Extensão do estilo *scripted client* que permite a carga parcial de dados em uma página
- AJAX = Asynchronous Javascript And XML
- Não é uma tecnologia, mas sim o agrupamento de tecnologias

AJAX

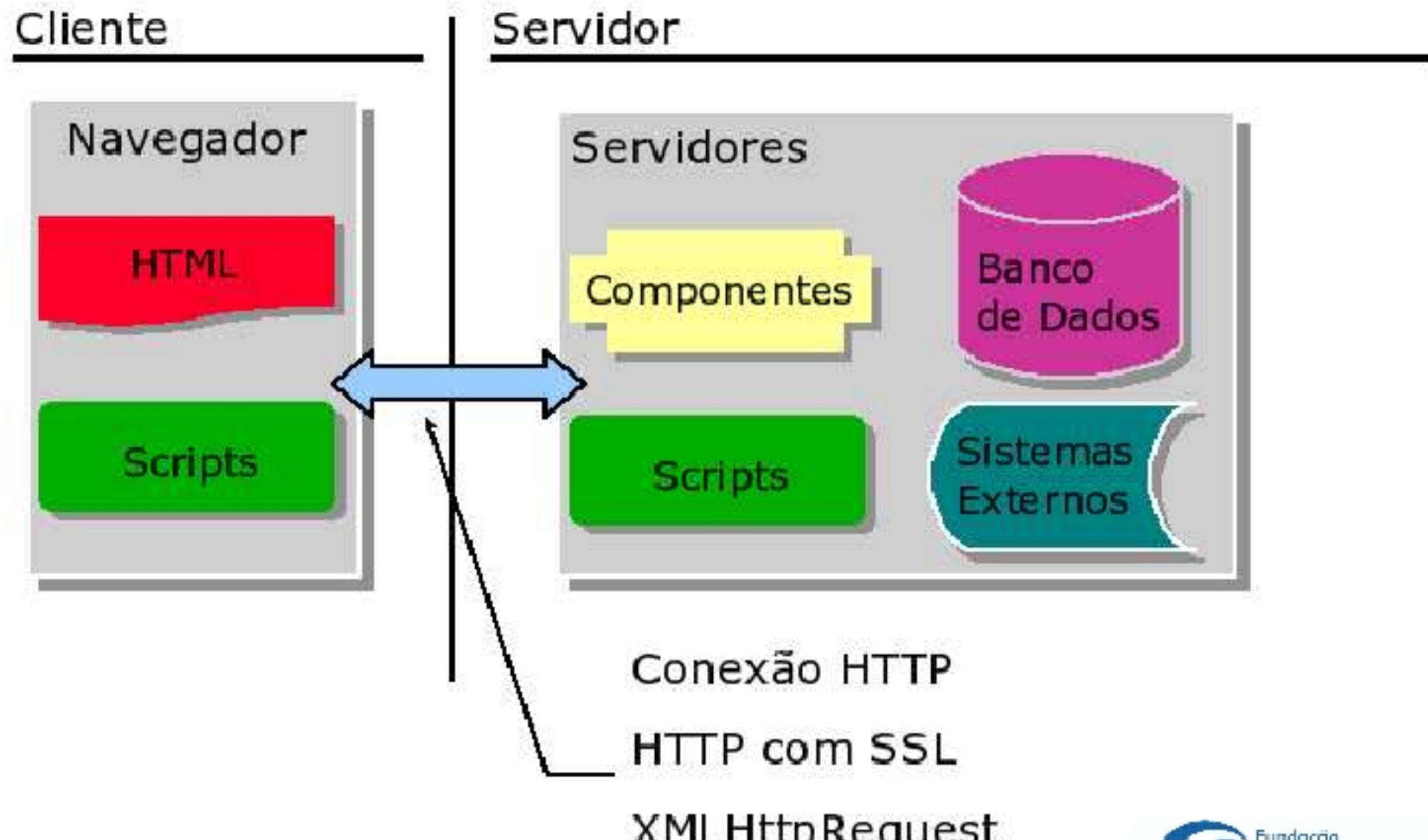
→ Vantagens

- Carga parcial e assíncrona de dados para o cliente

→ Limitações

- Dependência (limitada) com o navegador

AJAX



Thick Client

→ Características

- Extensão do estilo *scripted client* que permite que um cliente possua scripts e componentes
- O cliente passa a ser responsável por parte da lógica de negócio
- Aumenta a dependência entre a aplicação Internet e o navegador do cliente

Thick Client

→ Vantagens

- Interface com o usuário mais sofisticada
- Possibilidade de resposta mais rápida (menos transações)

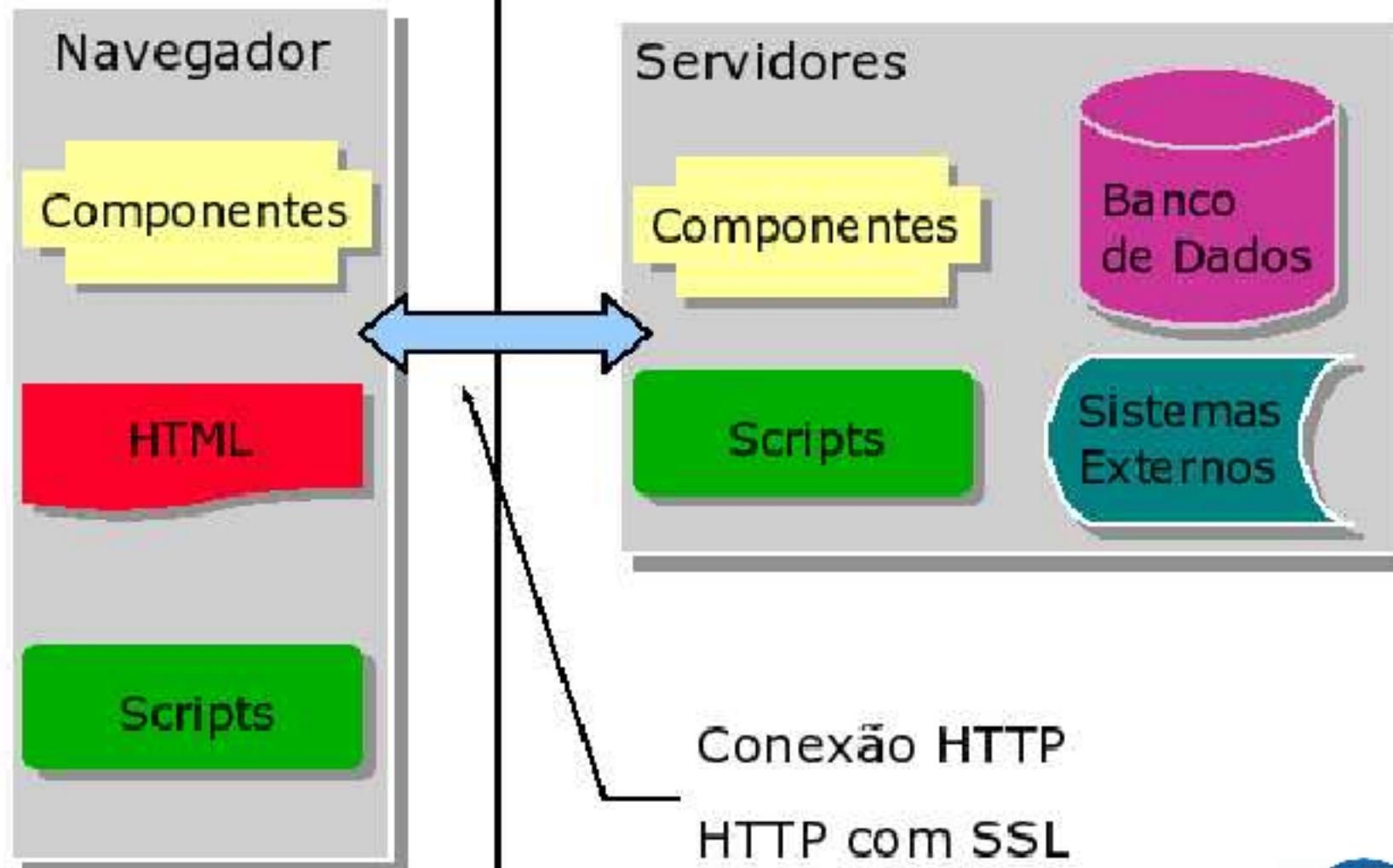
→ Limitações

- Dependência de plataforma (pode ser limitada)
- Distribuição da lógica de negócio (manutenção)

Thick Client

Cliente

Servidor



Web Delivery

→ Características

- A Internet é utilizada como meio de apresentação e de transação de componentes distribuídos
- Existe comunicação direta entre os componentes no cliente e no servidor
- A lógica de negócio não é restrita pelas limitações impostas pelas transações através da Internet
- O estilo é aplicável onde existe maior controle sobre as configurações da rede e do navegador

Web Delivery

→ Vantagens

- Maior dinâmica na aplicação (interface e lógica)
- Transações em protocolos mais eficientes

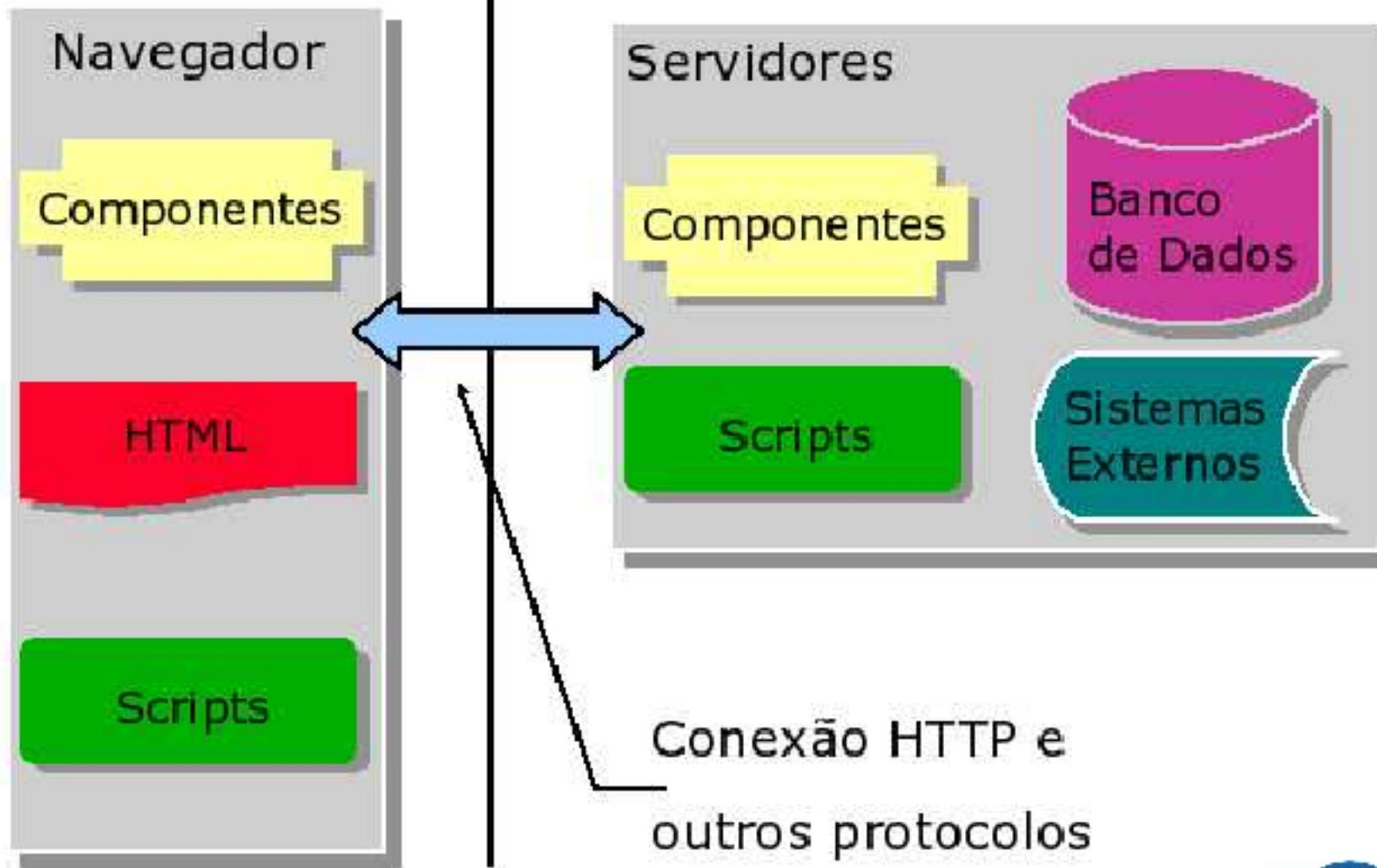
→ Limitações

- Dependência de plataforma
- Distribuição da lógica de negócio (manutenção)

Web Delivery

Cliente

Servidor



Cenário Exemplo

→ Consideremos um trecho de um caso de uso

- "Ao selecionar o estado a partir de uma combo-box, o usuário deverá selecionar a cidade em que se encontra a partir de uma segunda combo-box. Neste segundo controle, devem aparecer somente as cidades dentro do estado selecionado"

→ Que estilo arquitetural nos permite implementar esta funcionalidade ???

Estilo *Thin Client*



Implementação com o estilo *Thin Client*

- A inexistência de lógica no cliente exige uma transação HTTP entre o cliente e o servidor para o preenchimento da combo-box com os nomes das cidades
- A transação consome tempo e deve ser disparada através de um botão de algum formulário da página
- Formulário distribuído entre pelo menos duas páginas

Estilo *Scripted Client*



Implementação com o estilo *Scripted Client*

- A lógica no cliente se resume aos *scripts*, que neste caso devem manter as relações entre os estados e as cidades
- O grande volume de informações reduz a performance no *upload* das páginas
- A sobrecarga no servidor, devido ao envio das páginas e de sua construção, reduz o número de acessos simultâneos

Estilo AJAX

→ Implementação com o estilo AJAX

- Somente a lista de estados é carregada junto com a página HTML
- Em função do estado selecionado, a lista de cidades daquele estado é carregada
- Para cada novo estado selecionado, o cliente necessita acessar o servidor para carregar a lista de cidades daquele estado

Estilo *Thick Client*



Implementação com o estilo *Thick Client*

- A lógica no cliente pode residir em componentes
- Componentes conteriam as relações entre os estados e suas cidades
- Componentes podem ser previamente distribuídos, reduzindo o tempo de *upload* das páginas
- Problemas: manutenção dos componentes no cliente e dependência de plataforma !!!

Estilo *Web Delivery*



Implementação com o estilo *Web Delivery*

- Componentes no cliente podem requisitar informações do servidor para o preenchimento da combo-box com as cidades de um determinado estado
- A exigência de um componente no lado cliente resulta nos mesmos problemas encontrados no estilo *Thick Client*
- Outro problema: tecnologias menos comuns no mercado (mais caras!)

Bibliografia

→ Extra

- Jim Conallen, 2000, "Building web applications with UML", Addison-Wesley, Capítulo 7.