
PROMPT 2011

Módulo 3 – Aplicações Web

A World Wide Web e Introdução
aos Sistemas Distribuídos



Centro de Cálculo
Instituto Superior de Engenharia
de Lisboa

Luis Falcão (lfalcao@cc.isel.ipl.pt)
João Trindade (jtrindade@cc.isel.ipl.pt)

Agenda

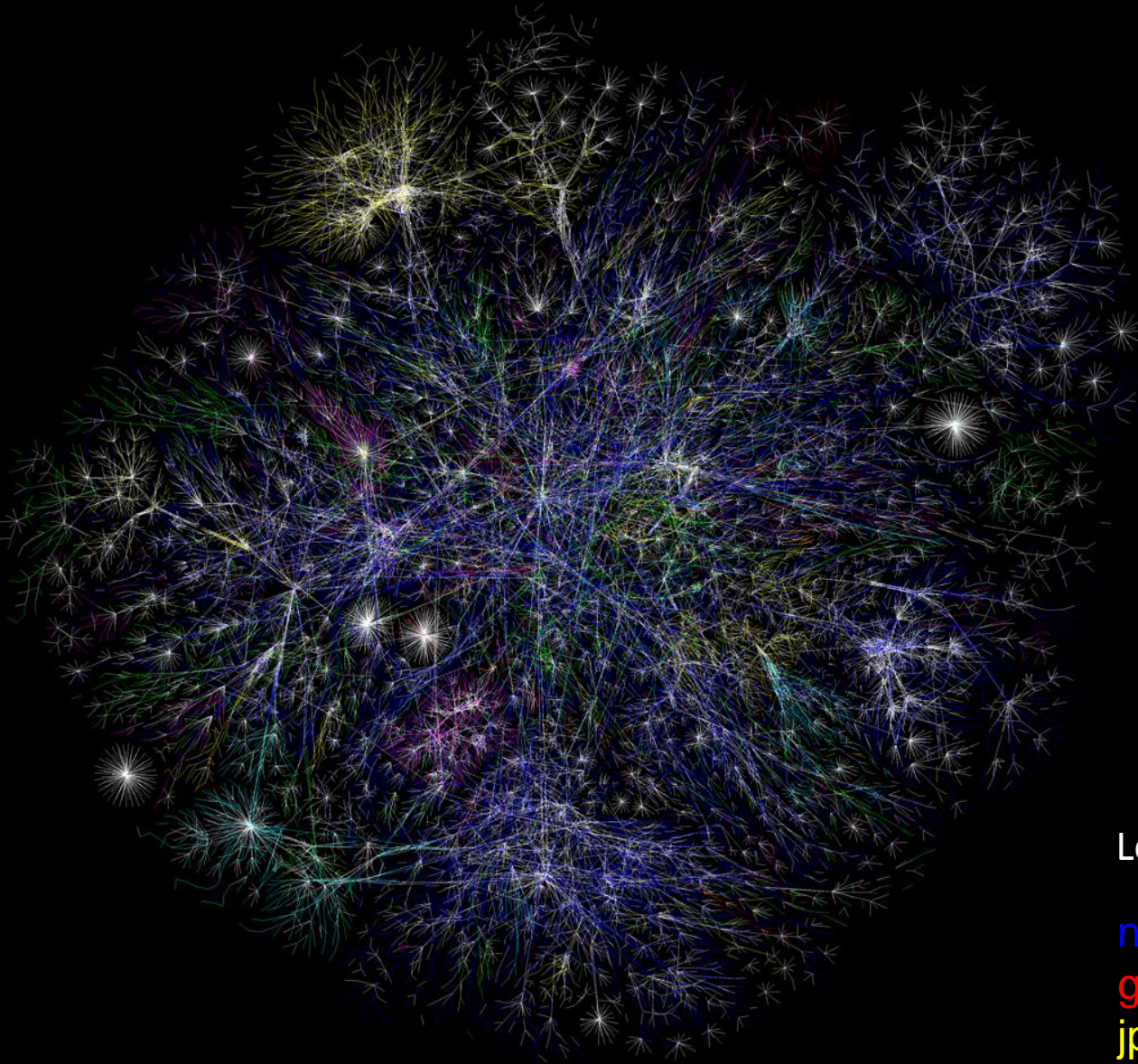
- A Internet e a *World Wide Web*
- Identificação de recursos
 - *Uniform Resource Locator* (URL)
- Sistemas distribuídos (introdução)
 - Distribuição: causas e consequências
 - Propriedades pretendidas
 - Organização de software em camadas
 - Arquitecturas de sistemas distribuídos
 - Modelos de interacção
 - Desenho de sistemas distribuídos
- O objecto de estudo
 - Aplicações Web

Internet

- O que é a Internet?
 - A rede IP
 - “Auto-estrada da informação”
- IETF - Internet Engineering Task Force
 - Organização responsável pela normalização de serviços e protocolos da Internet
 - <http://www.ietf.org/>
- Alguns dos protocolos mais utilizados
 - **HTTP** - Hyper Text Transfer Protocol
 - **FTP** – File Transfer Protocol
 - **SMTP** - Simple Mail Transfer Protocol
- W3C – WorldWideWeb Consortium
 - <http://www.w3.org/>

Rotas na internet (cerca de 30%)

Data: Jan 15 2005

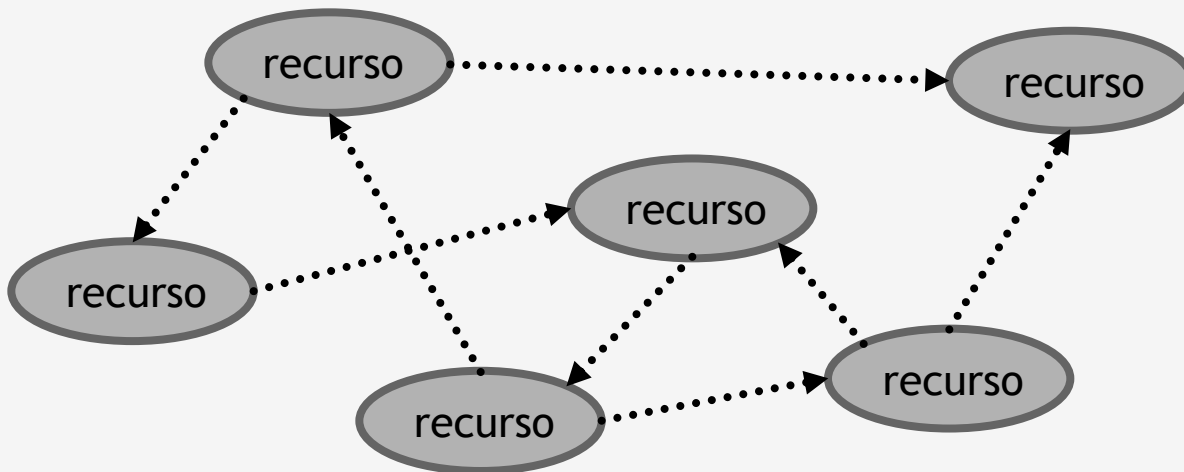


Legenda:

net, ca, us com, org mil,
gov, edu
jp, cn, tw, au de, uk, it, pl, fr
br, kr, nl unknown

World Wide Web – WWW

- O que é a *World Wide Web*?
 - Serviço mais utilizado na Internet
 - O utilizador, recorrendo a um browser, “vê” a Internet como um conjunto de recursos interligados (através de URI)

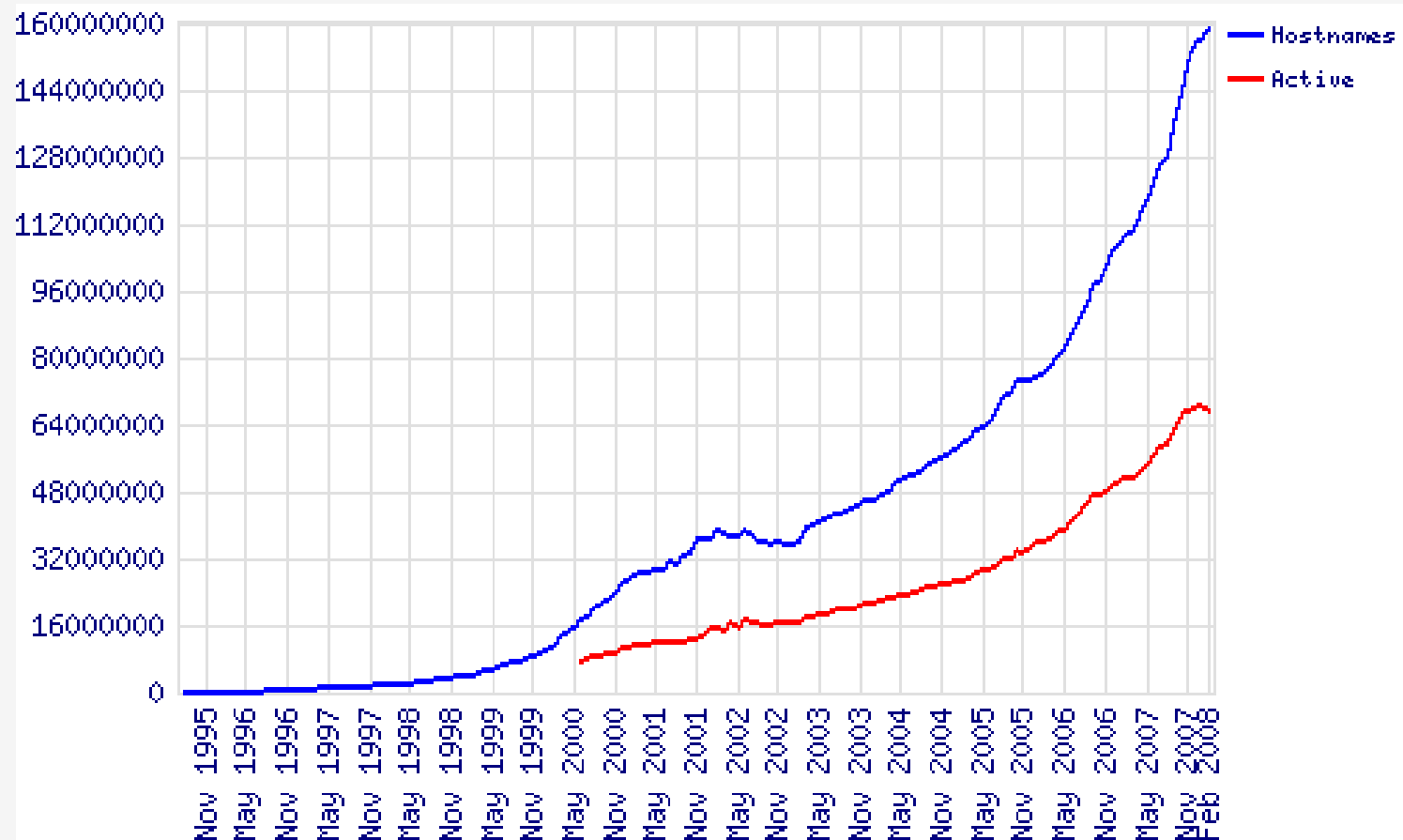


WWW – Historial (1)

- **Dez. 1991:** demonstração da WWW na Hypertext'91
- **1993:** No *National Center For Supercomputing Applications* (NCSA) é criado o primeiro *browser* (**Mosaic**)
- **1994:** Criado o *World Wide Web Consortium* (W3C)
 - Organização responsável pela normalização das tecnologias utilizadas na Web - <http://www.w3.org/>
- **O sucesso da Web promoveu-a a ambiente de execução de aplicações distribuídas, as aplicações Web**

WWW – Historial (2)

- Crescimento de domínios na Web
 - 158,209,426 domínios em Fevereiro de 2008 (+2.6 milhões que no mês anterior)
 - Apenas ~70,000,000 dos quais activos (~44%)

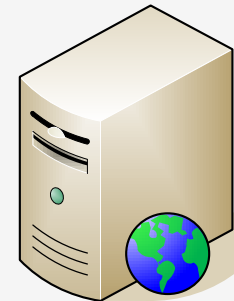


WWW – Intervenientes

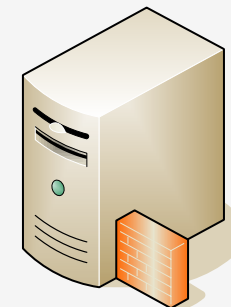
- **Browser** – Cliente HTTP



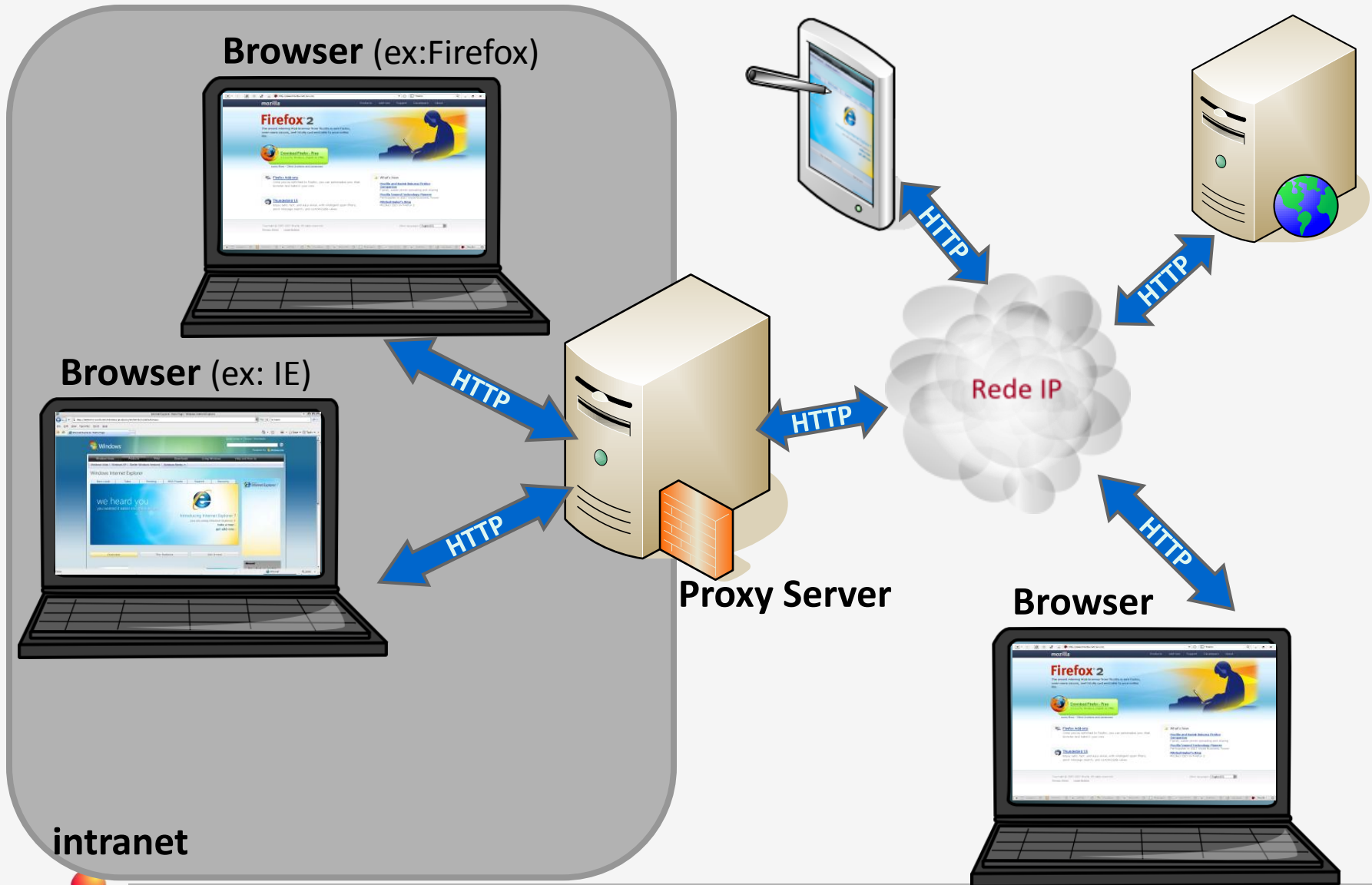
- **Servidor Web** - Servidor HTTP
(normalmente é usado o porto TCP 80)



- **Proxy** – Intermediário que realiza *caching*
– Pode também realizar controlo de acessos



WWW – Intervenientes (cont.)



Agenda (cont.)

- A Internet e a World Wide Web
- Identificação de recursos
 - *Uniform Resource Locator* (URL)
- Sistemas distribuídos (introdução)
 - Distribuição: causas e consequências
 - Propriedades pretendidas
 - Organização de software em camadas
 - Arquitecturas de sistemas distribuídos
 - Modelos de interacção
 - Desenho de sistemas distribuídos
- O objecto de estudo
 - Aplicações Web

Identificação de recursos

- **URI** - Uniform Resource Identifiers (RFC 3986)
 - Definem a sintaxe para identificadores de recursos

- Formato genérico de um URI

<esquema> : **<parte-específica-esquema>**

- Tipos de URIs

- **URN** - Uniform Resource Names (RFC 2141)
 - Identificadores persistentes, independentes da localização
 - **Exemplo:** *urn:ietf:rfc:2141*
- **URL** – Uniform Resource Locators (RFCs 2368, 1738, 1808)
 - Identificam por localização
 - **Exemplo:** *http://www.w3.org/Addressing/*

URL (*Uniform Resource Locators*)

- Caracteres utilizados nos URL:
 - Caracteres US-ASCII
 - Caracteres que podem ser reservados:
 - {“;”, “/”, “?”, “:”, “@”, “=”, “&”, “#”}
 - Se aparecerem fora do contexto para o qual foram reservados, devem ser codificados
 - Codificação de caracteres: %<código_hexadecimal>
 - Se o esquema tiver *links* relativos e uma estrutura hierárquica, os componentes da hierarquia devem ser separados por “/”
- Ex:
 - <http://labnet.cc.isel.ipl.pt/login.aspx?ReturnUrl=%2fblogs%2fdefault.aspx>

 %2f representa o carácter '/'

Esquemas

- Parte específica de esquemas que envolvem protocolos da família TCP/IP
<scheme>://[<user>[:<password>]@]<host>[:<port>]/<url-part>
- Alguns exemplos:
http://www.slbenfica.pt/register.asp?nome=Paulo%20Pereira&id=1
http://www.nasa.gov@137.12.132.10/index.aspx
ftp://anonymous:password@strauss.cc.isel.ipl.pt/pub?type=a
mailto:user@domain.org?subject=SLB&body=O%20Glorioso
- Para mais informações veja <http://www.w3.org/Addressing/>

Agenda (cont.)

- A Internet e a *World Wide Web*
- Identificação de recursos
 - *Uniform Resource Locator* (URL)
- Sistemas distribuídos (introdução)
 - Distribuição: causas e consequências
 - Propriedades pretendidas
 - Organização de software em camadas
 - Arquitecturas de sistemas distribuídos
 - Modelos de interacção
 - Desenho de sistemas distribuídos
- O objecto de estudo
 - Aplicações Web

- O que é um Sistema distribuído?

“A distributed system is one in which components located at networked computers communicate and coordinate their actions only by passing messages”

George Colouris et al,
in *“Distributed Systems: Concepts and Design”*

Sistemas Distribuídos: Motivação (1)

“It is easy to understand why networked systems are popular. Such systems allow the sharing of information and resources over a wide geographic and organizational spread.”

Michael D. Schroeder,
in *“Distributed Systems, chapter 1”*

“They allow the use of small, cost-effective computers.”

idem

“They can grow in small increments over a large range of sizes.”

ibidem

Sistemas Distribuídos: Motivação (2)

- Escalabilidade: Uma definição
 - Capacidade que o sistema tem para suportar o aumento do número de participantes (normalmente utilizadores) mantendo os níveis de qualidade de serviço (QoS) desejados
- Como suportar o aumento de escala?
 - *Scale-up* (ou *vertical scaling*)
 - Aumento da capacidade da máquina usada
 - Custos de *hardware* elevados
 - *Scale-out* (ou *horizontal scaling*)
 - Aumento do número de máquinas usadas
 - Custos de *hardware* menores
 - Ónus da distribuição colocado no *software*

Sistemas Distribuídos: Desafios

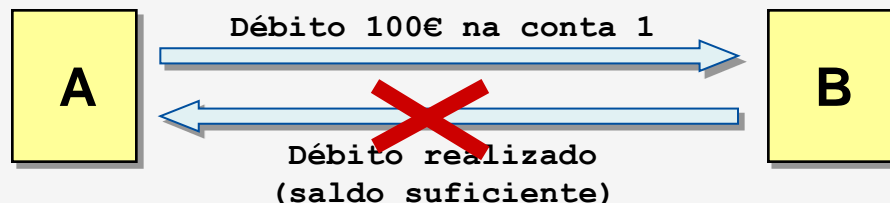
- Consequências da distribuição:
 - Troca fiável de mensagens
 - Segurança
 - Coordenação
- Propriedades pretendidas:
 - Escalabilidade
 - Disponibilidade (Desempenho e Fiabilidade)

“An interesting observation about network-based applications is that the best application performance is obtained by not using the network.”

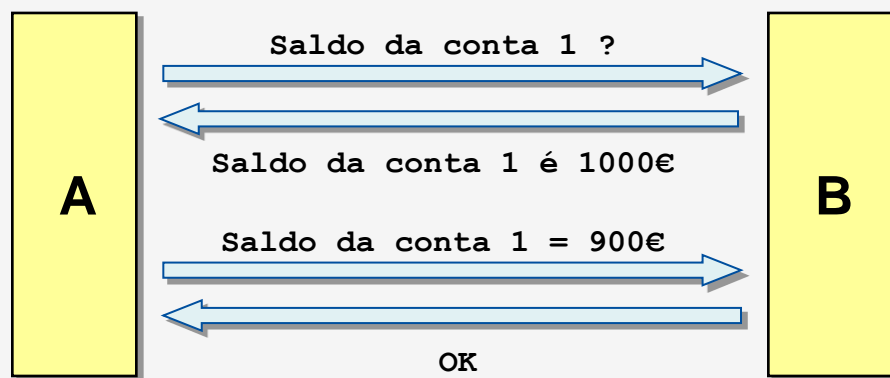
Roy Thomas Fielding,
in “Architectural Styles and the Design of Network-based Software Architectures”,
PhD dissertation

Fiabilidade na troca de mensagens

- O *stack* TCP/IP resolve o problema?



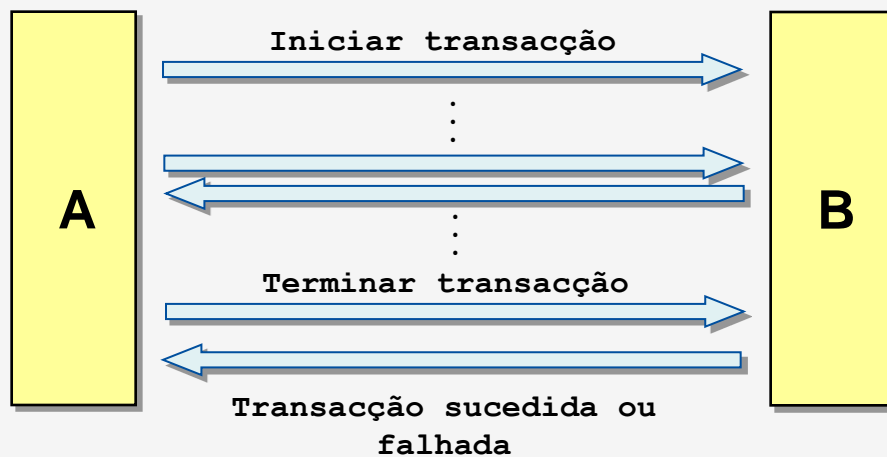
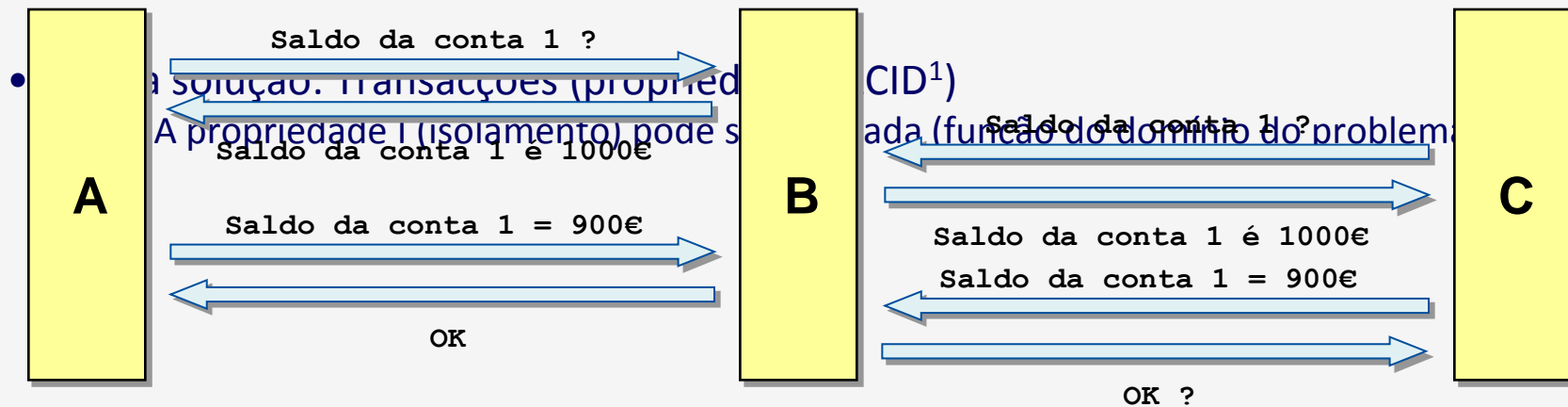
- Uma solução: mensagens idem potentes



- Outra solução: mensagens com marcas temporais

- Sincronização
 - No acesso concorrente a recursos partilhados
 - Uma escrita exclui todos os restantes acessos
 - Soluções:
 - Conservadoras (ou pessimistas): “*Check and act*”
 - Optimistas: “*Try and see*”
 - Exemplo: Transacções
- Acordo (por maioria ou consenso)
 - Acerca do valor de um recurso partilhado
 - Acerca do sucesso ou insucesso de uma ou mais operações
 - Exemplo: Transacções distribuídas

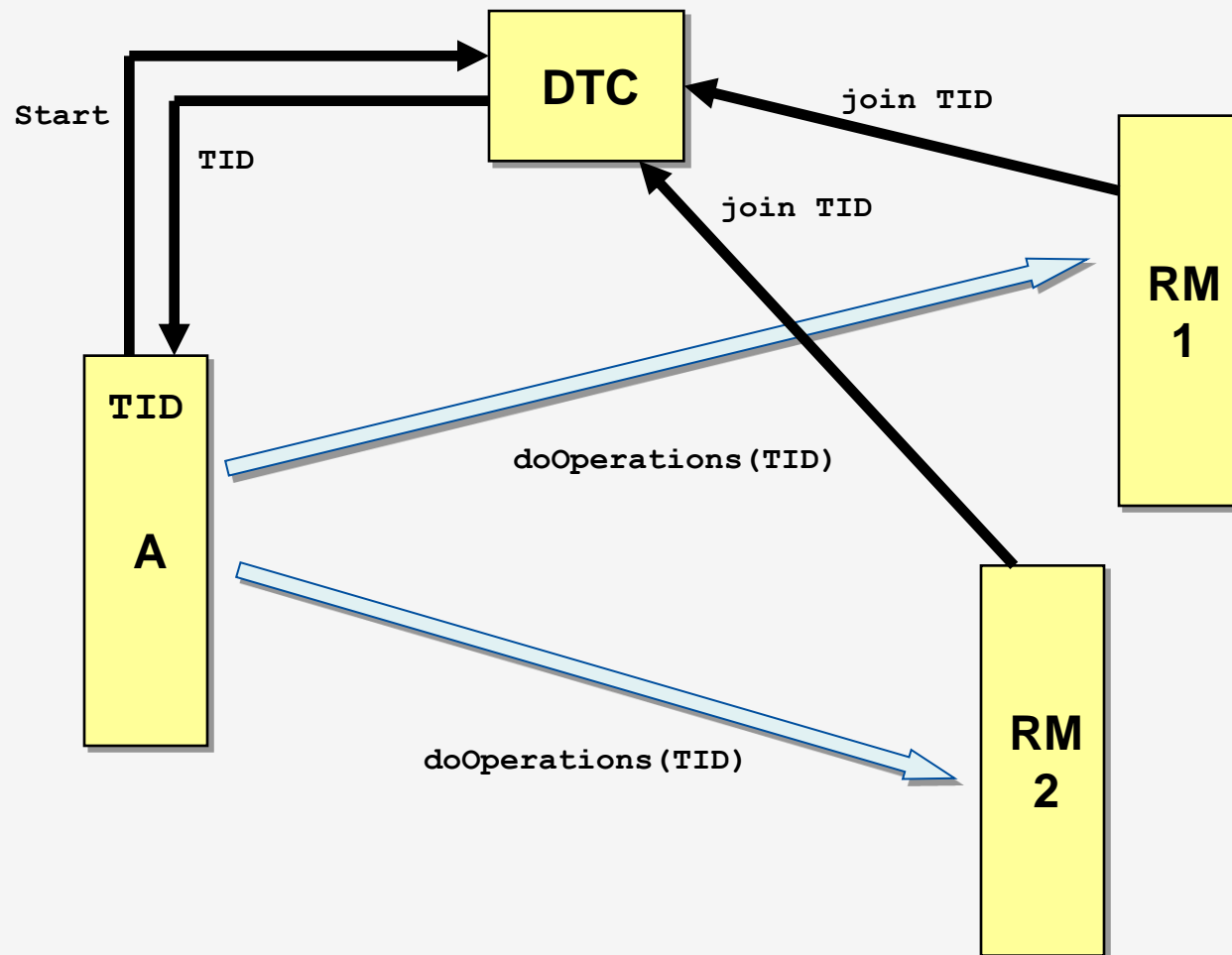
Sincronização



¹ Atomicity, Consistency, Isolation, Durability

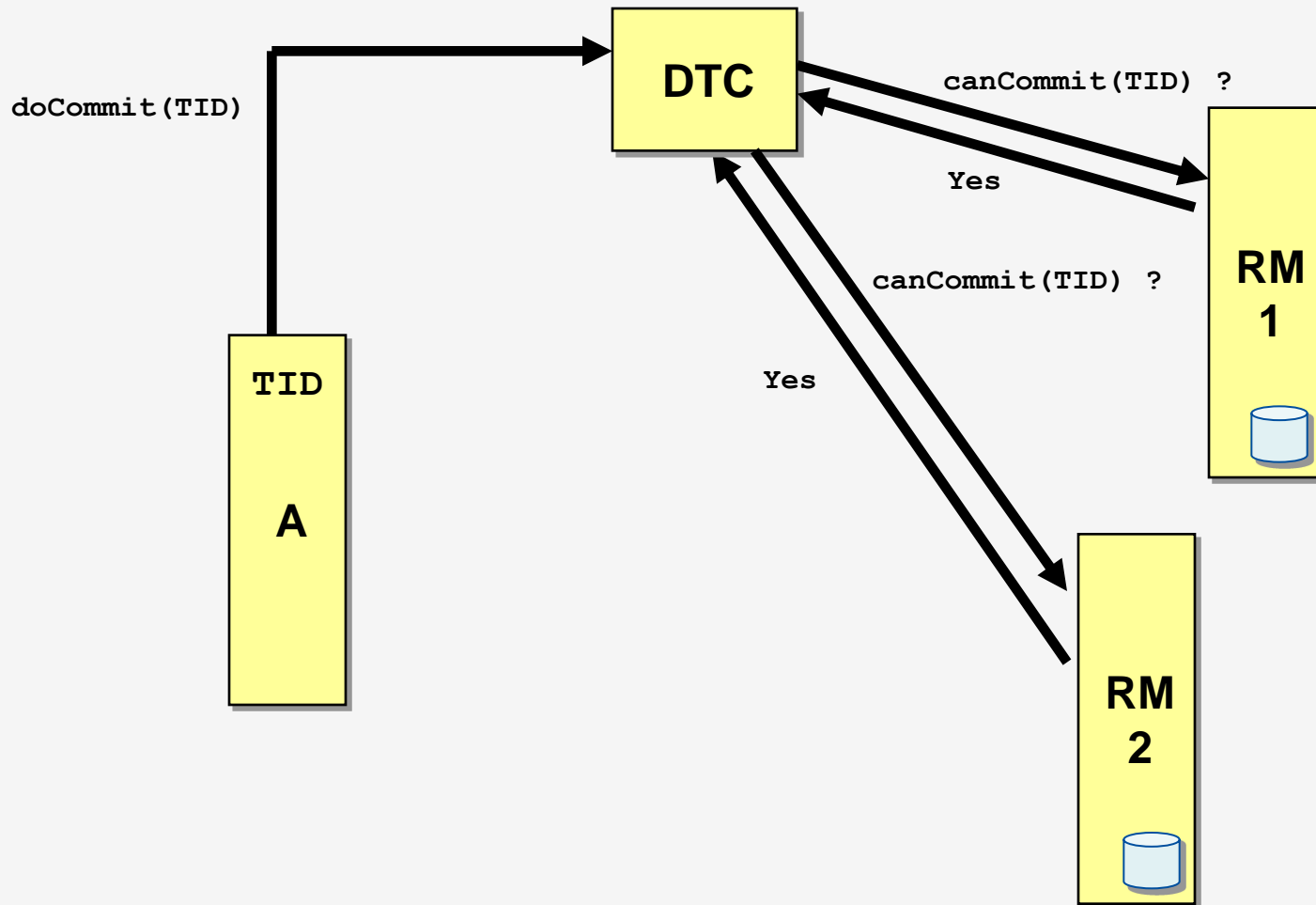
Acordo: Transacções distribuídas (1)

- Protocolo *Two phase commit (enrolling)*



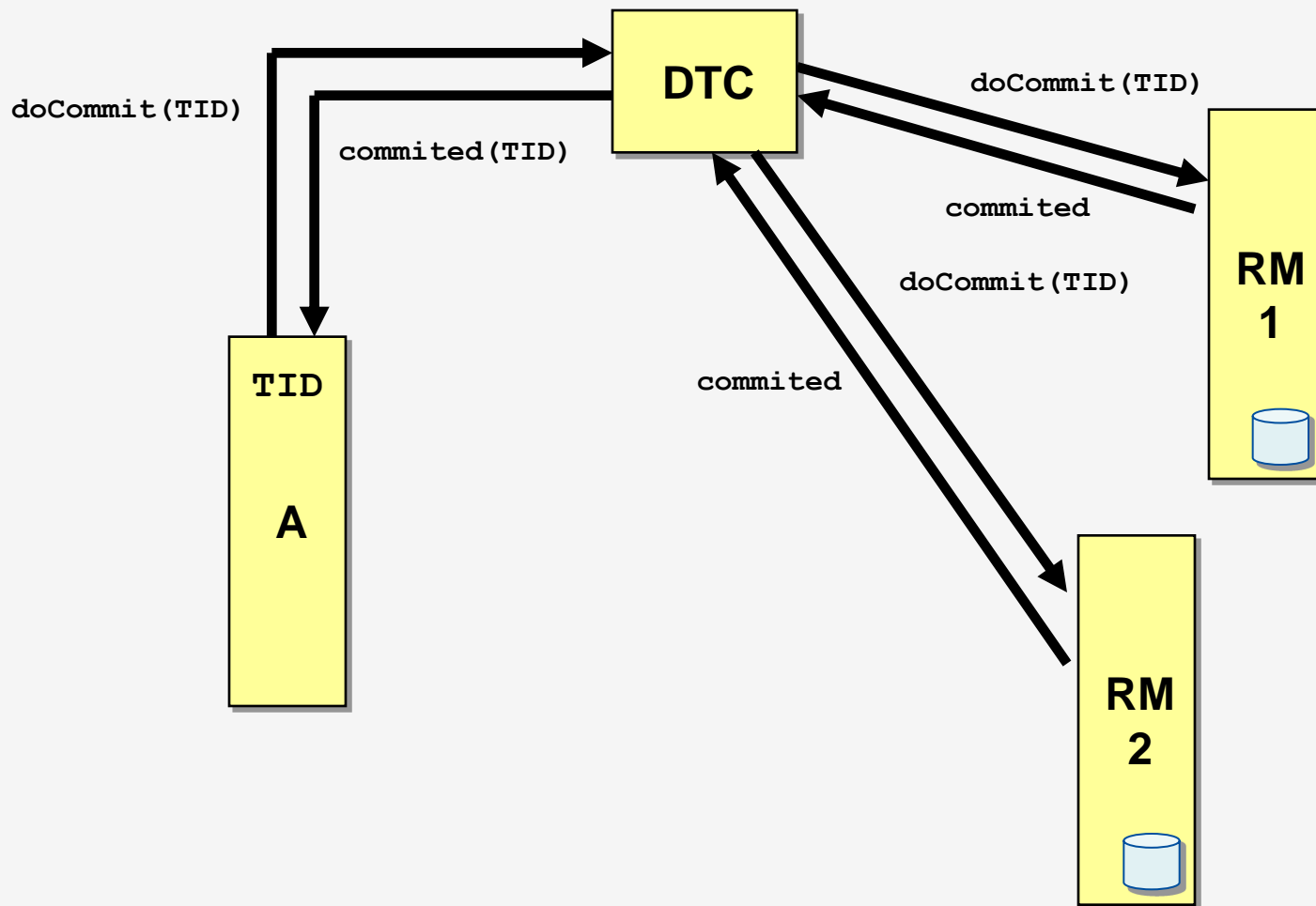
Acordo: Transacções distribuídas (2)

- Protocolo *Two phase commit (voting)*



Acordo: Transacções distribuídas (3)

- Protocolo *Two phase commit (committing)*



Propriedades pretendidas

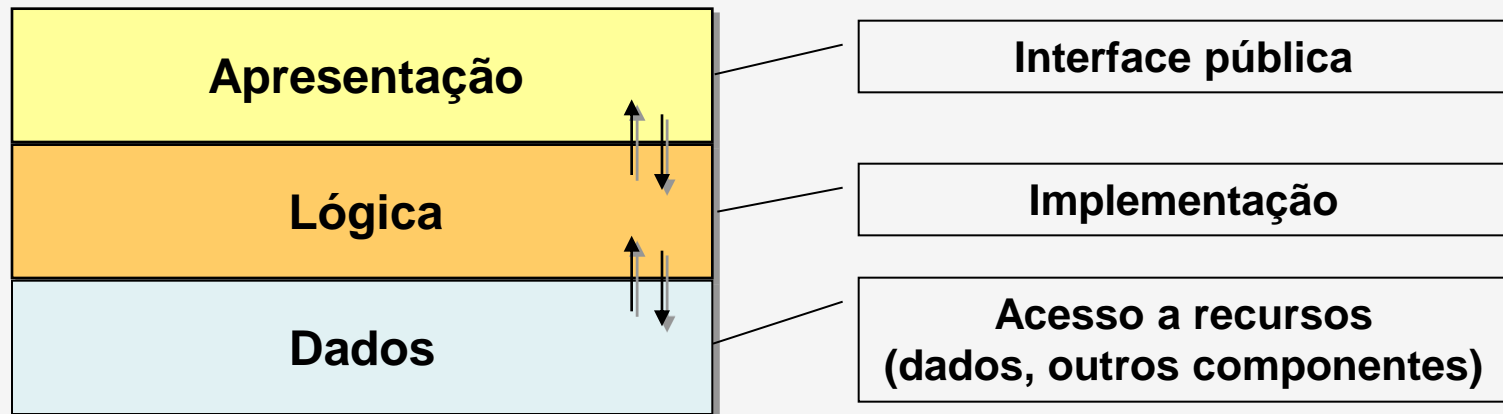
- Escalabilidade
- Disponibilidade
 - Desempenho
 - Latência
 - *Throughput*
 - Eficiência
 - Fiabilidade
 - Tolerância a falhas
- Facilidade de manutenção
 - Evolutiva
 - Correctiva

Agenda (cont.)

- A Internet e a *World Wide Web*
- Identificação de recursos
 - *Uniform Resource Locator* (URL)
- Sistemas distribuídos (introdução)
 - Distribuição: causas e consequências
 - Propriedades pretendidas
 - Organização de software em camadas
 - Arquitecturas de sistemas distribuídos
 - Modelos de interacção
 - Desenho de sistemas distribuídos
- O objecto de estudo
 - Aplicações Web

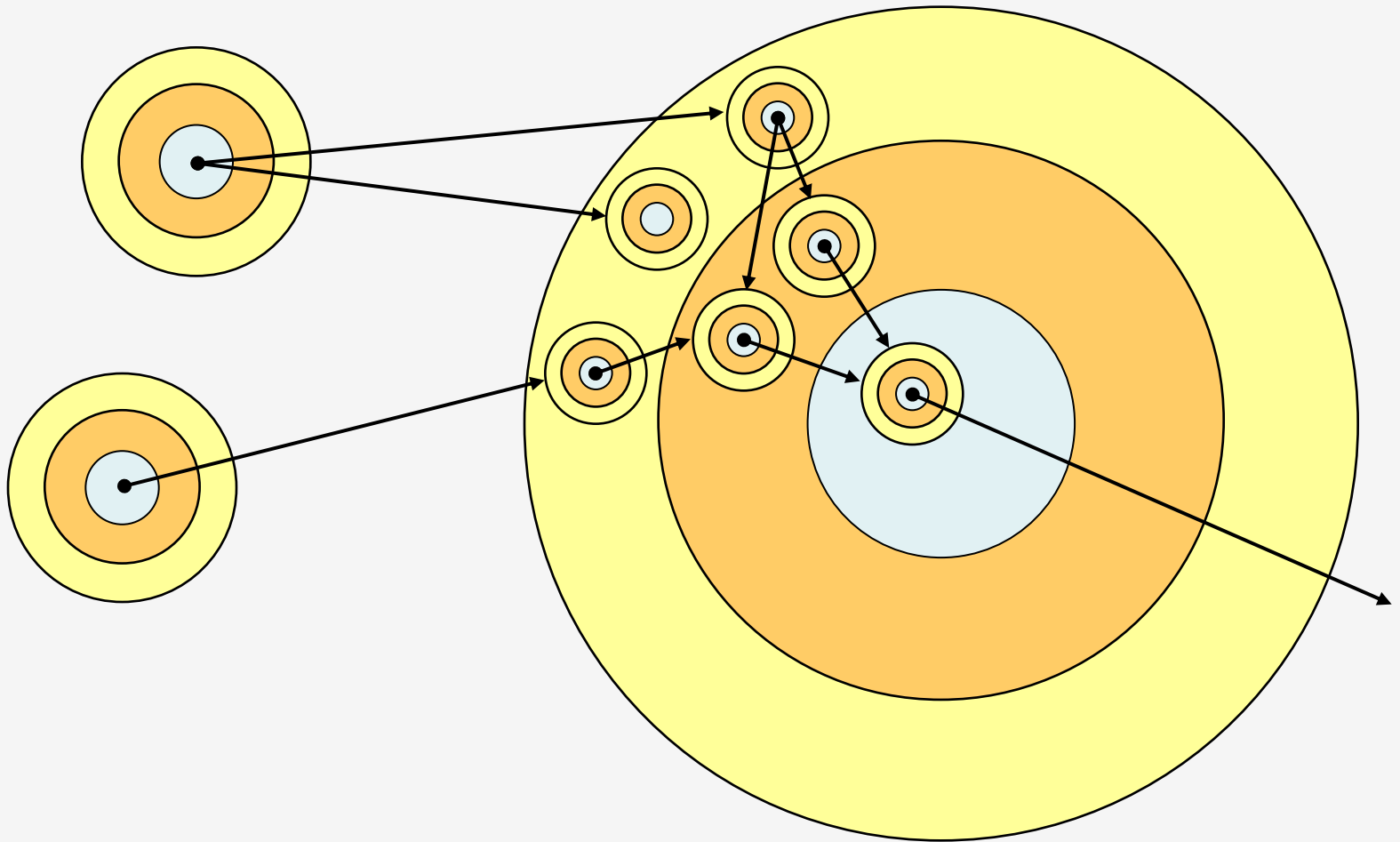
Organização de software em camadas (1)

- Designado de modelo *layered*
 - Favorece o isolamento



- Lógica (ou *Business Layer*) contém código que permanece válido
 - para qualquer apresentação
 - para qualquer fonte de dados
- Padrão de aplicabilidade recursiva

Organização de software em camadas (2)



- A separação lógica potencia a separação física!
 - E, conseqüentemente, a distribuição de carga

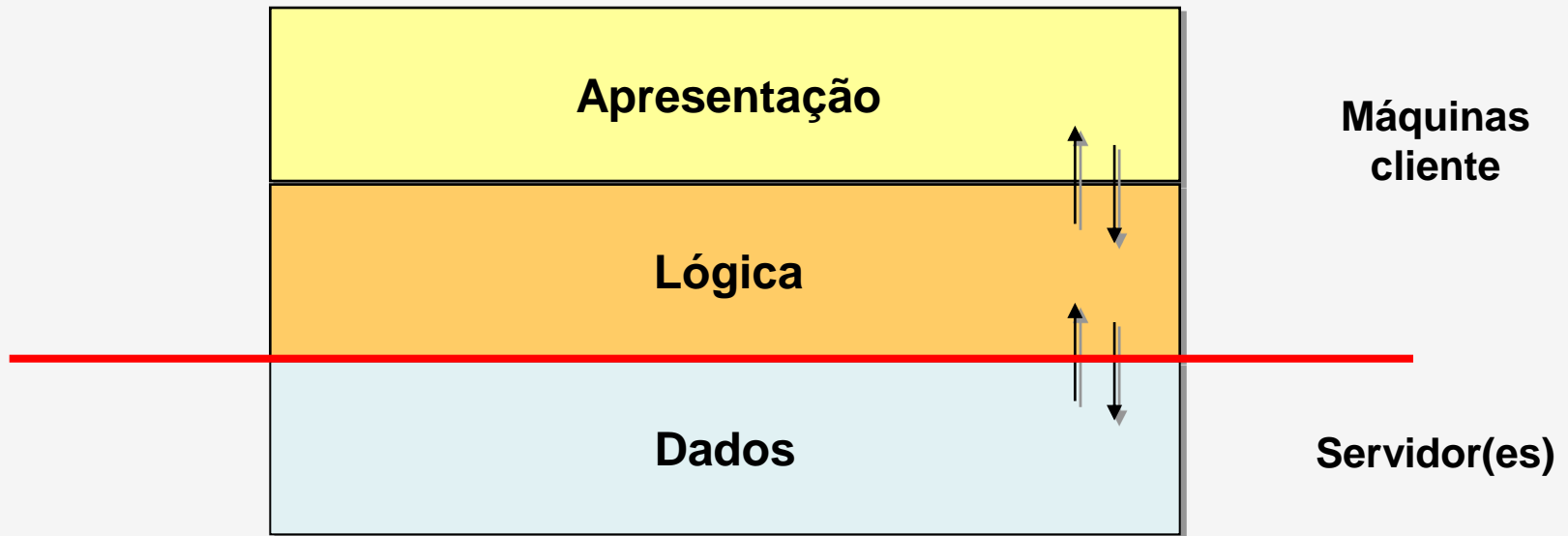
Agenda (cont.)

- A Internet e a *World Wide Web*
- Identificação de recursos
 - *Uniform Resource Locator* (URL)
- Sistemas distribuídos (introdução)
 - Distribuição: causas e consequências
 - Propriedades pretendidas
 - Organização de software em camadas
 - Arquitecturas de sistemas distribuídos
 - Modelos de interacção
 - Desenho de sistemas distribuídos
- O objecto de estudo
 - Aplicações Web

Arquitecturas de sistemas distribuídos

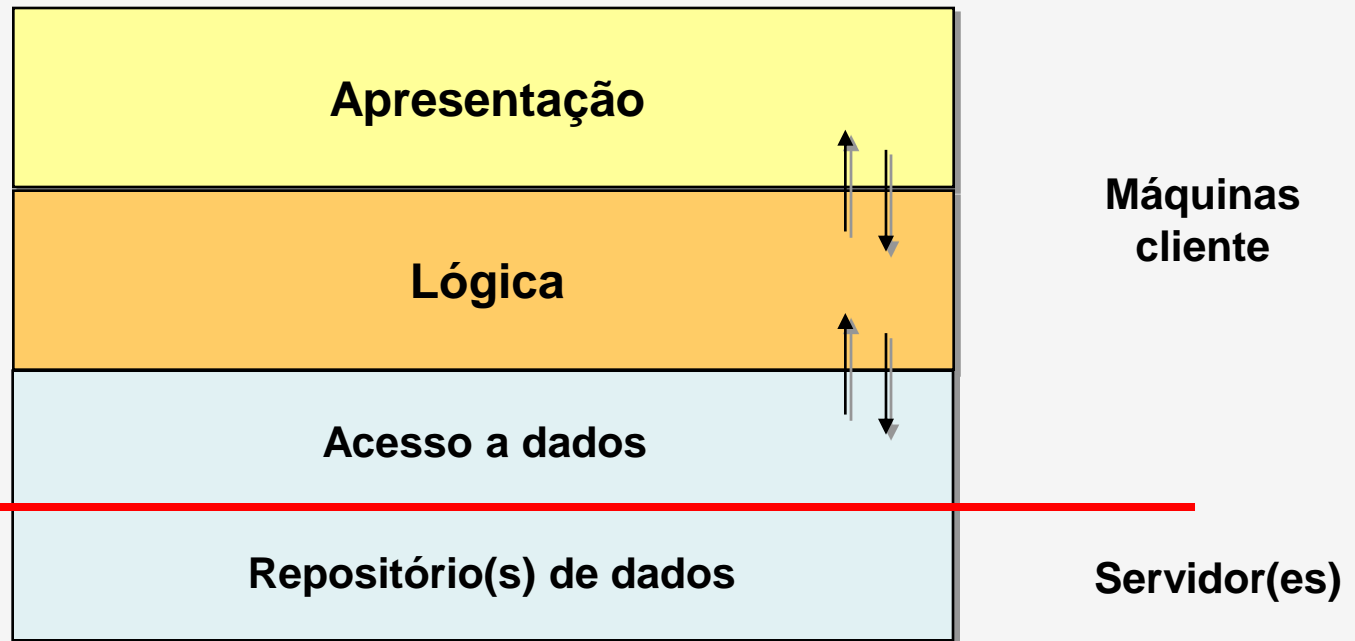
- Monolíticas (não distribuídas)
- Cliente - servidor
 - Baseiam-se na divisão de responsabilidades entre participantes, por exemplo, de acordo com critérios
 - de posse da informação
 - de prestação de serviço (exploração de negócio)
 - Taxonomia (função das divisões físicas)
 - *Fat client*
 - *Tiered*
 - *Rich client*
 - *Thin client*
- *Peer-to-peer*
 - Todos os participantes têm responsabilidades equivalentes

Arquitecturas *Fat client* (1)



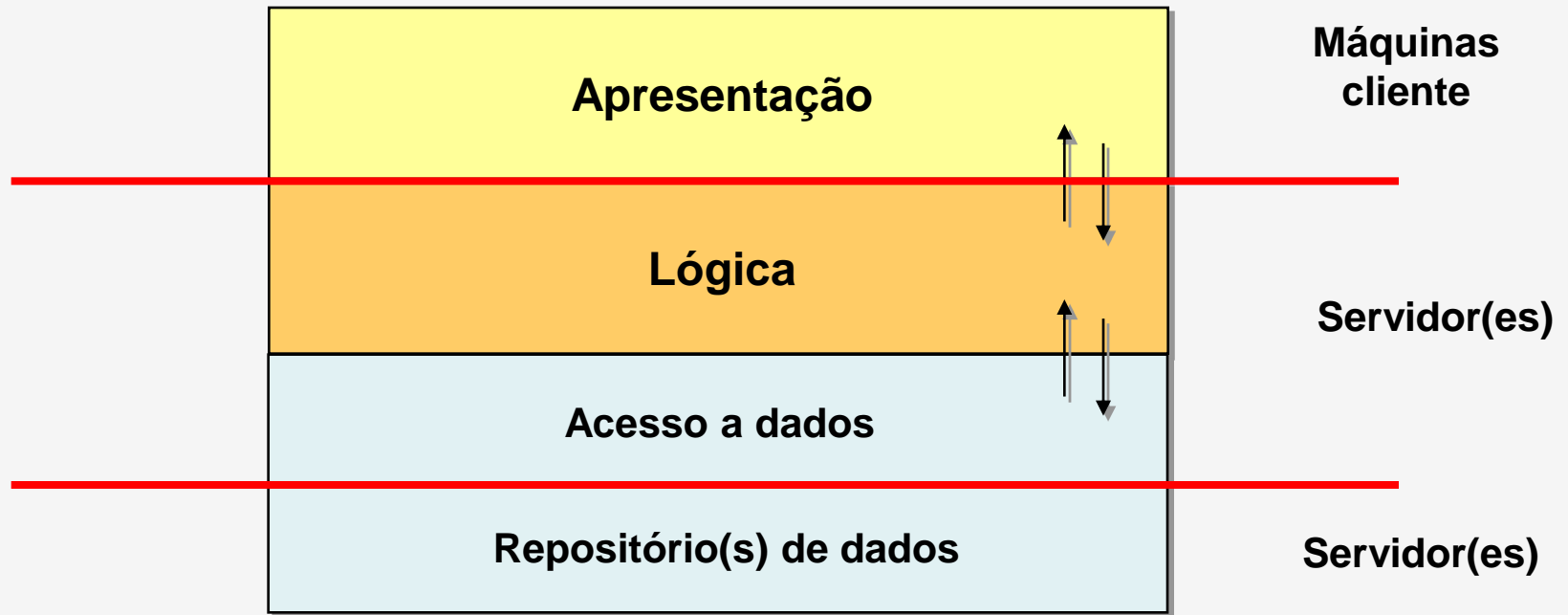
- Quais as vantagens e desvantagens?

Arquitecturas *Fat client* (2)



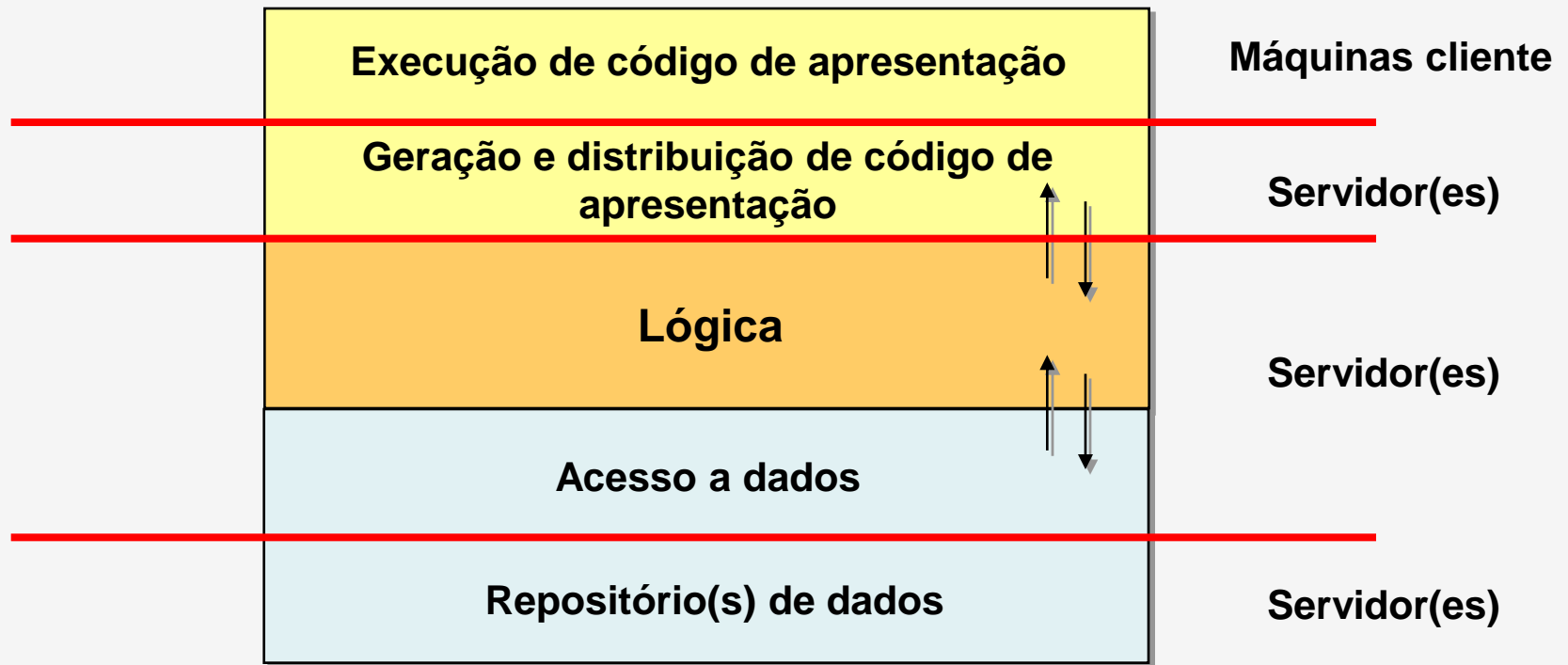
- O que se ganhou?

Arquitecturas Tiered: Rich client



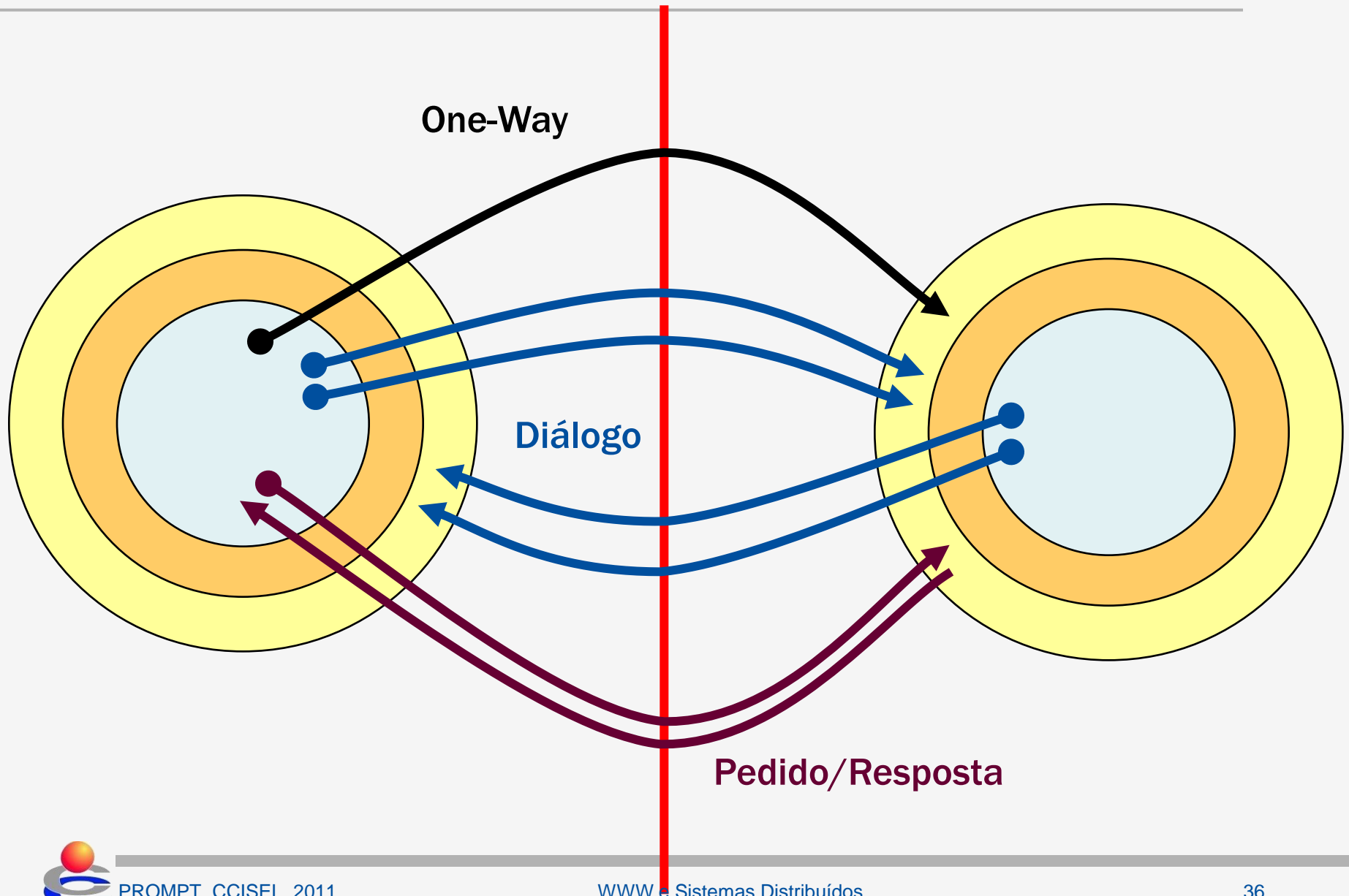
- E agora?

Arquitecturas *Tiered: Thin client*



- Separação física: Até onde?

Modelos de interacção (1)



Modelos de interacção (2)

- Pedido/Resposta
 - Comunicação de natureza síncrona
 - O emissor do pedido não prossegue antes de obter a resposta
 - Desvantagens, citando *Clemens Vasters*
 - "Request/Response forces your impatience upon the responding party"
 - "Unfair behavior because you don't know what stress the other side is having"
- *One-Way*
 - Comunicação de natureza assíncrona
 - O emissor não aguarda mensagem de resposta (logo o receptor pode reagir de forma deferida)
- Diálogos
 - Comunicação de natureza assíncrona onde ambos os intervenientes têm a **iniciativa** de enviar mensagens
 - Equivalente a *One-Way* bidireccional

Desenho de sistemas distribuídos (1)

- Como influenciar as propriedades desejadas?
- Não existem soluções universais (ainda?) para os desafios
 - A adequação das soluções existentes é função do domínio do problema
- No entanto, existem algumas ideias!

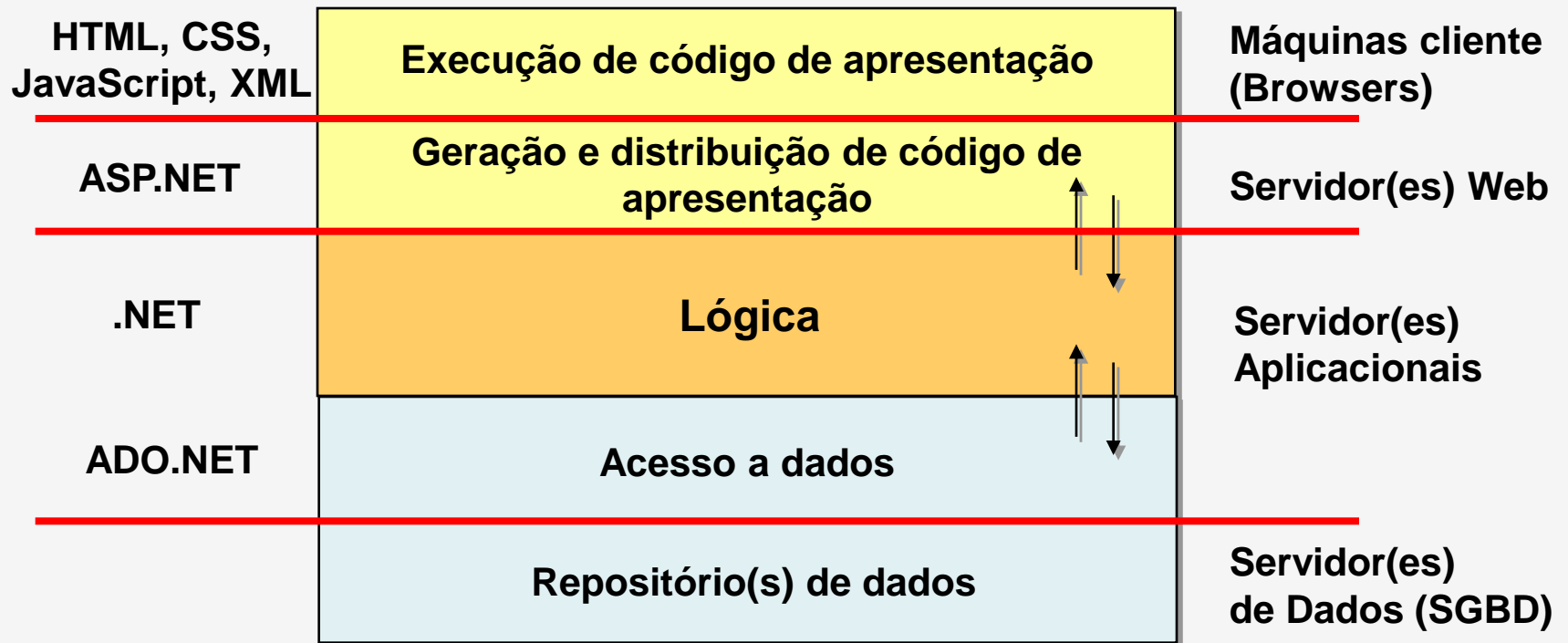
Desenho de sistemas distribuídos (2)

- Algumas ideias: (serão revisitadas)
 - Dados
 - *Caching* (tratamento difere de acordo com a natureza dos dados)
 - Replicação (total ou parcial)
 - Serviços
 - Replicação (*Clustering*)
 - Replicação favorece balanceamento de carga (estático e dinâmico)
 - Favorecimento de:
 - soluções *stateless*
 - modelos de interacção assíncronos
 - Gestão de recursos:
 - Activação *just in time*
 - Utilização de *pools*
 - Estado de conversação

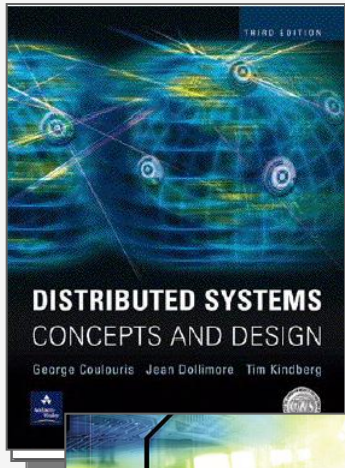
Agenda (cont.)

- A Internet e a *World Wide Web*
- Identificação de recursos
 - *Uniform Resource Locator* (URL)
- Sistemas distribuídos (introdução)
 - Distribuição: causas e consequências
 - Propriedades pretendidas
 - Organização de software em camadas
 - Arquitecturas de sistemas distribuídos
 - Modelos de interacção
 - Desenho de sistemas distribuídos
- O objecto de estudo
 - Aplicações Web

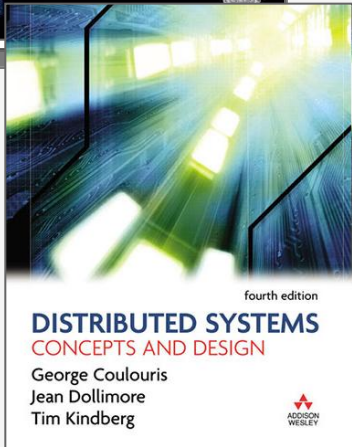
Arquitecturas de aplicações Web – Tiered (revisitado)



Bibliografia



Distributed Systems - Concepts and Design,
G. Coulouris et al, Addison Wesley, 2000



Distributed Systems: Concepts and Design,
Jean Dollimore et al, Addison Wesley, 2005