

Introdução aos Sistemas Distribuídos

CAPÍTULO I

Definição de sistema distribuído

Um sistema distribuído é um conjunto de computadores autónomos, interligados por uma rede de comunicação, percecionado pelos seus utilizadores como um único computador que providencia um serviço ou resolve um problema. – Tanenbaum & Steen, 2002

Um sistema distribuído é composto por computadores interligados em rede (hardware e software) que comunicam e coordenam suas ações apenas através do envio de mensagens. Coulouris et al., 2001

Estamos perante um sistema distribuído existe quando a falha de um computador que nunca ouvimos falar nos impede de fazer o nosso trabalho.
Leslie Lamport

O que é um sistema distribuído?

Conjunto de computadores ligados em rede, com software que permita a partilha de recursos e a coordenação de actividades, oferecendo idealmente um sistema integrado.

O que é um sistema distribuído?

Características:

- Comunicação através de mensagens
- Concorrência
- Partilha de recursos
- Sistema Assíncrono
- Falhas Independentes
- Heterogeneidade

O que é um sistema distribuído?

Características:

- Comunicação através de mensagens
 - Os componentes do sistema comunicam através de mensagens
 - Não existem variáveis globais partilhadas
 - Modelos de programação: cliente/servidor, modelo baseado em objectos, ...
- Concorrência:
 - Vários utilizadores usam o sistema em simultâneo
 - Necessário coordenar o acesso aos recursos partilhados: hw, sw, dados
- Partilha de recursos:
 - Impressoras, discos, ferramentas para trabalho cooperativo, bases de dados
 - Levatam-se questões de segurança:
 - Gestores de recursos controlam o acesso a recursos partilhados

O que é um sistema distribuído?

Características:

- Sistema Assíncrono:
 - Não existe um relógio global
 - Diferentes velocidades de processamento
 - Não existe um limite para o tempo de comunicação
- Falhas Independentes:
 - Falhas na rede (perda de mensagens, duplicação, reordenação)
 - Falhas em unidades de processamento
 - **Nota:** a falha de um componente não impede necessariamente os outros de funcionar

O que é um sistema distribuído?

Características:

- Heterogeneidade
 - Um sistema distribuído pode possuir:
 - Diferentes tipos de rede
 - Diferentes tipos de hardware
 - Diferentes representações de dados, diferente código máquina
 - Diferentes sistemas operativos
 - Diferentes interfaces para os protocolos de comunicação
 - Diferentes linguagens de programação
 - Diferentes representações de estruturas de dados como arrays ou registos, ...
- **Nota:** para tentar resolver o problema da heterogeneidade define-se uma camada desoftware intermédia chamada de *middleware*

The Eight Fallacies of Distributed Computing

“Essentially everyone, when they first build a distributed application, makes the following eight assumptions. All prove to be false in the long run and all cause *big* trouble and *painful* learning experiences.” – *Peter Deutsch*

1. The network is reliable
2. Latency is zero
3. Bandwidth is infinite
4. The network is secure
5. Topology doesn't change
6. There is one administrator
7. Transport cost is zero
8. The network is homogeneous

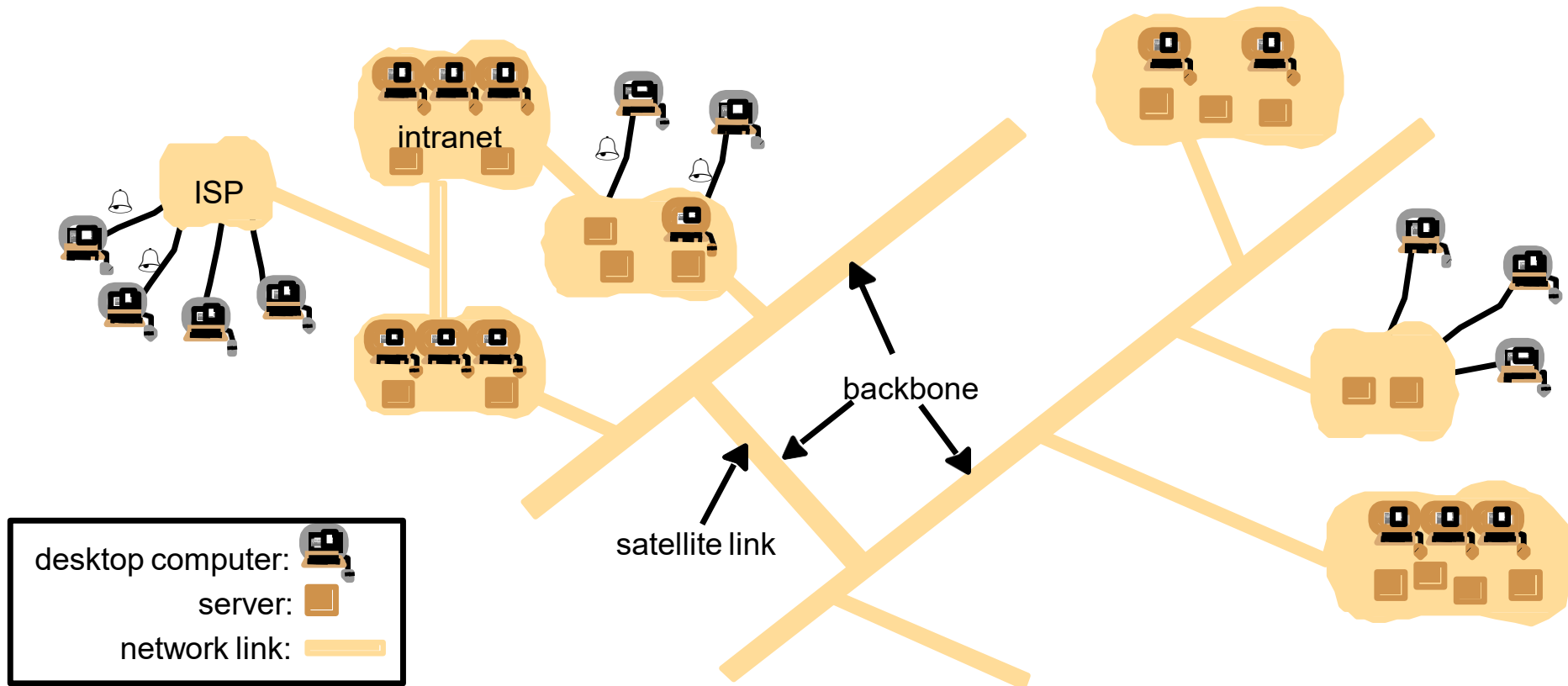
Redes

Estão em todo o lado:

- Internet
- Redes de Telefones Móveis
- Redes Corporativas
- Redes de Fábrica
- Redes de Campus
- Redes Domésticas
- Redes In-Car
- Redes Sem Fio (Wireless Networks)
- Redes Ad Hoc Móveis (MANET)
- Redes de Sensores (Sensor Networks)

A Internet

Infraestrutura de redes(s) que liga milhões de computadores



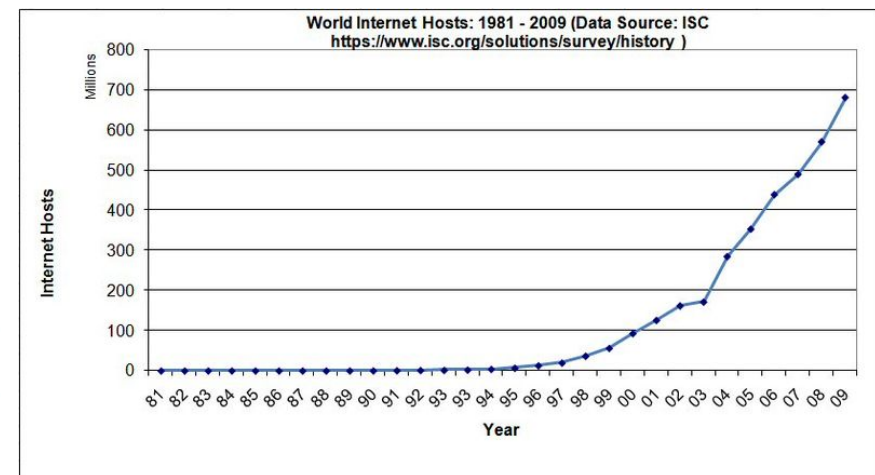
A Internet

O desenho e construção dos mecanismos de comunicação da Internet (protocolos Internet) permitiu que um programa em execução num qualquer ponto da rede possa enviar mensagens a programas em qualquer outro lugar.

World Wide Web (Web): Modelo de partilha de informação construído no topo da internet. Conjunto de servidores web acedidos através da internet a partir de web browsers.

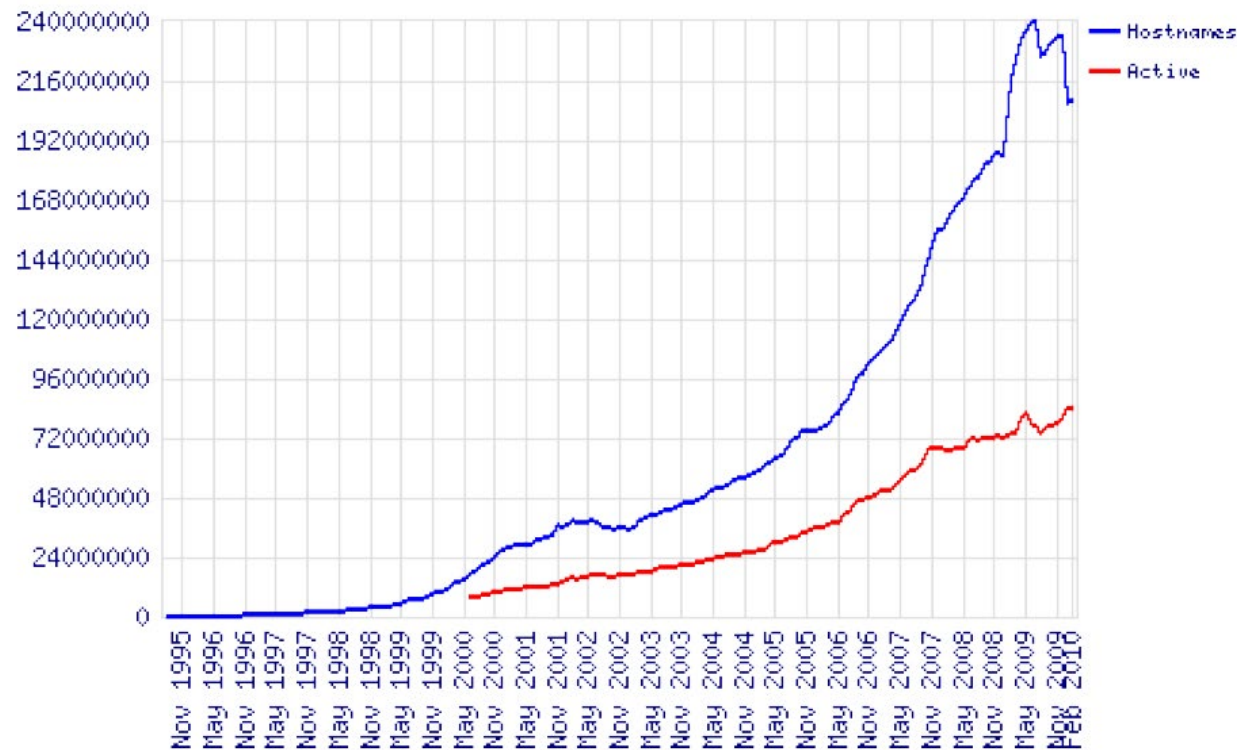
Computadores na Internet:

<i>Date</i>	<i>Computers</i>	<i>Web servers</i>
1979, Dec.	188	0
1989, July	130,000	0
1999, July	56,218,000	5,560,866



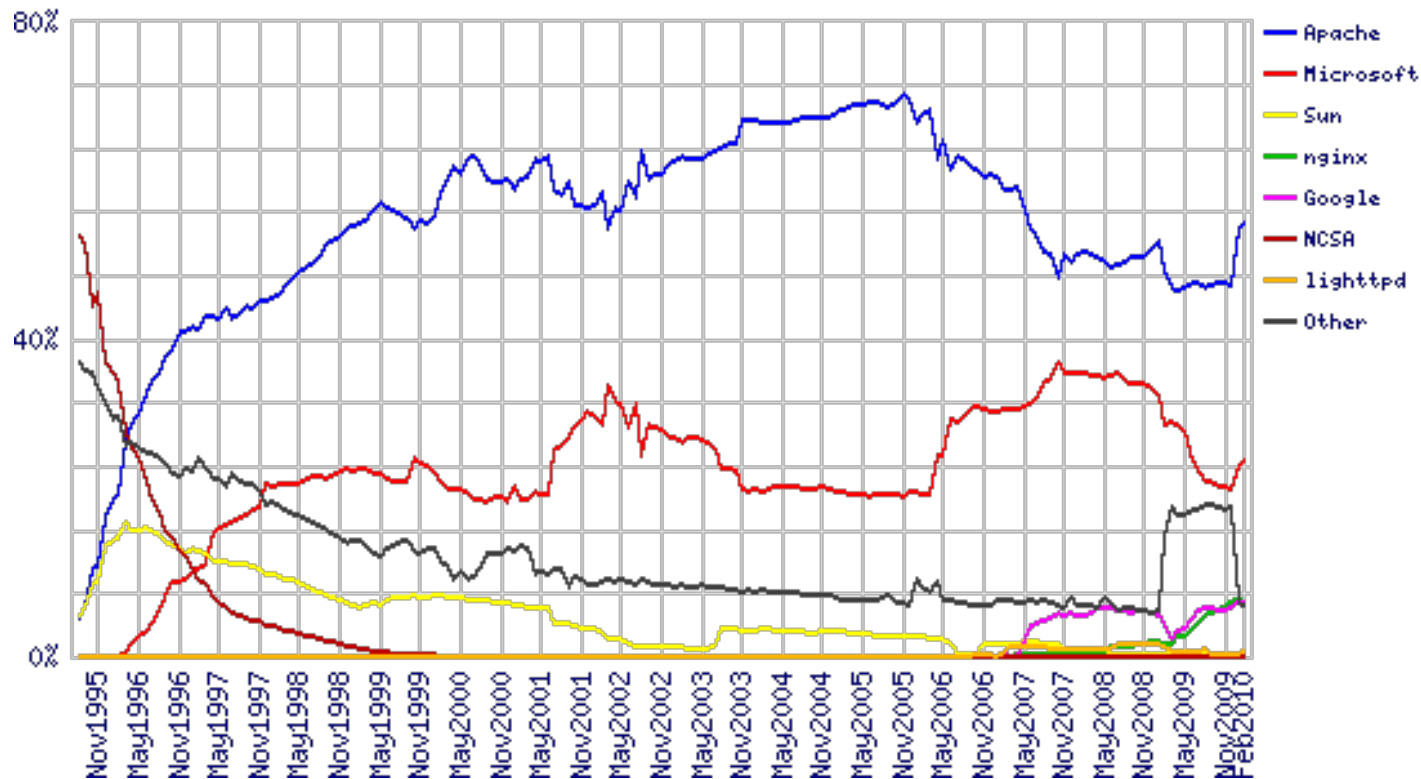
Exemplos de Sistemas Distribuídos: Aplicações Web

Total Sites Across All Domains from August 1995 to February 2010:



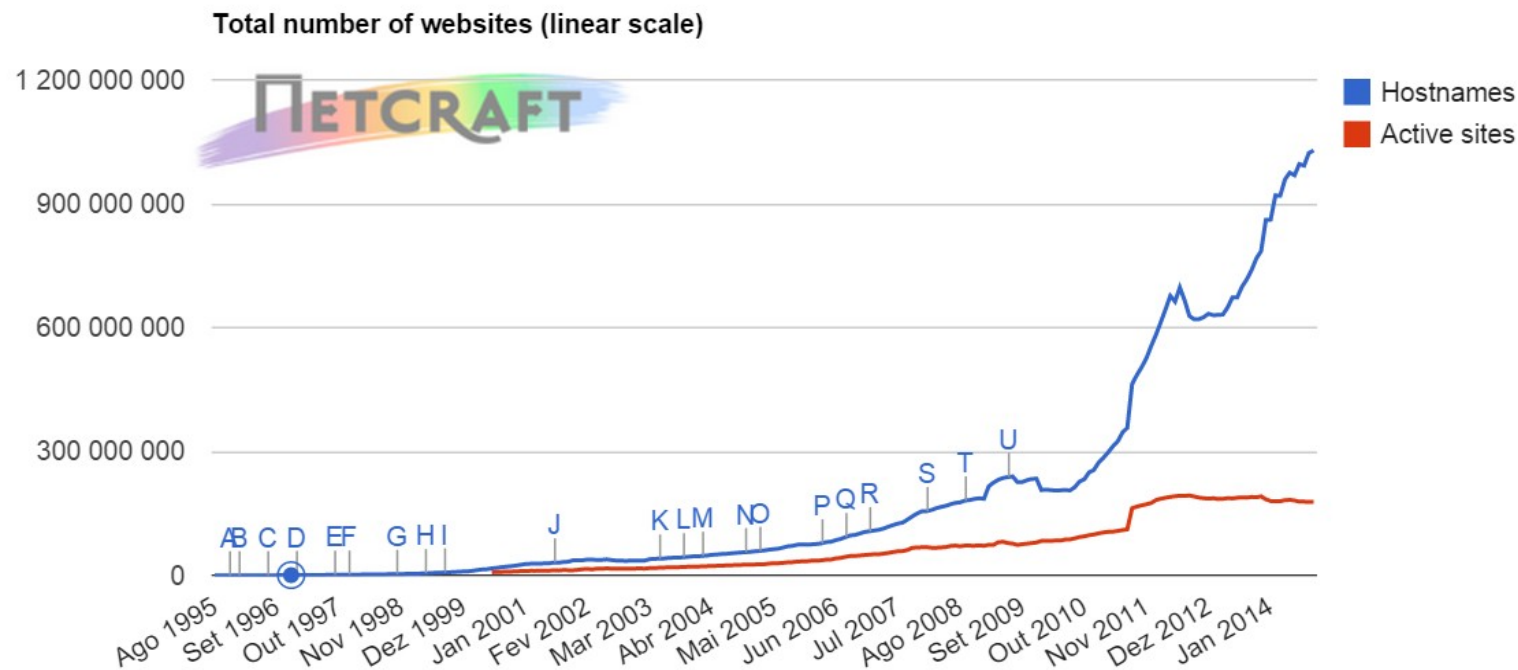
Exemplos de Sistemas Distribuídos: Aplicações Web

Market Share for Top Servers Across All Domains from August 1995 to February 2010:



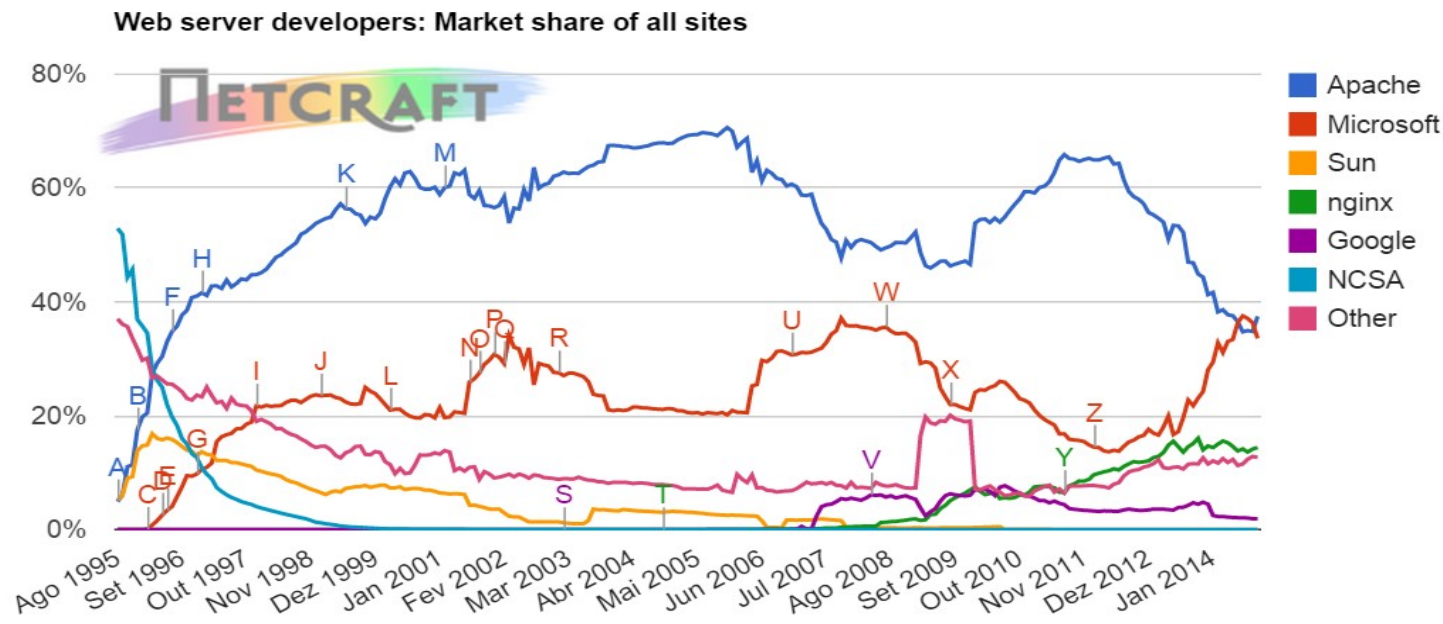
Exemplos de Sistemas Distribuídos: Aplicações Web

Total Number of Websites (linear scale) from August 1995 to October 2014:



Exemplos de Sistemas Distribuídos: Aplicações Web

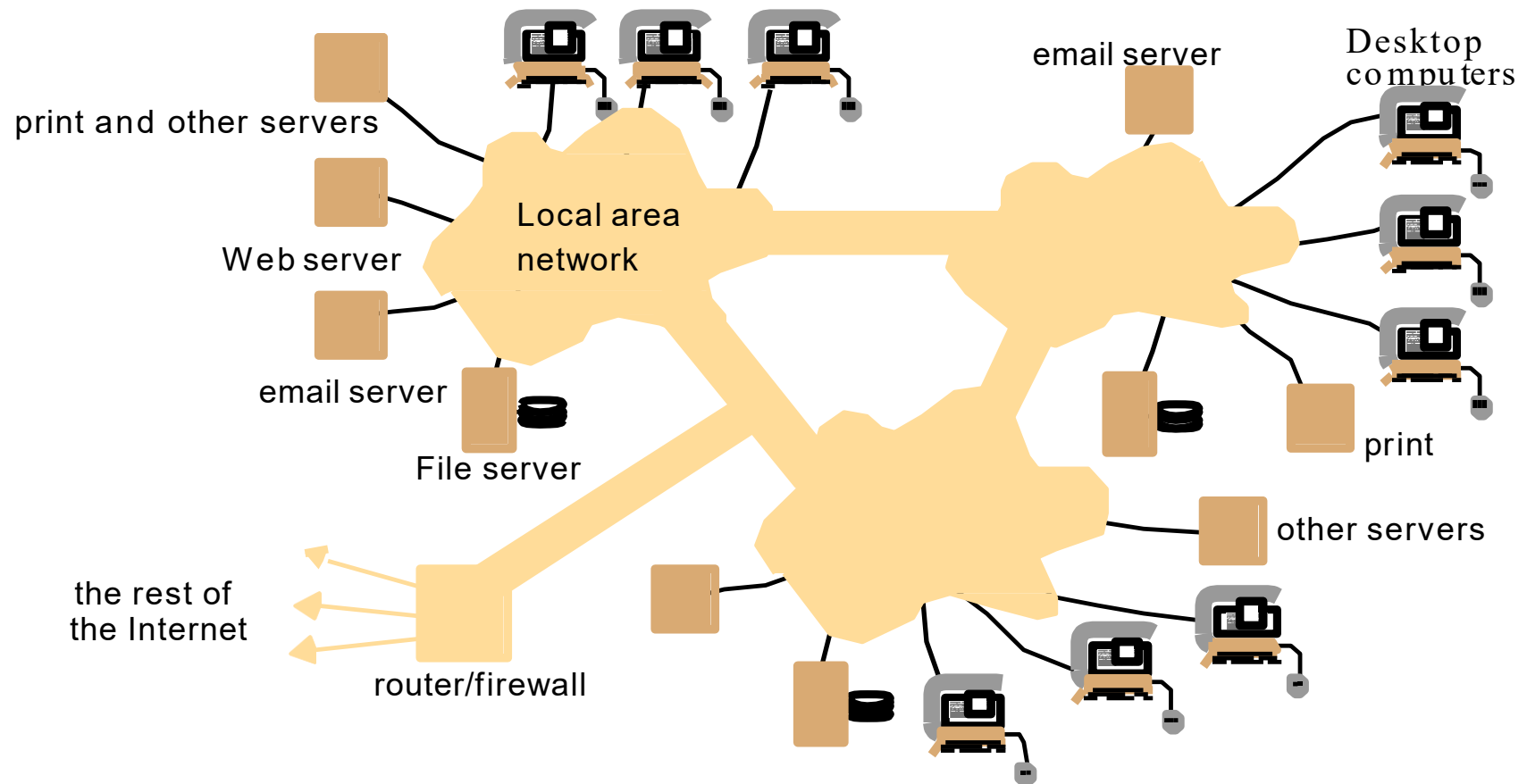
Total Number of Web Server Developers: Market share of all sites from August 1995 to October 2014:



Exemplos de Sistemas Distribuídos: Aplicações Web

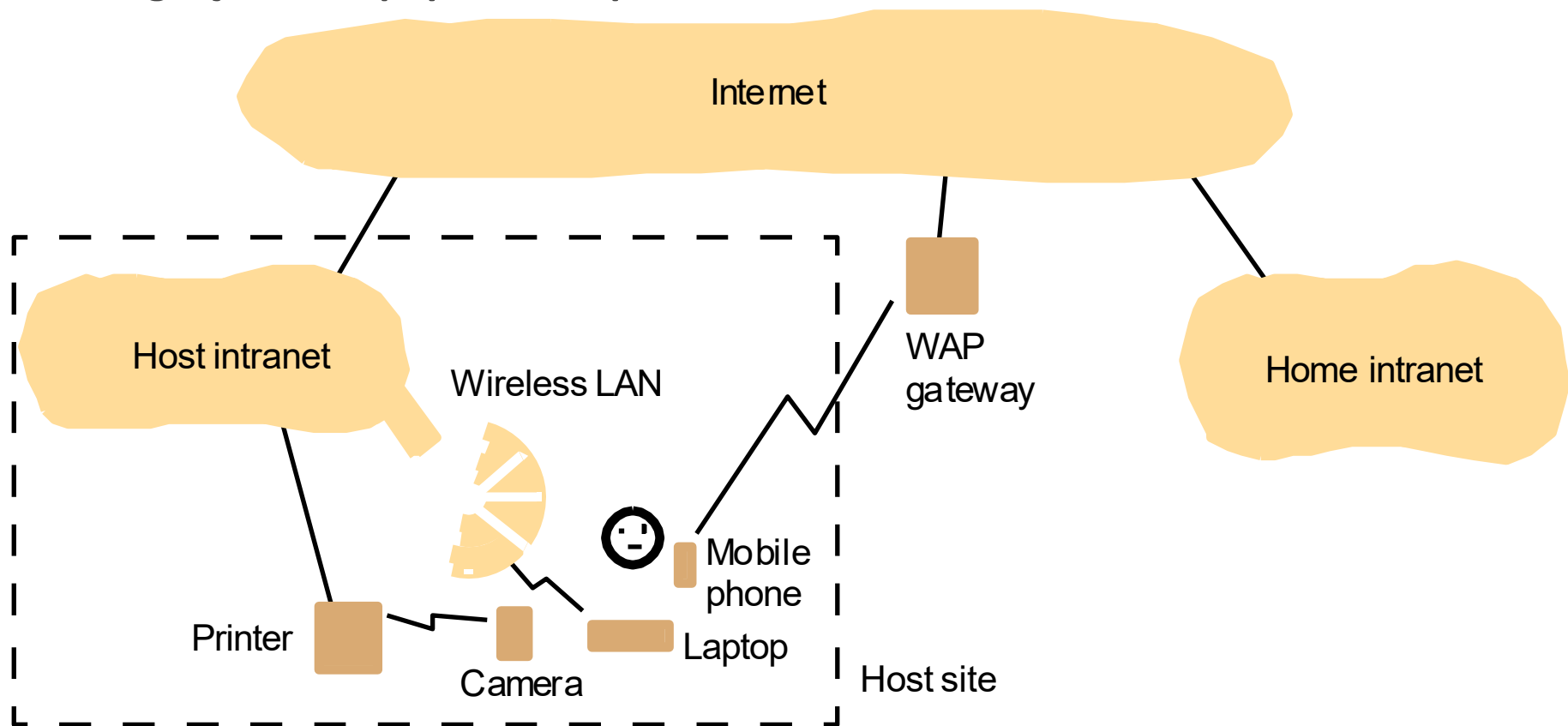


Intranets



Computação Móvel e Ubíqua

Integração de equipamento portátil em sistemas distribuídos



Computação Móvel e Ubíqua

Exemplos:

- Computadores portáteis
- Telefones móveis
- PDA's - Personal Digital Assistants
- Máquinas fotográficas digitais
- Pagers
- Wearable devices:
 - relógios com processador, ...
- Sistemas embebidos (computadores dentro de um produto):
 - Electrodomésticos, carros, ...

O termo computação Ubíqua pretende designar sistemas cuja utilização está de tal forma integrada na funcionalidade do produto que é transparente para o utilizador:

- Maiores restrições em termos de: custo, tamanho, potência/autonomia.

Outros Sistemas Distribuídos

Exemplos:

- Correio eletrónico
- Sistemas de ficheiros distribuídos
- FTP, Telnet
- Chat, I.M.
- Aplicações críticas (exigem fiabilidade e segurança):
 - Reserva de bilhetes em companhias de transportes
 - Comércio eletrónico
 - Máquinas Multibanco

Desafios na implementação de sistemas distribuídos

Escalabilidade

Abertura

Tolerância a Falhas

Segurança

Transparência

Desafios na implementação de sistemas distribuídos

Escalabilidade

Capacidade de o sistema se manter a funcionar de forma correta e à velocidade desejada independentemente do número de utilizadores.

É necessário:

- Desenhar o software de forma a que o aumento de utilizadores não exija grandes alterações
- Evitar algoritmos e estruturas de dados centralizadas (replicação de dados se necessário)
- Controlar o aumento de custos devido à disponibilização de mais recursos
- Controlar a perda de performance (replicação de serviços)
- Evitar o transbordo de certos limites de recursos (ex. Endereço IP com 32 bits, insuficiente)

Desafios na implementação de sistemas distribuídos

Sistema Aberto (“Openness”)

Capacidade de o sistema ser extensível, quer em hardware quer em software

- Novos componentes devem poder ser adicionados sem por em causa o funcionamento dos já existentes, e poder comunicar com eles.

Para isso é importante que:

- Sejam conhecidas as interfaces dos novos componentes através da publicação da sua documentação
- Utilizar protocolos e formatos standard

Exemplo de publicação de interfaces:

- Request For Comment (RFCs): www.ietf.org
 - Contém as especificações dos protocolos internet desde o início dos anos 80

Desafios na implementação de sistemas distribuídos

Tolerância a Falhas

Tolerar uma falha significa conter os seus efeitos de forma a que o sistema continue a funcionar.

Para isso é necessário:

1. Detecção da falha

- Dados corrompidos (mensagens ou ficheiros) podem ser detectados através de somas de verificação.

2. Localização da falha

- Se não houve resposta a um pedido, o que significa?
 - Falha na rede
 - Falha no nó destino
 - **Como distinguir?**

Desafios na implementação de sistemas distribuídos

Tolerância a Falhas

3. Mascarar / Tolerar a falha

- Algumas falhas podem ser ocultadas do utilizador se se utilizar redundância suficiente:
 - Quando uma mensagem não chega, pode ser retransmitida
 - Um ficheiro pode ser escrito em duplicado (um em cada disco)
 - Entre cada dois “routers” da internet, devem sempre existir dois percursos
 - Uma base de dados pode ser replicada em vários servidores

Desafios na implementação de sistemas distribuídos

Segurança

- Manter recursos computacionais seguros significa:
- Manter o nível de confidencialidade exigido pelos utilizadores
 - Proteção contra acessos não autorizados
- Garantir a integridade dos dados
 - Proteção contra alteração ou corrupção de dados ou programas
- Manter a disponibilidade do sistema
 - Proteção contra interferências com os meios de acesso aos recursos
- Alguns problemas por resolver:
 - Ataques do tipo “negação de serviço” (denial of service)
 - Segurança do código móvel

Desafios na implementação de sistemas distribuídos

Transparência

O sistema deve ser visto como um todo e não como uma coleção de componentes distribuídos.

No standard de “Open Distributed Processing” (ODP) foram definidos os seguintes tipos de transparência:

- Acesso
- Localização
- Concorrência
- Replicação
- Falhas
- Migração
- Desempenho
- Escalabilidade

Mais Importantes: Acesso e Localização em conjunto são referidos como transparência de rede.

Desafios na implementação de sistemas distribuídos

Transparência de acesso

Permite que o acesso a recursos locais e a recursos remotos seja feito através das mesmas operações (i.é, usando a mesma interface).

Exemplo: WinSCP vs FTP

Transparência de localização

Permite que os recursos possam ser acedidos sem o conhecimento da sua localização.

Exemplo: Programas de correio eletrónico

Transparência de concorrência

Permite que os vários clientes de um componente não necessitem de ter em conta o acesso concorrente ao componente.

Desafios na implementação de sistemas distribuídos

Transparência de replicação

Permite que os clientes de um componente não se apercebam se existe replicação e estão a usar uma réplica e não o original;

A utilização de várias instâncias de um componente pode ocorrer por:

- razões de desempenho ou de fiabilidade;

Os utilizadores do componente não necessitem de saber que o componente possa ser replicado.

Transparência de Falhas

Permite que o sistema funcione na presença de falhas de hardware ou software sem que utilizadores e programadores saibam como as falhas foram ultrapassadas.

Exemplo: um sistema de e-mail pode retransmitir uma mensagem até que a mesma seja entregue com sucesso.

Desafios na implementação de sistemas distribuídos

Transparência de Migração

Permite que um recurso possa mudar de localização sem que isso afete a sua utilização.

Exemplo: Telemóveis em movimento

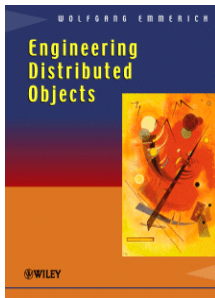
Transparência de Desempenho

Permite que o sistema seja configurado para melhorar o seu desempenho sem que os utilizadores se apercebam.

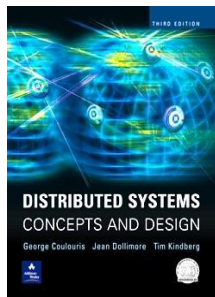
Transparência de Escalabilidade

Permite que o sistema seja expandido sem que os utilizadores se apercebam de como isso foi conseguido.

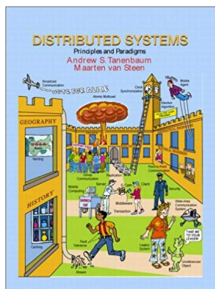
Bibliografia



From: Wolfgang Emmerich
Engineering Distributed Objects
John Wiley & Sons, Ltd 2000



From: Coulouris, Dollimore and Kindberg
Distributed Systems: Concepts and Design
Edition 4 © Addison-Wesley 2005



From: Andrew S., Tanenbaum and Van Steen, Maarten
Distributed Systems: Principles and Paradigms
Edition 2 © Pearson 2013

Questões?