

Redes de Computadores

Luiz Filipe Menezes Vieira
lfvieira@dcc.ufmg.br

Departamento de Ciência da Computação
Universidade Federal de Minas Gerais

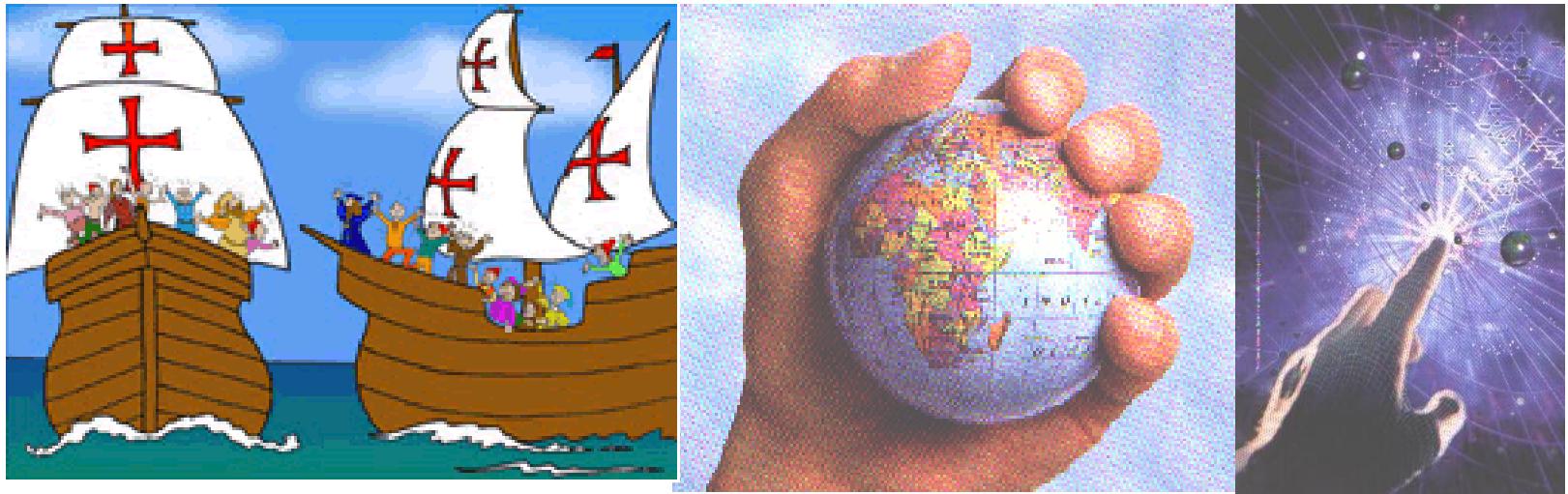
O Problema da Comunicação

Sumário

- Primórdios da comunicação sem fio
- Algumas definições e questões associadas
- Paradigmas computacionais
- Computação ubíqua
- Classificação dos protocolos
- Projeto de protocolos
- Trabalho prático 1

Primórdios da Comunicação Sem Fio

Qual é o tamanho do mundo do ponto de vista da comunicação?



Ao longo do tempo, o homem tem conseguido atingir distâncias cada vez maiores, em menor tempo ...
... e o mundo tem ficado cada vez “menor”!

Primórdios da Comunicação Sem Fio (CSF)

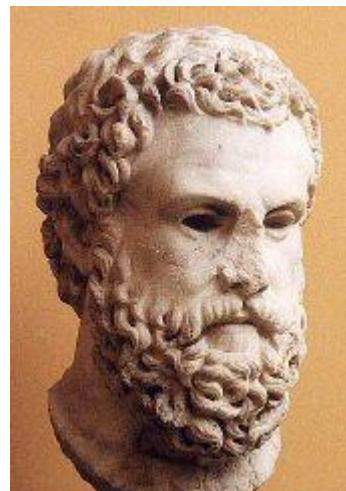
- Fogo
- Sinais de fumaça
- Bandeiras
- Telégrafo óptico
- ...



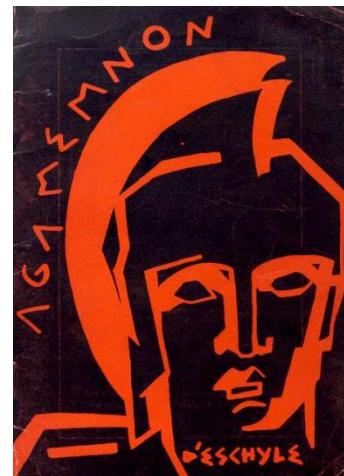
Pombo correio

Grécia antiga

Peça Agamemnon, escrita por Aeschylus em 458 a.C., descreve como sinais de fogo foram usados, supostamente, para comunicar a queda de Tróia para Atenas numa distância acima de 450 km



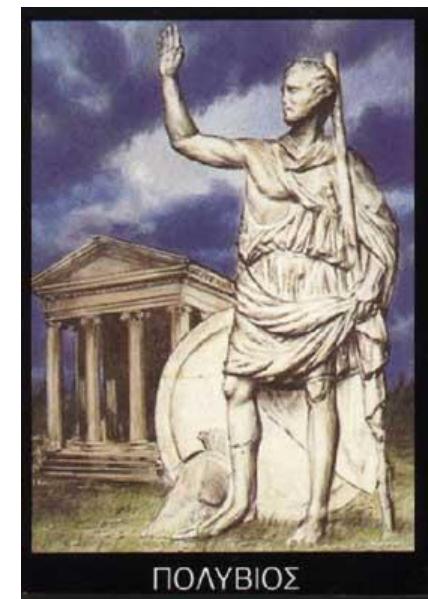
Aeschylus



Grécia antiga

- Problema com sinais de fogo:
 - ◆ Número de mensagens diferentes que podem ser enviadas é bastante limitado
- Problema identificado por Polybius, historiador grego do segundo século a.C.
 - ◆ Não havia como mandar uma nova msg que necessitasse atenção imediata

Estátua de Polybius achada em Cleitor, Grécia



Grécia antiga: Solução de Polybius

- Dois conjuntos de cinco tochas
- Suposições:
 - ◆ Há um acordo prévio entre os sinalizadores sobre a codificação das letras e as regras de comunicação a serem usadas
 - ◆ Alfabeto dividido em cinco conjuntos de cinco letras

Grécia antiga: Solução de Polybius

- Regras de comunicação:
 - ◆ Sinalizador que inicia a transmissão levanta duas tochas
 - ◆ Sinalizador que irá receber a mensagem levanta duas tochas como confirmação que está pronto
 - ◆ Sinalizador TX usa o conjunto de tochas da esquerda para indicar o conjunto do alfabeto a ser usado e o conjunto de tochas da direita para indicar a letra a ser transmitida
- ➔ Mensagens arbitrárias poderiam ser enviadas
- ➔ Método usado por quase 20 séculos!

Sinais de fumaça



Voz humana?



Técnicas de CSF mais “recentes”

Telégrafo óptico



Telégrafo em Condé, França
Novembro 1794



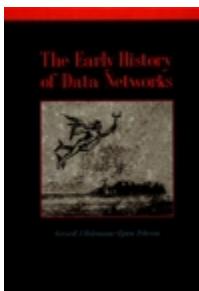
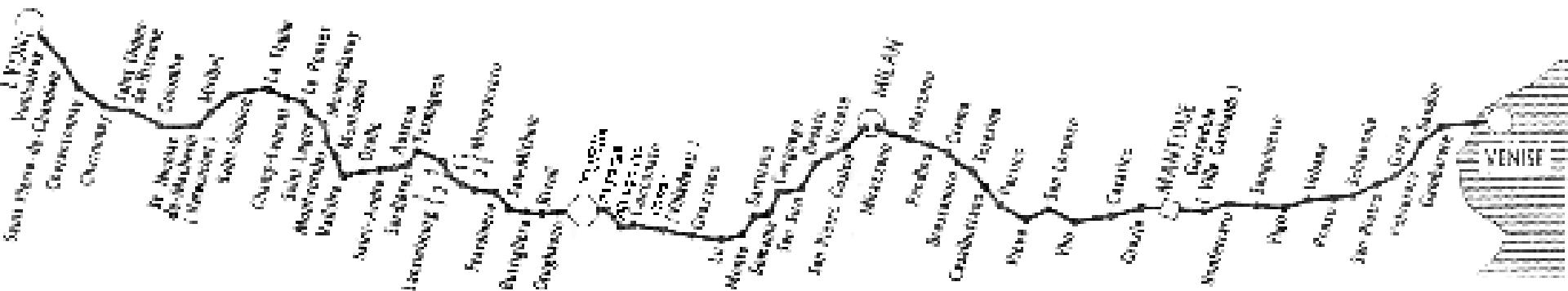
Primeiro Telégrafo Óptico
Paris-Lille, 1794, Claude Chappe



Técnicas de CSF mais “recentes”

Telégrafo óptico

Primeira rede de telecomunicações da Europa:
Lyon-Turin-Milão-Veneza



The Early History of Data Networks
Gerard J. Holzmann, Björn Pehrson
ISBN: 0-8186-6782-6, Hardcover, 304 pages
November 1994, Wiley-IEEE Computer Society Press

Técnicas de CSF mais “recentes”

Datagrama IP

Network Working Group
Request for Comments: 1149

D. Waitzman
BBN STC
1 April 1990

A Standard for the Transmission of IP Datagrams on Avian Carriers

Status of this Memo

This memo describes an experimental method for the encapsulation of IP datagrams in avian carriers. This specification is primarily useful in Metropolitan Area Networks. This is an experimental, not recommended standard. Distribution of this memo is unlimited.

Overview and Rational

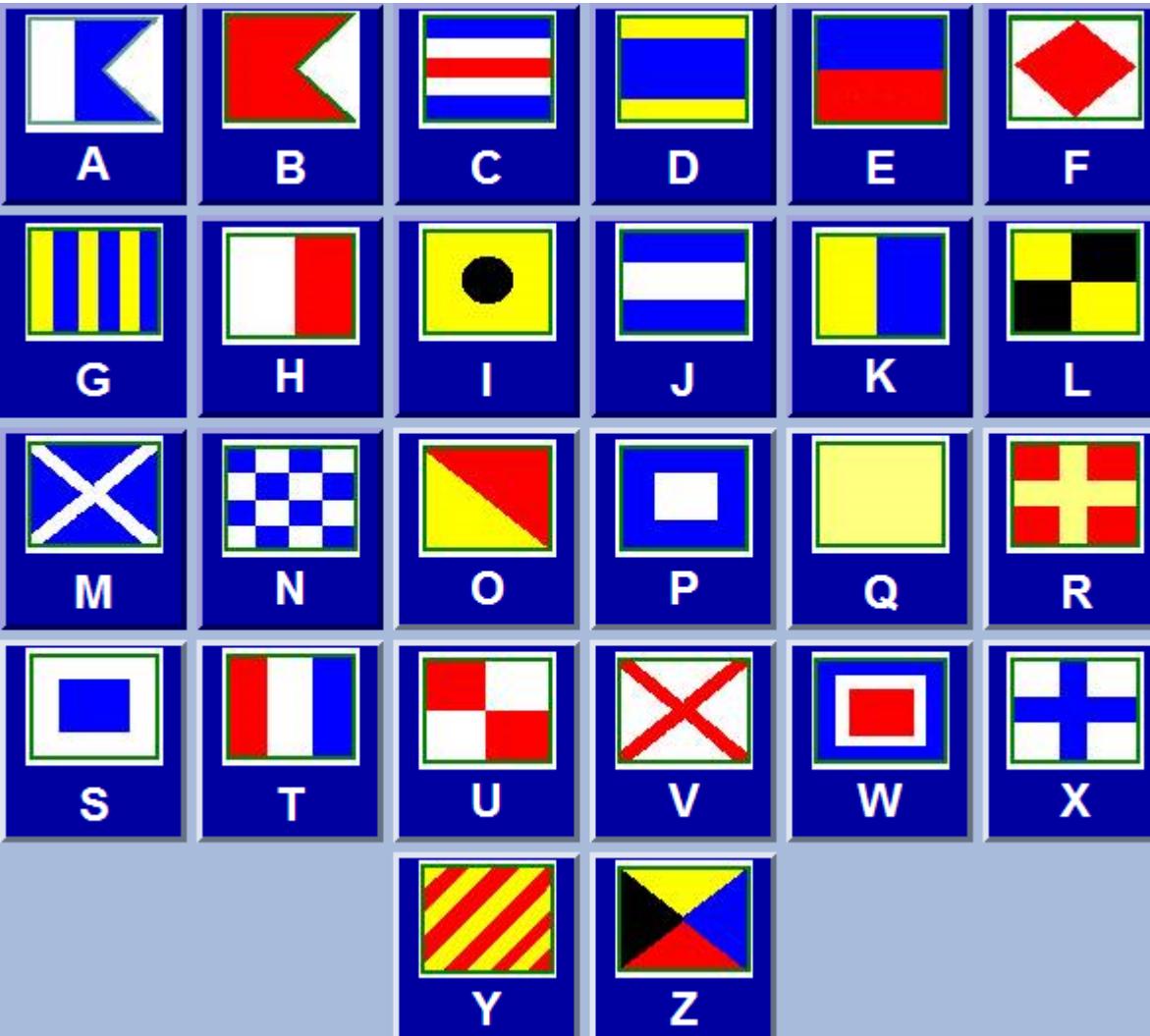
Avian carriers can provide high delay, low throughput, and low altitude service. The connection topology is limited to a single point-to-point path for each carrier, used with standard carriers, but many carriers can be used without significant interference with each other, outside of early spring. This is because of the 3D ether space available to the carriers, in contrast to the 1D ether used by IEEE802.3. The carriers have an intrinsic collision avoidance system, which increases availability. Unlike some network technologies, such as packet radio, communication is not limited to line-of-sight distance. Connection oriented service is available in some cities, usually based upon a central hub topology.

D. Waitzman. *A Standard for the Transmission of IP Datagrams on Avian Carriers*, RFC 1149, April 1990.



Técnicas de CSF mais “recentes”

Sinalização náutica – Bandeiras



As bandeiras de hastear utilizadas na sinalização náutica internacional para cada uma das letras do alfabeto.

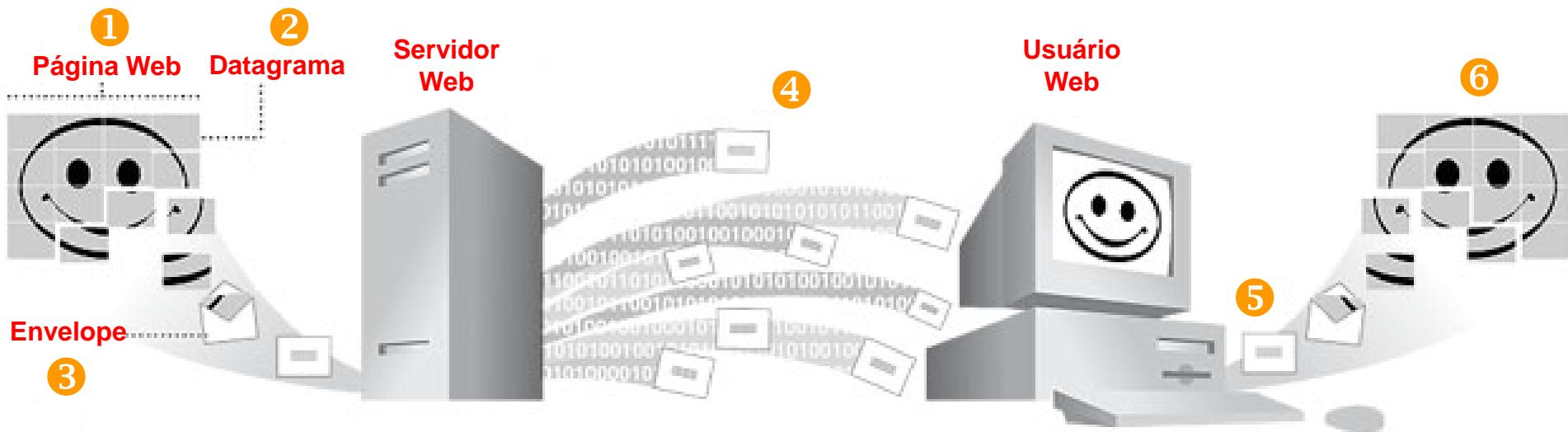
Ref: <http://www.naval.com.br/conhecimentos/gloss/gloss.htm>

Algumas Definições e Questões Associadas

O problema básico das redes de comunicação de dados

- Comunicação (troca de dados) entre “entidades comunicantes” de forma “adequada”
 - ➔ Problema mais amplo que existe em cenários onde há necessidade de trocar dados
 - Protocolos de comunicação são responsáveis por essa comunicação:
 - ◆ Conjunto de regras para a troca de dados entre entidades comunicantes
- ➔ Um protocolo é um algoritmo distribuído

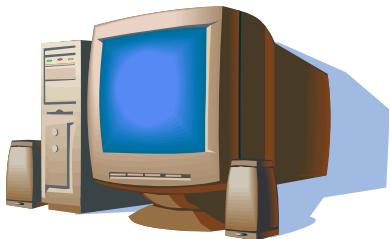
Protocolo de comunicação



Redes de computadores (Elementos computacionais)

- Conjunto de computadores autônomos interconectados entre si por uma tecnologia de comunicação de dados
- Atualmente, as redes interconectam mais que “computadores”
 - ◆ Na verdade, elementos computacionais!

Elementos computacionais “tradicionais”



Computador
pessoal



Laptop



PDA



Telefone
celular



Sensor

Outros elementos computacionais



Pager



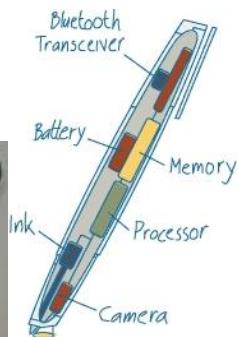
*Relógio
Telefone*



*Relógio
Telefone
Câmera*



*Telefone
Televisão*



*Caneta Scanner
Comunicador*

*Geladeira
Computador*

*Máquina de lavar
Computador*

*Microondas
Computador*

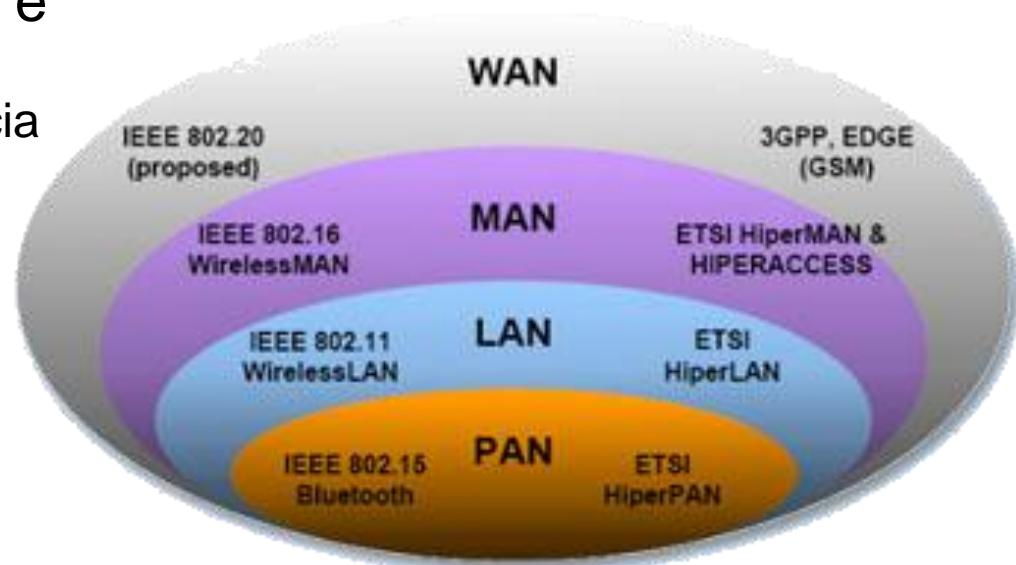
Redes usadas na comunicação de dados

Existem vários tipos de infraestruturas de comunicação com e sem fio:

- ◆ WAN – Redes de longa distância
- ◆ MAN – Redes metropolitanas
- ◆ LAN – Redes locais
- ◆ PAN – Redes pessoais

que levam a diferentes tipos de redes:

- ◆ Internet
- ◆ Redes locais (LANs, WLANs)
- ◆ Redes de sensores (WSNs)
- ◆ Redes móveis ad hoc (MANETs)
- ◆ Redes celulares (2G, 2.5G, 3G)



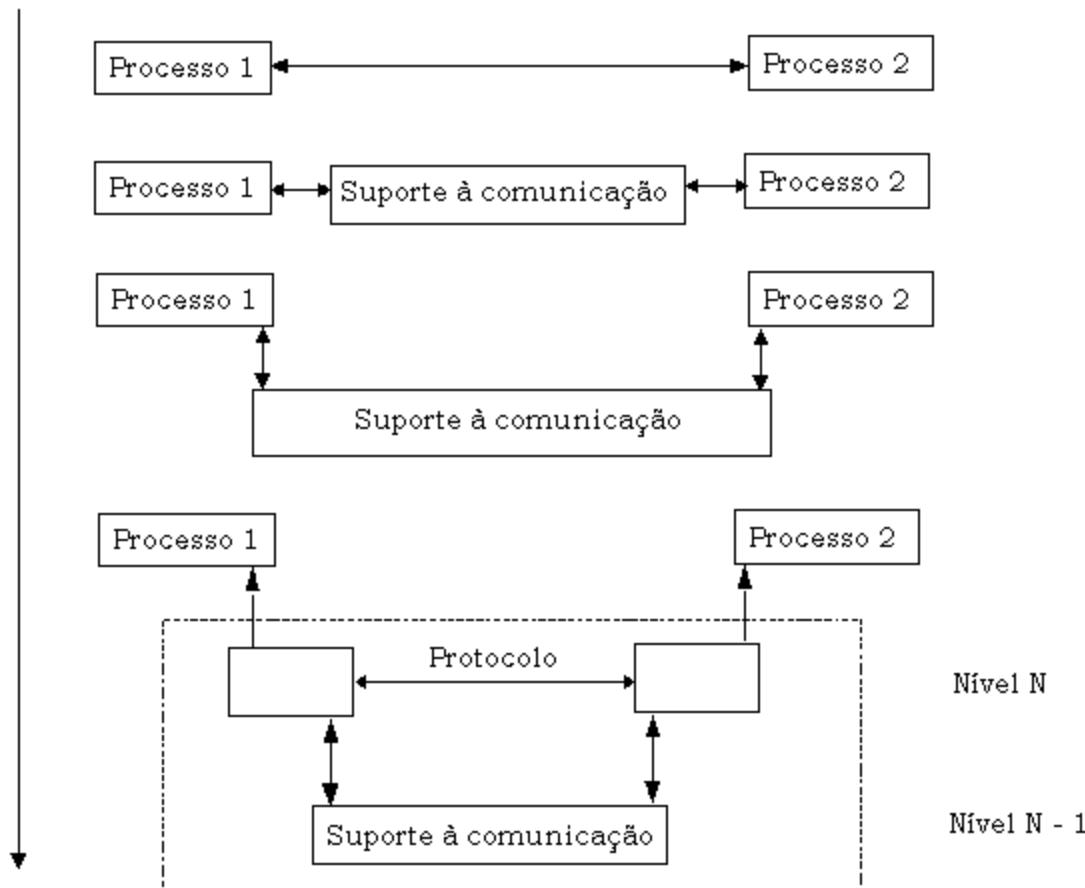
→ Redes com características diferentes

Como as redes são projetadas?

- De forma similar a qualquer outro sistema complexo
 - ◆ Software ➔ Módulos
 - ◆ Redes ➔ Camadas
- Objetivos:
 - ◆ Auxiliar no domínio da complexidade
 - ◆ Aumentar a eficiência e a facilidade de utilização
- Tendência para concepção e desenvolvimento de redes de computadores e sistemas distribuídos
- Implementação: *threads*, micro-kernels, etc

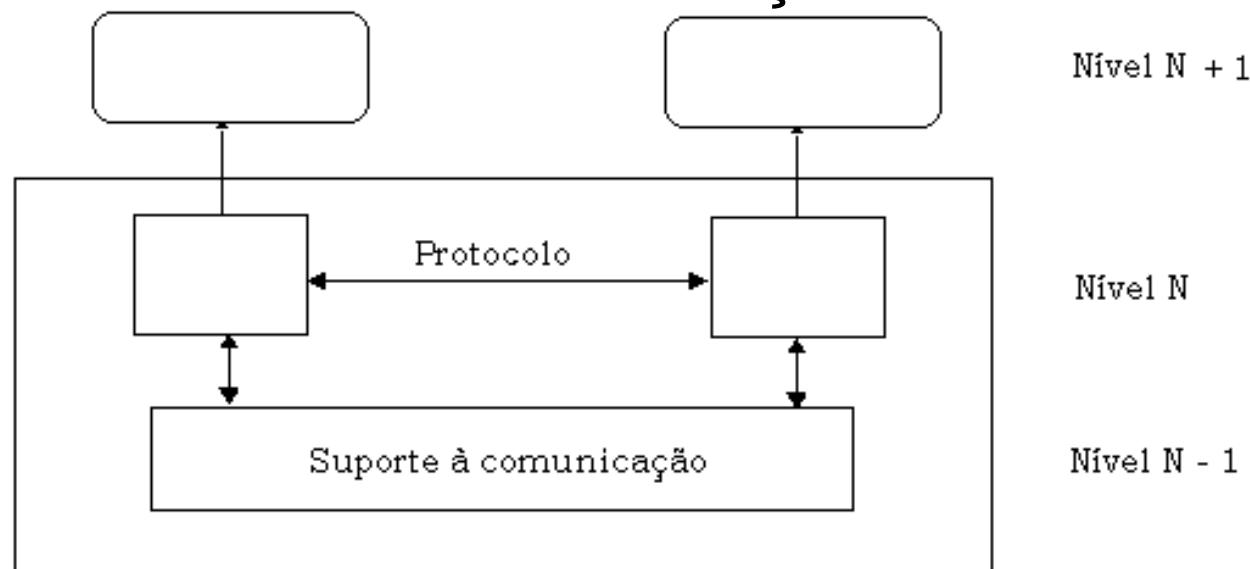
Organização em camadas das redes

Refinamento de
Abstração



Organização em camadas das redes

- Nível (hierarquia) → Camada (serviço)
- Interface com camada $N + 1$ (superior)
- Interface com camada $N - 1$ (inferior)
- Um protocolo é uma abstração lógica do processo físico de comunicação



Camada N

- Provê um conjunto de serviços para as camadas superiores
- Esconde detalhes da implementação dos serviços
- É implementada baseando-se nos serviços oferecidos pela camada $N-1$

Organização em camadas das redes

Vantagens

- Separação de funções
- Desenvolvimento por vários grupos
- Acoplamento em níveis intermediários
- Independência para implementação e modificação
- Facilidade para interligação de sistemas heterogêneos
- Facilidade para testes e depuração

Organização em camadas das redes

Desvantagens

- Overhead para o tratamento das unidades de informação das camadas
- Dependendo da pilha de protocolos pode haver duplicação de funções nas camadas
- Simplicidade e eficiência de um único nível para certas aplicações

Questão

- O que significa negociação no contexto de protocolos de uma rede de computadores?
- Dê alguns exemplos de tipos de negociação

Paradigmas Computacionais

Os quatro paradigmas da computação

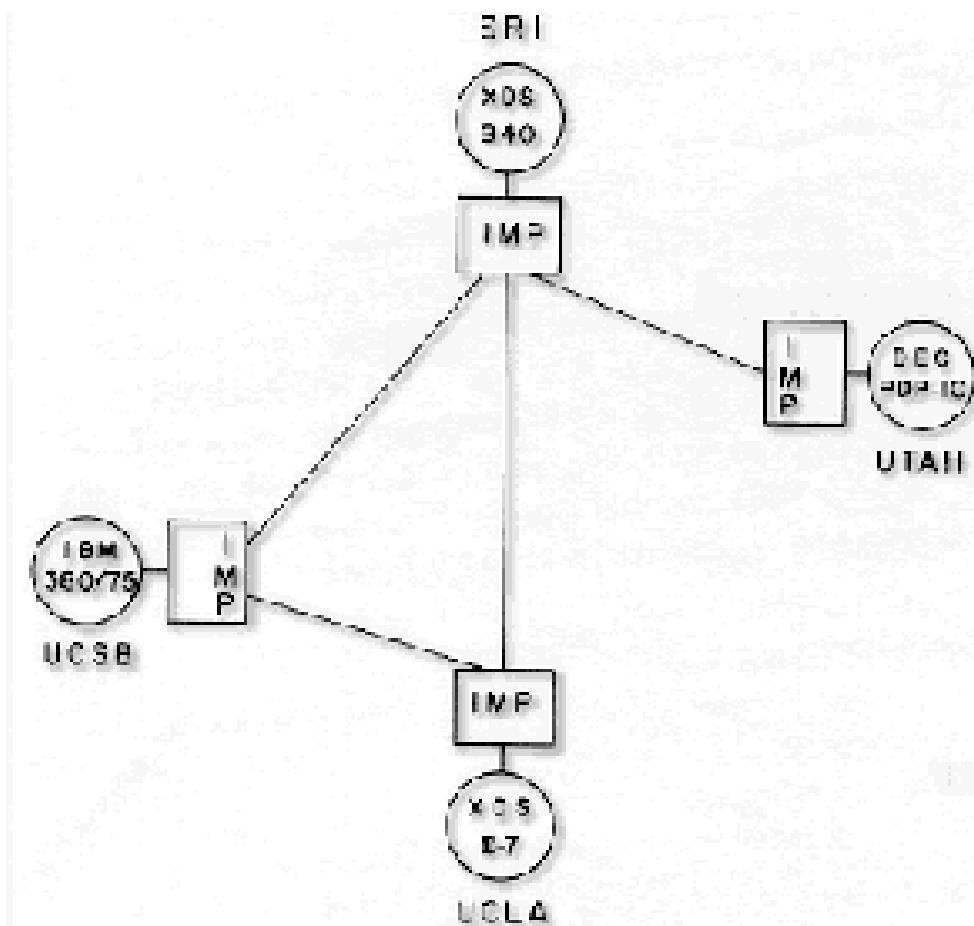
Fonte: **Networked computing in the 1990s.**
L.G. Tesler,
Scientific American,
265(3):54-61, Sept.
1991.

	Batch	Time-Sharing	Desktop	Network
Decade	1960s	1970s	1980s	1990s
Technology	Medium scale integration	Large scale integration	Very large scale	Ultra large scale
Location	Computer room	Terminal room	Desktop	Mobile
Users	Experts	Specialists	Individuals	Groups
User Status	Subservience	Dependence	Independence	Freedom
Data	Alphanumeric	Text	Fonts, Graphs	Script, Voice
Objective	Calculate	Access	Present	Communicate
User Activity	Punch & try (submit)	Remember & type (interact)	See & point (drive)	Ask & tell (delegate)
Operation	Process	Edit	Layout	Orchestrate
Interconnect	Peripherals	Terminals	Desktops	Palmtops
Applications	Custom	Standard	Generic	Components
Languages	Cobol, Fortran	PL/I, Basic	Pascal, C	Object Oriented

Os quatro paradigmas da computação e as redes

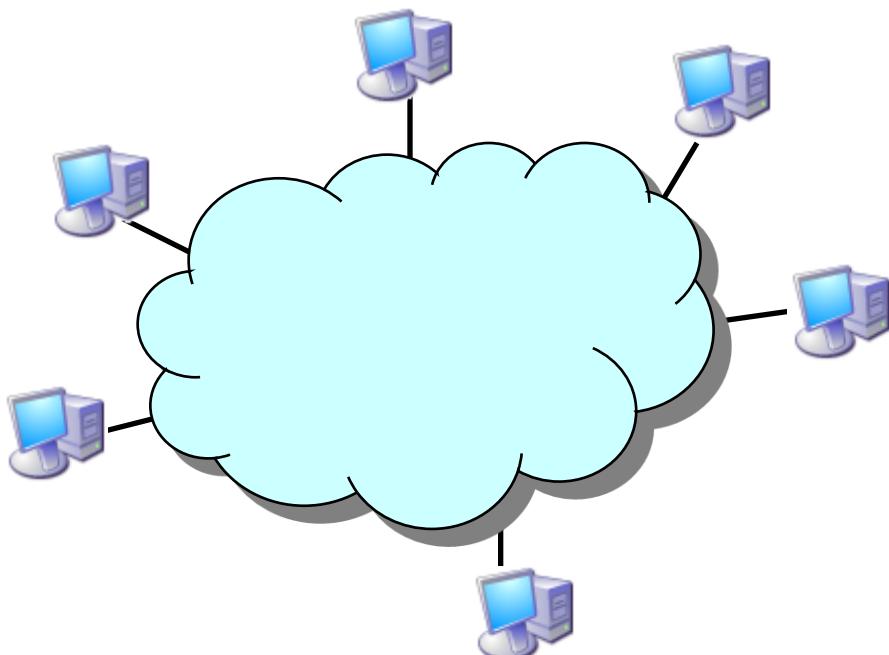
O surgimento da Internet

- Acontece no dia 1/9/1969, com a interligação dos primeiros quatro “roteadores” instalados na UCLA, SRI, UC Santa Barbara e University of Utah (EUA)
 - Surge com o nome de ARPANet
- As redes passam a ter um papel cada vez maior nos diferentes paradigmas computacionais

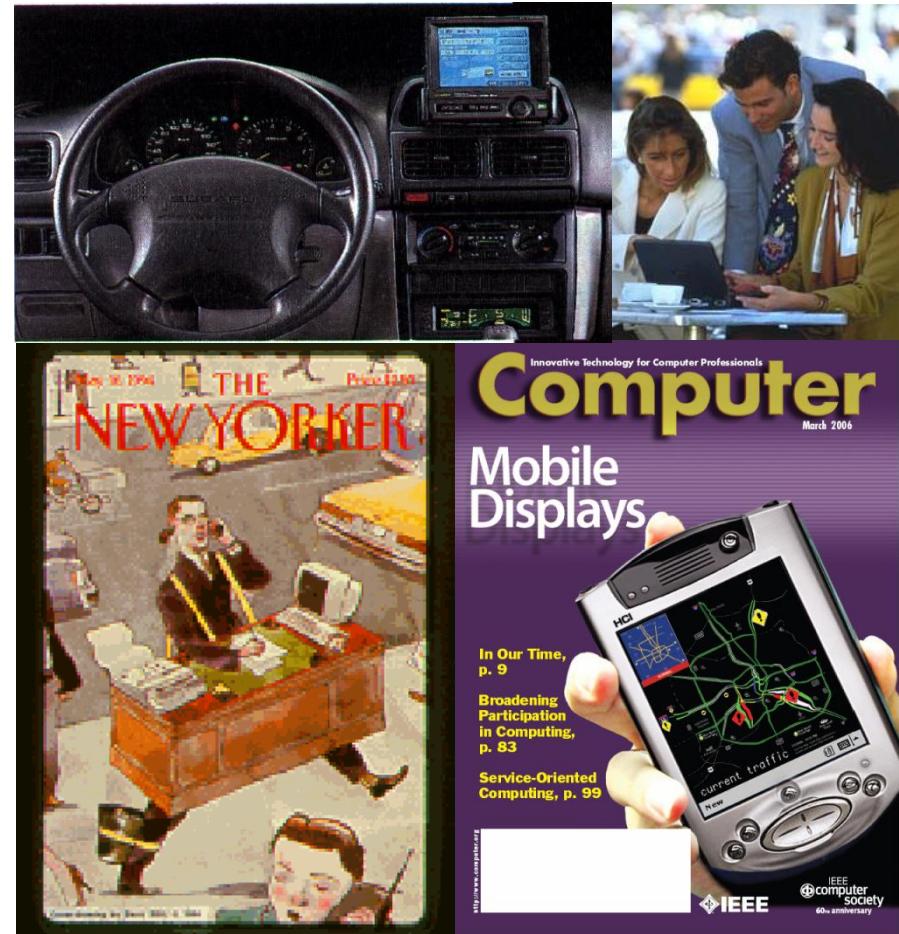


Quais paradigmas vão predominar?

Computação em rede



Computação móvel/ubíqua



What will the Internet be like 20 years from now?

Visão do Vinton Cerf
apresentada na
TIME, 18 de junho de 2000

What will the Internet be like 20 years from now?

Like the rest of infrastructure, the Internet will eventually seem to disappear by becoming ubiquitous. Most access will probably be via high-speed, low-power radio links. Most handheld, fixed and mobile appliances will be Internet enabled. This trend is already discernible in the form of Internet-enabled mobile telephones and personal digital assistants. Like the servants of centuries past, our household helpers will chatter with one another and with the outside help.

... Internet se tornará ubíqua ...

... Os acessos serão feitos através de enlaces de alta velocidade e baixa potência ...

... Diferentes dispositivos computacionais terão Internet embutida e conversarão entre si...

At some point, the armada of devices we strap to our bodies like tools on Batman's belt will coalesce into a smaller number of multifunction devices. Equipped with radio links, a PDA can serve as an appliance-control remote, a digital wallet, a cell phone, an identity badge, an e-mail station, a digital book, a pager and perhaps even a digital camera. There is sure to be a catchy name for this all-purpose Internet-enabled thingy, perhaps Wireless Internet Digital Gadget for Electronic Transactions, or WIDGET.

- ... *Teremos dispositivos computacionais com múltiplas funções ...*
- ... *Um nome “interessante” para esse dispositivo seria WIDGET ...*

So many appliances, vehicles and buildings will be online by 2020 that it seems likely there will be more things on the Internet than people. Internet-enabled cars and airplanes are coming online, and smart houses are being built every day. Eventually, programmable devices will become so cheap that we will embed them in the cardboard boxes into which we put other things for storage or shipping. These passive “computers” will be activated as they

Vinton Cerf (and not Al Gore) co-invented the Internet protocol called TCP/IP. He is a senior vice president at MCI WorldCom

- ... *Em 2020, possivelmente teremos mais “coisas” online (“aparelhos”, veículos, prédios, etc) que pessoas ...*
- ... *Dispositivos programáveis serão tão baratos que serão embutidos em caixas, etc...*

Computação Ubíqua

Computação ubíqua

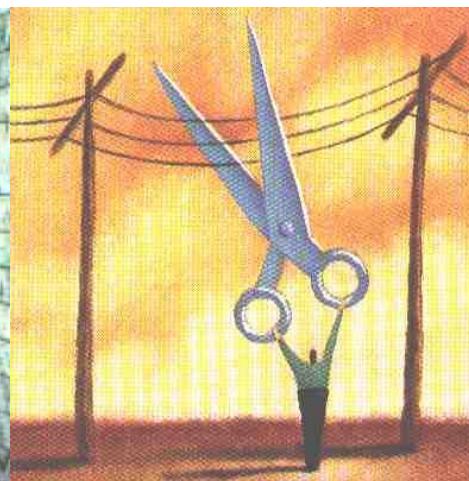
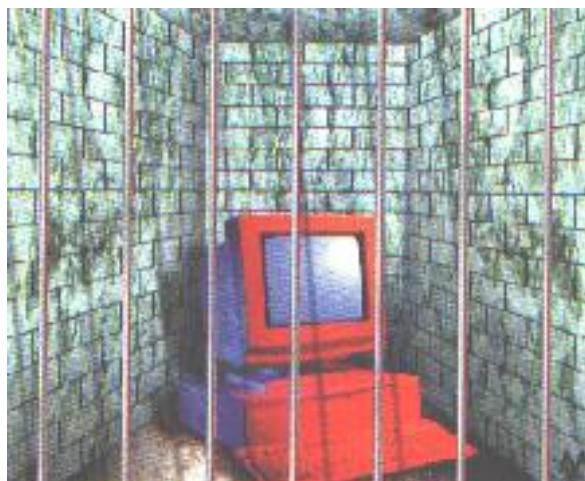
Processamento

+

Mobilidade

+

Comunicação sem fio



O problema básico de redes de computadores

- Comunicação (troca de dados, informação) entre entidades comunicantes
- ➡ Problema existente em cenários onde há necessidade de trocar dados

Classificação dos Protocolos

Classificação dos protocolos

- Quanto à conexão
- Quanto à confirmação

Conexão

- Sejam A e B duas entidades-pares em computadores distintos
 - Definição:
 - ◆ Comunicação lógica entre duas entidades A e B que permite a essas entidades trocarem dados entre si
- Entidades concordam em trocar dados entre si

Conexão

- Protocolos podem ser projetados para:
 - ◆ Estabelecer uma conexão a priori, antes de transferir dados
 - ▶ Protocolos orientados à conexão
 - ▶ Exemplo: Protocolo TCP
 - ◆ Não estabelecer uma conexão a priori, antes de transferir dados
 - ▶ Protocolo sem conexão
 - ▶ Protocolos Ethernet, IP, UDP

Conexão

- Protocolos orientados à conexão possuem três fases:
 1. Estabelecimento da conexão
 - ▶ Entidades concordam ou não em transferir dados
 2. Transferência de dados
 - ▶ Entidades transferem dados
 3. Término da conexão
 - ▶ Entidades terminam a conexão

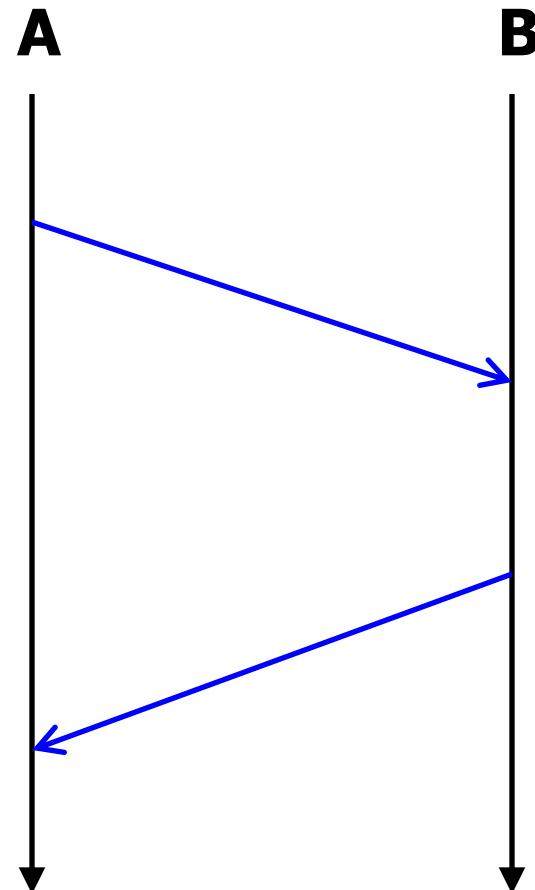
Conexão

- Protocolos que não são orientados à conexão possuem apenas uma fase:
 - ◆ Transferência de dados

Confirmação

- Definição:

- ◆ Msg que B envia para A para indicar o recebimento correto (ACK) ou não (NAK) de uma msg que A enviou para B anteriormente



Confirmação

- Protocolos podem ser projetados para:
 - ◆ Enviar uma msg de confirmação
 - ▶ Protocolo confirmado
 - ▶ Exemplo: Protocolo TCP
 - ◆ Não enviar uma msg de confirmação
 - ▶ Protocolo não confirmado
 - ▶ Protocolos Ethernet, IP, UDP

Conexão e Confirmação

■ Conexão:

- ◆ Refere-se a uma fase temporal onde msgs são trocadas entre as entidades
- ◆ O protocolo só passa para a próxima fase se a atual for concluída com sucesso

■ Confirmação:

- ◆ Diz respeito às msgs

Conexão e Confirmação

- Em cada fase da comunicação entre A e B, msgs podem ser confirmadas ou não
- Estabelecimento da conexão:
 - ◆ Msgs são sempre confirmadas
 - ◆ Por que?
- Transferência de dados e Término da conexão:
 - ◆ Podem ser confirmadas ou não

Projeto de Protocolos

Comentários iniciais

- Projetar protocolos que não sejam orientados à conexão e sem confirmação é mais simples
 - ◆ Por que?
- Projetar protocolos confirmados exige um cuidado bem maior

Exemplo: Problema dos dois exércitos

Na Grécia antiga, lugares maravilhosos como este ...



Vale perto de Almfiklia, Grécia

... podiam se transformar em cenários de guerra.



→ É quando algum filósofo propõe o “Problema dos dois exércitos”.

Problema dos dois exércitos

Cenário inicial



- Exército **Alfa** está em maior número que o exército **Gama** mas está dividido em duas metades, cada uma numa lateral do vale.
- Cada metade do exército **Alfa** está em menor número que o exército **Gama**.
- Objetivo do exército **Alfa**: coordenar um ataque ao exército **Gama** para ganhar a guerra.

Problema dos dois exércitos

O problema da coordenação

Exército Alfa

Lateral do vale



Exército Gama

Centro do vale



Exército Alfa

Lateral do vale



Vale perto de Almílkia, Grécia



1. General do exército **Alfa**, do lado esquerdo do vale, chama o seu melhor soldado para levar uma mensagem para o general do exército **Alfa** do lado direito:

 Vamos atacar conjuntamente o exército **Gama** amanhã às 6:00h?

- Observações:
- A **única** possibilidade de comunicação entre os dois generais é através de um mensageiro.
 - Os dois generais têm um “relógio perfeitamente sincronizado”, ou seja, **celes sabem pela posição do sol quando é 6:00h**.

Problema dos dois exércitos

O problema da coordenação

Exército Alfa

Lateral do vale



Exército Gama

Centro do vale



Exército Alfa

Lateral do vale



Vale perto de Almfiklia, Grécia



2. O soldado do exército **Alfa** atravessa as linhas inimigas e leva a mensagem até o general do outro lado.

Problema dos dois exércitos

O problema da coordenação

Exército Alfa

Lateral do vale



Exército Gama

Centro do vale



Exército Alfa

Lateral do vale



Vale perto de Almfiklia, Grécia



3. O general do exército **Alfa** do lado direito concorda em atacar o exército **Gama** no dia seguinte às 6:00h.

Problema dos dois exércitos

O problema da coordenação

Exército Alfa

Lateral do vale



Exército Gama

Centro do vale



Exército Alfa

Lateral do vale



Vale perto de Almfiklia, Grécia



4. O soldado do exército **Alfa** atravessa novamente as linhas inimigas e confirma com seu general o ataque para o dia seguinte.

Problema dos dois exércitos

O problema da coordenação

Exército Alfa

Lateral do vale



Exército Gama

Centro do vale



Exército Alfa

Lateral do vale



Vale perto de Almíklia, Grécia



→ Após esses quatro passos terem sido realizados com sucesso, vai haver ataque amanhã às 6:00h?

Projeto de protocolos confirmados

- Quais são os mecanismos necessários para projetar protocolos confirmados?
 - ◆ Msg de confirmação positiva (ACK) ou negativa (NAK)
- A confirmação é uma realimentação que a entidade A tem para saber se a msg foi entregue em B com sucesso ou não
 - ◆ O conhecimento no mundo distribuído dificilmente é homogêneo

Projeto de protocolos confirmados

- Sejam A e B duas entidades-pares em computadores distintos
- Cenário 1:
 - ◆ A envia $\langle \text{msg} \rangle$ para B
 - ◆ B envia $\langle \text{ack} \rangle$ para A
- O problema de projetar protocolos confirmados está resolvido?
 - ◆ Não, por que?

Projeto de protocolos confirmados

- Cenário 1':
 - ◆ A envia ⟨msg⟩ para B
 - ◆ B envia ⟨ack⟩ para A mas a msg de ⟨ack⟩ é perdida
(existem outros cenários similares)
 - ➔ A fica esperando por uma confirmação que não irá chegar
- Solução:
 - ◆ Temporizadores e retransmissão de msgs

Temporizadores

- Marca o tempo limite para que um evento ocorra
 - ◆ Exemplo: a chegada de uma msg
- Normalmente é implementado através de um relógio em hardware
 - ◆ Deve-se observar a precisão desse relógio

Temporizadores e retransmissão

■ Como funciona?

- ◆ Ao ocorrer uma ação, que implica em um evento, um temporizador é “disparado” (inicializado)
- ◆ Se o evento esperado não acontece dentro desse período de tempo, há uma temporização
- ◆ Neste caso, há uma retransmissão da msg
- ◆ O processo é repetido um número finito de vezes

Projeto de protocolos confirmados

- Cenário 2:
 - ◆ A envia $\langle \text{msg} \rangle$ para B e dispara um temporizador
 - ◆ B recebe $\langle \text{msg} \rangle$ e envia $\langle \text{ack} \rangle$ para A que é perdida
 - ◆ A temporiza, torna a enviar $\langle \text{msg} \rangle$, e dispara temporizador
 - ◆ B torna a enviar $\langle \text{ack} \rangle$ para A
 - ◆ Se A recebe $\langle \text{ack} \rangle$ então fim; caso contrário o processo é repetido um número finito de vezes
- O problema de projetar protocolos confirmados está resolvido?
 - ◆ Não, por que?

Projeto de protocolos confirmados

- 1º Problema:

- ◆ B deve distinguir o recebimento de duas msgs idênticas

Projeto de protocolos confirmados

■ 2º Problema (Cenário 2'):

- ◆ A envia $\langle \text{msg} \rangle$ para B e dispara um temporizador
- ◆ B recebe $\langle \text{msg} \rangle$ e envia $\langle \text{ack} \rangle$ para A, mas msg $\langle \text{ack} \rangle$ fica “presa” em algum ponto da rede
- ◆ A temporiza, torna a enviar $\langle \text{msg} \rangle$, e dispara temporizador
- ◆ B torna a enviar $\langle \text{ack} \rangle$ para A, que é recebida
- ◆ A envia $\langle \text{msg}^* \rangle$ para B, dispara um temporizador, mas $\langle \text{msg}^* \rangle$ não chega em B
- ◆ A msg $\langle \text{ack} \rangle$ que estava “perdida” na rede chega em A
- ➔ A entende que $\langle \text{msg}^* \rangle$ foi entregue com sucesso

Projeto de protocolos confirmados

- Solução:
 - ◆ Msgs devem ter identificadores
 - ◆ Identificadores são números inteiros e finitos
- O problema de projetar protocolos confirmados está resolvido?
 - ◆ Não, por que?

Projeto de protocolos confirmados

- Identificadores são repetidos e o problema anterior pode voltar a acontecer
- Solução:
 - ◆ Incorporar uma “marca de tempo” (*timestamp*) à msg
 - ◆ $\langle \text{acks} \rangle$ têm validade
- O problema de projetar protocolos confirmados está resolvido?

Projeto de protocolos

- Possui cinco partes:
 1. Serviço
 2. Ambiente onde será executado
 3. Vocabulário (msgs usadas)
 4. Codificação das msgs
 5. Regras de procedimento

Projeto de protocolos

- Regras de procedimento são chamadas também de protocolo (abuso de linguagem)
- São modeladas de diversas formas
 - ◆ TDFs (Técnicas de Descrição Formal) como Lotos, Estelle, SDL, etc.
 - ◆ Linguagens para descrição de sistemas distribuídos, de programação, etc.
 - ◆ CEFSM (Communicating Extended Finite State Machine)
 - ◆ ...

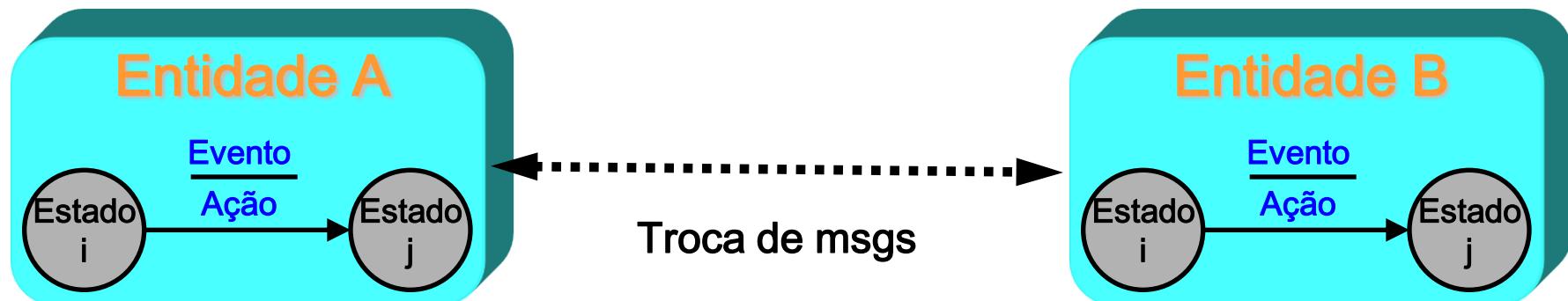
Projeto de protocolos usando CEFSM

Communicating Extended Finite State Machine
(Máquina de Estados Finitos Estendida e Comunicante)

MEF Estendida (transições podem ter predicados, atribuições a variáveis, etc) e Comunicante (cada entidade-par é modelada por uma MEF estendida que “conversa” entre si)



Projeto de protocolos usando CEFSM



Projeto de protocolos

Exemplos de eventos e ações

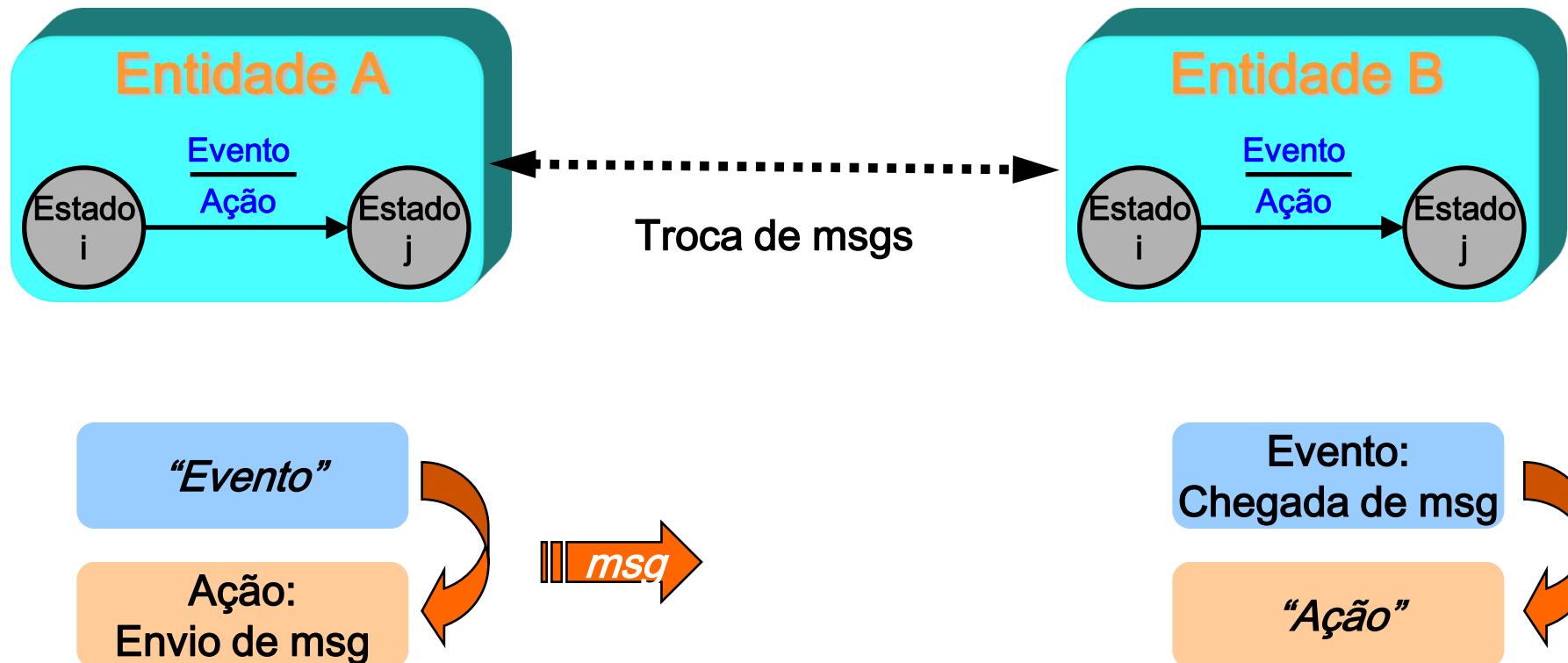
■ Eventos:

- ◆ Solicitação de serviço
- ◆ Condição que passou a ser verdadeira
- ◆ Ocorrência de uma temporização
- ◆ Recebimento de uma msg

■ Ações:

- ◆ Notificação de um serviço
- ◆ Atribuição de valores de variáveis de uma condição
- ◆ Inicializa uma temporização
- ◆ Envia uma msg

Projeto de protocolos usando CEFSM



Exercício: Máquina de refrigerante

Projete o “protocolo de comunicação” entre um usuário e o módulo de controle de uma máquina de refrigerante usando uma CEFSM

- ◆ Você deve projetar um autômato para:
 - ▶ Máquina de refrigerante
- ◆ Suposições iniciais:
 - ▶ Máquina de refrigerante só possui o refrigerante *R*
 - ▶ Refrigerante custa R\$1,00 e a máquina aceita apenas a nota de R\$1,00 e moedas de R\$0,50 e R\$0,25
- ◆ Identifique e trate outras suposições que devem ser feitas

Exercício: Máquina de refrigerante

Suposições:

1. A máquina de refrigerante possui um “hardware” para fazer a validação do dinheiro:
 - ◆ Módulo separado que só notifica a parte de controle da máquina caso o dinheiro recebido seja válido.
 - ◆ Observe que, neste caso, válido pode significar ser um valor definido pela especificação (nota de R\$1,00 e moedas de R\$0,50 e R0,25);
 - ◆ Pode significar ainda não ser falso.
2. A máquina de refrigerante devolve o dinheiro:
 - ◆ em excesso a R\$1,00, ou
 - ◆ caso não haja refrigerante disponível na máquina.

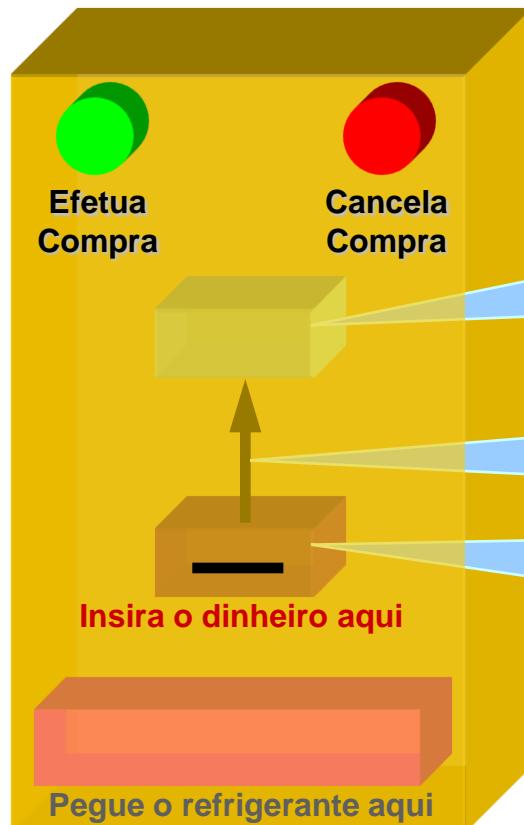
→ Nos dois casos, a parte de controle não fica “sabendo” da ocorrência disso.

Exercício: Máquina de refrigerante

Suposições:

3. A máquina de refrigerante não sofre uma falha eletromecânica:
 - ◆ A máquina sempre funciona corretamente durante uma transação.
4. O usuário pode desistir de comprar o refrigerante a qualquer momento:
 - ◆ Deve haver um botão para “Cancelar Compra” (CC), e
 - ◆ Deve haver um botão para “Efetuar Compra” (EA).
5. O usuário deve cancelar ou efetuar a compra dentro de um intervalo de tempo.

Exercício: Máquina de refrigerante



**Máquina de
Refrigerante**

Módulo de controle
da máquina (MCM)

Gera evento para MCM
sse dinheiro válido

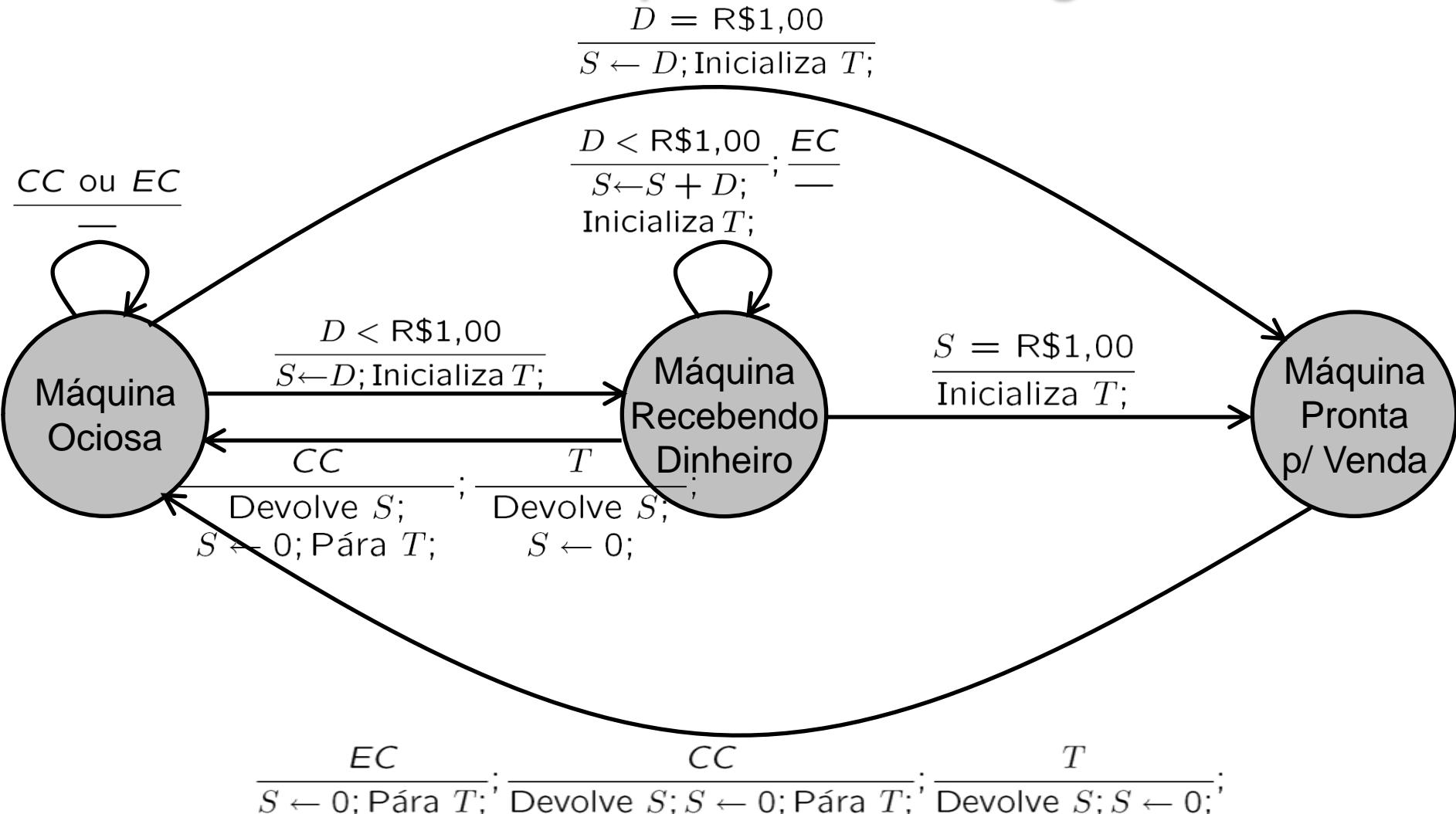
Hardware para
validar o dinheiro



Exercício: Máquina de refrigerante

- Estados:
 - ◆ Máquina ociosa
 - ◆ Máquina recebendo dinheiro
 - ◆ Máquina pronta para venda
- Variável:
 - ◆ S: indica a soma de dinheiro recebida pela máquina
- Eventos:
 - ◆ Dinheiro (D)
 - ◆ Efetua compra (EC)
 - ◆ Cancela compra (CC)
 - ◆ Temporização (T)

Exercício: Máquina de refrigerante



Algumas regras básicas de projeto

- Comece o projeto:
 - ◆ Fazendo as suposições necessárias
 - ◆ Identificando os eventos, ações, variáveis e possíveis estados
- Identifique um estado diferente da CEFSM para um estado real diferente
- Para cada estado da CEFSM, analise todos os possíveis eventos