Instituto de computação

Programa de Pós-Graduação
Instituto de Computação
Universidade Federal Fluminense

Aula 10

Camada de Aplicação

DNS e sistemas par-a-par

Igor Monteiro Moraes

Redes de Computadores

ATENÇÃO!

Instituto de computação

- Este apresentação é contém partes baseadas nos seguintes trabalhos
  - Notas de aula do Prof. José Augusto Suruagy Monteiro, disponíveis em http://www.nuperc.unifacs.br/Members/jose.suruagy/cursos
  - Material complementar do livro Computer Networking: A Top Down Approach, 5th edition, Jim Kurose and Keith Ross, Addison-Wesley, abril de 2009
  - Moraes, I. M., Campista, M. E. M., Moreira, M. D. D., Rubinstein, M. G., Costa, L. H. M. K., and Duarte, O. C. M. B. - "Distribuição de Vídeo sobre Redes Par-a-Par: Arquiteturas, Mecanismos e Desafios", in Minicursos do Simpósio Brasileiro de Redes de Computadores - SBRC'2008, pp. 115-171, Rio de Janeiro, RJ, Brazil, May 2008.

Instituto de Computação

Domain Name System (DNS)

Identificadores

Instituto de computação

- Uma pessoa → várias formas de identificação
  - Nome
  - Carteira de identidade
  - CPF
  - Carteira de motorista
  - Etc.

Identificadores

Instituto de computação

- Estações e roteadores na Internet
  - Endereço IP (ex.: 200.20.15.38)
    - Conjunto de bits
    - Tamanho fixo
    - Estrutura hierárquica
    - Pouco intuitivo para os usuários

    Bom para uma máquina
  - Nome (ex.: www.ic.uff.br)
    - Tamanho variável
    - Intuitivo para os usuários

    Bom para um humano

O que fazer?

Identificadores

Instituto de computação

- Estações e roteadores na Internet
  - Endereço IP (ex.: 200.20.15.38)
    - Conjunto de bits
    - Tamanho fixo
    - Estrutura hierárquica
    - Pouco intuitivo para os usuários

    Bom para uma máquina
  - Nome (ex.: www.ic.uff.br)
    - Tamanho variável
    - Intuitivo para os usuários

    Bom para um humano

Mapeamento

DNS (*Domain Name System*)

Instituto de computação

- Mapeamento entre nomes de domínio e endereços IP
  - Também faz o inverso: DNS reverso
- É composto por
  - Base de dados distribuída entre diferentes servidores
    - Organização hierárquica
  - Protocolo da camada de aplicação
    - Nós se comunicam para **resolver** nomes
- Mais um exemplo do princípio da Internet
  - Complexidade na borda da rede

DNS (*Domain Name System*)

Instituto de computação

- Serviços
  - **Traduz um nome para um endereço IP**
  - Permite o uso de "apelidos" para os nós (*aliasing*)
    - Servidores, estações, roteadores, etc.
    - Mapeamento de nomes canônicos e apelidos
  - Distribuição de carga
    - Conjunto de endereços IP mapeados em apenas um nome
    - Ex.: servidores Web replicados

DNS (*Domain Name System*)

Instituto de computação

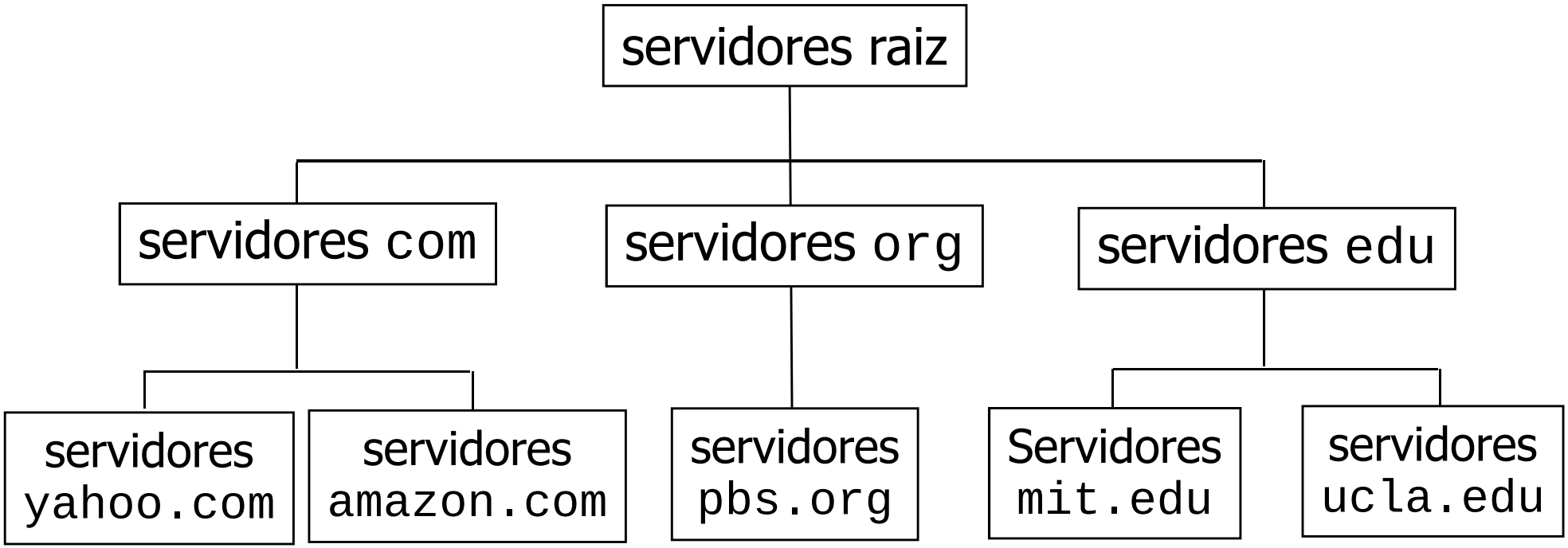
- Por quê é não é uma base de dados **centralizada**?
  - Ponto único de falha
  - Volume de tráfego
    - Requisições e respostas
  - Distância para um usuário
    - Maior tempo de resposta
  - Manutenção

Não é escalável!

DNS (*Domain Name System*)

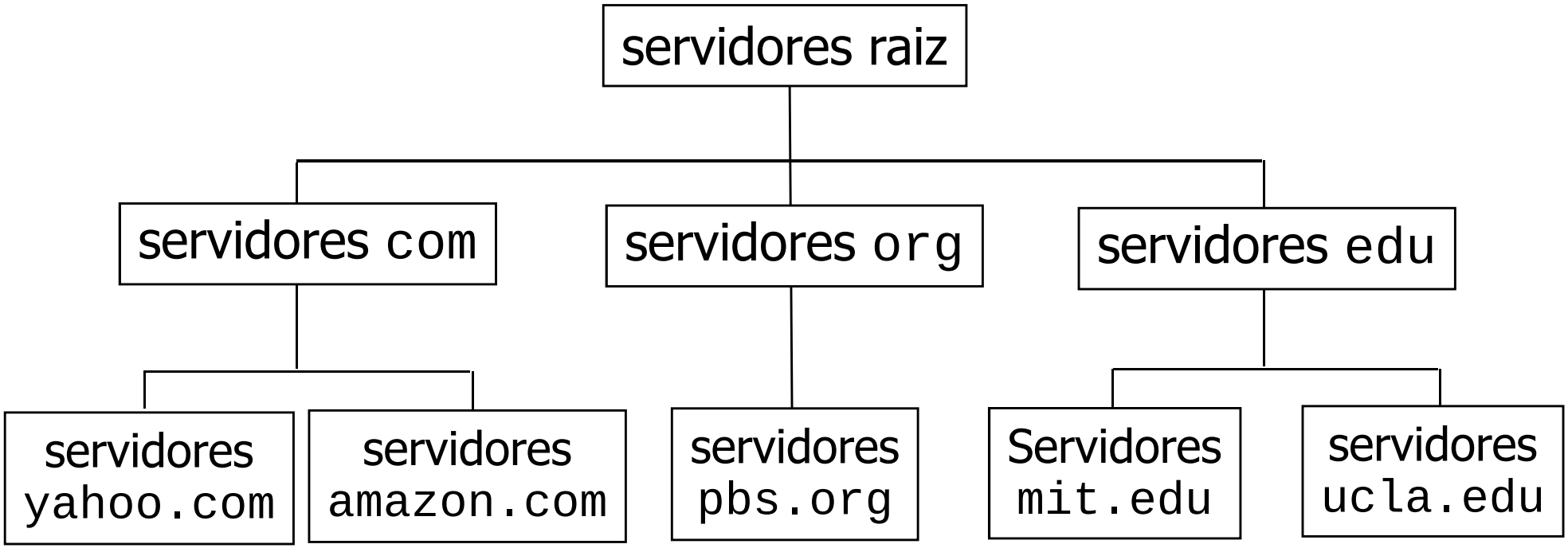
Instituto de computação

- Base de dados **distribuída** e **hierárquica**



DNS (*Domain Name System*)

- Base de dados **distribuída** e **hierárquica**

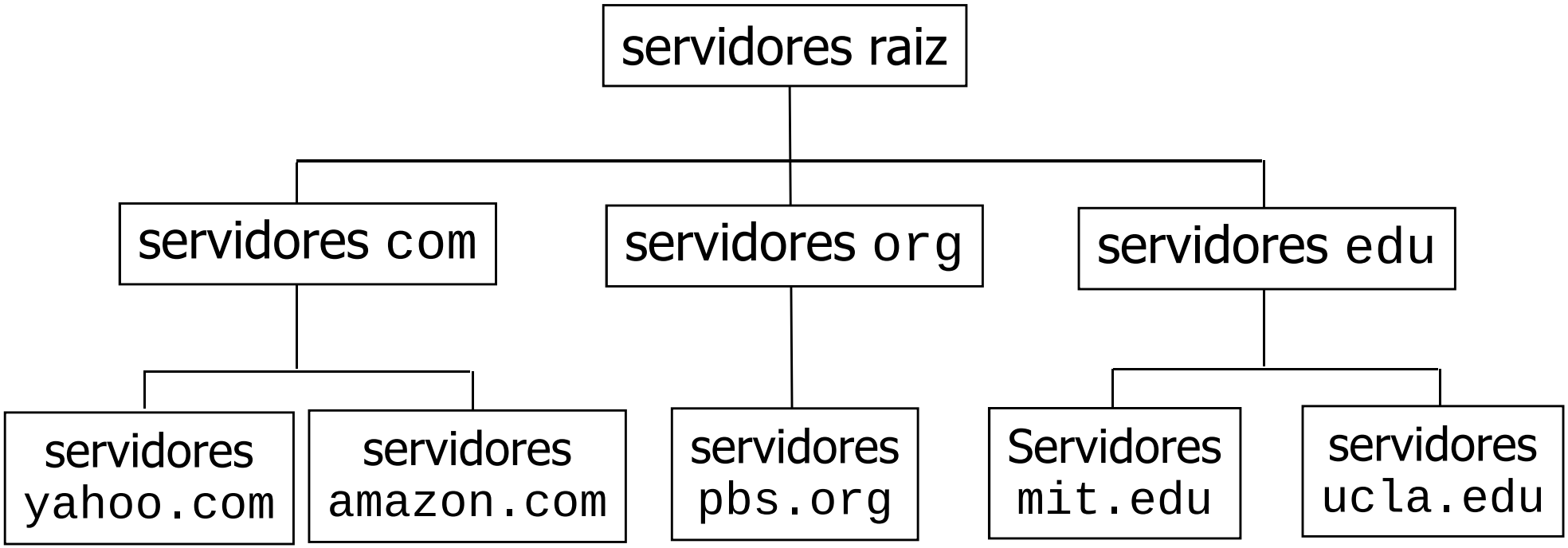


Cliente quer acessar `amazon.com`

DNS (*Domain Name System*)

Instituto de computação

- Base de dados **distribuída** e **hierárquica**

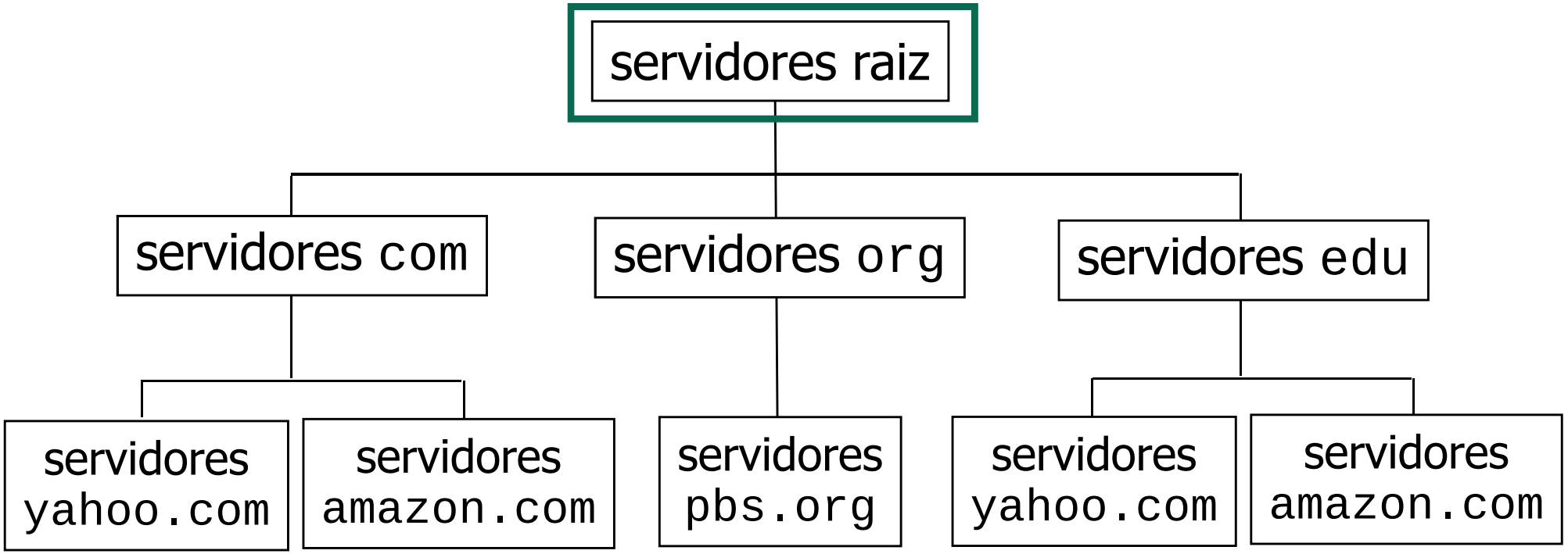


Descobrir o endereço IP de `amazon.com`

DNS (*Domain Name System*)

Instituto de computação

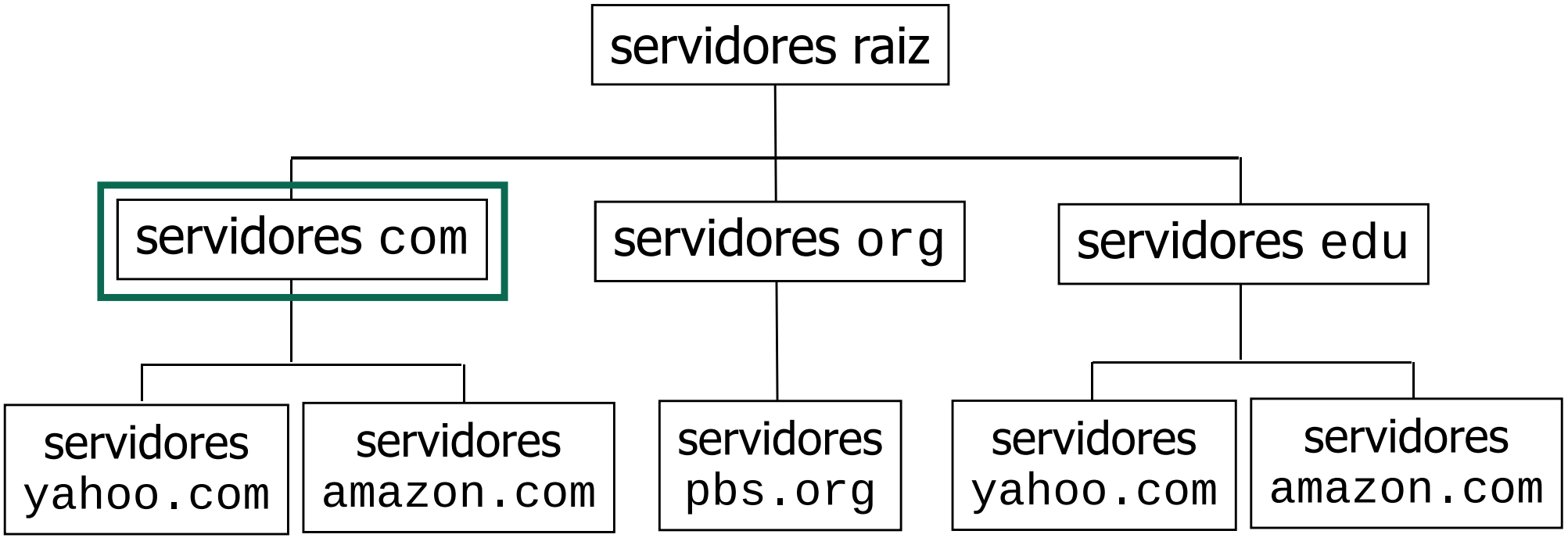
- Base de dados **distribuída** e **hierárquica**



Consulta ao servidor raiz para descobrir o servidor `.com`

DNS (*Domain Name System*)

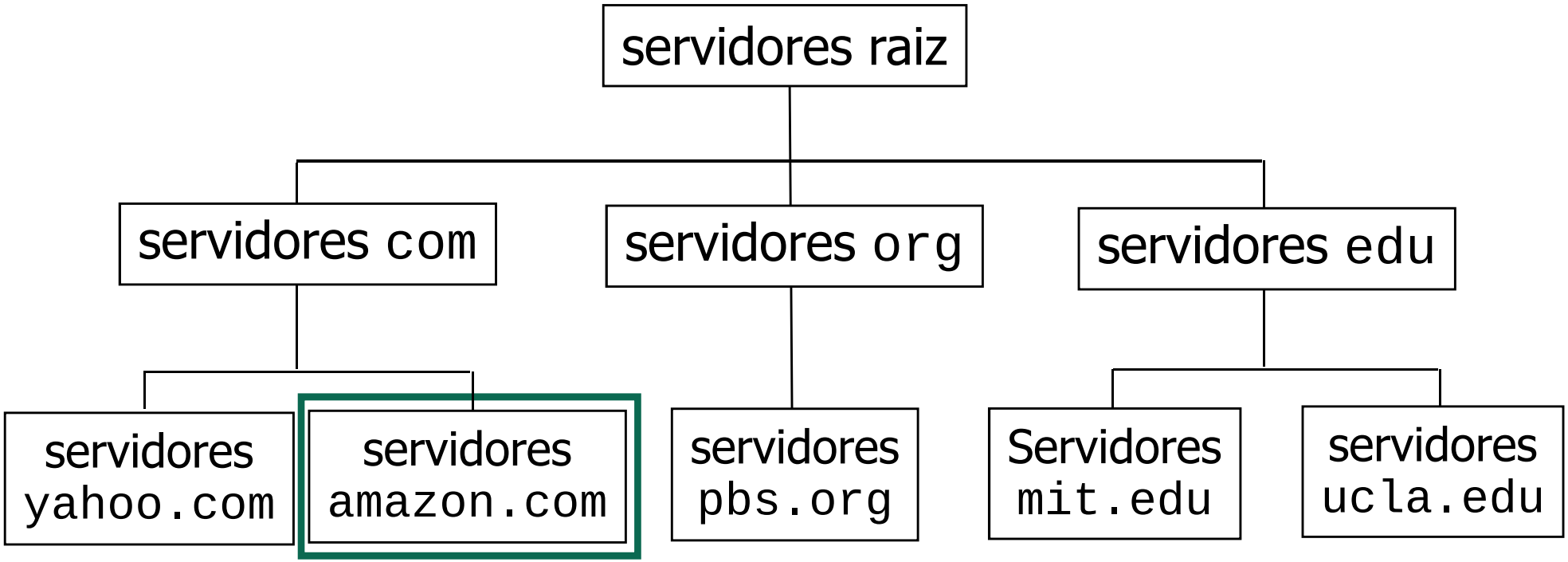
- Base de dados **distribuída** e **hierárquica**



Consulta ao servidor `.com` para descobrir o servidor `amazon.com`

DNS (*Domain Name System*)

- Base de dados **distribuída** e **hierárquica**



Cliente consulta servidor DNS do domínio `amazon.com` para obter endereço IP de `www.amazon.com`

Servidores Raiz

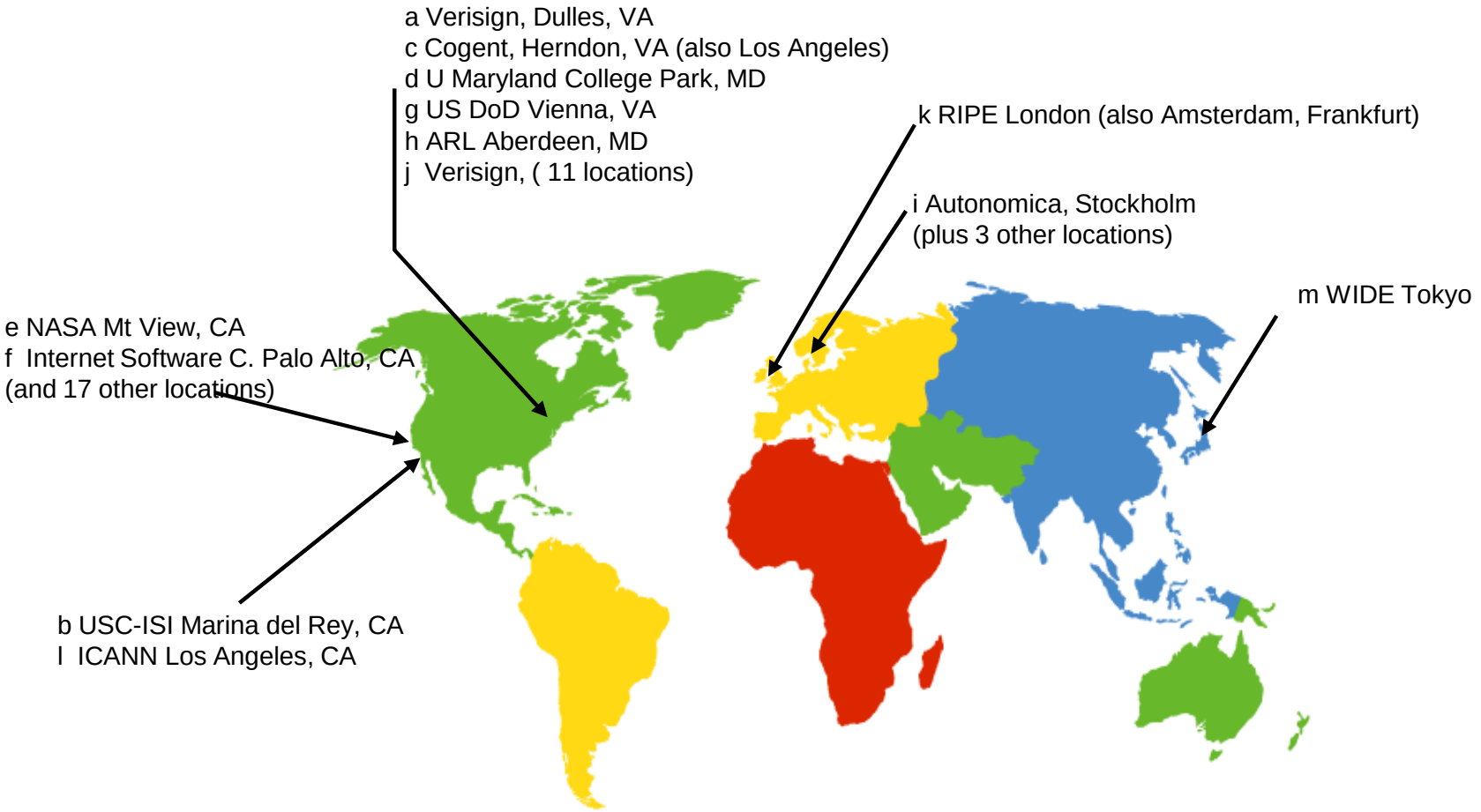
Instituto de computação

- Consultados por servidores locais que não conseguem resolver o nome

- Ao receber uma consultas
  - Procura o servidor oficial se mapeamento desconhecido
  - Obtém tradução
  - Devolve mapeamento ao servidor local

Servidores Raiz

- 13 ao redor do mundo
  - 10 somente nos EUA



Servidores Raiz

- 13 ao redor do mundo
  - Um pode ter várias réplicas espalhadas

Servidores de Domínio de Alto Nível

Instituto de computação

- Servidores TLD (*Top-level Domain*)
- Responsáveis por
  - Domínios como `com`, `org`, `net`, `edu`, etc.
  - Todos os domínios de países como `br`, `uk`, `fr`, `ca`, `jp`
- Network Solutions mantém servidores para domínio .com
  - Monopólio até 1999
- NIC.br (Registro .br) para domínio .br

Servidores Oficiais (*Authoritative*)

Instituto de Computação

- São os servidores das organizações
  - Mapeamentos **oficiais** entre nomes e endereços IP
    - Inclusive para outros servidores da organização (e.x., Web e correio).
  - Podem ser mantidos pelas organizações ou pelo provedor de acesso

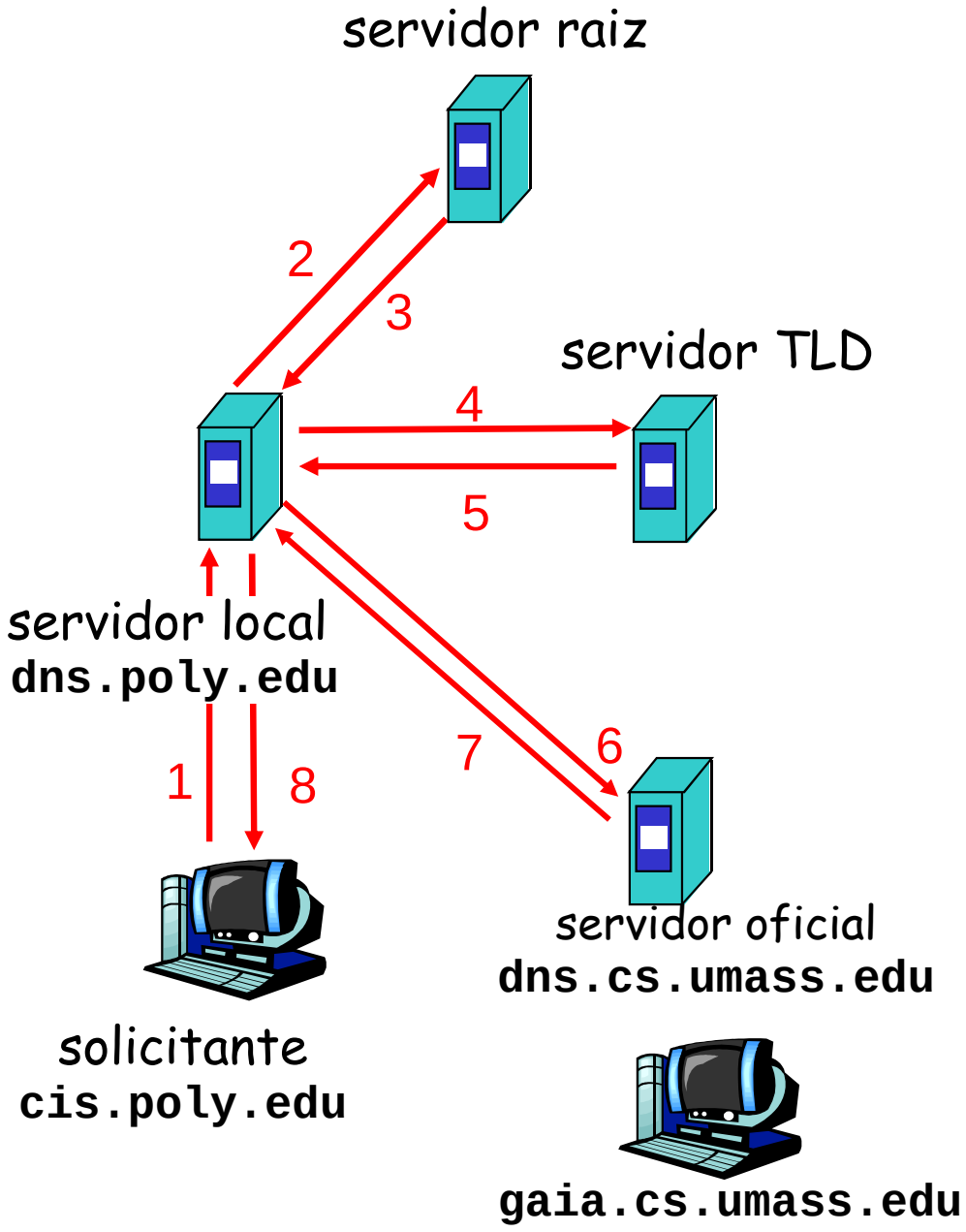
Servidor de Nomes Local

Instituto de computação

- Não pertence necessariamente à hierarquia
- Cada "provedor" possui um
  - ISP residencial, empresa, universidade, etc.
  - Também chamado de **servidor de nomes padrão**
- Quando uma estação faz uma consulta DNS
  - Ela é **primeiro** enviada para o seu **servidor local**
  - Atua como um intermediário
    - O servidor local é quem consulta os demais servidores da hierarquia

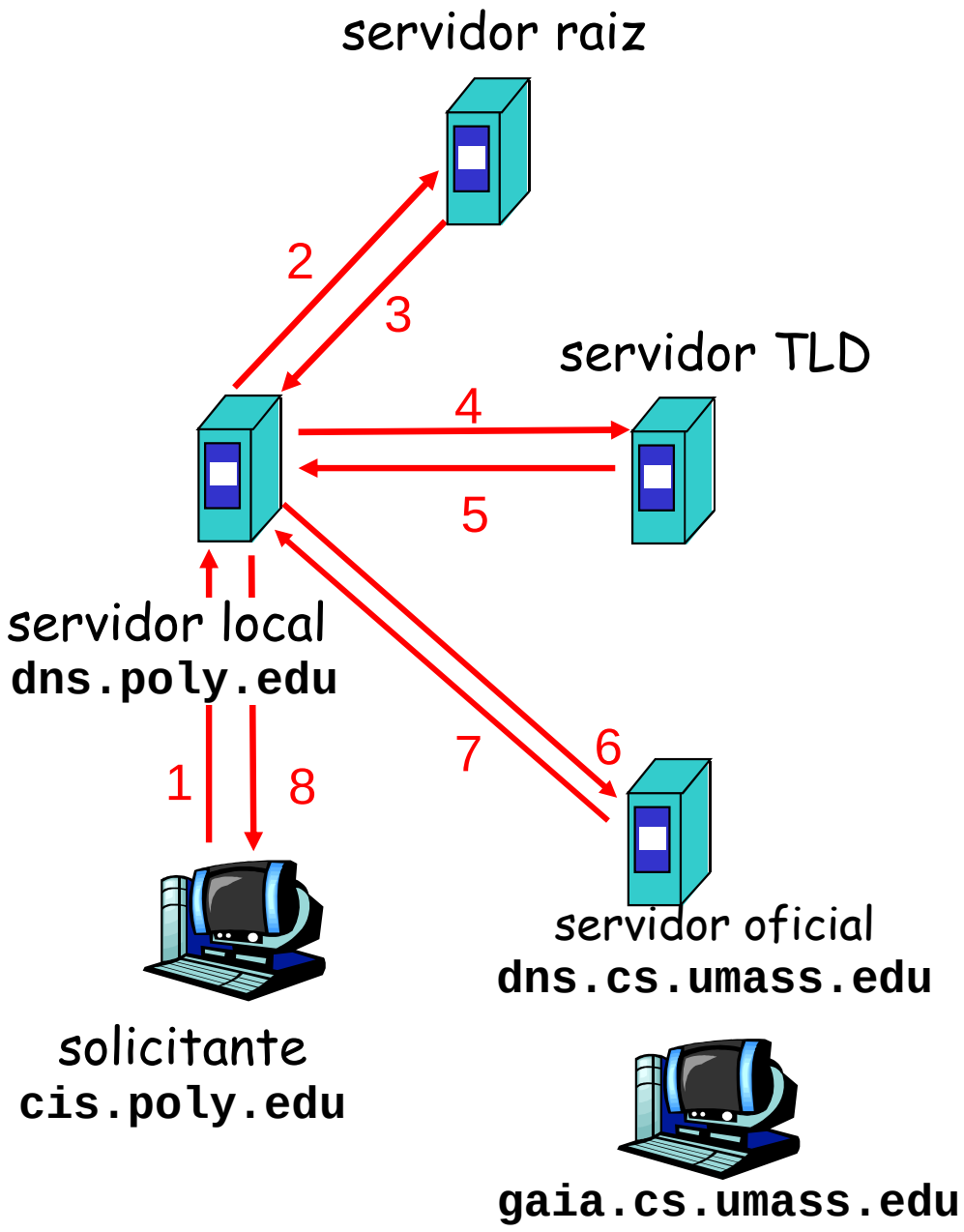
Exemplo de resolução de nome pelo DNS

- Estação em `cis.poly.edu` quer endereço IP para `gaia.cs.umass.edu`



Exemplo de resolução de nome pelo DNS

- Estação em `cis.poly.edu` quer endereço IP para `gaia.cs.umass.edu`



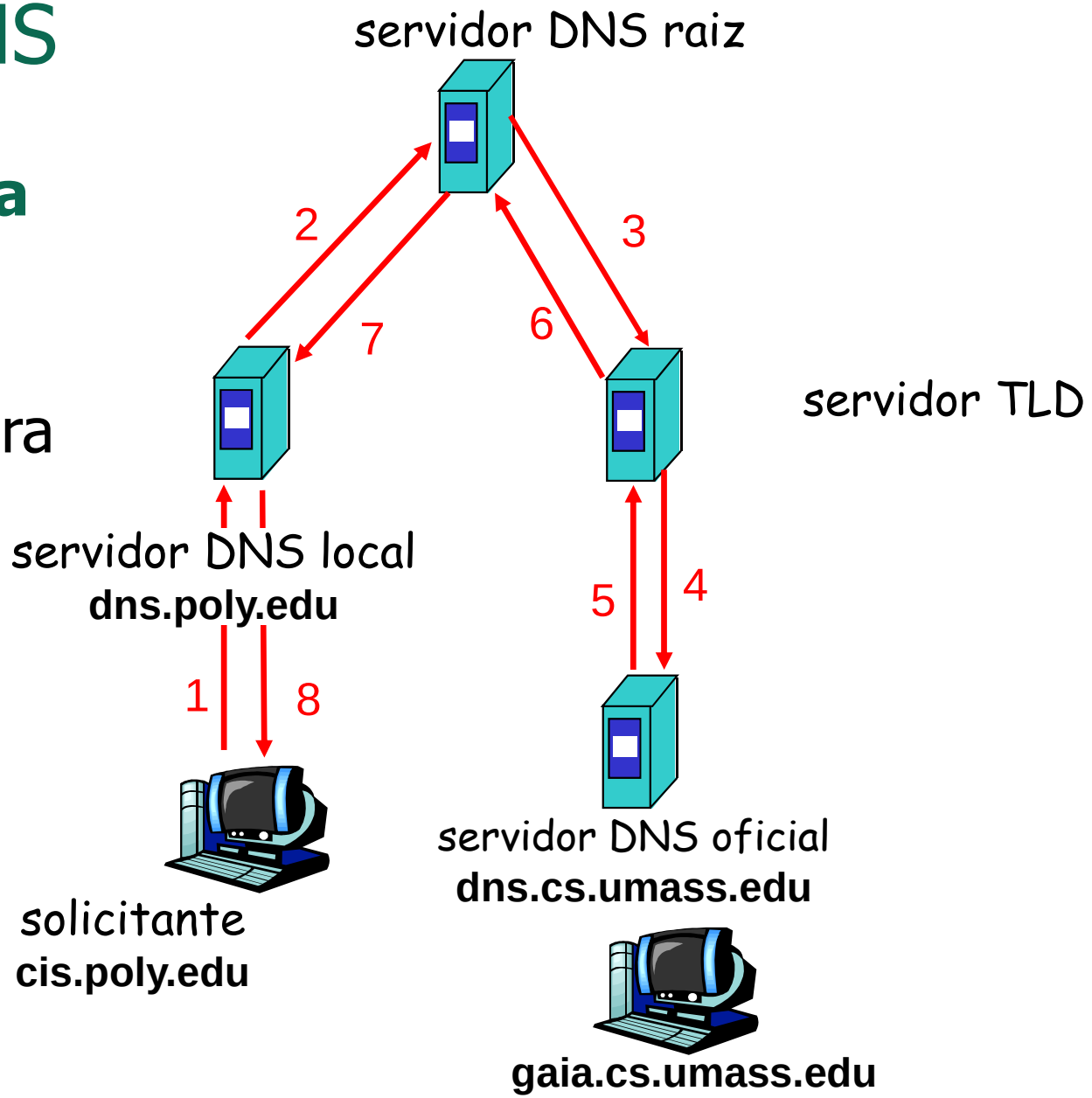
Consulta interativa

- Servidor consultado responde com o nome de um servidor de contato
  - "Não conheço este nome, mas pergunte para esse servidor"

Exemplo de resolução de nome pelo DNS

Consulta recursiva

- Transfere a responsabilidade de resolução do nome para o servidor de nomes contatado
  - Maior carga em servidores de maior altura



Uso do *Cache*

Instituto de computação

- Uma vez que um servidor qualquer aprende um mapeamento, ele o coloca em um ***cache*** local
  - Evitar a consulta a servidores de maior "altura"
  - Entradas no *cache* são sujeitas a temporização
    - Desaparecem depois de um certo tempo
    - Geralmente, 2 dias
- Servidores TLD tipicamente armazenados no *cache* dos servidores de nomes locais
- Servidores raiz acabam não sendo visitados com muita freqüência

Registros

Instituto de computação

- O DNS é uma base distribuída composta por **registros de recursos** (RR)

RR: **(nome, valor, tipo, TTL)**

- Significado de cada campo depende do tipo
  - Tipo A
    - **nome** é nome de uma estação
    - **valor** é o seu endereço IP
  - Tipo NS
    - **nome** é domínio (p.ex. `ic.uff.br`)
    - **valor** é endereço IP de servidor oficial de nomes para este domínio

Registros

Instituto de computação

- O DNS é uma base distribuída composta por **registros de recursos** (RR)

RR: **(nome, valor, tipo, TTL)**

- Significado de cada campo depende do tipo
  - Tipo CNAME
    - **nome** é o "apelido" (*alias*) para algum nome "canônico" (verdadeiro)
    - **valor** é o nome canônico
  - Tipo MX
    - **nome** é o domínio
    - **valor** é nome do servidor de correio para este domínio

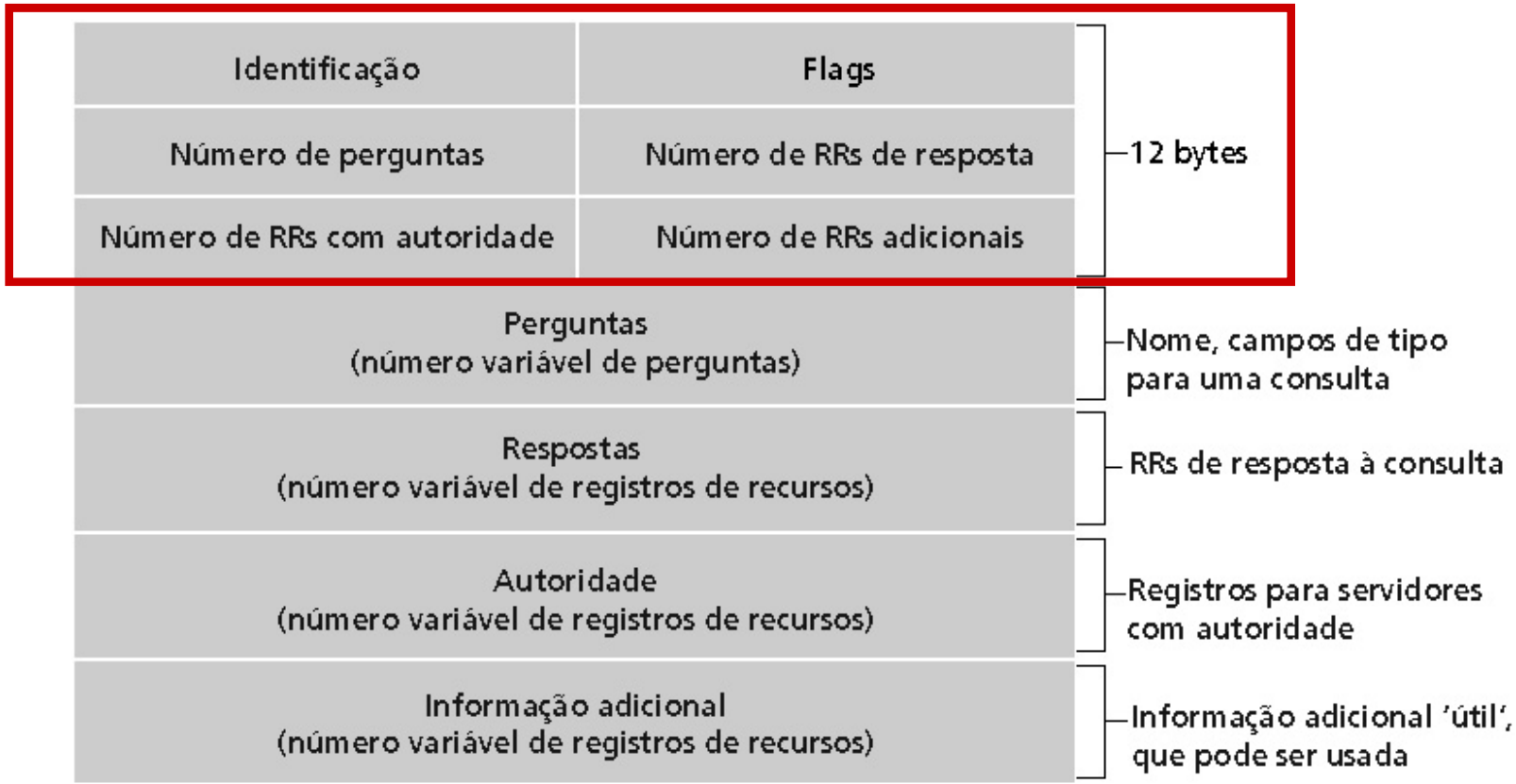
Mensagens

- O DNS é protocolo baseado em mensagens de **pedido** e **resposta**
  - As duas possuem o **mesmo formato**

```
[igor@recreio ~]#host -v www.ic.uff.br
Trying "www.ic.uff.br"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25480
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADDITIONAL: 3

;; QUESTION SECTION:
;www.ic.uff.br.                 IN      A

;; ANSWER SECTION:
www.ic.uff.br.          51390   IN      CNAME   tangerina.ic.uff.br.
tangerina.ic.uff.br.    51390   IN      A       200.20.15.38

;; AUTHORITY SECTION:
ic.uff.br.              51390   IN      NS      guanabara.rederio.br.
ic.uff.br.              51390   IN      NS      marlin.telecom.uff.br.
ic.uff.br.              51390   IN      NS      server.uff.br.
ic.uff.br.              51390   IN      NS      abacate.ic.uff.br.

;; ADDITIONAL SECTION:
marlin.telecom.uff.br.  50731   IN      A       200.20.10.17
abacate.ic.uff.br.      51390   IN      A       200.20.15.208
guanabara.rederio.br.   37310   IN      A       200.20.94.50

Received 223 bytes from 146.164.69.2#53 in 6 ms
```

Mensagens

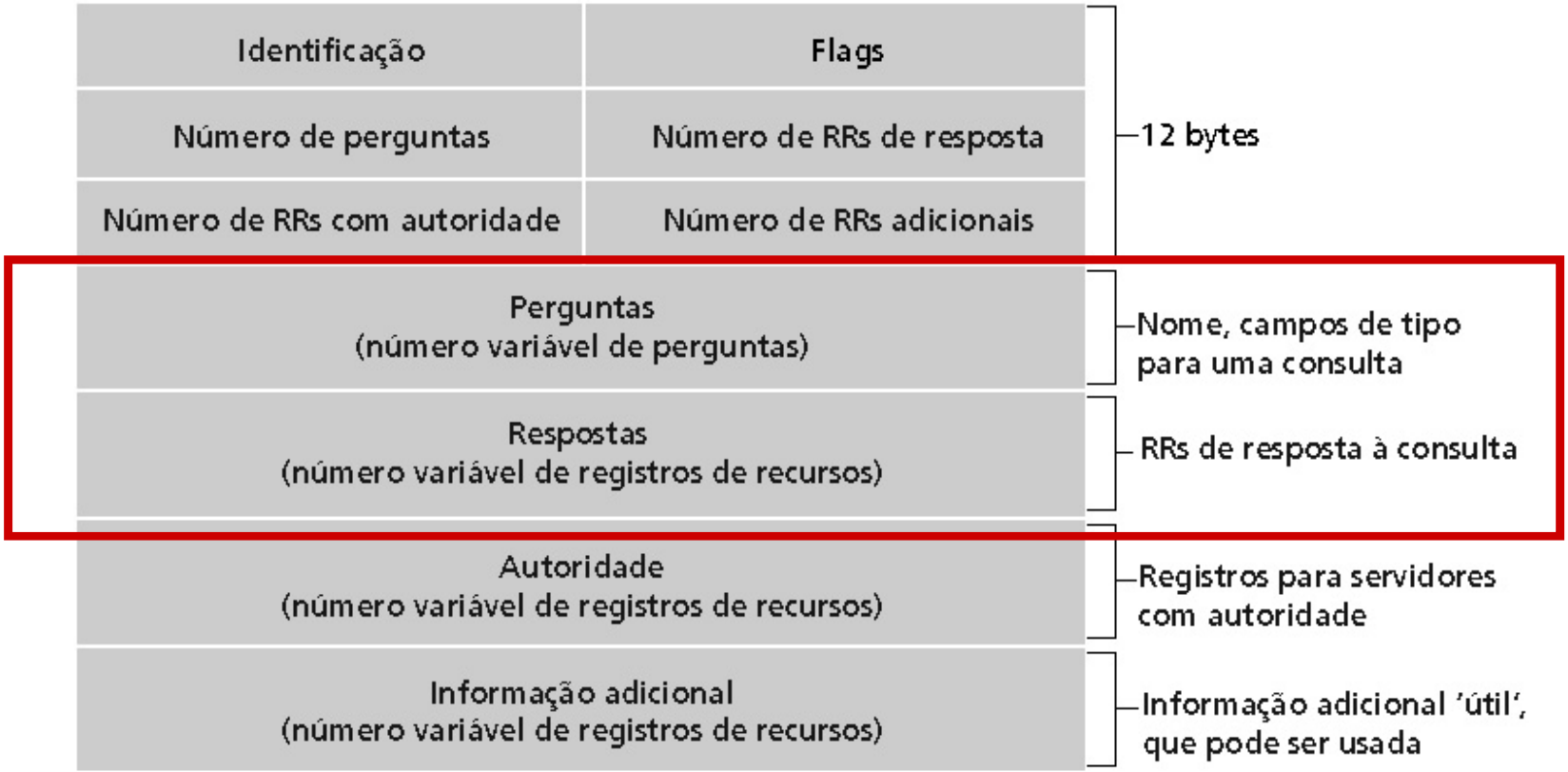
- O DNS é protocolo baseado em mensagens de **pedido** e **resposta**
  - As duas possuem o **mesmo formato**



```
[igor@recreio ~]#host -v www.ic.uff.br
Trying "www.ic.uff.br"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25480
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADDITIONAL: 3

;; QUESTION SECTION:
;www.ic.uff.br.                 IN      A

;; ANSWER SECTION:
www.ic.uff.br.          51390   IN      CNAME   tangerina.ic.uff.br.
tangerina.ic.uff.br.    51390   IN      A       200.20.15.38

;; AUTHORITY SECTION:
ic.uff.br.              51390   IN      NS      guanabara.rederio.br.
ic.uff.br.              51390   IN      NS      marlin.telecom.uff.br.
ic.uff.br.              51390   IN      NS      server.uff.br.
ic.uff.br.              51390   IN      NS      abacate.ic.uff.br.

;; ADDITIONAL SECTION:
marlin.telecom.uff.br.  50731   IN      A       200.20.10.17
abacate.ic.uff.br.      51390   IN      A       200.20.15.208
guanabara.rederio.br.   37310   IN      A       200.20.94.50

Received 223 bytes from 146.164.69.2#53 in 6 ms
```

Mensagens

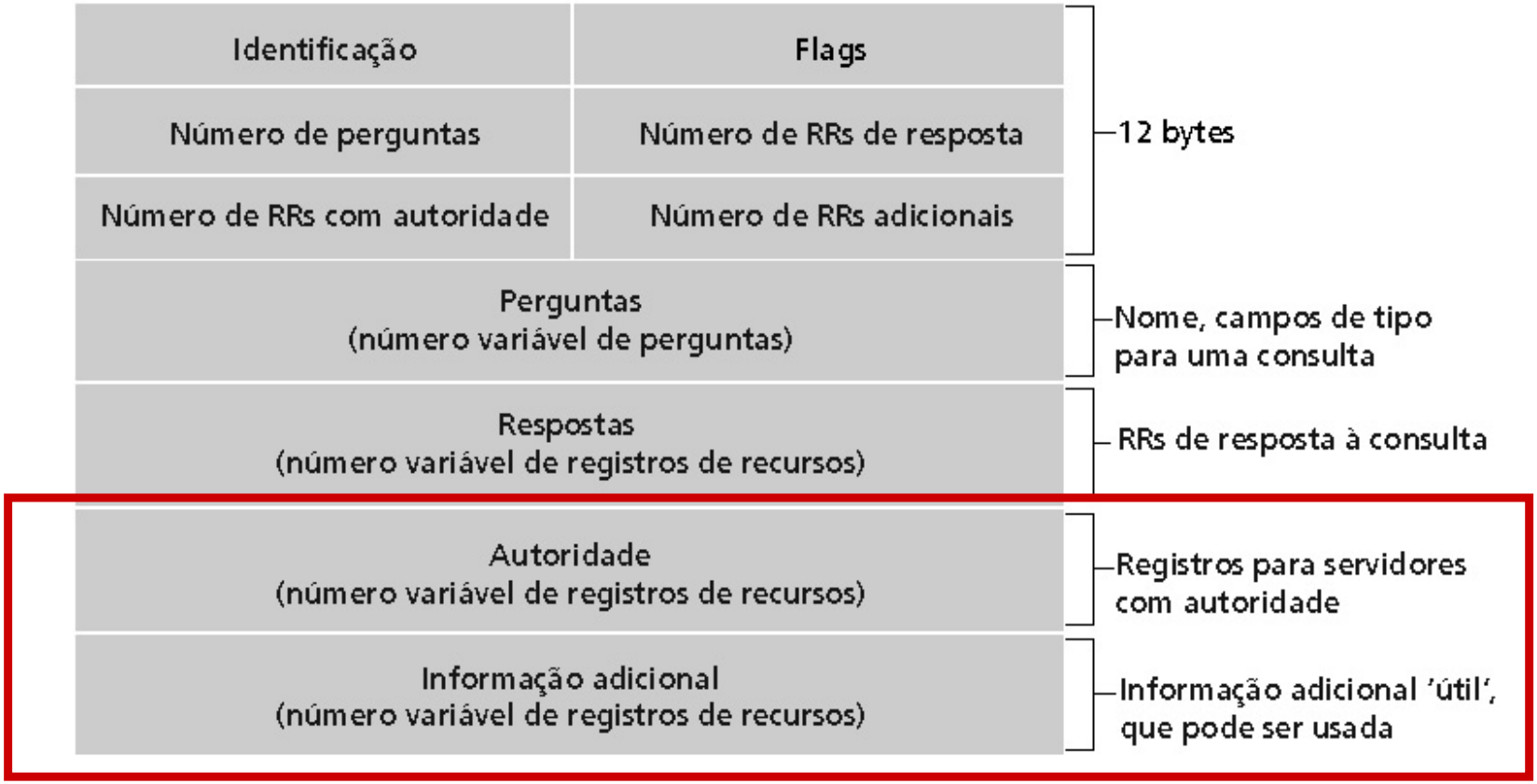
- O DNS é protocolo baseado em mensagens de **pedido** e **resposta**
  - As duas possuem o **mesmo formato**



```
[igor@recreio ~]#host -v www.ic.uff.br
Trying "www.ic.uff.br"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25480
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADDITIONAL: 3

;; QUESTION SECTION:
;www.ic.uff.br.                 IN      A

;; ANSWER SECTION:
www.ic.uff.br.          51390   IN      CNAME   tangerina.ic.uff.br.
tangerina.ic.uff.br.    51390   IN      A       200.20.15.38

;; AUTHORITY SECTION:
ic.uff.br.              51390   IN      NS      guanabara.rederio.br.
ic.uff.br.              51390   IN      NS      marlin.telecom.uff.br.
ic.uff.br.              51390   IN      NS      server.uff.br.
ic.uff.br.              51390   IN      NS      abacate.ic.uff.br.

;; ADDITIONAL SECTION:
marlin.telecom.uff.br.  50731   IN      A       200.20.10.17
abacate.ic.uff.br.      51390   IN      A       200.20.15.208
guanabara.rederio.br.   37310   IN      A       200.20.94.50

Received 223 bytes from 146.164.69.2#53 in 6 ms
```

Mensagens

- O DNS é protocolo baseado em mensagens de **pedido** e **resposta**
  - As duas possuem o **mesmo formato**



```
[igor@recreio ~]#host -v www.ic.uff.br
Trying "www.ic.uff.br"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25480
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADDITIONAL: 3

;; QUESTION SECTION:
;www.ic.uff.br.                 IN      A

;; ANSWER SECTION:
www.ic.uff.br.          51390   IN      CNAME   tangerina.ic.uff.br.
tangerina.ic.uff.br.    51390   IN      A       200.20.15.38

;; AUTHORITY SECTION:
ic.uff.br.              51390   IN      NS      guanabara.rederio.br.
ic.uff.br.              51390   IN      NS      marlin.telecom.uff.br.
ic.uff.br.              51390   IN      NS      server.uff.br.
ic.uff.br.              51390   IN      NS      abacate.ic.uff.br.

;; ADDITIONAL SECTION:
marlin.telecom.uff.br.  50731   IN      A       200.20.10.17
abacate.ic.uff.br.      51390   IN      A       200.20.15.208
guanabara.rederio.br.   37310   IN      A       200.20.94.50

Received 223 bytes from 146.164.69.2#53 in 6 ms
```

Instituto de Computação

Sistemas Par-a-Par

Modelo de Aplicações

Instituto de Computação

Rede orientada ao usuário

Rede orientada ao conteúdo

um usuário quer contatar outro usuário

acesso a terminal remoto (telnet), transferência de arquivos (FTP) e correio eletrônico (SMTP)

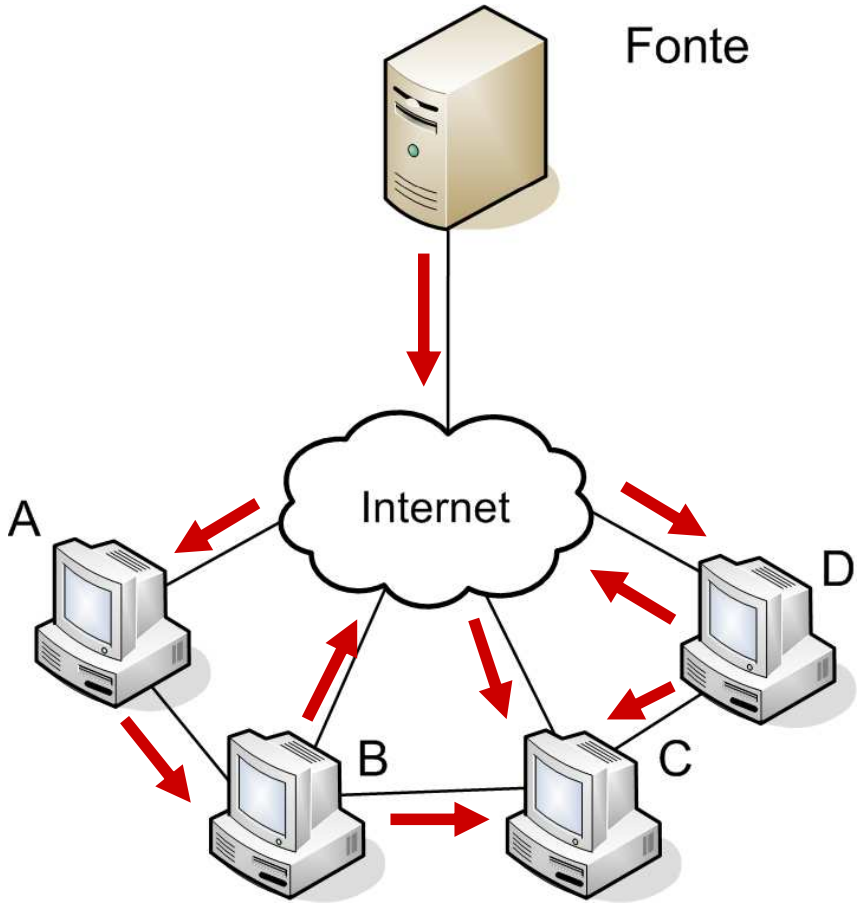
um usuário quer acessar um serviço ou dado específico

Não importa onde (em que estação) esse serviço ou dado está localizado

Sistemas par-a-par (BitTorrent), redes de distribuição de conteúdo (Akamai)

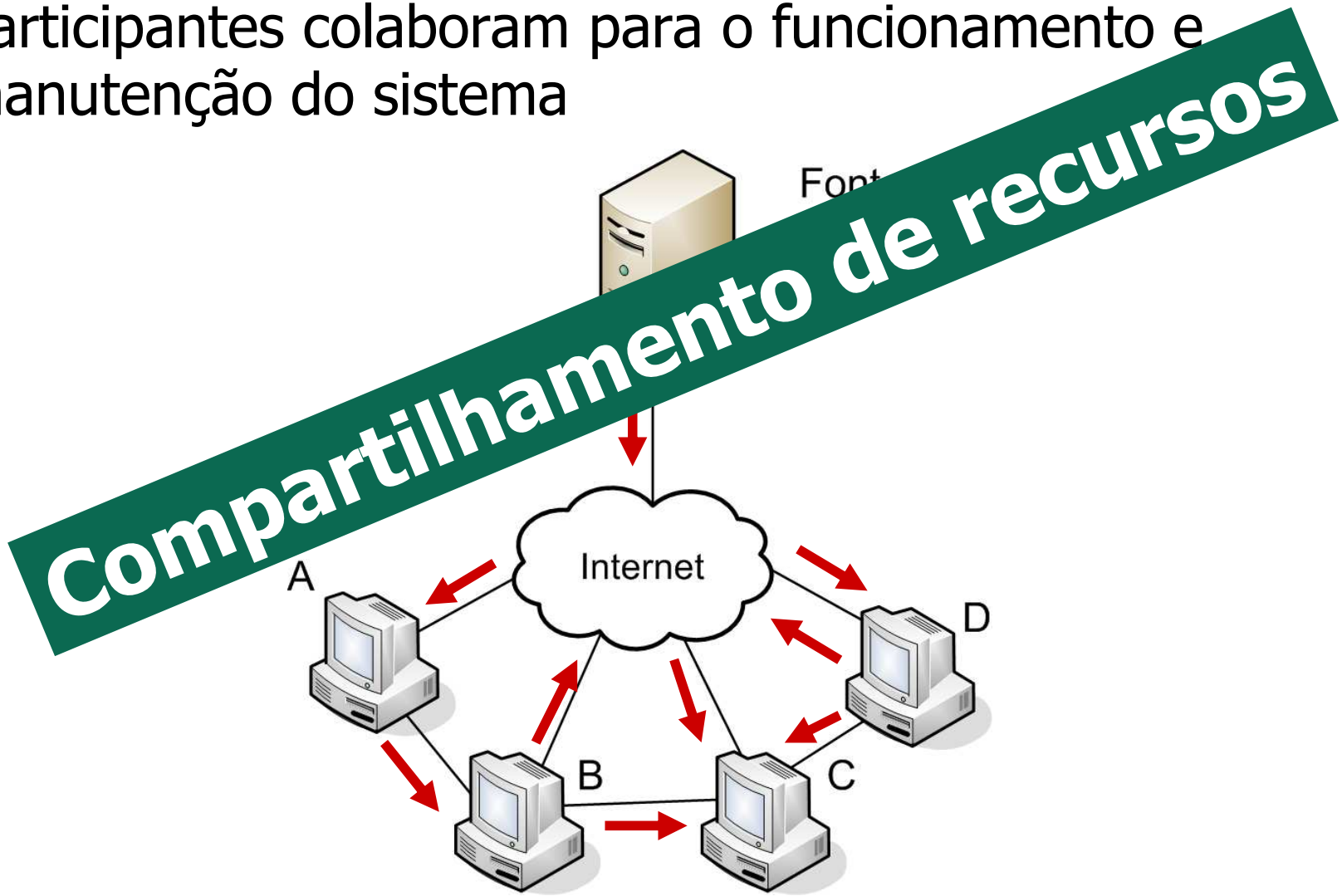
Sistemas Par-a-Par

- Participantes colaboram para o funcionamento e manutenção do sistema



Sistemas Par-a-Par

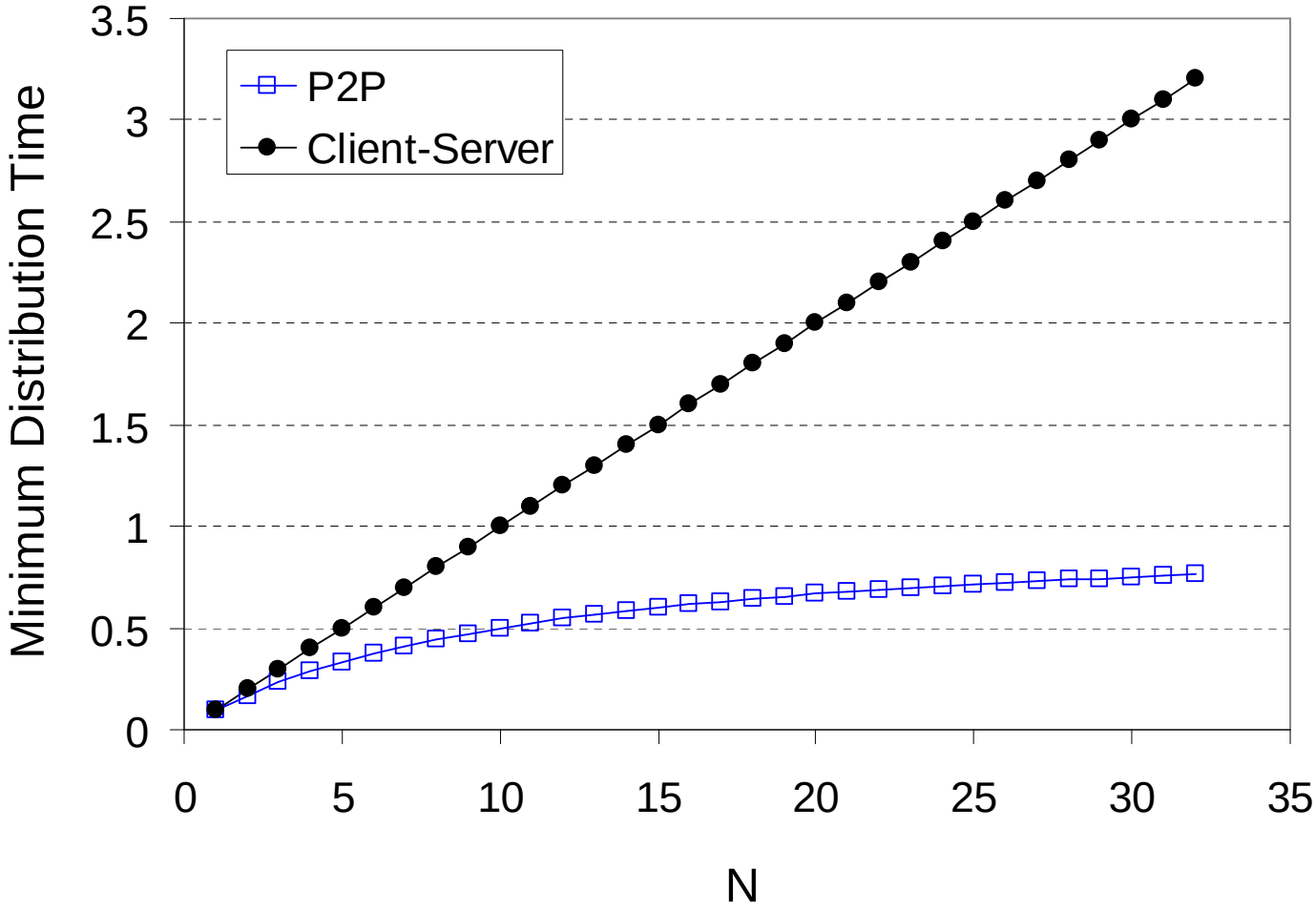
- Participantes colaboram para o funcionamento e manutenção do sistema



Cliente-servidor x P2P

Instituto de computação

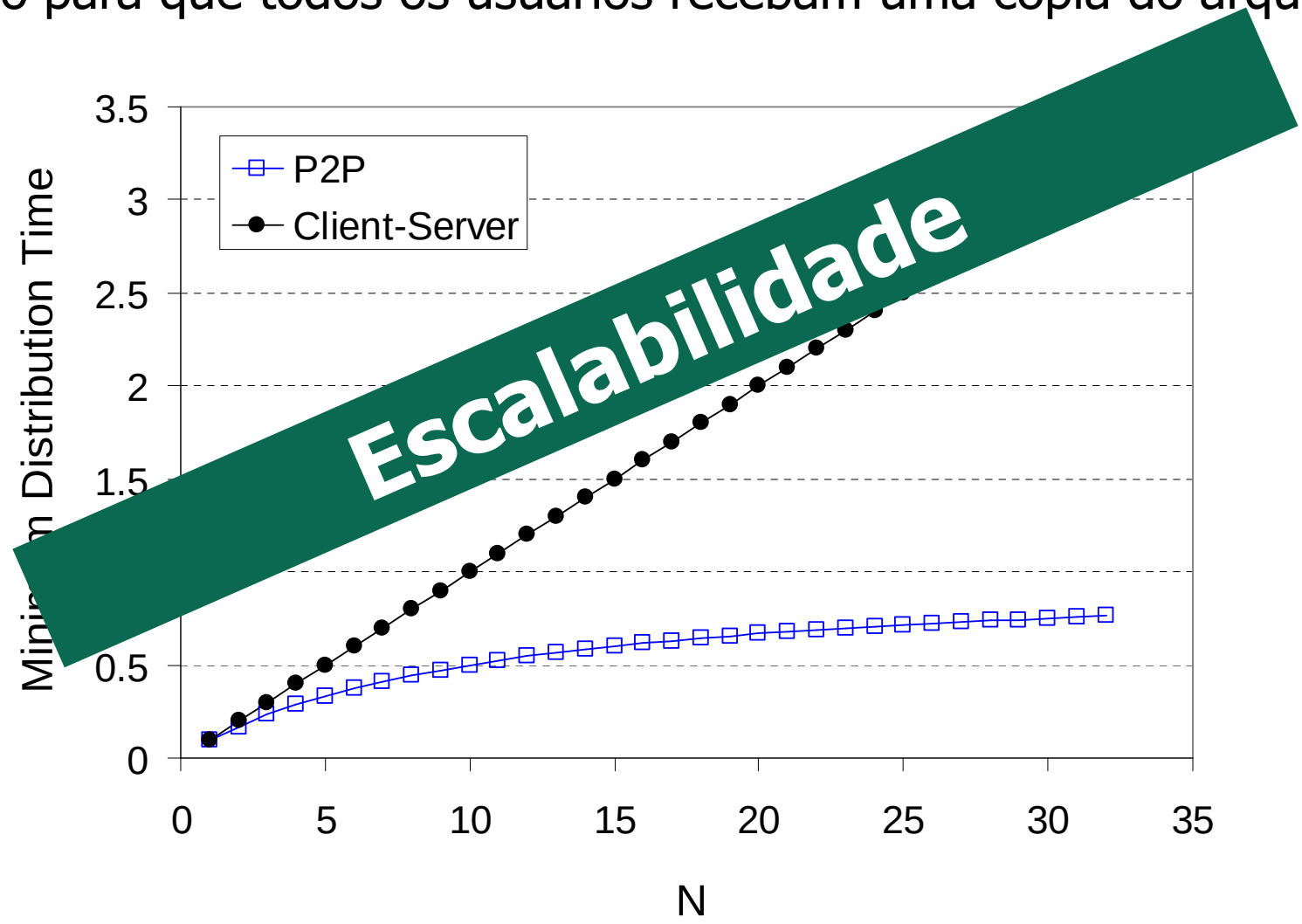
Tempo para que todos os usuários recebam uma cópia do arquivo



Cliente-servidor x P2P

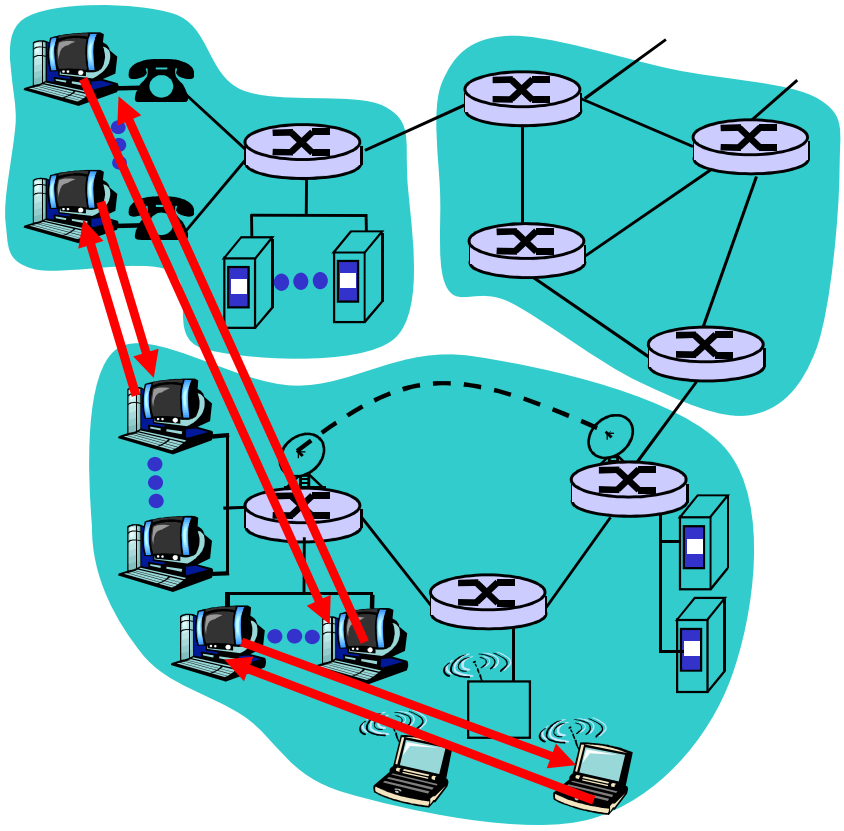
Instituto de computação

Tempo para que todos os usuários recebam uma cópia do arquivo



Sistemas Par-a-Par

- "Pura"
  - Comunicação direta entre sistemas finais
- Híbrida
  - Uso de servidores auxiliares
    - Ex.: Skype, BitTorrent, etc.



Sistemas Par-a-Par

Instituto de computação

- Compartilhamento de arquivos
  - Napster, Gnutella, Emule, Limewire, BitTorrent etc
- Distribuição de conteúdo multimídia
  - Áudio: Skype
  - Vídeo: SopCast, PPLive, PPStream, Joost etc.

Compartilhamento de Arquivos

Instituto de computação

- Primeiros sistemas P2P: idéia básica
  - Um usuário (par) quer receber um determinado arquivo
    - Uma música, um vídeo, um programa etc.
  - O primeiro passo é encontrar outros pares que possuem esse arquivo: **busca**
    - Obtém-se uma lista de parceiros
  - Em seguida, escolhe-se um parceiro que irá enviar o arquivo
  - Durante a recepção de um arquivo, pares podem enviar outros arquivos a quem interessar
    - Um par é tanto um cliente como um servidor temporário

Compartilhamento de Arquivos

Instituto de computação

- Primeiros sistemas P2P: idéia básica
  - Um usuário (par) quer receber um determinado
    - Uma música, um vídeo, um programa
  - O primeiro passo é encontrar os pares que possuem esse arquivo: **busca**
    - Obtém-se uma lista de parceiros
  - Em seguida, escolhe-se um parceiro que irá enviar o arquivo
  - Recebida a recepção de um arquivo, pares podem enviar os arquivos a quem interessar
    - Um par é tanto um cliente como um servidor temporário

Problema crítico: busca por arquivos

Busca

Instituto de computação

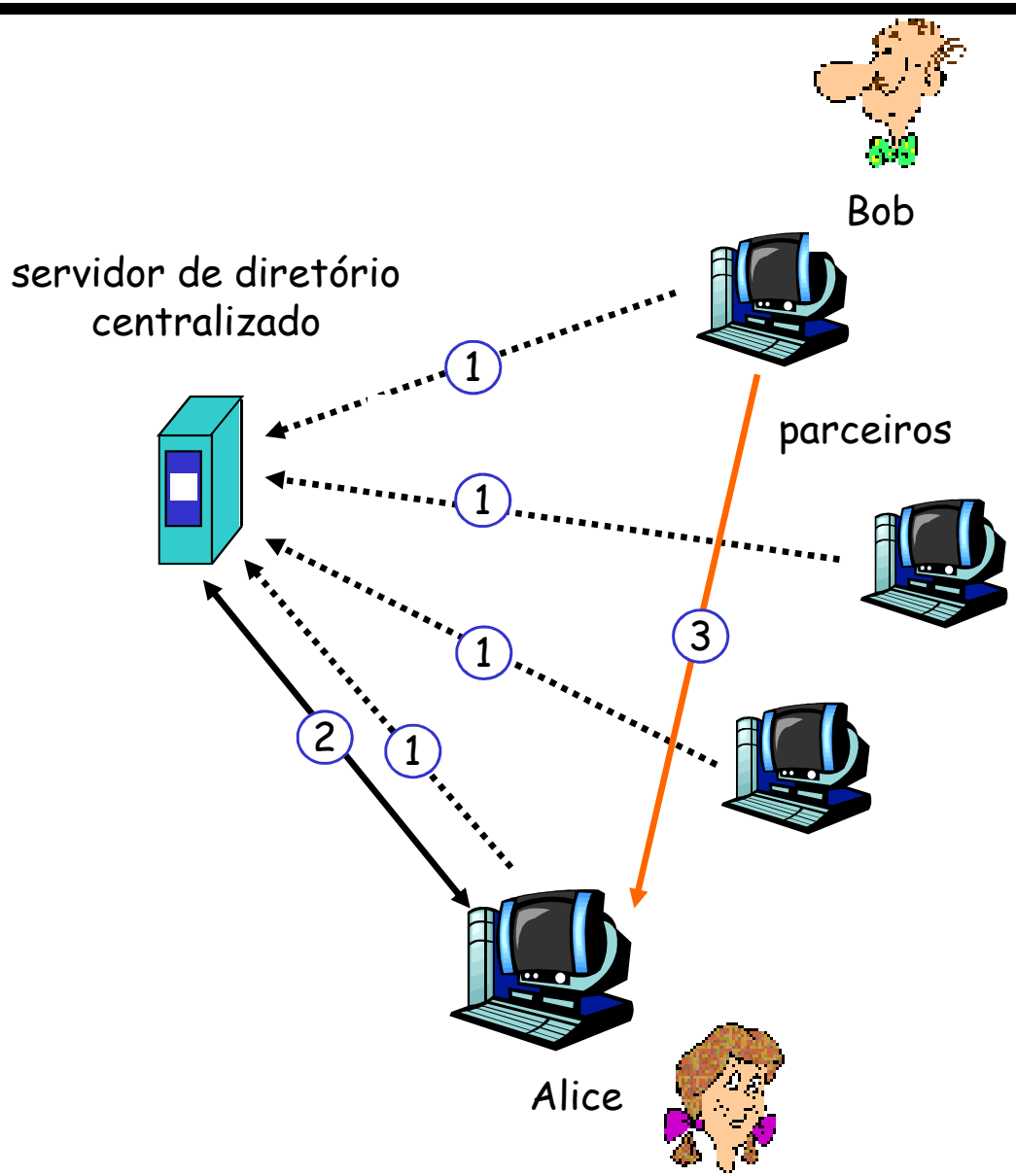
- Índice em um sistema par-a-par
  - Registra dinamicamente as localizações dos arquivos compartilhados pelos pares
  - Mapeia informação na localização de um par
- Pares devem informar ao índice os conteúdos que possuem
- Pares buscam no índice para descobrir onde podem encontrar os arquivos

Como construir o índice?

Diretório Centralizado

Napster

1) Quando um parceiro conecta ele informa ao servidor central o seu:
   - Endereço IP
   - Conteúdo
2) Alice consulta sobre a música "Hey Jude"
3) Alice solicita o arquivo a Bob

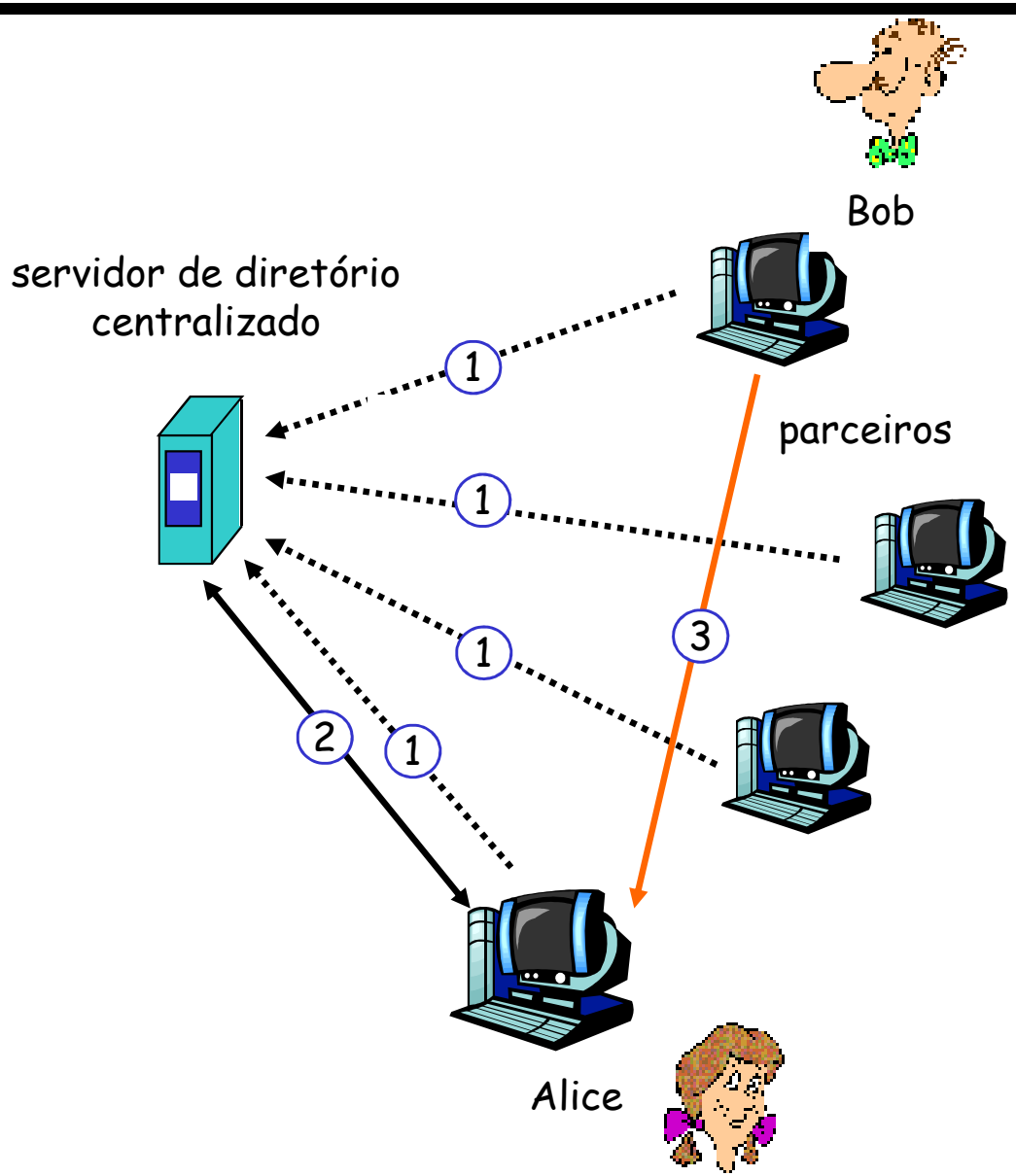


Diretório Centralizado

Instituto de computação

Problemas

1. Ponto único de falha
2. Gargalo de desempenho
3. Violação de Direitos Autorais



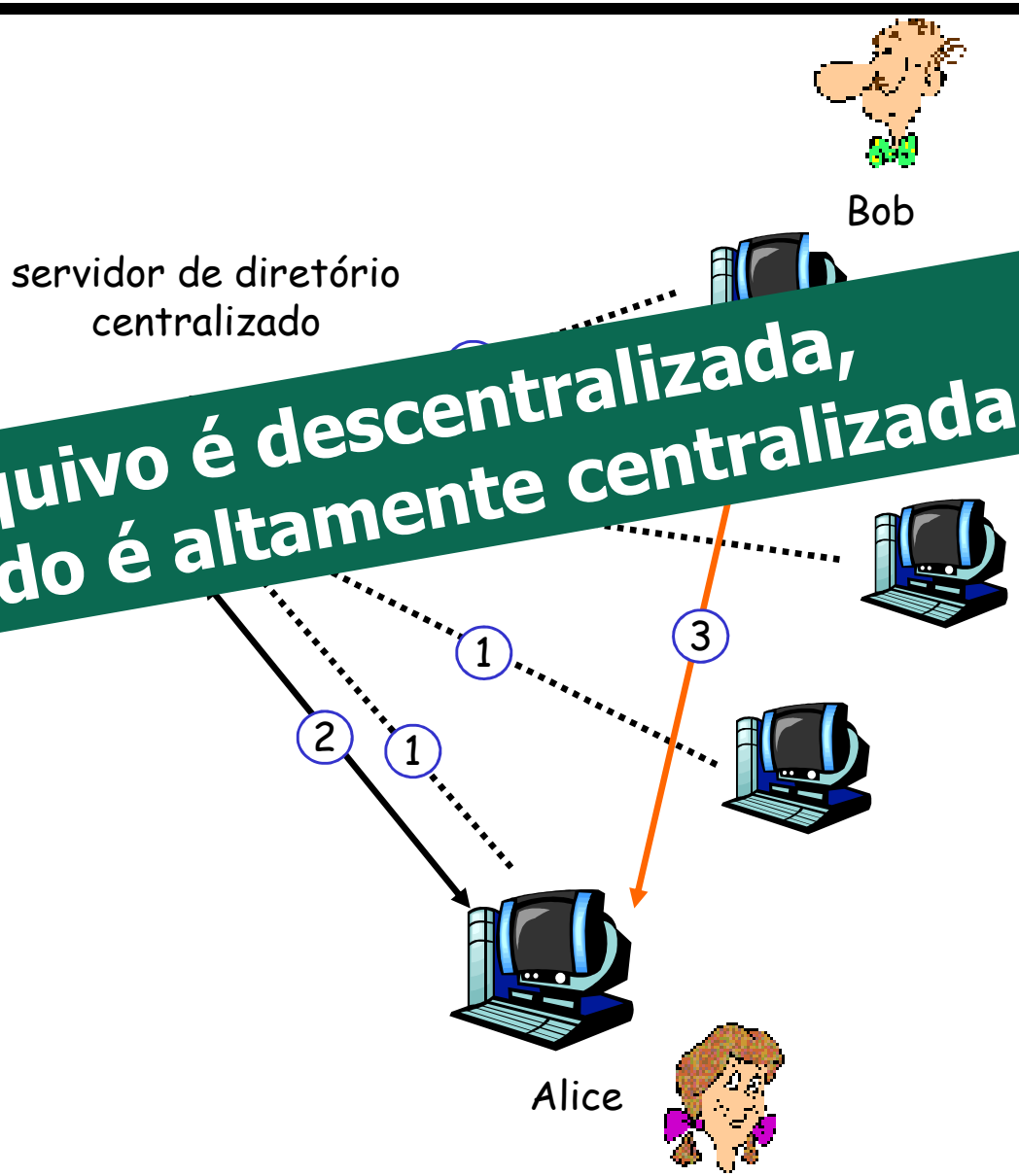
Diretório Centralizado

Instituto de computação

Problemas

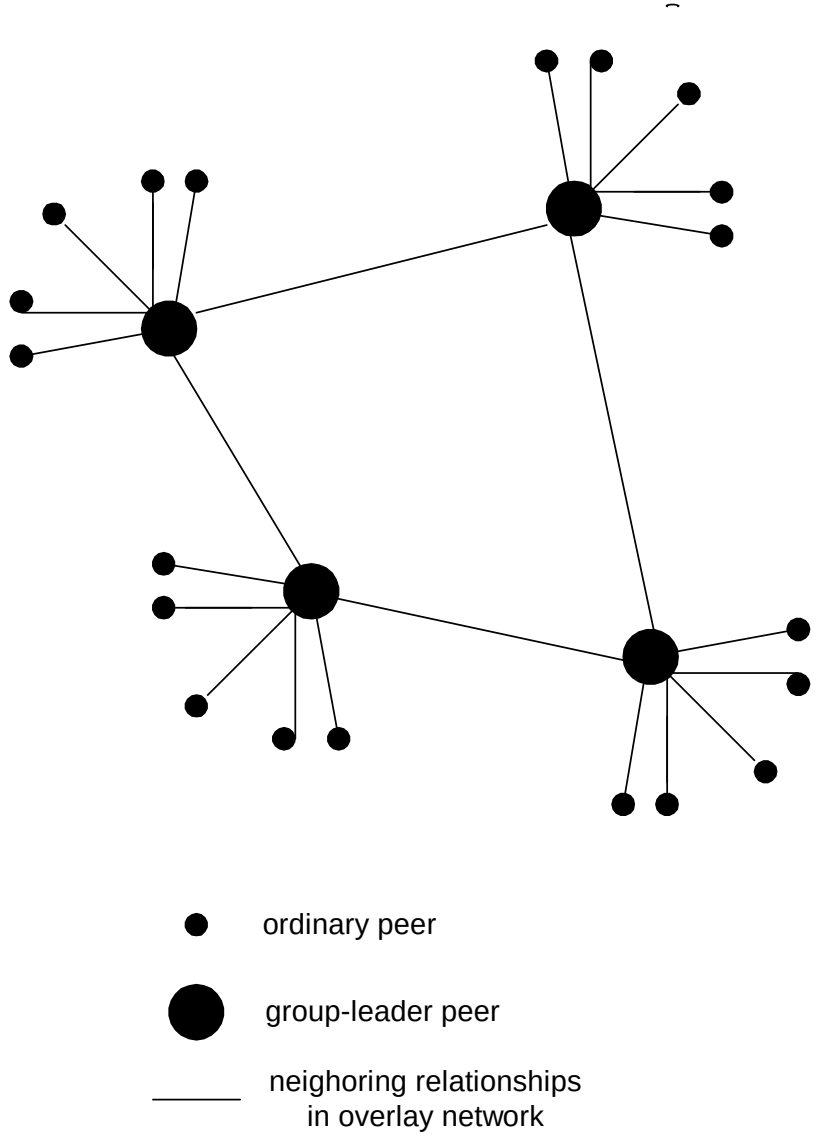
1. Ponto único de falha
2. Gargalo de desempenho
3. Violação de Direitos Autorais

A transferência de arquivo é descentralizada, mas a busca pelo conteúdo é altamente centralizada



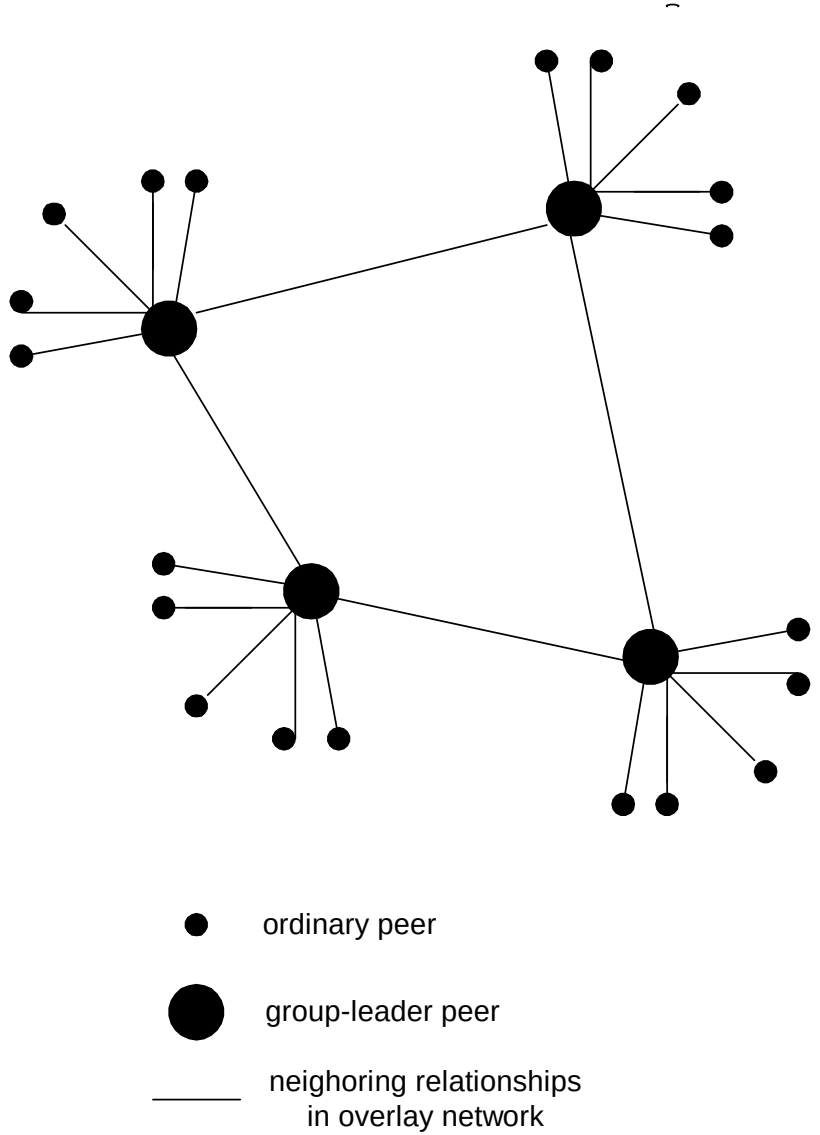
Hierarquia

- Conhecimento da localização dos arquivos é distribuído entre os pares
  - Nenhum par conhece a localização de todos os arquivos
- Ex.: Gnutella



Hierarquia

- Cada parceiro é um líder de grupo ou está alocado a um líder de grupo
  - Conexão TCP entre cada par e o seu líder de grupo
  - Conexões TCP entre alguns pares de líderes de grupos
- Líder de um grupo
  - Mantém registro sobre o conteúdo de todos os seus filhos



Hierarquia

- Cada parceiro é um líder de grupo ou está alocado a um líder de grupo
  - Conexão TCP entre cada par e o seu líder de grupo
  - Conexões TCP
- Líd grupo
  - Mantém registro sobre o conteúdo de todos os seus filhos

Líderes de grupo só conhecem parte da localização dos arquivos

Tabelas *Hash* Distribuídas

Instituto de computação

- DHTs (*Distributed Hash Tables*)
- Informações representadas por um par **`(chave,valor)`**
  - (1980, Igor)
  - (Led Zeppelin IV, 192.168.2.1)

- Pares fazem buscas com base nas chaves
  - DHT retorna o valor associado a chave procurada
    - Chave: "Led Zeppelin IV", valor: 192.168.2.1

- Pares também podem inserir pares **`(chave,valor)`**na tabela

Tabelas *Hash* Distribuídas

Instituto de computação

- Aplicar uma função *hash* para gerar as chaves
  - Pequena probabilidade de colisão → **identificação única**
  - Tamanho fixo
  - Espalhamento
    - Distribuição de carga
  - Algumas garantem que não é possível a partir de um valor de *hash* retornar à informação original

Identificadores da DHT

Instituto de computação

- Para cada par é atribuído um identificador inteiro entre $[0,2^n-1]$
  - Cada identificador é representado por *n* bits
    - É o tamanho da função *hash*
- Cada chave também é identificada por um inteiro no mesmo intervalo
  - Obtido pela aplicação de uma função *hash*
    - Ex.: chave = h("Led Zeppelin IV")

Como atribuir chaves aos pares?

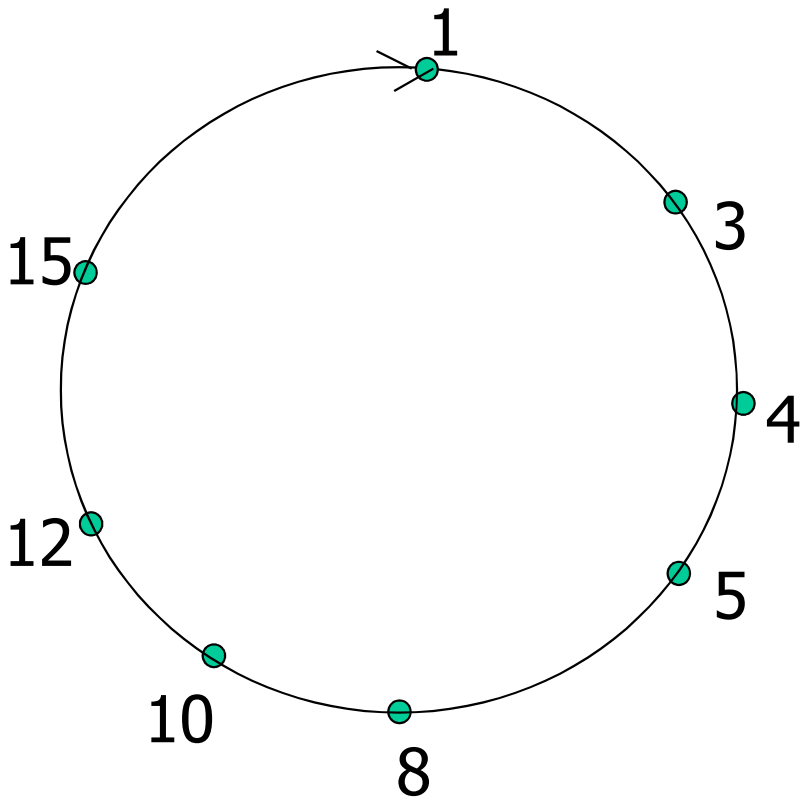
Instituto de computação

- Ponto-chave
  - Atribuir pares (chave,valor) aos parceiros
    - Que par sabe o valor de uma dada chave?
- Regra: atribuir uma chave ao par com o identificador mais próximo do identificador da chave
  - Mais próximo é o **sucessor imediato** da chave

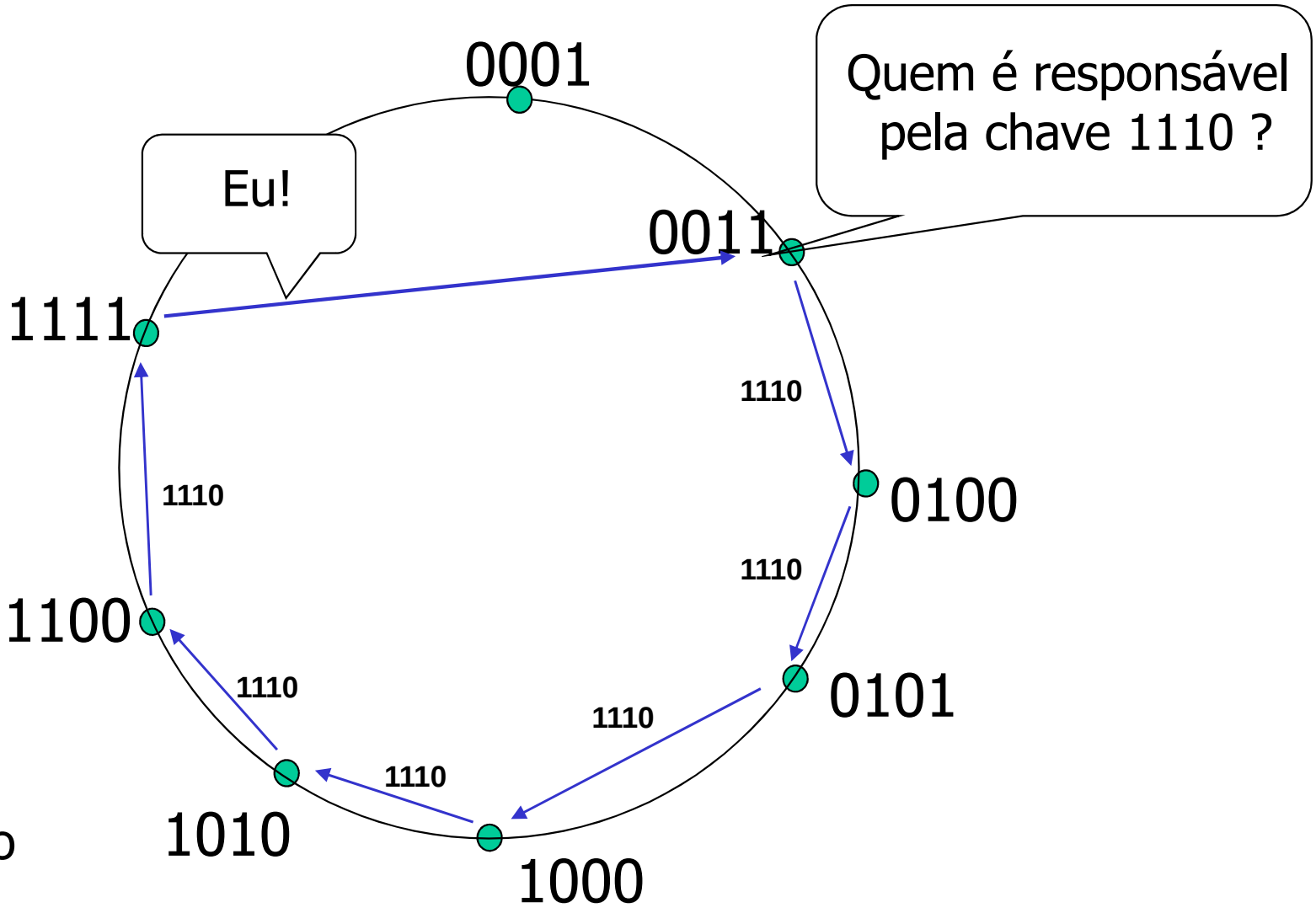
- Ex: n=4 [0…15]; pares: 1,3,4,5,8,10,12,14;
  - chave = 13 → successor é o par = 14
  - chave = 15 → successor é o par = 1

DHT Circular

- Cada par só conhece o seu sucessor imediato e o seu antecessor
- Rede sobreposta (*overlay network*)

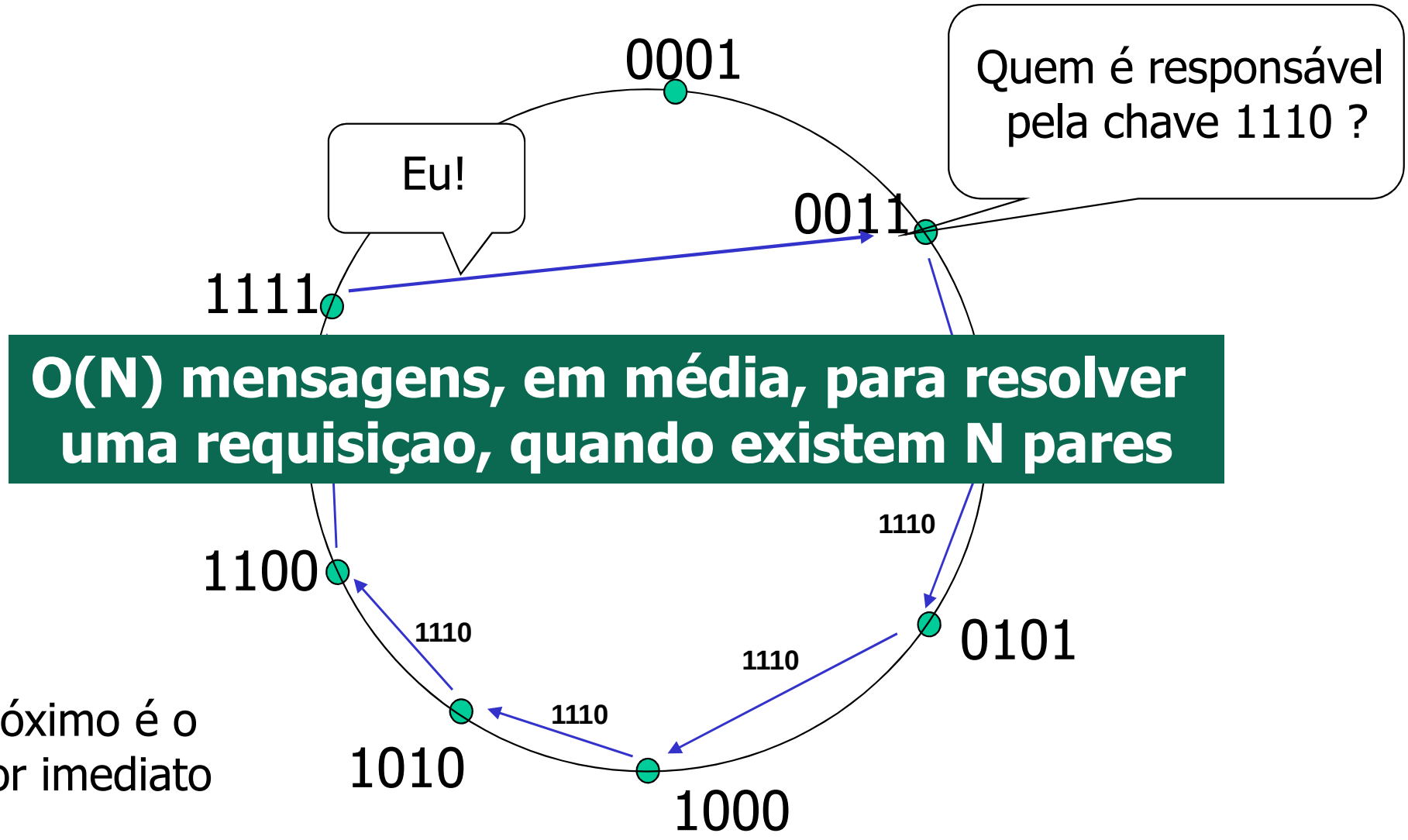


DHT Circular



Mais próximo é o sucessor imediato

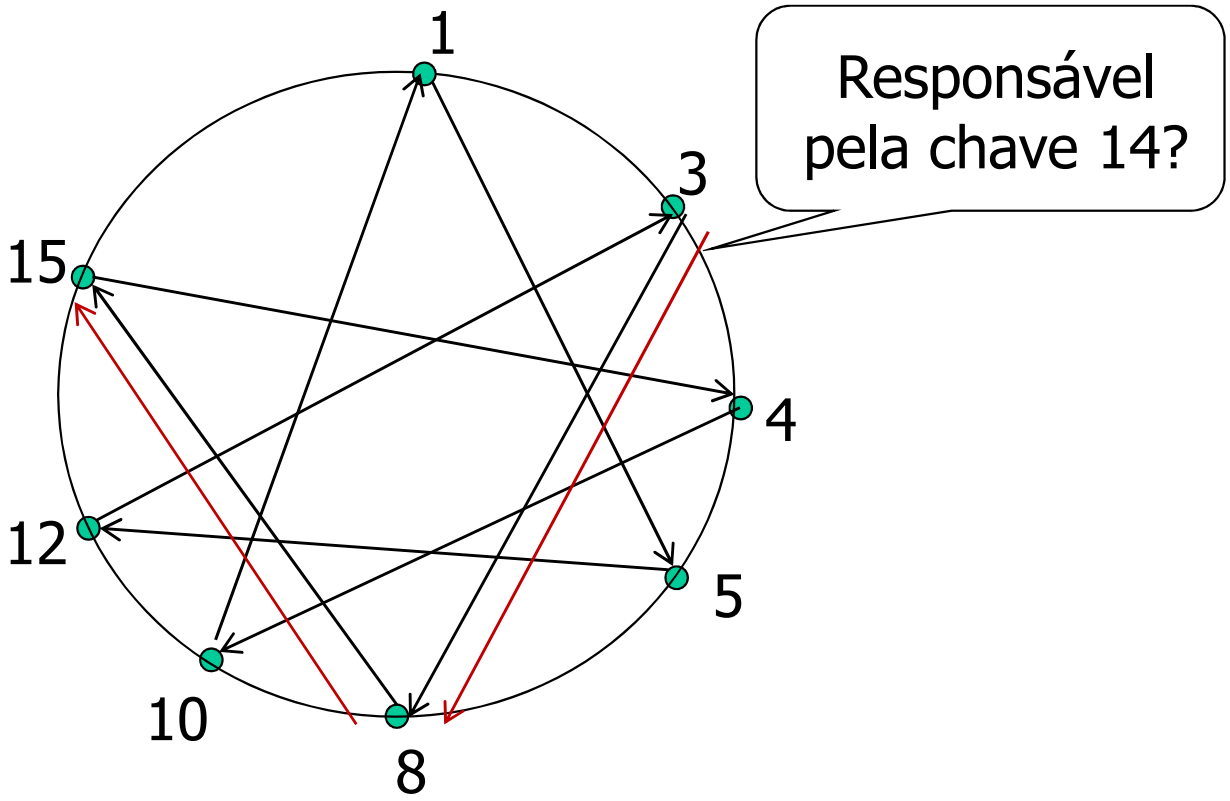
DHT Circular



Mais próximo é o sucessor imediato

DHT Circular com Atalhos

- Cada par sabe o endereço do seu antecessor, do seu sucessor e de alguns outros pares (atalhos)
  - Reduzir o número de mensagens: O(N) → O(log N)
    - No exemplo: 6 mensagens → 2 mensagens



Exemplo: Pastry

Instituto de computação

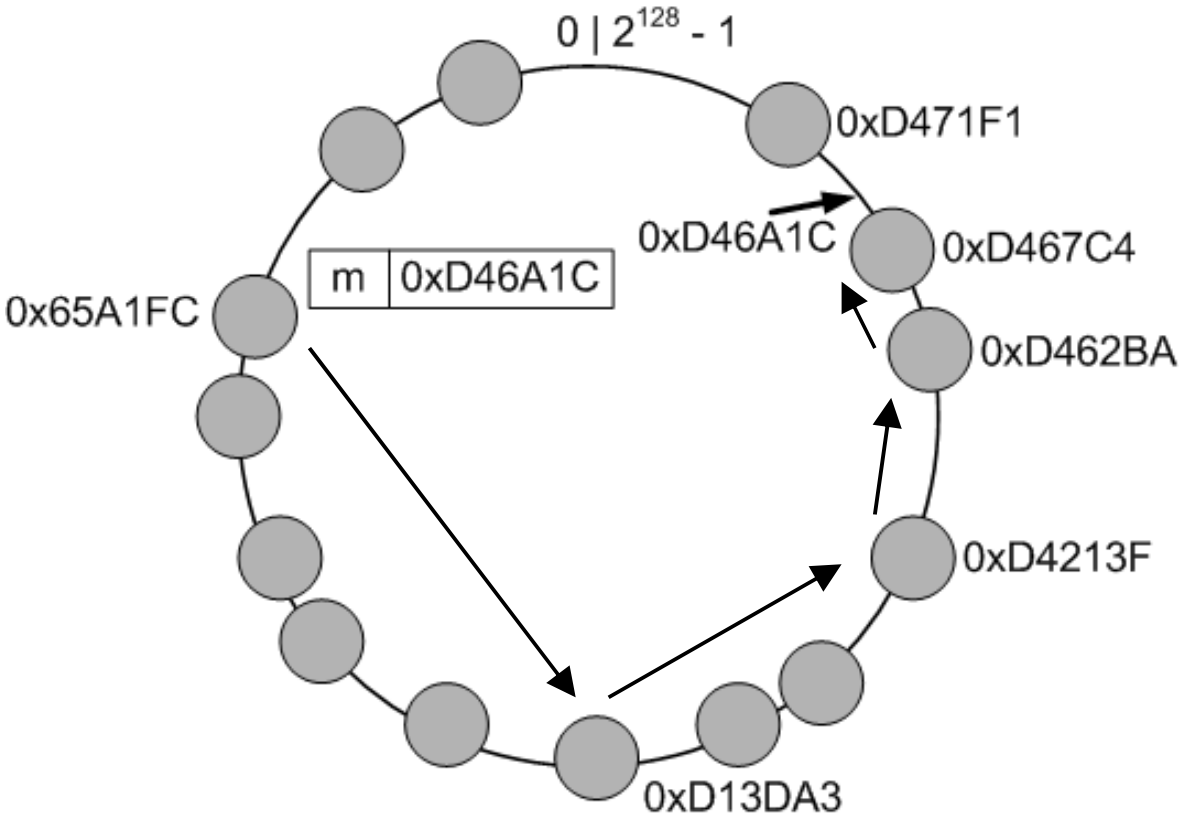
- Recomendado para **construção da rede sobreposta**
- Estima a proximidade entre os nós para construir os enlaces da rede sobreposta
  - Nós são capazes de medir sua distância para outro nó de endereço IP conhecido
  - Construção de caminhos próximos aos da camada de rede
    - Característica desejada
- Distância
  - Número de saltos (`traceroute`)
  - Tempo de ida-e-volta (`ping`)
  - Vazão (par de pacotes)

Funcionamento do Pastry

Instituto de computação

- Para cada nó → um identificador de 128 bits
  - Função *hash* do endereço IP ou da chave pública do nó
  - Conjunto de identificadores uniformemente distribuído
- Para cada objeto → uma chave de 128 bits
- Dada uma mensagem e uma chave
  - A mensagem é encaminhada para o nó com identificador numericamente mais próximo da chave

Funcionamento do Pastry



BitTorrent

Instituto de computação

- Ponto-chave: arquivos divididos em **pedaços** que são **espalhados** pelos pares
- *Torrent* ou enxame (*swarm*)
  - Grupo de pares trocando pedaços de um arquivo
- *Tracker* ou rastreador
  - Registra pares participantes de um *torrent*
    - Usuários interessados em um mesmo arquivo
  - Endereço do *tracker* obtido em um arquivo **`.torrent`**
    - Como obter esse arquivo está **fora do escopo** do protocolo

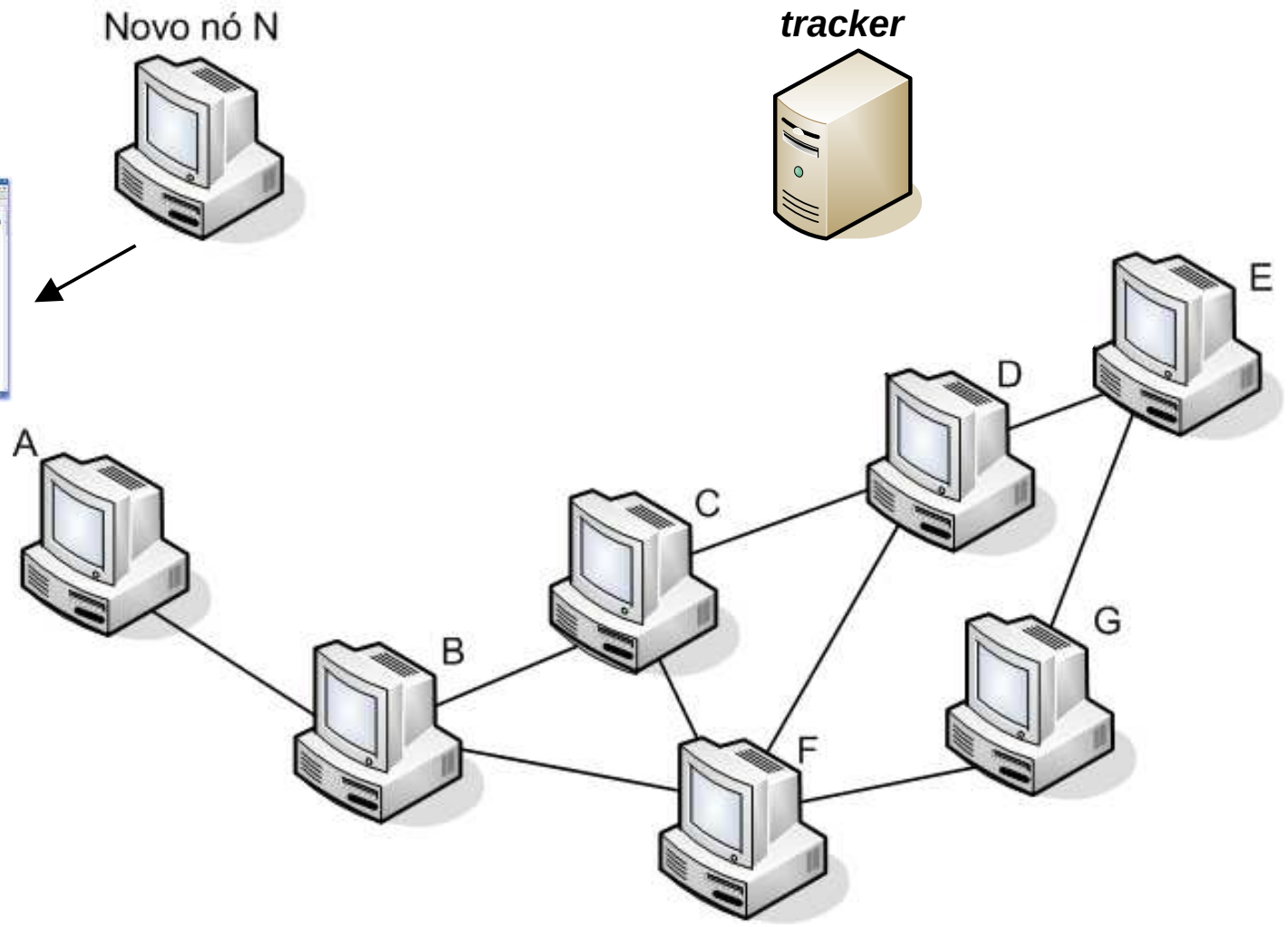
BitTorrent

Instituto de computação

- Ponto-chave: arquivos divididos em **pedaços** que são **espalhados** pelos pares
- *Torrent* ou enxame (*swarm*)
  - Grupo de pares trocando pedaços de um arquivo
- *Tracker* ou rastreador
  - Registra pares participantes de um *torrent*
    - Usuários interessados em um mesmo arquivo
  - Endereço do *tracker* obtido em um arquivo **`.torrent`**
    - Como obter esse arquivo está **fora do escopo** do protocolo

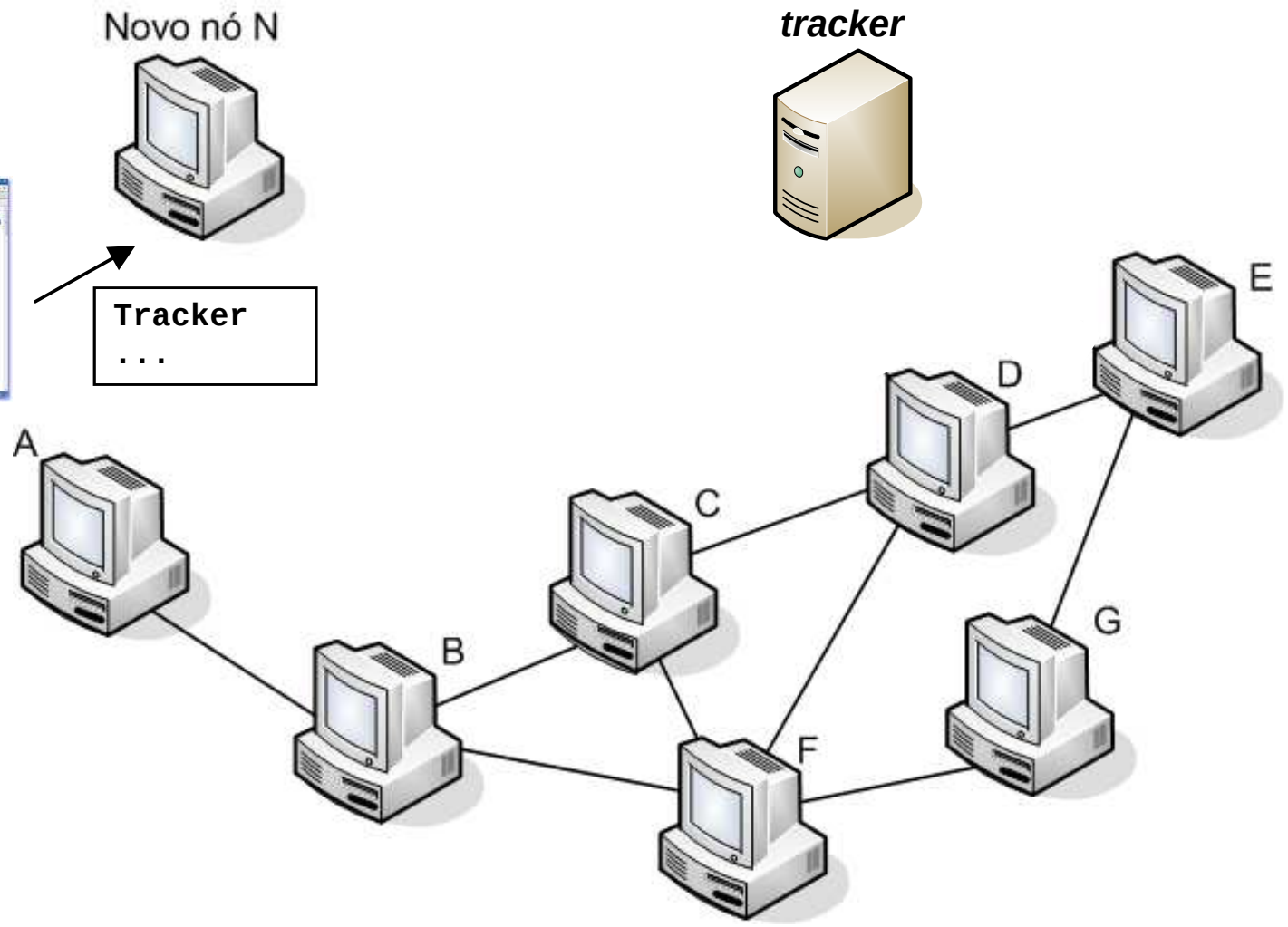
Vantagem: reduz a complexidade da busca por arquivos de interesse

BitTorrent



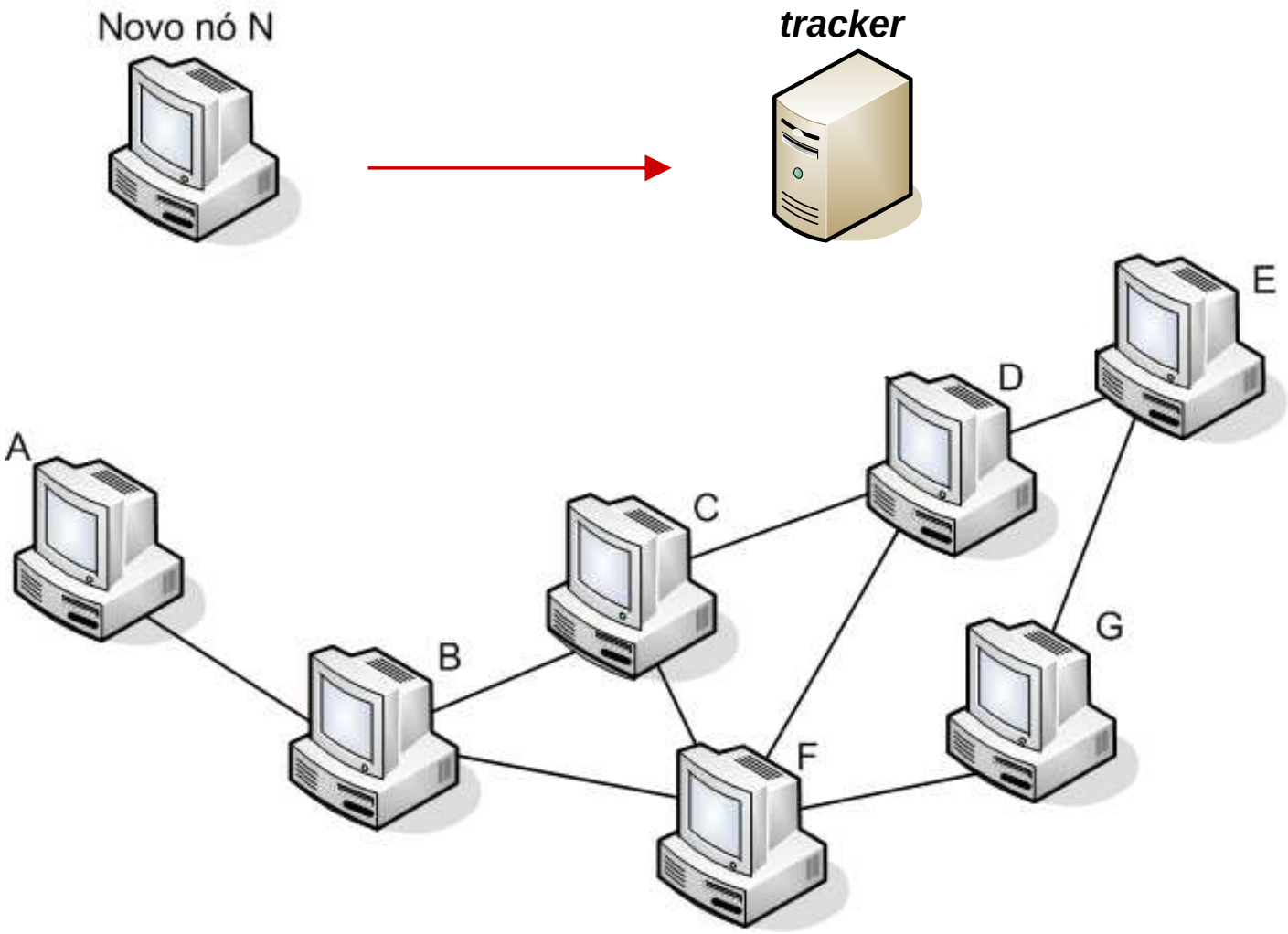
Nó interessado em um arquivo obtém um arquivo `.torrent`

BitTorrent



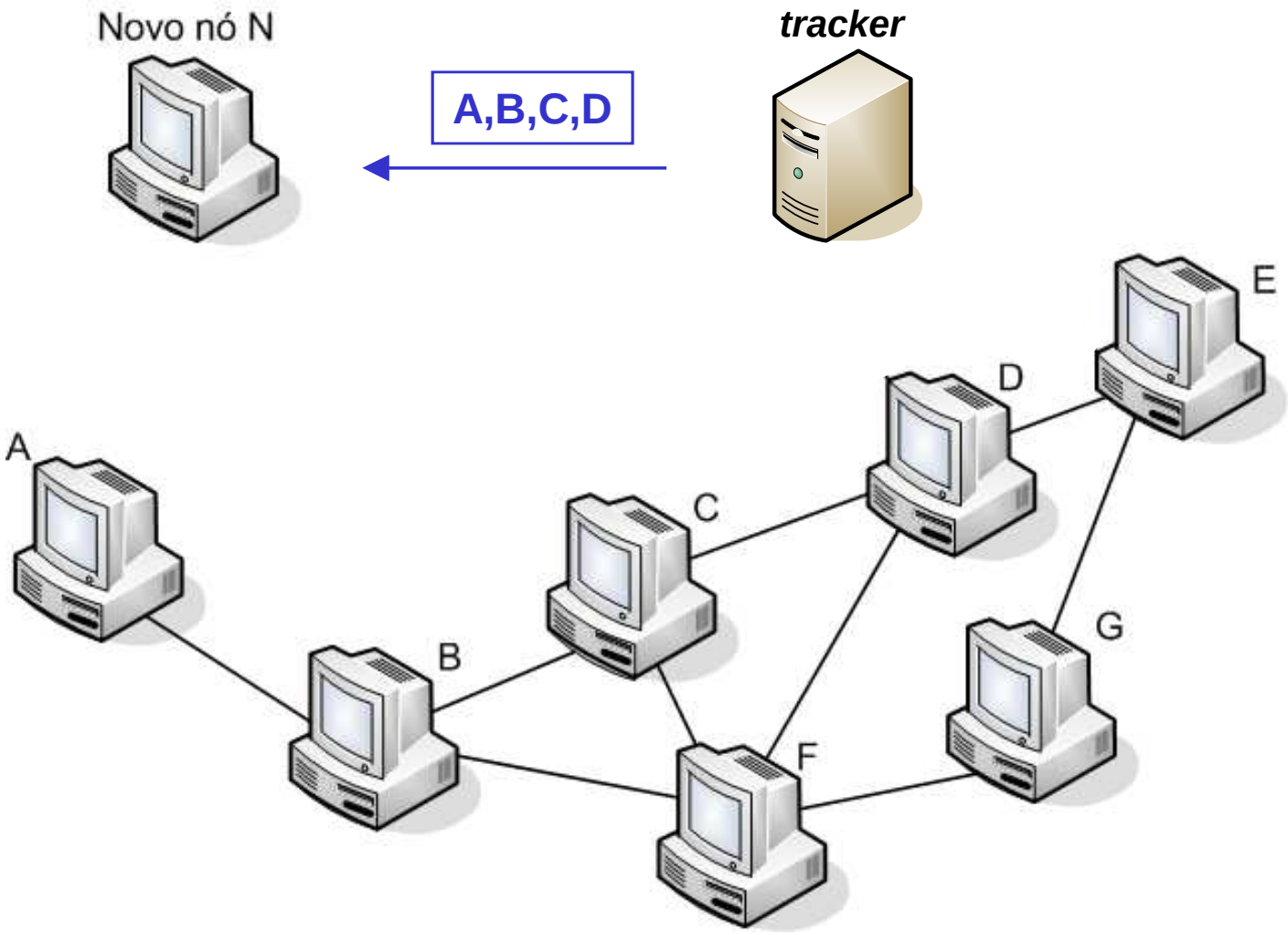
O endereço do rastreador (*tracker*) está nesse arquivo

BitTorrent



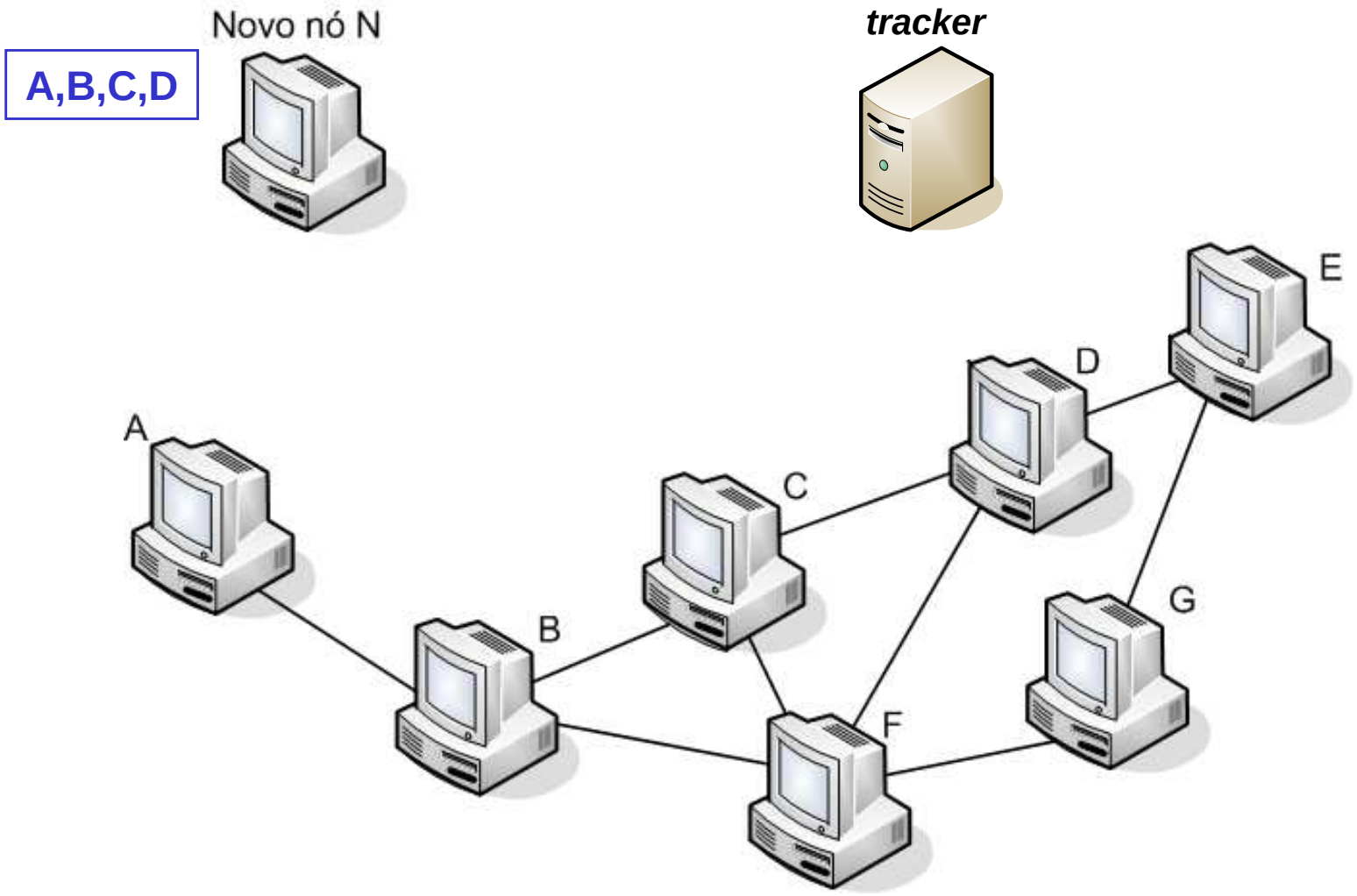
Novo nó N se registra no rastreador (*tracker*)

BitTorrent



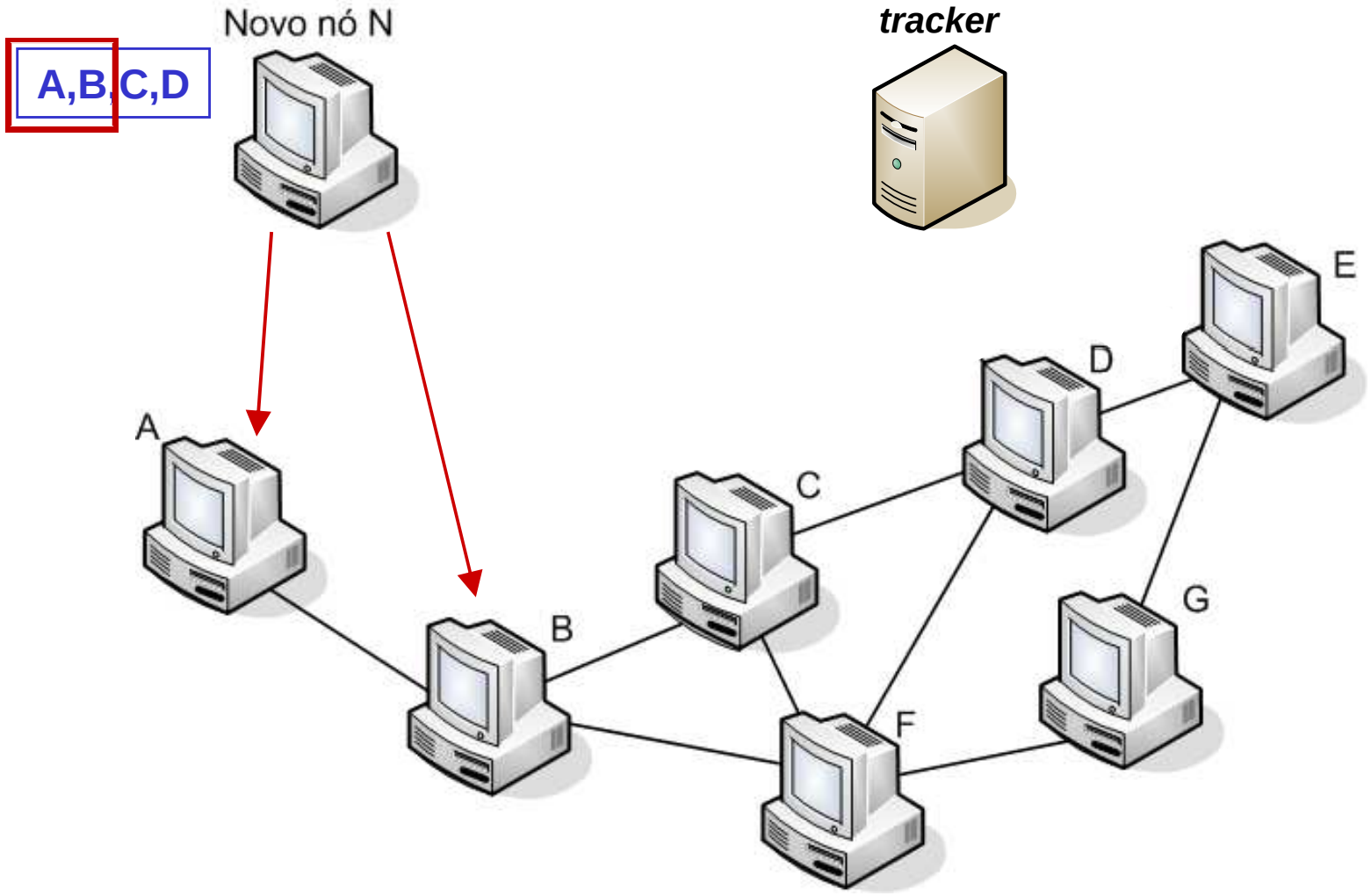
O *tracker* envia para o nó N um lista de possíveis vizinhos

BitTorrent



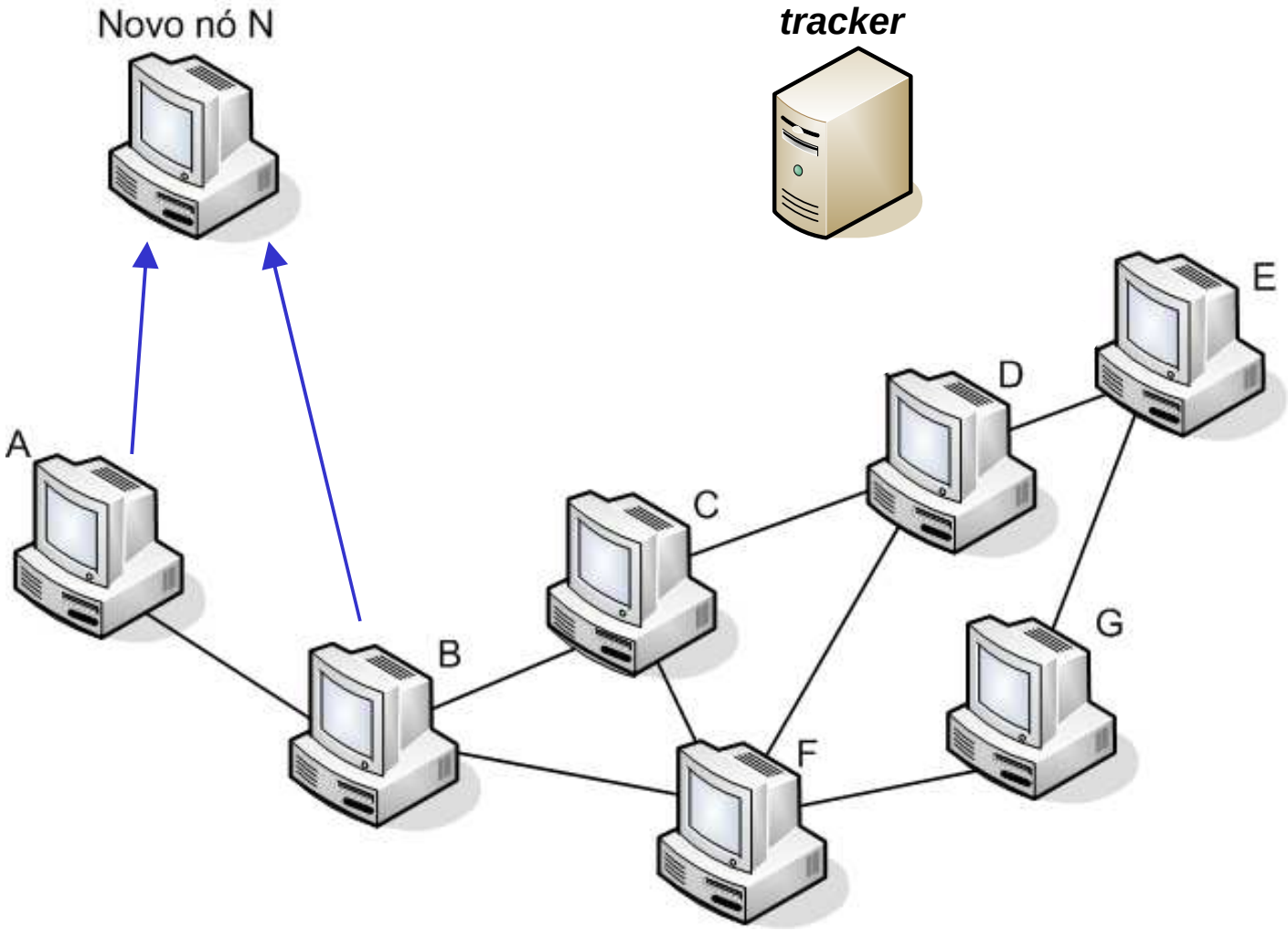
O nó N escolhe aleatoriamente um subconjunto de candidatos a vizinhos

BitTorrent



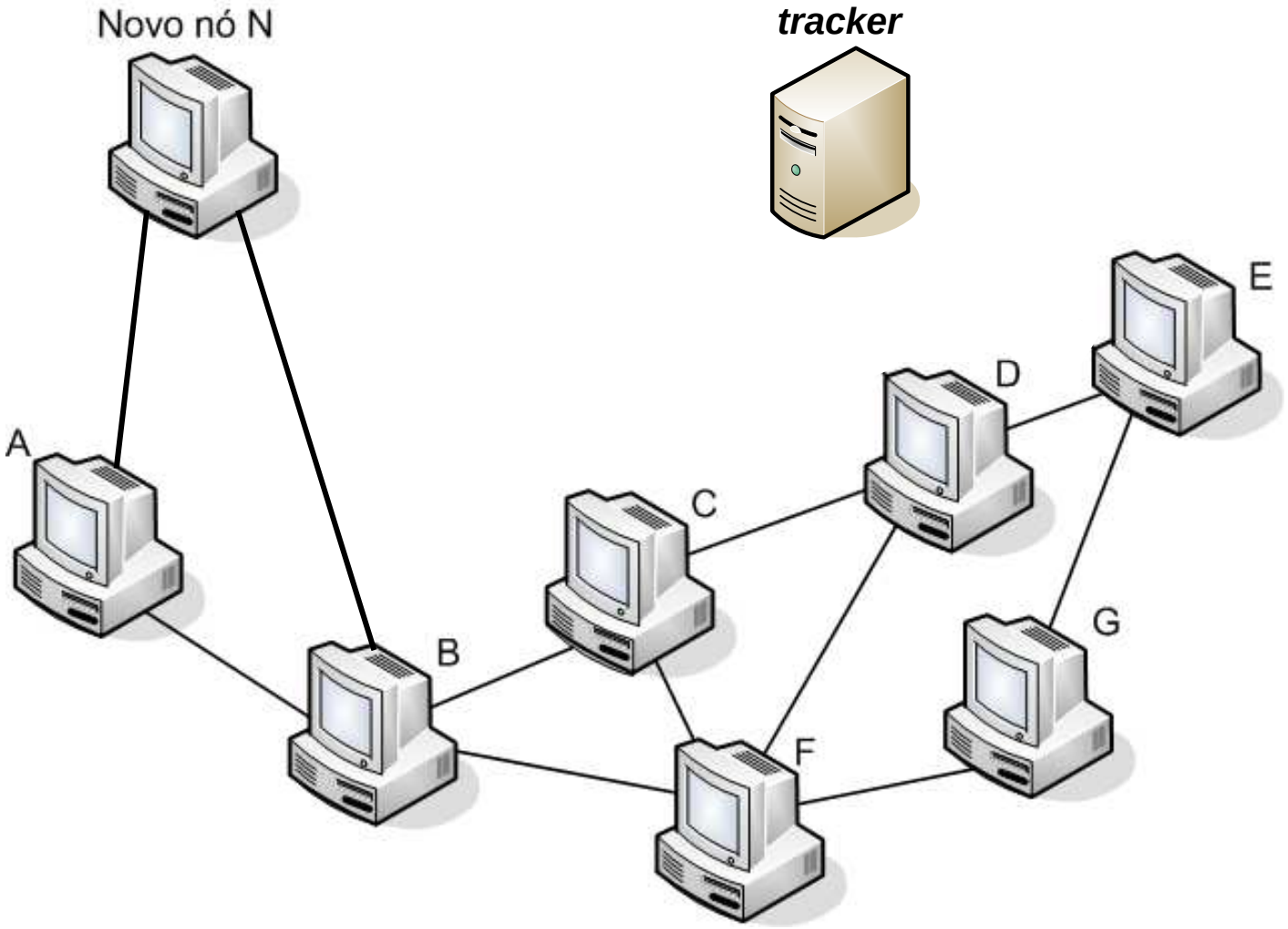
Novo nó envia uma mensagem para estabelecer sua vizinhança

BitTorrent



Em caso de resposta positiva, os enlaces são criados e os nós se tornam vizinhos

BitTorrent



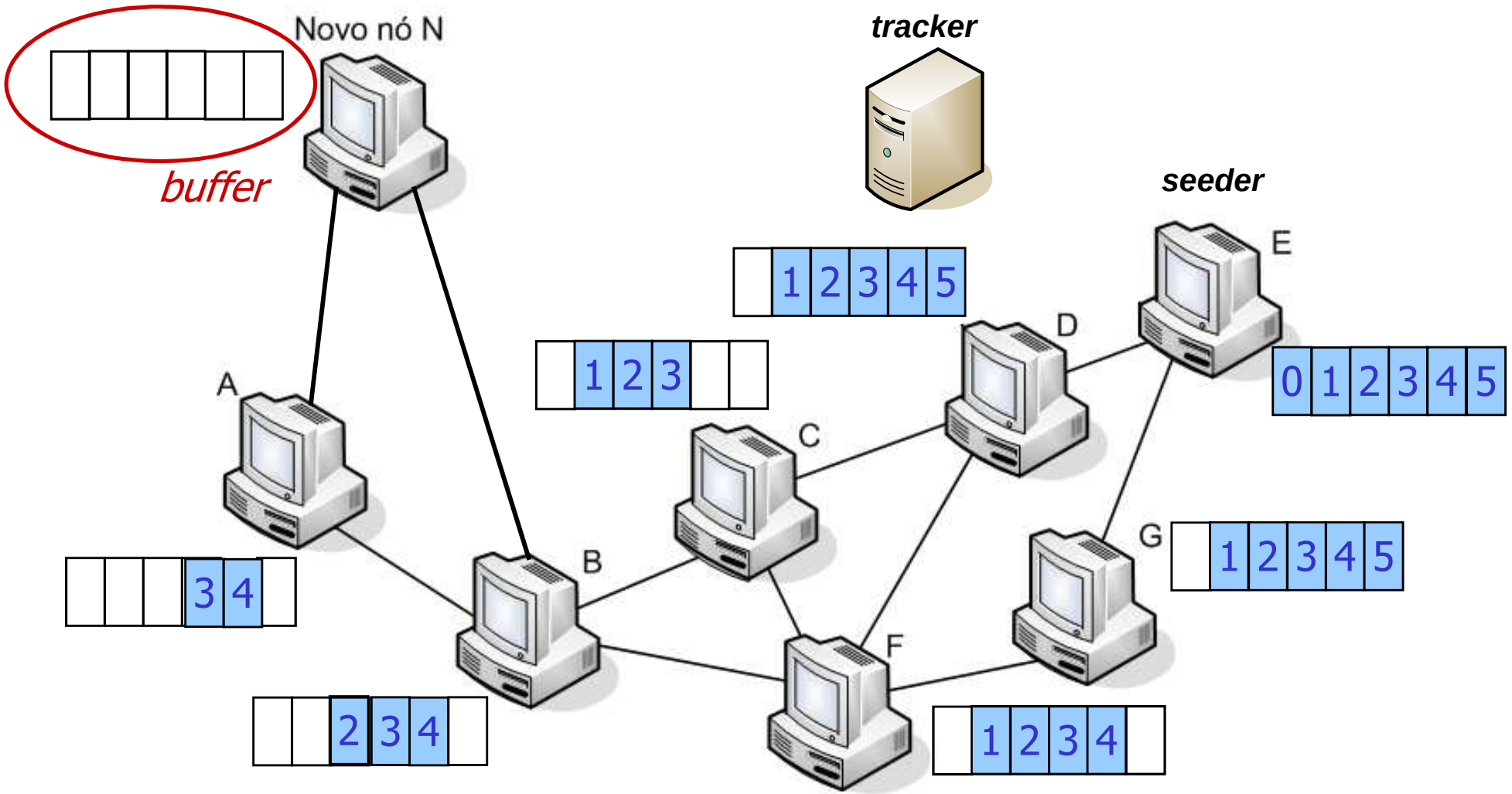
Em caso de resposta positiva, os enlaces são criados e os nós se tornam vizinhos

BitTorrent

Instituto de computação

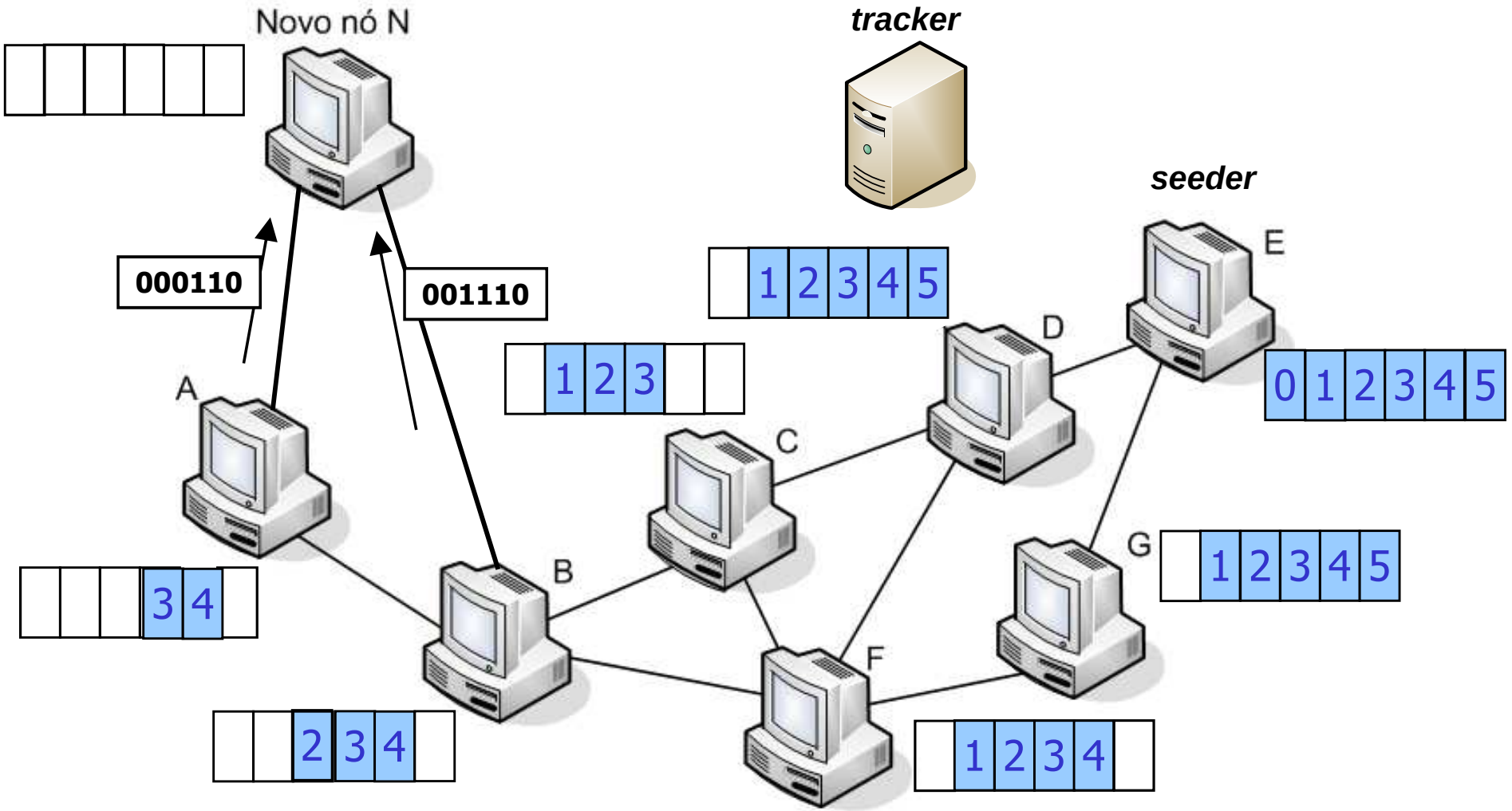
- Arquivo dividido em **pedaços** de 256 KB
- Nó recém-chegado ao enxame
  - Não tem nenhum pedaço, mas irá acumulá-los com o tempo
  - Enquanto faz o *download*, par carrega pedaços para outros pares
    - A troca de pedaços é feita **somente** entre os pares
- Semeadores (*seeders*)
  - Pares que possuem todos os pedaços de um arquivo
- Sangue-sugas (*leechers*)
  - Pares que estão baixando os pedaços de um arquivo

BiTorrent



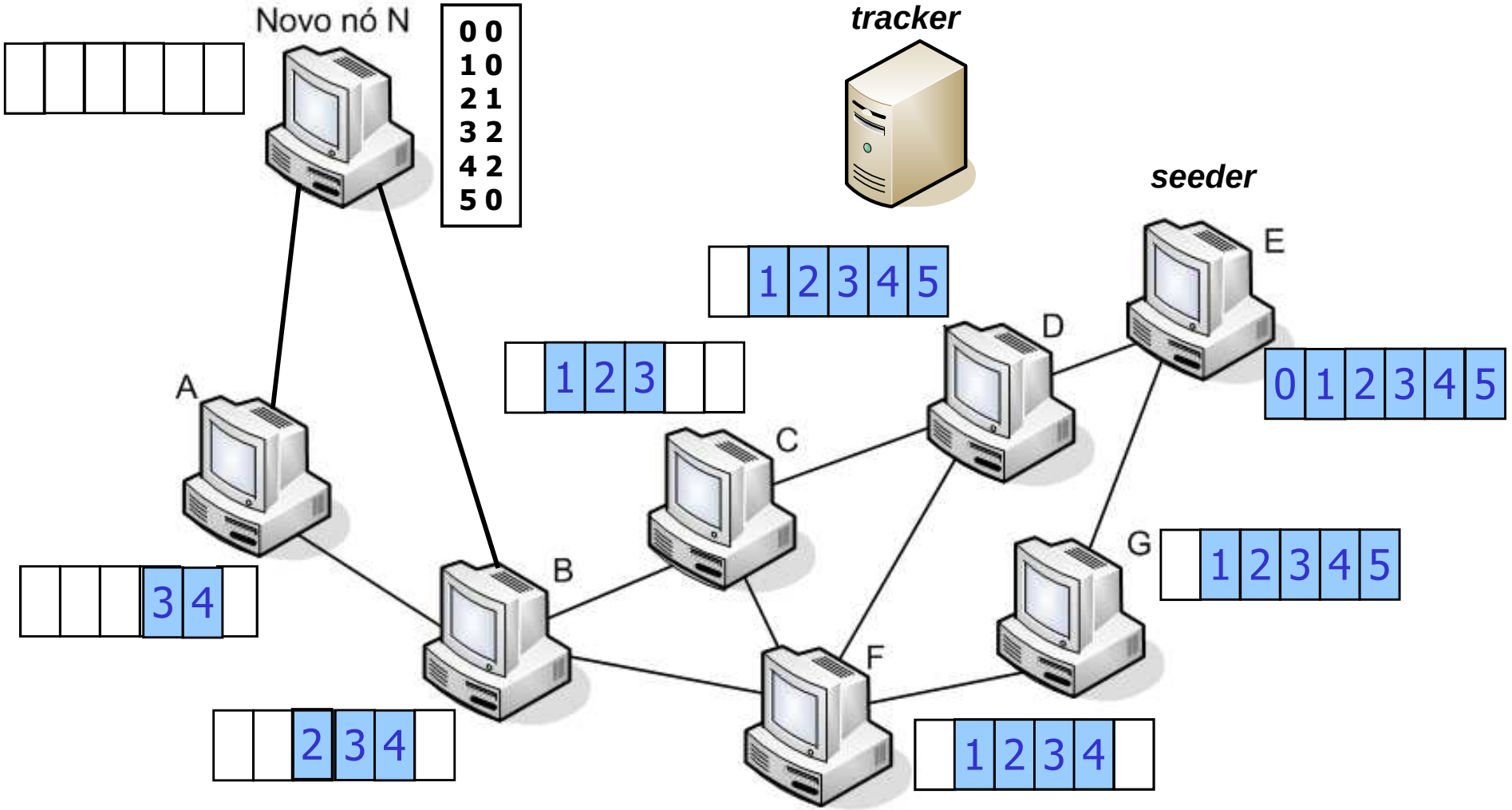
O nó N recém-chegado inicialmente não possui nenhum pedaço

BiTorrent



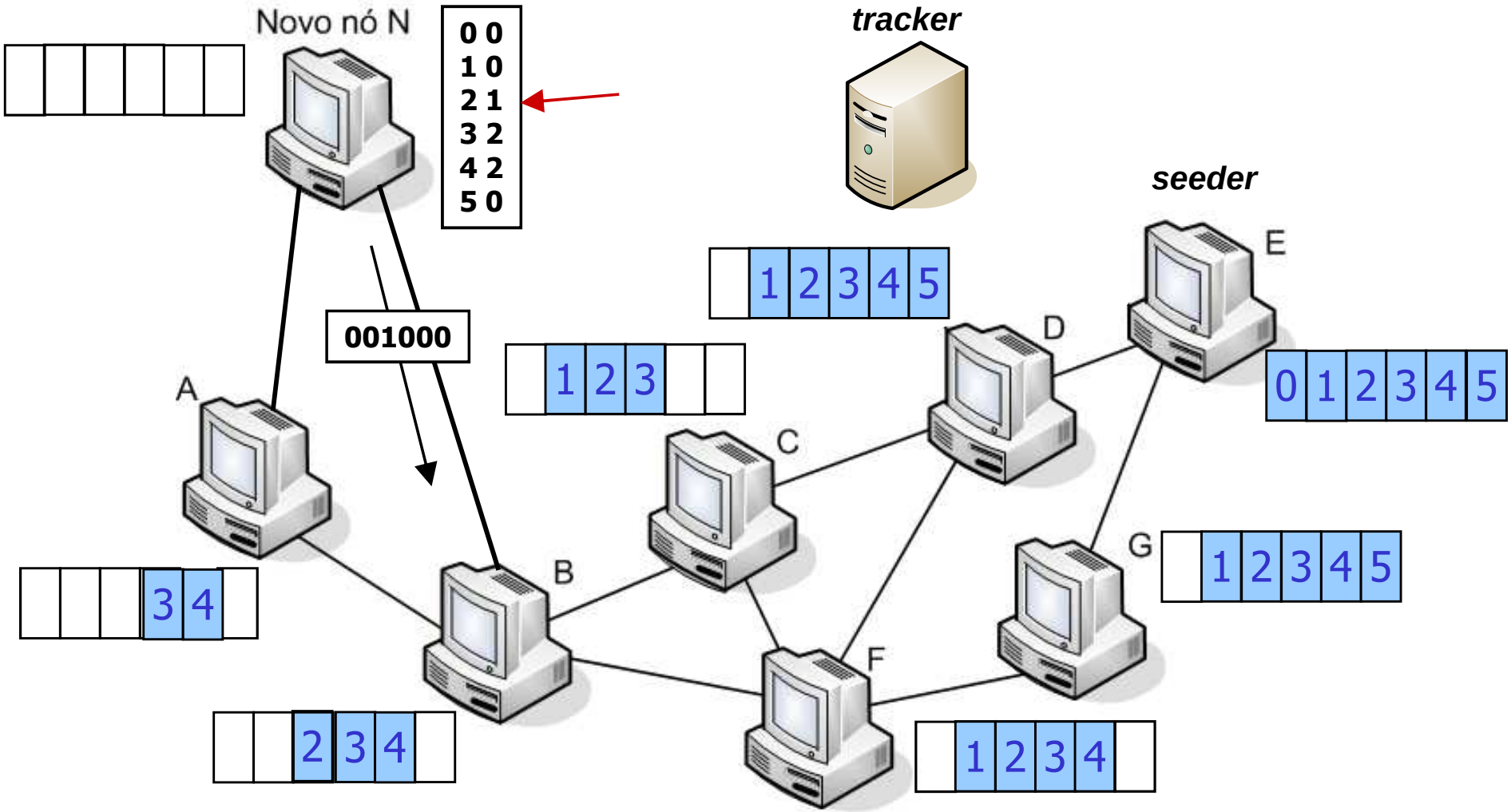
Vizinhos enviam um mapa dos pedaços que possuem

BiTorrent



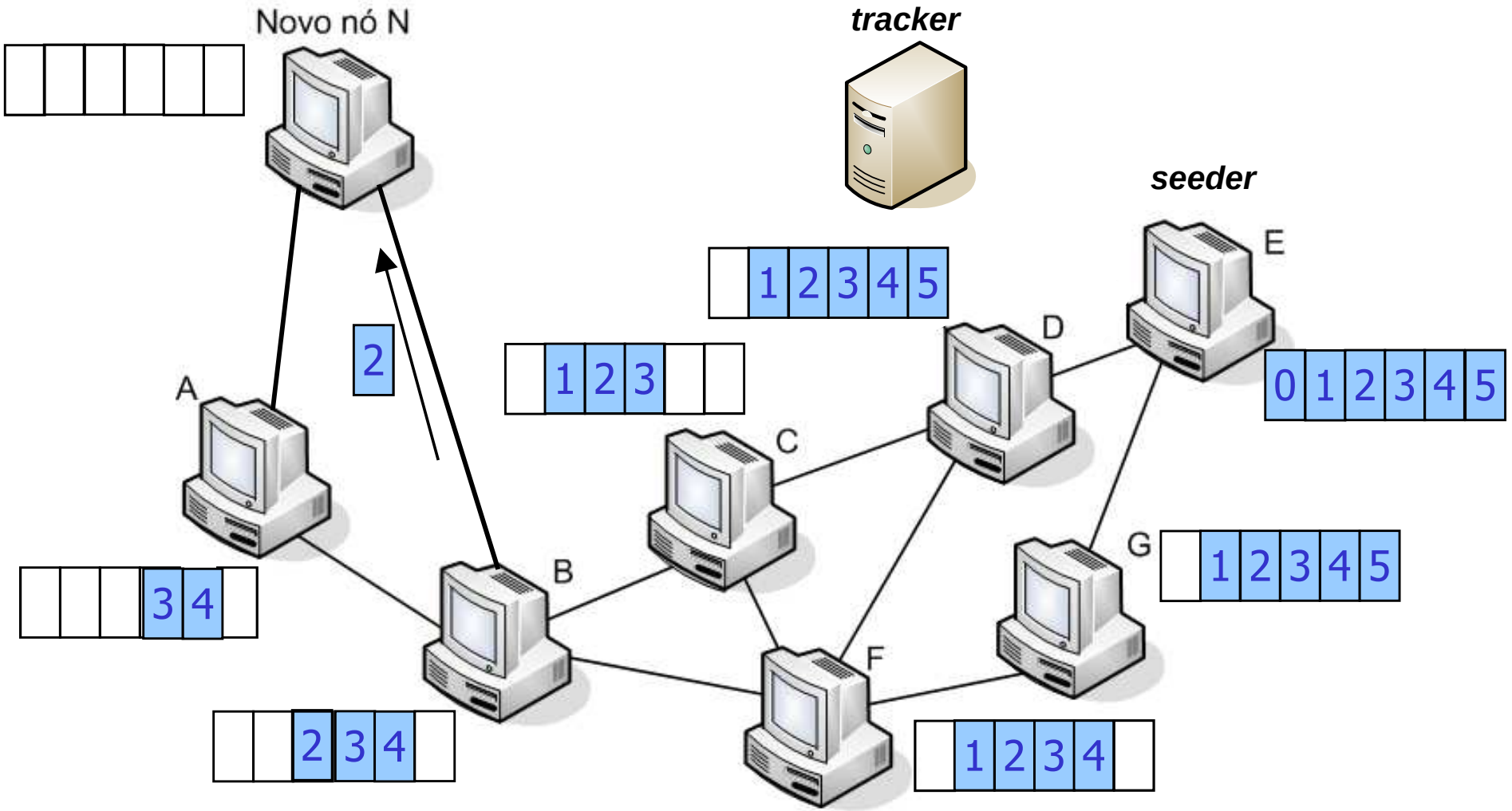
O nó N recebe os mapas dos seus parceiros

BiTorrent



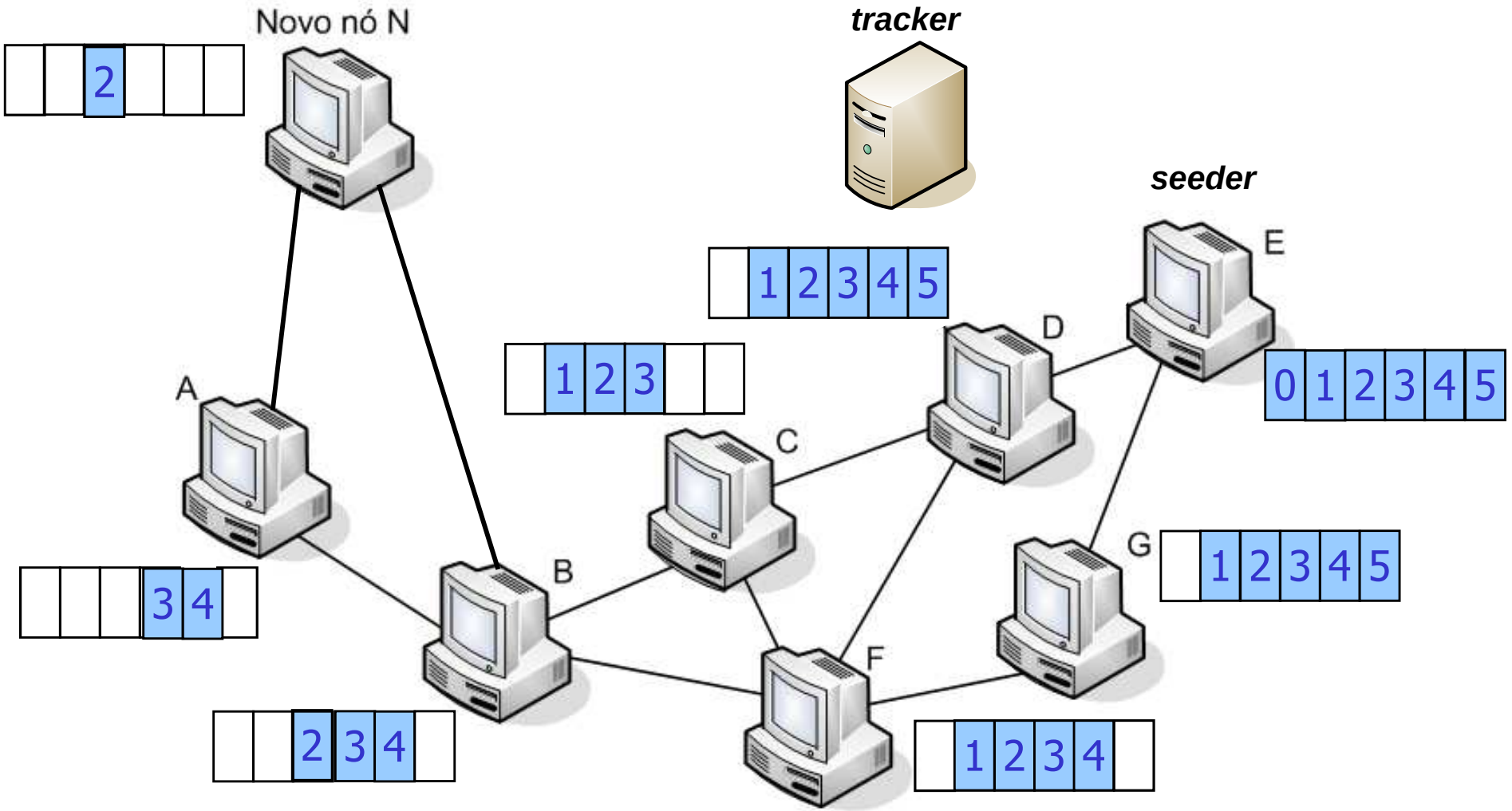
N solicita o pedaço 2 a B → **mais raro primeiro**

BiTorrent



B envia o pedaço para N

BiTorrent



O pedaço é armazenado no *buffer*

BitTorrent: Olho-por-Olho

Instituto de computação

- Cada para possui até 4 vizinhos
  - Pedaços são trocados **somente entre os vizinhos**

- Olho-por-olho (*tit-for-tat*)
  - Um par só deve continuar enviando pedaços para os vizinhos que mais enviam pedaços para ele

BitTorrent: Olho-por-Olho

Instituto de computação

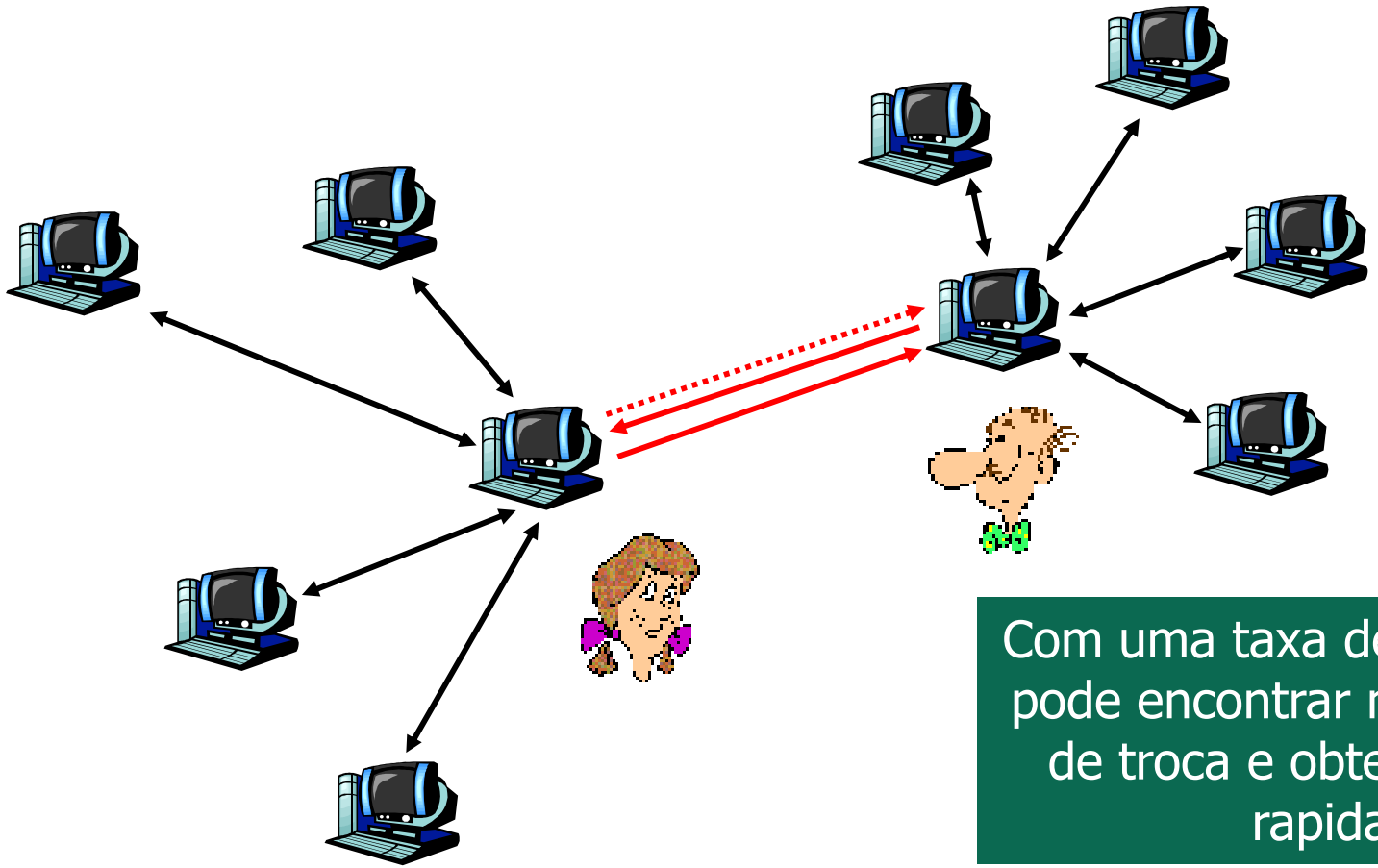
- Olho-por-olho (*tit-for-tat*)
  - Vizinhos são reavaliados a cada 10 s
    - De acordo com o número de pedaços recebidos
  - Vizinho que menos envia pedaços (pior) é sempre substituído a cada 30 s
  - Par seleciona **aleatoriamente** outro par e começa a enviar pedaços
    - **Seleção otimista** (*optimistically unchoke*)
      - Não sabe se o novo par será melhor que o atual pior

BitTorrent: Olho-por-Olho

(1) Alice seleciona Bob de forma otimista

(2) Alice se torna um dos quatro melhores provedores de Bob; Bob age da mesma forma

(3) Bob se torna um dos quatro melhores provedores de Alice



Com uma taxa de *upload* mais alta, pode encontrar melhores parceiros de troca e obter o arquivo mais rapidamente!

BitTorrent

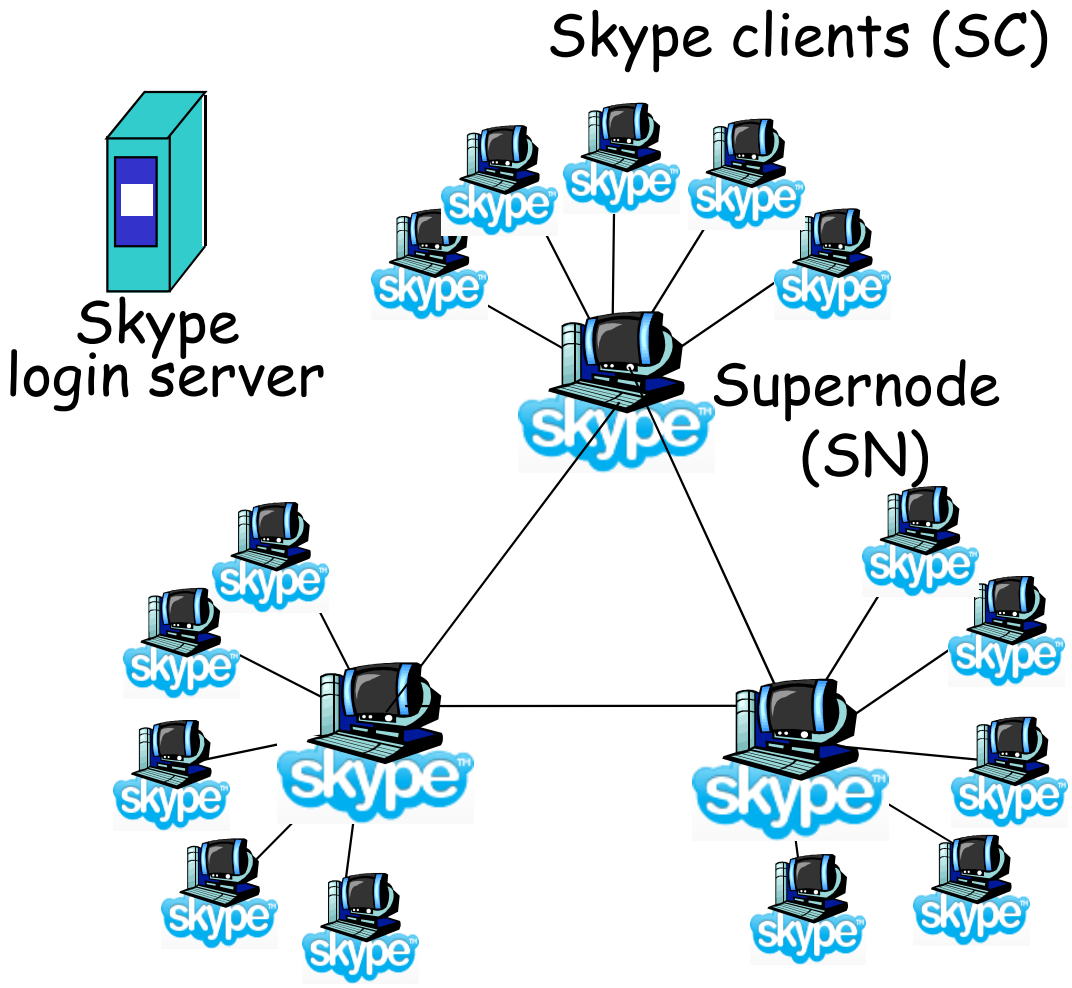
Instituto de computação

- Mas...
  - E se o novo vizinho enviar menos pedaços do que o atual pior?
  - E se o novo vizinho for um caroneiro (*free-rider*)?
    - Só recebe e não envia nenhum pedaço

- Mais um problema: pares podem entrar e sair do enxame a qualquer instante
  - Quando o par obtém todo o arquivo, ele pode (egoisticamente) sair ou permanecer (altruisticamente)

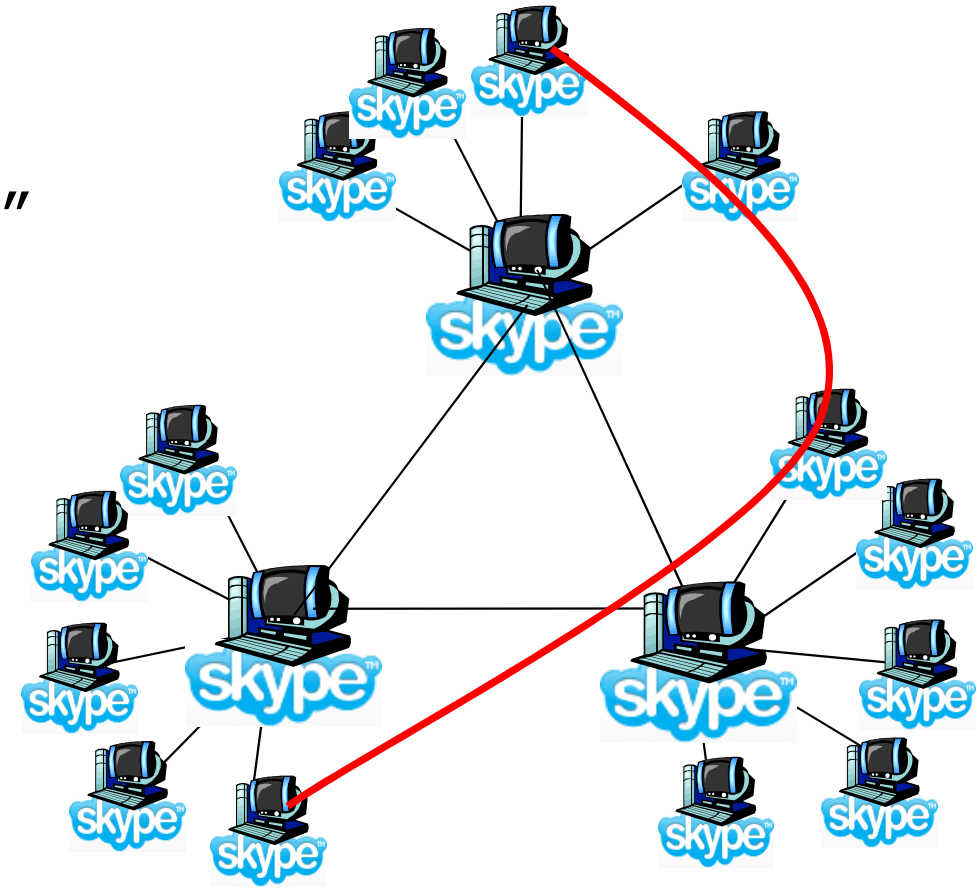
Skype

- Protocolo proprietário da camada de aplicação
  - Funcionamento estimado através de engenharia reversa
- Comunicação entre pares de usuários é P2P
- Overlay hierárquico com super-nós (SNs)
- Índice mapeia nomes dos usuários em endereços IP; distribuído através dos SNs



Skype

- Uso de intermediários (*relays*)
- Problema quando tanto Alice como Bob estão atrás de "NATs"
  - O NAT impede que um par externo inicie uma chamada com um par interno
- Solução
  - Intermediário é escolhido, usando os SNs de Alice e de Bob.
  - Cada par inicia sessão com o intermediário
  - Pares podem se comunicar através de NATs através do intermediário



Sistemas de Vídeo Par-a-Par

Instituto de computação

- Sucesso do compartilhamento de arquivos
  - Indicativo do potencial para distribuição de vídeo

Compartilhamento de arquivos	Distribuição de vídeo
Longas transferências sem restrições de tempo	Requisitos estritos de banda passante e tempo
Indexação e busca eficientes	Comunicação eficiente
Arquivos disponíveis a partir da publicação	Exibição durante um período de tempo

Sistemas de Vídeo Par-a-Par

Instituto de computação

- Usuários simultâneos
  - Característica da distribuição de vídeo
    - Audiência de um programa
  - Mais usuários → mais recursos compartilhados
  - É possível atender os requisitos das aplicações de vídeo

SopCast

Instituto de computação

- http://www.sopcast.com

PPLive

- http://www.synacast.com/en/

Arquiteturas de Distribuição

Instituto de computação

- Duas arquiteturas
  - Em árvore
  - Em malha

Arquiteturas de Distribuição

Instituto de computação

- Árvore
  - Uma ou múltiplas árvores
    - A fonte é a raiz
  - Relações de pai e filho
    - Um pai encaminha os dados somente para os filhos
  - Um participante deve se inscrever na árvore
    - Vídeo recebido sem novas requisições

Arquiteturas de Distribuição

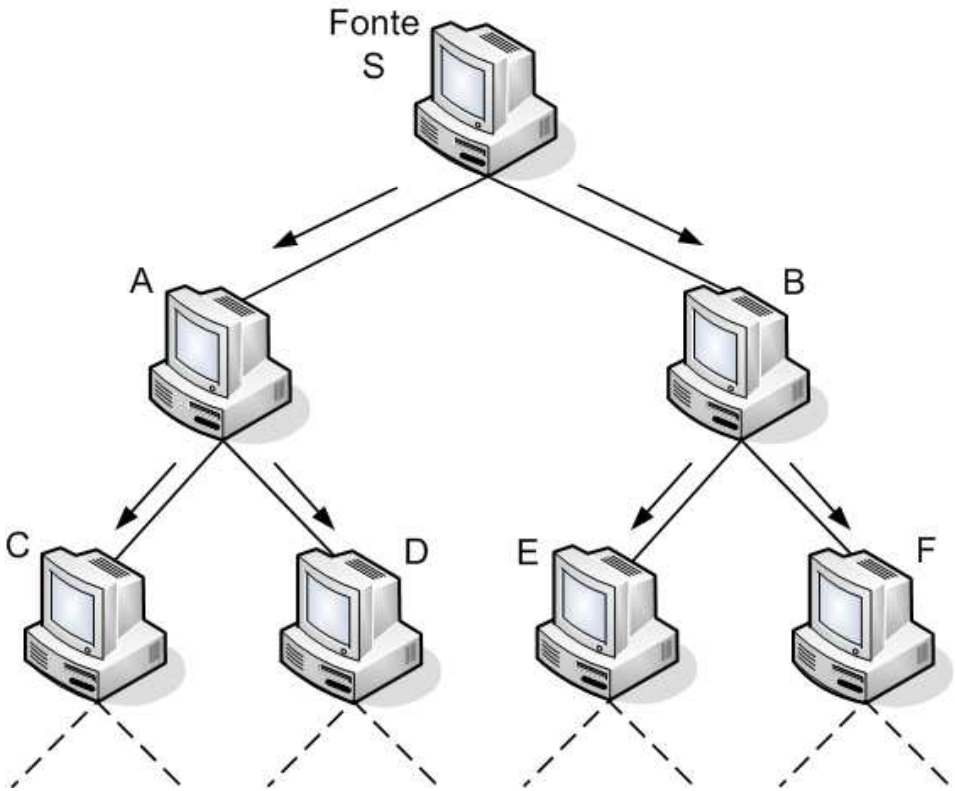
Instituto de computação

- Malha
  - Malha de distribuição
  - Participantes não possuem funções específicas
    - Receber e encaminhar para quaisquer nós
    - Sem uma organização hierárquica
  - Vídeo é dividido em pedaços (*chunks*)
    - Espalhados pelos participantes
    - Localização dos pedaços
    - Uma requisição por pedaço

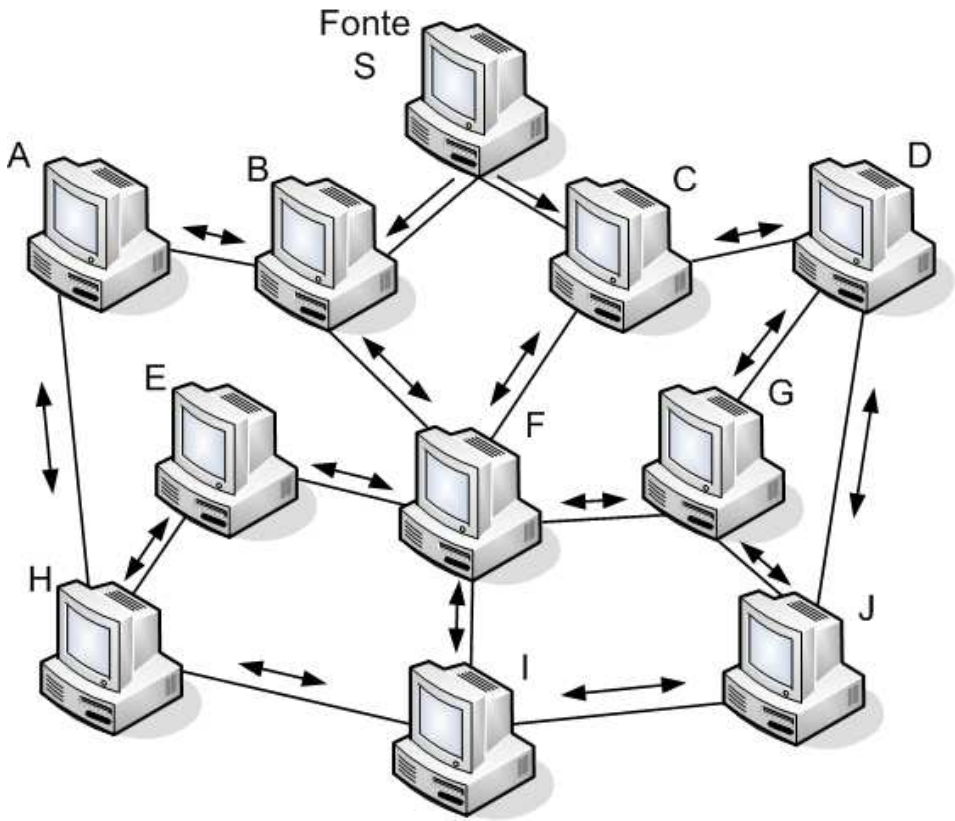
Arquiteturas de Distribuição

Instituto de computação

Árvore



Malha



Arquiteturas de Distribuição

Instituto de computação

Árvore	Malha
Hierarquia no encaminhamento	Sem hierarquia no encaminhamento
Uma requisição à fonte	Uma requisição a cada pedaço
Aumentar a eficiência do encaminhamento	Aumentar a robustez à dinâmica dos participantes

Arquiteturas de Distribuição

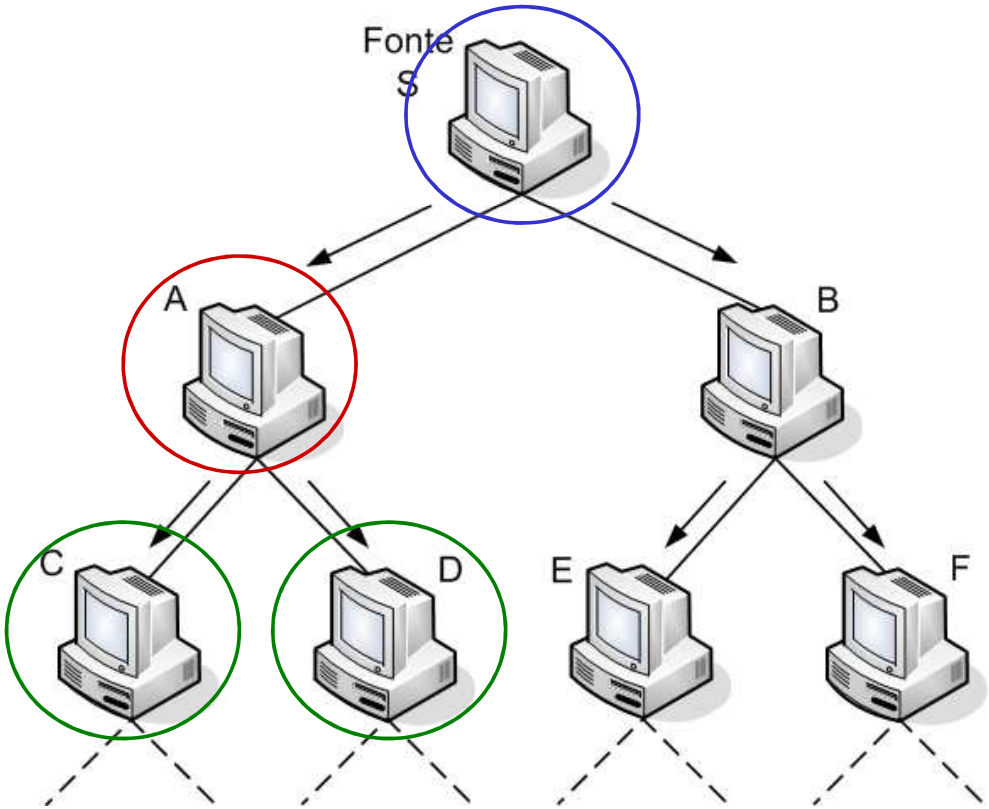
Instituto de Computação

Árvore	Malha
Hierarquia no encaminhamento	**Sem hierarquia no encaminhamento**
Uma requisição à fonte	Uma requisição a cada pedaço
Aumentar a eficiência do encaminhamento	Aumentar a robustez à dinâmica dos participantes

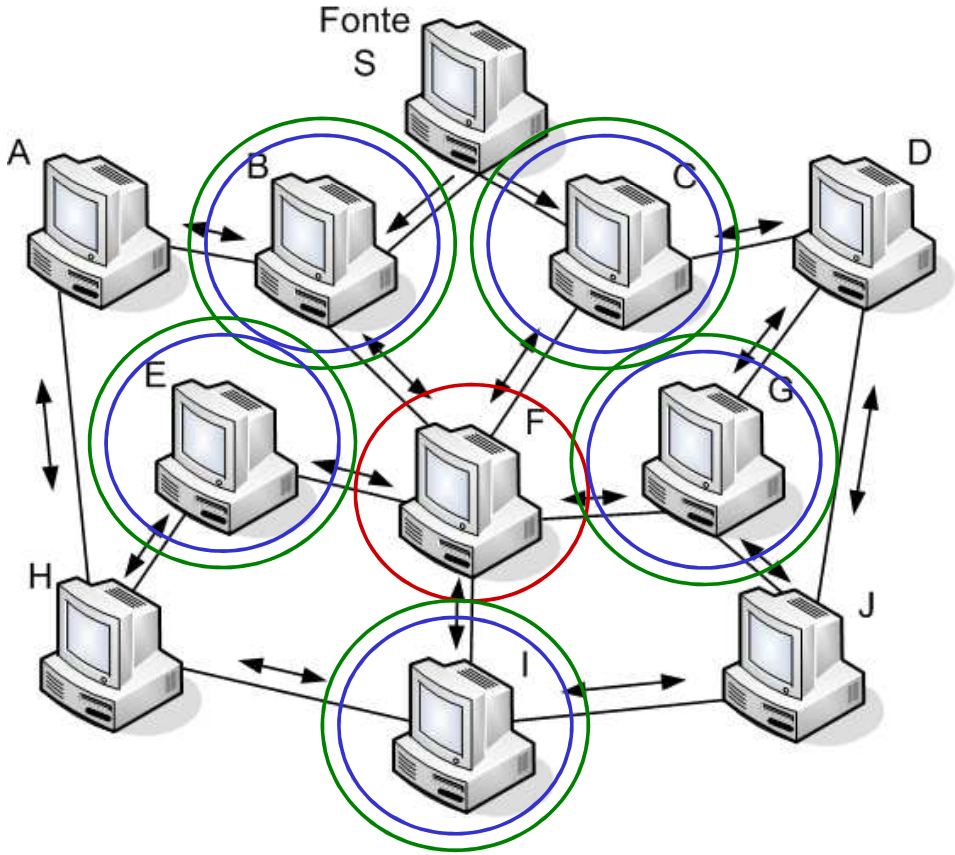
Arquiteturas de Distribuição

Instituto de computação

Árvore



Malha



Arquiteturas de Distribuição

Instituto de Computação

Árvore	Malha
Hierarquia no encaminhamento	Sem hierarquia no encaminhamento
Uma requisição à fonte	**Uma requisição a cada pedaço**
Aumentar a eficiência do encaminhamento	Aumentar a robustez à dinâmica dos participantes

Arquiteturas de Distribuição

Instituto de Computação

Árvore	Malha
Hierarquia no encaminhamento	Sem hierarquia no encaminhamento
Uma requisição à fonte	Uma requisição a cada pedaço
Aumentar a eficiência do encaminhamento	**Aumentar a robustez à dinâmica dos participantes**

Desafios

Instituto de computação

- Segurança
- Incentivos à Cooperação

Instituto de computação

Programa de Pós-Graduação
Instituto de Computação
Universidade Federal Fluminense

Aula 10

Camada de Aplicação

DNS e sistemas par-a-par

Igor Monteiro Moraes

Redes de Computadores