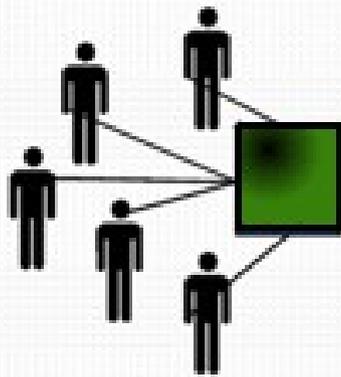


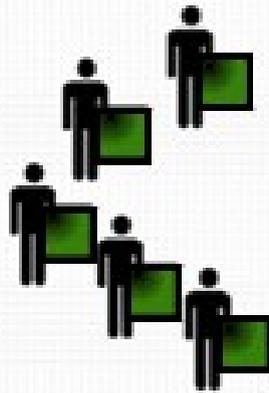
Disciplina

Sistemas de Computação

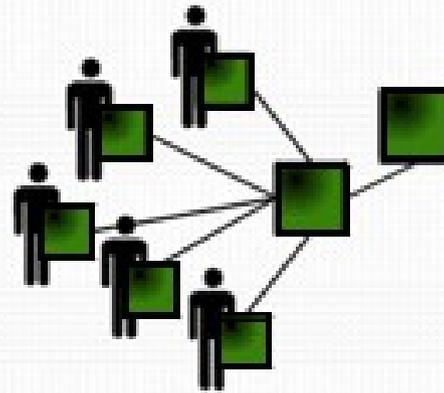
Aula 16



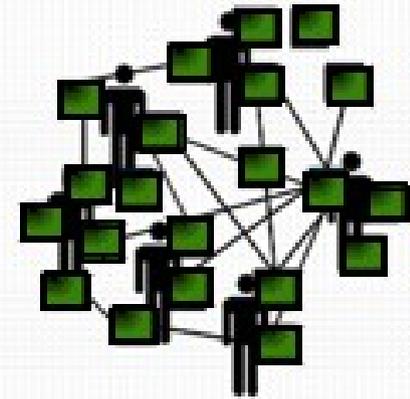
1950 : Mainframe



1980: Micro computer



1990: Internet



200? Diffuse IT

Organização da Aula

- Nomeação
 - Nomeação Simples
 - Nomeação Estruturada

Nomeação

- Para que servem os nomes?
 - Compartilhar recursos
 - Identificar entidades de maneira única
 - Fazer referência a localizações
- Nome deve ser resolvido para a entidade à qual se refere
 - Sistema de Nomeação (ex. DNS)

Nomeação

- Em um SD, a implementação de um sistema de nomeação costuma ser distribuída por várias máquinas
 - Modo como é feita a distribuição desempenha papel fundamental na eficiência e escalabilidade do sistema

Nomes, Identificadores e Endereços

- **Entidades:** Máquinas, impressoras, discos, processos, usuários, páginas Web, janelas gráficas, mensagens, etc.
- São acessadas através de um **ponto de acesso**, ou simplesmente, **endereço**
 - Ex: **Servidor e seu número IP**
- Portanto, um endereço pode ser utilizado como uma maneira de nomear, identificar uma entidade
 - **Problema:** Entidade pode mudar facilmente de ponto de acesso!
- Ex: **Servidor Web alocado em outra rede**

Nomes, Identificadores e Endereços

Como nomear entidades, sem utilizar especificamente seu endereço, ou seja, nomeá-las independentemente da sua posição física (localização)?

Identificadores ou Nomes amigáveis a seres humanos

Identificadores

- Em muitos casos, são cadeias aleatórias de bits, com as seguintes propriedades:
 - Um identificador referencia, no máximo, UMA entidade
 - Cada entidade é referenciada por, no máximo, um identificador
 - Um identificador sempre referencia a mesma entidade, isto é, nunca é reutilizado
- Ex: Identificadores de entidades em sistemas P2P baseados no sistema Chord

Nomes amigáveis

- Nomes representados por uma cadeia de caracteres
 - Pathnames, domínios na Internet, números de processos
 - `/etc/passwd`
 - `http://www.ic.uff.br`

Como resolvemos nomes e identificadores para endereços?

Sistemas de Nomeação

Sistemas de Nomeação

- Mantém uma vinculação **nome-endereço**
- Na forma mais simples
 - Tabela de pares (nome,endereço)
 - Contudo, sistemas que abrangem redes de grande porte, uma tabela centralizada não vai funcionar
- Em SDs, usualmente o nome é decomposto em várias partes, e a resolução é realizada por meio de consultas recursivas

Sistemas de Nomeação

- Três Classes
 - Nomeação Simples
 - Nomeação Estruturada
 - Nomeação Baseada em Atributo

Nomeação Simples

- Aplicada a identificadores
 - Cadeias aleatórias de bits → nomes simples
 - Não contém sequer uma informação sobre como localizar o ponto de acesso de uma entidade associada

Nomeação Simples

- Problema: Dado um identificador, como localizar o ponto de acesso (endereço)?
 - Soluções Simples: broadcasting
 - Tabelas de Hash Distribuídas (DHT)

Nomeação Simples: Broadcasting e Multicasting

- Consideradas soluções simples
- Aplicáveis somente a redes locais

Nomeação Simples: Broadcasting

- Recursos oferecidos por redes locais nas quais todas as máquinas estão conectadas a um único cabo ou seu equivalente lógico
- Como funciona?
 - Mensagem que contém o identificador da entidade é enviada a todas as máquinas da rede
 - Cada uma das máquinas verifica se tem esta entidade
 - Máquinas com ponto de acesso para a entidade, enviam uma msg que contém o endereço

Nomeação Simples: Broadcasting

- Se torna ineficiente quando a rede cresce
 - Largura de banda da rede é desperdiçada, com grande número de mensagens de requisição
 - Aumento da probabilidade de colisões de mensagens, diminuindo o throughput do sistema
 - Grande número de máquinas pode ser interrompido por requisições que não podem responder

Nomeação Simples: Multicasting

- Somente um grupo restrito de máquinas recebe a requisição
- Banco de Dados Replicado
 - Endereço multicast é associado a uma entidade replicada
 - Multicasting é usado para localizar a réplica mais próxima
 - Requisição para o endereço multicast, cada réplica responde com seu endereço IP
 - Réplica mais próxima → aquela cuja resposta chega antes

Tabelas de Hash Distribuídas (DHT)

- Exemplo: Nós são organizados logicamente em um anel (Chord)
 - Usa um espaço de identificadores de m bits para designar nós e entidades específicas (arquivos, processos)
 - Número m bits é usualmente 128 ou 160
 - Entidade com chave k cai sob a jurisdição do nó que tenha o menor identificador $id \geq k \rightarrow \text{succ}(k)$

Tabelas de Hash Distribuídas (DHT)

- Como resolver com eficiência uma chave k para o endereço de $\text{succ}(k)$?
 - Abordagem linear
 - Tabela de Derivação

Tabelas de Hash Distribuídas (DHT)

- Abordagem linear
 - Cada nó p monitora o sucessor $\text{succ}(p+1)$ e o predecessor $\text{pred}(p)$
 - Ao receber uma requisição para a chave k , p repassa a requisição para os seus vizinhos, a menos que $\text{pred}(p) < k \leq p \rightarrow p$ retorna o próprio endereço
 - Não escalável!

Tabelas de Hash Distribuídas (DHT)

- Tabela de derivação (finger table)
 - Possui, no máximo, m entradas
 - Denotando a tabela de derivação de p por Ft_p

$$Ft_p[i] = \text{succ}(p + 2^{i-1})$$

i -ésima entrada aponta para o primeiro nó que sucede p
por no mínimo 2^{i-1}

$M=5, p=1:$

$l=1 \Rightarrow 2$

$l=2 \Rightarrow 3$

$l=3 \Rightarrow 5$

$l=4 \Rightarrow 9$

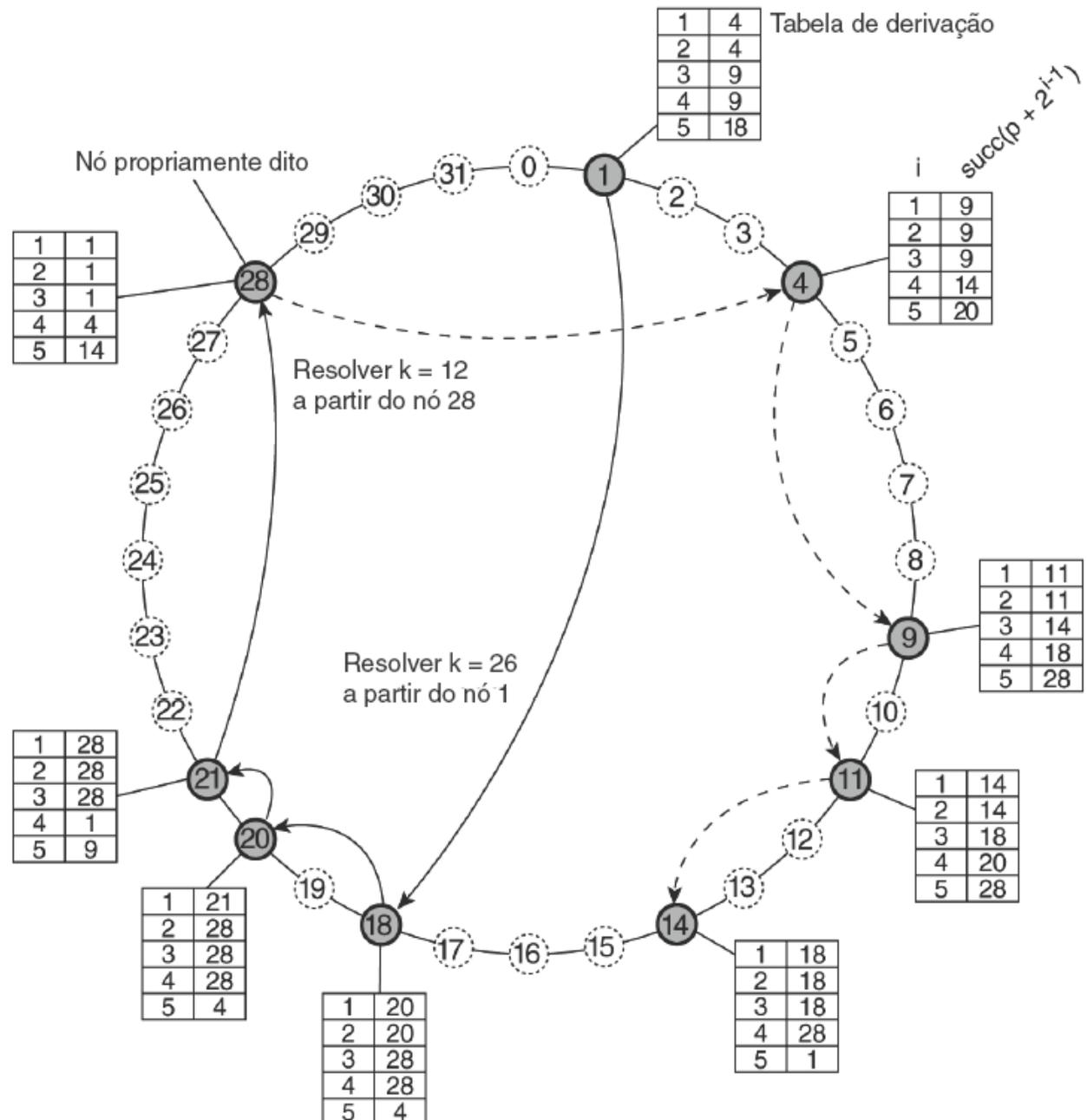
$l=5 \Rightarrow 17$

Tabelas de Hash Distribuídas (DHT)

a) Suponhamos que $p = 4$ receba uma requisição para $k = 7 \rightarrow \text{succ}(p+1) \rightarrow$ repassa a requisição ao nó $= 9$

b) Suponhamos que $p = 4$ receba uma requisição para $k = 3 \rightarrow$ como $\text{pred}(4) = 1 < 3 \leq 4 \rightarrow$ retorna o próprio endereço

$M=5, p=1:$
 $l=1 \Rightarrow 2$
 $l=2 \Rightarrow 3$
 $l=3 \Rightarrow 5$
 $l=4 \Rightarrow 9$
 $l=5 \Rightarrow 17$



Tabelas de Hash Distribuídas (DHT)

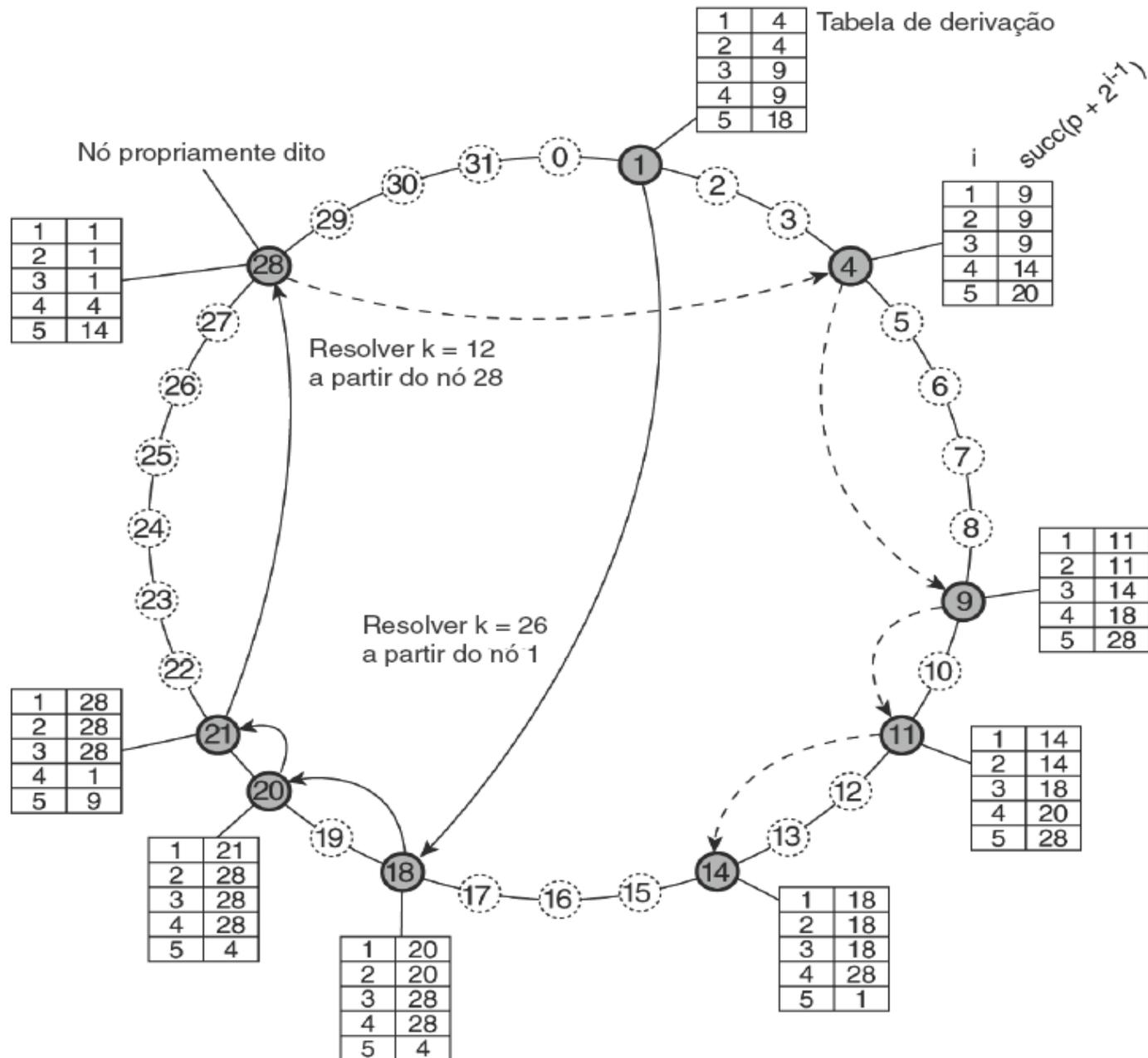
- Como encontrar uma entidade k ?
 - Referências na tabela de derivação são atalhos para nós existentes no espaço de identificadores
 - Distância do atalho em relação ao nó p aumenta exponencialmente à medida que o índice na tabela de derivação cresce
 - Para consultar uma chave k , o nó p repassará a requisição ao nó q com índice j na tabela de derivação de p

$$q = FT_p[j] < k \leq FT_p[j+1]$$

Tabelas de Hash Distribuídas (DHT)

- 1) Considere a resolução de $k=12$, a partir do nó 28
- 2) Nó 28 consultará $k=12 \rightarrow$ verifica que $FT[4] < k \leq FT[5]$
- 3) Requisição será repassada para o nó 4
- 4) O nó 4 selecionará o nó 9, porque $FT[3] < k \leq FT[4]$
- 5) O nó 9 selecionará o nó 11
- 6) O nó 11 repassará o pedido ao nó 14

Consulta $\rightarrow O(\log(N))$ passos



Exercício:

Tabelas de Hash Distribuídas

1) Crie as Tabelas de Derivação, considerando $M=4$

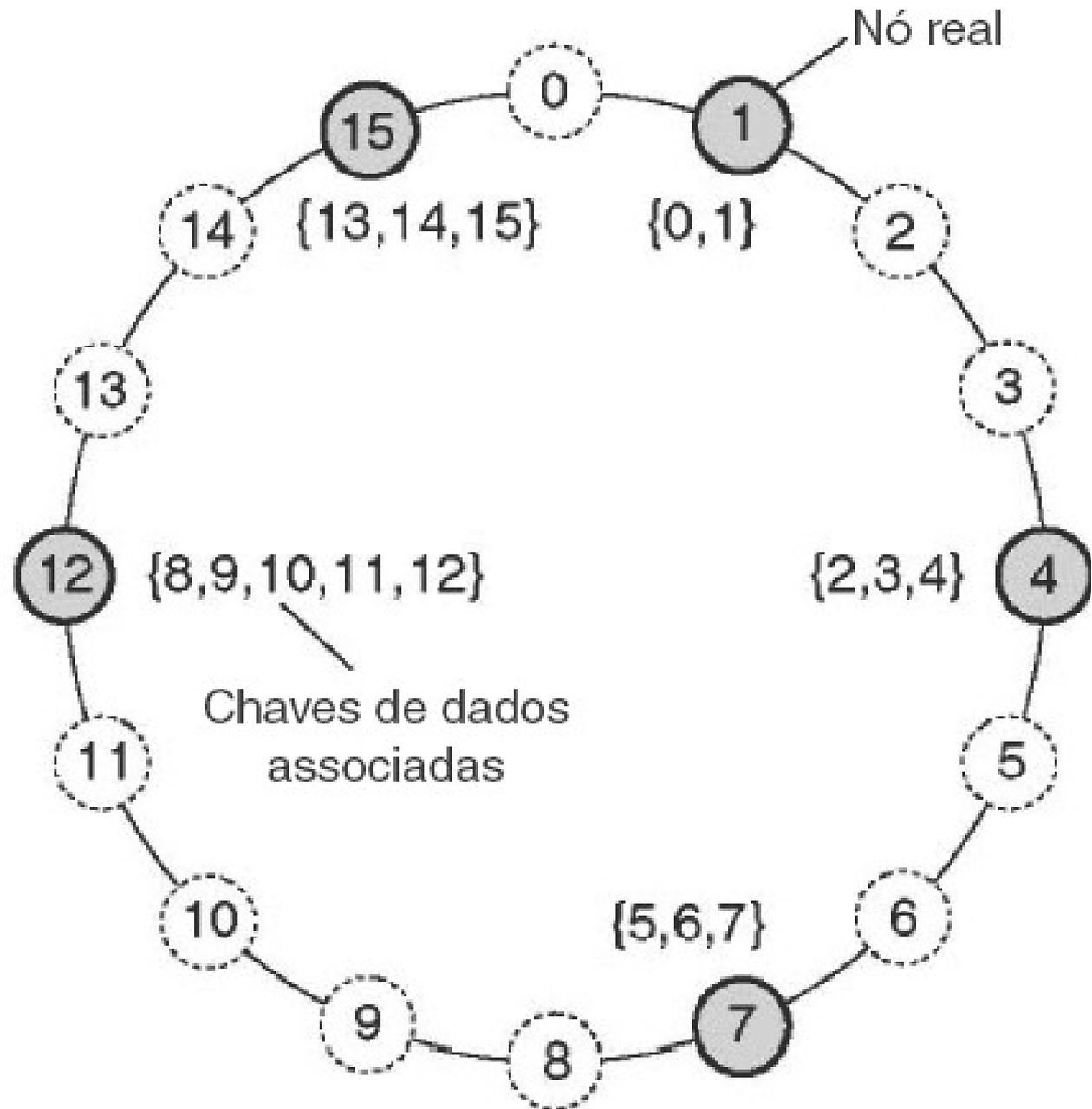
$$Ft_p[i] = \text{succ}(p + 2^{i-1})$$

2) Como seria a resolução do $K=12$ a partir do nó 1?

3) Considere a chegada do nó 10

4) Construa a TD para o nó 10

5) A chegada do nó 10 ocasionaria alguma alteração nas demais Tds? Quais?



Nomeação Estruturada

- Nomes simples são bons para máquinas, mas não são convenientes para a utilização de seres humanos
- Sistemas de nomeação comumente suportam nomes estruturados
 - Exemplo: Nomeação de arquivos, hosts na Internet

Nomeação Estruturada

Resolução de nomes

- Espaços de nomes oferecem um mecanismo para armazenar e recuperar informações sobre entidades por meio de nomes
- Dado um nome de caminho, deve ser possível consultar qualquer informação armazenada no nó referenciado por aquele nome
- **Problema:** Para resolver um nome, precisamos de um nó de diretório.
 - Como escolher este nó inicial?

Nomeação Estruturada

Resolução de nomes

- São implícitos ao contexto em que a resolução de nomes está se aplicando
 - www.cs.vu.nl: início da resolução é feito através do servidor de nome DNS (raiz)
 - [/home/steen/mbox](#): início da resolução ocorre no servidor local NFS

Como resolver nomes?

- Resolução Iterativa
 - Servidor responde somente o que sabe, o nome do próximo servidor que deve ser buscado
 - Cliente procura iterativamente os outros servidores
- Resolução Recursiva
 - Servidor passa a busca para o próximo servidor que pode encontrar
 - Para o cliente, somente existe uma mensagem de retorno: o endereço do nome ou 'não encontrado'

Resolução de nomes: Interativa

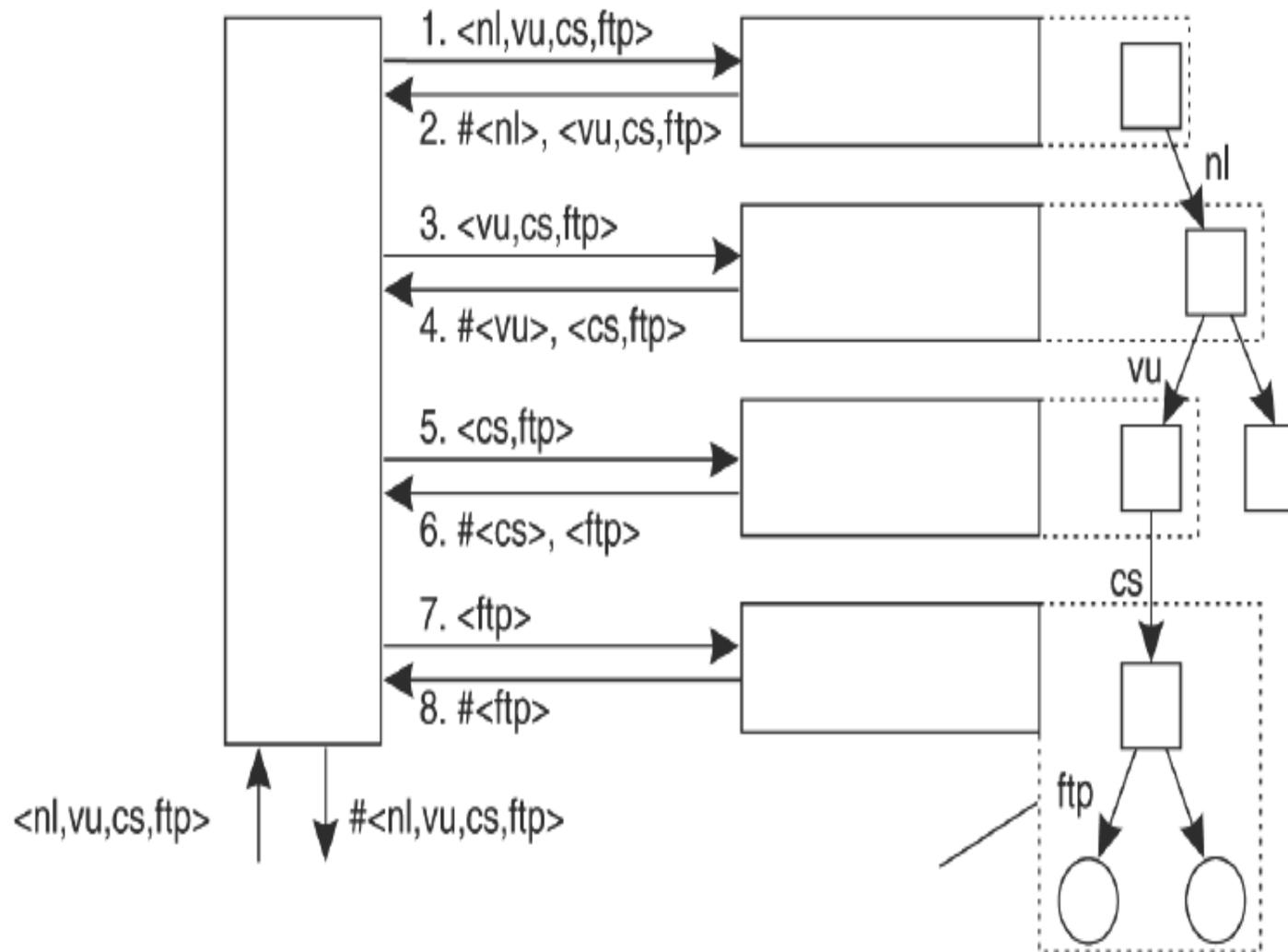


Figura 5.14 Princípio da resolução iterativa de nomes.

Resolução de nomes: Recursiva

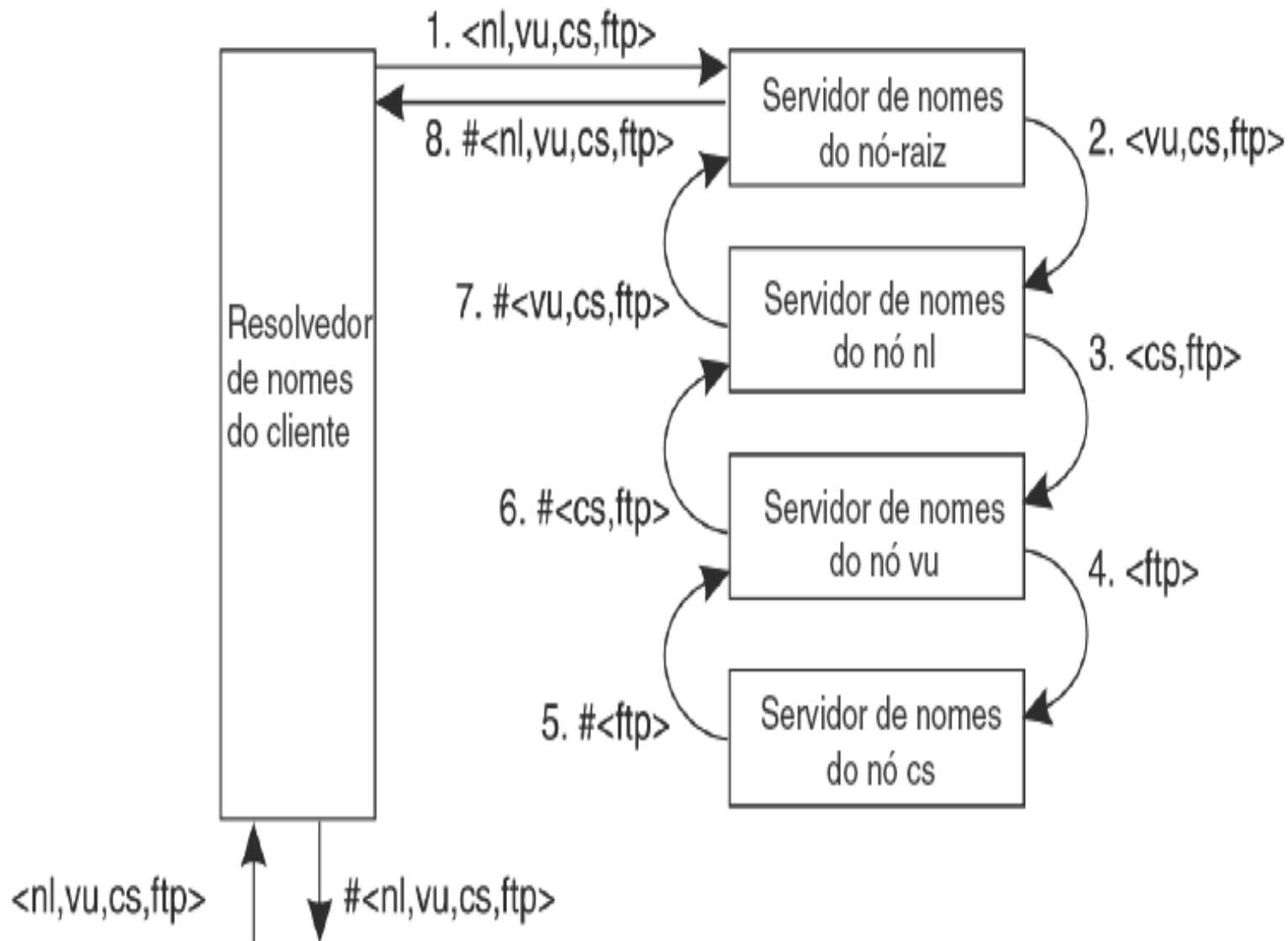


Figura 5.15 Princípio da resolução recursiva de nomes.

Resolução de nomes: Recursiva vs Iterativa

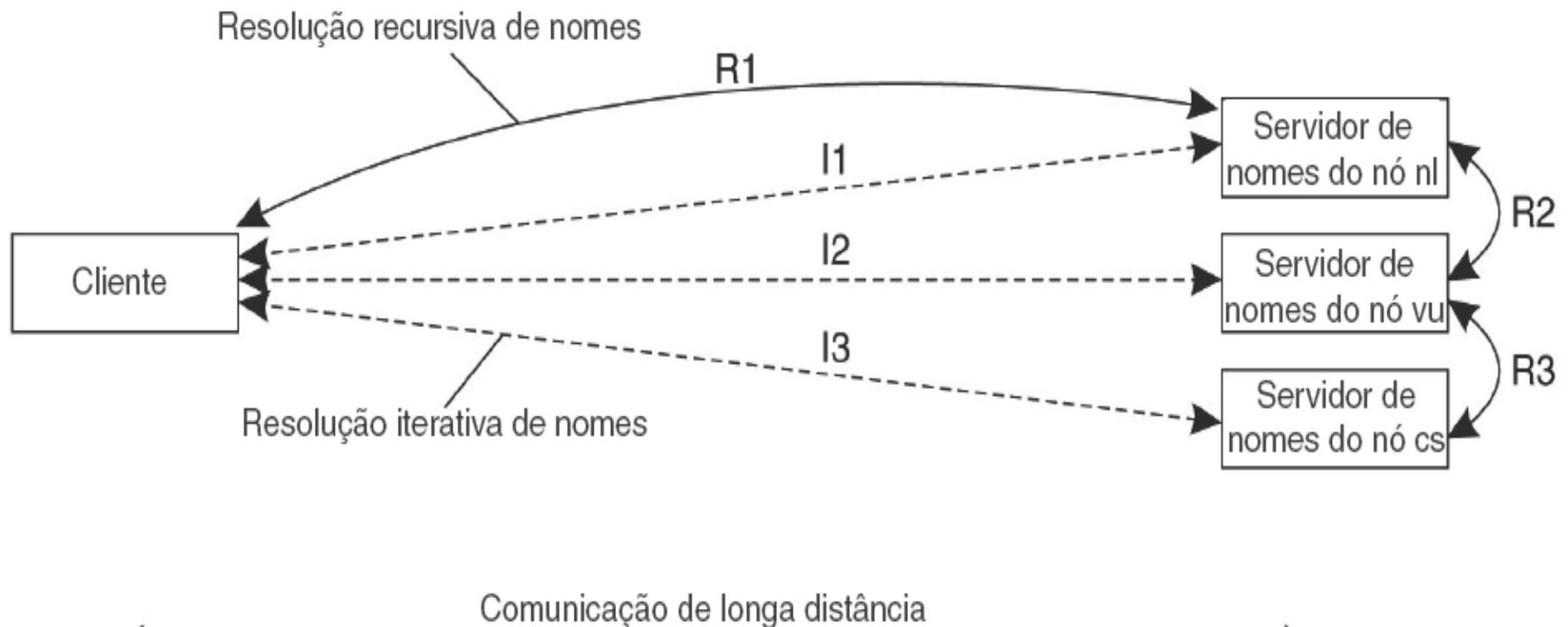


Figura 5.16 Comparação entre resolução recursiva e iterativa de nomes no que diz respeito aos custos de comunicação.

Nomeação Baseada em Atributo

- Fornecer uma descrição da entidade que está sendo procurada
- Descrever uma entidade em termos de pares <atributo, valor>
- Serviços de diretório retornam uma ou mais entidades que atendam a descrição do usuário