

Sistemas Peer-to-Peer

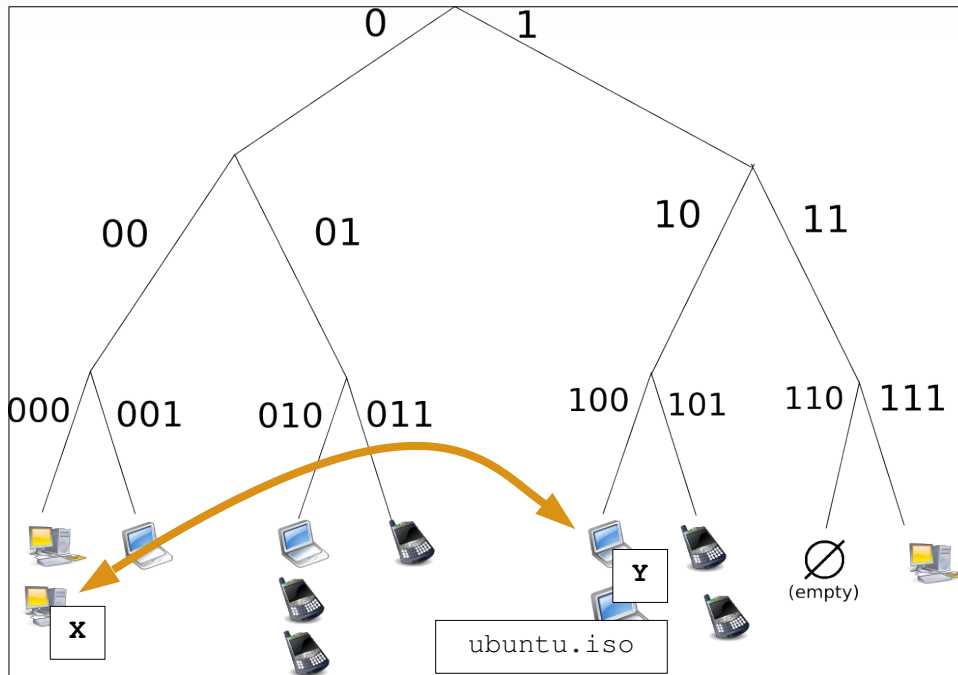
4. Topologias Estruturadas

Francisco Sant'Anna
francisco@ime.uerj.br



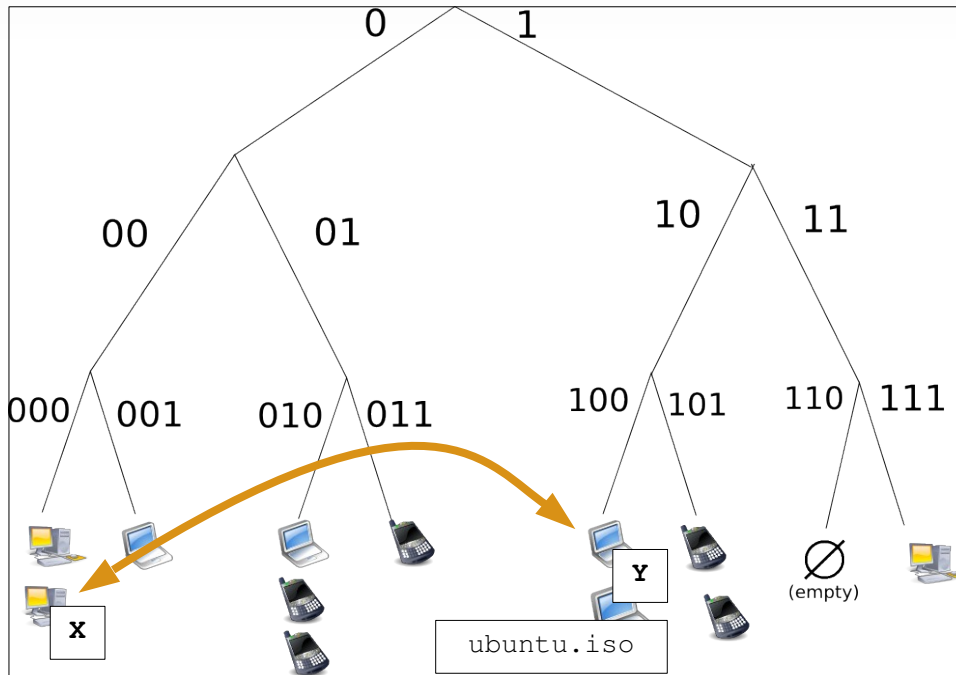
Topologia Estruturada

- Como encontrar o IP de Y partindo de X?
- Só tenho meus vizinhos.
- Como rotear entre X e Y?



Topologia Estruturada

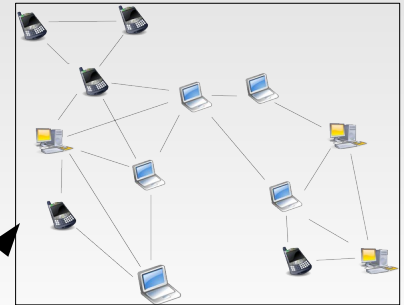
- Como encontrar o IP de Y partindo de X?
- Só tenho meus vizinhos.
- Como rotear entre X e Y?



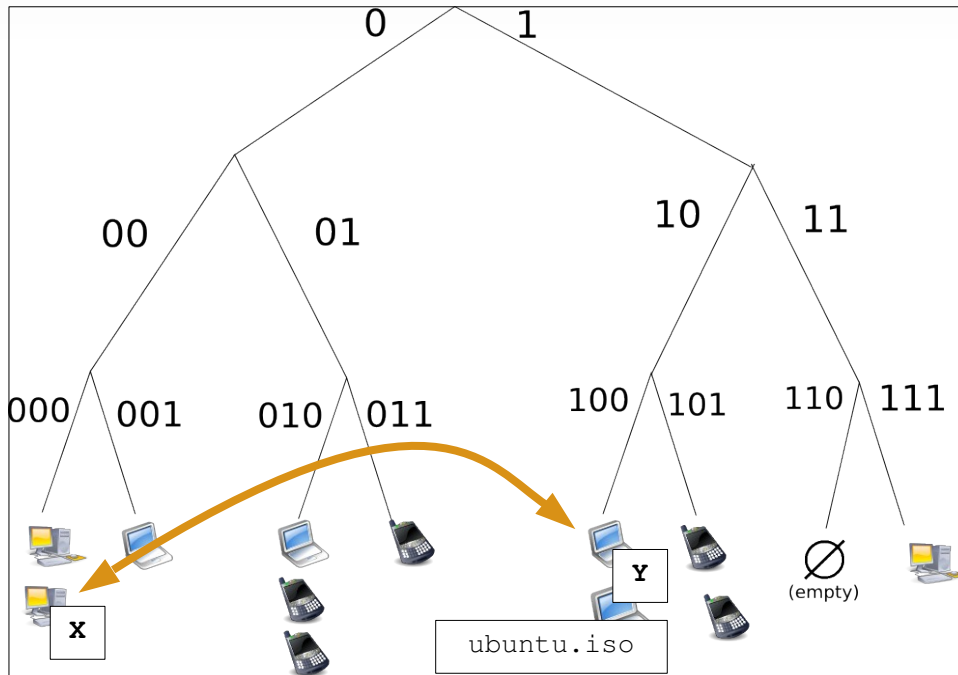
- Quem são meus vizinhos?

Topologia Estruturada

- Como encontrar o IP de Y partindo de X?
- Só tenho meus vizinhos.
- Como rotear entre X e Y?

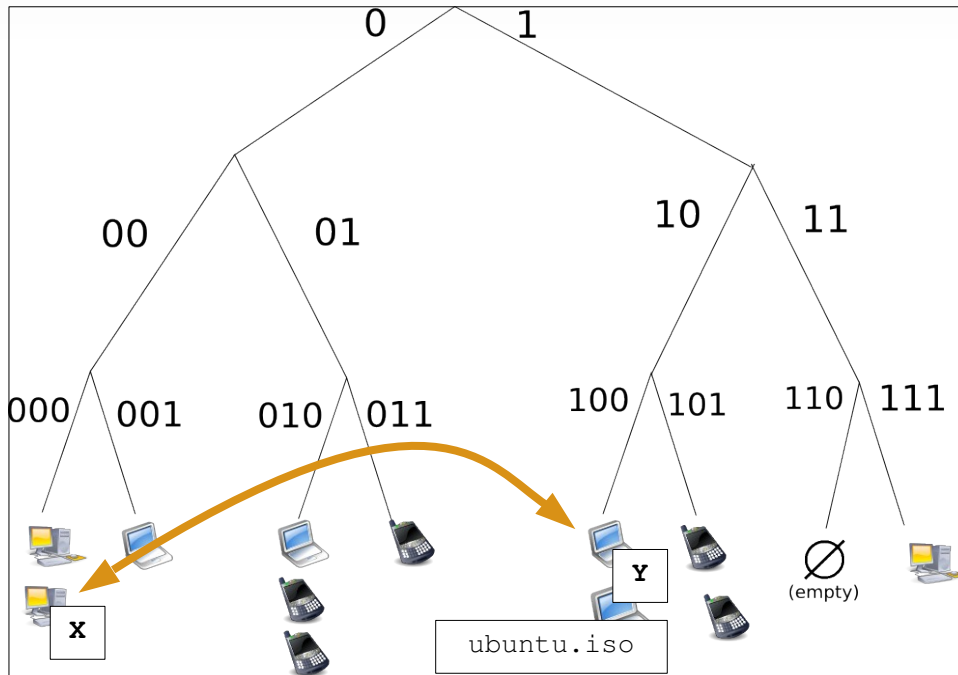
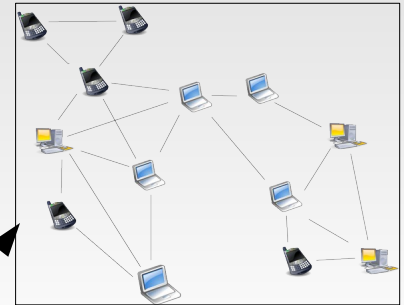


- Quem são meus vizinhos?



Topologia Estruturada

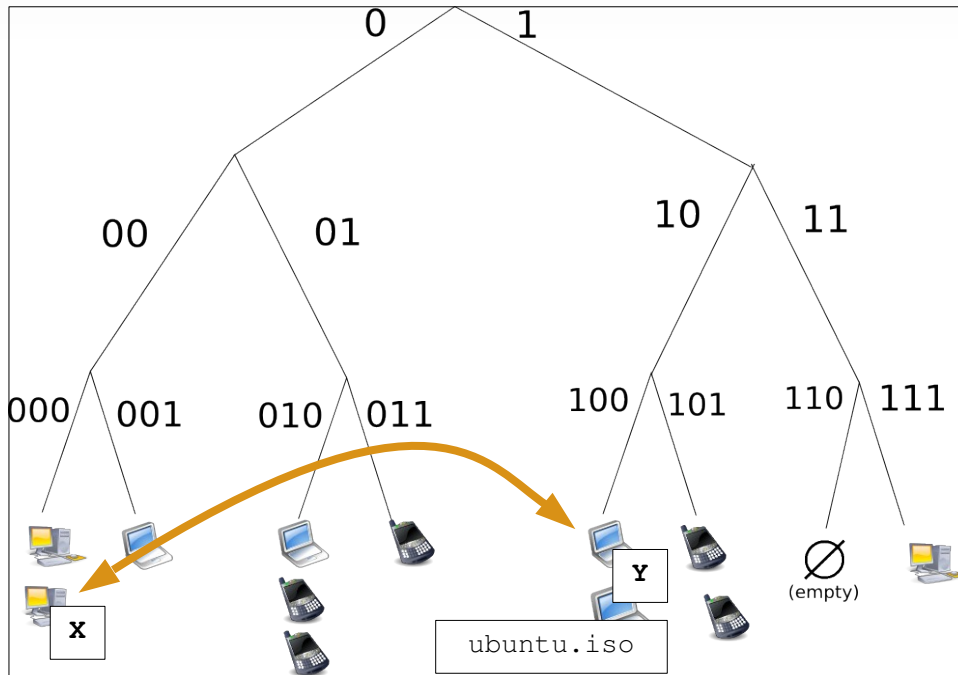
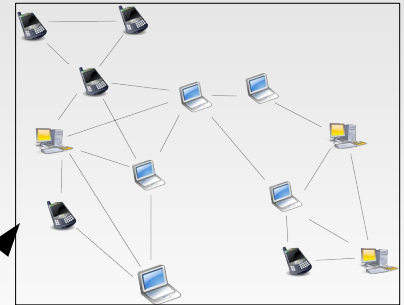
- Como encontrar o IP de Y partindo de X?
- Só tenho meus vizinhos.
- Como rotear entre X e Y?



- Quem são meus vizinhos?
- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`

Topologia Estruturada

- Como encontrar o IP de Y partindo de X?
- Só tenho meus vizinhos.
- Como rotear entre X e Y?



- Quem são meus vizinhos?
- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
- Como dados e máquinas são identificados?

Topologia Estruturada

Topologia Estruturada

- Qual máquina armazena qual dado?

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`
 - **Dado *10321* é armazenado pela máquina *10321***
 - (ou a mais próxima)
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?
 - `HASH(ubuntu.iso) = 1531 [0..N[`
 - `HASH(MAC-ADDR) = 10321 [0..N[`

Topologia Estruturada

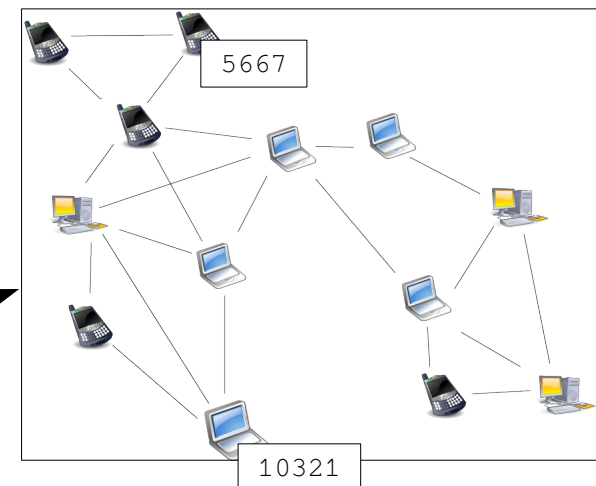
- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`
 - **Dado 10321 é armazenado pela máquina 10321**
 - (ou a mais próxima)
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?
 - `HASH(ubuntu.iso) = 1531 [0..N[`
 - `HASH(MAC-ADDR) = 10321 [0..N[`
- **Quem são meus vizinhos na rede overlay?**

Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`
 - **Dado 10321 é armazenado pela máquina 10321**
 - (ou a mais próxima)
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?
 - `HASH(ubuntu.iso) = 1531 [0..N[`
 - `HASH(MAC-ADDR) = 10321 [0..N[`
- **Quem são meus vizinhos na rede overlay?**
 - Sou a máquina 5667 e quero o dado 10321

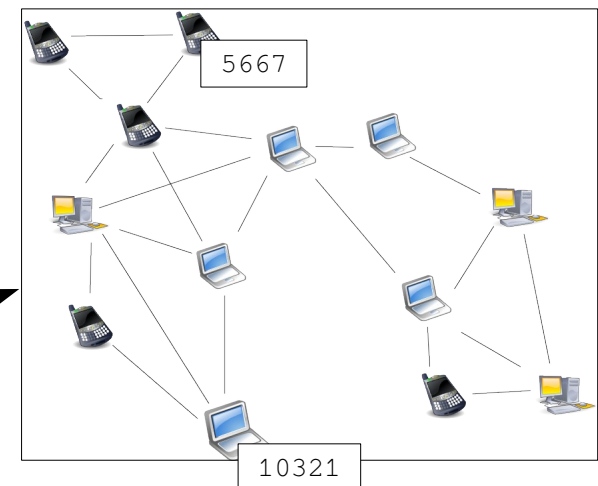
Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`
 - **Dado 10321 é armazenado pela máquina 10321**
 - (ou a mais próxima)
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?
 - `HASH(ubuntu.iso) = 1531 [0..N[`
 - `HASH(MAC-ADDR) = 10321 [0..N[`
- **Quem são meus vizinhos na rede overlay?**
 - Sou a máquina 5667 e quero o dado 10321



Topologia Estruturada

- Qual máquina armazena qual dado?
 - `LOCATE(ubuntu.iso) = Y`
 - A resposta tem que ser a mesma em qualquer peer
 - A resposta tem que ser local (não centralizada)
 - Idealmente:
 - `LOCATE(ubuntu.iso) = 155.12.1.20`
 - **Dado 10321 é armazenado pela máquina 10321**
 - (ou a mais próxima)
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?
 - `HASH(ubuntu.iso) = 1531 [0..N[`
 - `HASH(MAC-ADDR) = 10321 [0..N[`
- **Quem são meus vizinhos na rede overlay?**
 - Sou a máquina 5667 e quero o dado 10321
 - Como faço o download do arquivo?

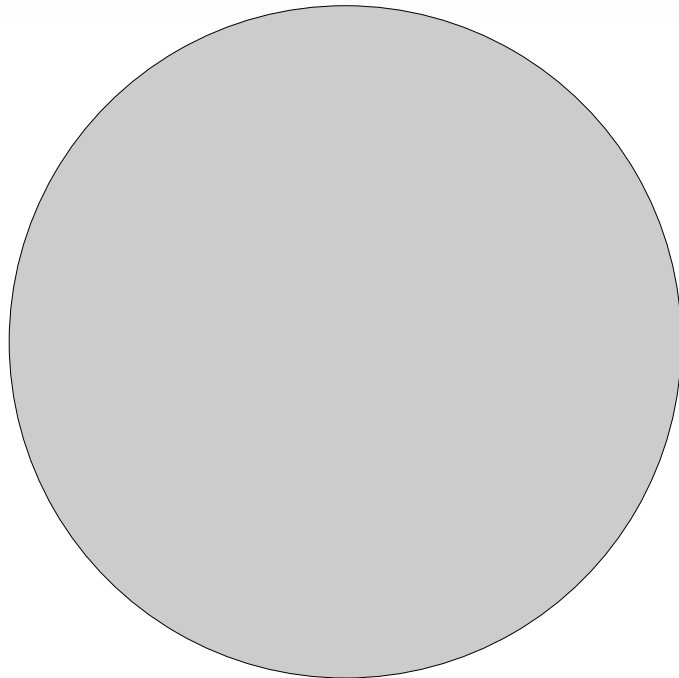


Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?

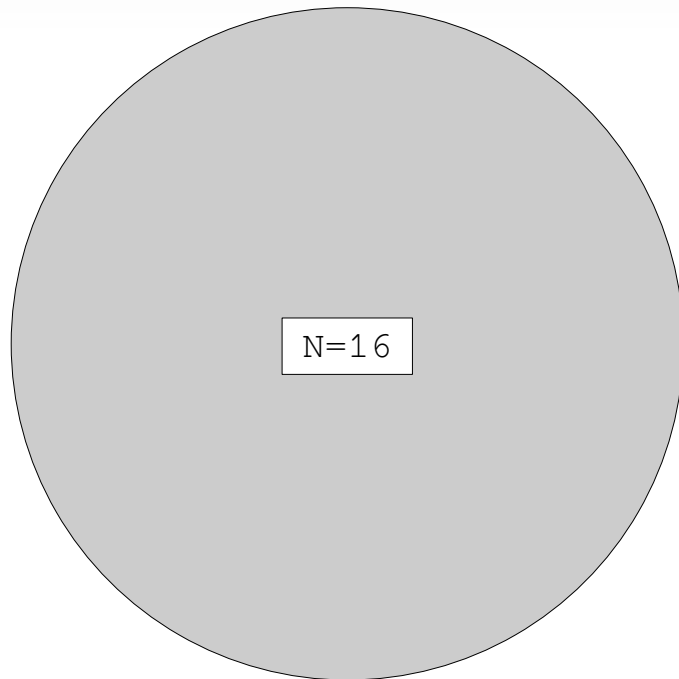
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



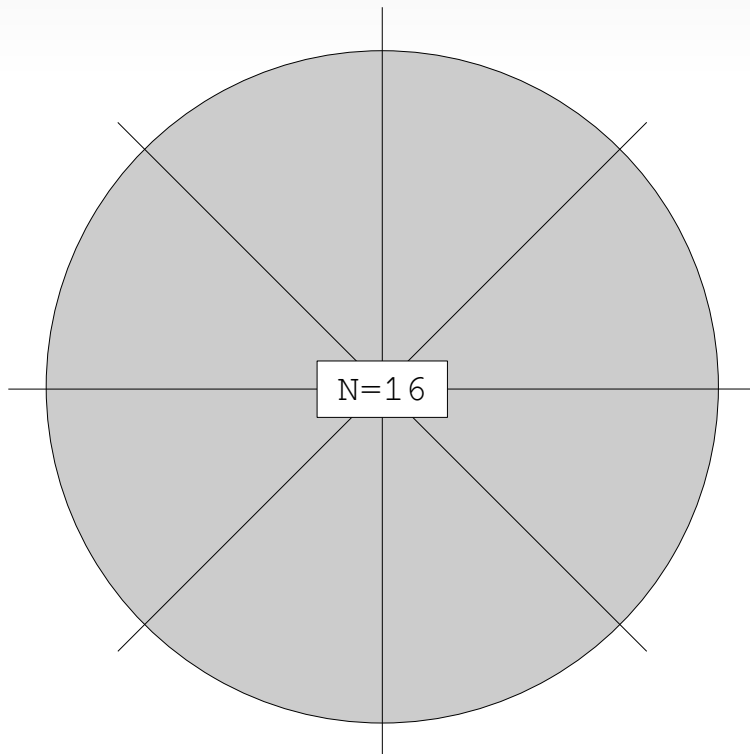
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



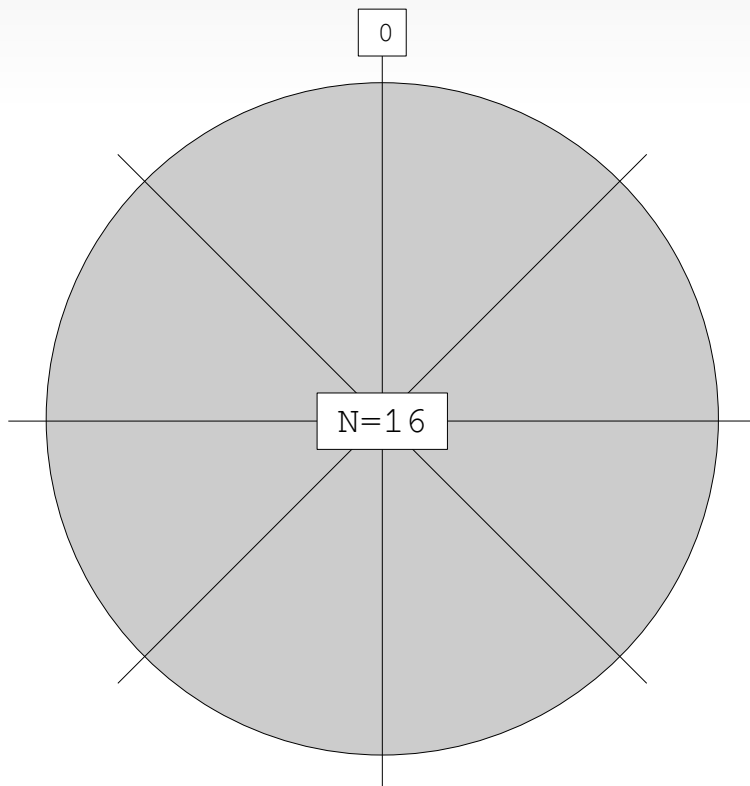
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



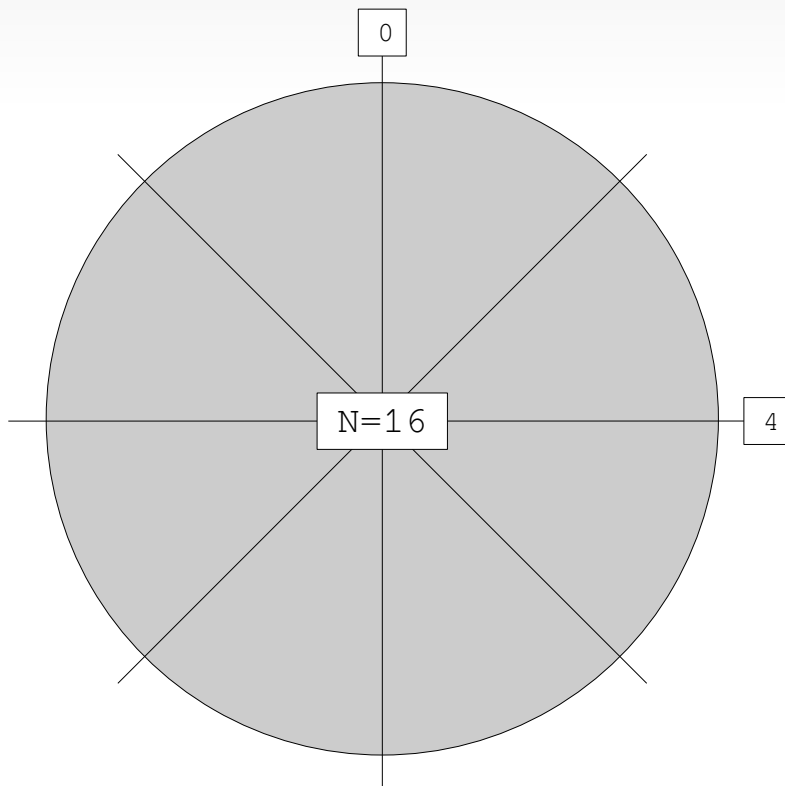
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



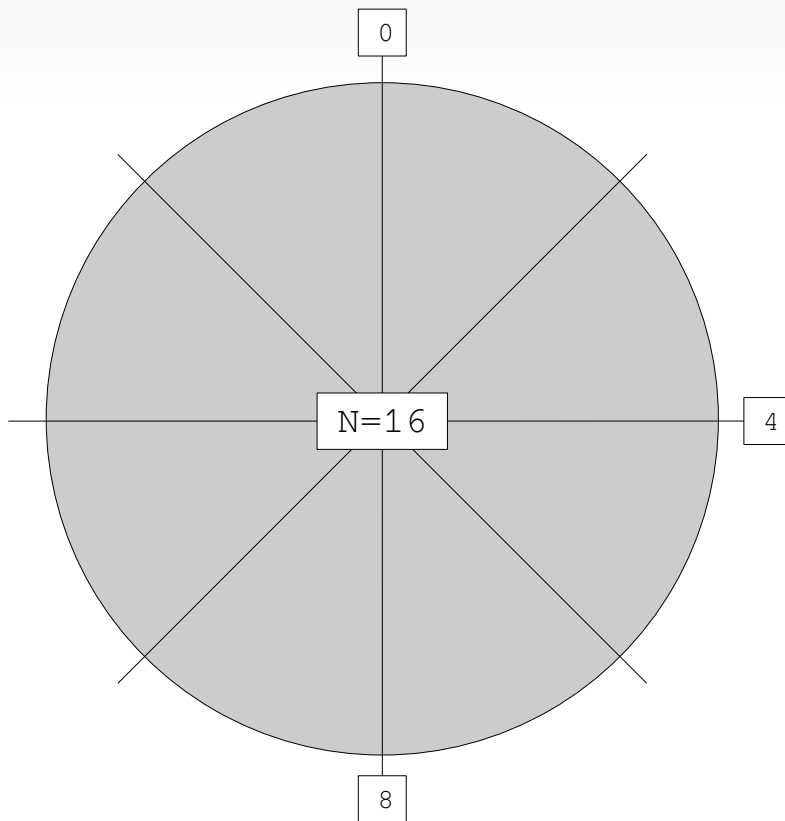
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



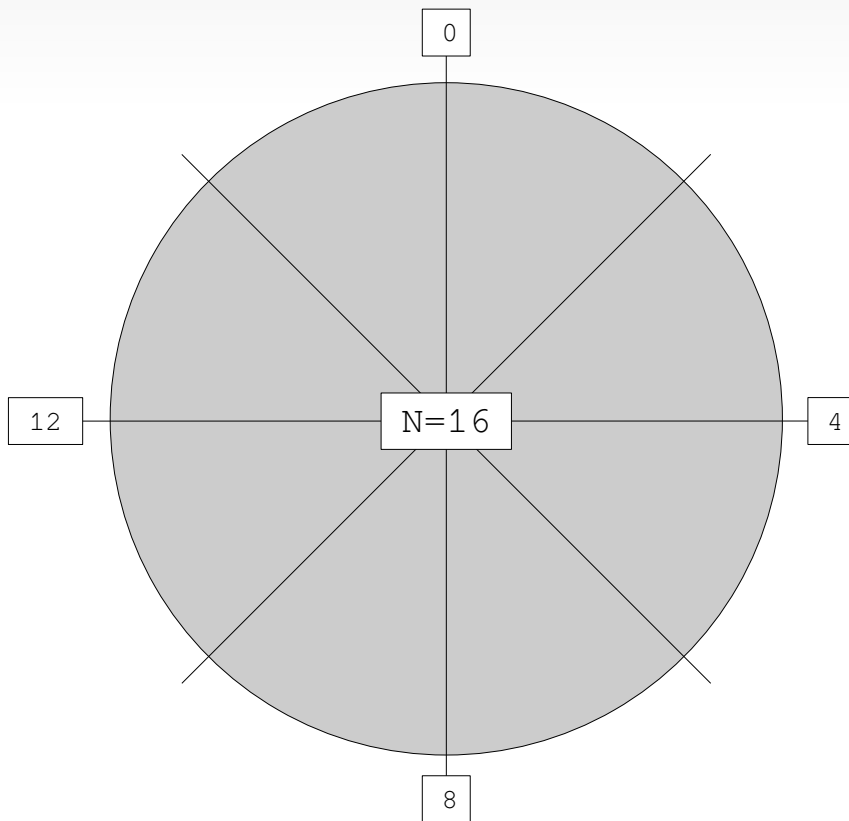
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



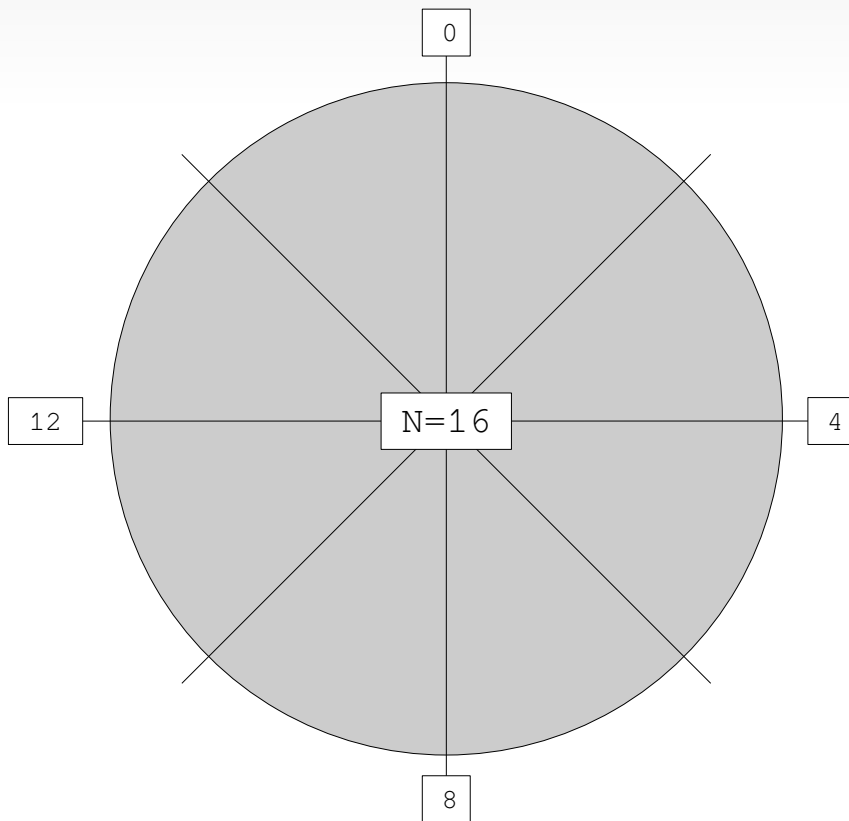
Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



Identificação dos Recursos

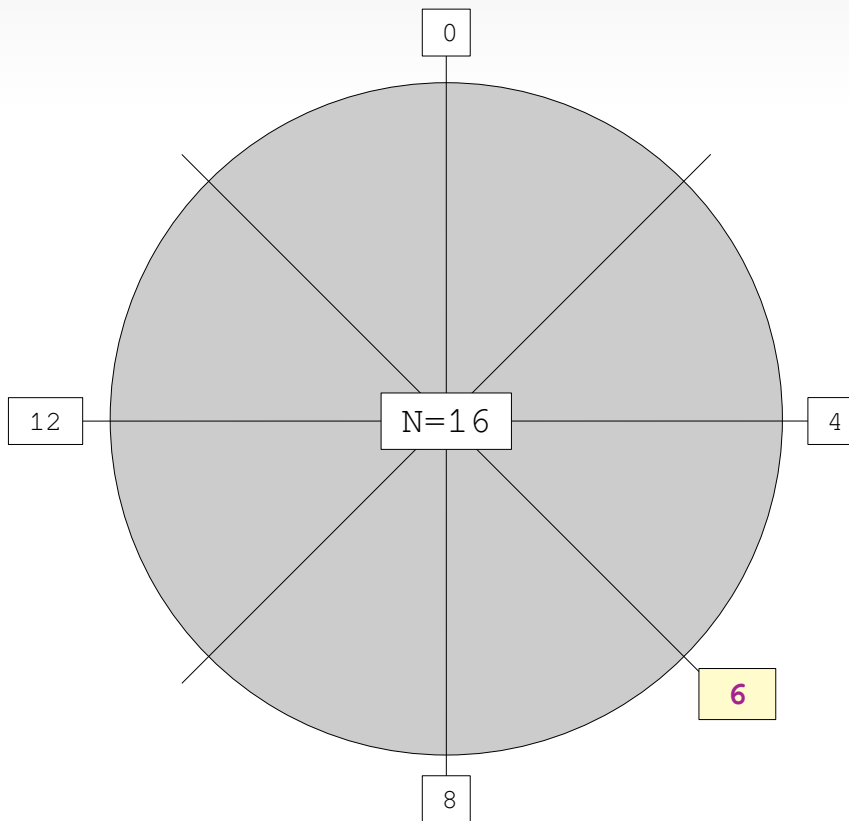
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

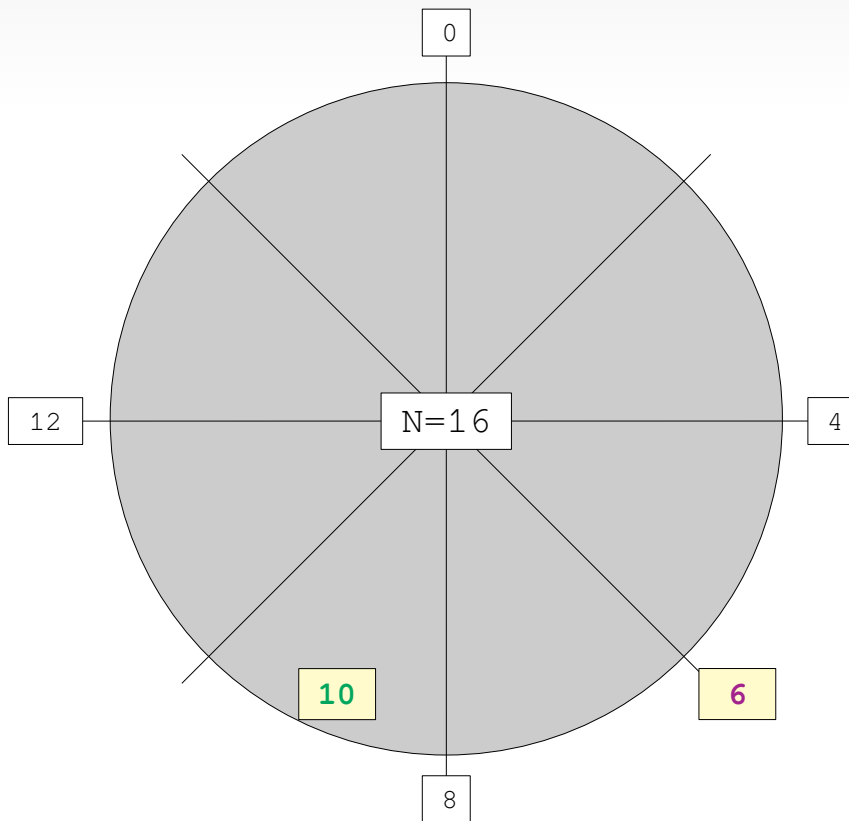
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

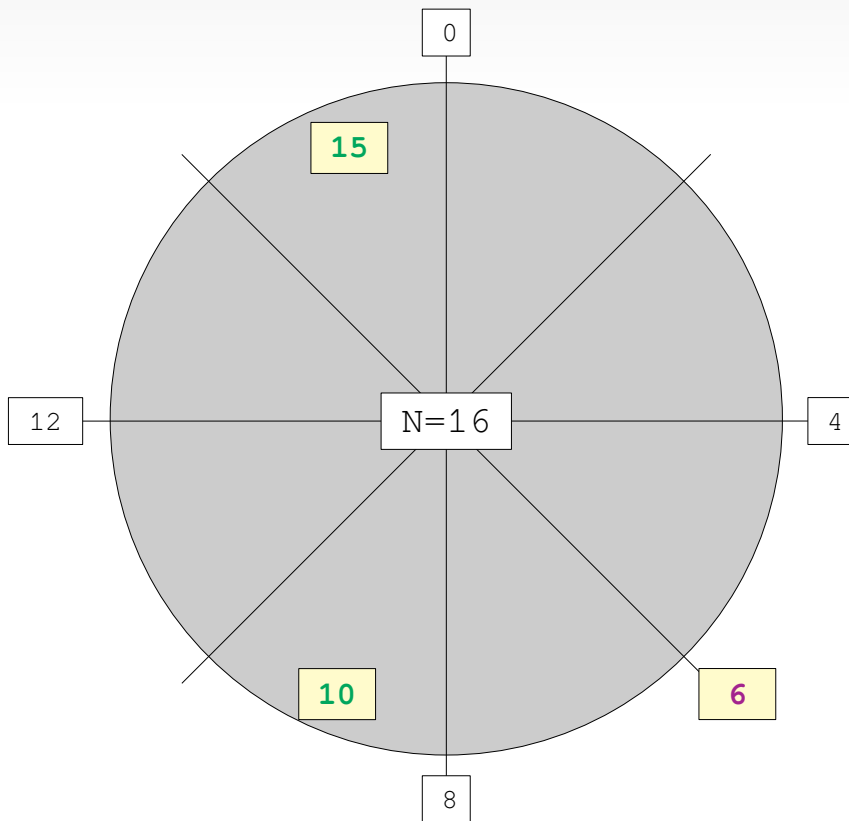
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

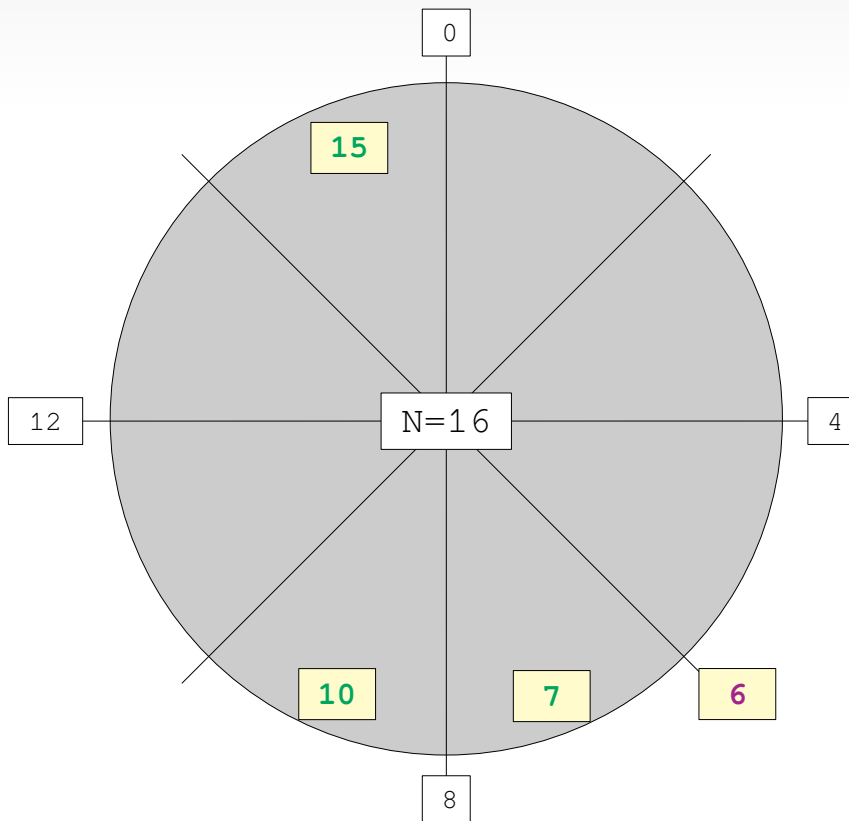
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

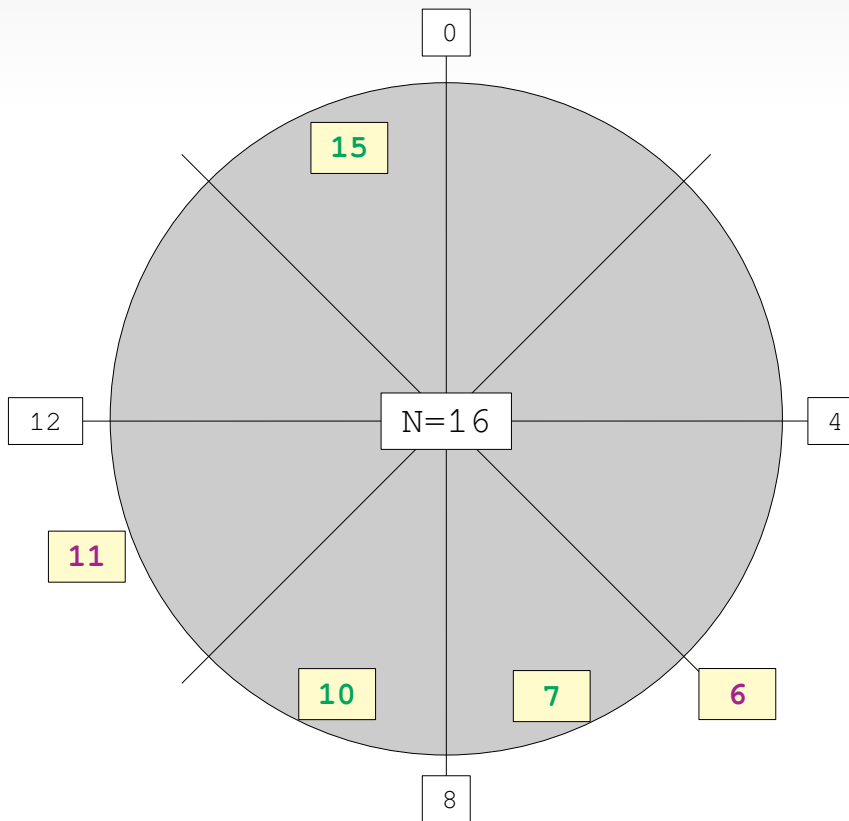
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

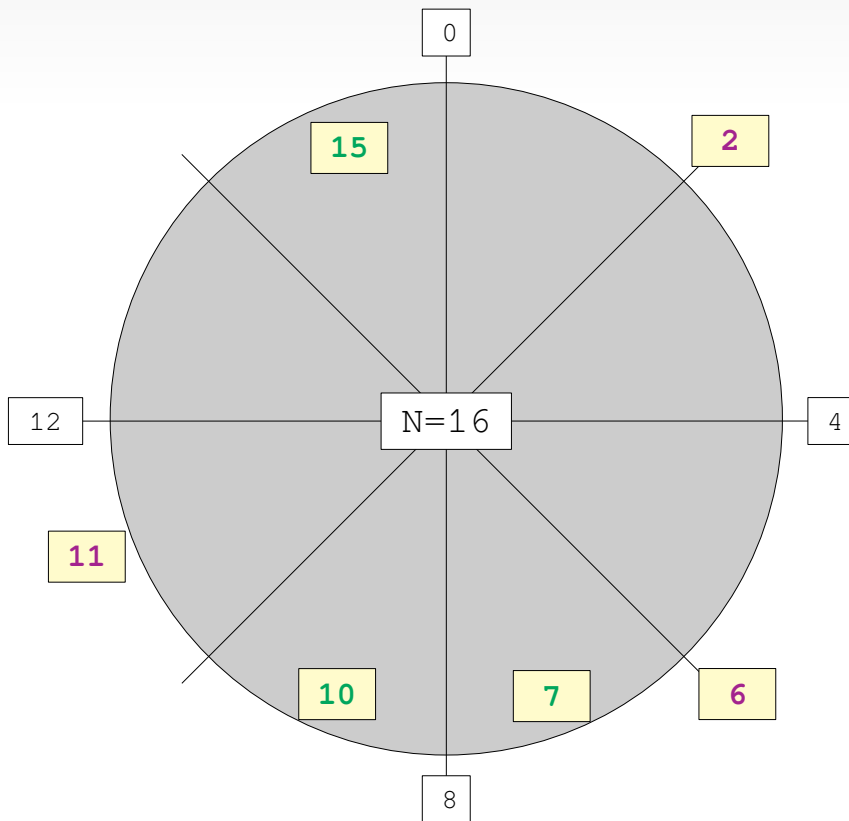
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

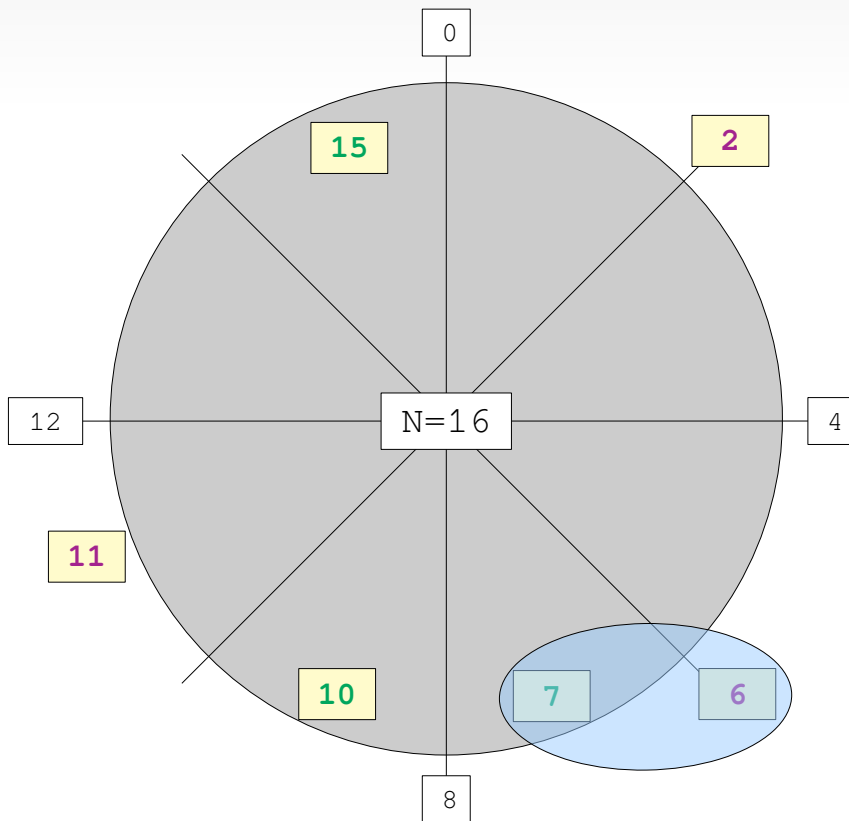
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

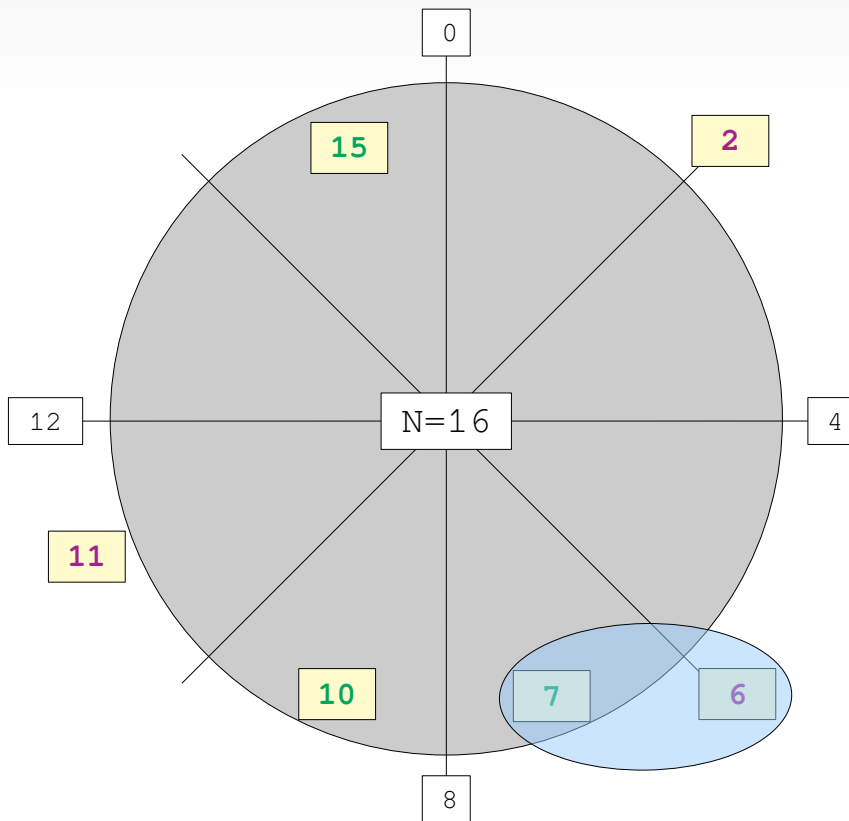
- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



<code>HASH(ubuntu.iso)</code>	=	6	[0..16[
<code>HASH(MAC-ADDR)</code>	=	10	[0..16[

Identificação dos Recursos

- Como dados e máquinas são identificados?
 - E se usarmos identificadores lógicos no mesmo espaço?



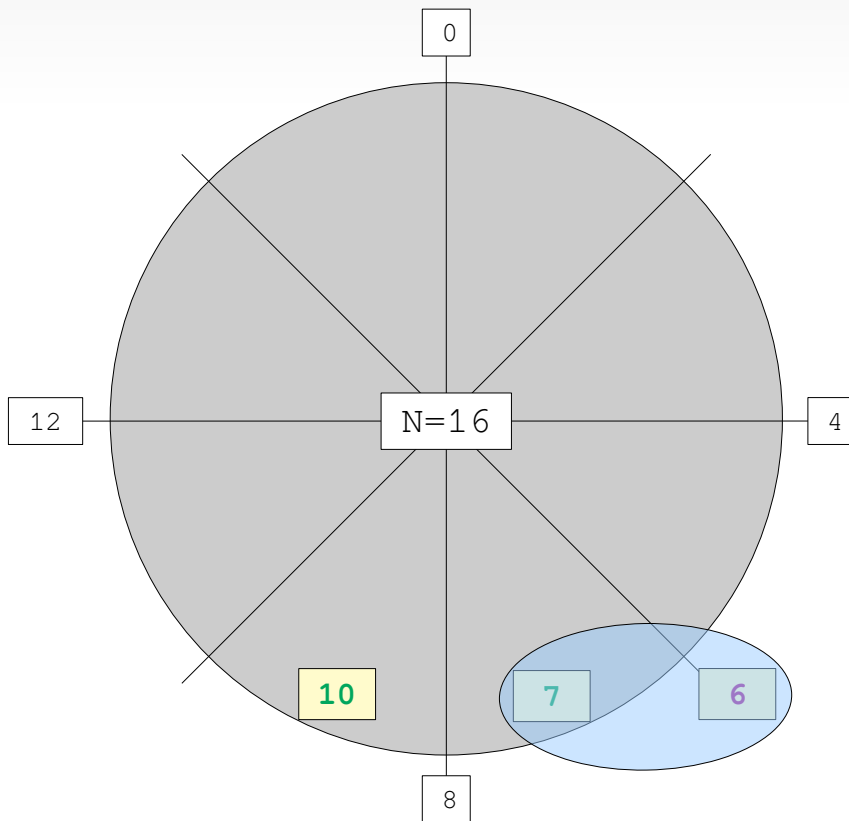
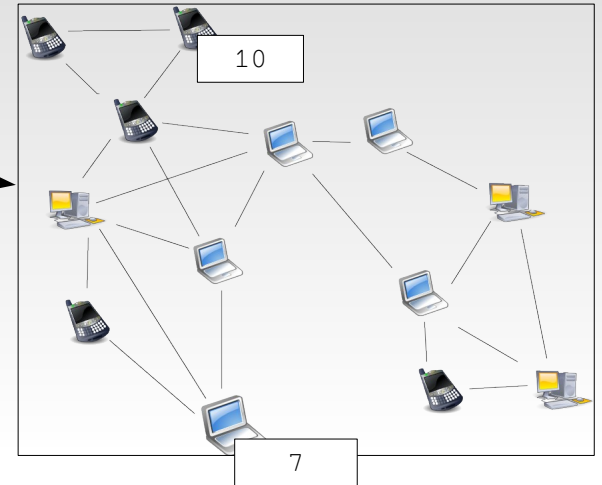
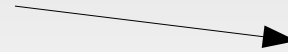
$\text{HASH}(\text{ubuntu.iso})$	$=$	6	$[0..16[$
$\text{HASH}(\text{MAC-ADDR})$	$=$	10	$[0..16[$

Consistent
Hashing

Roteamento

- **Quem são meus vizinhos na rede overlay?**

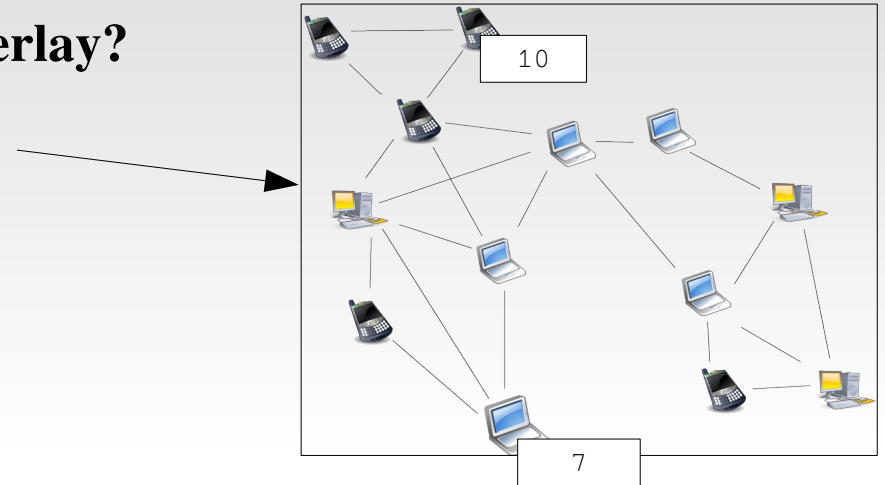
- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



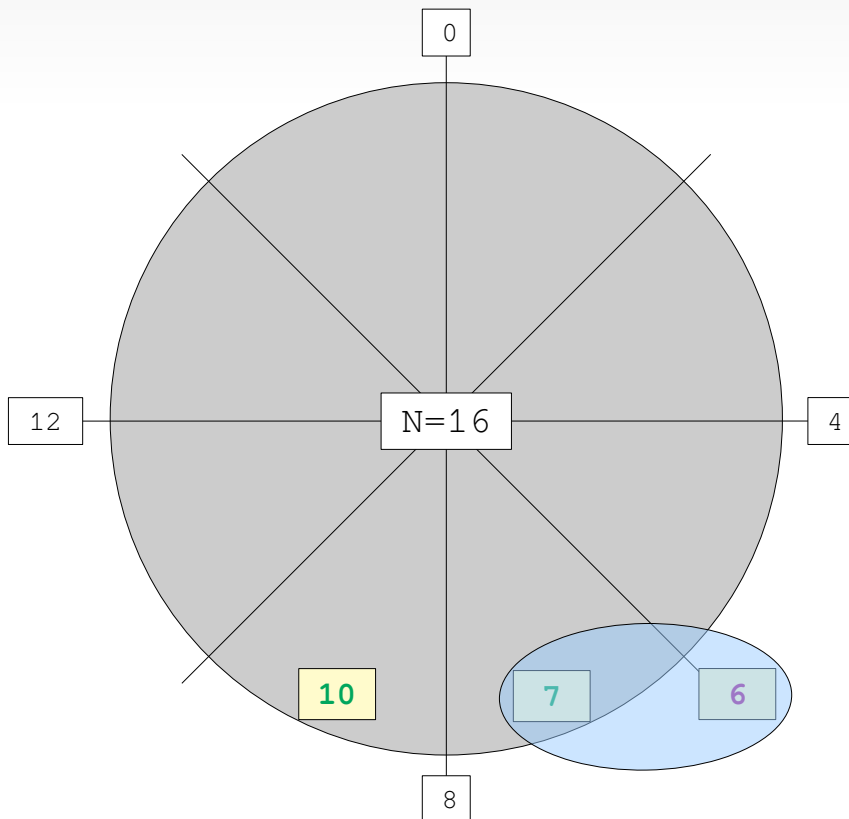
Roteamento

- **Quem são meus vizinhos na rede overlay?**

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



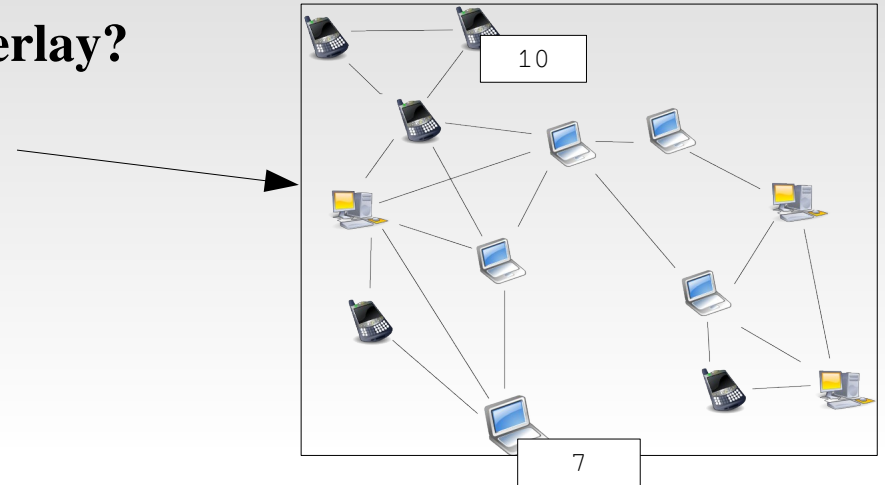
- 1-hop routing



Roteamento

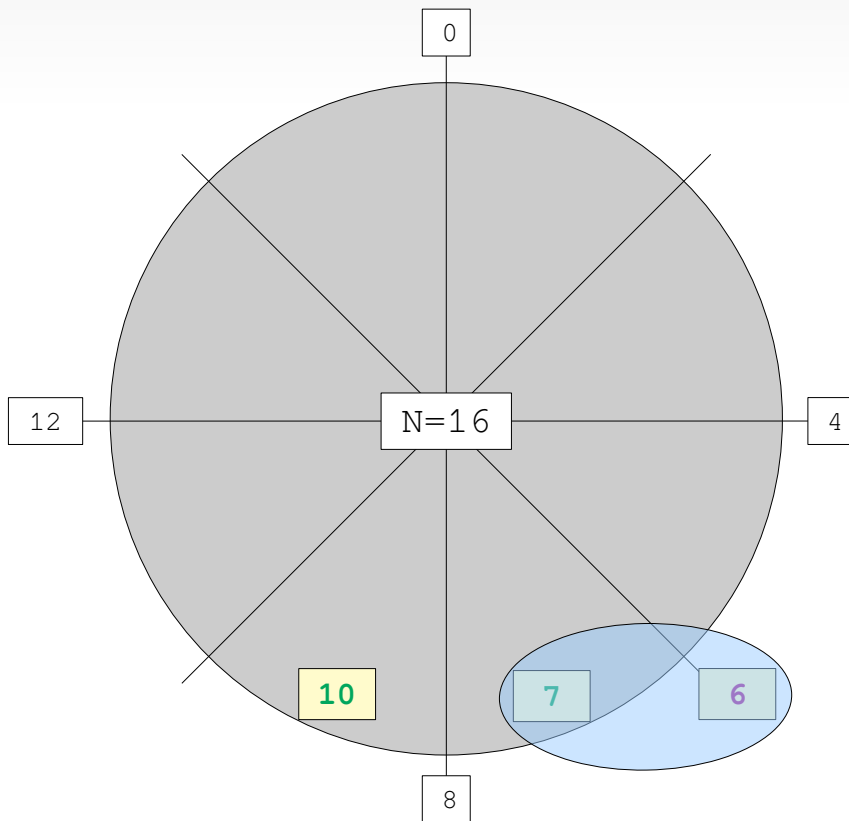
- **Quem são meus vizinhos na rede overlay?**

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- **1-hop routing**

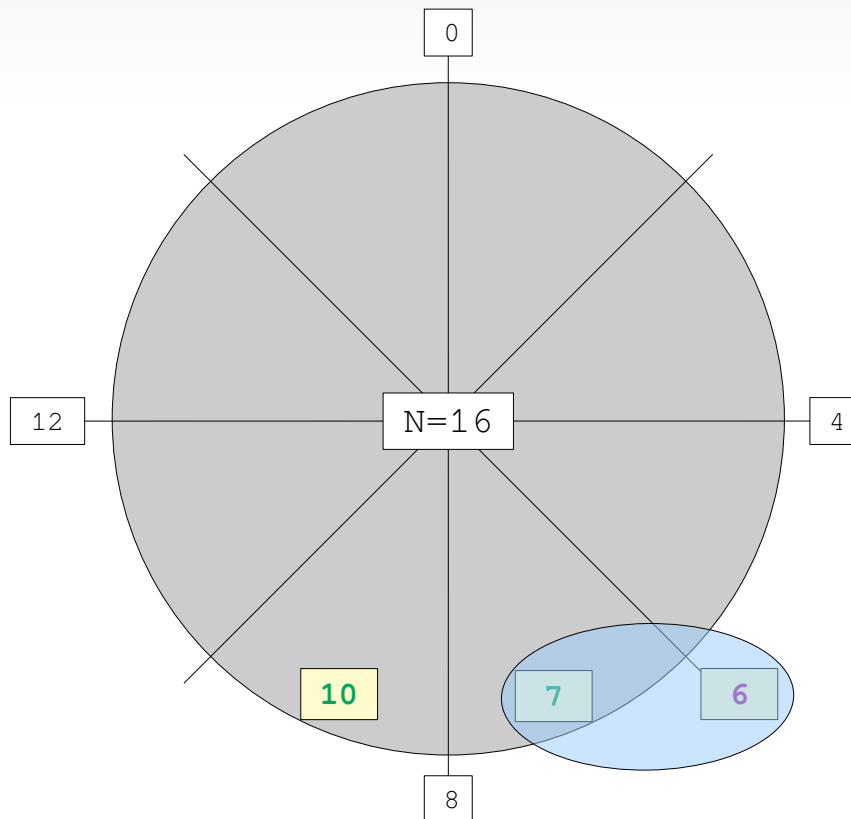
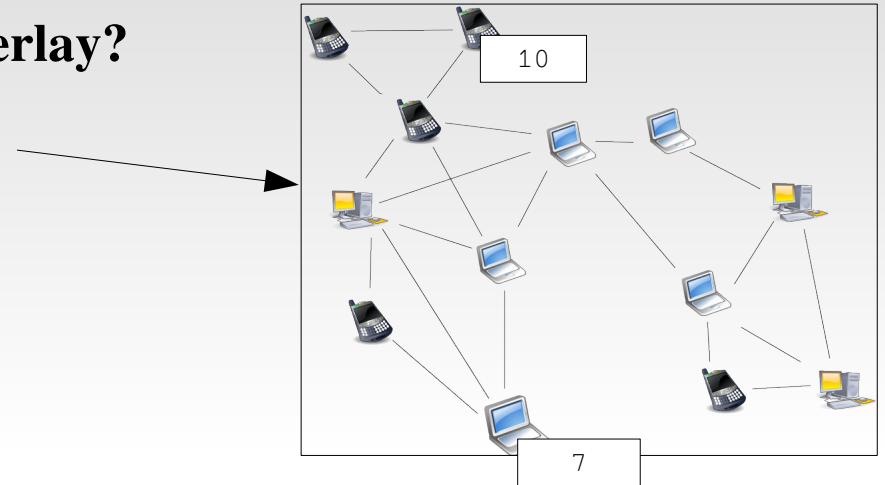
- todas as máquinas guardam ID→IP



Roteamento

- **Quem são meus vizinhos na rede overlay?**

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



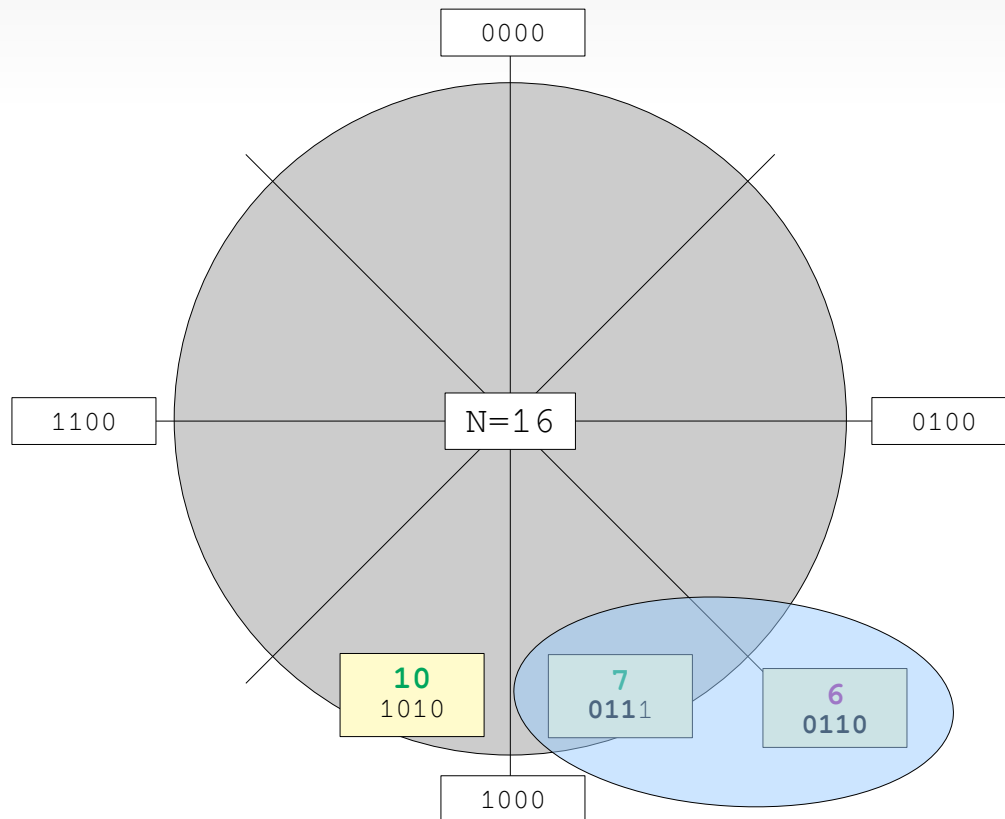
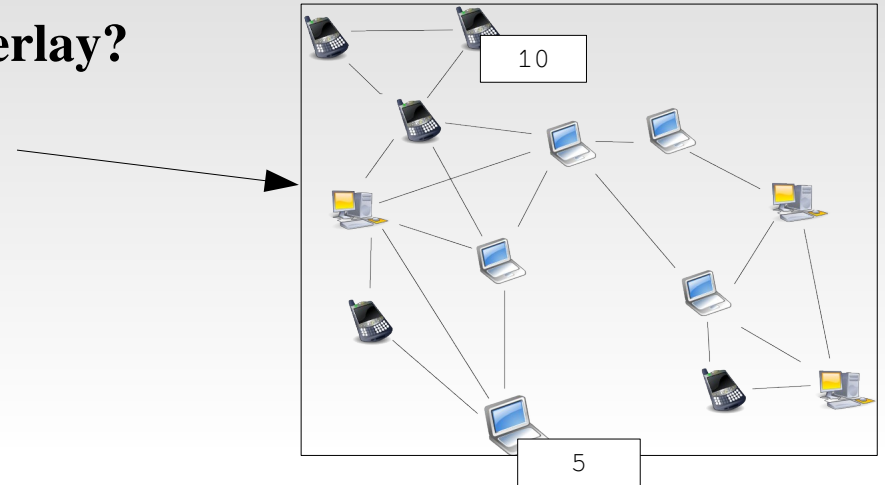
- **1-hop routing**

- todas as máquinas guardam ID→IP
- ~~tamanho~~ + manutenção

Roteamento

- **Quem são meus vizinhos na rede overlay?**

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



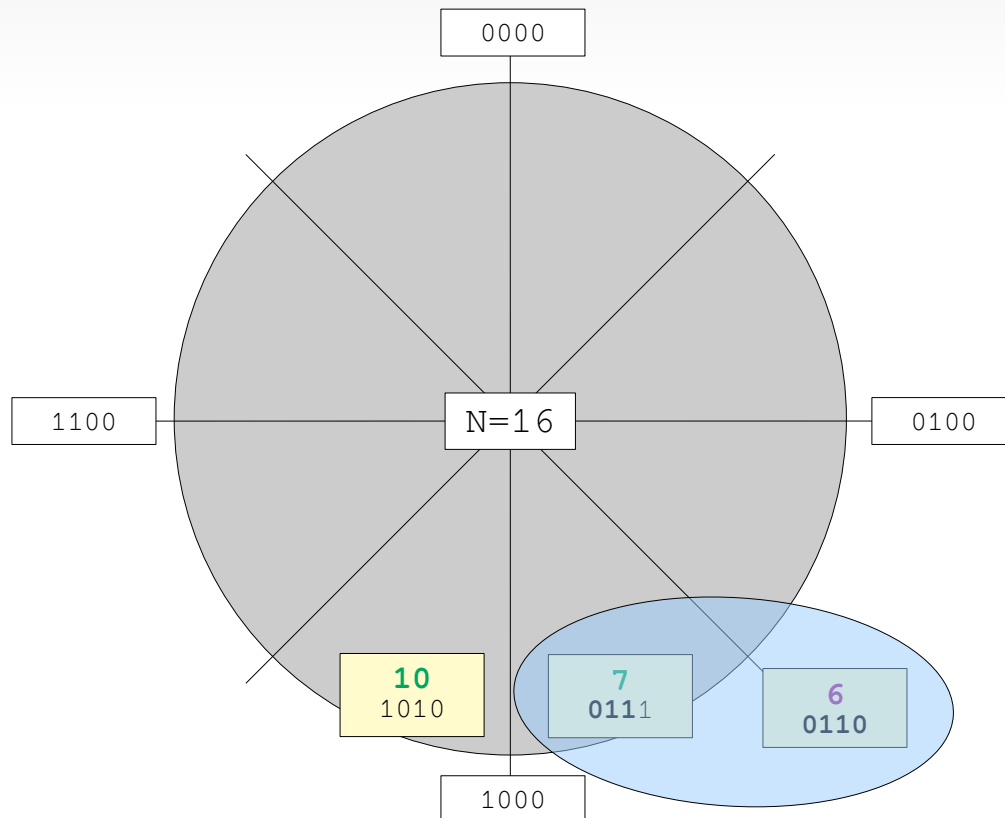
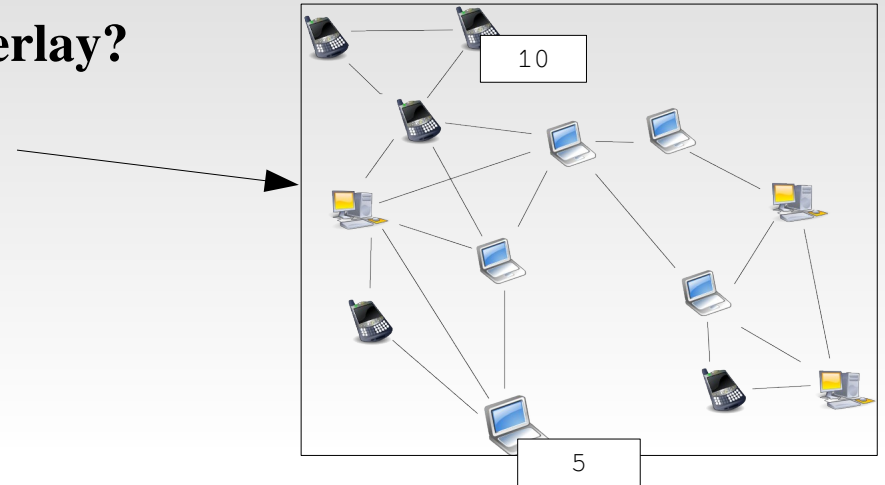
- **1-hop routing**

- todas as máquinas guardam ID→IP
- ~~tamanho + manutenção~~

Roteamento

- **Quem são meus vizinhos na rede overlay?**

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- **1-hop routing**

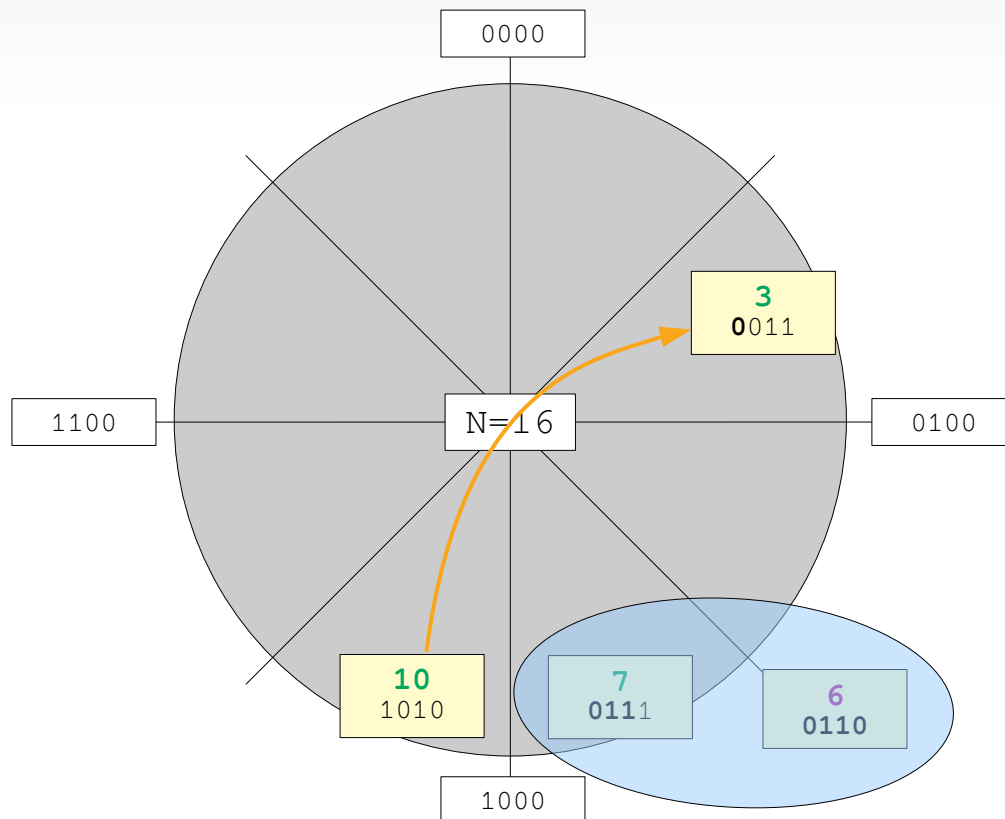
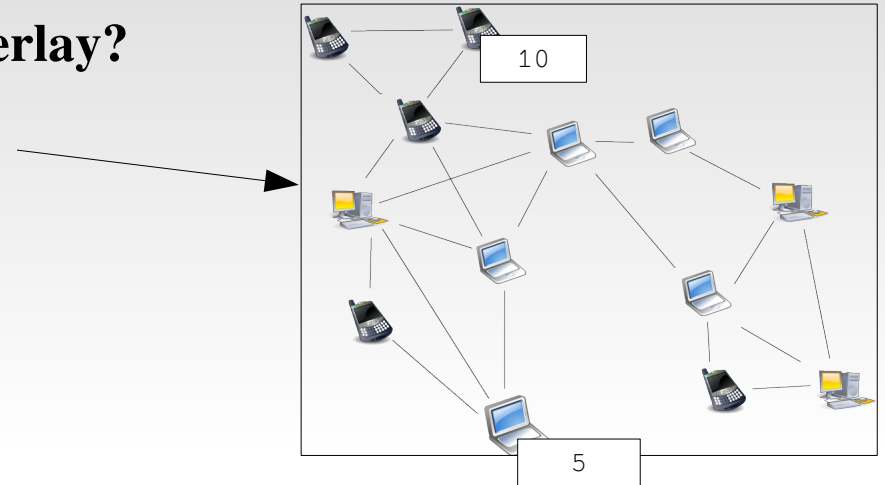
- todas as máquinas guardam ID→IP
- ~~tamanho~~ + manutenção

- **Prefix-based routing**

Roteamento

- **Quem são meus vizinhos na rede overlay?**

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- **1-hop routing**

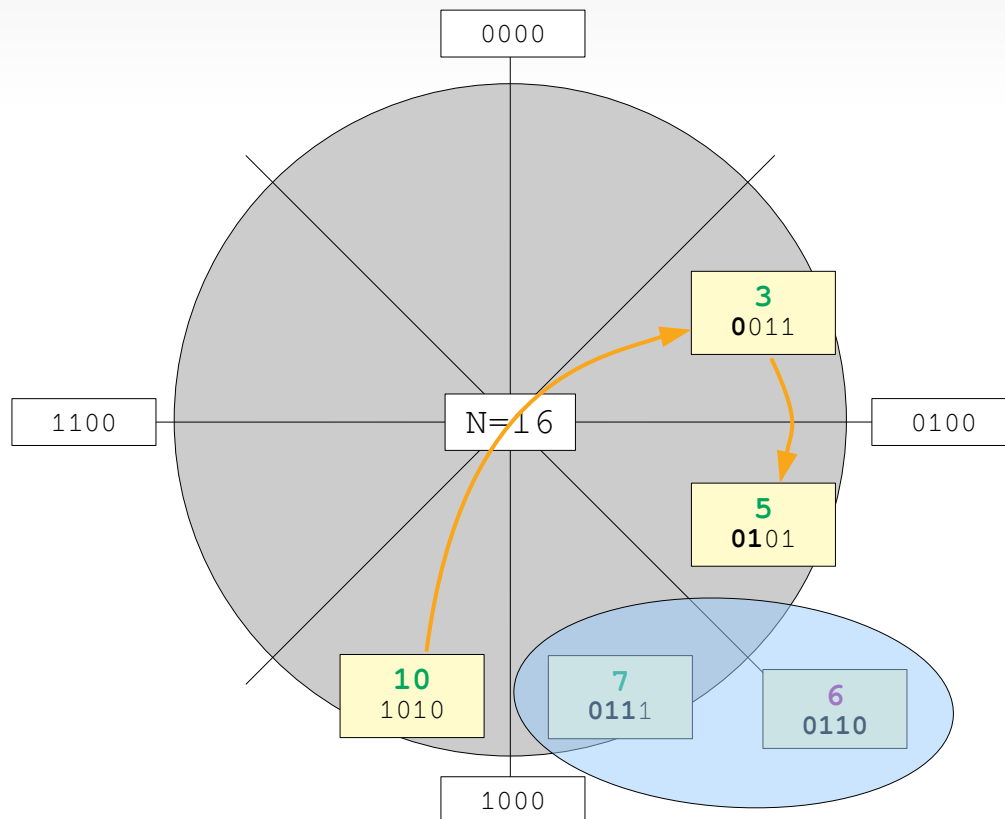
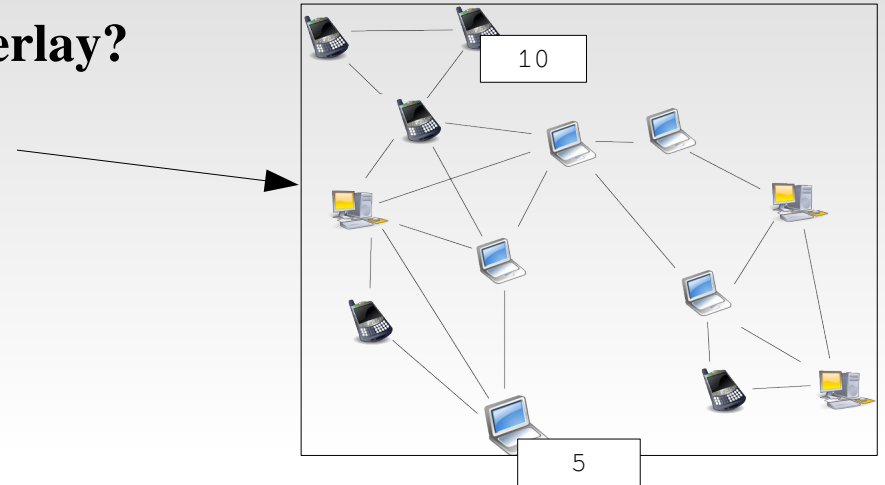
- todas as máquinas guardam ID→IP
- ~~tamanho~~ + manutenção

- **Prefix-based routing**

Roteamento

- Quem são meus vizinhos na rede overlay?

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- 1-hop routing

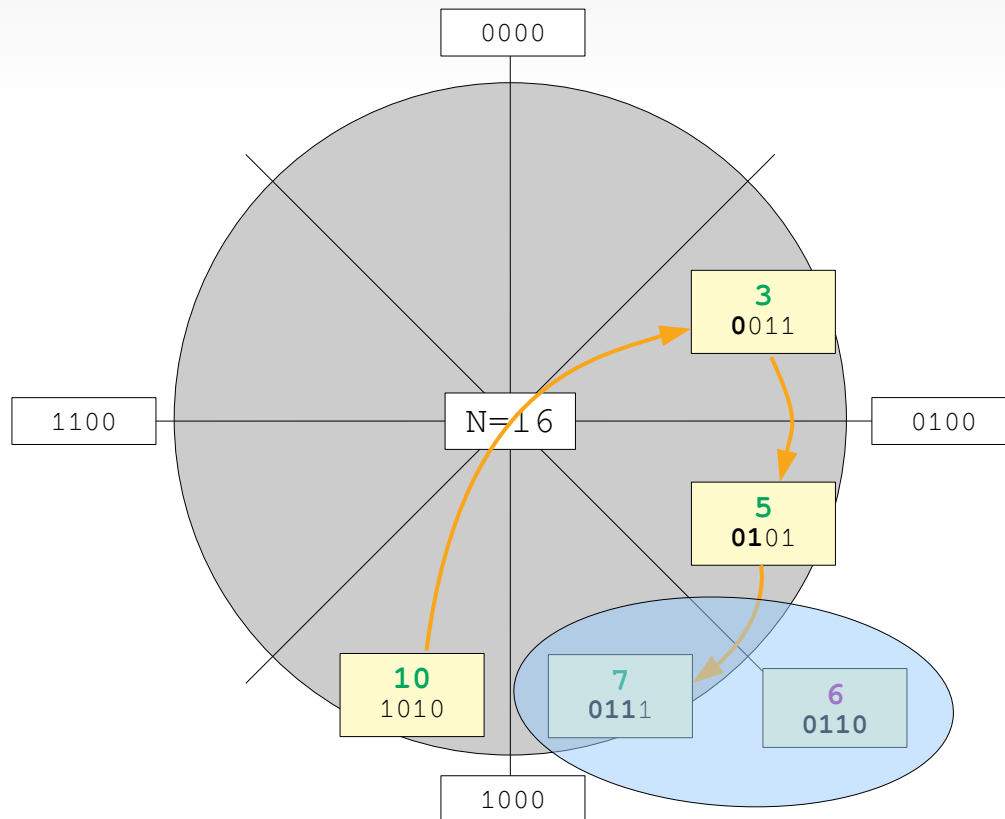
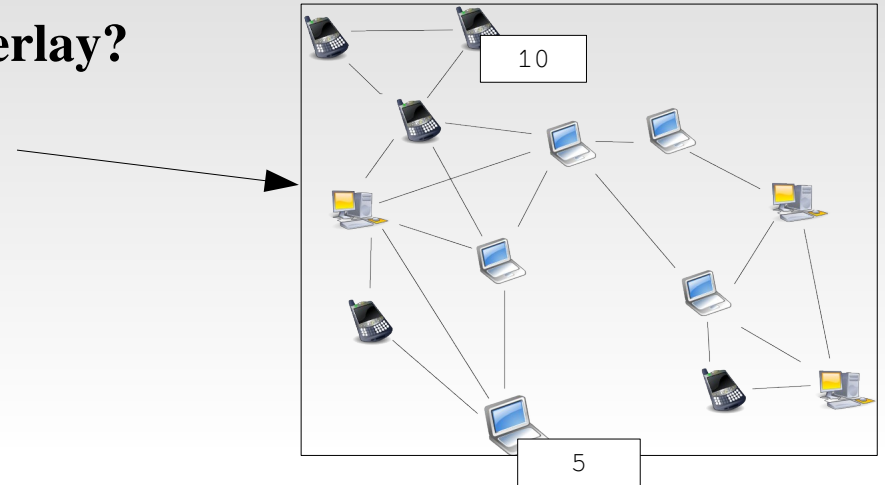
- todas as máquinas guardam ID→IP
- ~~tamanho + manutenção~~

- Prefix-based routing

Roteamento

- Quem são meus vizinhos na rede overlay?

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- 1-hop routing

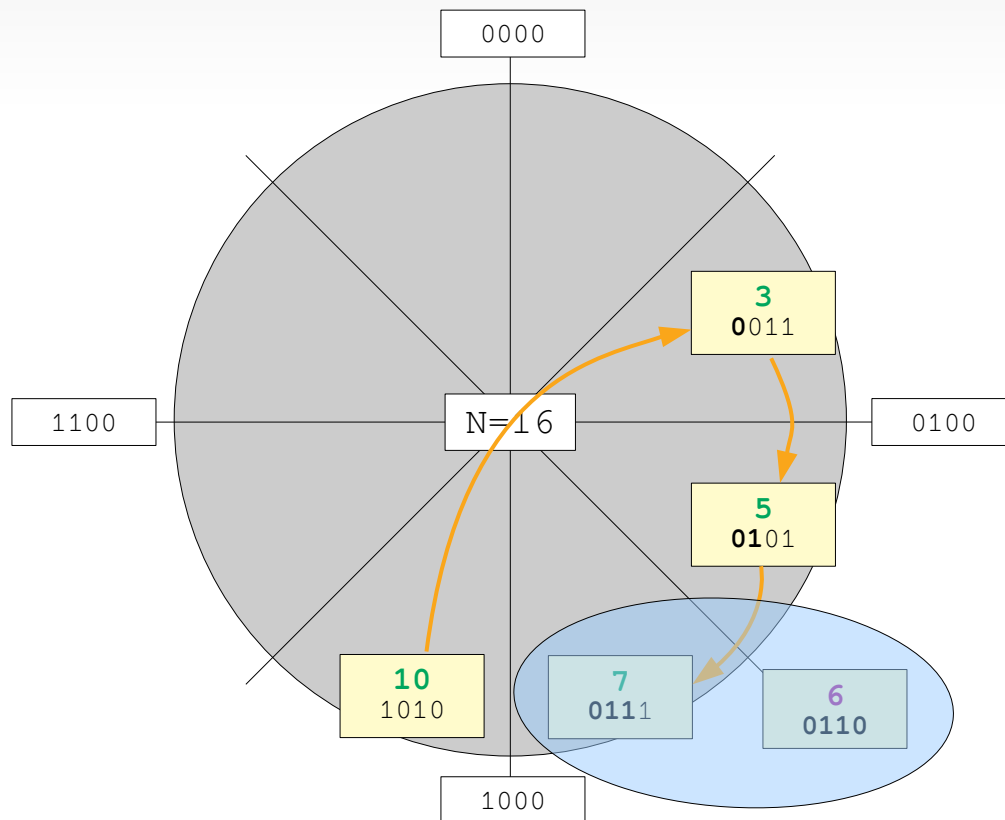
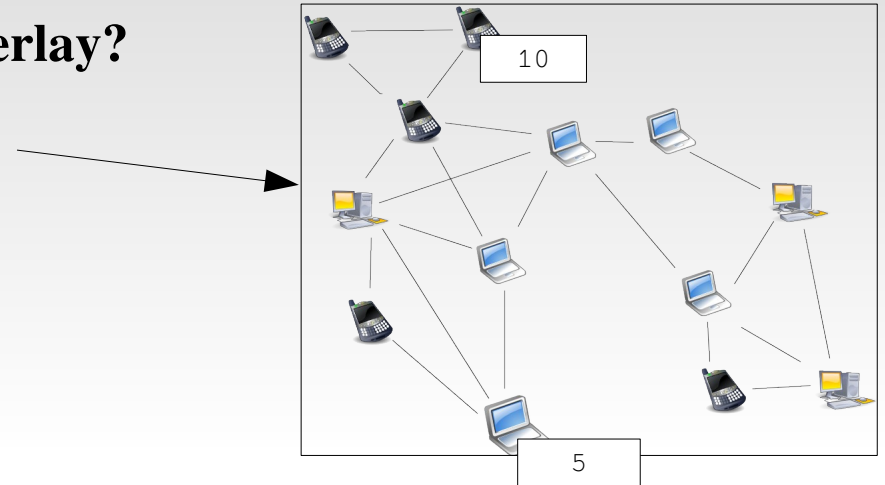
- todas as máquinas guardam ID→IP
- ~~tamanho~~ + manutenção

- Prefix-based routing

Roteamento

- Quem são meus vizinhos na rede overlay?

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- 1-hop routing

- todas as máquinas guardam ID→IP
- ~~tamanho~~ + ~~manutenção~~

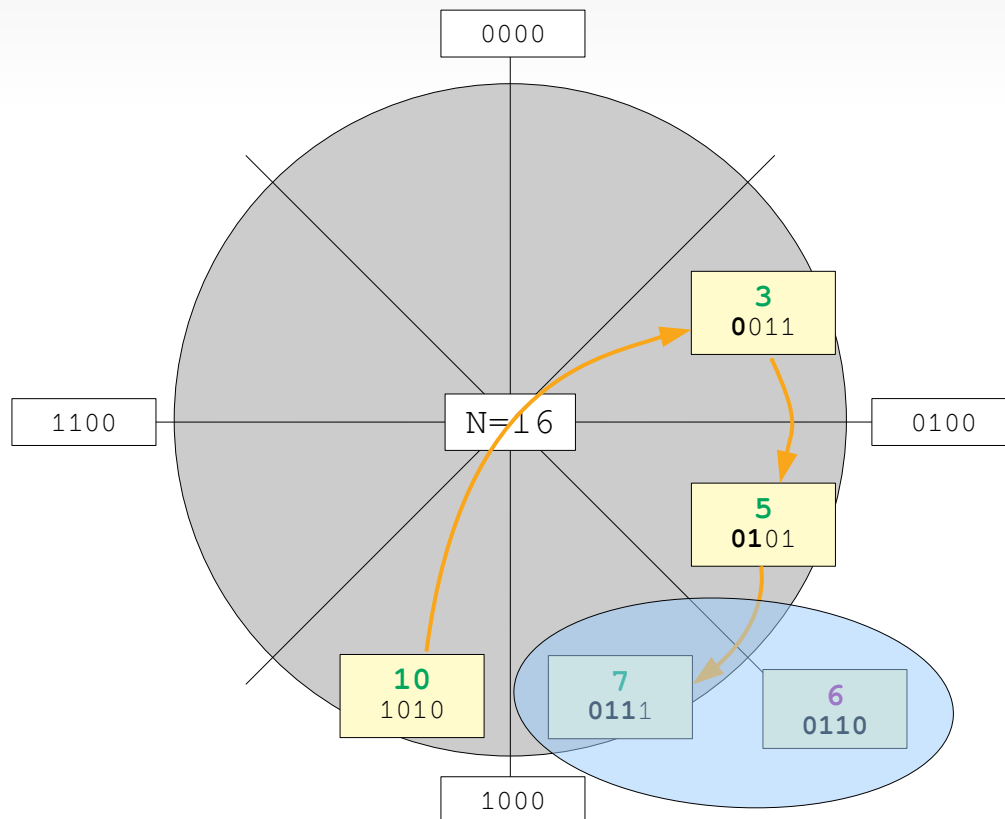
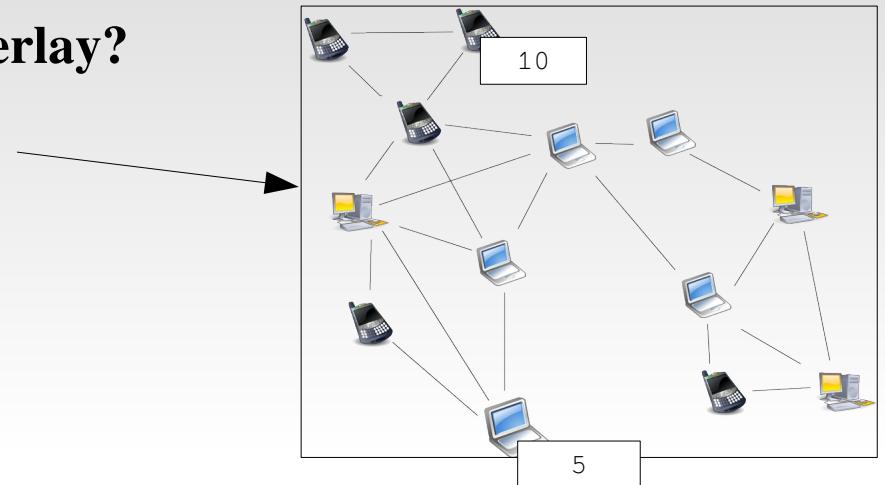
- Prefix-based routing

- tamanho + manutenção + saltos

Roteamento

- Quem são meus vizinhos na rede overlay?

- Sou a máquina 10 e quero o dado 6
- Como faço o download do arquivo?



- 1-hop routing

- todas as máquinas guardam ID→IP
- ~~tamanho~~ + ~~manutenção~~

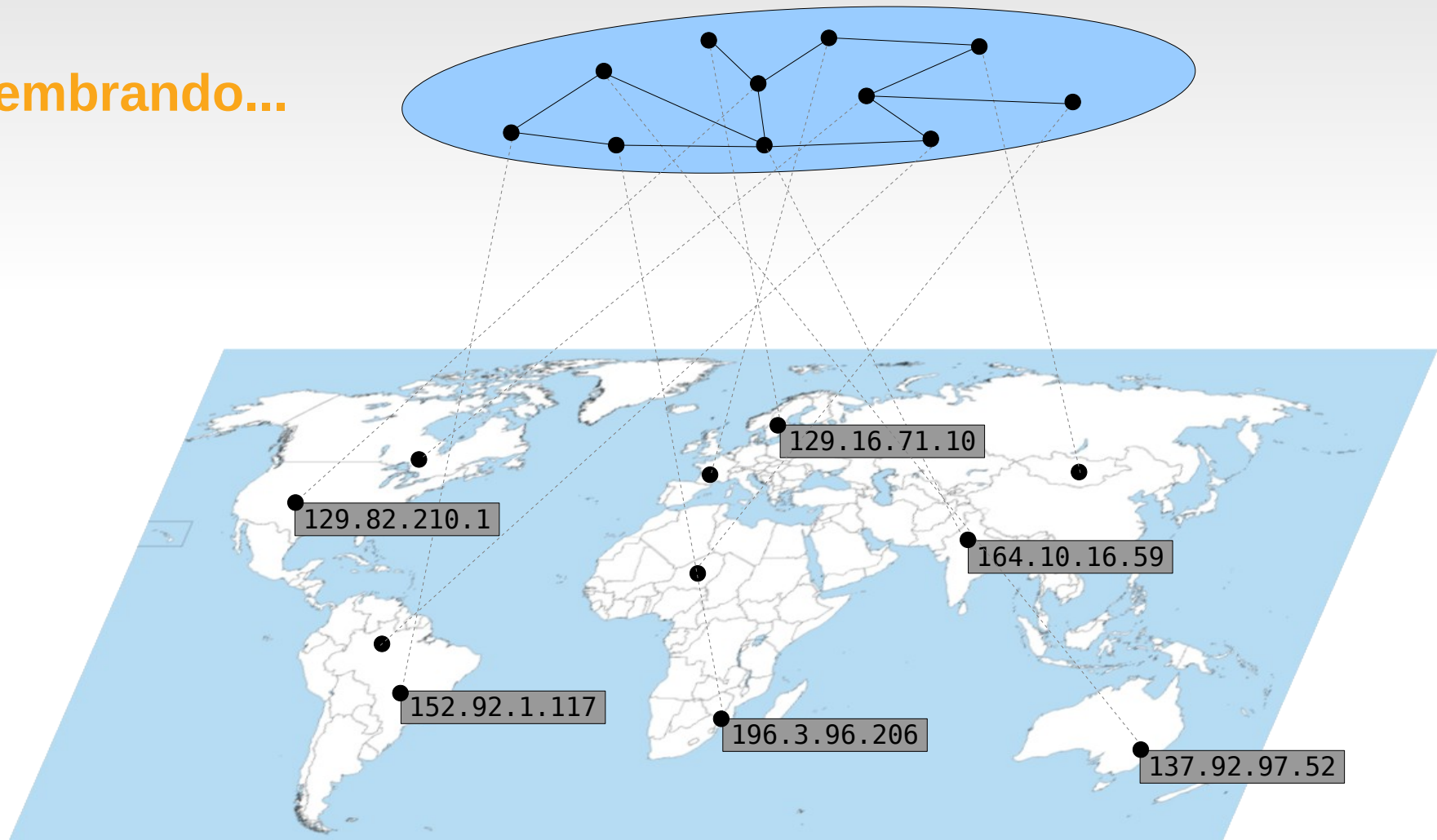
- Prefix-based routing

- tamanho + manutenção + saltos

Key-Based
Routing

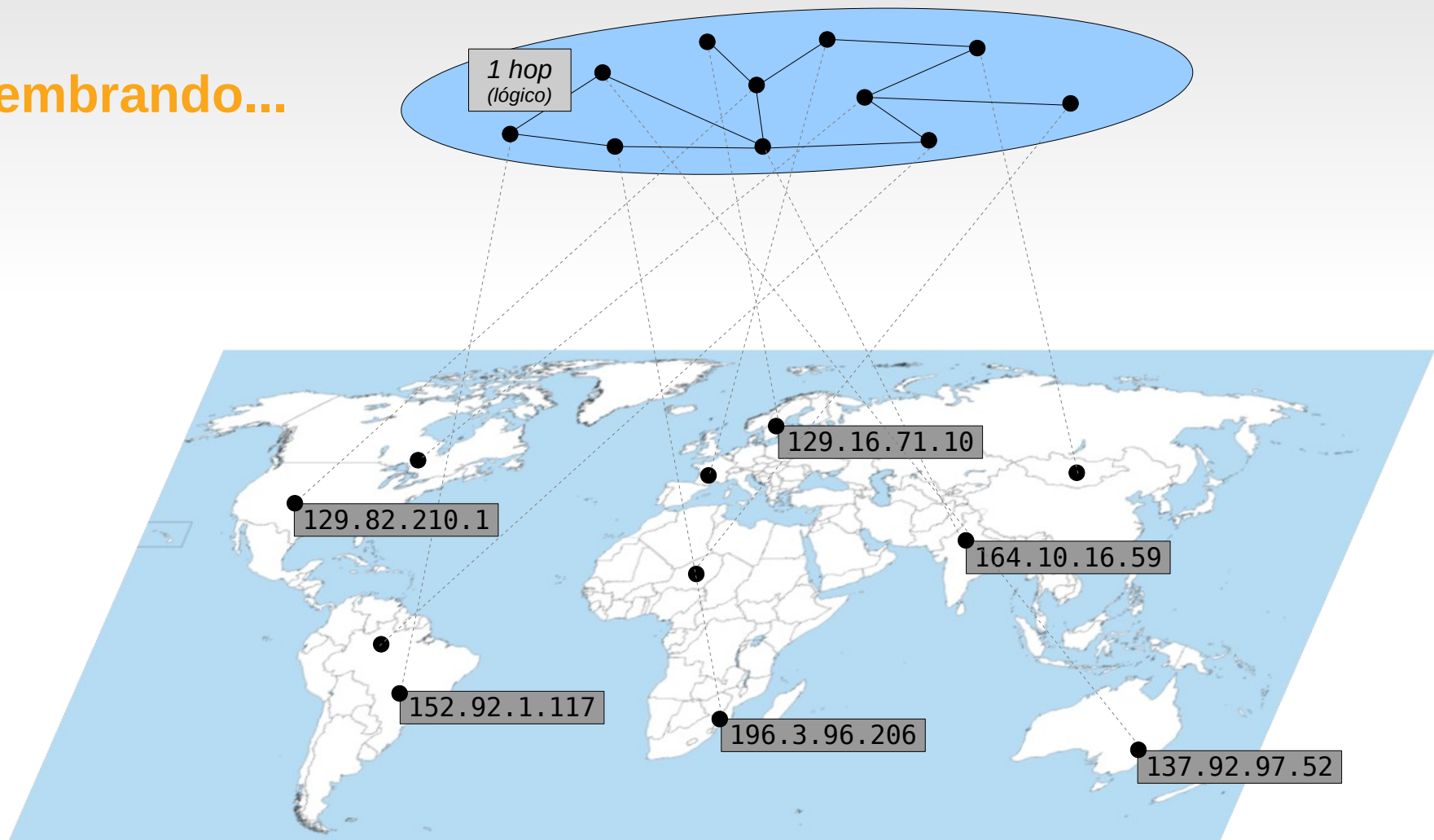
Topologia Overlay

lembrando...



Topologia Overlay

lembrando...



Topologia Overlay

lembrando...

