



Corso di SISTEMI DISTRIBUITI
E CLOUD COMPUTING

Chord: Protocollo di Routing Peer-to-Peer per Reti Distribuite

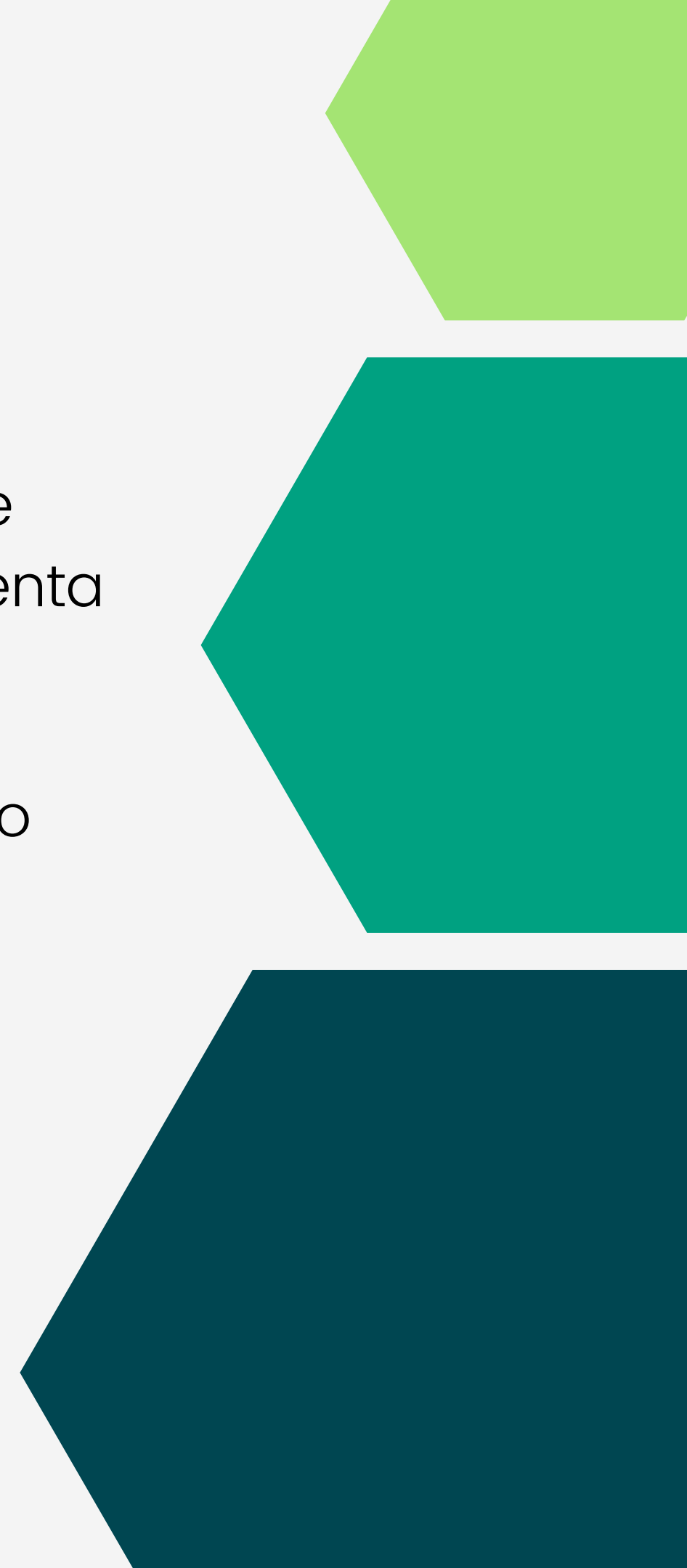
A cura di
Marina Calcaura
0334044

Elenco dei contenuti

- Introduzione
- Obiettivi
- Struttura della rete
- Routing
- Join di un nodo
- Leave volontario di un nodo
- Gestione delle risorse
(inserimento e recupero)

Le reti P2P spesso coinvolgono un gran numero di nodi e risorse distribuite su vasta scala. La ricerca efficiente di dati in tali reti diventa complessa e costosa in termini di latenza.

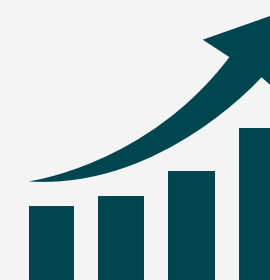
È qui che entra in gioco **Chord**, un protocollo di routing distribuito che fornisce una struttura scalabile ed efficiente per la distribuzione di dati in una rete decentralizzata.



Obiettivi del protocollo Chord



Stabilire un sistema di
indirizzamento e
ricerca efficiente



Mantenere la
scalabilità con un
numero crescente
di nodi



Supportare l'inserimento e la
rimozione dinamica dei nodi

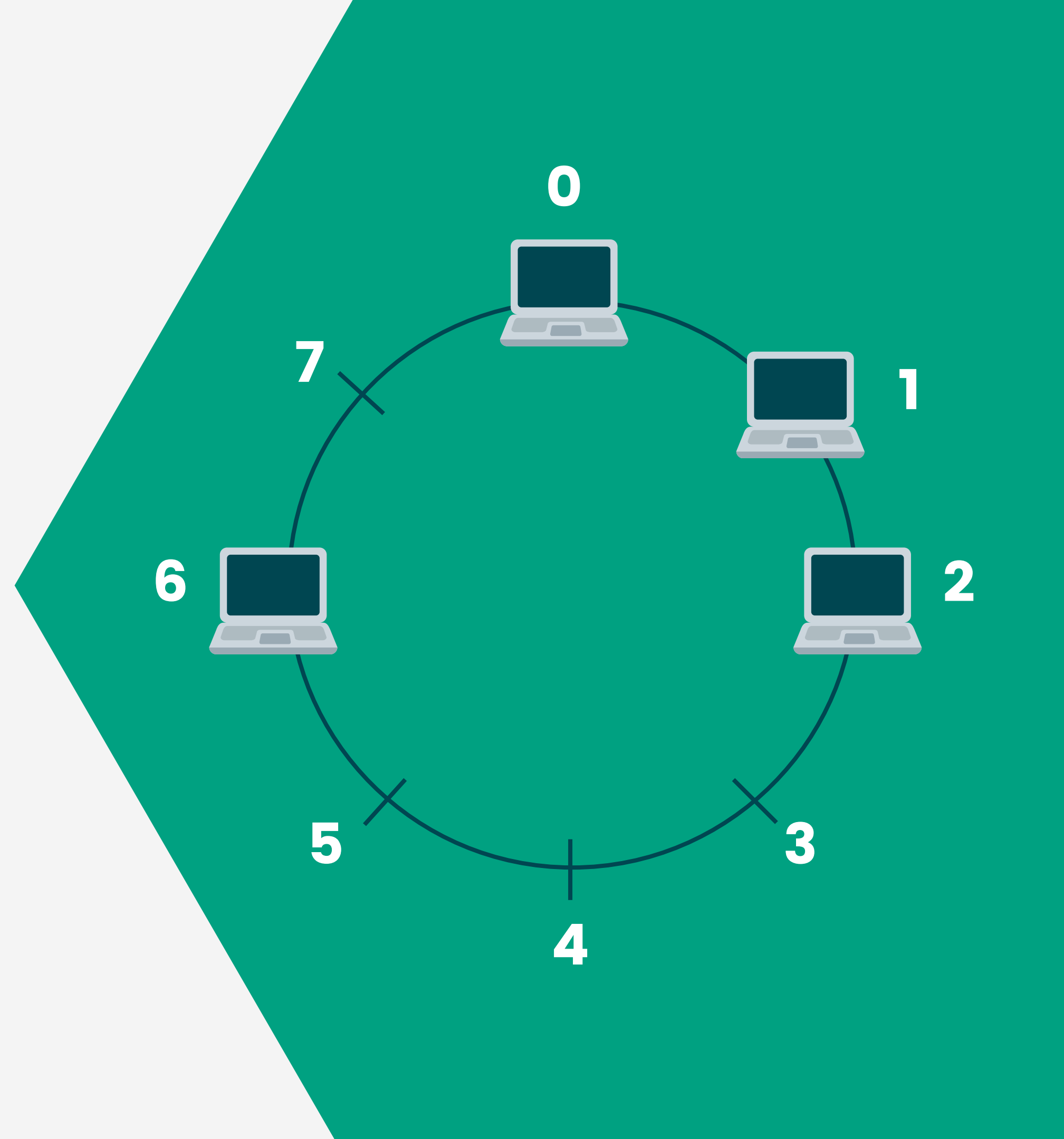
Struttura della rete Chord

Ring-based: nodi
disposti in un
anello virtuale

Consistent
hashing

Ogni nodo è
responsabile di
un intervallo
specifico di
identificatori

Ogni nodo ha un
identificatore
univoco, generato
utilizzando la
funzione di hash
SHA-256

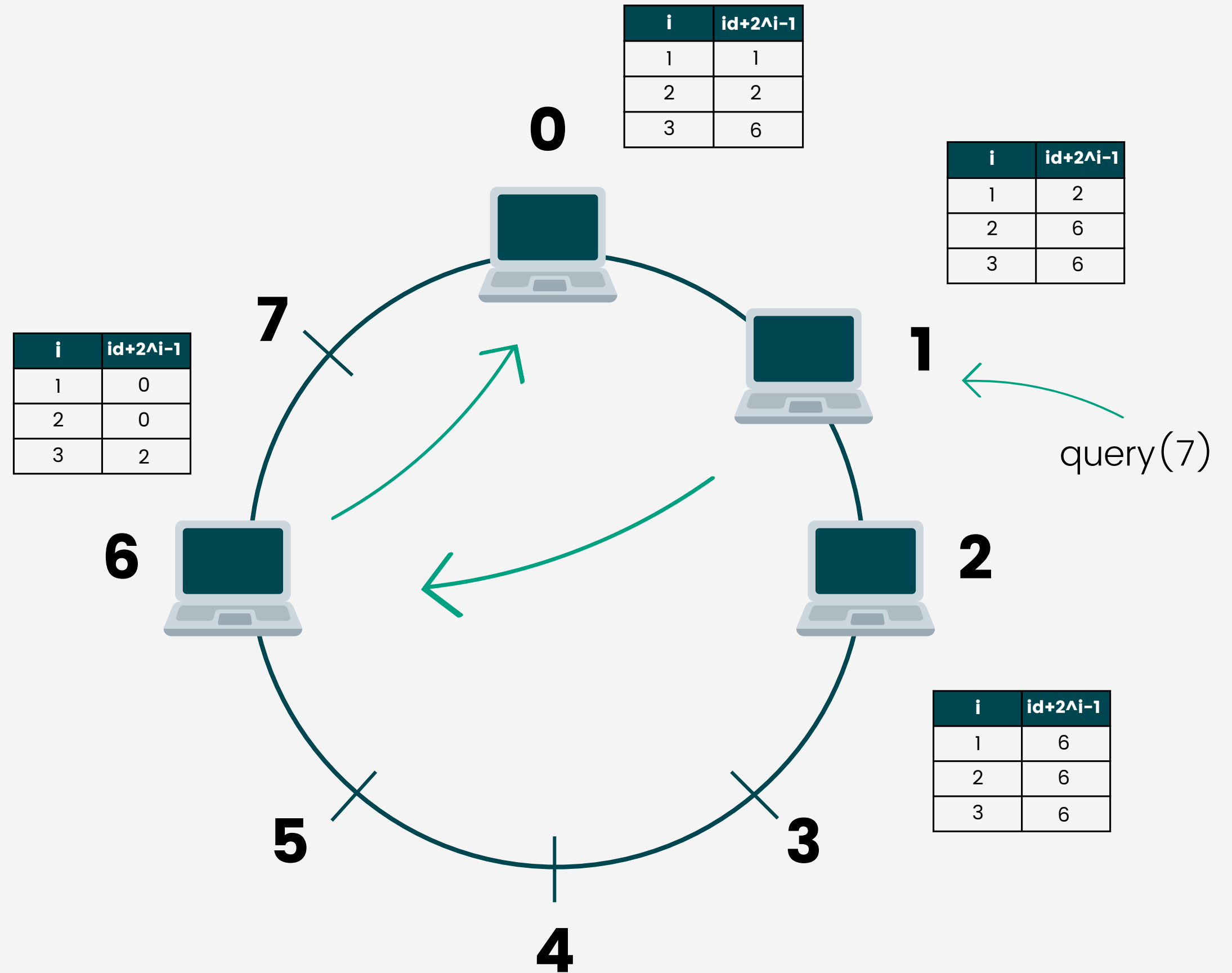


Routing

Ogni nodo mantiene un insieme di finger, puntando a nodi con identificatori lontani

Utilizzo di finger tables per le rotte efficienti

Ricerca di un nodo obiettivo con complessità logaritmica



Server registry



Fornisce il punto di accesso del sistema per il client e per i nodi, mantenendo una lista ordinata dei nodi

Restituisce il predecessore e il successore a eventuali nodi che ne fanno richiesta

Join di un nodo

Identificazione del punto di ingresso nella rete grazie al server registry

Fase nella quale il nuovo nodo identifica il suo successore e predecessore per comprendere la sua posizione nell'anello Chord

Scambio di informazioni

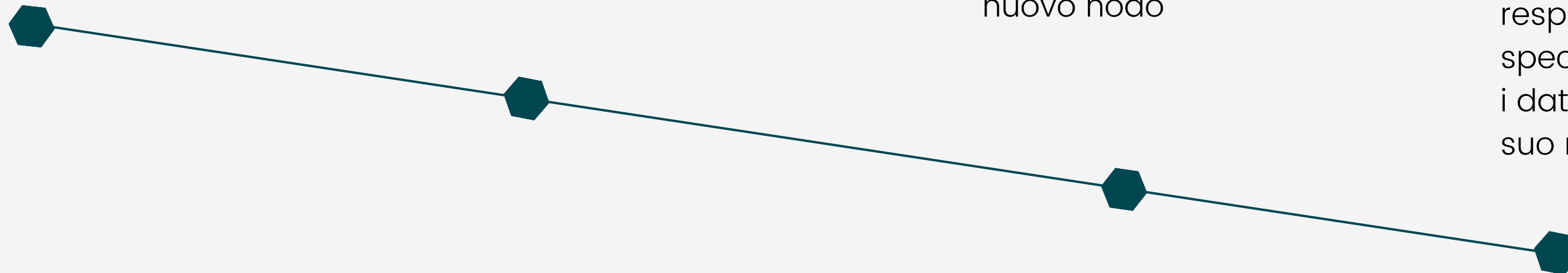
Il nuovo nodo informa il suo successore della sua presenza, inizializza la propria finger table e informa il predecessore della sua presenza

Aggiornamento delle tabelle dei finger dei nodi esistenti

Processo di aggiornamento delle tabelle finger dei nodi esistenti per includere il nuovo nodo

Stabilimento della responsabilità per un intervallo di identificatori

Il nuovo nodo assume la responsabilità di un intervallo specifico della rete e recupera i dati di questo intervallo dal suo nodo successore



Leave volontaria di un nodo

Trasferimento dei dati

Tutti i dati di cui era responsabile il nodo diventano di responsabilità del successore

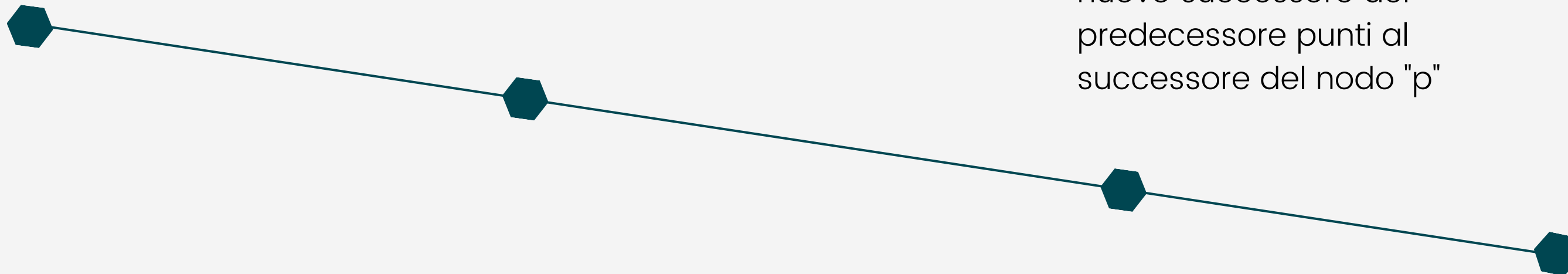
Aggiornamento del predecessore del successore

Il predecessore del successore del nodo che sta lasciando (nodo "p") verrà regolato in modo che il suo puntatore indirizzi il nodo che era il predecessore diretto di "p" prima della sua partenza.

Aggiornamento del successore del predecessore

L'aggiornamento del successore del predecessore del nodo che ha avviato la leave (nodo "p") verrà effettuato in modo che il nuovo successore del predecessore punti al successore del nodo "p"

Rimozione del nodo dalla rete



Gestione delle risorse

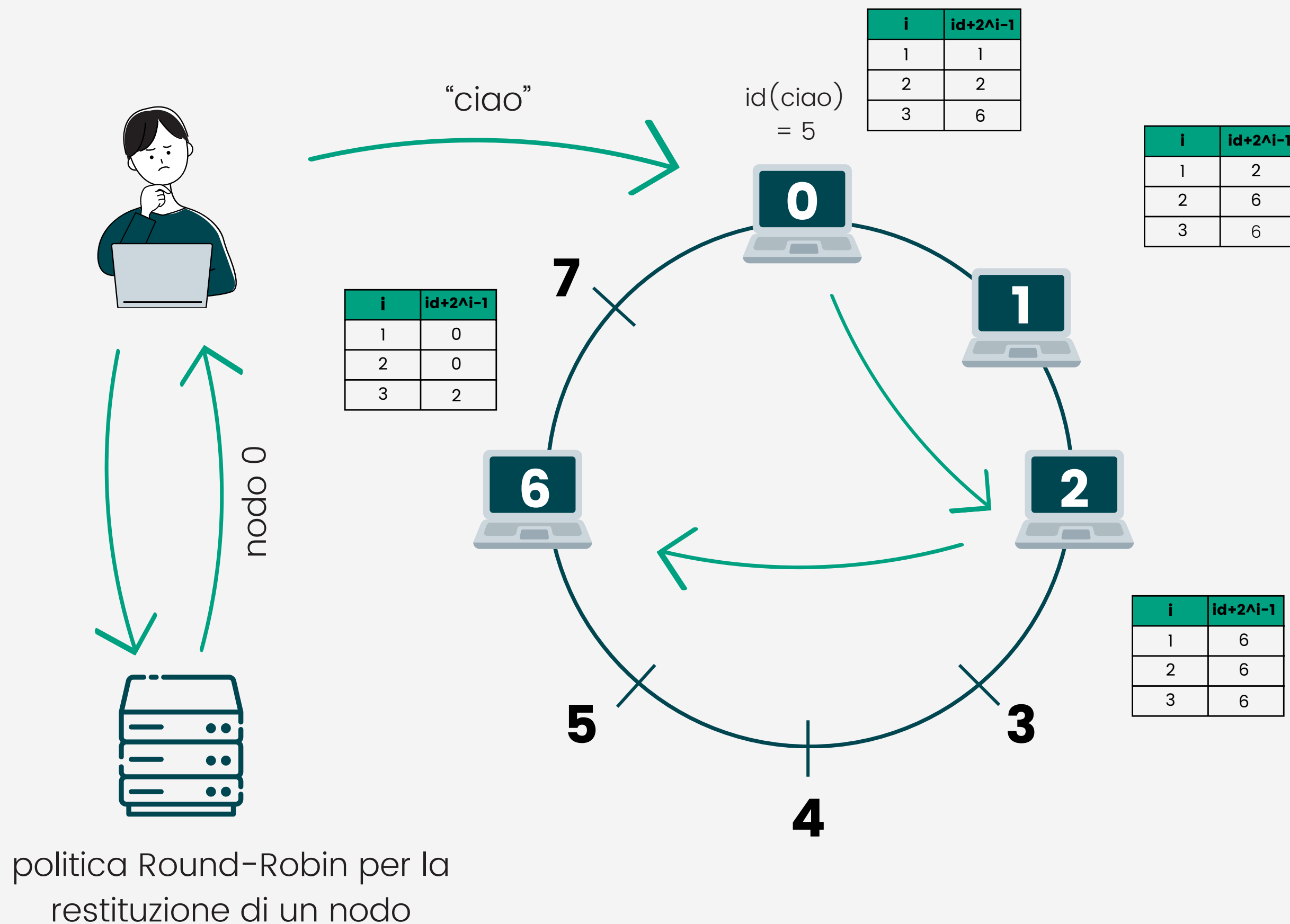
Inserimento di una risorsa



Recupero di una risorsa



“PUT”



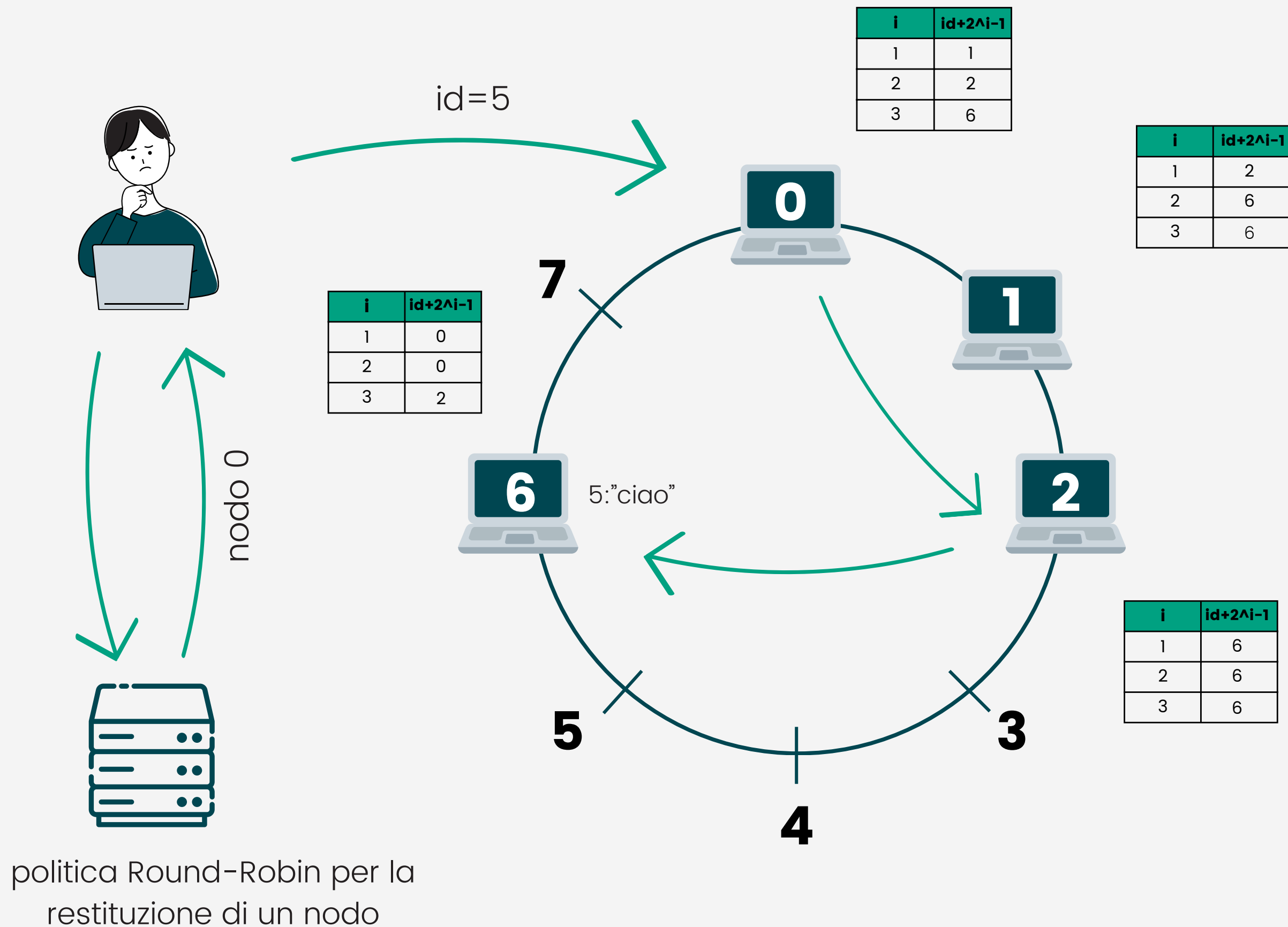
Il nodo contattato verifica se l'ID della risorsa appartiene al suo intervallo di ID:

- se sì, salva la risorsa.
- in caso contrario, verifica se l'ID appartiene all'intervallo di responsabilità del suo successore.

Se l'ID appartiene all'intervallo del successore:

- la richiesta viene inviata al successore.
- altrimenti, utilizza la finger table per instradare la richiesta al nodo corretto, ossia al nodo "j" dove vale la relazione: $FT[j] \leq ID < FT[j+1]$.

“GET”



Il nodo contattato verifica se l'ID della risorsa appartiene al suo intervallo di ID:

- se sì, restituisce la risorsa.
- in caso contrario, verifica se l'ID appartiene all'intervallo di responsabilità del suo successore.

Se l'ID appartiene all'intervallo del successore:

- la richiesta viene inviata al successore.
- altrimenti, utilizza la finger table per instradare la richiesta al nodo corretto, ossia al nodo "j" dove vale la relazione: $FT[j] \leq ID < FT[j+1]$.

Grazie per l'attenzione!

[https://github.com/
marinacalcaura/pr
ogetto_sdcc.git](https://github.com/marinacalcaura/progetto_sdcc.git)