

BITTORRENT

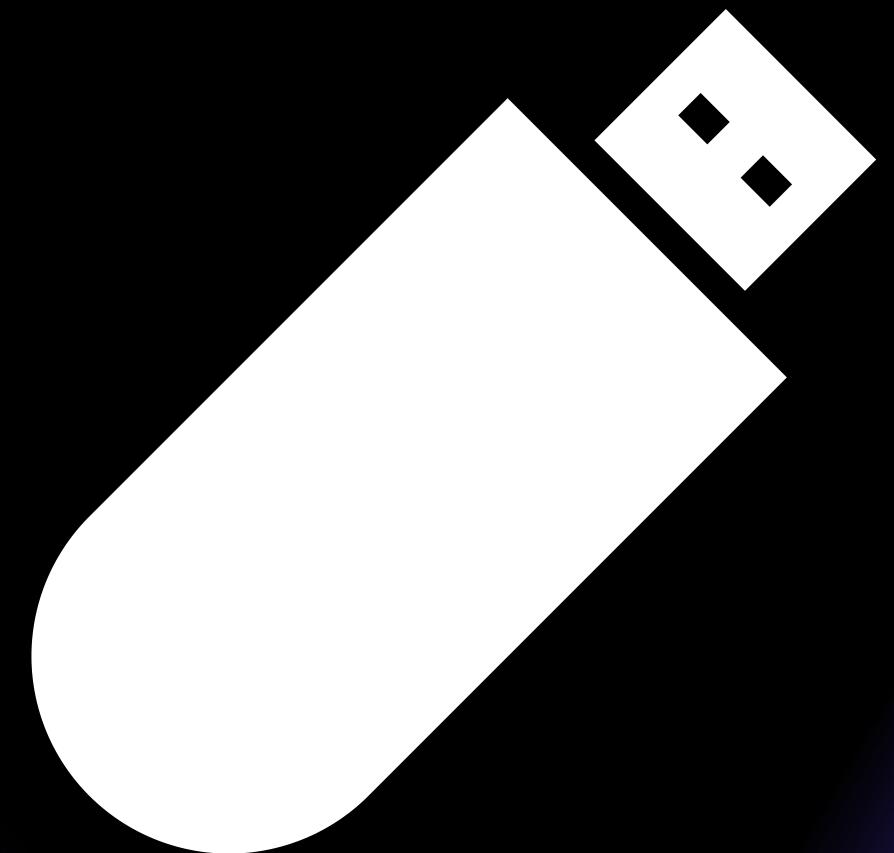
多人案傳輸的好選擇 資工二

BITTORRENT是啥？

是一種下載協議

常見傳檔案的方式：

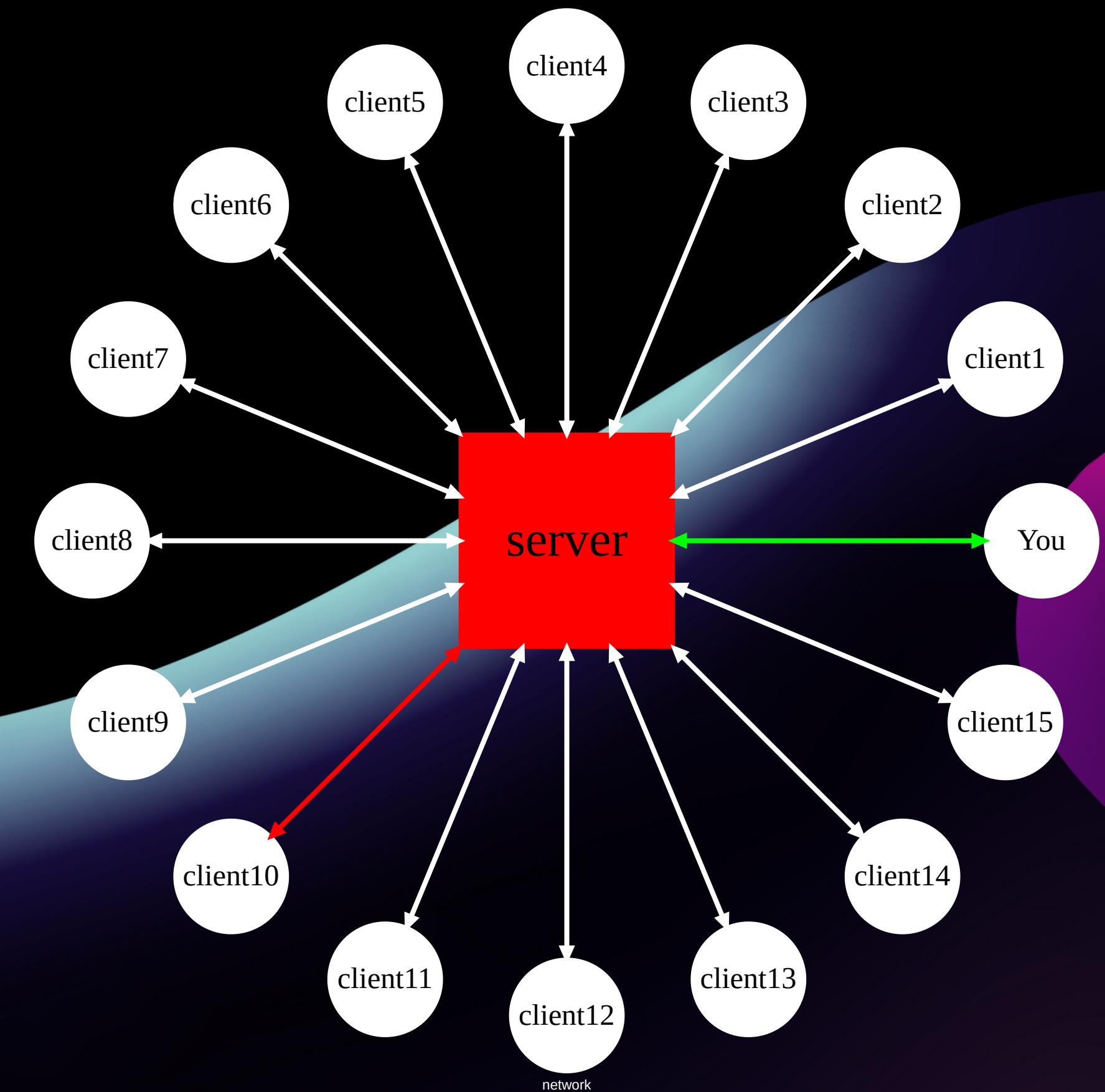
1. 網路雲端
2. 通訊軟體
3. USB



傳統網路架構

集中式網路

如果對方的網路上傳很慢
或是SERVER 壞了
就無法讀取

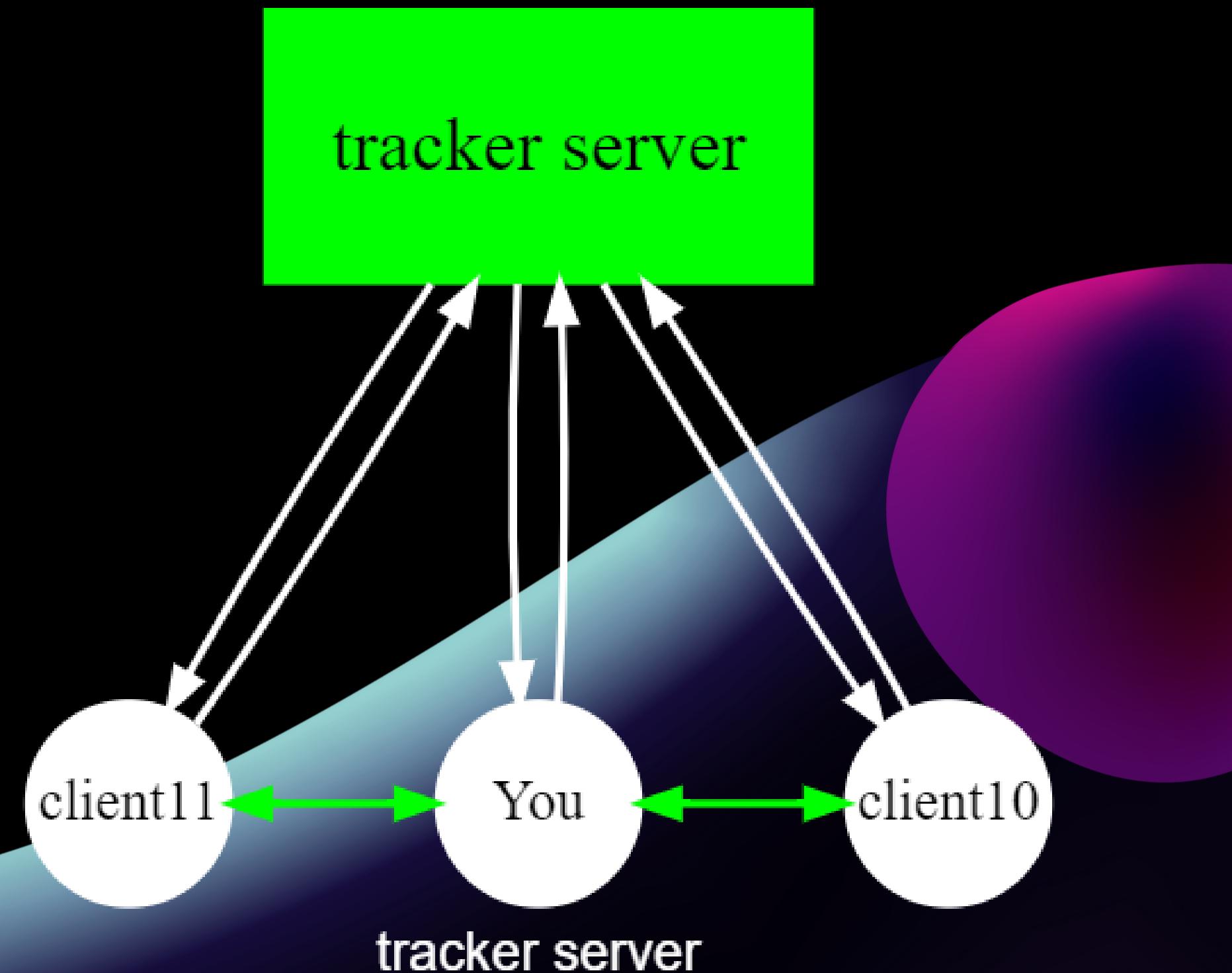


Tracker server

分布式網路

在藉由tracker server互相連線後
就直接交換IP 位置
不再透過server 進行

缺點:還是需要和server連線

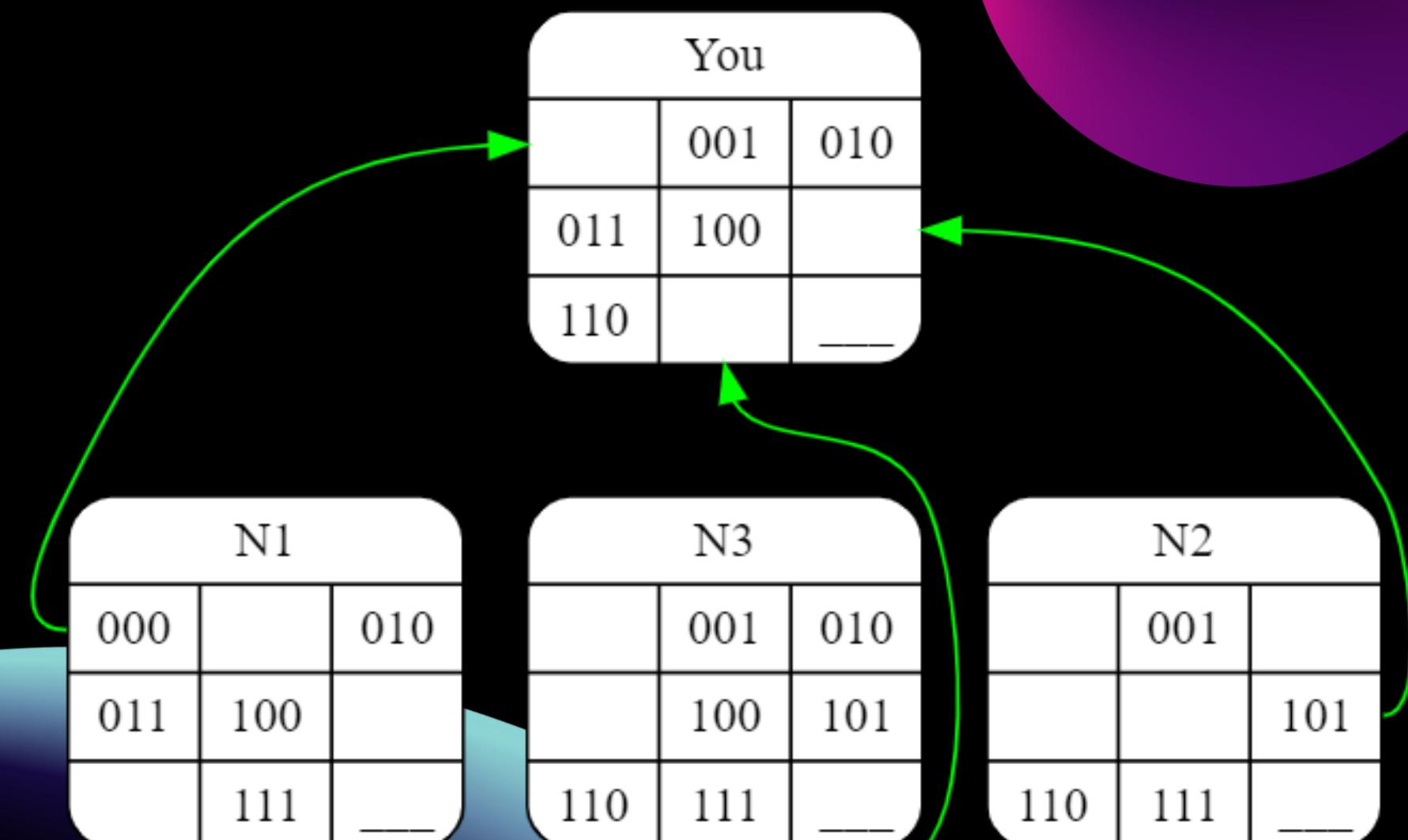


檔案切分

分布式網路

多人傳輸時有人檔案還在下載無法上傳給你？

把檔案細切割成多個小塊最後再組裝起來



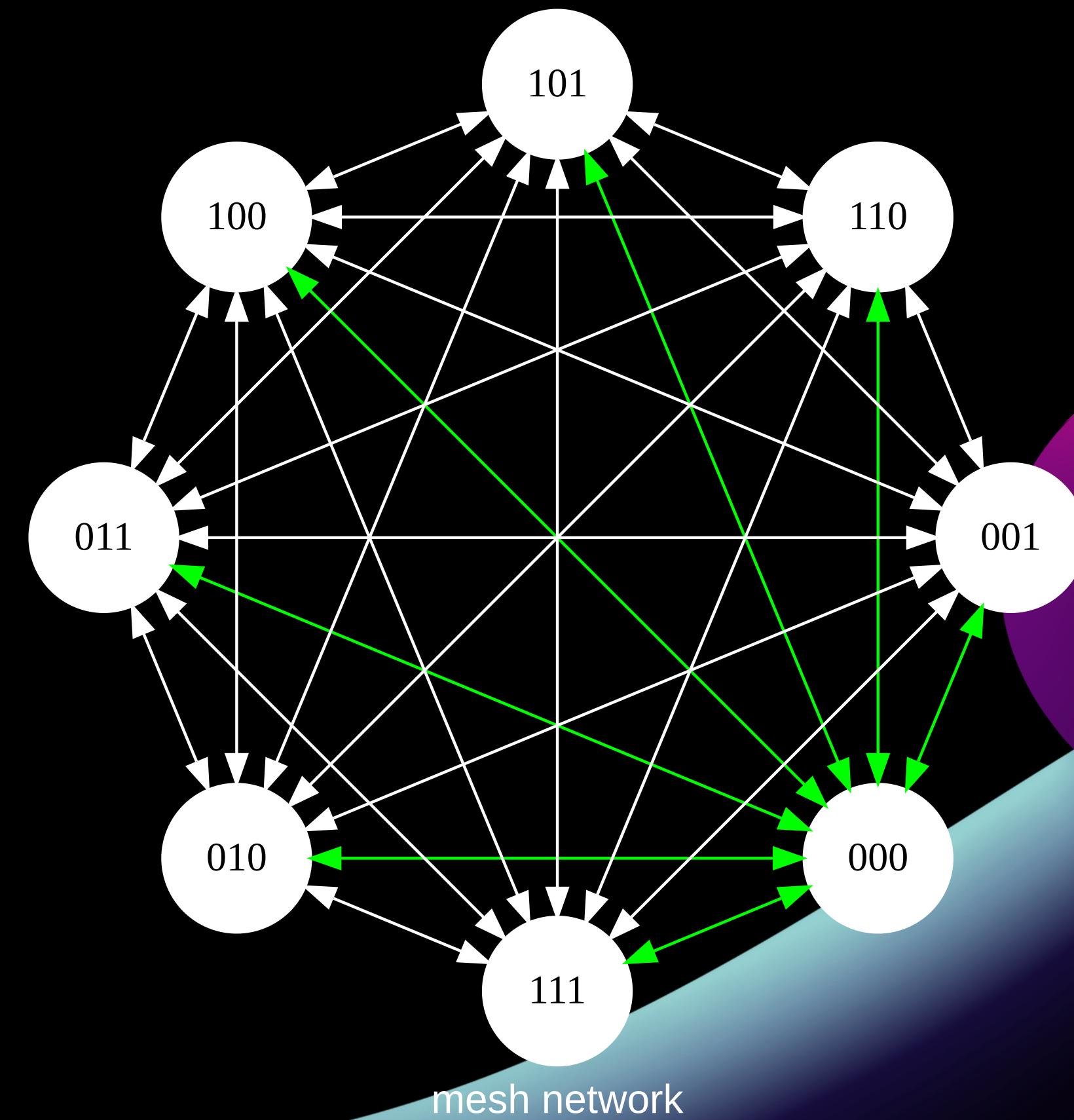
P to P網路架構

分布式網路

每個節點都互相知道其他節點

減少路徑上的所需要經過節點

在多人時可同時從其他節點下
載提高速度

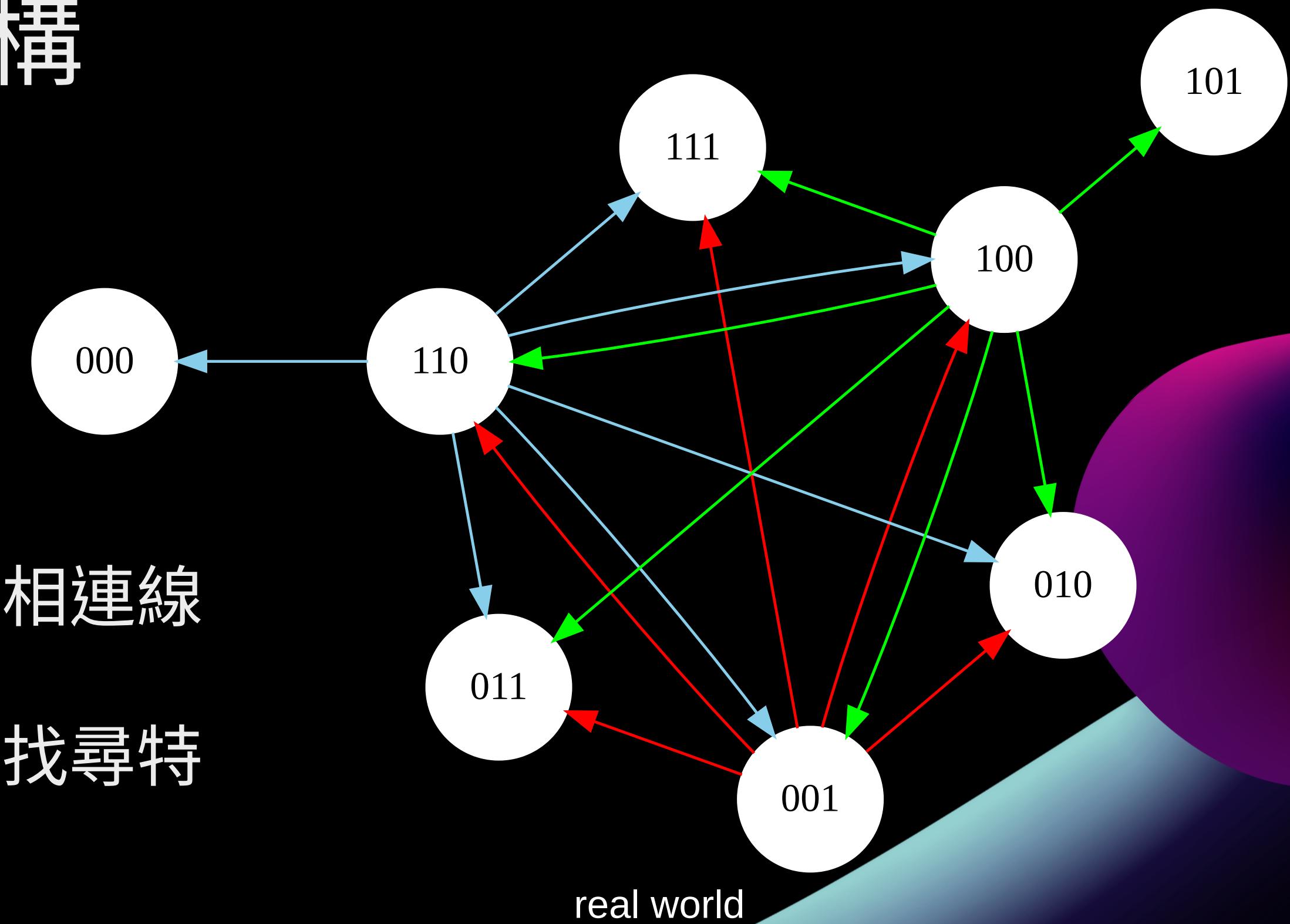


P to P網路架構

分布式網路

現實上每個節點時常變動
所以不太可能給個節點都互相連線

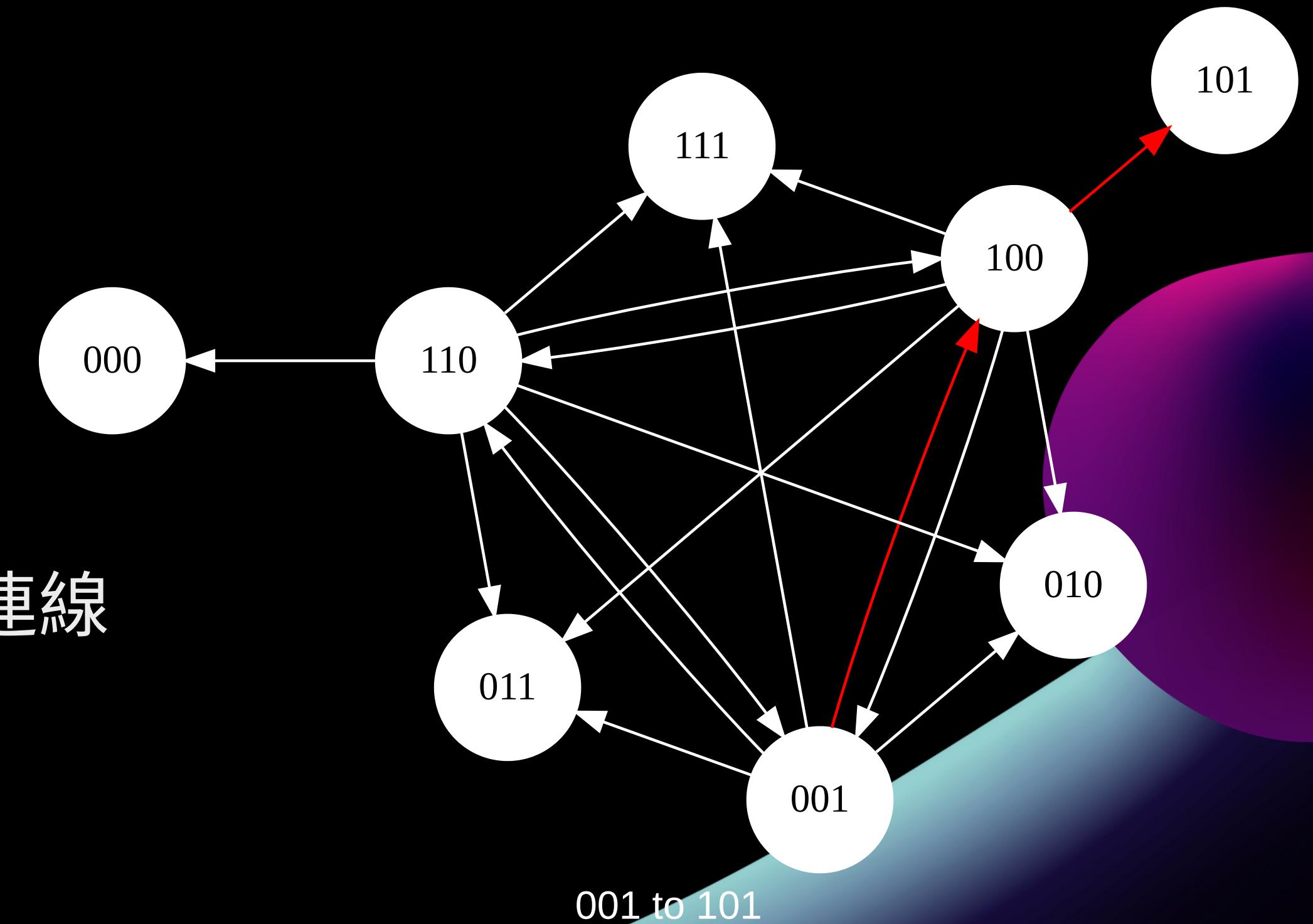
以及要如何從動態的節點中找尋特
定節點？



Kademlia 網路

分布式網路

現實上每個節點時常變動
所以不太可能給個節點都互相連線



index

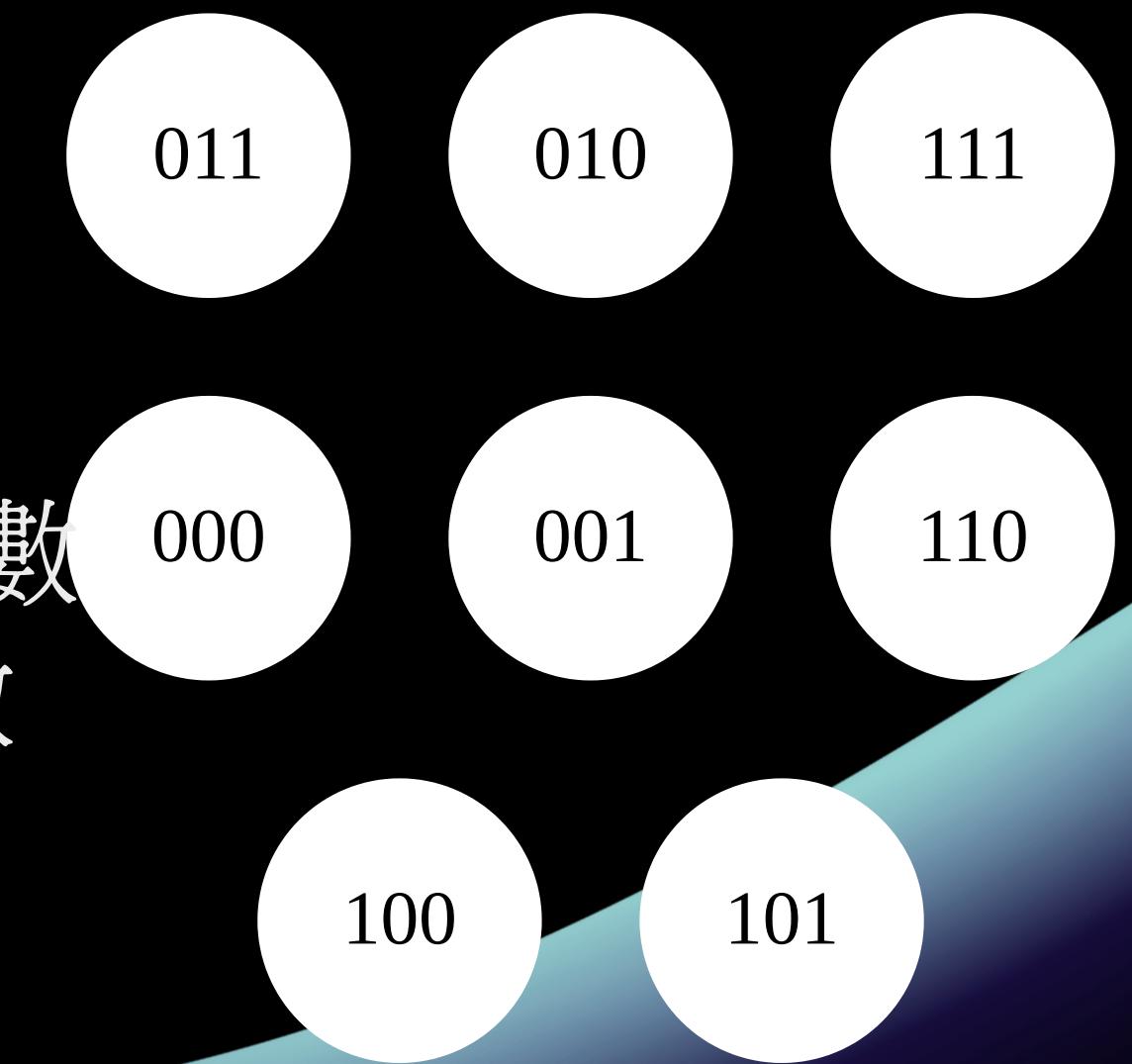
Kademlia

每個節點需要獨立的index

通常使用hash產生

Kademlia SHA1: 160 bit = 48位數

Ethereum sha3: 256 bit = 77位數

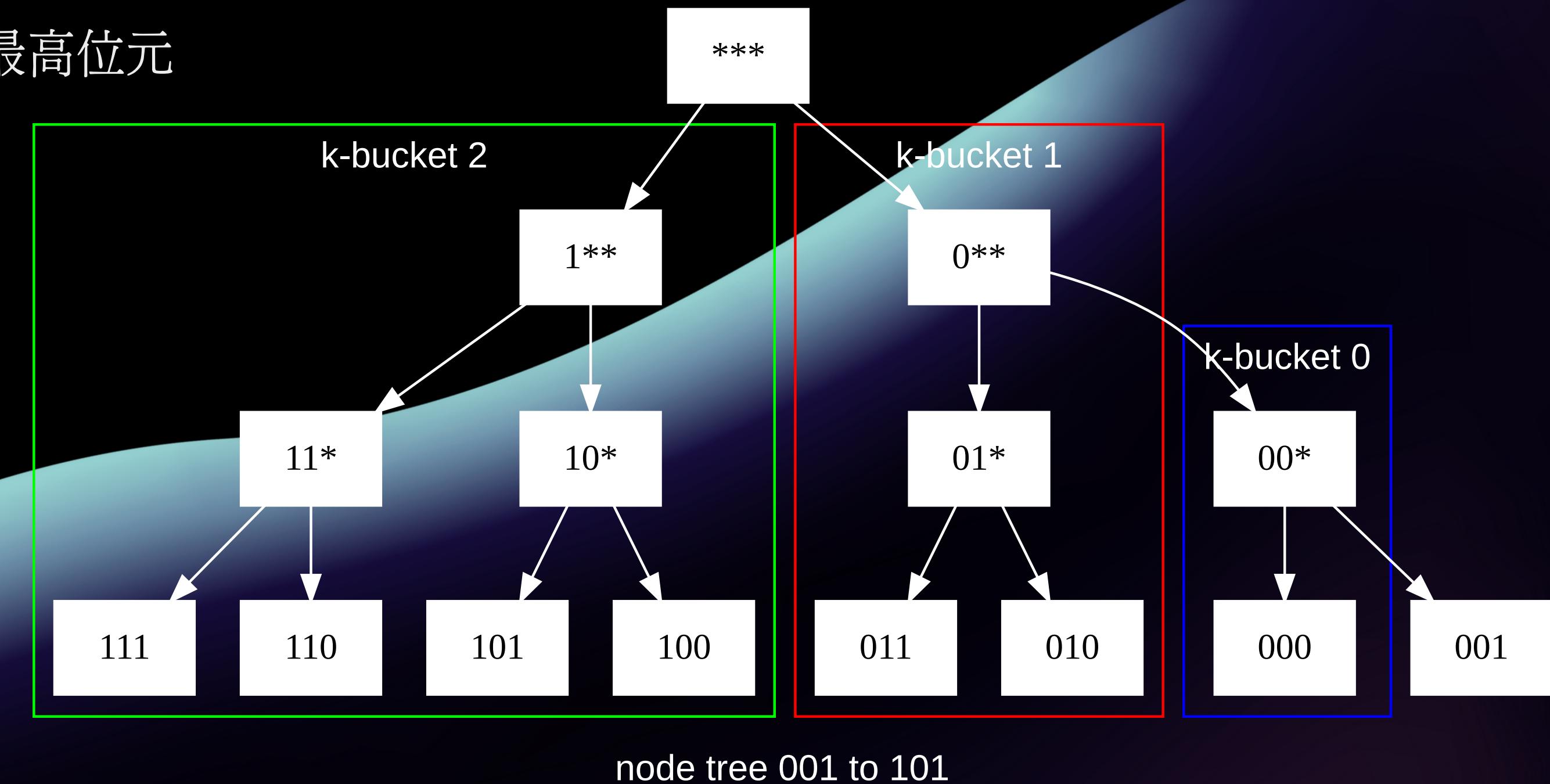


距離 和 k-bucket

Kademlia k-bucket

距離:節點id互相xor的最高位元

$001 \text{ xor } 101 = 100$
最高位元為2
所以在 k-bucket 2



距離 和 k-bucket

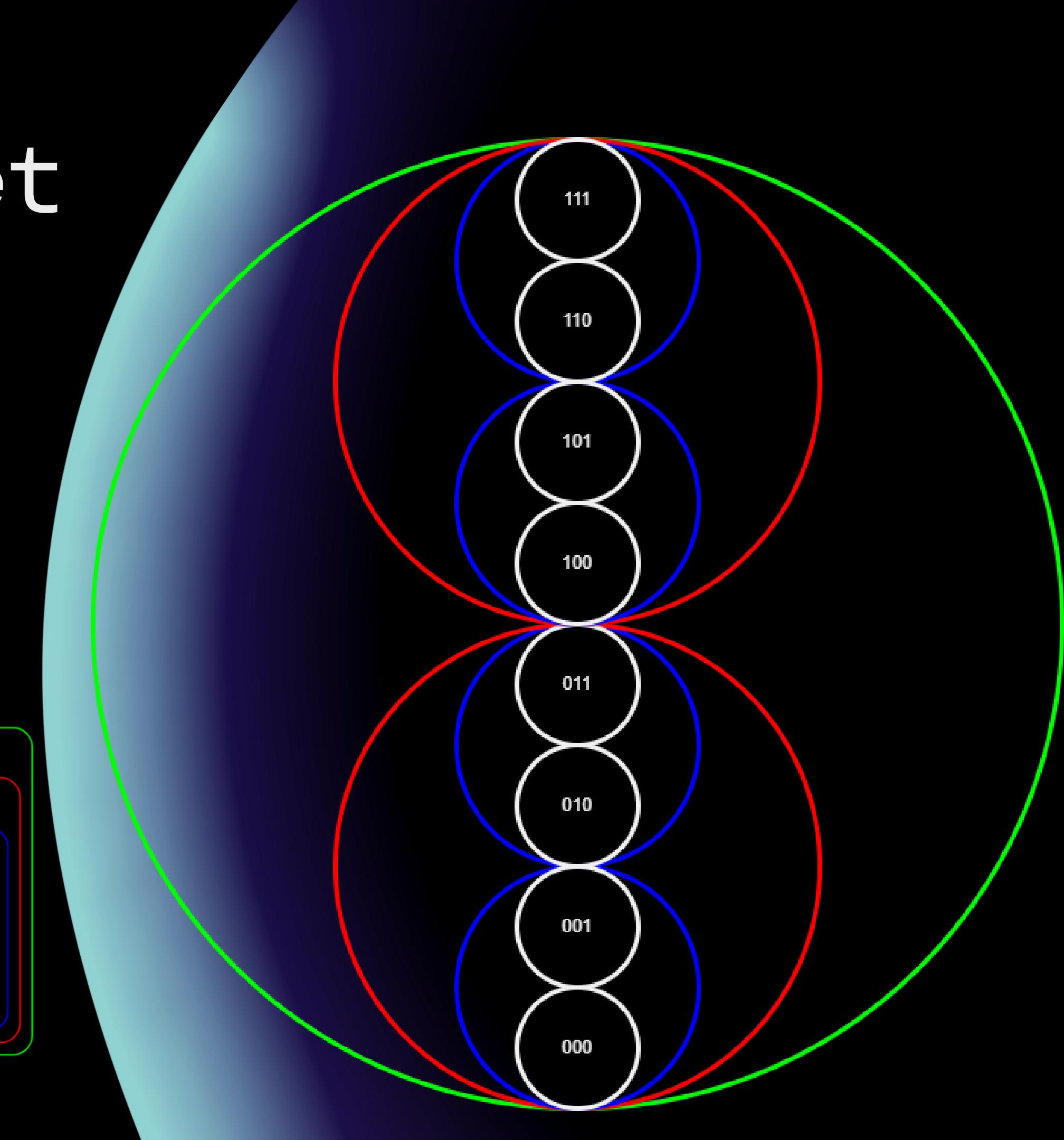
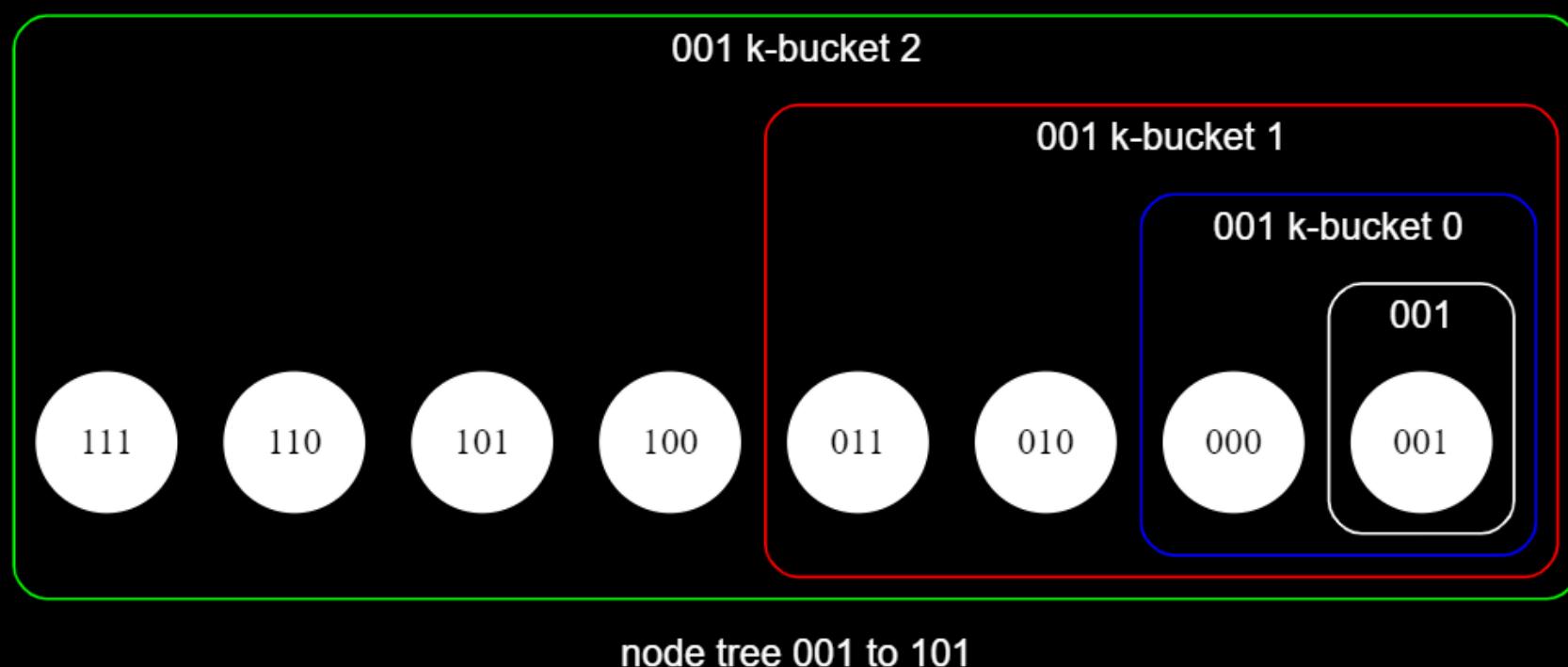
Kademlia xor

xor 的意義

$A \text{ xor } B == B \text{ xor } A$

硬體執行速度快

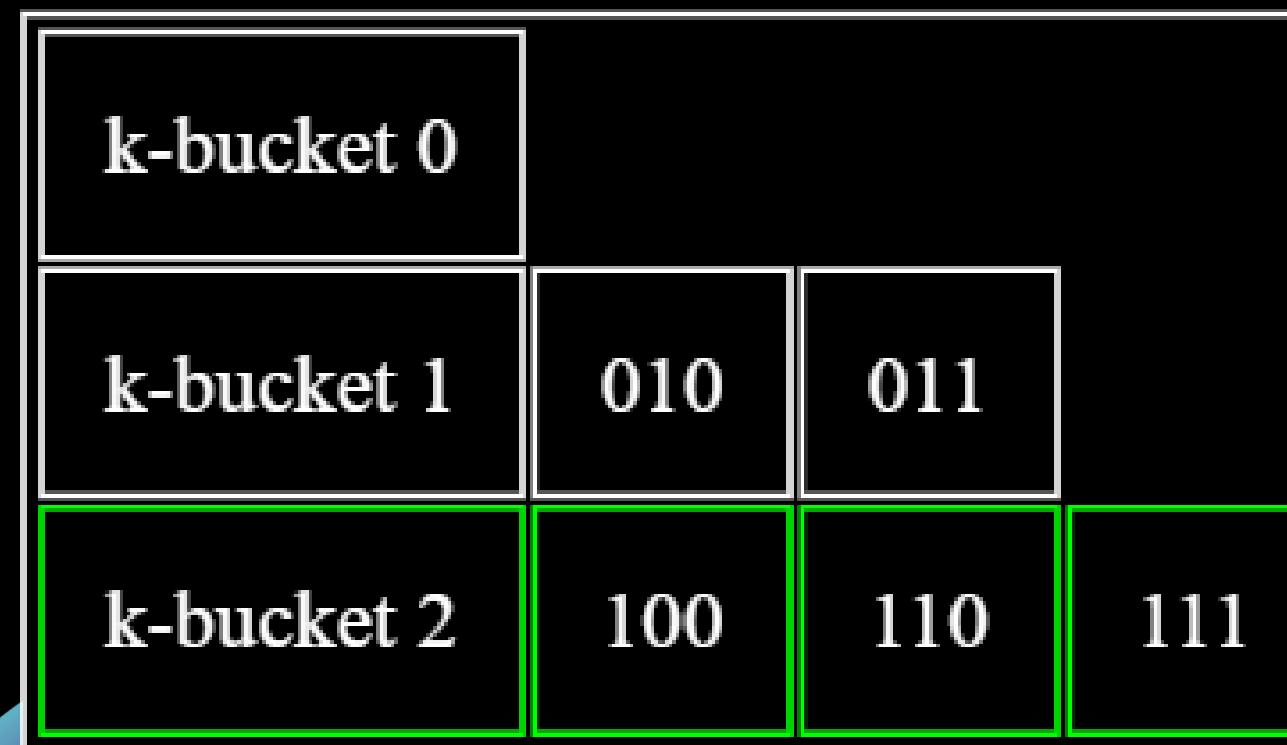
距離與節點數成指數性關係



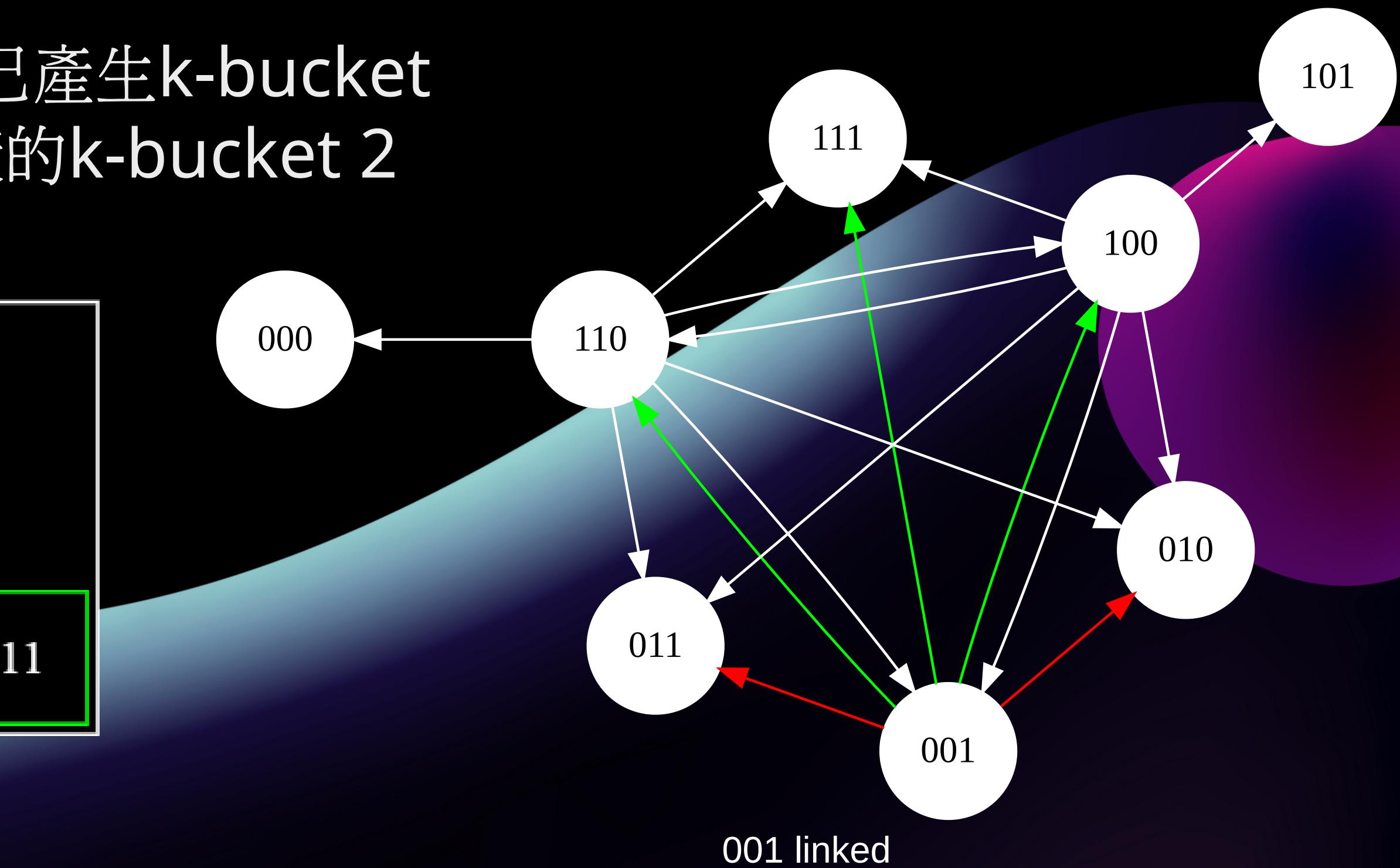
距離 和 k-bucket

Kademlia 001 to 101

用該節點已知的節點和自己產生k-bucket
選擇和目標節點 距離一樣的k-bucket 2



k-bucket table for 001



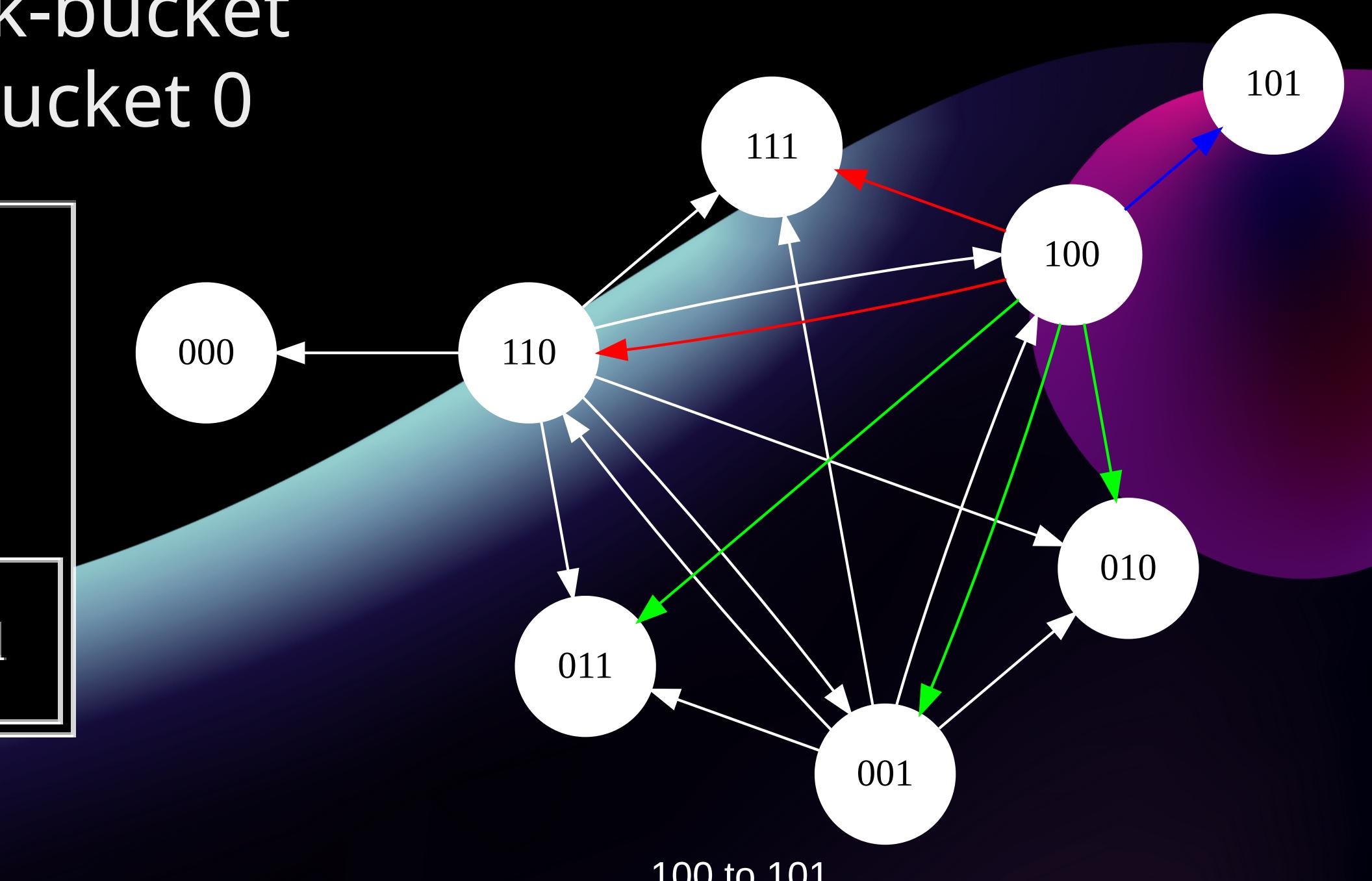
距離 和 k-bucket

Kademlia 100 to 101

用該節點已知的節點和自己產生k-bucket
選擇和目標節點 距離一樣的k-bucket 0

k-bucket 0	101		
k-bucket 1	110	111	
k-bucket 2	001	010	011

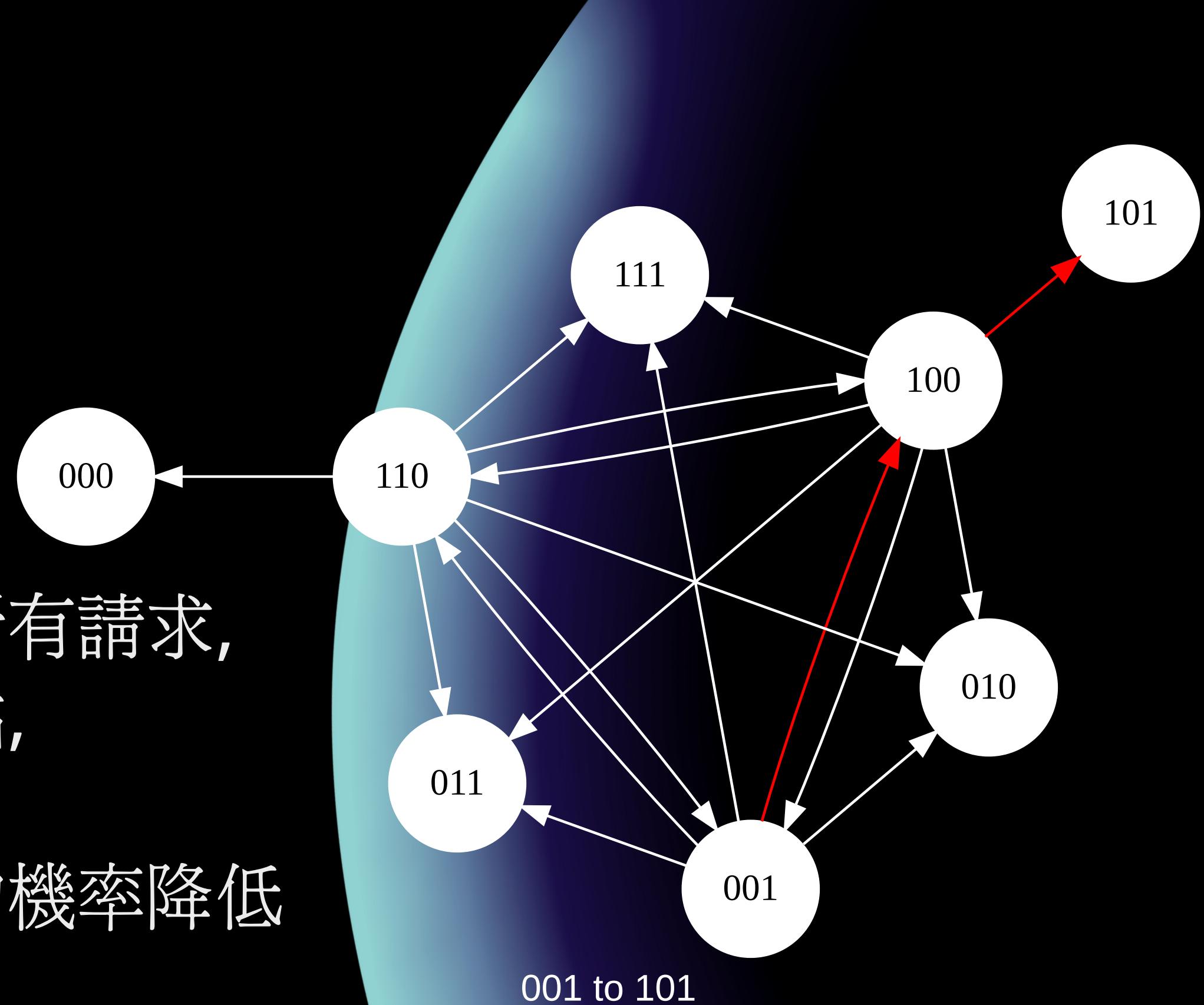
k-bucket table for 100



DHT嗅探器

Kademlia 資安

可以輕易偽裝成節點的記錄下所有請求，
由於每個請求都會經由其他節點，
無法具體知道請求來源，
且隨著節點數量增加攔截封包的機率降低



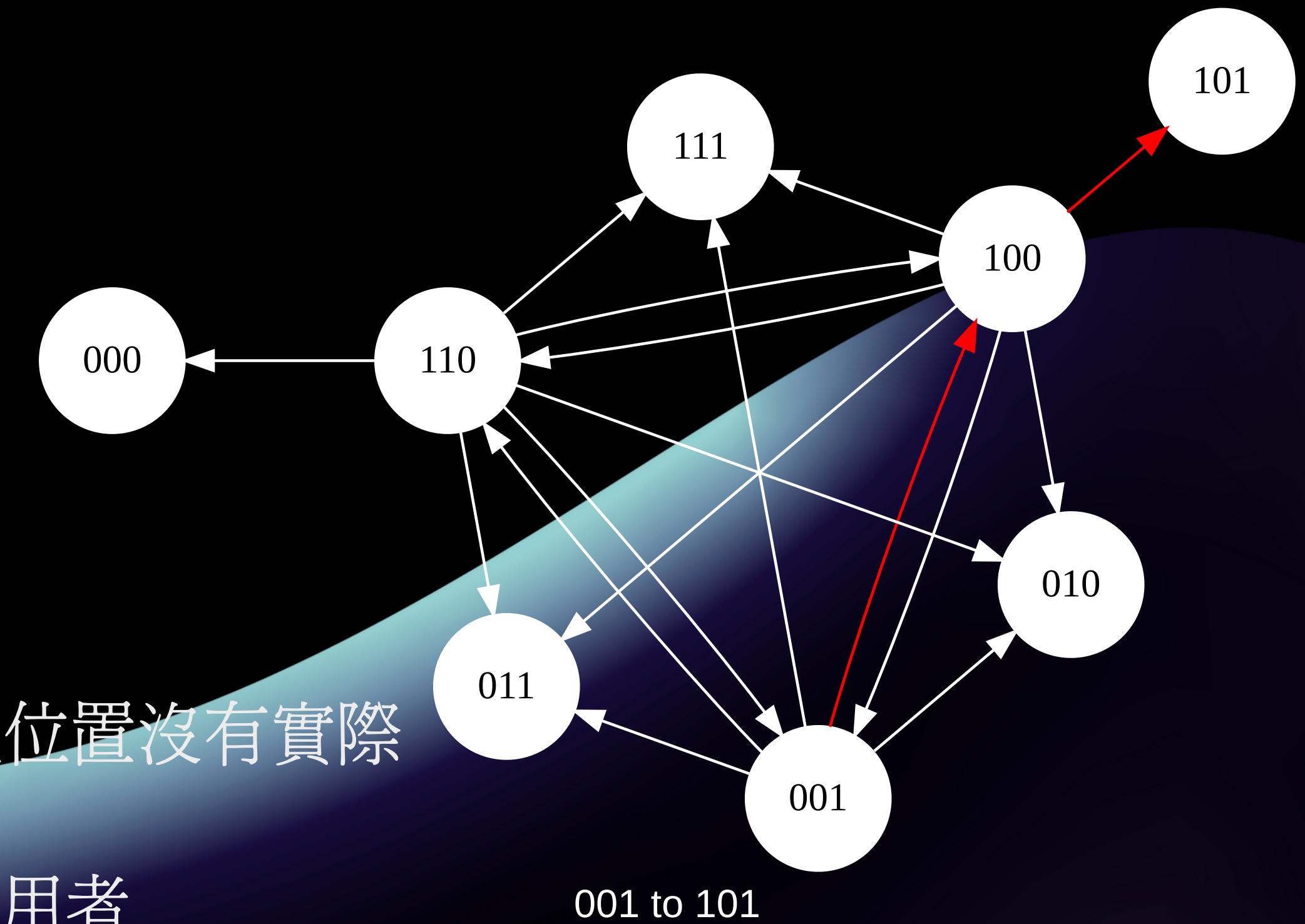
Kademlia 總結

優點：

- 搜尋速度快 $O(\log_2(n))$
- 匿名性
- 有效避免 DDOS 攻擊

缺點：

- 節點 id 經由 hash 產生與物理位置沒有實際關聯
- leecher：只下載不上傳的使用者



應用 distributed hash table

彼特幣: 找尋於其他地方的鏈

Bittorrent: 查找檔案和IP

等有關去中心化技術都有使用



實際使用

The Pirate Bay

海盜灣:是一個專門儲存、分類及搜尋
Bittorrent種子檔案及磁力連結的網
站，由瑞典的民間反著作權組織海盜署
於2003年成立





實際使用

bittorrent

File Edit Torrent View Help

Show: All 1 ▾ All 25 ▾ Search...

Black Adam (2022) - 1080p - HQ Final Print Rip - 1500MB - x264 - AAC- QRips.mkv
1.09 GB of 1.68 GB (65.1%) - 9 minutes left

Downloading from 30 of 50 connected peers - 1.28 MB/s ▾ 585 kB/s ▾

1.28 MB/s ▾ 585 kB/s ▾ Ratio: 0.1

Information						
	Up	Down	%	Status	Address	Client
🔒	274 kB/s	274 kB/s	100%	TDEXI	186.96.212.85	qBittorrent 4.4.5
🔒	195 kB/s	195 kB/s	100%	TDEX	86.85.6.27	Deluge 2.0.3.54
🔒	127 kB/s	127 kB/s	100%	DEX	72.21.17.10	libTorrent (Rakhsasa) 0.13.8
🔒	125 kB/s	125 kB/s	100%	TDEH	207.191.244.176	µTorrent 3.5.5
🔒	118 kB/s	118 kB/s	100%	TDEI	91.148.228.67	Deluge 2.1.1.54
🔒	44 kB/s	44 kB/s	100%	TDEX	212.102.35.156	Deluge 2.1.1.54
🔒	44 kB/s	44 kB/s	99%	TDUEX	185.240.244.188	qBittorrent 4.1.7
🔒	8 kB/s	39 kB/s	80%	TDUEX	136.158.66.136	-Lr3400-
🔒	16 kB/s	23 kB/s	43%	TODUI	61.9.101.82	-Lr1100-
🔒	5 kB/s	15 kB/s	73%	TDUEX	102.67.178.248	-Lr1100-
🔒	12 kB/s	12 kB/s	100%	TDEX	197.15.249.50	qBittorrent 4.4.3
🔒	12 kB/s	12 kB/s	100%	TDEX	102.66.86.249	FD6mIKUL
🔒	4 kB/s	8 kB/s	25%	TDUEX	41.86.43.62	-Lr1100-
🔒	8 kB/s	6 kB/s	28%	TDUEX	200.5.117.73	-Lr1100-
🔒	4 kB/s	4 kB/s	23%	TDUEX	41.78.203.26	-Lr1100-
🔒	1 kB/s	1 kB/s	80%	TDUEX	200.90.65.192	BitTorrent 7.0.5
🔒			36%	TDUEX	197.101.172.117	-Lr1100-
🔒			10%	T	T: Peer is connected over uTP	4.4.3
🔒			100%	T	D: Downloading from this peer	4.4.3
🔒			23%	T	U: Uploading to peer	4.4.3
🔒			100%	T	E: Encrypted connection	4.4.3
🔒	2 kB/s		7%	T	X: Peer was discovered through Peer Exchange (PEX)	4.4.3
🔒			100%	TDEX	142.115.108.57	Transmission 3.00
🔒	4 kB/s		37%	TDUEX	181.214.218.73	qBittorrent 4.4.5
🔒			41%	TDuEX	154.72.168.104	-Lr1100-
🔒			56%	TDUEX	136.158.26.89	-Lr1100-
🔒			27%	TDuEX	122.167.41.153	qBittorrent 4.4.3