



Lezione 1
SISTEMI P2P:
INTRODUZIONE
MATERIALE DIDATTICO:
CAPITOLO 1 BUFORD

Laura Ricci
23-2-2010

INFORMAZIONI UTILI

L'esame può essere sostenuto dagli studenti di:

- Lauree Specialistiche in Informatica, Tecnologie Informatiche, Informatica per l'Economia e l'Azienda.
- Lauree Magistrali in Informatica, Informatica per l'Economia e l'Azienda
- Laurea triennale (26): utilizzare i 9 crediti liberi, se non già utilizzati per altri esami/seminari.
- Laurea Triennale (L31)

Prerequisiti

- Reti di Calcolatori
- Algoritmica

MATERIALE DIDATTICO

Materiale Didattico:

- lucidi delle lezioni (scaricare quelli di quest'anno)
- tutorial e materiale pubblicato sulla pagina del corso
- **Libro di Testo:**
P2P Networking and Applications, Jhon F. Budford, H. Yu, E.K. Lua, Morgan Kaufmann

Testi di Consultazione

- Peer to Peer Systems and Applications , R. Steinmetz, K. Wehrle, LNCS. 3485, Springer Verlag, 2005
- Peer to Peer Computing, Principles and Applications, Q. Hieu Vu, M. Lupu, B. Chin Ooi, Springer Verlag, 2010

SVOLGIMENTO DEL PROGETTO

Modalità di esame: Scritto o progetto (a scelta) + orale obbligatorio per tutti

Durante il corso verranno proposti diversi progetti, che possono essere realizzati mediante i seguenti supporti

- Peersim
 - simulatore orientato alla simulazione di reti P2P
 - altamente scalabile (fino a 10000 peer per semplici protocolli)
 - <http://peersim.sourceforge.net/>
- Overlay Weaver
 - Supporta diversi algoritmi di routing
 - Definisce una semplice interfaccia verso diverse **distributed hash tables**
 - <http://overlayweaver.sourceforge.net/>
 - La simulazione di una rete P2P richiede l'implementazione di un insieme di interfacce/l'estensione di un insieme di classi JAVA

SVOLGIMENTO DEL PROGETTO

- In alternativa è possibile progettare una rete P2P 'reale' e testarne il comportamento sulla rete locale del polo e/o su rete geografica.
- In questo caso è possibile utilizzare JXTA
 - Un supporto per lo sviluppo di applicazioni P2P
 - Estensione di JAVA: gruppi, advertisements, pipes
 - <https://jxtadev.java.net/>
- Esempi di proposte di progetto (anni precedenti)
 - realizzazione di
 - una chat in JXTA
 - overlay strutturati in JXTA
 - simulazione di
 - reti P2P non strutturate
 - Distributed Hash Tables

SISTEMI PEER TO PEER: INTRODUZIONE

- Definizione: Un **sistema Peer to Peer** è un insieme di entità autonome (peers), capaci di **auto-organizzarsi**, che **condividono** un insieme di **risorse distribuite** presenti all'interno di una **rete di computers**. Il sistema utilizza tali risorse per fornire una determinata funzionalità in **modo completamente o parzialmente decentralizzato**.
- **Peer**= Pari, uguale (persona di pari grado, coetaneo)
- Risorse condivise:
 - Informazioni (Files)
 - Spazio di memorizzazione (Distributed File System)
 - Potenza di calcolo
 - Banda

CONDIVISIONE DI RISORSE

- P2P: riguarda il **dare e ricevere** da una comunità. Ogni peer fornisce una risorsa ed ottiene in cambio altre risorse.
 - situazione più comune: si offre musica al resto della comunità, si ottiene altra musica in cambio (Napster, Gnutella,...)
 - un peer ha sia funzionalità di client, che di server (funzionalità simmetrica = **SERVENT**)
- Ma un peer può decidere di **offrire gratuitamente** risorse, ad esempio per partecipare ad una 'giusta causa'.
 - Ricerca di vita extra-terrestre
 - Ricerca sul cancro
- Le risorse condivise si trovano **'ai bordi'** di Internet, cioè sono fornite direttamente dai peers, non esistono nodi 'special purpose' definiti solo per la loro gestione.

CONDIVISIONE DI RISORSE

- La connessione dei peers al sistema è **intermittente (transiente)**: le disconnessioni e le riconessioni al sistema sono frequenti
- Le risorse offerte dai peers vengono aggiunte e tolte **dinamicamente** al sistema
- Ad un peer può essere associato un indirizzo IP diverso per ogni diversa connessione al sistema
 - ⇒
E' impossibile localizzare una risorsa mediante un indirizzo IP statico
 - ⇒
Definizione di nuovi **meccanismi di indirizzamento**, definiti a livello superiore rispetto al livello IP

FILE SHARING: UNA 'KILLER APPLICATION'

- **Condivisione di files:** nasce con il rapido successo di Napster, alla fine degli anni 90, circa 10 anni dopo l'affermazione del World Wide Web
- **Prima generazione: Napster**
 - introduce un insieme di servers su cui gli utenti registrano i **descrittori** dei files che intendono condividere
 - la trasmissione dei contenuti (download/upload) avviene con modalità P2P
 - La presenza di directory centralizzate è stato il 'tallone di achille' di questa applicazione
 - Napster è stato giudicato responsabile per non aver rispettato le leggi sul copyright, proprio perchè avrebbe potuto individuare i contenuti scambiati illegalmente tra gli utenti mediante l'analisi delle directory centralizzate

FILE SHARING: UNA 'KILLER APPLICATION'

- Seconda generazione
 - Non esiste alcun punto di centralizzazione
 - Sia la ricerca dei files che il trasferimento del contenuto avviene in modo **completamente distribuito** tra i peer
 - Gnutella, FastTrack/Kazaa, BitTorrent
 - Freenet: introduce diversi meccanismi per garantire l'anonimato dei peer
 - Sostituzione del mittente di un messaggio
 - Tecniche di encrypting
- Effetti laterali della diffusione delle applicazioni di file sharing: modifica radicale della modalità di fruizione di contenuti musicali
 - dai CD a **musica online**
 - iTunes

FILE SHARING: UN'APPLICAZIONE KILLER

- Come funziona un'applicazione per la **condivisione di files musicali**
- Tipico utilizzo del sistema P2P
 - Un utente U possiede un applicativo P2P sul suo notebook
 - Si interconnette ad Internet in modo intermittente: ottiene un nuovo indirizzo IP per ogni diversa connessione
 - Offre alla comunità alcune canzoni, registrandole in una directory condivisa, associando ad ogni canzone alcune chiavi per individuarla (titolo, autore, data di pubblicazione,...)
 - U è interessato a reperire una canzone ed invia una query al sistema
 - L'applicativo visualizza informazioni circa gli altri peers che posseggono la canzone richiesta
 - U sceglie un peer P (vedremo in seguito in base a quale criterio)
 - Il file viene copiato dal PC di P a quello di U
 - Mentre U effettua il download, altri utenti possono effettuare upload di qualche file messo a disposizione da U

FILE SHARING: UN'APPLICAZIONE KILLER

Il software P2P consente:

- All'utente di definire una directory, nel proprio file system, dove memorizzare i files che vuole condividere con la comunità. Ogni altro peer può reperire files da quella directory
 - Il peer si comporta come un **web server**
- All'utente di copiare files dalle directory condivise dagli altri utenti
 - Il peer si comporta come un **client**
- Agli utenti di individuare il materiale richiesto, mediante queries sottoposte al sistema
 - Funzionalità analoga a **Google**
- L'applicativo sull'host dell'utente si comporta come un **servlet**

FILE SHARING: PROBLEMI

- Molti applicativi P2P
 - sono scaricabili gratuitamente dalla rete, ma possono contenere meccanismi di **spyware** o di **malware**,
 - in realtà quindi il software non è gratuito, ma viene pagato attraverso un'invasione della privacy dell'utente, spesso inconsapevole.
- **Spyware** = un software che
 - raccoglie informazioni riguardanti l'attività online di un utente (siti visitati, acquisti eseguiti in rete etc) **senza il suo consenso**
 - trasmette tali informazioni tramite la rete ad un'organizzazione che utilizza tali informazioni per trarne profitto, solitamente attraverso l'invio di pubblicità mirata.
- **Malware** = un software che può causare danni più o meno gravi al computer su cui viene eseguito.

FILE SHARING : PROBLEMI

Pollution (= Inquinamento)

- Viene immessa nella rete P2P una grossa mole di **materiale corrotto**
- I peer non riescono a distinguere il materiale integro da quello corrotto, scaricano e contribuiscono alla diffusione di materiale corrotto
- Il numero delle copie corrotte può superare quello di quelle integre
- Esistenza di vere e proprie pollution companies: **Overpeer**

Presenza di **free riders**

- Peer che utilizzano l'applicazione per scaricare contenuti dagli altri peer, ma non contribuiscono a fornire contenuti
- Soluzioni:
 - Meccanismo di incentivi (crediti in eMule)
 - Meccanismo per l'individuazione e la penalizzazione dei free riders
- **Peer churn**: presenza intermittente dei peer sulla rete

SISTEMI PEER TO PEER: OLTRE IL FILE SHARING

- P2P file sharing
 - Napster
 - Gnutella, KaZaa
 - eMule
 - BitTorrent
- Instant Messaging and Voice over P2P
 - IMP systems: Instant Message and Presence Applications
 - VoP2P: Skype
- P2P TV
 - Video Streaming applications
- Content Distribution Network (CDN)
- P2P distributed storage
 - OceanStore, Freenet
- P2P entertainment
 - Multiplayer games

IMP AND VoP2P

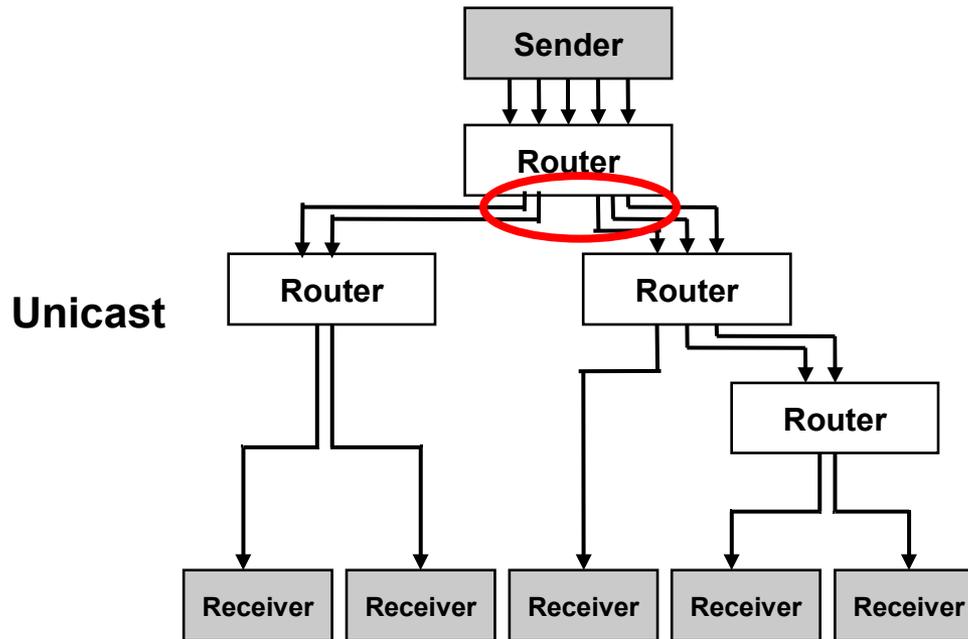
- IMP (Instant Messaging and Presence) applications sviluppate a partire dalla metà degli anni '90 sono spesso progettate secondo il modello client-server e
 - Possono offrire servizi di voce/video desktop to desktop
 - Microsoft Messenger
 - Yahoo!Messenger
 - Jabber
 - Scarsa diffusione del servizio voce/video
- Skype:
 - VoP2P client introdotto nel 2003
 - 10 milioni di utilizzatori negli ultimi anni
 - offre sia servizi di chiamata desktop-to-desktop che desktop-to-public switched-telephone-network
 - Buona qualità di voce è alla base del successo della applicazione
 - Include funzionalità tipiche di un'applicazione IMP

P2PTV: VIDEO STREAMING

- **File sharing**: consente di condividere anche file multimediali contenenti audio/video, tuttavia il video viene prima scaricato (almeno in parte) sul peer e visualizzato successivamente
- Video Streaming:
 - trasferimento e visualizzazione **in real time** dello stream video
 - **cooperazione tra i peer** per la distribuzione del contenuto (content distribution networks)
 - Bittorrent
 - bufferizzazione di un certo numero di video frames
- Applicazioni
 - Joost
 - Zattoo

CONTENT DISTRIBUTION NETWORKS

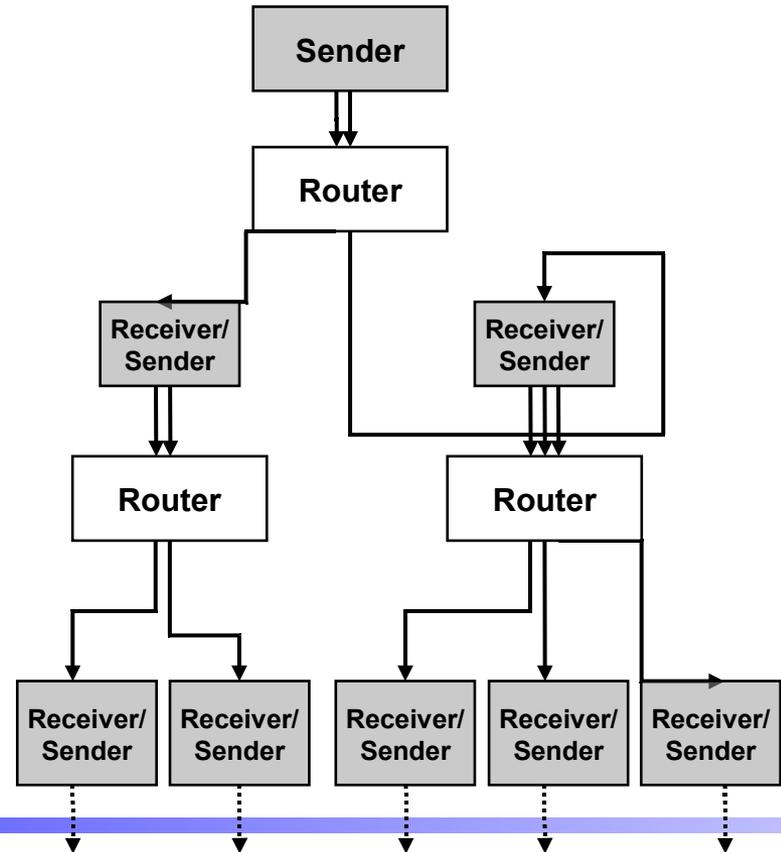
- Un server pubblica un nuovo contenuto (esempio: nuova versione di un gioco, nuova release di un sistema operativo,...)



- Modello client server: collo di bottiglia rappresentato dall'unico server centralizzato

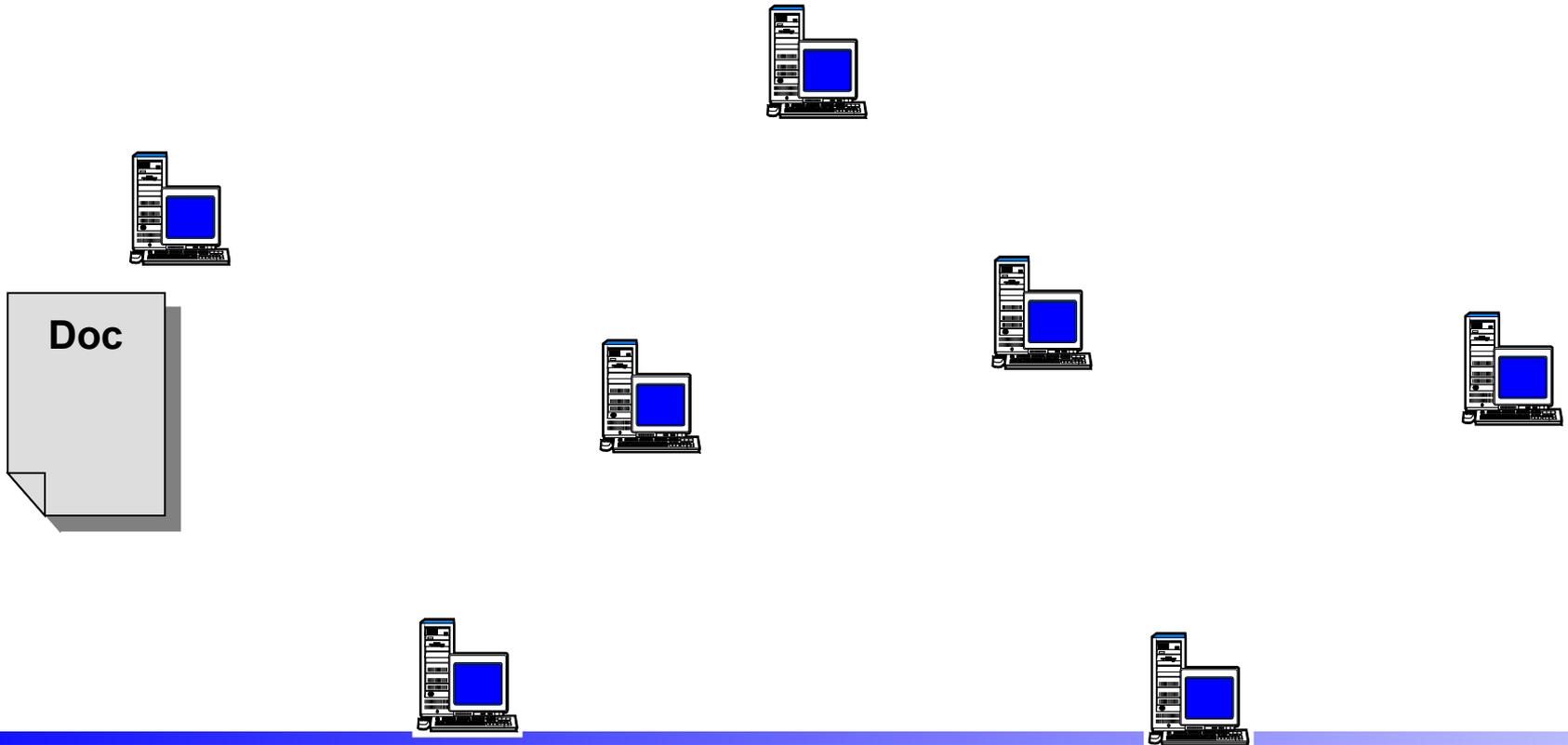
P2P CONTENT DISTRIBUTION

- Approcci P2P
 - Ottenere un miglior bilanciamento nell'uso della banda di comunicazione, utilizzando canali di trasmissione meno utilizzati
- Peer-to-Peer Content Distribution
 - Le richieste iniziali di files sono servite da un server centralizzato
 - Ulteriori richieste vengono inoltrate automaticamente ai peer che hanno già ricevuto e replicato, in precedenza, i files



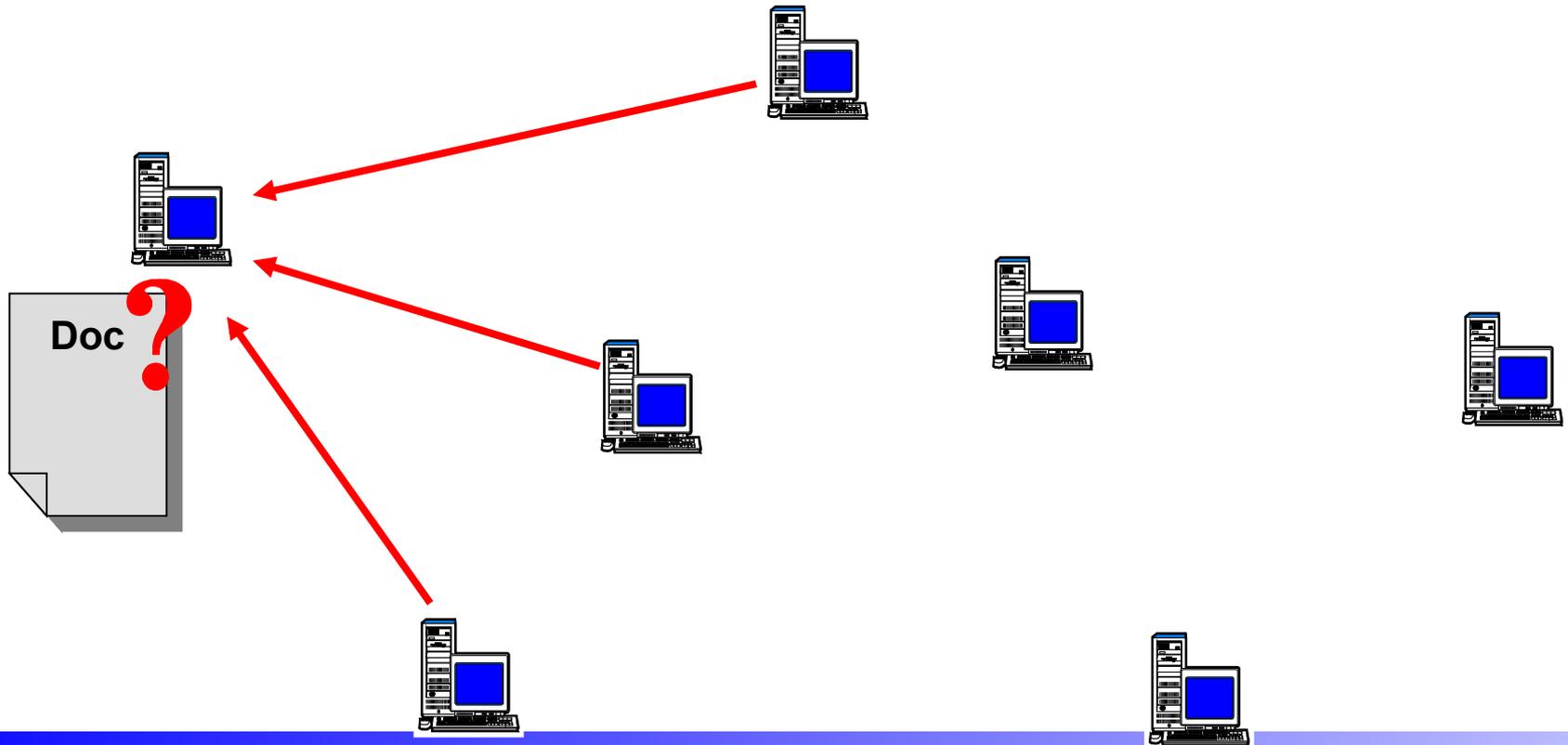
P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)



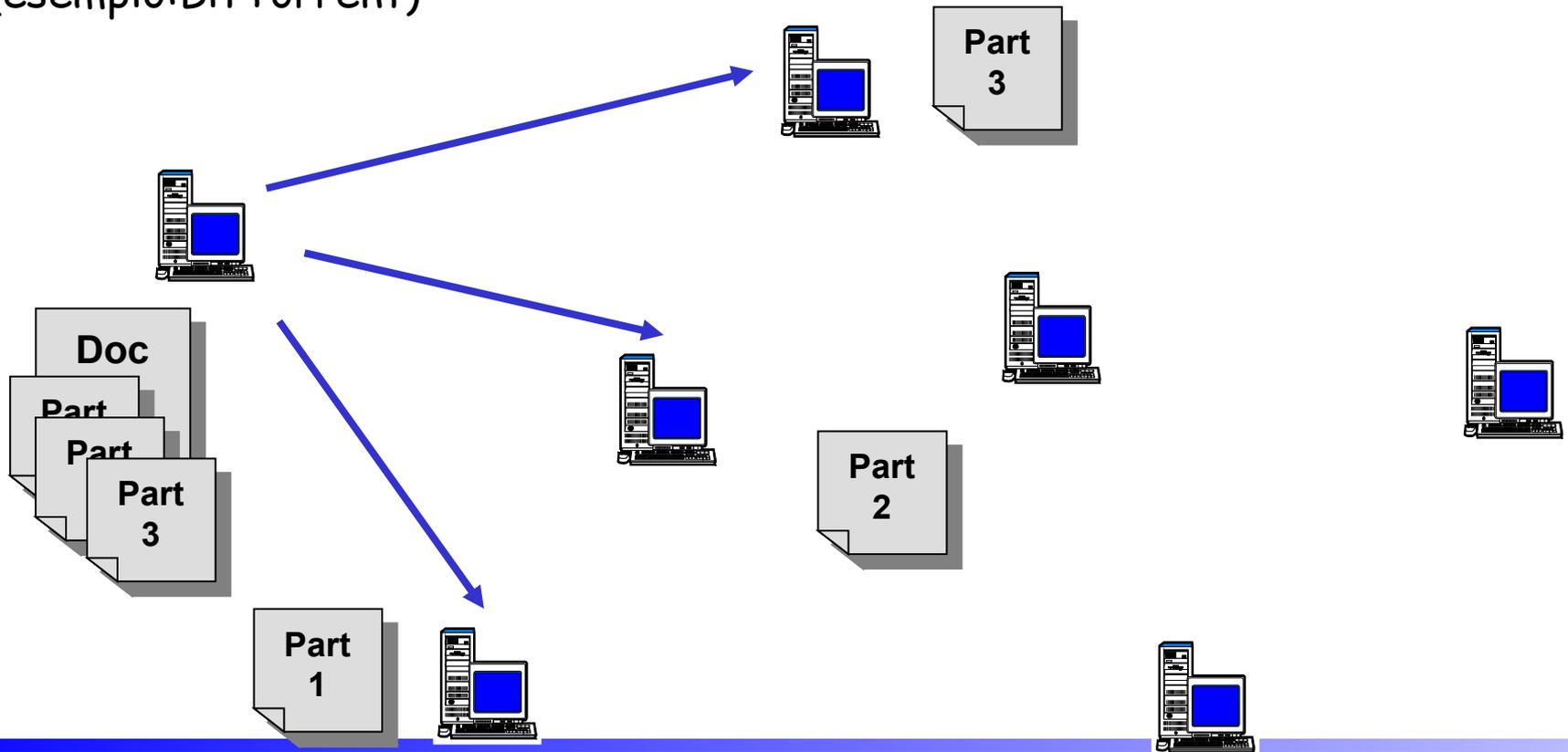
P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)



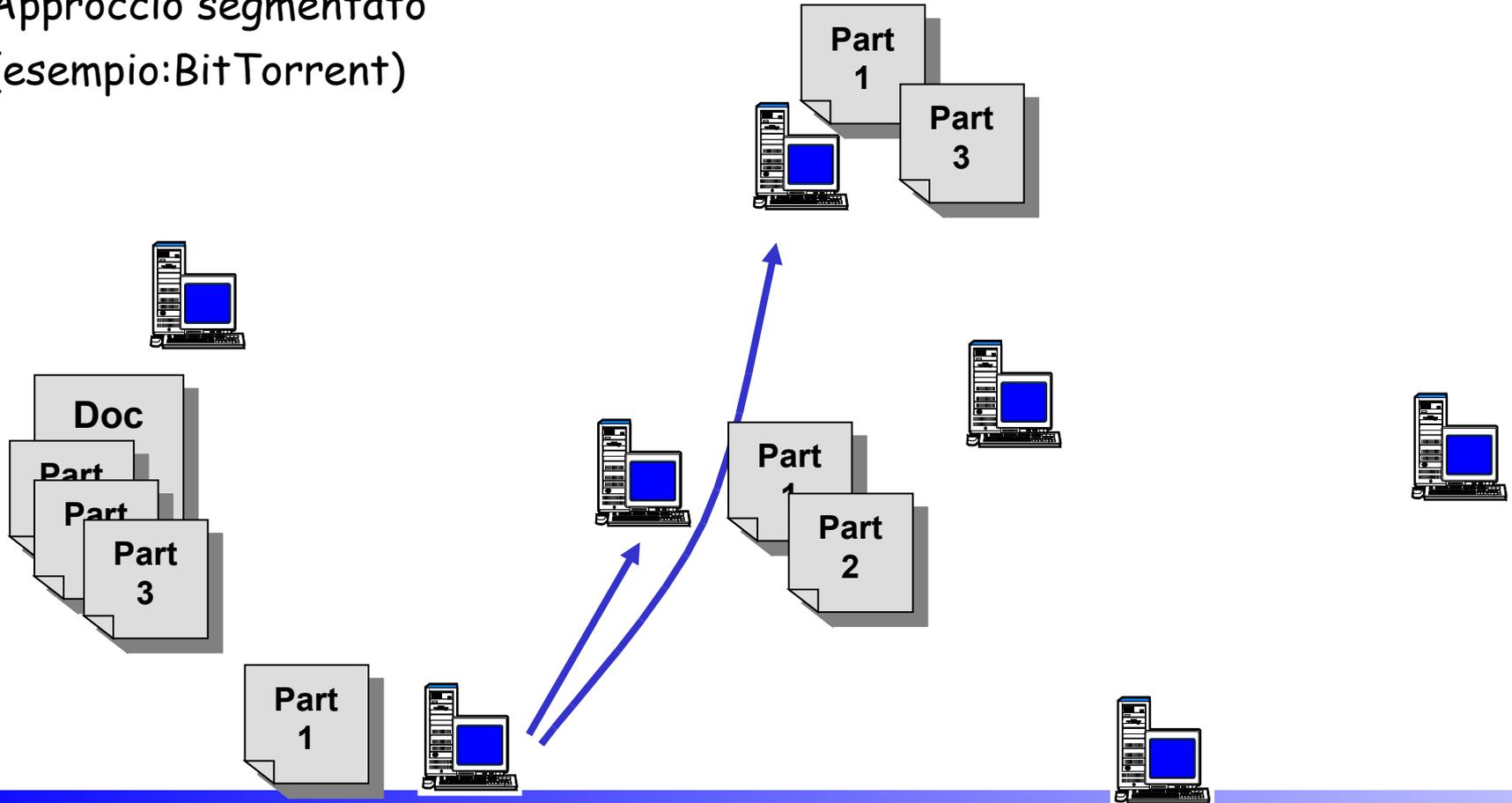
P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)



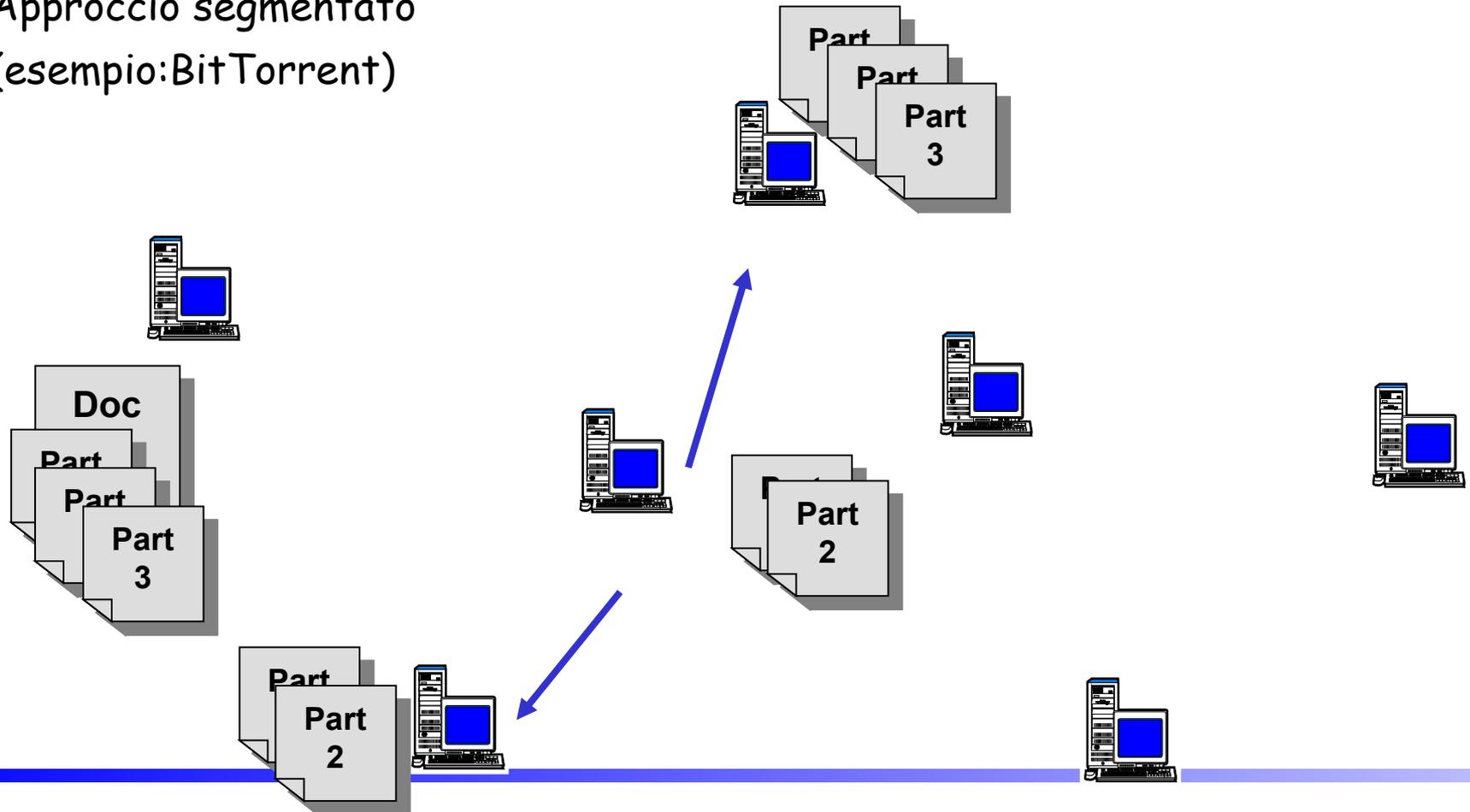
P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)



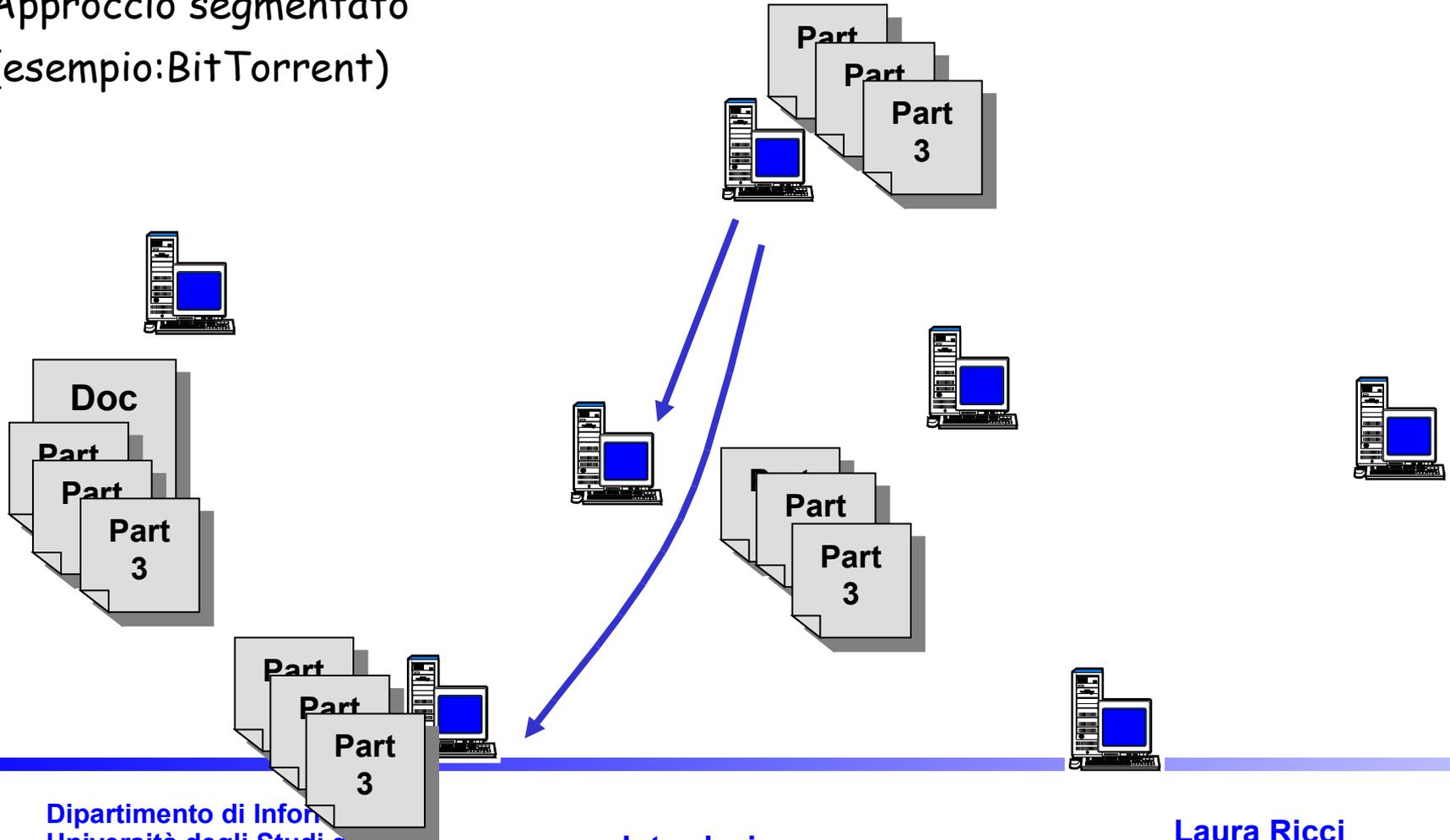
P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)



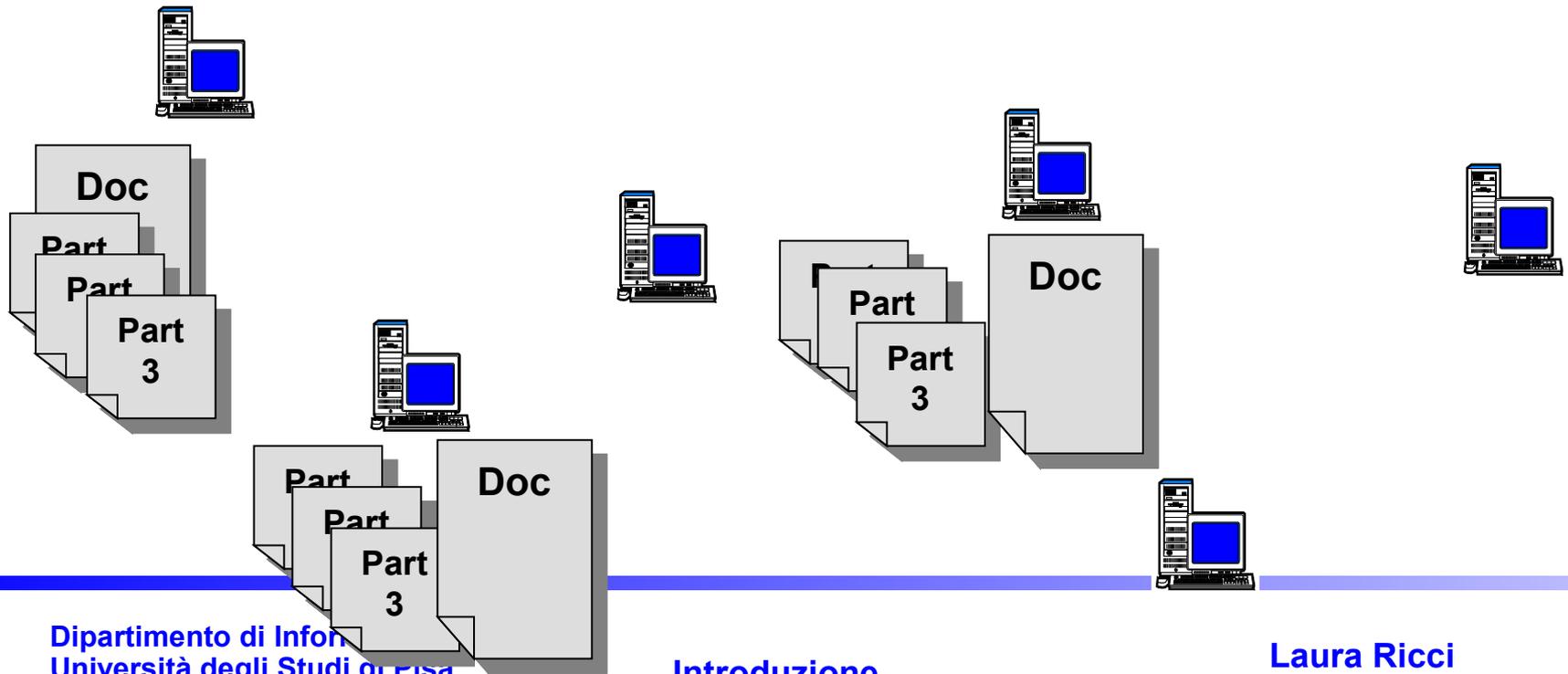
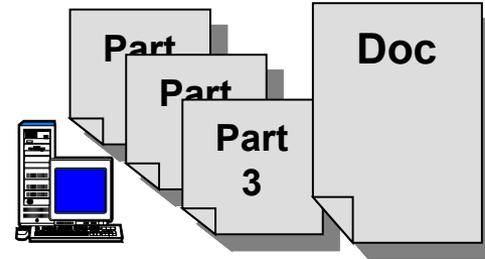
P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)



P2P CONTENT DISTRIBUTION

- Uso combinato del modello P2P e di quello client-server consente di ottimizzare gli accessi ad un server
- Approccio segmentato (esempio: BitTorrent)

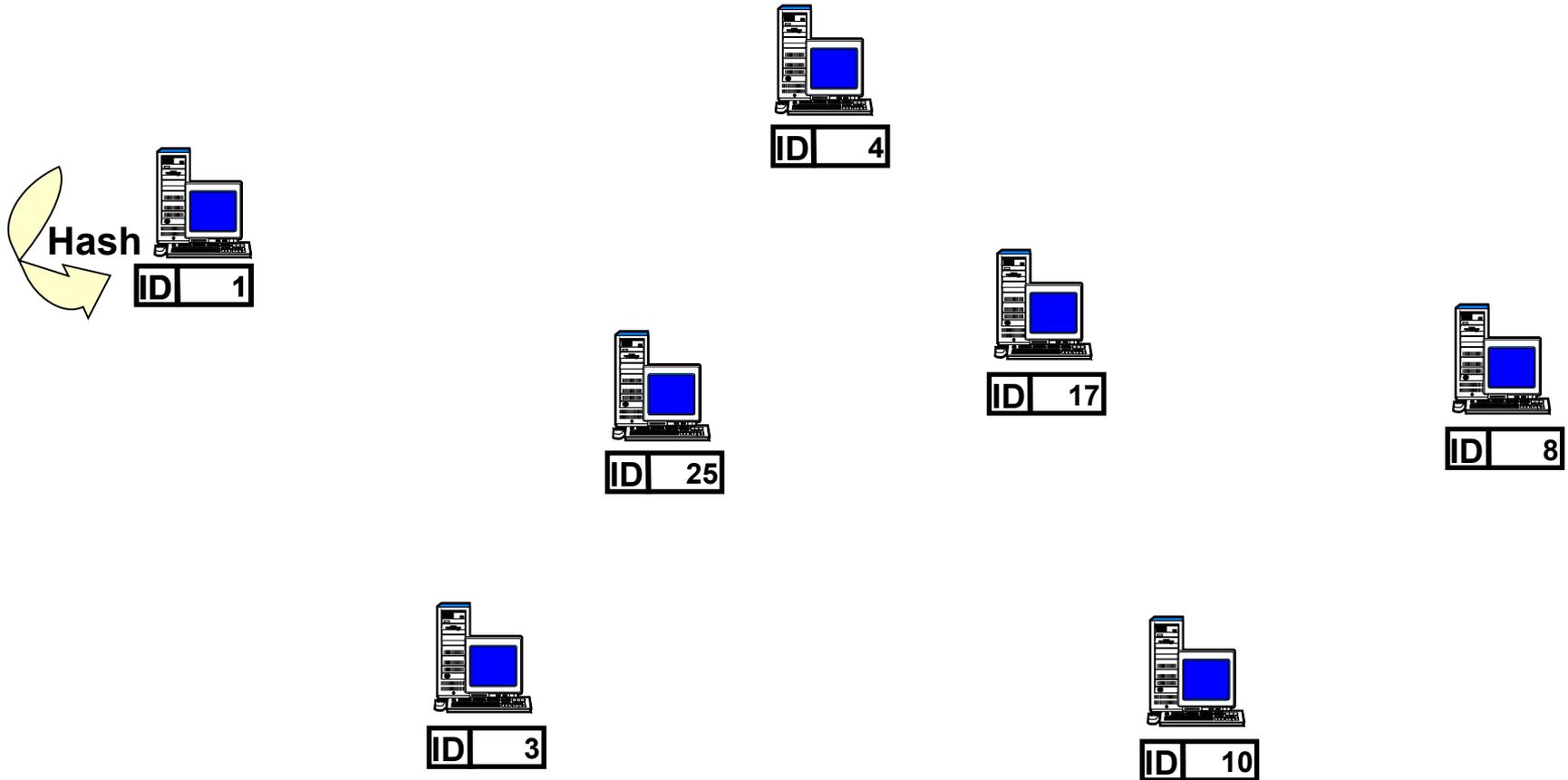


P2P STORAGE NETWORKS

- Una **P2P Storage Network** è un cluster di computers connessi in rete che utilizza tutta la memoria resa disponibile dai peer per definire un servizio distribuito di memorizzazione
 - Esempi: PAST, Freenet, OceanStore, XtremOS Directory Service
- **Organizzazione:**
 - Associazione peer-identificatori mediante **funzioni hash**
 - Ogni peer offre una parte del proprio spazio di memorizzazione, o paga una quota di denaro
 - A seconda del suo contributo, ad ogni peer viene assegnato un volume massimo di dati che possono essere aggiunti alla storage network
 - Assegnazione documenti-identificatori mediante una **funzione hash** calcolata sul nome o su parte del contenuto del file
 - La memorizzazione e la ricerca di files nella rete è guidata dagli identificatori associati dalla funzione hash a peers e files.

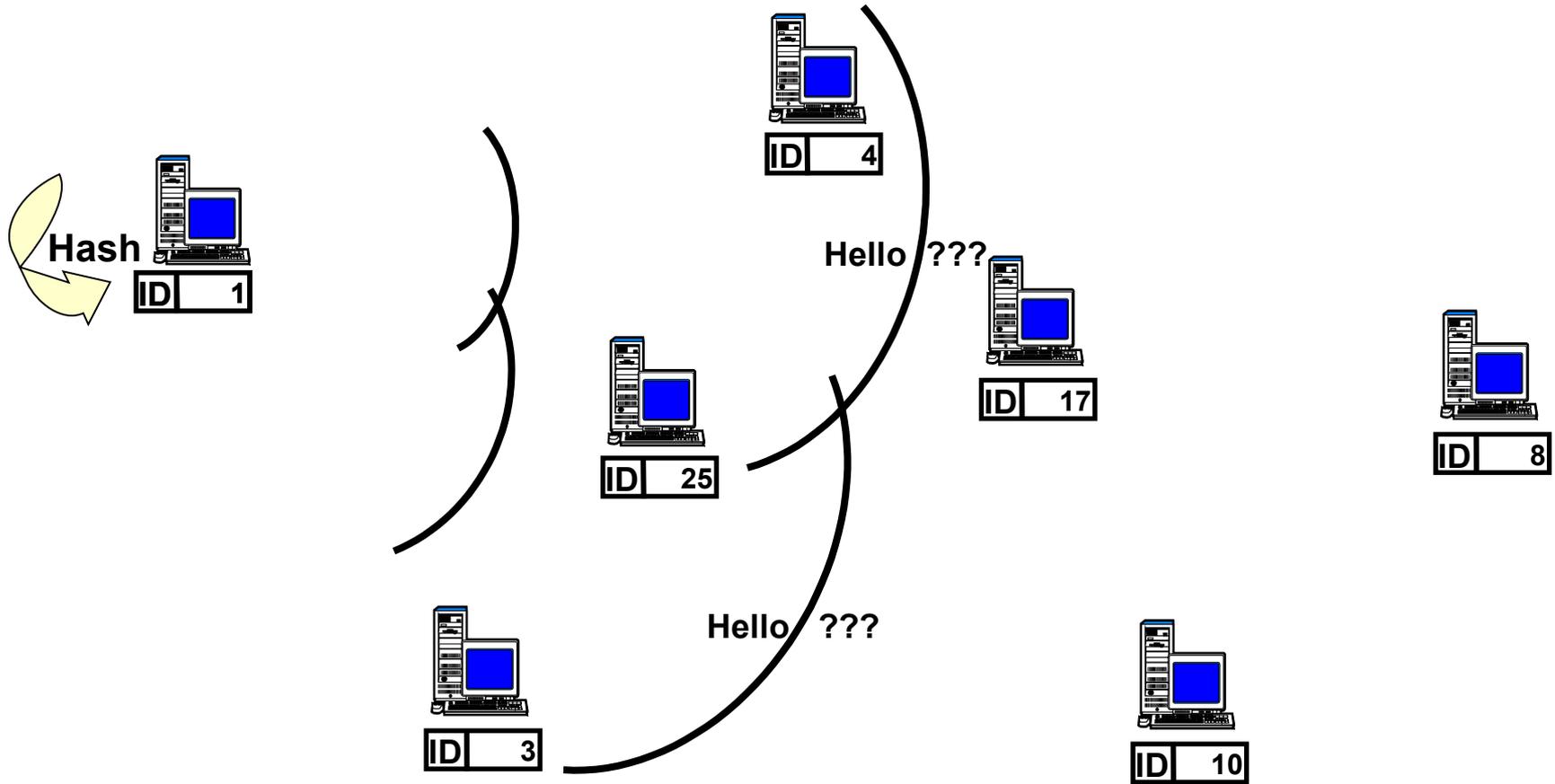
P2P STORAGE NETWORKS

Costruzione



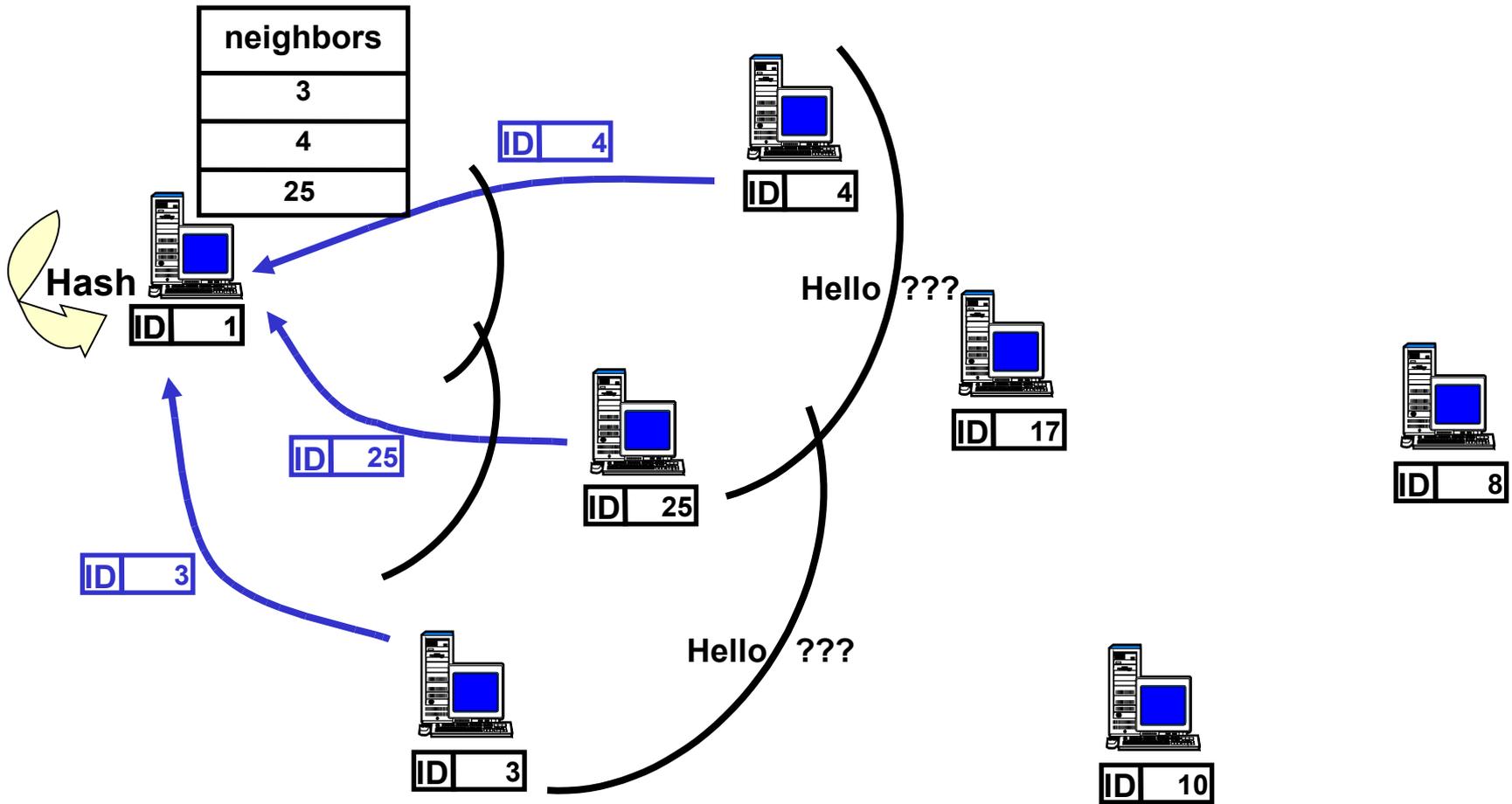
P2P STORAGE NETWORKS

Costruzione



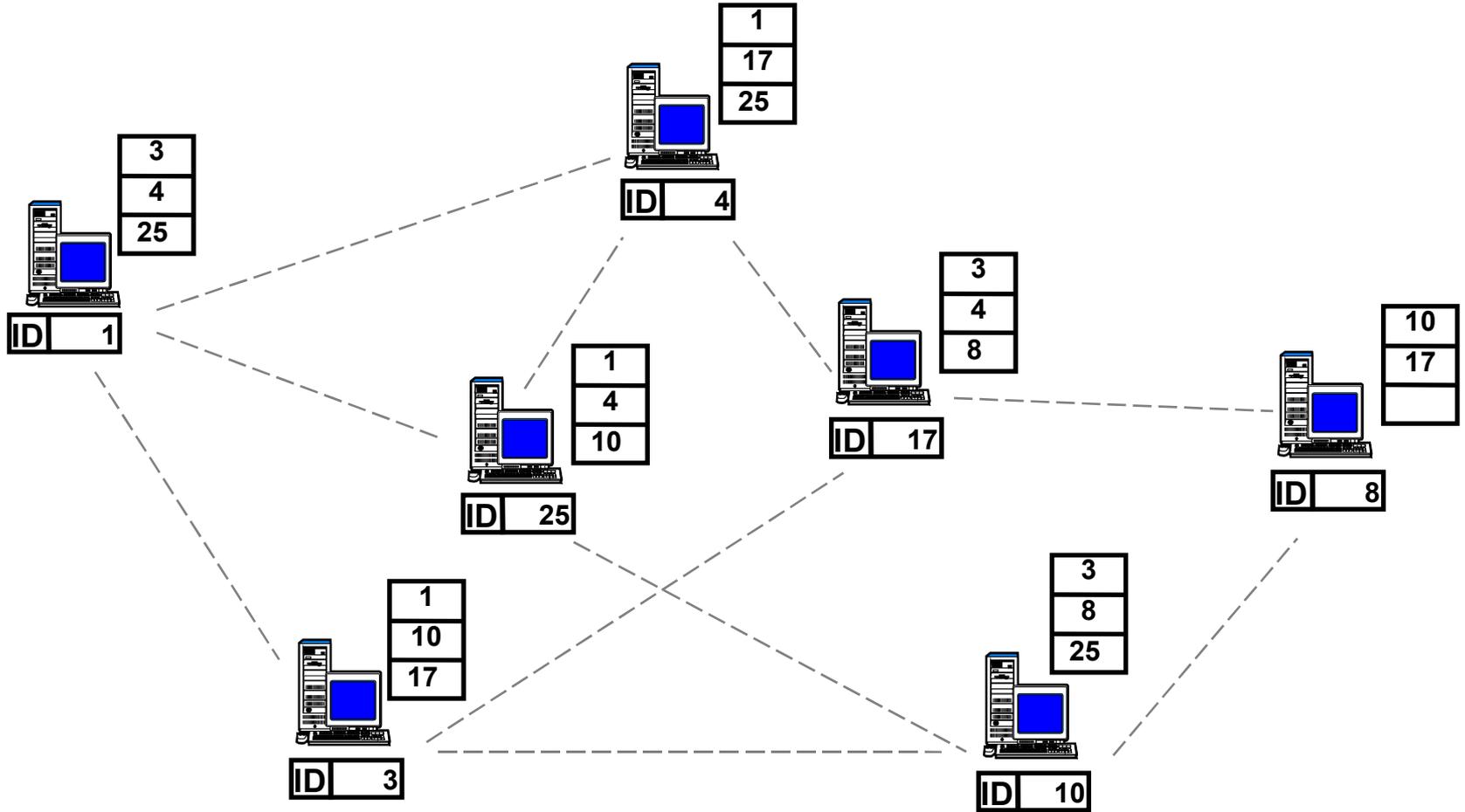
P2P STORAGE NETWORKS

- Costruzione



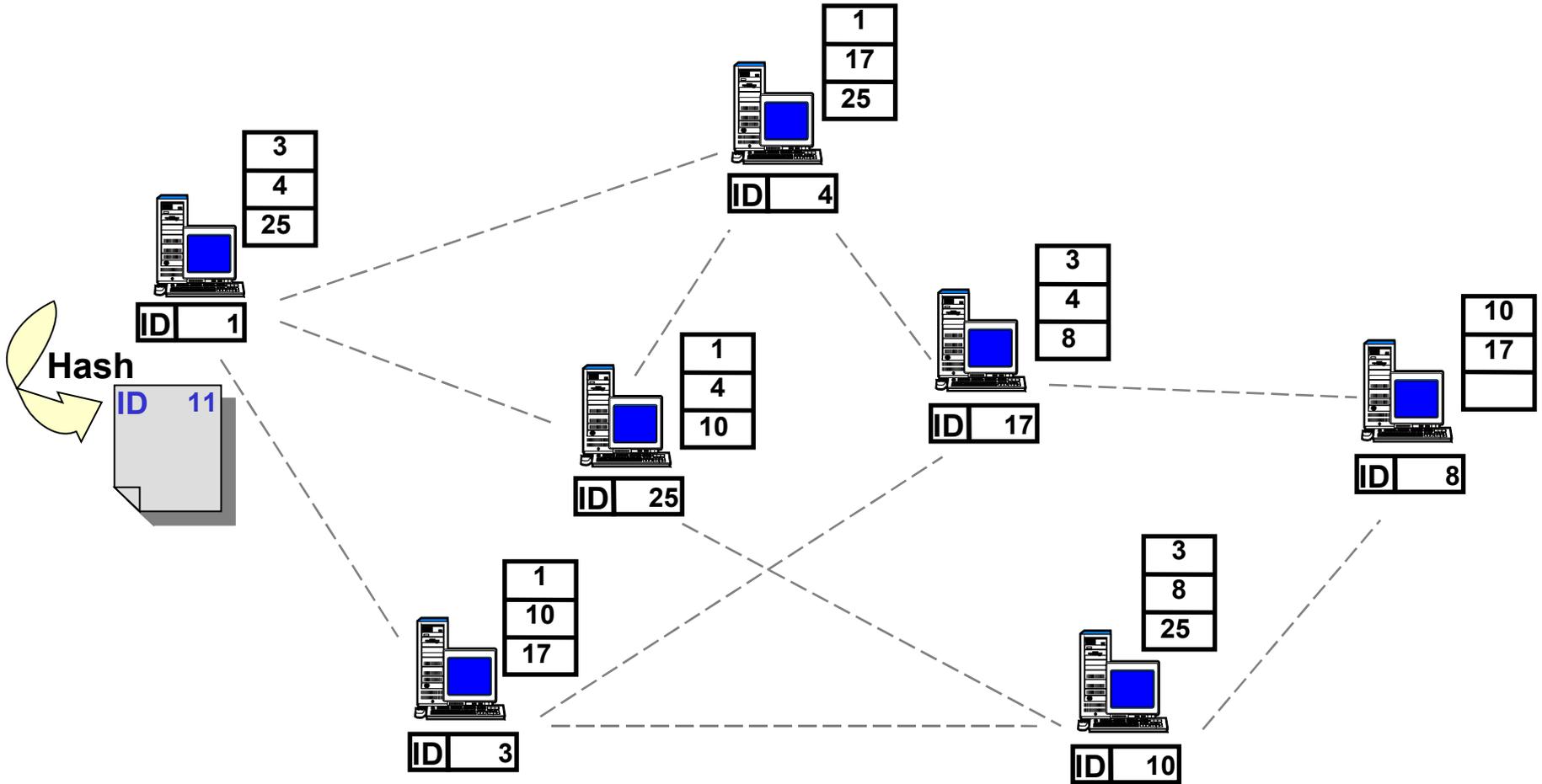
P2P STORAGE NETWORKS

Memorizzazione di Documenti



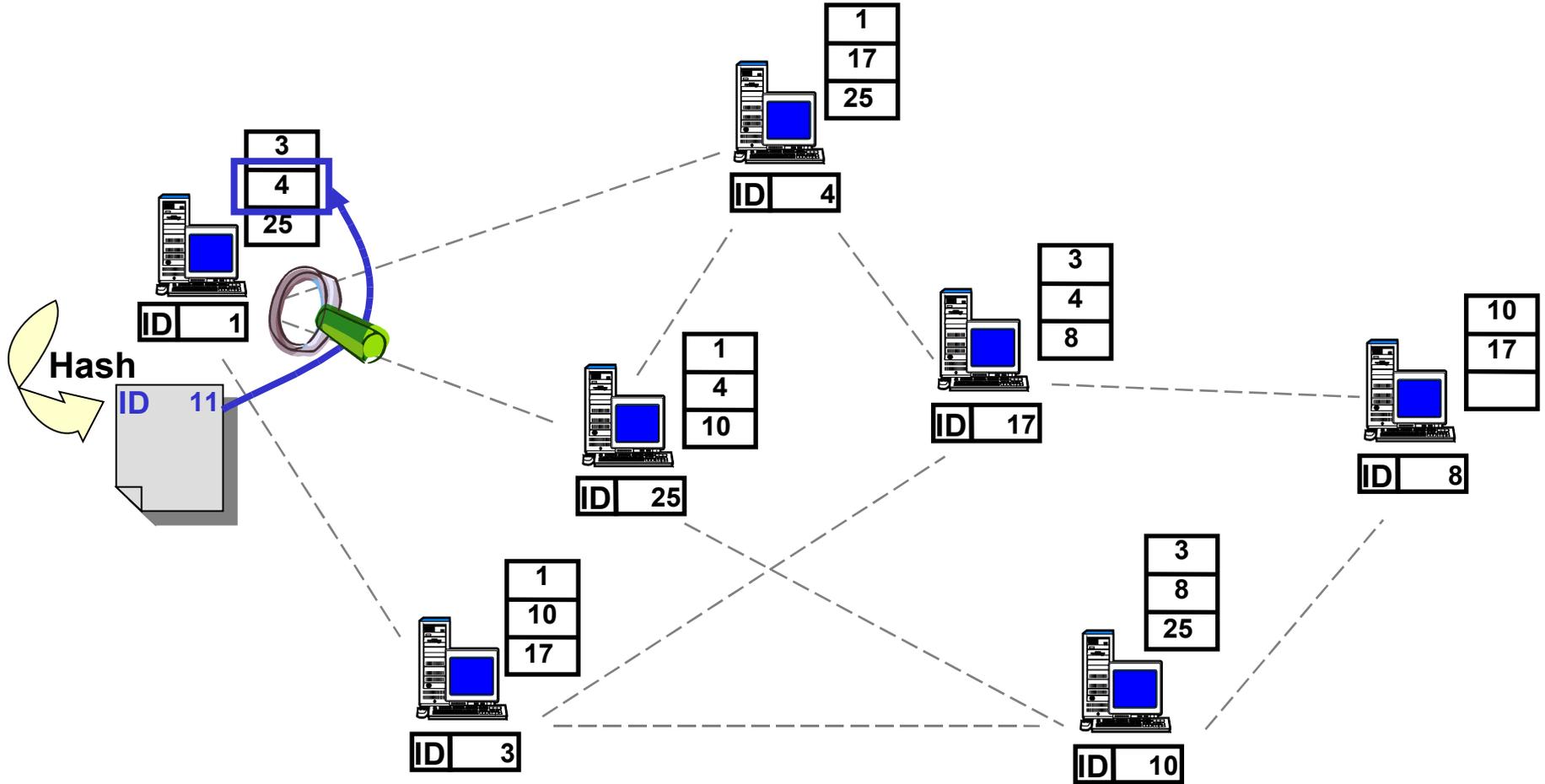
P2P STORAGE NETWORKS

Memorizzazione di Documenti



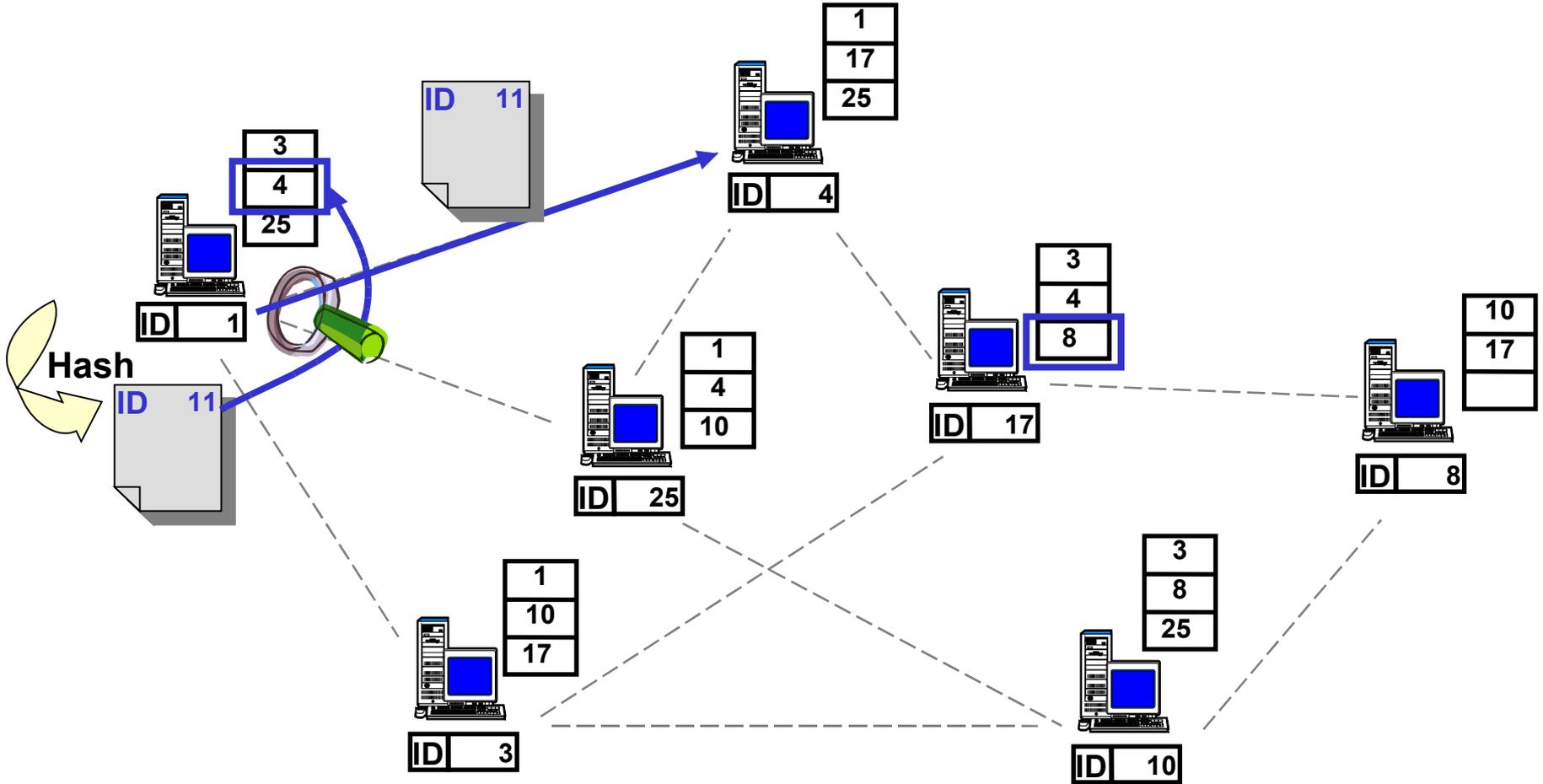
P2P STORAGE NETWORKS

- Memorizzazione di Documenti



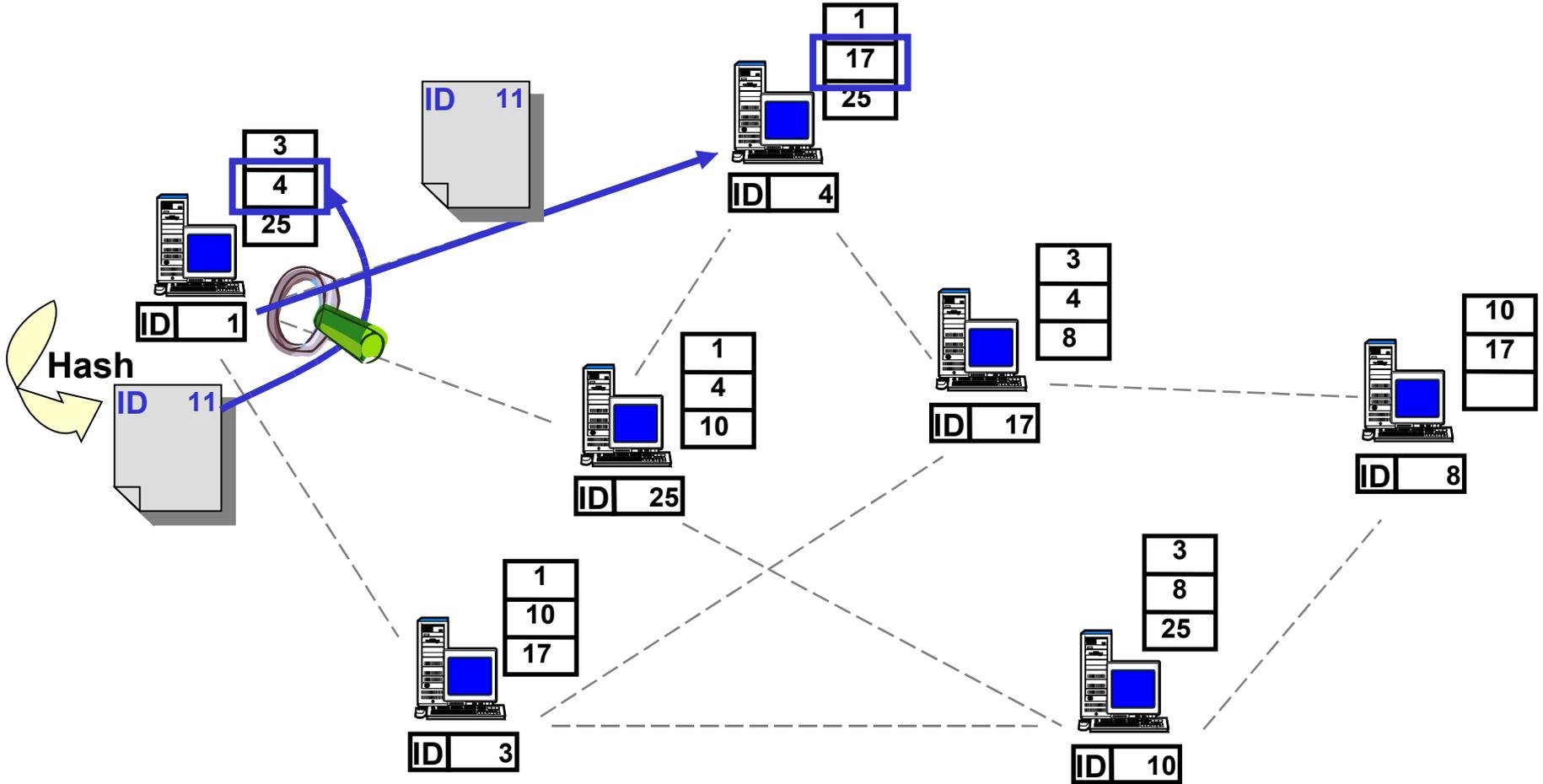
P2P STORAGE NETWORKS

Memorizzazione di Documenti



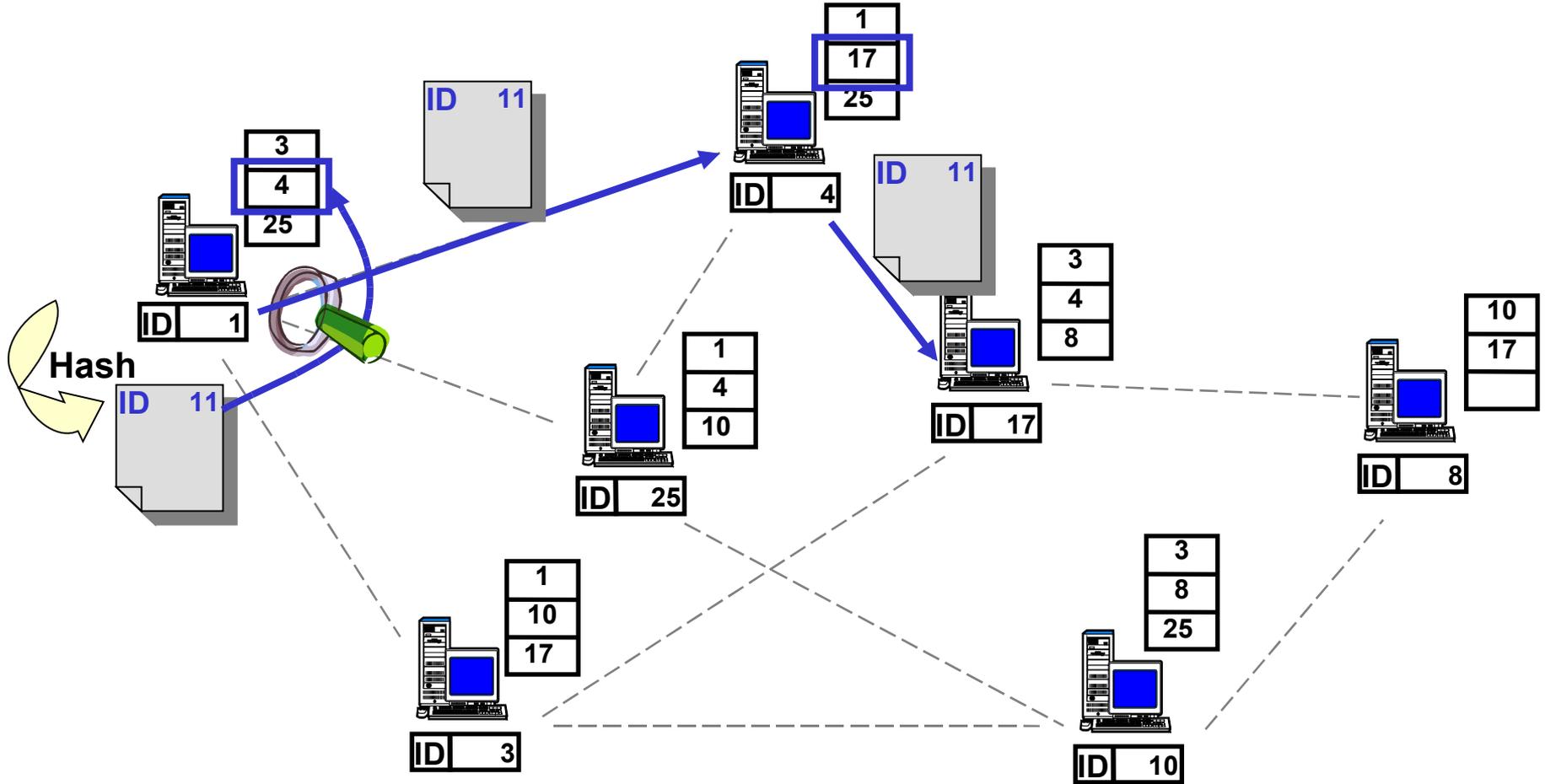
P2P STORAGE NETWORKS

Memorizzazione di Documenti



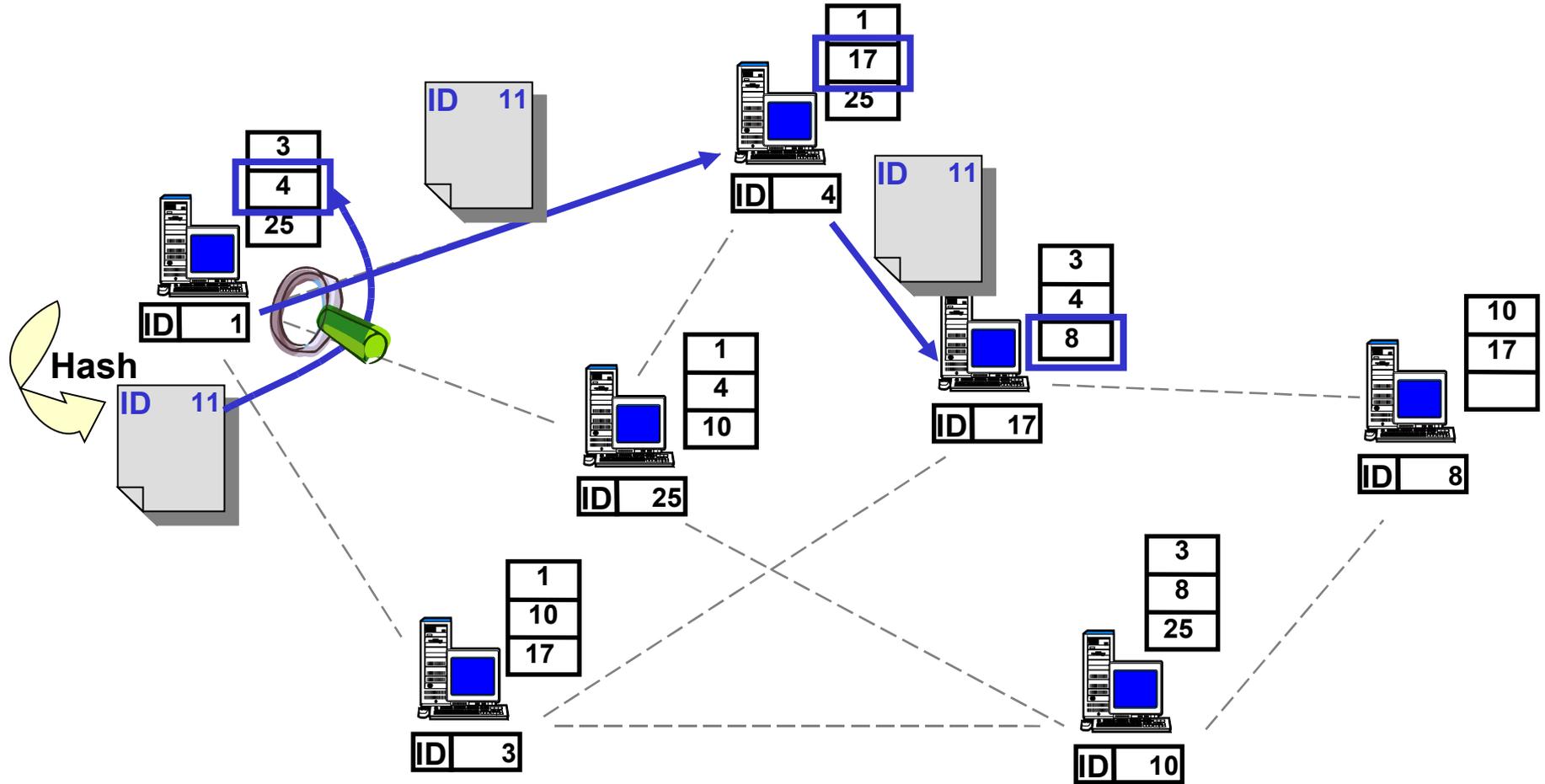
P2P STORAGE NETWORKS

- Memorizzazione di Documenti



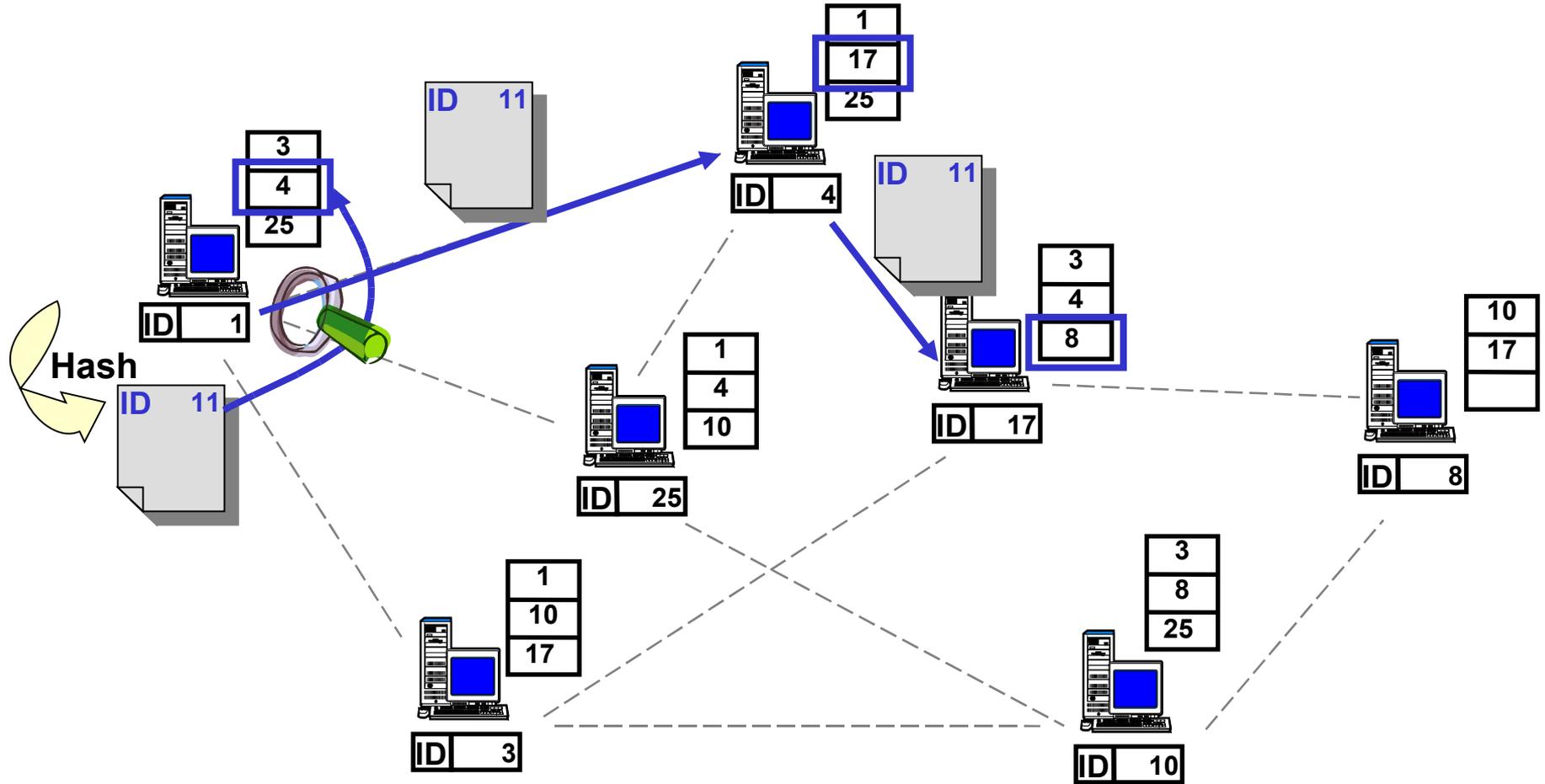
P2P STORAGE NETWORKS

Memorizzazione di Documenti



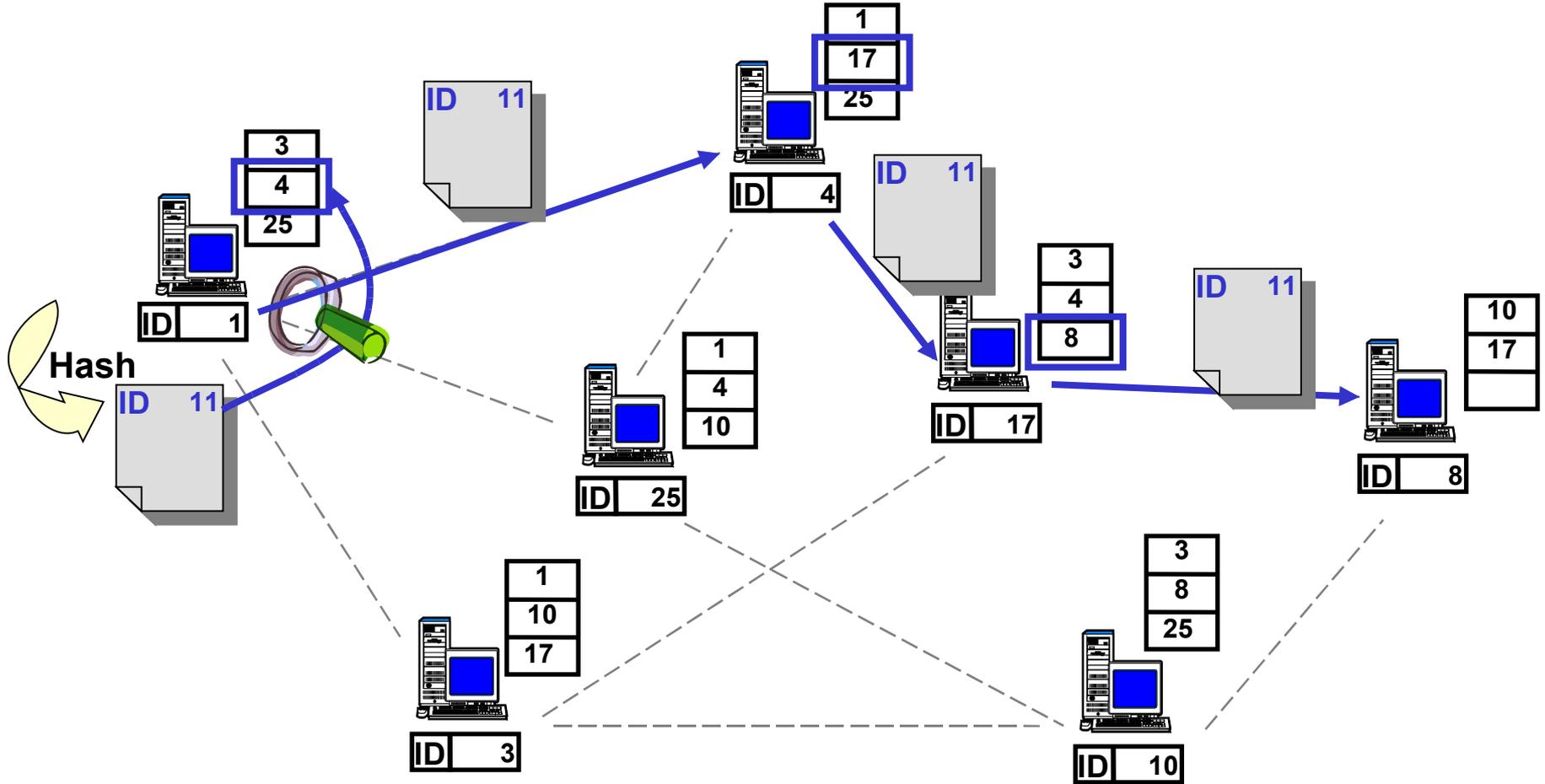
P2P STORAGE NETWORKS

Memorizzazione di Documenti



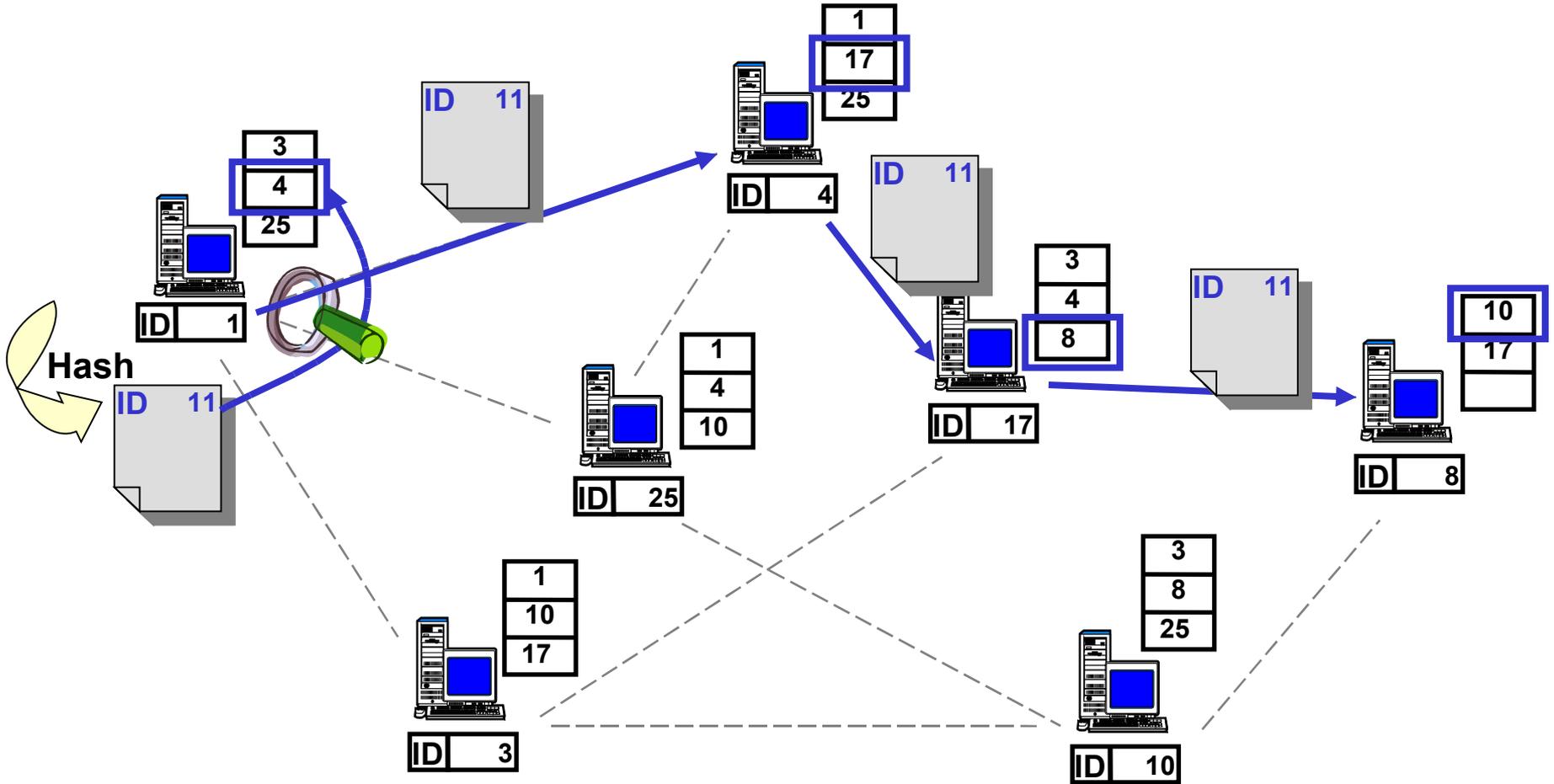
P2P STORAGE NETWORKS

- Memorizzazione di Documenti



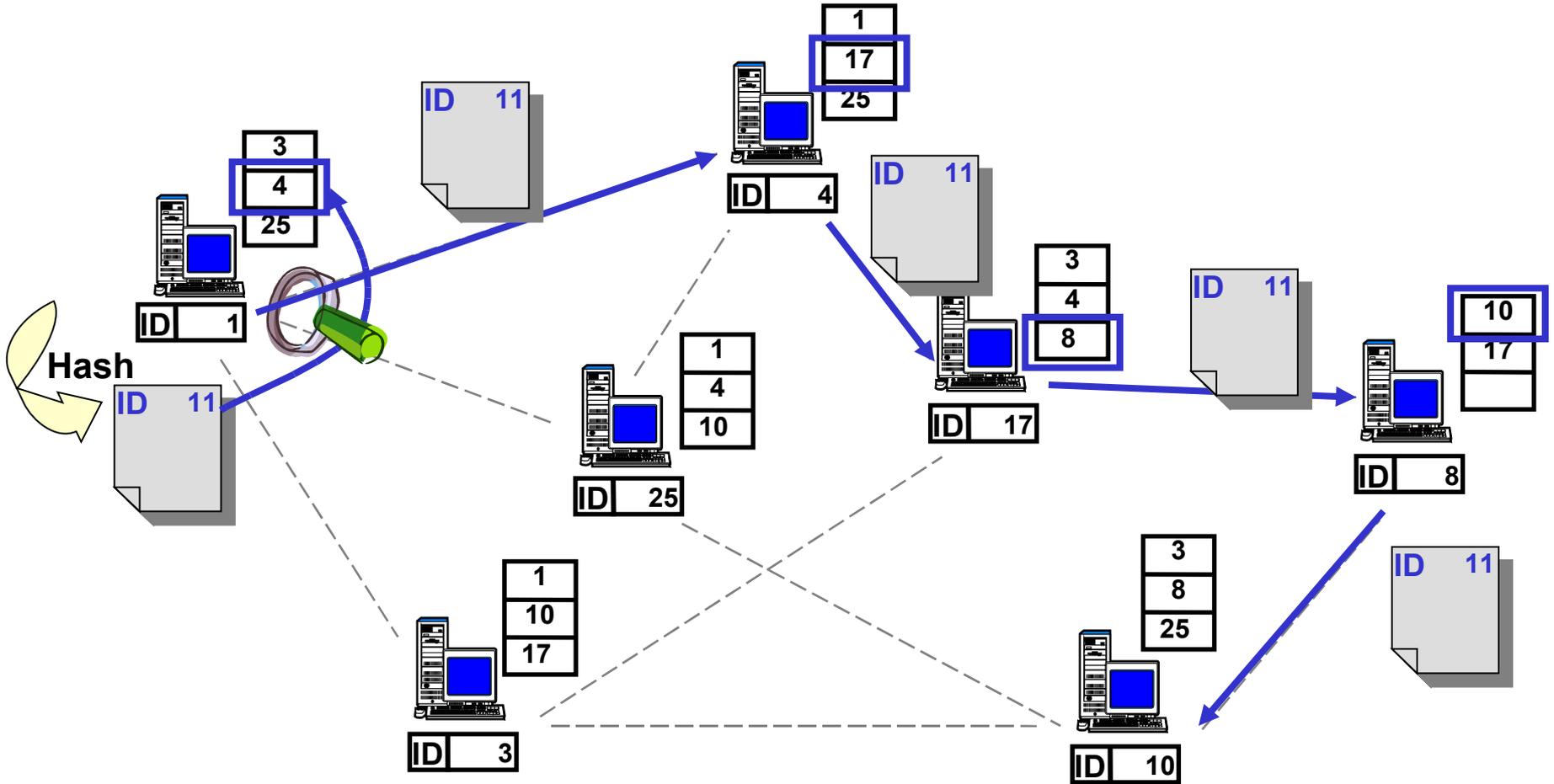
P2P STORAGE NETWORKS

Memorizzazione di Documenti



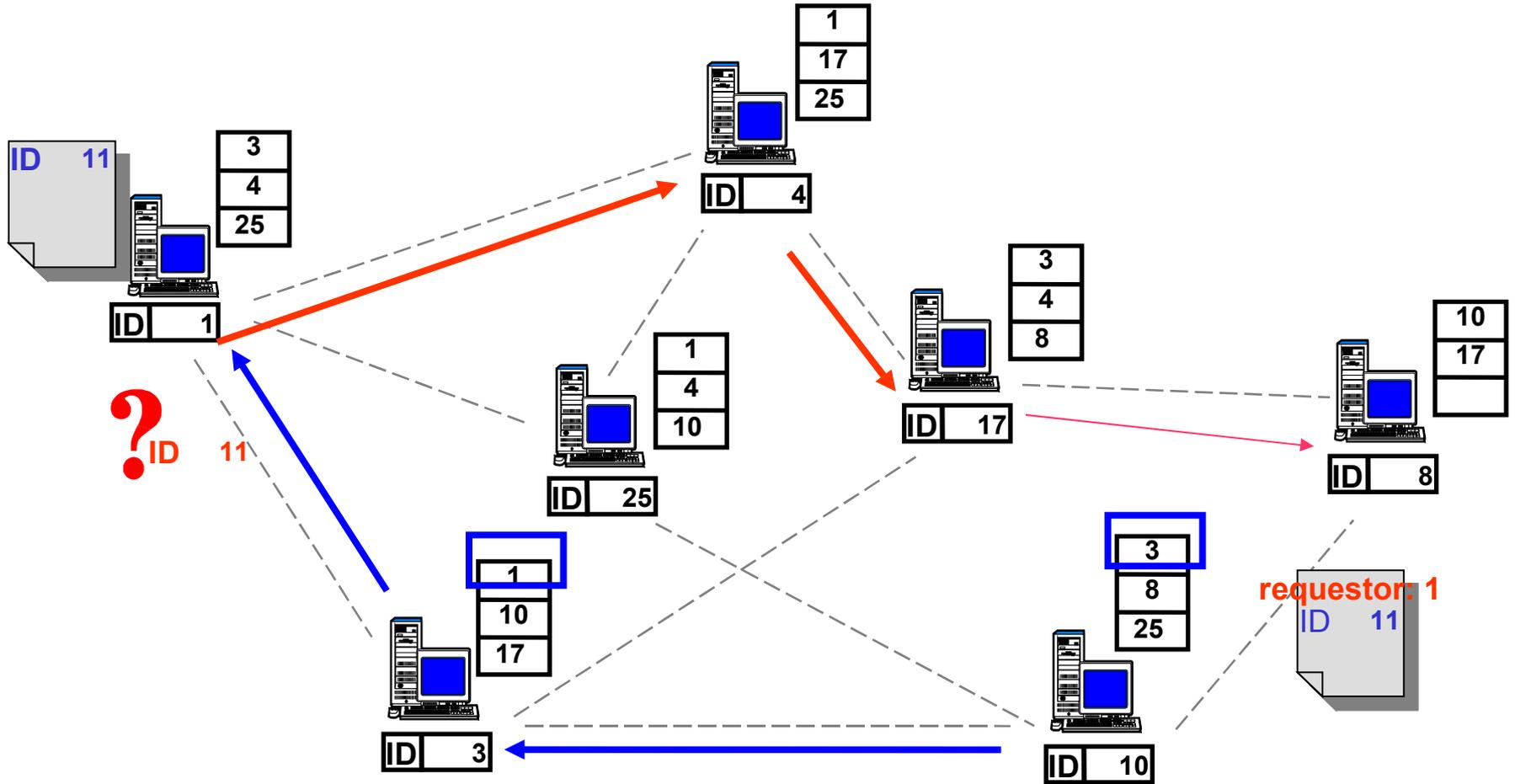
P2P STORAGE NETWORKS

Memorizzazione di Documenti



P2P STORAGE NETWORKS

Ricerca di Documenti



P2P STORAGE NETWORKS

- Realizzate come DHT (Distributed Hash Tables)
- Utilizzate per:
 - Realizzazione di directory distribuite (XtreemOS)
 - Memorizzazione distribuita dell'indice dei file condivisi
 - Rete KAD di Emule
 - File condivisi memorizzati in una directory condivisa
 - L'indice dei file (associazione nome file-peer che condividono i files) viene memorizzato in una directory distribuita (storage network) memorizzata in una DHT (Kademlia)
- File systems distribuiti

OVERLAY P2P

- Un **protocollo P2P** definisce l'insieme dei messaggi che i peer si scambiano, il loro formato, la loro semantica
- Caratteristica comune a tutti i protocolli P2P:
 - identificazione dei peer mediante un **identificatore unico**, generalmente creato mediante l'applicazione di una **funzione hash**
 - i protocolli sono definiti a **livello applicazione dello stack TCP/IP** e definiscono una strategia di routing
- **Overlay Network: rete logica** definita tra i peer a **livello applicazione**
 - **La rete definisce dei links logici tra i peer, che non corrispondono a collegamenti fisici**
 - ad ogni link logico possono corrispondere
 - più hops fisici
 - attraversamento di un insieme di routers

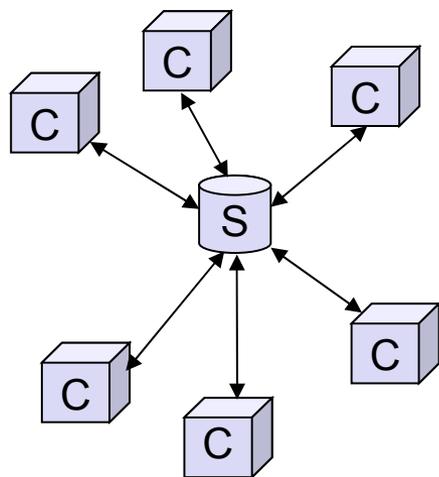
OVERLEY P2P: CLASSIFICAZIONE

Sistemi P2P **non strutturati** (Gnutella, Kazaa,...)

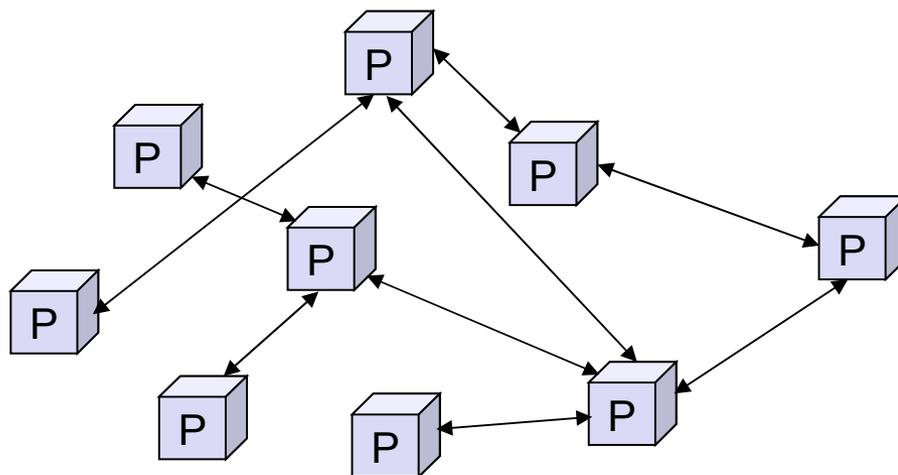
- Un nuovo peer si connette in modo casuale ad un certo numero di peer già attivi all'interno del sistema
- La rete costruita dinamicamente dai peers (overlay network) risulta **non strutturata**
- Algoritmi di Ricerca di informazioni sulla rete: basati su
 - **directory centralizzato** (Napster).
 - **flooding** (Gnutella),Costo della ricerca= lineare in N , dove N è il numero di nodi della rete
- Problema: scalabilità.

OVERLAY NON STRUTTURATI

- I peer interagiscono direttamente tra di loro senza l'intervento di un server centralizzato
- Paradigma di interazione basato su una **cooperazione decentralizzata**, piuttosto che su **coordinamento centralizzato**



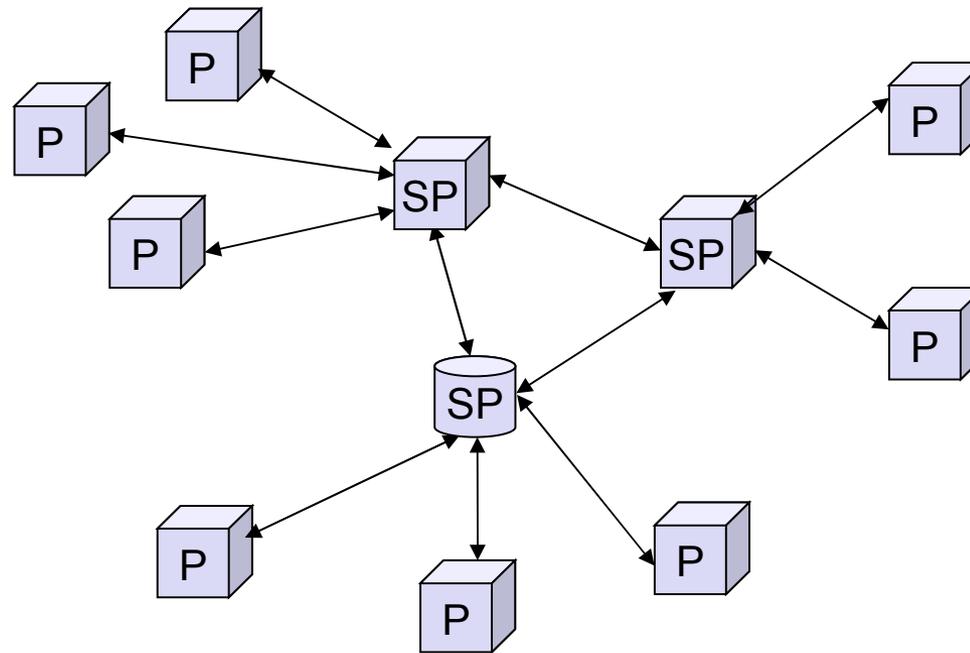
Client/Server



Peer to Peer

OVERLAY NON STRUTTURATI IBRIDI

- Per migliorare la performance del sistema, sono possibili **soluzioni ibride**
 - Definizione dinamica di **Superpeers** che indicizzano le risorse dei peer
 - Le risorse possono essere scambiate direttamente tra i peer



Peer to Peer Ibrido

OVERLAY P2P: CLASSIFICAZIONE

Sistemi P2P strutturati

- La scelta dei vicini a cui un nuovo peer si deve collegare è effettuata in base ad un determinato criterio
- La rete dei peer (overlay network) risulta **strutturata**
- obiettivo: garantire la scalabilità
 - La struttura della rete garantisce che la ricerca di una informazione abbia complessità limitata (ad esempio $O(\log N)$).
 - Complessità limitata anche nel caso di aggiunta di un nuovo peer, o eliminazione di un peer
- Utilizzo di distributed hash tables

SISTEMI P2P: GENERAZIONI

- Se si considerano gli overlay P2P per il file sharing, si può già parlare di 'generazioni' di sistemi P2P, per gli altri tipi di applicazioni è ancora prematuro
- Le diverse generazioni si riferiscono alle modalità con cui viene **gestito l'overlay** e si con cui si indicizzano i dati
- **Prima Generazione: Sistemi con punti di Centralizzazione**
 - Un server centralizzato che coordina e distribuisce dei task ai peer
 - Seti @home, Napster, Bittorrent,
- **Seconda generazione, Overlay non strutturati**
 - Peer to peer puro, nessun server
 - Gnutella, Kazaa, Skype (a parte la fase di autenticazione)
- **Terza generazione, Overlay strutturati**
 - Basati su DHT
 - CAN, Chord, Pastry, Emule,...

PRIMA GENERAZIONE: SETI@home

- SETI@Home: Search for Extraterrestrial Intelligence
- Progetto di ricerca universitario (UC California, Berkeley) supportato anche da alcune industrie. Inizio: fine anni 90
- Analisi delle emissioni radio ricevute dallo spazio, raccolte dal telescopio Arecibo, situato a PortoRico
- Scopo: costruire un 'supercomputer' aggregando la potenza computazionale offerta da computers connessi in Internet, sfruttando i cicli di inattività
- Costo ridotto di un fattore 100 rispetto all'acquisto di un super computer
 - 200 milioni di dollari per l'acquisto di un supercomputer 30 TeraFlops
 - Il progetto è costato meno di 1 milione di dollari

PRIMA GENERAZIONE: SETI@home

- Utilizza un Data Server centralizzato ed un insieme di clients
- Client = *Screen Saver Program* sviluppato per diverse piattaforme
- Un unico data server distribuisce i dati
 - Problema: scalabilità ed affidabilità ridotte
 - In certi periodi i client riescono a connettersi al server solo durante la notte
 - Definizione di proxy servers: il proxy si connette al server centrale durante la notte e distribuisce i dati durante il giorno
- Alto rapporto computazione/comunicazione
 - Ogni client si connette per un breve intervallo di tempo per ricevere i dati, effettua la computazione (richiesta computazionale alta), invia i risultati
 - Non c'è comunicazione tra i peers

PRIMA GENERAZIONE: SETI@home

- Spesso non viene considerata una vera architettura P2P
- I peer
 - Chiedono tasks da eseguire ad un server centralizzato
 - Non possono sottomettere task al server
- 'Aggregazione' di un alto numero di risorse di calcolo'
 - Ricerca di numeri primi
 - Protein folding
 - Ricerca di vita extra-terrestre

PRIMA GENERAZIONE: BITTORRENT

- Concepito per il download di grosse moli di dati
- Allo stato attuale è il protocollo che genera il maggior volume di traffico Internet (secondo alcune statistiche >60%)
- Focus sul **download parallelo di contenuti**, piuttosto che su strategie sofisticate di ricerca del contenuto
- I file vengono divisi in chunks per il download
- Do ut des: definizione di strategie per l'**incentivazione della collaborazione**
- Diverse strategie per la scelta dei pezzi di file da scaricare
- L'insieme dei peer che partecipano al download di un file (swarm) è coordinato da un server (tracker)
 - Tracker = punto di centralizzazione
 - Le ultime versioni (trackerless) utilizzano una DHT

SECONDA GENERAZIONE: GNUTELLA

- P2P puro (o flat P2P): tutti i nodi hanno lo stesso ruolo e sono interconnessi mediante un overlay non strutturato
- Goal: ricerca completamente decentralizzata di files
- Ogni nodo
 - Condivide alcuni files
 - Riceve le query sottomesse dall'utente e le forwarda ai vicini
 - Forwarda le query provenienti dai vicini
 - Risponde alle query, analizzando se i file che ha messo in condivisione soddisfano la query
 - Esplora periodicamente la rete per conoscere eventuali nuovi nodi
- Meccanismo di propagazione delle query = **flooding**
- Problema principale: bassa scalabilità dovuta all'alto numero di messaggi scambiati sull'overlay

SECONDA GENERAZIONE: KAZAA

- insieme a Napster, Bittorrent, Emule...uno dei sistemi per il file sharing più popolari...
- caratteristica innovative: struttura gerarchica con superpeers
- i nodi hanno caratteristiche diverse
 - memoria
 - CPU
 - connettività
 - uptime
 - raggiungibilità
- si sfrutta l'eterogeneità dei peer per assegnare loro dei ruoli diversi
 - nodi stabili e con buona risorse (calcolo, connettività) diventano superpeer
 - Peer semplici: peer 'instabili', con poca potenza di calcolo, nati

TERZA GENERAZIONE: OVERLAY STRUTTURATI

Distributed Hash Tables:

- Assegnamento di identificatori mediante funzioni hash ai peer ed ai dati
- Lo stesso spazio di identificatori per peer e dati
- Definizione di una funzione di mapping dati-peer
- **Key based search:**
 - Definizione di algoritmi di routing in cui ogni hop sull'overlay è guidato dal contenuto della chiave che si sta cercando
- Diverse proposte sviluppate in ambito accademico
 - CHORD, CAN, PASTRY,...
- Alcuni sistemi di file sharing 'commerciali' utilizzano la DHT Kademia sviluppata in ambito accademico
 - Rete KAD di Emule
 - Trackerless Bittorrent

P2P: VANTAGGI

- Vantaggi per l'utente: offrire risorse computazionali 'in eccesso' (cicli idle di CPU, spazio di memoria inutilizzato, banda inutilizzata,...) in cambio di risorse/servizi/partecipazione a social networks,...
- Vantaggi per la comunità
 - **self scaling property**: la partecipazione di un numero sempre maggiore di utenti/host incrementa in modo naturale le risorse del sistema e la sua capacità di servire un maggior numero di richieste
- Vantaggio per il venditore: diminuzione di costo per il lancio di una nuova applicazione
 - client/server: richiede l'uso di una server farm con alta connettività sulla rete in modo da soddisfare le richieste di milioni di utenti
 - fault tolerance: la server farm deve essere replicata in diverse locazioni
 - i server devono essere gestiti in modo da offrire un servizio 24*7

P2P: IL FUTURO

- Il successo di una applicazione P2P dipende in gran parte dalla definizione di una 'massa critica' di utenti
- P2P application critical mass: Livello di partecipazione ad una applicazione P2P che consente alla applicazione stessa di 'auto-sostenersi'
- Nei primi sistemi P2P hanno giocato un ruola fondamentale per il raggiungimento della massa critica la novità dell'applicazione
 - File sharing: free content, ampia selezione di contenuti
 - VoP2P: basso costo di chiamate internazionali, buona qualità di audio,...
 - P2PTV: accesso ad una vasta gamma di programmi, possibilità di accesso da ogni luogo (viaggi,....)
- Il successo di nuove applicazioni P2P dipenderà oltre che da una buona ingegnerizzazione delle applicazioni,
 - Dall'appealing della nuova applicazione
 - dalla definizione di ragionevoli 'business model'