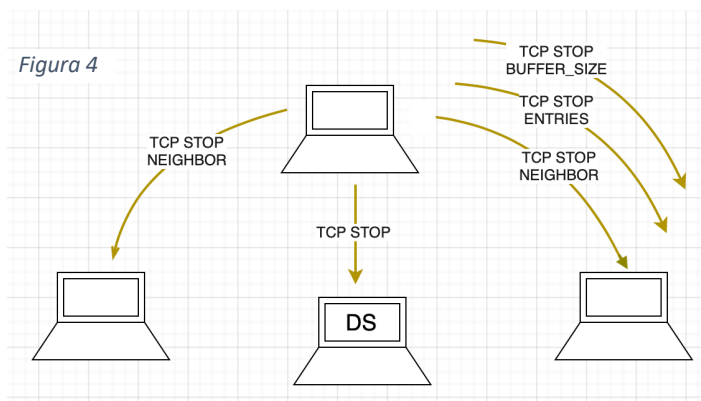


La rete è strutturata ad anello come si può vedere da Fig. 1 dove ogni peer è connesso con socket di tipo TCP sia al DS sia ai due vicini.

La fase di connessione con il peer (rappresentata in Fig. 2) avviene come segue: prima il peer prova molteplici volte a mandare un messaggio di HELLO tramite socket UDP, successivamente il DS risponderà con un messaggio contenente i suoi vicini, infine il peer instaurerà una connessione TCP con il DS.

Successivamente il peer che ha ricevuto i vicini dal DS stabilirà una connessione TCP con il vicino di destra (come in Fig 3). Questo procedimento verrà effettuato da ogni peer e quindi si verrà a formare un anello.



La fase di disconnessione (rappresentata in Fig. 4)(interamente in TCP) di un peer dalla rete avverrà mandando prima le proprie entries (se presenti, previo invio della dimensione del messaggio) e successivamente dei suoi vicini in modo che i vicini riescano a richiudere l'anello; infine verrà comunicato al server la volontà di abbandonare la rete (messaggio di STOP)

Il formato dei messaggi scambiati con il server: tipo del messaggio, indirizzo mittente, 2 campi (usati in fase di connessione di un peer) contenenti eventuali numeri di porta, campo contenente il numero di peer connessi alla rete (usato quando viene richiesto dai peer per la ricerca di un dato aggregato).

I messaggi scambiati tra peer sono formati dai seguenti campi: tipo del messaggio, tipo del servizio (totale o variazione), tipo entry (N/T), data inizio periodo, data fine periodo, direzione seguita dal messaggio (l/r), porta richiedente del servizio, dimensione del buffer, ttl, buffer per i dati.

Quando un Peer esegue una richiesta di dati aggregati (come si vede da Fig. 5) prima contatterà i suoi vicini con un messaggio di richiesta REQ_DATA in UDP, i suoi vicini

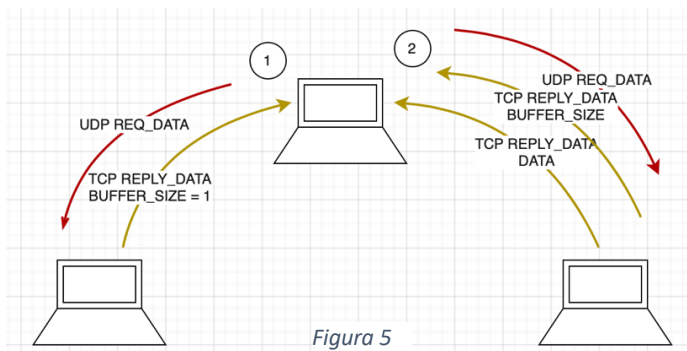


Figura 5

risponderanno con un messaggio (contenente nel buffer): se possiedono già il dato aggregato invieranno prima la dimensione totale del messaggio che inoltreranno subito dopo, se al contrario non posseggono il dato aggregato inoltreranno una dimensione del buffer non valida (1).

In quest'ultimo caso si procederà come segue in Fig. 6:

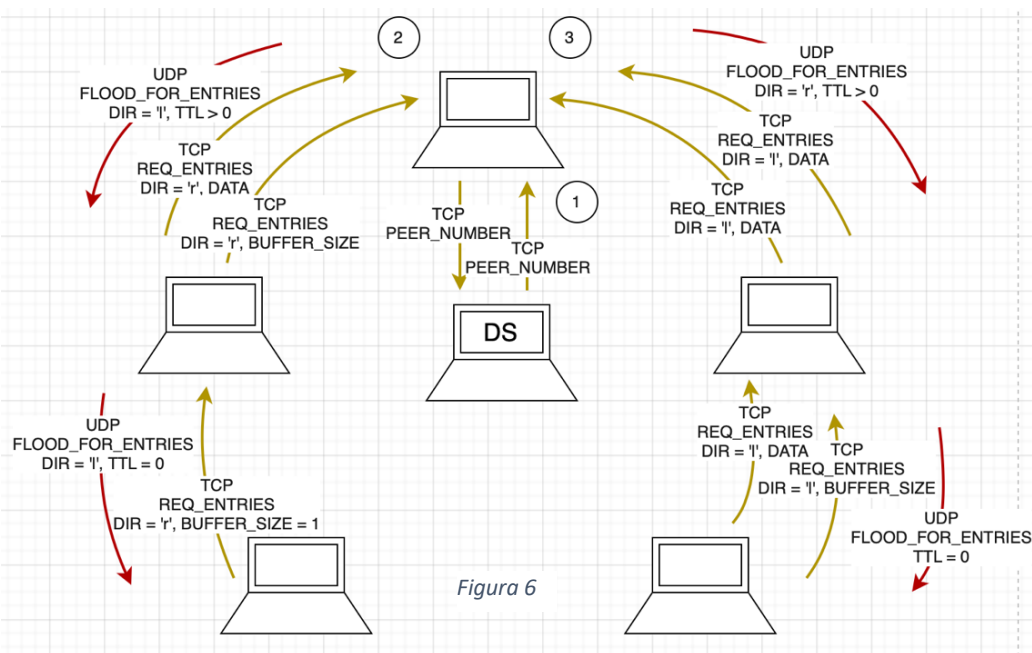


Figura 6

Il Peer richiederà al DS il numero di Peer presenti sulla rete, successivamente manderà un messaggio in UDP di FLOOD FOR ENTRIES sia a sinistra che a destra specificando la direzione (l/r), e un campo TTL dove verrà inserito un valore che sarà la metà o la metà + 1 (a seconda che il numero sia pari o dispari) dei Peer connessi alla rete. A

questo punto ogni peer che lo riceve decrementa il campo TTL e se diverso da zero lo inoltrerà al vicino successivo seguendo la direzione. Così facendo ciascuno dei due messaggi percorrerà metà anello. Quando il campo TTL arriverà ad essere zero verrà generato il messaggio REQ_ENTRIES (in TCP) dall'ultimo peer che invertirà la direzione e se possiede entries inoltrerà la dimensione del messaggio e successivamente le entries, altrimenti il valore del buffer non valido (dimensione = 1).

Le entries salvate nei register hanno lunghezza fissa e presentano i seguenti campi: timestamp inserimento, porta (su 5 cifre), lettera (T -> tampone/N -> nuovo caso), simbolo (+/T -> totale), valore (su 5 cifre).

Note sulla gestione di casi particolari: se sono presenti solo 2 peer sulla rete verrà inserito nel campo dir il valore 'u', in modo che non riceva messaggi duplicati (dato che nel caso di 2 peer vicino destro e sinistro coincidono). Se un peer abbandona la rete e poi immediatamente dopo (o comunque prima delle 18) si riconnette sulla stessa porta al suo primo inserimento di una entry nel register, verranno cancellati i dati aggregati calcolati al precedente accesso (contrassegnati dal simbolo T). In questo modo quando qualcuno richiederà le entries per il calcolo di un dato aggregato verrà ricalcolato il totale e mandato quindi un totale (relativo a quel giorno - porta) maggiore; questo ci permetterà di ignorare totali minori per quel giorno - porta mantenendo la consistenza dei dati sulla rete.