



PARTNER: UNIDATA

RESPONSABILE: ROSARIO ALGOZZINO

Technical Report TR4.1.2

LC3 – Laboratorio pubblico-privato di ricerca
sul tema della Comunicazione delle Conoscenze Culturali

PAG 1 DI 11

Technical Report

TR4.1.2

Overlay network e sue applicazioni

DRAFT

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--

Abstract

Le applicazioni peer to peer sono applicazioni distribuite in cui ogni nodo della rete fornisce servizi a tutti gli altri nodi e usa quelli da essi forniti. Il paradigma peer to peer si può quindi contrapporre alla classica organizzazione client-server, in cui pochi nodi servitori hanno il ruolo di fornire i servizi per una grossa quantità di nodi clienti. In questo senso i vantaggi del paradigma peer to peer risiedono nella sua struttura decentralizzata, che garantisce un altissimo livello di scalabilità e di liveness. L'organizzazione della overlay network, cioè della rete formata dai nodi che compongono l'applicazione e dai loro collegamenti con gli altri nodi, è un'importante scelta progettuale di un'applicazione peer to peer. Tale rete deve infatti realizzare i meccanismi tramite i quali ogni nodo è in grado di individuare correttamente gli altri nodi con i quali è interessato a comunicare. Il problema dell'organizzazione risulta particolarmente arduo da risolvere in quanto deve tenere conto della totale dinamicità della rete, dovuta al fatto che generalmente ogni nodo si può inserire o rimuovere in qualunque momento. La prima generazione di applicazioni peer to peer ha quindi affrontato il problema riconducendosi a classiche soluzioni client-server (in cui alcuni server centrali si occupano di fornire i servizi per l'individuazione dei nodi), oppure basandosi su reti non strutturate (in cui l'individuazione dei nodi può essere effettuata tramite broadcast). Ultimamente sono state però proposte delle tecniche per la gestione di overlay network decentralizzate e strutturate, che forniscono garanzie sulla complessità di esecuzione delle operazioni e sull'affidabilità dei risultati. Tali tecniche possono quindi essere usate come base per una nuova generazione di applicazioni peer to peer che superi le limitazioni introdotte in passato.

1. Distribuzione del Servizio

Le applicazioni peer to peer si basano sul concetto di distribuzione del servizio, per cui ogni nodo della rete deve in qualche modo contribuire a realizzare una parte del servizio complessivo. Il paradigma peer to peer è quindi più facilmente applicabile in applicazioni che si adattano in maniera naturale a tale concetto, come ad esempio nei sistemi di file-sharing, dove ogni nodo mette a disposizione i file che esso contiene a tutti gli altri nodi della rete. La distribuzione del servizio porta però con sé il problema della localizzazione dei nodi che forniscono il servizio richiesto. Tale localizzazione può essere effettuata attraverso la gestione di un indice centrale, ma questo ricondurrebbe il sistema peer to peer ad un sistema client-server. Viceversa, in mancanza di tale indice, la localizzazione può essere effettuata contattando tutti i nodi della rete con messaggi in broadcast. Per evitare la congestione il broadcast può essere limitato da un time to live. Una soluzione più

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--



sofisticata consiste nella realizzazione di un servizio distribuito per la localizzazione dei nodi. Tale servizio può presentarsi in vari modi, dei quali ne citiamo alcuni principali:

Distributed hash table (DHT) Fornisce funzionalità equivalenti a quelle di una tabella hash classica, con la differenza che in questo caso il servizio è realizzato in modo distribuito fra tutti i nodi della rete. In particolare il sistema fornisce un dizionario che associa contenuti a chiavi.

Decentralized object location and routing (DOLR) Permette ai nodi di pubblicare generiche risorse (eventualmente replicate), e di individuare le repliche più vicine delle risorse di interesse.

Multicast Realizza canali per lo scambio di informazione. I nodi hanno la possibilità di sottoscrivere ai canali, e di spedire messaggi a tutti i nodi che appartengono ad un dato canale. Questi servizi possono essere realizzati a partire da un'overlay network che fornisca i meccanismi del key based routing. In tale sistema ad ogni nodo è associato un indirizzo univoco preso da un determinato spazio degli indirizzi. L'algoritmo di routing è quindi in grado, a partire da un indirizzo arbitrario, di spedire un messaggio al nodo attivo che ha l'indirizzo più vicino all'indirizzo scelto, secondo un'opportuna funzione di distanza fra indirizzi. Nel seguito dell'articolo si esaminerà più in dettaglio quali siano i componenti di un'applicazione peer to peer basata su servizi di questo genere, e come tali componenti possano essere realizzati.

2. Architettura Applicativa

In letteratura sono stati presentati una varietà di sistemi basati su overlay network strutturate, fra cui Chord[10], Tapestry[11], Kademlia[4] e Pastry[8]. Tali sistemi sono generalmente progettati dalle fondamenta per la realizzazione di uno specifico servizio usando un'architettura monolitica. In questo documento si cercherà invece di proporre un framework generico all'interno del quale possono essere esaminati tutti i sistemi proposti. In Fig. 1 è presentata una possibile architettura di un nodo di un'applicazione peer to peer basata su overlay network strutturate. Ai livelli inferiori sono presenti gli strati OSI fino a quello di trasporto. Sopra al livello di trasporto è presente uno strato denominato "servizi di rete", che si occupa di completare le funzionalità fornite dallo strato di trasporto realizzando canali di comunicazione dalle caratteristiche desiderate ed un sistema per il monitoring dello stato della rete. Al di sopra dei servizi di rete è costruito il componente che si occupa della gestione della overlay network, ed in particolare della tabella e degli algoritmi di routing. Infine il middleware peer to peer è il componente che amalgama i

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--

servizi offerti dalla rete e dalla overlay network, realizzando le funzionalità ad alto livello su cui si basa l'applicazione.

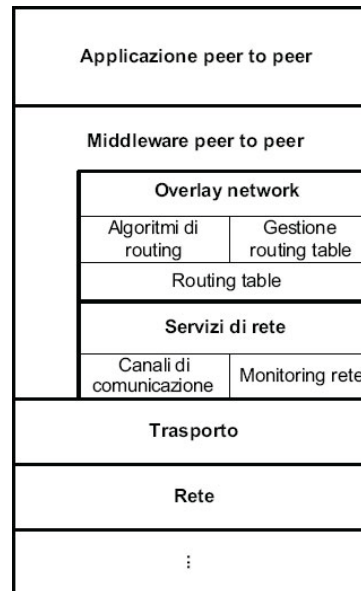


Fig. 1

3. Servizi di Rete

I servizi di rete rappresentano tutte quelle funzionalità richieste dalla overlay network che devono essere fornite dalla rete sottostante. Tali funzionalità possono essere divise in due gruppi:

- a. La realizzazione di canali di comunicazione fra i nodi.
- b. Un sistema per il monitoring dello stato della rete.

La prima funzionalità è richiesta dalla overlay network per effettuare lo scambio dei messaggi da un nodo ad un altro. La seconda funzionalità è richiesta per gestire la tabella di routing in modo tale che contenga riferimenti validi e con le caratteristiche desiderate. Poichè lo strato “servizi di rete” è un completamento dello strato di trasporto, essi possono essere considerati concettualmente come un’unica entità. A seconda del livello di servizi offerto dallo strato di trasporto, lo strato dei servizi di rete sarà più o meno esteso. Al limite lo strato di trasporto potrebbe già fornire tutti i servizi richiesti dalla overlay network. Supponendo di basarsi su una rete per lo scambio non affidabile di pacchetti di lunghezza massima prefissata (come Internet), il sistema dovrà probabilmente realizzare meccanismi per la trasmissione di messaggi di lunghezza arbitraria, e meccanismi per garantire certi

PROGETTO LC3	Revisione n*	2			
--------------	--------------	---	--	--	--



livelli di affidabilità. IP fornisce già le funzionalità per la frammentazione e il riassetto dei pacchetti troppo lunghi, fino ad una dimensione massima di 64 KB. Generalmente è però preferibile evitare di incorrere in questa frammentazione, in quanto impone un certo overhead e diminuisce la probabilità di corretta consegna del pacchetto riassetto. Infatti, poiché la consegna di ogni singolo frammento è inaffidabile, la probabilità di corretta consegna di tutti i frammenti decresce in maniera esponenziale all'aumentare del numero di frammenti. Il sistema può quindi rilevare la maximum transmission unit (cioè la lunghezza massima dei pacchetti spedibili senza frammentazione) attraverso la tecnica della MTU discovery[5]. I messaggi di lunghezza arbitraria gestiti negli strati superiori dovranno quindi essere suddivisi opportunamente in modo tale che ogni singolo frammento non superi la MTU. Ogni frammento potrà poi essere inviato con un sistema di comunicazione affidabile, così che la probabilità di consegna dell'intero messaggio non diminuisca all'aumentare del numero dei frammenti. Il canale di comunicazione affidabile può essere costruito utilizzando meccanismi di conferma della ricezione, time-out di ritrasmissione, e sliding window. Tali meccanismi generalmente richiedono il mantenimento di uno stato sui due end-point di comunicazione, e perciò risultano in una comunicazione connection oriented. Il monitoring della rete consiste invece in un sistema di predizione che, a partire da dati storici di comportamento della comunicazione fra i nodi, sia in grado di stimare quale sarà il comportamento futuro della rete. Il monitoring è generalmente effettuato su due parametri: il round trip time, cioè il tempo di andata e ritorno di un pacchetto verso un determinato destinatario, e il loss rate, cioè la probabilità che un pacchetto sia perso dalla rete. Questi dati, oltre ad essere utili per lo strato superiore ai fini della gestione della tabella di routing, possono essere usati dal sistema di comunicazione come base per meccanismi di controllo della congestione. I probabili candidati come protocolli per lo strato di trasporto sono UDP[6], TCP[7], e SCTP[9]. UDP fornisce un sistema per lo scambio non affidabile di pacchetti, per cui in questo caso la maggior parte delle funzionalità dovrà essere realizzata dallo strato superiore. TCP fornisce un sistema per lo scambio affidabile e in ordine di flussi di byte, ed è quindi molto più vicino alle funzionalità richieste dalla overlay network. SCTP fornisce un sistema per lo scambio affidabile di messaggi di lunghezza arbitraria, con garanzie di ordinamento fra messaggi selezionabili a piacere. Questo protocollo è quindi teoricamente il più adatto ad essere utilizzato per la realizzazione di applicazioni P2P.

4. Overlay Network

La overlay network deve fornire il supporto necessario alla realizzazione dei servizi degli strati superiori. In tal senso è necessario progettare un sistema che realizzi il key based routing. La prima scelta progettuale consiste nella definizione di uno spazio degli indirizzi.

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--



Generalmente si utilizza uno spazio estremamente ampio, come ad esempio i numeri a 160 bit. Spazi di queste dimensioni hanno la caratteristica di poter contenere il message digest di funzioni hash crittografiche (come ad esempio SHA-1 [2]). Le proprietà delle funzioni hash crittografiche permettono di assegnare facilmente indirizzi univoci a qualunque tipo di risorsa. Ad ogni nodo della rete dovrà perciò essere assegnato un indirizzo preso da tale spazio. A seconda del livello di sicurezza desiderato, tale indirizzo potrà essere generato casualmente dal nodo stesso (come ad esempio un universally unique identifier [3]), oppure generato a partire da dati verificabili (come ad esempio l'indirizzo IP), o in alternativa assegnato da un organismo centrale con certificati basati su firma digitale per garantire alti livelli di sicurezza. Sullo spazio degli indirizzi si dovrà inoltre definire una funzione di distanza. Tale funzione può essere piuttosto semplice come un bitwise XOR fra i due indirizzi, oppure avere una definizione più complessa. In generale tale funzione sarà comunque scelta accuratamente in modo tale da organizzare lo spazio degli indirizzi in una certa struttura desiderata. Due esempi di strutture possibili sono l'anello e l'albero. Una struttura ad anello (usata ad esempio in Chord) si può basare su una funzione di distanza che effettua la differenza fra i due indirizzi. In questo caso i nodi possono essere concettualmente disposti in un anello in modo tale che gli indirizzi siano sempre crescenti (a parte ovviamente nel punto di discontinuità corrispondente all'indirizzo massimo). La distanza fra due nodi sarà quindi tanto maggiore quanto più tali nodi sono distanti all'interno dell'anello. Una struttura ad albero può invece basarsi sul bitwise XOR (come in Kademia). In questo caso si può concettualmente pensare ad un trie, cioè un albero in cui gli indirizzi sono presenti sulle foglie, e ad ogni diramazione di livello n-esimo corrisponde una differenza nell'n-esimo bit dell'indirizzo di tutte le foglie dei due sotto alberi corrispondenti. La distanza fra due nodi sarà quindi tanto più grande quanto è più basso il livello della prima radice comune dei due sotto alberi in cui sono presenti tali nodi. Una volta scelta la struttura dello spazio degli indirizzi si deve quindi decidere nei dettagli la topologia della rete. Ogni nodo dovrà infatti mantenere una serie di collegamenti verso altri nodi in modo da risultare in una rete con la proprietà che, a partire da un qualunque nodo, si possa raggiungere qualsiasi altro nodo conoscendone semplicemente l'indirizzo. Tale proprietà può essere garantita se si garantisce che detto D l'indirizzo di destinazione e d la funzione di distanza definita precedentemente, qualunque nodo di indirizzo A è collegato ad almeno un nodo di indirizzo B tale che $d(B,D) < d(A,D)$, oppure $d(A,D)$ è minima. Questa proprietà permette più in generale di individuare per ogni indirizzo il nodo attivo con indirizzo più vicino ad esso, permettendo quindi la forma completa del key based routing. Se supponiamo di avere una struttura ad anello, la proprietà richiesta può essere garantita collegando ogni nodo con il successivo nell'anello. Percorrendo i collegamenti si arriverà quindi sicuramente a trovare il nodo con indirizzo più vicino a

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--

quello desiderato in un tempo $O(N)$, dove N è il numero totale di nodi. Generalmente sarà però richiesto un sistema di routing che presenti una migliore scalabilità. Questo può essere ottenuto aggiungendo nuovi collegamenti fra i nodi. Ad esempio se consideriamo la struttura ad albero, possiamo fare in modo che ogni nodo sia collegato ad almeno un altro nodo presente in ciascun sotto albero che ha la prima radice comune al livello n -esimo, per ogni n . In questo modo a partire da ogni nodo si potrà raggiungere qualunque altro nodo compiendo al massimo H passaggi, dove H è l'altezza dell'albero. Supponendo che l'albero sia binario, la sua altezza sarà pari a $\log_2 N$ dove N è la dimensione dello spazio degli indirizzi. Il tempo per effettuare il routing sarà quindi $O(\log N)$. I collegamenti fra i nodi saranno gestiti come connessioni a livello di trasporto o semplici riferimenti contenuti all'interno della routing table. Gli algoritmi di gestione della tabella dovranno quindi essere in grado di inicializzarla all'avvio del nodo e aggiornarla nel tempo considerando la dinamicità della rete. In particolare è necessario progettare un protocollo in grado di:

- c. Rilevare i riferimenti per inicializzare una routing table di nodo appena aggiunto alla rete.
- d. Aggiungere un nuovo nodo fra i riferimenti delle routing table esistenti dove necessario.
- e. Aggiornare le routing table a seguito di una partenza volontaria di un nodo.
- f. Aggiornare le routing table a seguito di un failure di un nodo.

La progettazione di questi meccanismi deve tenere conto di una serie di caratteristiche desiderate:

Robustezza La rete sarà probabilmente costantemente sottoposta a parecchi join, leave e failure di nodi in contemporanea. Il sistema deve quindi essere sufficientemente robusto per continuare a fornire i servizi di key based routing anche sotto queste condizioni.

Affidabilità Riguarda il livello di correttezza dei risultati di un key based routing in una rete in cui i nodi cambiano dinamicamente (possibilmente in contemporanea all'esecuzione dell'operazione di routing). Difficilmente è possibile dare garanzie in questo senso, ma si possono progettare reti con probabilità di correttezza più o meno elevate. Inoltre gli strati superiori dovranno tenere conto dell'inaffidabilità della rete e affrontarla attraverso opportuni meccanismi di replicazione.

Ottimalità In genere la tabella di routing e quindi i collegamenti fra i vari nodi possono essere composti in una varietà di modi, tutti ugualmente corretti. L'ottimalità riguarda la creazione di tabelle di routing in modo da ottimizzare certe caratteristiche, come ad esempio la velocità di trasmissione dei messaggi.

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--

Minimo overhead Il sistema di gestione delle routing table richiede un protocollo che può consumare una quantità di banda non indifferente. Questo overhead va considerato, e se possibile va ridotto. Ad esempio in Kademia molti sforzi sono stati fatti nel tentativo di rendere la rete il più possibile simmetrica, in modo tale che i messaggi scambiati per gestire la routing table di un nodo possano essere utilizzati anche dai nodi riceventi per aggiornare le proprie tabelle.

A partire da routing table che presentano le caratteristiche descritte, l'algoritmo di key based routing è concettualmente molto semplice. Partendo da un nodo è sufficiente seguire il riferimento con indirizzo più vicino all'indirizzo di destinazione presente nella tabella di routing, fino a raggiungere un nodo che non contiene riferimenti a nodi più vicini di se stesso. Tale nodo è la destinazione. Il routing può essere effettuato in due modi. Nel primo modo, detto ricorsivo, il messaggio è spedito da un nodo all'altro fino a raggiungere la destinazione, in modo equivalente a quanto avviene nel routing per le reti fisiche. Questo sistema è generalmente utilizzato nelle overlay network che effettuano l'ottimizzazione delle tabelle, in quanto in mancanza di ottimizzazione il percorso di routing potrebbe essere estremamente elevato risultando in un'overhead non accettabile. Nel secondo modo, detto iterativo, il mittente si occupa di interrogare uno ad uno tutti i nodi che si trovano nel percorso di routing chiedendo un riferimento al nodo successivo. Una volta rilevato il riferimento al nodo destinazione, il messaggio viene spedito direttamente ad esso utilizzando il routing della rete fisica. Questa seconda tecnica permette inoltre di effettuare un routing "parallelo", in cui più strade sono percorse contemporaneamente nel tentativo di raggiungere la destinazione più rapidamente. Lo svantaggio di questo sistema è però che l'overhead complessivo potrebbe essere più elevato rispetto alla prima tecnica applicata in una rete ottimizzata.

5. Servizi Applicativi

La overlay network non è altro che una base per la realizzazione dei servizi ad alto livello richiesti dall'applicazione peer to peer. Nel seguito se ne esamineranno tre:

6. Distributed Hash Table

Una tabella hash locale è un dizionario che associa contenuti a chiavi. La struttura dati è una lista di bucket, che sono generici contenitori di coppie chiave contenuto. Ogni bucket ha un indice che va da 0 al numero di bucket $N - 1$. L'inserimento di una coppia nel dizionario consiste nel calcolo del codice hash h della chiave, nel calcolo dell'indice $i = h \bmod N$, e nell'inserimento della coppia nel bucket di indice i . L'estrazione di un contenuto associato ad una chiave consiste nell'individuazione dell'indice i del bucket responsabile di

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--



tale chiave, e nella ricerca all'interno del bucket della coppia desiderata. In una tabella hash distribuita i bucket sono rappresentati dai nodi del sistema. L'indice del bucket è sostituito dall'indirizzo del nodo. A partire dal codice hash di una chiave, il nodo responsabile per la memorizzazione della coppia è quello attivo che ha indirizzo più vicino al codice hash. Si noti che questa mappatura garantisce un minimo spostamento di coppie in corrispondenza dell'aggiunta o della rimozione di un nodo, al contrario di quanto avviene nel caso locale. Un'overlay network come quelle descritte nel paragrafo precedente fornisce i meccanismi per l'individuazione del nodo con indirizzo più vicino al codice hash calcolato. A partire da questo servizio è perciò semplice progettare un protocollo per l'inserimento o la rimozione di una coppia nel dizionario distribuito. Oltre a questo sarà inoltre necessario progettare un sistema per lo spostamento delle coppie fra i nodi in caso di join o leave, in modo tale da mantenere sempre la gestione delle chiavi sui nodi corretti. Infine è necessario gestire il problema dei failure dei nodi e quello della non affidabilità della overlay network. Entrambi possono essere attenuati a piacere tramite la replicazione. In genere si usano due tecniche. La prima si può utilizzare in alcuni tipi di overlay network che permettono l'individuazione degli n nodi più vicini ad un determinato indirizzo. In questo caso l'inserimento di una coppia avviene su tutti tali nodi, mentre per la lettura è sufficiente che questa sia effettuata da un singolo nodo. La seconda tecnica consiste semplicemente nel calcolo di n codici hash differenti per ogni chiave. La coppia è quindi inserita in ogni nodo responsabile di uno di questi n codici.

7. Decentralized object location and routing

Questo servizio permette ai nodi di pubblicare risorse generalmente replicate, e di individuare la posizione delle risorse di interesse più vicine sulla overlay network. Per ogni risorsa viene calcolato un indirizzo, ad esempio come message digest di una funzione hash crittografica. Il nodo attivo più vicino a tale indirizzo è detto radice della risorsa. Il nodo che pubblica la risorsa spedisce quindi un messaggio di notifica a tutti i nodi che si trovano nel percorso verso la radice. Ognuno di tali nodi gestisce una tabella in cui viene inserito l'indirizzo della risorsa insieme ad un riferimento al nodo che l'ha pubblicata. L'individuazione del nodo che ha pubblicato una risorsa avviene spedendo un messaggio opportuno verso la radice di tale risorsa. Ogni nodo intermedio che si trova nel percorso fra il richiedente e la radice controlla se contiene un riferimento per tale risorsa nella propria tabella. Se tale riferimento esiste, il routing viene deviato verso il nodo che ha pubblicato la risorsa, altrimenti procede al successivo nodo verso la radice. Il funzionamento dell'algoritmo di DOLR risiede nel fatto che le overlay network hanno generalmente la proprietà per cui i percorsi fra due nodi A e B e un nodo C tendono a sovrapporsi con probabilità tanto più elevata quanto più ci si avvicina al nodo C. Inoltre in

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--



una rete ottimizzata i percorsi AC e BC tenderanno a sovrapporsi tanto più rapidamente quanto più sono vicini i nodi A e B. Da queste proprietà discende che il messaggio di ricerca del publisher di una risorsa intersecherà probabilmente il percorso di pubblicazione del publisher più vicino al nodo che effettua la ricerca. La ricerca darà quindi generalmente il risultato migliore fra tutte le repliche disponibili.

8. Multicast

Il multicast consiste nella spedizione di un messaggio verso un intero gruppo di nodi. La composizione del gruppo avviene dinamicamente attraverso un protocollo opportuno. Ad ogni gruppo di multicast si assegna un indirizzo univoco. Il nodo attivo con indirizzo più vicino a tale indirizzo viene detto radice del gruppo. Quando un nodo intende inserirsi nel gruppo, esso spedisce un messaggio di join verso la radice. I nodi sul percorso che ricevono il messaggio controllano quindi se fanno già parte del gruppo. Solo in caso negativo si inseriscono nel gruppo in maniera “passiva” (agendo solo come relay) e propagano il messaggio ai successivi nodi verso la radice. In ogni caso inseriscono il nodo da cui hanno ricevuto il messaggio in una apposita lista di “figli”. Quando un nodo intende lasciare il gruppo, esso manda un messaggio di leave verso la radice. I nodi sul percorso rimuovono il nodo che gli ha passato il messaggio dalla loro lista dei figli. Nel caso facciano parte del gruppo in maniera passiva e la loro lista dei figli sia vuota, si rimuovono totalmente dal gruppo propagando il messaggio ai successivi nodi verso la radice. Questo protocollo gestisce quindi un albero fra la radice e tutti i nodi che fanno parte del gruppo. Quando un qualunque nodo vuole spedire un messaggio in multicast verso un gruppo, è perciò sufficiente che lo spedisca alla radice di tale gruppo. Dalla radice il messaggio può venire poi ricorsivamente propagato a tutti i figli, finché non ha raggiunto tutte le foglie dell’albero, completando la spedizione multicast.

9. Conclusioni

I concetti esposti, sono particolarmente adatti per reti P2P concepite con un gran numero di nodi. Quello che si vuole sperimentare è l’estensione ad un livello gerarchico superiore di P2P. Tale approccio garantirebbe un servizio di disponibilità delle risorse controllabile in quanto i nodi facenti parte di tale rete di livello superiore non sarebbero semplici nodi terminali di PC utente.

Con tali presupposti si potrebbe immaginare un sistema di distribuzione geografico basato su reti “Overlay Network” che prendano spunto per il loro colloquio ed organizzazione su i concetti esposti sopra.

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--



Tale rete potrebbe essere ad accesso controllato, ed ogni nodo di essa un server di distribuzione dei contenuti. Per quanto riguarda la loro logica di distribuzione, si vuole implementare un sistema di distribuzione che cambi a seconda delle necessità stesse del nodo. Tale approccio potrebbe così far nascere una rete di auto distribuzione in base alle tipologie di contenuti richieste, che a quel nodo pervengono. Una prima implementazione di tale sistema, potrebbe basarsi sulla inferenza statistica delle richieste di contenuti ed attivare autonomamente una sequenza di replicazioni di detti contenuti da un nodo all'altro.

Tale approccio, potrebbe in sintesi, essere utilizzato per la distribuzione logica dei contenuti su diversi nodi, garantendo così all'utilizzatore finale la migliore qualità e disponibilità dei contenuti richiesti.

Riferimenti Bibliografici

- [1] Frank Dabek, Ben Zhao, Peter Druschel, John Kubiawicz, and Ion Stoica. Towards a common API for structured P2P overlays. In Proc. of IPTPS, pages 33–44, Berkeley, CA, Feb 2003.
 - [2] D. Eastlake 3rd and P. Jones. US Secure Hash Algorithm 1 (SHA1). RFC 3174, September 2001.
 - [3] P. Leach, M. Mealling, and R. Salz. A Universally Unique Identifier (UUID) URN Namespace. RFC 4122, July 2005.
 - [4] Petar Maymounkov and David Mazières. Kademlia: A peer-to-peer information system based on the xor metric. In IPTPS, pages 53–65, 2002.
 - [5] J.C. Mogul and S.E. Deering. Path MTU discovery. RFC 1191, November 1990.
 - [6] J. Postel. User Datagram Protocol. RFC 768, August 1980.
 - [7] J. Postel. Transmission Control Protocol. RFC 793, September 1981.
- Updated by RFC 3168.

PROGETTO LC3	Revisione n*	2		
--------------	--------------	---	--	--