

Indice

Indice	1
Introduzione	3
1 Qualità di Servizio nei sistemi multimediali distribuiti	7
1.1 Applicazioni multimediali	7
1.1.1 Problematiche e requisiti di un'applicazione multi- mediale	9
1.2 Qualità di Servizio	11
1.2.1 Definizioni generali	11
1.2.2 Qualità di Servizio nelle applicazioni multimediali distribuite	13
1.2.3 Gestione della Qualità di Servizio	13
1.3 Conclusioni	16
2 Soluzioni di caching per garantire Qualità di Servizio	17
2.1 Livelli di caching	17
2.1.1 Livello WWW (World Wide Web)	18
2.1.2 Livello IA (Applicazioni Internet)	19
2.2 Caratteristiche dei sistemi di caching	19
2.3 Il caching come strumento per garantire Qualità di Servizio	21
2.4 Architettura dei sistemi di caching	23
2.4.1 Caching gerarchico	24
2.4.2 Caching distribuito	25
2.4.3 Caching ibrido	26

2.4.4	Confronto tra le differenti tipologie di caching	27
2.5	Utilizzo delle metodologie di caching	29
2.6	Conclusioni	30
Conclusioni		31
Ringraziamenti		33
Bibliografia		35
Elenco delle figure		37

Introduzione

In questi ultimi anni, il campo dei sistemi multimediali distribuiti ha mostrato una straordinaria crescita grazie al lavoro svolto in molte Università e Centri di Ricerca e alla diffusione di applicazioni multimediali commerciali che rapidamente stanno emergendo e stanno modificando il modo di creare il software.

Una tipica applicazione multimediale trasporta delle informazioni all'utente finale in differenti modi, utilizzando audio e video, immagini, grafici e animazioni, in aggiunta al tradizionale testo. Il denominatore comune di tutte le applicazioni multimediali è l'elevata richiesta di risorse e di garanzie di *Qualità di Servizio*: riuscire a garantire un livello elevato di Qualità di Servizio è dunque uno degli obiettivi primari nello sviluppo delle applicazioni multimediali.

In questo lavoro di tesi l'attenzione sarà posta sullo sviluppo di un componente che andrà ad inserirsi all'interno di un'applicazione multimediale distribuita per la fruizione di presentazioni multimediali. Verranno affrontati due aspetti fondamentali che contribuiscono al miglioramento dell'applicazione: l'incremento della Qualità di Servizio e l'apertura verso altre applicazioni che utilizzano informazioni dello stesso tipo.

Mediante la realizzazione di un sistema di caching distribuito per la descrizione dei contenuti multimediali si cercherà di aumentare il livello di Qualità di Servizio che l'applicazione offre all'utente finale. L'utilizzo del caching permette una drastica riduzione dei tempi di attesa a fronte della richiesta di un dato, specialmente se la frequenza di richiesta del dato è molto elevata. Il sistema di caching sarà caratterizzato da un meccanismo di gestione delle politiche di accesso, che permetterà di ottimizzare

la scrittura dei dati all'interno delle cache, in base alla particolare politica adottata.

Il secondo aspetto che verrà preso in considerazione nello sviluppo del componente è l'apertura dell'applicazione, ossia lo scambio di informazioni con altre applicazioni dello stesso tipo. Per far fronte a questa esigenza, in questi ultimi anni, sono stati realizzati degli standard per la descrizione di informazioni tramite un linguaggio universale. Si cercherà, dunque, di creare uno schema per la rappresentazione di informazioni di tipo multimediale che sia il più possibile aderente ai moderni standard in merito.

La tesi è stata organizzata nel seguente modo.

Nel Capitolo 1 vengono presentati ed analizzati alcuni concetti base sulla Qualità di Servizio. Dopo una breve introduzione sulle applicazioni multimediali e sulle problematiche legate al loro sviluppo, si passa alla discussione delle caratteristiche e dei principi che regolano la Qualità di Servizio all'interno di un'applicazione multimediale.

L'argomento del Capitolo 2 è il caching: dopo aver dato alcune definizioni sul caching e sul funzionamento di un sistema di caching, viene messa in evidenza l'importanza che riveste il caching come strumento per garantire Qualità di Servizio. Successivamente, vengono analizzate in dettaglio le tipologie dei sistemi di caching: il caching di tipo gerarchico, di tipo distribuito e di tipo ibrido, evidenziando per ciascun tipo i pro e contro.

Per facilitare al lettore la comprensione dei capitoli successivi, nel Capitolo 3, vengono presentate le due infrastrutture in cui sarà sviluppato il modulo. Si parte dalla descrizione dei paradigmi basati sulla mobilità di codice, per poi passare alla descrizione dell'infrastruttura per agenti mobili SOMA e, infine, alla presentazione di *MUM*, l'applicazione multimediale distribuita per la fruizione di presentazioni multimediali all'interno della quale andrà ad inserirsi il nuovo componente.

Nel Capitolo 4, si cercherà di identificare uno schema per la rappresentazione delle informazioni di tipo multimediale che sia il più possibile aderente alle specifiche degli standard correnti. Inizialmente vengono perciò

analizzati due standard internazionali per la descrizione di presentazioni multimediali, mettendone in evidenza le caratteristiche fondamentali e le mancanze. Si passerà successivamente alla realizzazione di uno schema per la descrizione che racchiuda al suo interno le caratteristiche migliori dei due standard precedentemente analizzati.

Nel Capitolo 5, si entra nello sviluppo vero e proprio del modulo. In questo capitolo, infatti, viene condotta l'analisi del sistema di gestione dei contenuti multimediali. Verranno dapprima messi in evidenza i requisiti del componente. Si passerà successivamente alla loro analisi e si riuscirà a ricavare un primo schema che rappresenta le classi fondamentali del componente.

Nel Capitolo 6, si passa alla fase di progettazione del sistema di gestione dei contenuti multimediali. Partendo dalla rappresentazione ottenuta dall'analisi svolta nel precedente capitolo, il componente verrà suddiviso in più parti, e per ognuna verranno descritte le classi che la compongono. Si passerà, poi, alla descrizione delle interazioni fra le classi che compongono ogni singola parte del componente.

Nella prima parte del Capitolo 7 vengono delineate le principali scelte implementative e sono messe in evidenza le motivazioni che hanno portato a queste scelte. Nella seconda parte del capitolo, l'attenzione è infine dedicata al test del sistema: si parte dalla verifica del corretto funzionamento del componente e, infine, si passa all'analisi delle prestazioni che l'applicazione raggiunge utilizzando il nuovo componente.

Capitolo 1

Qualità di Servizio nei sistemi multimediali distribuiti

In questo primo capitolo, verranno dapprima forniti i concetti essenziali propri delle applicazioni multimediali distribuite: il concetto di presentazione multimediale, la tipologia delle applicazioni multimediali e le problematiche legate alla loro realizzazione.

La definizione dei requisiti delle applicazioni multimediali distribuite e di una buona base per il loro supporto può essere racchiusa in un unico concetto che prende il nome di *Qualità di Servizio*. Essa rappresenta l'insieme delle caratteristiche quantitative e qualitative che un sistema multimediale distribuito deve avere per fornire in maniera ottimale le funzionalità richieste.

Il concetto di Qualità di Servizio rappresenta un aspetto molto importante di tutte le moderne applicazioni multimediali distribuite: verranno, perciò, nel seguito del capitolo, analizzati gli aspetti salienti che caratterizzano il concetto di Qualità di Servizio.

1.1 Applicazioni multimediali

Lo scopo principale di un'applicazione multimediale è quello di manipolare informazioni di varia natura e di fornirle all'utente. Le informazioni vanno dal tradizionale e semplice testo, a più moderne e comples-

se tipologie, come audio, video, immagini, grafici e animazioni. Esempi di applicazioni multimediali possono essere le applicazioni per il *video on demand*, i sistemi per le video-conferenze o per la *tele-educazione*.

L'informazione viene molto spesso indicata con il termine *presentazione multimediale*. Le presentazioni multimediali maggiormente utilizzate nelle applicazioni multimediali consistono di una sequenza continua di campioni finiti aventi una stretta dipendenza temporale fra loro: tali presentazioni vengono definite *continue* e il termine *stream* è utilizzato per indicare una sequenza di campioni costituenti una presentazione continua.

Largamente parlando, le applicazioni multimediali possono essere classificate nel seguente modo:

- **applicazioni per la presentazione:** sono caratterizzate dall'utilizzo di informazioni memorizzate in uno o più dispositivi con capacità elevata. Gli utenti possono ottenere le informazioni che desiderano in tempo reale attraverso una rete a larga banda direttamente sul display dei propri dispositivi. Ad esempio, in un sistema di video on demand, essi hanno la possibilità di selezionare fra i differenti video presenti e di richiedere la presentazione del video che hanno richiesto.

La difficoltà nello sviluppo di questo tipo di applicazioni è legata all'alta mole di dati e ai severi vincoli temporali che caratterizzano le presentazioni di tipo continuo: lo sviluppo di queste applicazioni dipende dunque dai progressi nei sistemi di comunicazione, nei dispositivi di memorizzazione e nella capacità di elaborazione.

- **applicazioni per la conversazione:** sono utilizzate per veicolare comunicazioni in tempo reale tra persone situate in luoghi differenti, ad esempio, per la video-conferenza. Queste applicazioni sono molto più sensibili al ritardo delle applicazioni per le presentazioni, poiché l'informazione è creata in tempo reale e dovrebbe essere ricevuta in tempo reale. I dati prodotti sono validi per un limitato periodo e una volta che questo lasso di tempo è passato i dati diventano inutilizzabili.
-

- **applicazioni ibride:** sono le applicazioni multimediali che non possono essere inserite in nessuna delle due precedenti categorie. Esse vengono utilizzate sia per la presentazione che per la conversazione: ad esempio, un sistema per la video conferenza che permetta non soltanto la comunicazione tra i partecipanti, ma anche la visione di presentazioni memorizzate in un dispositivo.

1.1.1 Problematiche e requisiti di un'applicazione multimediale

Il denominatore comune di tutte le applicazioni multimediali è l'elevata richiesta di risorse e di garanzie di Qualità di Servizio, richiesta che aumenta all'aumentare della criticità dell'applicazione.

Nello sviluppo di un'applicazione multimediale i progettisti devono tenere in considerazione determinati limiti entro i quali muoversi: i limiti possono essere di natura fisica (ad esempio, la carenza di risorse) oppure possono dipendere dal software che l'applicazione utilizza (ad esempio, le limitazioni di un particolare protocollo di rete utilizzato).

Per comprendere le caratteristiche di un'applicazione multimediale, verranno ora prese in considerazione le principali problematiche legate al loro sviluppo e alla loro diffusione, le quali obbligano i progettisti a seguire determinati principi per poter garantire Qualità di Servizio nelle loro applicazioni.

Carenza di risorse fisiche

Uno dei fattori che maggiormente limita la diffusione delle applicazioni multimediali è l'alta richiesta di risorse fisiche: in particolare, fra tutte le risorse fisiche, quelle più interessate sono:

- **CPU:** è la risorsa più contesa in quanto è coinvolta in diverse fasi della fruizione della presentazione multimediale. Nel caso in cui i dati devono essere inviati essa ha il compito di codificarli e di impacchettarli; se i dati devono essere ricevuti essa deve spaccettarli,
-

decodificarli e poi effettuare il *rendering*. Queste sono solo alcune delle più importanti operazioni che una CPU deve effettuare sui dati multimediali;

- **Banda trasmissiva:** l'alta mole di dati che caratterizza una presentazione multimediale fa sì che un'applicazione multimediale abbia bisogno di un'elevata banda trasmissiva per garantire la fruizione corretta della presentazione multimediale. Sono stati sviluppati, per questo motivo, degli algoritmi di compressione dei dati, il cui utilizzo riduce notevolmente la richiesta di banda trasmissiva ma va ad influire sulle prestazioni legate alla CPU, in quanto essa è costretta ad un carico maggiore di lavoro;
- **Memoria:** la trasmissione e la visualizzazione delle presentazioni multimediali richiedono un'elevata quantità di memoria. Essa viene utilizzata per effettuare operazioni di *buffering* dei dati, in maniera da continuare la visualizzazione in caso di momentanei ritardi nella trasmissione, o per memorizzare i dati durante le fasi di codifica/decodifica.

Mancanza di espressività nelle piattaforme di supporto

Un secondo problema nello sviluppo delle applicazioni multimediali è la carenza di un supporto che garantisca la loro corretta realizzazione.

Sia a livello delle infrastrutture di rete sia a livello dei sistemi operativi, viene infatti prevalentemente fornito il supporto per lo sviluppo di applicazioni con architetture di tipo *best-effort*, che non offrono garanzie di Qualità di Servizio.

Ci sono due modi per far fronte a questa carenza: agire a livello del sistema operativo, intervenendo sul kernel e inserendo opportune funzionalità, oppure sviluppare uno strato intermedio che realizzi il supporto necessario per la gestione delle risorse. Questo strato intermedio, che viene comunemente indicato con il termine *middleware*, è posto tra il sistema operativo e l'applicazione ed ha il compito di fornire a quest'ultima le funzionalità che il sistema operativo non è in grado di offrirle.

Eterogeneità delle piattaforme e dei dispositivi

Un altro problema è dovuto all'eterogeneità delle piattaforme e dei dispositivi con cui si accede al contenuto del sistema.

L'adozione di un middleware riesce a facilitare il superamento di questo problema, nascondendo agli sviluppatori dell'applicazione multimediale i dettagli della piattaforma sottostante.

Quando, però, il dispositivo dispone di una capacità limitata per riuscire a ospitare il middleware, occorre studiare delle soluzioni particolari per ogni singolo dispositivo.

1.2 Qualità di Servizio

1.2.1 Definizioni generali

Il concetto di Qualità di Servizio (*Quality of Service, QoS*) può essere così definito: *La Qualità di Servizio è l'effetto complessivo della prestazione del servizio, che determina il grado di soddisfazione di un utente del servizio [?].*

Una definizione di Qualità di Servizio, più specifica della precedente, può essere: *La Qualità di Servizio è descritta in termini di un insieme di caratteristiche, rilevate dall'utente, che riguardano le prestazioni di un sistema. Essa è espressa in un linguaggio comprensibile all'utente e ed è rappresentata da un numero di parametri, ognuno dei quali ha un valore soggettivo o oggettivo [?].*

La Qualità di Servizio include parametri che esprimono il comportamento delle prestazioni del sistema e parametri che esprimono le caratteristiche di altri servizi, quali, ad esempio, la protezione o la priorità. Esistono due tipologie di parametri:

- I **parametri oggettivi**, che possono essere direttamente osservati e misurati nei punti in cui l'utente accede al servizio (ad esempio, ritardo, banda trasmissiva, etc.);
 - I **parametri soggettivi**, che dipendono dalle apparecchiature dell'utente (ad esempio, la qualità degli altoparlanti) e dalle sue impressioni, e che non possono essere misurati direttamente.
-

I giudizi sulla Qualità di Servizio provengono da due parti (come è mostrato nella figura 1.1): il fornitore (*provider*) dei servizi e il cliente (*customer*). Nella vita reale, le due opinioni spesso sono differenti, dato che ciascuna parte si basa sui propri termini di giudizio. I fornitori creano un servizio implementando specifiche funzioni e procedure, utilizzando in molti casi, servizi aggiuntivi con proprie prestazioni individuali.

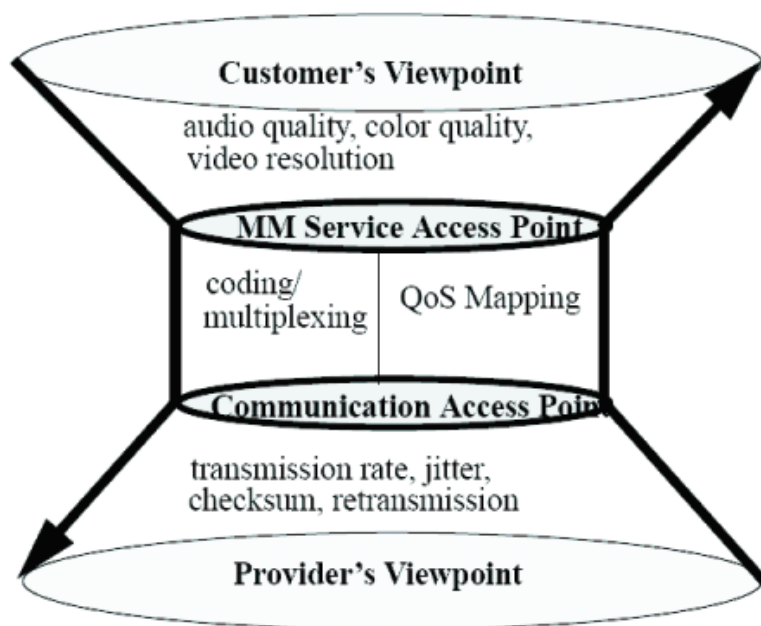


Figura 1.1: Differenti viste della Qualità di Servizio.

Il servizio è fornito nel *Multimedia (MM) Service Access Point*: questa interfaccia rappresenta un filtro che permette di separare i parametri che misurano le prestazioni visibili dall'utente da ciò che invece riguarda le caratteristiche interne del fornitore.

Ciò che il fornitore sa sui requisiti dell'utente è ristretto alle opinioni dell'utente. L'utente, infatti, richiede parametri individuali di Qualità di Servizio, ad esempio la risoluzione del video o la qualità del colore, per le sue applicazioni.

1.2.2 Qualità di Servizio nelle applicazioni multimediali distribuite

Per soddisfare i requisiti delle applicazioni multimediali, tutti i componenti del sistema devono riuscire ad ottenere un certo livello di Qualità di Servizio. Ciascun componente, sia esso hardware o software, realizza una particolare funzione, che può essere l'elaborazione degli stream di dati oppure la loro trasmissione.

I componenti fondamentali sono i sistemi di comunicazione (come, ad esempio, l'insieme dei protocolli oppure le reti) e i sistemi finali (cioè quelli utilizzati dall'utente).

In un sistema per il video on demand, per fornire una presentazione con una desiderata Qualità di Servizio, il server che contiene la presentazione, la rete, i protocolli di trasporto, i *codec* e l'applicazione utente, devono garantire un certo livello di Qualità di Servizio (Figura ??). Ciascuno di questi componenti è responsabile di sé stesso, attraverso dei meccanismi di prenotazione delle risorse, per raggiungere determinati livelli di ritardo, o altro, in maniera da fornire lo stream in accordo con i requisiti dell'utente.

1.2.3 Gestione della Qualità di Servizio

Un tipico scenario per l'utilizzo della Qualità di Servizio in un applicazione multimediale distribuita coinvolge i seguenti passi:

1. L'utente richiede un servizio multimediale, ad esempio una presentazione, con una desiderata Qualità di Servizio. In questo modo, l'utente può specificare il valore di un insieme di parametri, ad esempio la risoluzione di un video.
 2. In base alla richiesta dell'utente il sistema si configura, prenotando le risorse necessarie a garantire la Qualità di Servizio richiesta.
 3. Per assicurare che tutti i componenti possano garantire la Qualità di
-

Servizio richiesta, avviene una negoziazione della Qualità di Servizio.

4. Una volta che il sistema è stato configurato correttamente, l'utente può finalmente ricevere quello che ha richiesto.

In ogni caso, durante la fase di fruizione del materiale multimediale da parte dell'utente, il sistema può monitorare la Qualità di Servizio correntemente fornita e prendere le appropriate decisioni quando non sia più possibile garantire quella richiesta dall'utente (ad esempio, a causa di una congestione della rete). In base alle differenti applicazioni multimediali, tre azioni possono essere intraprese: la sessione applicativa è abortita, viene rinegoziata la Qualità di Servizio con l'utente, oppure l'applicazione cerca di adattarsi come meglio può.

Per assicurare che i requisiti richiesti dall'utente siano soddisfatti è essenziale la gestione della Qualità di Servizio. Essa consiste in un insieme di attività che permettono al fornitore dei servizi di garantire la Qualità di Servizio desiderata. Le sue funzioni possono essere così descritte:

- **Specifica:** l'utente deve essere in grado di specificare i requisiti desiderati tramite un linguaggio a lui familiare. In generale, l'utente specifica i valori per una lista di parametri rilevanti: i valori richiesti possono essere espressi in termini di limiti superiori e inferiori, possono essere dei valori esatti o possono essere di altro tipo, ad esempio una lista di valori accettabili. Un altro approccio per la specifica della Qualità di Servizio è quello di usare delle classi, dividendo i requisiti dell'applicazione in: realizzabili, irrealizzabili, dipendenti dal tempo e indipendenti dal tempo. Questo approccio, però, è incompleto e poco flessibile, e perciò tende a non essere utilizzato.
 - **Mapping:** si tratta della traslazione delle rappresentazioni della Qualità di Servizio nei differenti livelli del sistema: la traslazione dei requisiti dell'utente nei parametri rilevanti per il fornitore del servizio è un aspetto di fondamentale importanza per riuscire a supportare ogni possibile richiesta.
-

- **Negoziazione:** il ruolo di questa fase è quello di trovare un accordo tra i valori richiesti dall'utente per la Qualità di Servizio e il sistema. Quando l'utente richiede una certa Qualità di Servizio, il sistema verifica se può garantirla: in caso affermativo, alloca le risorse necessarie e fa partire la comunicazione; in caso negativo, fa partire una fase di negoziazione per riuscire ad accordarsi con l'utente. Se anche questa fase non ha esito positivo, la richiesta dell'utente viene respinta.
- **Monitoraggio:** questa fase permette una continua misurazione della Qualità di Servizio attualmente fornita. Essa fondamentalmente si suddivide in due parti:
 - Individuare e notificare ogni violazione della Qualità di Servizio e possibilmente anche la causa.
 - Memorizzare le informazioni acquisite.

Non bisogna tuttavia abusare di questa fase perché richiede un utilizzo di risorse che vengono sottratte alle risorse complessive del sistema.

- **Controllo:** permette di prendere dei provvedimenti quando l'utente viola la Qualità di Servizio inizialmente accordata col sistema. Essa permette di proteggere il sistema da utenti scorretti (siano essi consapevoli o no) che possano influire sulle garanzie dell'esistente Qualità di Servizio.
 - **Adattamento:** il compito di questa fase è quello di mantenere, nel modo migliore possibile, la Qualità di Servizio accordata durante la fase di negoziazione. Un protocollo di adattamento viene eseguito ogni qual volta che viene rilevata una violazione alla Qualità di Servizio. Quando il mantenimento della Qualità di Servizio accordata non è possibile, essa deve essere in grado di contenere i danni della degradazione delle prestazioni.
-

- **Rinegoziazione:** questa fase può essere iniziata o da parte dell'utente o da parte del sistema. L'utente, ad esempio, potrebbe voler richiedere una qualità migliore, mentre il sistema potrebbe iniziare la fase di rinegoziazione a causa di una riduzione delle risorse e quindi ridurre la qualità fornita per evitare l'interruzione del servizio.
- **Stima dei costi:** implica il calcolo del costo relativo ad un servizio richiesto dall'utente. Poiché le applicazioni multimediali distribuite forniscono una vasta gamma di classi di Qualità di Servizio, la stima dei costi è un'attività molto complessa, ma di fondamentale importanza: senza questa fase, infatti, il sistema potrebbe rischiare di bloccarsi, non riuscendo ad eseguire più nessuna operazione, a causa dei provvedimenti non presi per limitare le richieste da parte degli utenti, nel caso in cui queste stiano sottraendo tutte le risorse a disposizione del sistema.
- **Terminazione:** Quando il servizio è terminato, le risorse riservate possono essere rilasciate. Ogni componente viene informato della terminazione dell'applicazione e del rilascio delle risorse: il sistema, quindi, può procedere all'aggiornamento di tutte le informazioni.

1.3 Conclusioni

In questo primo capitolo sono stati messi in evidenza i meccanismi che permettono ad un'applicazione multimediale distribuita di garantire la Qualità di Servizio.

Nel prossimo capitolo, l'attenzione verrà posta su una modalità che può essere utilizzata per garantire i principi di Qualità di Servizio presentati in questo capitolo: il caching.

Verranno analizzati gli aspetti del caching e le diverse tipologie dei sistemi di caching, per riuscire ad identificare un buon modello per il modulo che è oggetto di sviluppo del lavoro di tesi.

Capitolo 2

Soluzioni di caching per garantire Qualità di Servizio

Come è emerso dall'analisi fatta nel precedente capitolo, per riuscire a garantire Qualità di Servizio in un'applicazione multimediale distribuita occorre rispettare dei principi e mettere in pratica delle determinate politiche. In questo capitolo verrà preso in considerazione l'utilizzo del *caching* all'interno di un'applicazione distribuita come strumento atto a garantire Qualità di Servizio.

Viene dapprima fornita una definizione di caching e una descrizione dei meccanismi utilizzati per la sua realizzazione, per poi successivamente passare all'analisi dei vari livelli e delle differenti tipologie di caching, evidenziando di ciascuna i pro e i contro. Si riuscirà, in questo modo, a determinare un modello che sarà successivamente utilizzato nello sviluppo del progetto di tesi.

2.1 Livelli di caching

Negli anni '60, per colmare il divario in termini di velocità tra la CPU e la memoria principale, Maurice V. Wilkes introdusse il seguente concetto di cache: *La cache è una memoria piccola ma veloce, con lo scopo di mantenere alcuni dati per un efficiente prossimo riutilizzo* [?].

Col passare del tempo, tale concetto è stato adottato ad ogni livello

della gerarchia dei sistemi informatici: nonostante ogni livello presenti problematiche ed esigenze particolari, i principi che derivano da tale definizione restano ugualmente validi.

Negli attuali sistemi informatici, viene fatto largo uso del caching. Si possono distinguere vari livelli di caching:

- CPU: Sistemi monoprocesso;
- SMP: Sistemi multiprocesso con memoria condivisa;
- DSM: Sistemi con memoria condivisa distribuita;
- DFS: File System Distribuiti;
- WWW: World Wide Web;
- IA: Applicazioni Internet.

Non si vuole, in questa sede, entrare nei particolari dei primi quattro livelli, in quanto ciò sarebbe superfluo ai fini del progetto di tesi. Verranno perciò presi in considerazione solamente gli ultimi due livelli, a causa della loro similarità con le infrastrutture su cui andrà a svilupparsi il progetto di tesi.

2.1.1 Livello WWW (World Wide Web)

Con la crescita esponenziale, che in questi ultimi ha caratterizzato il World Wide Web, il traffico sulla rete Internet rischia di diventare un problema veramente critico. Diverse direzioni vengono percorse per far fronte in maniera efficiente ed economica a questo problema.

La soluzione più ovvia sarebbe quella di aumentare la larghezza di banda della rete, ma tale soluzione non è per il momento né economicamente né tecnicamente affrontabile. Anche se sono stati ottenuti dei miglioramenti a livello di protocolli, questi non sono del tutto sufficienti.

Per questo motivo, l'uso massiccio ed ottimizzato del caching può determinare forti riduzioni di traffico sulla rete, miglioramenti a livello di carico sui server e riduzione dei tempi di attesa per gli utenti del Web.

All'interno di Internet, si fa utilizzo del caching in tre differenti entità: nei *Web Browser*, nei nodi della rete, nei *Web Server*. Il caching ha una caratteristica comune in tutte e tre queste entità: si tratta della modalità di funzionamento. Quando, in ciascuna delle tre entità, perviene la richiesta di un documento, se è presente nella cache una copia, il richiedente riceve una copia di tale documento; se il documento non è presente in cache, la richiesta viene inoltrata verso un'altra entità (che potrebbe essere un'altra cache o il server di partenza).

2.1.2 Livello IA (Applicazioni Internet)

Questo livello racchiude applicazioni distribuite più generiche rispetto ai sistemi Web e comprende una molteplicità di casi, con caratteristiche molto differenti. Anche se le implementazioni concrete possono differire enormemente, la modalità di gestione della cache risulta un aspetto che le accomuna.

2.2 Caratteristiche dei sistemi di caching

La funzionalità base di una cache è molto semplice; tuttavia, una sua efficace realizzazione può dipendere da più fattori. Il progetto di un sistema di caching efficiente, infatti, risulta essere tutt'altro che banale e implica la necessità di dover fare numerose scelte: quante unità di cache utilizzare, dove sistemarle geograficamente, come strutturarle, quali dati ogni cache dovrà mantenere, per quanto tempo, e così via.

Ogni problema che si presenta può essere affrontato seguendo diverse politiche. Per poter essere analizzate in maniera più organica possono essere distinte le seguenti classi (non si vuole entrare tuttavia nei particolari di ogni classe perché ciò risulterebbe superfluo ai fini del progetto di tesi. Una trattazione più esaustiva si trova in [?]):

- **prefetching**: si propone di anticipare e prevedere le esigenze future degli utenti, attraverso l'analisi di particolari elementi, come, ad

esempio, le richieste passate degli utenti oppure i documenti correlati con quelli richiesti. In questo modo, possono essere memorizzate nella cache le copie dei documenti che si ritiene saranno richiesti in futuro, per averli già a disposizione;

- **routing:** poiché i sistemi di caching sono organizzati su più nodi, esiste il problema del percorso che eventuali richieste devono seguire. Il problema è simile al *network routing*, anche se non può essere affrontato con gli stessi strumenti, a causa delle differenze strutturali tra i due sistemi;
 - **placement:** in un sistema formato da più cache che cooperano tra loro, è di fondamentale importanza bilanciare il carico dei dati all'interno delle singole cache, per non sovraccaricare l'intero sistema. Questa politica si occupa proprio del modo in cui tutto ciò viene garantito. Essa riveste particolare importanza nel presente progetto di tesi in quanto, come si vedrà nei successivi capitoli, esso è focalizzato sulla realizzazione di politiche di placement all'interno di un'applicazione multimediale distribuita.
 - **coherency:** ogni singola cache deve mantenere una copia aggiornata dei documenti che contiene. Può infatti succedere che il documento originale venga modificato, mentre la copia presente nella cache non venga modificata: questa politica riguarda i modi in cui è possibile garantire la massima coerenza tra i documenti e le copie presenti all'interno del sistema;
 - **removal:** quando in una cache si esaurisce lo spazio per memorizzare i dati, si dovrà provvedere ad eliminarne alcuni. Questa politica si occupa delle modalità in base alle quali eliminare dei dati, ad esempio, rimuovendo i documenti più vecchi o quelli che occupano più risorse, a altro. Questa politica influisce più di ogni altra sulle prestazioni della cache in quanto determina quali oggetti devono essere mantenuti in cache e quali invece devono essere rimpiazzati.
-

2.3 Il caching come strumento per garantire Qualità di Servizio

Andremo, in questo paragrafo, ad analizzare per quali motivi la realizzazione di un sistema di caching all'interno di un'applicazione multimediale distribuita contribuisca al raggiungimento dei principi di Qualità di Servizio introdotti nel precedente capitolo.

L'utilizzo di un sistema di caching influisce sia sui parametri oggettivi che misurano la Qualità di Servizio sia sulle modalità con cui il sistema cerca di garantire un determinata Qualità di Servizio.

I parametri oggettivi maggiormente influenzati risultano essere il tempo di trasmissione, l'utilizzo di banda trasmissiva e la quantità di spazio utilizzato nei dispositivi di memorizzazione. I primi due ottengono dei miglioramenti dall'utilizzo del caching, mentre il terzo subisce un peggioramento.

- Il **tempo di trasmissione** dei documenti diminuisce notevolmente in quanto la memorizzazione di copie nei nodi in cui sono presenti delle cache accorcia il percorso che i documenti avrebbero dovuto affrontare per giungere al client se si fossero trovati esclusivamente nel server centrale. Più è grande la popolarità di un documento, più è probabile che una copia di quel documento si trovi in una cache più vicina a chi ne fa richiesta, riducendo così il tempo di trasmissione di quel documento;
 - L'utilizzo di **banda trasmissiva** si riduce per il fatto che i documenti, o le copie di essi, trovandosi in nodi più vicini a chi ne fa richiesta, compiono dei percorsi più brevi. Tuttavia bisogna mettere in evidenza che la cooperazione tra le entità che effettuano il caching introduce nel sistema delle informazioni aggiuntive, informazioni che, però, essendo di piccole dimensioni rispetto ai documenti, non influiscono in maniera rilevante sul consumo di banda trasmissiva all'interno del sistema;
-

- Lo **spazio utilizzato nei dispositivi** per la memorizzazione dei documenti o delle copie di essi, al contrario dei due precedenti parametri, aumenta. Ogni entità incaricata di effettuare il caching ha bisogno di una determinata quantità di memoria per memorizzare i propri dati: tuttavia, il peggioramento di questo parametro può essere considerato irrilevante sia perché il miglioramento degli altri è molto più importante ai fini del raggiungimento di una determinata Qualità di Servizio sia perché le moderne tecnologie assicurano dispositivi di memorizzazione sempre più capienti.

Le modalità utilizzate per garantire una determinata Qualità di Servizio che sono influenzate dall'utilizzo di un sistema di caching sono la negoziazione, l'adattamento e la rinegoziazione.

Durante la fase di negoziazione, il sistema cerca di accordarsi con il cliente, sulla Qualità di Servizio che deve essere fornita: nel far ciò, il sistema deve tener conto di eventuali copie del documento richiesto che siano migliori per l'utente, ad esempio, in termini di vicinanza o di banda trasmissiva disponibile.

Nella fase di adattamento, cercando di mantenere la Qualità di Servizio accordata, può far fronte ad eventuali malfunzionamenti della rete, come ad esempio la perdita del nodo dal quale si stava richiedendo il documento, fornendo come alternativa degli altri nodi sui quali è presente il documento.

Anche nella fase di rinegoziazione, la presenza di copie di un documento in nodi più o meno vicini all'entità che richiede il documento, fa sì che il sistema possa offrire differenti scenari di Qualità di Servizio.

Risulta perciò evidente che un utilizzo massiccio dei sistemi di caching riesca a migliorare le prestazioni di un'applicazione multimediale distribuita, che può così offrire livelli più alti di Qualità di Servizio. Nel progettare un sistema di caching occorre naturalmente analizzare le caratteristiche e le esigenze dell'applicazione, per poter identificare l'architettura più consona all'applicazione.

2.4 Architettura dei sistemi di caching

Molti dei moderni sistemi di caching sono formati da più entità che utilizzano una o più cache. Queste entità gestiscono le proprie cache cooperando fra di loro, migliorando in questo modo le prestazioni delle cache isolate.

Per far in modo che più entità cooperino su larga scala, migliorando effettivamente in tal modo le prestazioni dell'intero sistema, esse vengono organizzate mediante particolari architetture, che definiscono la struttura e i protocolli per garantire la cooperazione.

Negli attuali sistemi di caching, si possono distinguere due tipologie di struttura: centralizzata e distribuita. La prima è caratterizzata dalla presenza di un'unica cache in un server centrale, che ha il compito di soddisfare tutte le richieste; la seconda è invece caratterizzata dalla presenza di più cache, posizionate all'interno dei vari nodi del sistema.

Nei sistemi che utilizzano una struttura distribuita, esistono fondamentalmente due tipologie di protocolli per la cooperazione fra le cache: gerarchico e distribuito.

Prendendo in considerazione la struttura della rete Internet, la quale non è altro che il più grande sistema distribuito, possono essere messe in evidenza le caratteristiche dei due protocolli: in figura ?? è mostrata la classica topologia della rete mondiale. Tutte le reti sono interconnesse fra loro formando una struttura ad albero: tutti i client sono collegati alle reti istituzionali; le reti istituzionali sono collegate alle reti regionali; le reti regionali sono collegate alle reti nazionali; infine, le reti nazionali sono interconnesse tra loro tramite collegamenti transoceanici o via satellite.

La struttura appena presentata può essere schematizzata, come in figura ??, mettendo in evidenza le cache presenti all'interno di ogni rete e di ogni nodo.

Questa struttura resta invariata sia che si utilizzi un protocollo di caching gerarchico sia che se ne utilizzi uno distribuito: a cambiare è solamente il modo in cui le cache cooperano fra loro.

Esiste, tuttavia, una terza tipologia di protocollo, caratterizzata dal-

l'utilizzo di funzionalità presenti sia nel caching gerarchico sia in quello distribuito.

2.4.1 Caching gerarchico

La cooperazione tra le differenti cache si basa fundamentalmente sul fatto che ogni cache può comunicare solamente con quella del livello superiore (ad eccezione della cache presente nella radice della gerarchia) e con quelle degli eventuali figli. Il protocollo stabilisce come devono essere fornite le funzionalità di inserimento, aggiornamento, eliminazione e richiesta dei documenti.

Consideriamo, ad esempio, il caso di richiesta di un documento da parte di un'entità nel livello client: nel caso in cui la richiesta non potesse essere soddisfatta dalla cache del client, essa verrebbe inoltrata alla cache della rete istituzionale. Se anche nella cache della rete istituzionale la richiesta non potesse essere soddisfatta, essa verrebbe inoltrata alla cache del livello superiore, in questo caso alla cache della rete regionale. Si procederebbe così, giungendo fino alla cache del livello più elevato della gerarchia, nel caso in cui la richiesta non potesse essere soddisfatta da nessuna cache intermedia. Se il documento non dovesse essere presente in nessuna cache, la richiesta verrebbe inviata direttamente al server di origine. Una volta che il documento richiesto venisse trovato, esso percorrerebbe la gerarchia al contrario e, in base al protocollo, potrebbe essere memorizzato da ogni cache che esso attraversa, in maniera da averlo disponibile a fronte di una successiva richiesta.

Le funzionalità di inserimento, aggiornamento ed eliminazione devono essere realizzate in maniera tale da garantire la consistenza dei dati presenti all'interno di ogni cache: ad esempio, quando si inserisce un nuovo dato all'interno della cache, bisogna far in modo che almeno la cache presente nella radice della gerarchia (e possibilmente una o più cache intermedie) sia a conoscenza del dato, per garantire la conoscenza dello stesso all'interno di tutto il sistema.

Nel caso in cui un documento venisse modificato o eliminato, tutte le

cache in cui fosse presente quel documento dovrebbero venirne a conoscenza, in maniera tale da mantenere nella cache una copia aggiornata del documento e garantire, in questo modo, la consistenza dei dati all'interno del sistema.

Ci sono molteplici problemi associati a questo tipo di architettura:

- ogni gerarchia introduce dei ritardi addizionali, che fanno aumentare il ritardo complessivo delle comunicazioni, riducendo le prestazioni del sistema;
- le cache dei livelli più elevati possono diventare dei colli di bottiglia, a causa dei ritardi introdotti dalle code di attesa;
- più copie dello stesso documento sono memorizzate in differenti livelli, con la possibilità che sussistano delle inconsistenze, nel caso in cui il protocollo non fosse stato realizzato in maniera corretta.

2.4.2 Caching distribuito

Utilizzando un protocollo di tipo distribuito, ogni cache presente all'interno del sistema coopera con le altre cache: deve esistere, perciò, un meccanismo che permetta alle varie cache di scambiarsi le informazioni necessarie per fornire le funzionalità di inserimento, aggiornamento, eliminazione e richiesta dei documenti. Alcuni di questi meccanismi sono:

- le cache possono interrogarsi a vicenda quando non riescono a realizzare una determinata funzionalità attraverso un particolare linguaggio (un esempio è *Inter Cache Protocol* [?]). Tuttavia, utilizzando questo sistema, c'è il rischio di aumentare in maniera significativa l'utilizzo della banda trasmissiva e di introdurre dei ritardi indesiderati all'interno del sistema;
- le cache possono mantenere dei sommari di ciò che è contenuto nelle altre cache, in maniera da evitare di effettuare delle richieste ogni volta che non riescono a realizzare una funzionalità. Questi sommari vengono poi periodicamente aggiornati tramite lo scambio di

informazioni tra le varie cache: per far ciò, può essere utilizzata un protocollo di tipo gerarchico, che tuttavia è utilizzata solamente per lo scambio di informazioni sulle cache e non dei documenti veri e propri;

- le cache possono basarsi su un meccanismo centralizzato di memorizzazione dei documenti, ottenuto mediante un identificatore univoco all'interno del sistema. Le cache non memorizzano delle copie dei documenti, bensì mantengono in memoria una lista degli identificatori dei documenti con l'indicazione della posizione. All'atto di richiesta di un documento, ad esempio, verrà fornita l'indicazione del nodo in cui è memorizzato il documento.

2.4.3 Caching ibrido

Il caching gerarchico e quello distribuito vengono già in larga parte utilizzati all'interno di Internet: negli USA, NLANR fornisce un'architettura per il caching gerarchico per distribuire informazioni popolari; in Europa, molti paesi hanno sviluppato dei sistemi di caching gerarchico per ridurre il numero di richieste che attraversano il già congestionato collegamento transoceanico.

Tuttavia, sono allo studio degli schemi ibridi, che racchiudono caratteristiche presenti sia nei protocolli di tipo gerarchico sia in quelli di tipo distribuito: essi si basano fondamentalmente su una suddivisione in livelli. Le entità presenti all'interno di un livello comunicano con le entità presenti all'interno dei livelli adiacenti tramite un protocollo di tipo gerarchico. All'interno di ogni livello, vengono invece utilizzati dei protocolli di comunicazione di tipo distribuito.

In [?] si trova una trattazione esaustiva in cui vengono messi a confronto i differenti protocolli di caching: ciò che emerge da questa trattazione è che l'utilizzo dei protocolli ibridi di caching riduce i ritardi, l'utilizzo della banda trasmissiva e il carico all'interno delle singole cache. Non si vuole entrare nei dettagli di tutto ciò, tuttavia, visto l'importanza che ri-

veste il risultato della trattazione, verranno ora discusse le prestazioni dei protocolli di caching in base ai principali parametri che li caratterizzano.

2.4.4 Confronto tra le differenti tipologie di caching

In questo paragrafo, verrà preso in considerazione il comportamento dei differenti protocolli di caching descritti nei precedenti paragrafi, sulla base dei parametri quantitativi che li caratterizzano: il tempo di connessione, il tempo di trasmissione, il tempo totale e la banda trasmissiva.

Tempo di connessione

Il tempo di connessione è la prima parte del ritardo percepito da un generico utente, dopo una richiesta di un documento.

In figura ??, è mostrato il tempo di connessione per il caching gerarchico, distribuito e ibrido, per un generico documento, la cui popolarità è indicata in figura con λ_{tot} , in un arco di tempo di 24 ore.

Come si può vedere dalla figura, le prestazioni del caching gerarchico, distribuito e ibrido sono pressoché uguali: per i documenti poco popolari (con λ_{tot} piccola) il tempo di connessione è molto alto per il fatto che la richiesta di un documento viaggia sempre verso il server di origine.

Quando il numero delle richieste di un documento aumenta in maniera significativa, c'è aumento della probabilità di trovarlo in una cache; se la richiesta di un documento varia tra una al giorno e una al minuto, il caching gerarchico e quello ibrido sono di una volta e mezzo più veloci di quello distribuito; in particolare, se la richiesta di un documento varia tra una al giorno e una al secondo, il caching ibrido è leggermente più efficiente di quello gerarchico.

Quando, infine, un documento è molto popolare, i tempi di connessione del caching gerarchico e distribuito sono veramente simili, poiché la probabilità di trovare un documento in una cache vicina è molto alta.

Tempo di trasmissione

La seconda parte del ritardo percepito da un generico utente, dopo una richiesta di un documento, è il tempo di trasmissione.

In figura ?? è mostrato il tempo di trasmissione di un documento: in questo caso, il caching distribuito e quello ibrido hanno delle prestazioni migliori rispetto al caching gerarchico quando la richiesta di un documento varia da una volta al giorno a una volta al minuto.

Nel caso in cui, invece, un documento sia poco popolare il tempo di trasmissione è molto alto, perché esso deve essere prelevato direttamente dal server di origine, mentre nel caso abbia una popolarità molto alta, la probabilità di trovarne una copia in una cache vicina è molto alta.

Tempo totale

Il tempo totale è la somma del tempo di connessione e del tempo di trasmissione. Per i documenti più grandi è molto rilevante il tempo di trasmissione che il tempo di connessione, mentre, per i documenti più piccoli, predomina il tempo di connessione.

In figura ?? è mostrato il tempo totale per un documento di grandi dimensioni, utilizzando le tre tipologie di caching: in questo caso, la situazione è simile a quella del tempo di trasmissione, con un leggero calo delle prestazioni per il caching distribuito e per quello ibrido.

Banda trasmissiva

L'utilizzo di banda trasmissiva è uno dei parametri più importanti che caratterizzano un sistema di caching.

In figura ?? è mostrato l'utilizzo della banda trasmissiva per caching di tipo gerarchico, distribuito e ibrido. In quest'ultimo caso, sono presenti due differenti configurazioni in cui varia il numero di cache cooperanti in ogni livello (k).

Come si può notare dalla figura, il caching gerarchico e quello ibrido hanno all'incirca le stesse prestazioni, al contrario del caching distribuito, il quale mostra un peggioramento quando la richiesta di un documento è

compresa tra una volta al giorno e una al minuto. Questo fatto è dovuto all'alta mole di informazioni che le cache devono scambiarsi per cooperare. Nel caso del caching ibrido, la presenza di cache intermedie riesce a diminuire il traffico tra le cache dello stesso livello, diminuendo così i ritardi e aumentando le prestazioni del sistema.

2.5 Utilizzo delle metodologie di caching

I concetti e gli strumenti descritti in questo capitolo sono alla base delle scelte effettuate nella fase di sviluppo del presente lavoro di tesi. La scelta dell'architettura di caching da utilizzare, con relativa struttura e relativi protocolli, è stata fondata sulle considerazioni sulle differenti architetture di caching presentate in questo capitolo.

Come verrà chiarito nei prossimi capitoli, il presente lavoro di tesi si prefigge lo sviluppo di un componente all'interno di un'applicazione multimediale distribuita, che realizzi dei meccanismi di caching. In particolare, l'attenzione si concentrerà sulla creazione di un sistema di caching di tipo distribuito e sulla gestione di una politica introdotta in questo capitolo, il placement.

La prerogativa fondamentale di un sistema di caching di tipo distribuito e dell'applicazione della politica di placement è la cooperazione tra le varie entità che gestiscono il sistema: per questo motivo, è stato necessario basarsi fortemente sulle considerazioni effettuate in questo capitolo al fine di scegliere una struttura e un protocollo per lo scambio di informazioni fra queste entità.

Dalle considerazioni fatte nel paragrafo 3.3, è emerso che la soluzione ideale è la realizzazione di un sistema di caching con una struttura distribuita che fa uso di un protocollo di tipo gerarchico per lo scambio di informazioni tra le varie entità, dal momento che questo sistema di caching dovrà trattare dati di piccole dimensioni.

2.6 Conclusioni

In questo capitolo è stato preso in considerazione uno dei fattori con cui è possibile garantire la Qualità di Servizio in un sistema multimediale distribuito: il caching.

L'analisi effettuata permetterà di scegliere un'opportuna architettura da utilizzare nello sviluppo di un sistema di caching per il progetto di tesi.

Nel prossimo capitolo verranno esaminate le infrastrutture sulle quali verrà sviluppato il lavoro, in maniera tale da permettere al lettore la comprensione dei successivi capitoli.

Conclusioni

Nel corso dello sviluppo del progetto di tesi sono state affrontate due tematiche molto importanti nel campo dei sistemi multimediali distribuiti: il miglioramento della Qualità di Servizio all'interno di un'applicazione multimediale distribuita mediante la realizzazione di un sistema di caching e l'apertura di un'applicazione multimediale nei confronti di altre applicazioni che fanno uso di informazioni dello stesso tipo.

Siamo partiti dall'analisi dei principi che regolano la Qualità di Servizio all'interno di un sistema multimediale distribuito, osservando come una loro corretta applicazione stia alla base dell'incremento delle prestazioni dell'applicazione.

L'attenzione si è poi concentrata sulla scelta di un sistema di caching da inserire all'interno dell'applicazione, in maniera tale da incrementarne la Qualità di Servizio. La scelta è ricaduta su un sistema di caching di tipo distribuito che faccia uso di un protocollo di caching di tipo ibrido. Questo tipo di protocollo coniuga le funzionalità dei protocolli di tipo gerarchico e distribuito, ed è sembrato quello più adatto alle esigenze dell'applicazione, dal momento che garantisce un'efficiente e funzionale gestione del sistema di caching, utilizzando nello stesso tempo una quantità limitata di risorse computazionali.

Una delle prerogative fondamentali di un'applicazione multimediale è l'apertura, ossia la sua capacità a scambiare informazioni con altre applicazioni. Dall'analisi degli standard internazionali per la rappresentazione di informazioni di tipo multimediale è emerso che ancora molto deve essere fatto in quest'ambito per definire un modello esaustivo che possa essere utilizzato dalla maggior parte delle applicazioni. È stato perciò ne-

cessario definire un nuovo schema per la rappresentazione di contenuti di tipo multimediale, prendendo in considerazione le caratteristiche migliori degli standard correnti.

Lo sviluppo è stato articolato in più fasi, con livello di dettaglio crescente: si è partiti dalla fase di analisi, per poi passare alla progettazione e, infine, all'implementazione.

È stato necessario fissare delle ipotesi di funzionamento, che in qualche modo limitano le potenzialità dell'utilizzo di un sistema di caching. Non si è voluta, tuttavia, precludere la possibilità per il futuro sviluppo di altre modalità di funzionamento: per questo motivo nella fase di progettazione del componente sono stati presi in considerazione diverse tipologie di protocolli di caching ed è stato solo in fase di implementazione che si è optato per uno di questi.

Il test del sistema è riuscito a mettere in evidenza due aspetti fondamentali del nuovo componente: il suo corretto funzionamento all'interno dell'applicazione multimediale e l'effettivo incremento delle prestazioni dell'intero sistema. È emerso, infatti, che i tempi necessari al reperimento di informazioni sui contenuti multimediali presenti all'interno del sistema diminuiscono notevolmente grazie al suo utilizzo, a fronte di un minimo aumento dell'utilizzo di banda trasmissiva necessaria allo scambio di informazioni tra le entità che compongono il sistema di caching.

La realizzazione del sistema di caching apre nuove prospettive di miglioramento, che potrebbero essere oggetto di eventuali sviluppi futuri.

La creazione di meccanismi di *quoting* per l'ottimizzazione dello spazio su disco utilizzato dalle cache oppure l'applicazione di politiche di rimozione del dato presente nella cache sono due aspetti che potrebbero essere presi in considerazione in un successivo ampliamento del componente.

Infine, in futuro si potrebbe cercare di raffinare ulteriormente lo schema per la rappresentazione delle informazioni, incrementando il livello di dettaglio della descrizione di un dato multimediale, facendo fronte, in questo modo, alla continua evoluzione dei contenuti multimediali.

Ringraziamenti

I miei ringraziamenti vanno al prof. Antonio Corradi per avermi offerto l'opportunità di sviluppare il presente lavoro di tesi e per la fiducia mostrata nei miei confronti. Ringrazio i miei due correlatori, il prof. Paolo Bellavista e l'ing. Luca Foschini, per l'assistenza, le indicazioni, i consigli e la disponibilità fornitemi sempre con puntualità e precisione.

Bibliografia

[COU01] G. Coulouris, J. Dollimore, T. Kindberg, *Distributed Systems Concepts and Design*, Third Edition, Addison-Wesley, Milano, 2001.

[HAF95] A. Hafid, G. v. Bochmann, *An Approach to Quality of Service Management for Distributed Multimedia Systems*, International Conference on Open Distributed Processing (ICODP-95), Australia, Feb. 1995, pp. 319-340.

[MUM] <http://www-lia.deis.unibo.it/research/MUM/>

Elenco delle figure

1.1	Differenti viste della Qualità di Servizio.	12
-----	---	----