A cura di Arianna Curti e Luigi Faverzani

Progettazione di un Server Video On Demand

Il video digitale

Un video tradizionale si presenta in un formato che viene chiamato “analogico”, per produrre video digitale quest’ultimo viene codificato e compresso in modo tale da poter essere memorizzato su dischi magnetici, inviato all’utente via rete e riprodotto dopo essere stato decodificato.

Uno dei vantaggi derivanti dall’uso di video digitale è la resistenza all’usura (non considerando l’eventuale degrado dei supporti di memorizzazione): il video rimane nelle condizioni originarie, anche se copiato, finché non è esplicitamente modificato o cancellato, inoltre il video può essere trasmesso da un server con sufficiente capacità di memorizzazione ai client che lo riproducono dopo averlo decompresso.

La tv interattiva: video on demand

Il servizio multimediale che ci proponiamo di offrire agli utenti della nostra rete consiste nella realizzazione di una forma di televisione interattiva basata sul Video on Demand, questo permette ad un utente di richiedere in qualsiasi momento la trasmissione di un film o di un evento sportivo a sua scelta, senza essere vincolato da orari o palinsesti prestabiliti e di pagare solo per quello che realmente vede, è una sorta di videoteca personale direttamente a casa propria, gli unici strumenti richiesti all’utente sono una connessione alla rete ed una set top box da collegare al televisore.

Le componenti

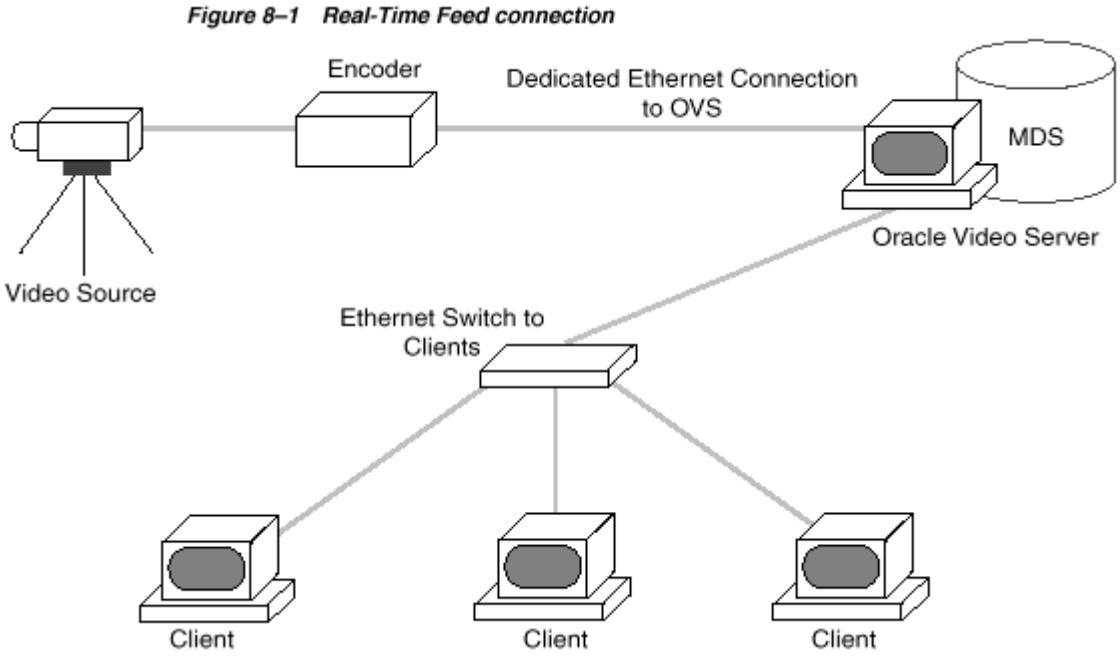
Tre sono le componenti necessarie alla realizzazione di un sistema per il VoD:

1. Un server responsabile della consegna di video stream all’interno di una rete di distribuzione, questo server può essere implementato come un semplice video tape library/player o come un sofisticato Massively Parallel Processing (MPP) computer; il livello di specializzazione dipende dal tipo di servizio che si intende offrire.
2. Una rete per la distribuzione dei video stream alle set top box. Può essere una rete pubblica usata anche per altri servizi o una rete dedicata al VoD, anche qui il tipo di rete dipende dal livello del servizio offerto.

3. Una set top box per la ricezione e la decodifica dei video in un formato per la televisione, nonché per l'invio delle richieste dell'utente al server.

Il tipo di servizio che noi intendiamo fornire è il cosiddetto "True VoD", esso consiste nell'assegnazione ad ogni utente di una comunicazione one to one con il server: c'è un unico video stream dedicato ad ogni utente, il quale ha sul flusso di immagini lo stesso controllo che avrebbe con un video registratore: ffw, rwd, pause… etc.

Inoltre siamo anche in grado di fornire il cosiddetto "real time feed" ossia la possibilità di vedere eventi in diretta, l'unico ritardo che viene introdotto è quello legato al tempo necessario per codificare il video prima di trasmetterlo; tuttavia viene offerta una funzionalità in più rispetto alla normale televisione.



Poiché le immagini sono anche bufferizzate nel server prima di essere trasmesse, è possibile quindi eseguire le stesse operazioni di pause, restart, rew che si hanno a disposizione sui film. A differenza di quello che accade con i film però la possibilità di muoversi in uno stream live è limitata (in termini di minuti) dalle dimensioni del buffer sul server. L'utente ha così a disposizione una finestra temporale di dimensioni stabilite dal fornitore di servizio.

Nel caso True VoD le funzionalità richieste alle tre componenti fondamentali sono:

1. Il server deve essere in grado, non solo di rispondere alla richiesta di un filmato specifico in tempo reale, ma deve anche rispondere a tutte le richieste dei comandi VCR_like che l'utente gli invia;
2. Per quanto riguarda la rete non è richiesta una grossa quantità di banda dedicata al singolo utente, come nel caso del Near Video on Demand, tuttavia è necessario che la capacità complessiva sia sufficientemente ampia da poter supportare il maggior numero possibile di stream concorrenti, la topologia di rete necessaria in questo caso è quella di una Switched network.
3. La set top box deve essere in grado di supportare sessioni interattive con l'utente, un canale attivo verso il server e deve decomprimere i video stream in tempo reale.

L'architettura del sistema

Il tipo di architettura che abbiamo adottato è quella progettata dalla Oracle Corporation per il loro sistema di video service, essa prevede tre "tier": client (set top box), server e database:

- Client tier: invia le richieste, decodifica e visualizza le immagini, è realizzato per mezzo dell'**Oracle Video Client** (OVC), il software che permette di ricevere stream video alle set top box;
- Server tier: realizzato per mezzo dell'**Oracle Video Server**, riceve le richieste, comunica con il database server per ottenere i titoli ed i video_files, invia in rete l'oggetto della richieste verso il client;
- Database tier: costituito dal database **Oracle 8**, mantiene e permette l'accesso accede alle informazioni (metadati) sui video disponibili e le comunica al server, gestisce inoltre le operazioni necessarie alla tassazione degli utenti.

Oracle Video Server

L'Oracle Video Server è un software altamente scalabile in grado di gestire la distribuzione di video a più clients in maniera concorrente ed in tempo reale.

Le caratteristiche principali di questo pacchetto riguardano:

- Content;
- Storage;
- Delivery;

Content

Il tipo di video che OVS immagazzina e distribuisce è chiamato **content.**

Due sono le forme in cui esso è memorizzato:

- Physical content – è formato dai file fisici contenenti dati multimediali;
- Logical content – è la rappresentazione logica dei dati che viene presentata agli utenti;

Physical content

Il contenuto fisico dei file è costituito da video codificati e compressi, nel nostro caso il formato scelto è quello MPEG-2 a 3 Mbps, esso fornisce full-motion e full-screen video con audio stereo ad alta fedeltà e Dolby Digital AC-3.

Associato ad ogni content file viene creato (tramite una tagging utility) un **tag file**, esso contiene le informazioni necessarie per eseguire il controllo del flusso di immagini (pause, seek, ffw, rwd…).

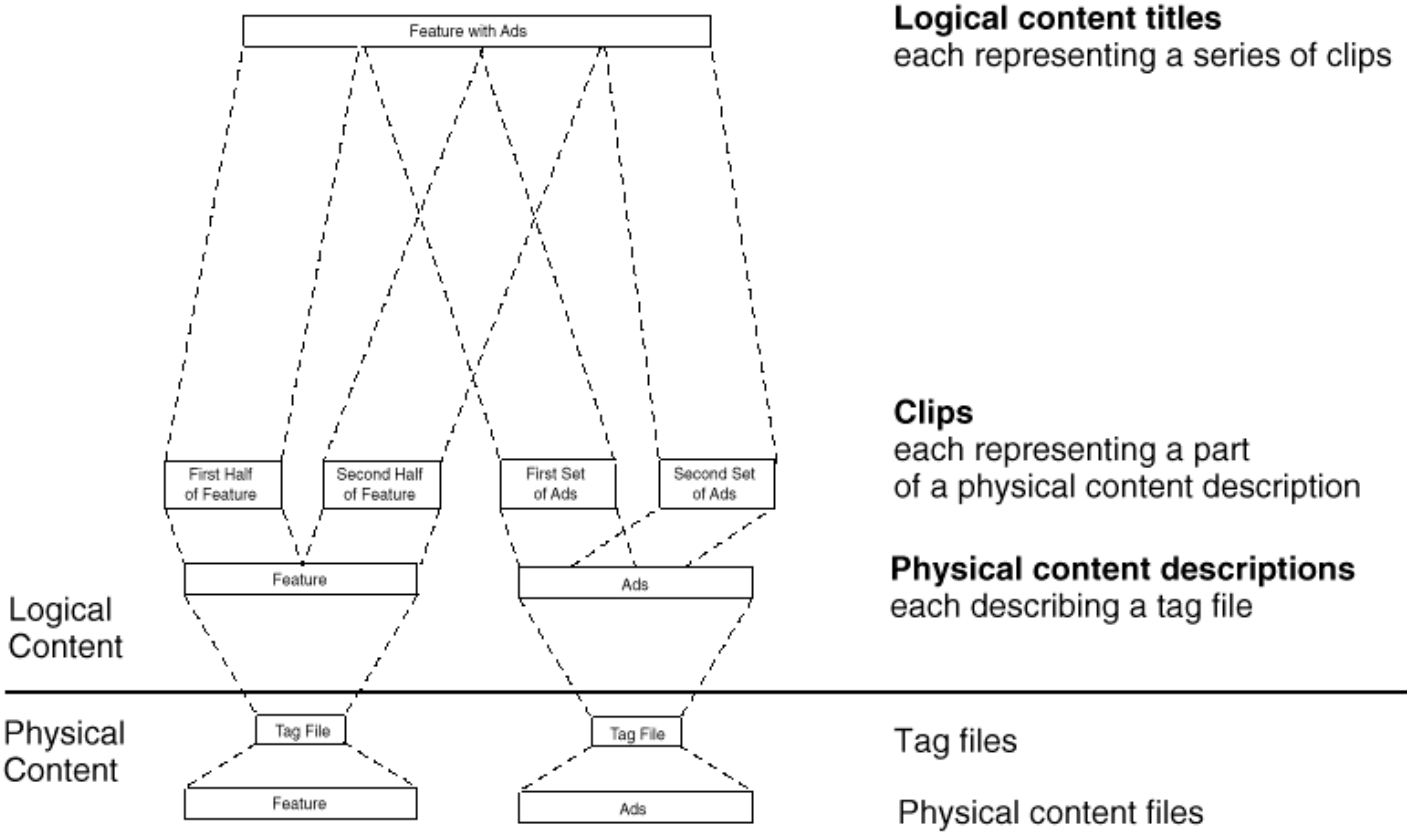
Se un'applicazione client richiede un'operazione di controllo su un content file, OVS legge il tag file associato per determinare quale parte del content file deve inviare al client.

Quando le immagini riguardano eventi inviati in diretta è necessario che il dispositivo che codifica il video sia in grado di creare in tempo reale anche il tag file, questo per permettere il controllo VRC_like anche su questo tipo di filmati.

Logical content

E' costituito da una collezione di titoli, ciascuno rappresenta una successione di porzioni di physical content file, non si tratta solo dei titoli dei film, l'amministratore del sistema può creare personalmente delle sequenze di clip, ad esempio per aggiungere in testa ad un insieme di video notizie di aggiornamento sul catalogo dei filmati disponibili da far conoscere agli utenti, o l'annuncio di un evento che verrà trasmesso in diretta, pubblicità e così via.

Figure 2–1 Logical content and physical content



Storage

I dati sono immagazzinati secondo una gerarchia, questa è gestita dal HSM (Hierarchical Storage Management) ed è composta da:

- memoria (primary storage);
- volumi MDS di dischi (secondary storage);
- DVD juke-box ed eventualmente unità a nastro di grande capacità (tertiary storage);

HSM è dotato delle routine che permettono di caricare file dai DVD/nastri ai dischi qualora i video richiesti non siano attualmente disponibili su disco.

OVS immagazzina i file fisici nell'Oracle Media Data Storage (MDS), un file system per memorizzare e trasferire video file ininterrottamente in tempo reale, e i file logici nel database Oracle8.

L'MDS riesce a garantire accesso in tempo reale anche in caso di errori nei dischi e di variazione nella latenza.

I file memorizzati sono:

- video file;
- audio file ;
- tag file;

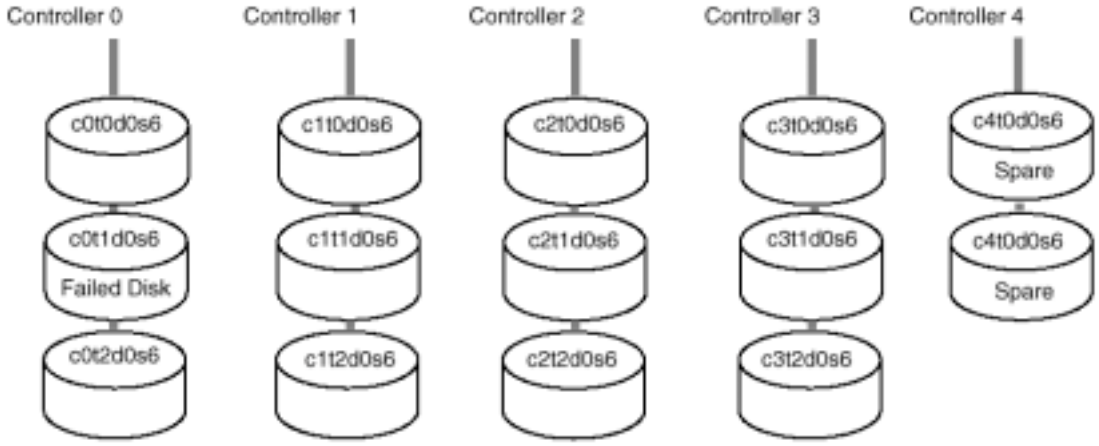
Viene forzata su tutti i file la write consistency, ossia un file può essere modificato da un solo processo alla volta, non può essere rinominato, rimosso, troncato o bloccato in read only mode mentre è in corso una modifica.

I dischi in cui sono memorizzati i file sono raggruppati in **volumes,** ogni volume ha una **Table of Content**, che elenca per i file nel volume la loro posizione sui dischi.

Ogni file è poi spezzato in **stripes**, porzioni della stessa dimensione che vengono memorizzate in modo non contiguo su dischi diversi all'interno del volume, questa tecnica incrementa enormemente il bit rate dei filmati e permette l'accesso concorrente a un medesimo file video da parte di utenti che richiedono stripes diverse.

Per garantire fault tolerance viene utilizzata un'architettura RAID: i dischi di un volume vengono divisi in sottoinsiemi di dimensione fissa (raid set) e i dati sono memorizzati con una certa ridondanza, utilizzando informazioni di parità e sono distribuiti su tutti i dischi del RAID set, in caso di fallimento di uno dei dischi del raid set i dati perduti sono ricostruiti "on the fly" basandosi sulle informazioni contenute nei dischi integri dell'insieme senza il bisogno di interrompere il servizio. I dati appena ricostruiti vengono memorizzati su dischi di scorta chiamati *spare*.

Figure 7–1 MDS Volume With Failed Disk and Two Spare Disks



Delivery

OVS è in grado di trasmettere video stream agli utenti su richiesta ed in tempo reale.

Invece di memorizzare l'intero file scaricato dal server prima di riprodurlo, il client riproduce immediatamente il video non appena arriva (un buffer di piccole dimensioni è comunque presente sull'hardware del client).

Nella richiesta il video file viene identificato per mezzo o di un logical content title o di un tag file, OVS è anche in grado di trasmettere senza interruzione una sequenza specifica di video, purché si fornisca l'insieme di tag files associato.

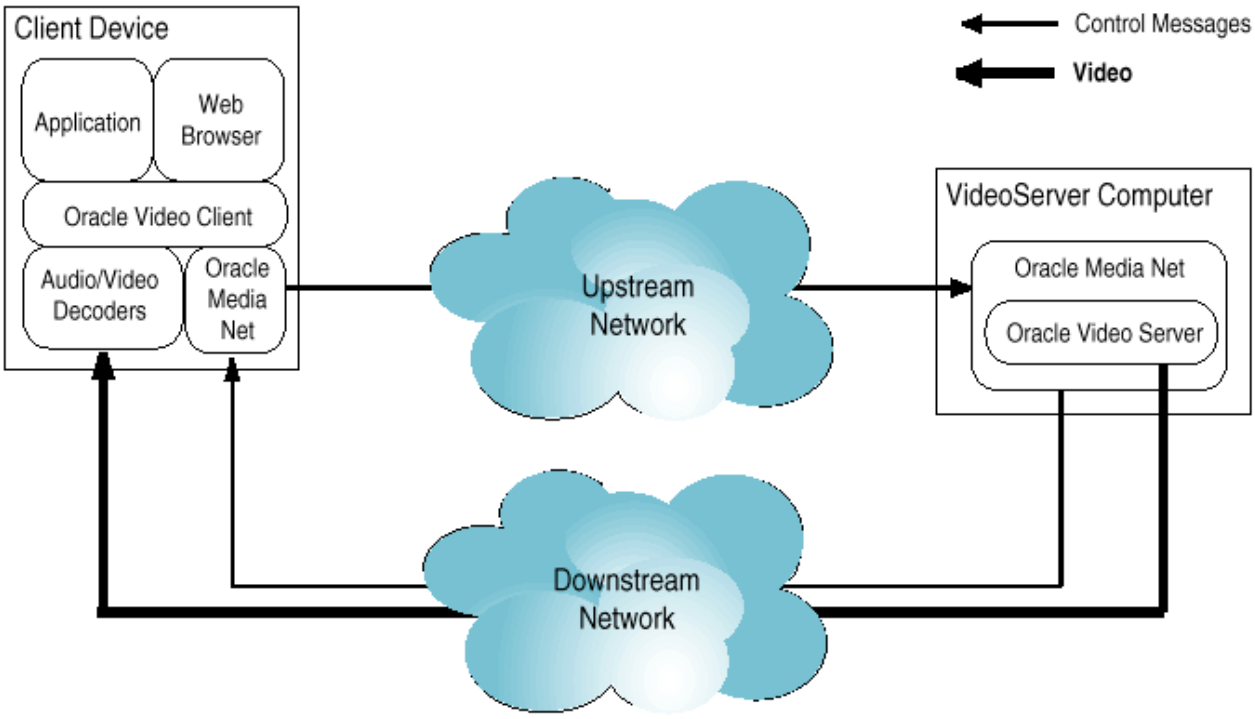
Come già evidenziato è possibile richiedere operazioni di controllo del flusso sia sui video stream tradizionali sia sulle immagini trasmesse in diretta.

I processi

Quando viene effettuata una richiesta da un client questa viene gestita da diversi processi del server che ricevono e analizzano la richiesta, recuperano i dati e li inviano al processo utente.

- **Session and Circuit service:** un circuit è un path di comunicazione tra il client e l'OVS , nella nostra implementazione assume due ruoli differenti: Upstream (trasporta messaggi di controllo, come ad esempio richieste di stream video, tra il client e il server) e Downstream (trasporta messaggi di controllo e dati video tra il server e il client).

Figure 2–10 Communication in an asymmetric network



Una Sessione viene stabilita ogni volta che un client si connette al server e comprende: un identificatore per il client, uno o più circuiti, informazioni sullo stato della comunicazione e sulle risorse.

Il servizio di Session and Circuit alloca e gestisce i canali di comunicazione in accordo con moduli di rete di terze parti (vedi Channel Manager) i quali mantengono informazioni sulla tipologia della rete che non sono direttamente accessibili dal servizio, inoltre ha la possibilità di conservare informazioni riguardo i protocolli di rete utilizzati; quando un client conclude la sua sessione il S.C.S. dealloca tutte le risorse precedentemente assegnate.

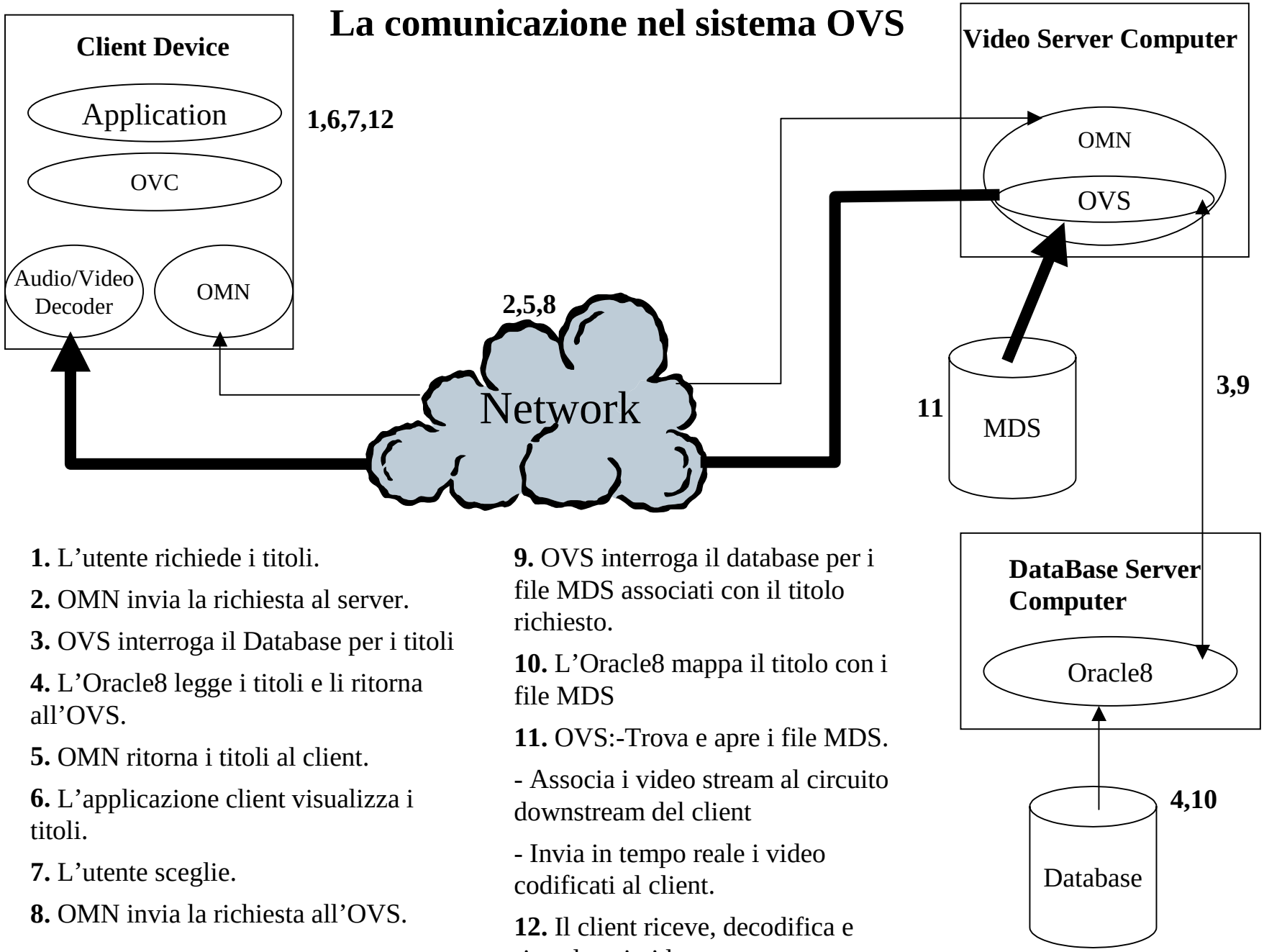
- **Stream Service:** accetta le richieste e istruisce la Video Pump sui video da inviare al client secondo la seguente modalità:

1. Contatta il Content Service per scomporre una richiesta riguardante un Logical Content Title nei Tag Files che la compongono.
2. Legge i tag File associati alla richiesta, ognuno dei quali descrive quali porzioni di video devono essere inviati.
3. Istruisce la Video Pump su quali parti del Physical Content File deve riprodurre.

- **Content Service:** mantiene nel database le informazioni per mappare i Logical Content con i Physical Content. Quando la Tagging Utility crea i tag file chiama il Content Service per generare il Logical Content basato sul contenuto. Un'applicazione client può richiedere quali video sono presenti nell'OVS, il Content Service recupera dal database la lista dei logical content e la invia al client. Il CS risponde anche alle richieste dello Stream Service per scomporre i Logical Content nei Tag Files associati a i dati fisici.
- **Media Data Store Directory Service:** l'OVS memorizza i Physical Content nel Media Data Store un file system studiato per immagazzinare e inviare video in tempo reale e senza interruzione. Il Directory Service associato controlla l'accesso ai file MDS e gestisce la loro disposizione su disco. I processi che leggono o scrivono sui file MDS devono prima ottenere l'accesso tramite il Directory Service: per esempio quando una video pump viene istruita per riprodurre video apre il file inviando un messaggio al Directory Service che gli restituisce una piccola struttura dati contenente il layout del file. Con questa informazione la video pump accede al contenuto del file direttamente evitando così che il DS diventi un collo di bottiglia nelle operazioni di I/O.
- **Video Pump:** legge i file video dall'MDS e li spedisce in rete in tempo reale. Quando un client richiede dei video, la Video Pump riceve un messaggio dallo stream service, legge la porzione di file appropriata dall'MDS e invia i dati video in rete sul canale downstream stabilito.

  La video pump inoltre fornisce le funzioni di controllo del flusso congiuntamente allo stream service.

La comunicazione nel sistema OVS



1. L'utente richiede i titoli.

2. OMN invia la richiesta al server.

3. OVS interroga il Database per i titoli

4. L'Oracle8 legge i titoli e li ritorna all'OVS.

5. OMN ritorna i titoli al client.

6. L'applicazione client visualizza i titoli.

7. L'utente sceglie.

8. OMN invia la richiesta all'OVS.

9. OVS interroga il database per i file MDS associati con il titolo richiesto.

10. L'Oracle8 mappa il titolo con i file MDS

11. OVS:-Trova e apre i file MDS.

- Associa i video stream al circuito downstream del client

- Invia in tempo reale i video codificati al client.

12. Il client riceve, decodifica e riproduce i video.

Oracle Video Server Manager (VSM)

Oracle Video Server Manager è l'applicazione che permette all'amministratore di sistema di gestire e monitorare i server OVS e i processi dei Clients, le operazioni che VSM consente di fare sono :

- avviare e bloccare istanze dei processi dell'OVS;
- osservare lo stato dei servizi più critici;
- creare e gestire logical content titles e clips;
- monitorare:
  - dischi e files dell'MDS;
  - i supporti terziari di memorizzazione;
  - le trasmissioni in tempo reale;
  - i clients dell'OVS;
- deframmentare i volumi dell'MDS;
- Caricare e registrare i physical content files.

Oracle Video Client (OVC)

E' la parte di software che è predisposta alla ricezione e riproduzione dei video provenienti dall'OVS. A runtime l'OVC software che gira sul client device si occupa di connettersi all'OVS, accettare input dall'applicazione client, richiedere video e audio all'OVS e riprodurli. L'hardware della set-top-box si occupa della decodifica e decompressione dei dati, mentre l'OVC software assicura il riordino dei pacchetti e il buffering di una piccola quantità di dati affinché possano essere processati nel giusto ordine dal decoder. L'OVC può dinamicamente connettersi e disconnettersi da differenti istanze del server senza bisogno di fermarsi e ripartire.

Oracle Media Net

Oracle Media Net è l'infrastruttura di rete che realizza la comunicazione tra OVS e i processi client in ambiente distribuito mantenendo trasparente all'applicazione la reale locazione dei servizi ed il tipo di protocollo usato.

La trasparenza riguarda anche gli stessi processi che compongono un server, quando, a causa di un sovraccarico o di guasti, un insieme di risorse non è più

disponibile localmente, OMN si occupa di incaricare una applicazione remota di eseguire le operazioni richieste.

Nel caso che la domanda di servizi ad un determinato server cresca, l'amministratore di sistema può avviare nuove istanze di alcuni processi del server, OMN distribuisce le richieste tra tutte le istanze disponibili e bilancia il carico tra queste, se al contrario un processo fallisce OMN ridistribuisce le richieste pendenti tra quelli rimasti.

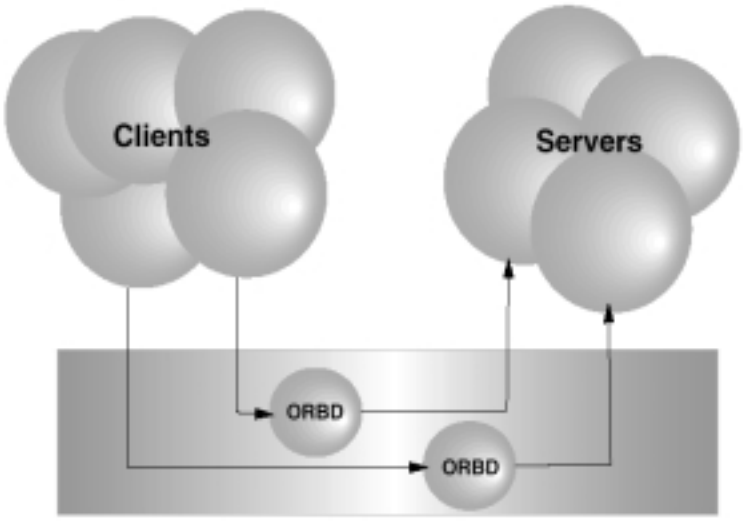
Inoltre è presente un servizio di logging col quale vengono registrate tutte le azioni compiute dai vari processi del sistema e dagli utenti, in questo modo è sempre possibile ricostruire una qualsiasi sequenza di eventi accaduti in un determinato intervallo di tempo.

I processi

I principali processi dell'OMN coinvolti nella trasmissione di video stream sono:

- **ORB Deamon:** riceve le richieste di servizi dai client ,ne bilancia il carico tra le diverse istanze ed in caso di fallimento ridistribuisce le richieste tra i processi attivi. L'ORB Deamon si occupa di gestire e mantenere trasparente alle applicazioni client e server la rispettiva posizione nella rete; inoltre si occupa di trasportare e convertire i dati tra sistemi differenti. Quando un sevizio viene avviato registra il suo nome e il suo indirizzo Media Net all'ORB Deamon.

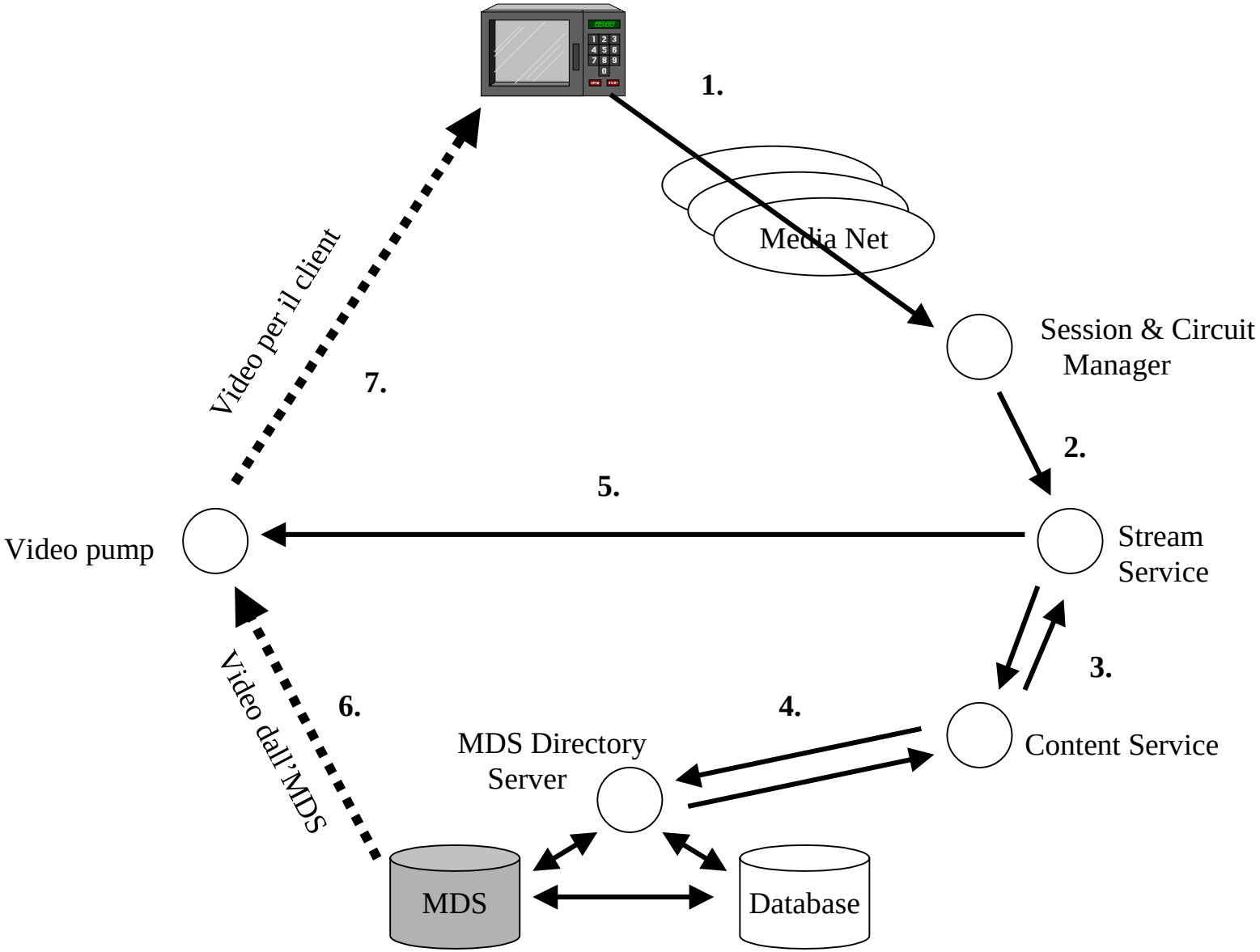
Figure 1–5 Handling Two Simultaneous Requests to the ORB



- **Name Server:** mappa i nomi dei demoni ORB con il loro indirizzo Media Net, quando un demone ORB viene avviato registra il suo nome e indirizzo Media Net al Name Server. In tutto il sistema OVS deve essere attivo uno e uno solo Name Server.

- **Address Server:** Ogni componente ha un Media Net Address che lo identifica univocamente nella rete OMN. L'Address Server mappa gli indirizzi media net con i corrispondenti indirizzi fisici:
  - quando un client device si connette all'OVS fornisce il suo indirizzo fisico e l'Address Server gli assegna un Media Net Address.
  - quando un processo server parte fornisce il suo indirizzo fisico e l'Address Server gli assegna un Media Net Address.
  - prima che un client possa inviare un messaggio al server, l'ORB Deamon (attivo sul client) contatta l'Address Server per ottenere l'indirizzo fisico del server.

  L'Address Server è unico in tutto il sistema ed è anche responsabile di invalidare gli indirizzi dei processi falliti: quando un processo è idle periodicamente invia all'Address Server un *heartbeat* che segnala il suo stato di processo attivo. Se trascorre più di un minuto tra due segnali successivi l'AS considera il processo fallito e rimuove il suo indirizzo.
- **Logging Daemon:** registra gli errori ed i messaggi di warning provenienti dalle diverse componenti del sistema nei logfiles.
- **Log Reader:** interpreta e rende periodicamente disponibili alla lettura (da parte dell'amministratore di sistema) le informazioni memorizzate nei log files.


1.
Media Net
Session & Circuit
Manager
2.
5.
Video pump
Stream
Service
3.
4.
6.
7.
Video per il client
Video dall'MDS
MDS Directory
Server
Content Service
MDS
Database

Data management system

Le informazioni utilizzate in applicazioni come la tassazione degli utenti o il monitoraggio del traffico, sono gestite da un database replicato Oracle8.

Il Ralational Database Management system (RDBMS) supporta il linking e la distribuzione delle differenti istanze del db fornendo:

- Location transparency – risolve in modo trasparente i riferimenti ai dati;
- Distributed queries – permette l'esecuzione di query che si riferiscono sia a dati locali che remoti;
- Distributed update – è in grado di eseguire modifiche sui dati sia locali che distribuiti;
- Replicazione dei dati e risincronizzazione – può gestire la replicazione dei dati e ritardare la sincronizzazione degli update distribuiti fatti sui dati replicati, ciò è particolarmente importante nei sistemi di VoD perché in questo modo si possono rimandare gli updates sui dati ai momenti in cui il carico in rete è minore.

Dal momento che un servizio di VoD prevede che gli utenti paghino per i servizi di cui usufruiscono, è necessario introdurre una procedura di tassazione, le informazioni che occorrono sono sempre gestite dal database e comprendono:

- Dati anagrafici dell'utente;
- Un identificatore per l'utente;
- Una password per l'accesso ai servizi;
- Eventuali restrizioni alle scelte dei servizi;
- Informazioni per l'accredito (es. numero di carta di credito o recapito)
- Storia dei pagamenti effettuati;
- Debito attuale e importo del prossimo pagamento;

Vengono inoltre gestite tutte le informazioni necessarie per stendere statistiche di marketing: i titoli dei film più richiesti, le date e la durata dei collegamenti, quanti film diversi sono stati richiesti, i tipi di VCR controls usati, etc.

Componenti Hardware

Il server

Per quel che riguarda il server, compatibilmente con le specifiche fornite dalla Oracle, abbiamo deciso di adottare il prodotto della nCUBE: il sistema MediaCUBE.

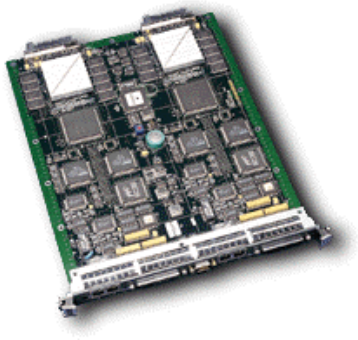
Media CUBE è stato progettato per garantire la distribuzione di una grande quantità di materiale multimediale (video digitale) su di una rete, fornendo:

- Potenza di calcolo;
- Banda di comunicazione;
- Memoria di sistema;
- Memorie di massa;
- Connettività di rete;
- Video delivery

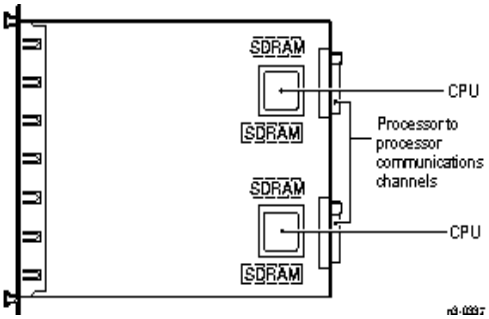
Può essere configurato secondo le esigenze delle diverse applicazioni, noi abbiamo scelto la versione più potente, il Media CUBE 3000.

Le componenti base che abbiamo utilizzato sono:

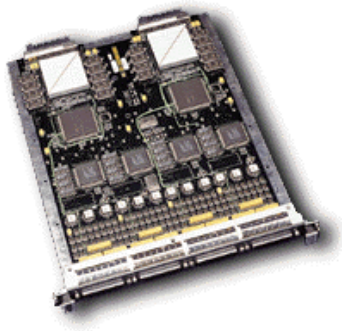
- *Ethernet Server Module (ESM):* ESM riceve i dati in input dai dispositivi di storage attraverso due porte ultra wide SCSI e li trasferisce sulla rete attraverso 4 porte ethernet 100 BaseT, ha posto per 2 processori nCUBE3, ognuno dei quali è in grado di supportare fino a 40 stream da 3 Mbps, ed è dotato di 64 MB di Dram.



- *Media Control Module (MCM): Gli* MCM permettono di aggiungere altra capacità di calcolo e banda di interconnessione al sistema, di solito sono usati per i processori dedicati al controllo del flusso (ffw, rwd, etc).



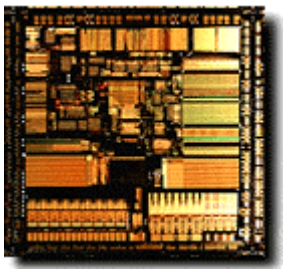
- *Native SCSI Module (nSCSI):* connette i dispositivi di storage (dischi o nastri) attraverso 4 porte UW SCSI, ha due processori per gestire l'I/O.



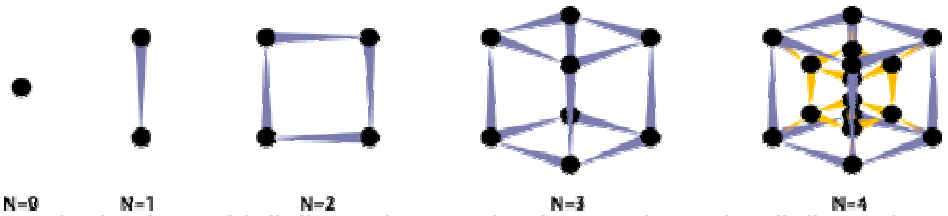
- *PCI Module*: è una scheda che supporta fino a due schede PCI e contiene un solo processore nCUBE3.

I moduli sono combinati in uno o più shelves, ognuno dei quali ha lo spazio per contenere una qualsiasi combinazione di 48 moduli. Ogni modulo ha installati due processori, questi sono connessi secondo una topologia ad ipercubo.

I processori nCUBE3



I processori, proprietari, installati sul server sono degli nCUBE3. Questi sono stati studiati e progettati per essere utilizzati in un'architettura a passaggio di messaggi, sono direttamente accoppiati ai banchi di Synchronous DRAM e a gli altri nodi (processori) formando una rete ipercuboidale per mezzo di collegamenti bidirezionali. Un ipercubo è un cubo n-dimensionale, questa topologia di connessione realizza il cosiddetto Massive Parallel Processing (MPP) l'unico in grado di garantire banda passante e capacità di I/O sufficienti per realizzare True VoD, man mano che vengono aggiunti processori il cubo cresce di dimensioni.



Sommando due ipercubi di dimensione N si ottiene un ipercubo di dimensione N+1.

Questo sistema di interconnessione permette di raggiungere una dimensione dell'ipercubo pari a nove e quindi di collegare tra loro fino a 512 processori. l'nCube viene definito *fully-integrated-single-chip* ossia incorpora tutti i diversi sottosistemi che normalmente troverebbero posto in chip separati. Ogni singolo chip è costruito in tecnologia a 0,5 micron, contiene 2,7 milioni di transistor, funziona ad una tensione di 3,3 volt e integra diverse funzionalità:

- Unità aritmetica a 64 bit per gli interi
- Unità floating-point a doppia precisione
- MMU a 64 bit
- Controller per l'accesso all'SDRAM, è possibile indirizzare fino a un Gbyte di memoria fisica per processore, con una banda passante di 640 Mbyte/s
- Cache per dati e istruzioni, 8kbytes ciascuno
- Pipelining per dati e istruzioni
- 16 canali DMA (8 in Send e 8 in Receive) per la comuinicazione con la memoria
- Router adattivo dei messaggi con 18 porte di I/O

Routing dei messaggi

Essendo questo un sistema basato su scambio di messaggi e essendo la topologia della “rete” interna dinamica (dipende dal numero di processori installati), è necessario un sistema di instradamento dei messaggi. La nCube usa sette diversi metodi di instradamento. Il più importante di questi è il *Maze Adaptive Routing* che permette di individuare un tragitto valido tra due *interlocutori* bypassando nodi/porzioni di tragitti bloccati, guasti o sovraccarichi. Viene cercato il tragitto migliore tra due nodi testando i percorsi candidati con un pacchetto *scout* partendo da i più brevi e passando poi a quelli più lunghi. Quando il pacchetto scout torna al mittente un viene individuato un path. Questo algoritmo, come gli altri utilizzati è deadlock free.

Dimensionamento del sistema

Per dimensionare il sistema ci siamo basati sui risultati di un test realizzato alla nCube che ne ha testato il funzionamento per la trasmissione di 5000 video stream concorrenti a 3Mbps, il sistema ha continuato a funzionare correttamente per tutte le 72 ore di trasmissione, inoltre abbiamo seguito le indicazioni contenute nel “Administrator Guide” dell’ ”Oracle video server for nCube”.
Gli elementi da tenere in considerazione per strutturare correttamente il server sono:

- Quanti utenti si presuppone richiederanno contemporaneamente la trasmissione di video;
- La qualità audio e video che si intende fornire;
- Quanti titoli (files) si intende memorizzare e quanto spazio su disco è richiesto per ogni video;
- La disponibilità che il sistema deve avere nel tempo (24 ore al giorno, 7 giorni su 7);

Gli utenti

Il nostro progetto prevede un picco di utenza di 14000 richieste contemporanee da gestire attraverso 4 server, quindi una media di 3500 richieste per ogni server.

Ogni uscita fast Ethernet ha una banda nominale di 100Mbps, tuttavia il throughput reale è sostanzialmente inferiore, intorno ai 50-60 Mbps. Per calcolare il

numero di video stream concorrenti che una interfaccia può effettivamente sostenere si usa la seguente formula:

$$\frac{\textit{Throughput dell'interfaccia}}{\textit{Bitrate del video stream}} = \textit{video stream concorrenti}$$

Nel nostro caso, usando un bitrate di 3 Mbps, per ogni uscita Fast Ethernat quindi si ha:

$$\frac{60\ \text{Mbps}}{3\text{Mbps/ stream}} = 20\ \text{stream}$$

su ogni modulo ESM ci sono 4 uscite, otteniamo 80 stream che vengono trasmessi da ogni modulo, ci servono dunque 44 schede ESM per server.

La capacità di calcolo

Il sistema da noi considerato è fortemente I/O_bound, più processori contiene il server e maggiore è la potenza del sistema.

Le video pump, i processi responsabili di inviare i video dal sever agli utenti, possono gestire un gran numero di stream concorrenti, quando operano su server con un alto numero di processori.

Purché si lasci sempre uno dei processori dedicato alle operazioni di gestione del sistema, ad ognuno dei restanti può essere assegnata una istanza di video pump.

Noi abbiamo 88 processori disponibili (2 per ogni modulo ESM), si è calcolato che per ogni 7 assegnati a video pump ne viene lasciato uno per gli altri processi dell'OVS ,abbiamo quindi 77 istanze di video pump, ognuna di esse riesce a gestire circa 36 stream concorrenti (36 x 77 = 2772 streams), abbiamo bisogno di potenza di calcolo aggiuntiva per arrivare a 3500 stream, precisamente ci servono circa 20 video pump in più e 3 processori di servizio, per questo vengono aggiunte 12 schede MCM le quali incorporano 2 processori ciascuna.

La gestione dello storage

Poiché il contenuto dei video deve essere codificato e compresso prima di essere disponibile alla trasmissione, la quantità di spazio richiesto per memorizzare il pacchetto di film, dipende dal tipo di codifica utilizzato secondo la formula:

$$\textit{storage requirement (MB)} = \frac{\textit{total playlenght (seconds) x bitrate(Mbps)}}{\textit{8 bit/byte}}$$

Nel nostro allestimento. sono previsti 1000 film disponibili, memorizzati in parte direttamente sui dischi e in parte su dispositivi di memorizzazione esterni (DVD juke-box), per un totale di circa 2000 ore di video.

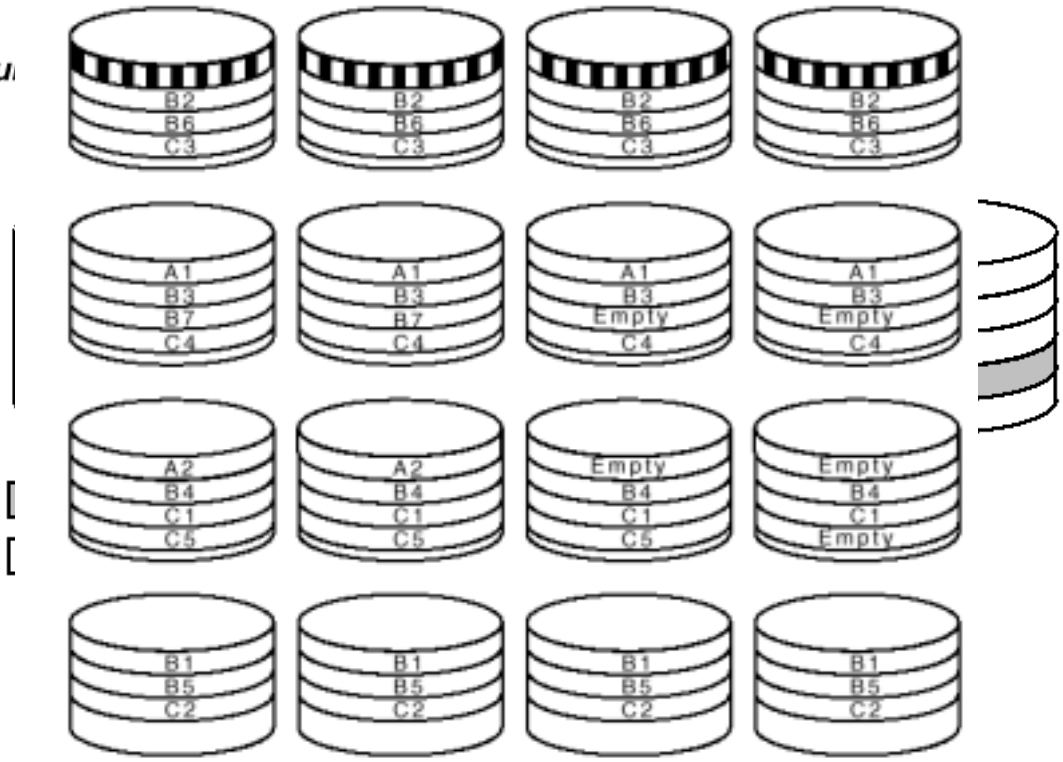
Inoltre bisogna tenere conto che per ogni film viene creato un tag file per memorizzare le informazioni necessarie al sistema per la gestione delle operazioni di localizzazione dei video, questo comporta un aumento di circa il 10% dello spazio richiesto, otteniamo quindi:

$$\frac{2000 \times (3600 \text{ sec}) \times (3\text{Mbps})}{8 \text{ bit /byte}} \times 1.1 = 2900 \text{ GB totali}$$

Noi abbiamo calcolato di tenere solo 500 film direttamente accessibili da HD e di memorizzare i rimanenti nei DVD juke-box o nelle unità a nastro, abbiamo perciò 1450 GB di dati.

Per garantire protezione dagli errori viene utilizzata la tecnologia RAID con parità, questa migliora sia la fault tolerance che le performance: i dati anziché essere memorizzati in maniera contigua sui dischi vengono suddivisi in strisce (stripes) queste sono memorizzate in maniera continua una per ogni disco del RAID SET: il primo blocco è memorizzato sul primo disco, il secondo blocco sul secondo disco e così via, questa modalità permette l’accesso in parallelo alla varie porzioni di file e riduce i tempi per il recupero dei dati.

Figure 2–6 MDS Volume Showing Space Allocations



= table of contents
A1 = first RAID stripe of file A
A2 = second RAID stripe of file A
B1 = first RAID stripe of file B
Empty = unused portion of allocated RAID stripe

Per garantire protezione dagli errori si salvano informazioni di parità, per ogni blocco si memorizzano anche dati di parità che permettono di ricostruire i dati in caso di fallimento di un disco all'interno del RAID SET.

La dimensione del RAID nel nostro caso è di cinque dischi, per la parità viene usato circa il 20% dello spazio disponibile, quindi lo spazio su disco necessario per i dati e la parità diventa:

$$\textit{Disk space} = \frac{\textit{content x RAID size}}{\textit{RAID size} - 1}$$

$$\frac{1450\ \text{GB} \times 5}{4} = 1812{,}2\ \text{GB}$$ spazio totale compreso l'overhead del RAID

Per eseguire la correzione di dati è necessario che ci siano dei dischi di riserva su cui copiare le informazioni man mano che vengono ricostruite, abbiamo quindi deciso di assegnare un disco aggiuntivo per ogni gruppo da 5 (la conformazione a RAID 5 permette la ricostruzione dei dati di un solo disco per ogni RAID SET). Nel nostro caso abbiamo a disposizione 88 bus SCSI, con gruppi da 5 dischi ne utilizziamo solo 85 e otteniamo 17 gruppi, ci bastano quindi 17 dischi di supplemento; noi abbiamo deciso di utilizzare dischi da 9 GB, il numero totale di dischi richiesti è perciò:

$$Dischi = \frac{spazio\ totale\ richiesto}{Dim\ di\ 1\ disco} + dischi\ di\ riserva$$

$$\frac{1812{,}2\ GB}{9\ GB} + 17 = 222 \text{ dischi totali}$$

A questo punto si deve decidere la dimensione del blocco di dati (stripe size) che viene trasferito, questo incide sul throughput dei dischi e sulla quantità di memoria richiesta per il caching dei dati, poiché noi abbiamo a disposizione una banda sufficientemente larga abbiamo scelto una stripe size di 64 Kbyte.

Questo valore influenza anche il numero di file che si possono memorizzare, infatti per memorizzare la “Table of content” di un volume viene riservato un solo stripe su ogni disco del RAID, le informazioni associate ad ogni file occupano 64 byte, quindi il massimo numero di file memorizzabile è dato da:

$$Files = \frac{dim\ dello\ stripe\ (in\ byte)\ x\ (\ RAID\ size-1)}{64\ bytes\ per\ file}$$

$$\frac{64\ KB\ x\ 4}{64\ B} = 4095 \text{ files}$$

Nel nostro caso abbiamo 500 video files e 500 tag files, quindi siamo abbondantemente sotto il limite.

Banda richiesta e banda fornita

E’ necessario controllare che il sistema possa realmente fornire la banda richiesta da tutti gli stream concorrenti programmati.

La capacità effettiva sostenuta dal sistema è data dal minore tra questi due valori:

- La banda totale fornita dai dischi, data dal numero totale dei dischi per la banda di ognuno di essi;
- La banda totale dei bus SCSI, data dal numero di bus per la capacità di ognuno;

La banda richiesta è data dal numero di video stream concorrenti e dalle operazioni di gestione del sistema che aggiungono un overhead di circa il 20%:

banda richiesta = video stream x bitrate x 1.2

3500 video stream x 3 Mbps x 1.2 = 12600 Mbps= 1575 MBps banda richiesta

La banda totale sostenuta dai dischi è uguale al numero di dischi (eccetto gli spares) moltiplicata per la banda di ogni disco, aggiustata di un fattore che tiene conto della protezione RAID:

$$\textit{total disk bandthwith} = \frac{\textit{numero dischi x banda per disco x (RAID size –1)}}{\textit{RAID size}}$$

$$\frac{205 \text{ dischi x } 10 \text{ MBps x } 4}{5} = 1640 \text{ MBps banda sostenuta dai dischi}$$

N.B. i 10 MBps si riferiscono alla banda sostenuta dichiarata di dischi IBM UW SCSI2 DDRS-392130 da 9.1GB.

La banda totale sostenuta dai bus SCSI è uguale al numero di bus per la banda di ciascuno aggiustata di un fattore dovuto alla protezione RAID:

$$\textit{Total bus bandthwith} = \frac{\textit{numero di bus x banda sostenuta x (RAID size –1)}}{\textit{RAID size}}$$

$$\frac{85 \text{ bus x } 24 \text{ MBps x } 4}{5} = 1632 \text{ MBps banda sostenuta dai bus}$$

Come si vede la banda richiesta è inferiore ad entrambi i valori: il nostro sistema risulta correttamente dimensionato.

Supporti per alloggiare i dischi

I dischi vengono sistemati nei cosiddetti disk shelf (DS/SHF-EXP), ognuno di questi è in grado di contenere 6 dischi ed è fornito di alimentazione ridondante.

I segnali per la gestione dei guasti sono processati direttamente dai controller SCSI; i segnali sono inviati ai controller attraverso lo stesso cavo che trasporta i dati da e per i dischi, i dispositivi di fault detection riescono a rilevare guasti nel sistema di alimentazione, alle ventole e ai dischi.

Quando un guasto è rilevato, il software di sistema indica in quale dei disk shelf è avvenuto attraverso uno status LED, questo segnala anche quale tipo di guasto si è verificato.

I disk shelf sono a loro volta alloggiati in appositi vani PSB_20.

Per ognuno dei nostri server sono necessari 37 DS/SHF_EXP ,sistemati in due PSB-20.

Sistema operativo Transit

Transit è il sistema operativo che gira su ogni nodo del server Media Cube. E' un light-weight microkernel derivato dallo Unix Plan 9 dell'AT&T.

Le sue caratteristiche sono:

- Software per la comunicazione a ipercubo
- Algoritmi ottimizzati per l'hardware in questione
- Ottimizzato per l'Oracle Video Server
- Alte prestazioni e affidabilità.

Transit distribuisce il carico di lavoro tra le cpu interconnesse e i moduli di I/O, fornisce un'interfaccia ai dispositivi hardware, gestisce i processi dell'OVS su ogni nodo; inoltre mette a disposizione un file system u9fs che crea una particolare gerarchia sulla system console rendendola disponibile ai processi.

La system console

La console scelta è una workstation Sun, con sistema operativo Solaris, e:

- CPU Sun SPARC;
- Ram : minimo 64 MB;
- Dischi: 2 GB occupati dal sistema operativo, circa 600 MB per l'Oracle Video Server e 64 MB di swap space.

La set-top-box

Al lato client abbiamo scelto di utilizzare le set- top-box prodotte dalla Stellar One Corporation, indicate anche dalla Oracle come compatibili con il loro software.

Le Netris 3000 Interactive Broadband Set-Top Terminals sono in grado di fornire servizi video e audio di qualità digitale, come il video-on-demand: 25 frames per secondo di video full_motion e full_screen combinato con suono stereo.

La set-top-box è dotata delle seguenti funzionalità per un sistema VoD:

- apparato di rete, hardware con relativo software;
- video /audio decoder MPEG-1/2;
- scheda video per la generazione in overlay di testo e grafica;

Il sistema operativo ed il software client dell'OVS system risiedono nella memoria non volatile del sistema, ogni aggiornamento che si rendesse necessario dopo la prima installazione verrebbe scaricato ed eseguito sfruttando la rete sottostante; l'intera procedura è completata in modo trasparente all'utente, si riducono così i costi dell'assistenza tecnica a domicilio.

Periodicamente il software di diagnostica esegue un check up del sistema e comunica eventuali anomalie e malfunzionamenti al servizio di assistenza tecnica e contemporaneamente visualizza sul video dell'utente messaggi che descrivono i problemi rilevati; questi controlli possono essere eseguiti anche su comando esplicito inviato dal centro di erogazione dei servizi.

Specifiche Tecniche

Base Platform

- CPU della classe Pentium Intel con bus PCI
- 32 MB di RAM espandibili fino a 128 MB

- memoria non volatile: Hard disk EIDE
- Processore grafico dedicato 32 bit colore e risoluzione grafica fino a 1600 x 1200 punti per pollice
- segnale audio a 16 bit elaborato da un microprocessore DSP.

Network interface

E' stata creata una serie di modelli per adattarsi alle più diverse esigenze ,quello da noi scelto è il NETRIS 3020 che ha un'interfaccia Ethernet 10/100BaseT.

MPEG Video Decoding

- MPEG-1 e MPEG-2

Eventualmente è disponibile anche un'altra set-top-box più economica prodotta dalla Acorn. Anche in questo caso la compatibilità con l'OVS è garantita dalla stessa Oracle. Le funzionalità sono simili a quelle fornite dalla macchina della Stellar One ma i costi sono ridotti da alcune differenze sul piano tecnico, utilizzo di memoria ROM al posto di HD eide, uso di processori più economici.

Conclusioni

Complessivamente ognuno dei quattro server nCube 3000 da noi utilizzato risulta così configurato:

- 111 processori nCUBE (proprietari);
- 7 GB di memoria principale SDRAM;
- 176 porte Ethernet 4x44 Eternet Server Module (proprietarie);
- 222 dischi IBM UW SCSI2 DDRS-392130 da 9.1GB;
- 88 interfacce Ultra Wide SCSI (proprietarie);
- 1 PCI Module

A ciascuno dei server è collegata una workstation Sun utilizzata come system console.

Stima dei costi.

Premessa

A causa della totale mancanza di collaborazione da parte delle aziende produttrici di hardware e software, non ci è stato possibile reperire la maggior parte dei prezzi

delle attrezzature; abbiamo così stimato i costi in base al prezzo medio di mercato delle singole componenti.

Ncube 3000

- Processori nCube3: L. 1.500.000 cad x 111 = L. 1.665.000.000;
- Porte Fast Ethernet : L. 100.000 cad x 176 = L. 17.600.000;
- Controller UW SCSI : L. 400.000 cad x 88 = L. 35.200.000;
- Memoria SDRAM : L. 13.125.000 per 7 GB (blocchi da 64 MB);
- Dischi IBM UW SCSI 2 DDRS-392130 da 9.1GB:
  L. 500.000 x 222 = L. 111.000.000;
- Racks, alimentazione, contenitori esterni dischi, etc : L. 400.000.000;

Stima costo totale dei quattro server: L. 2.241.925.000 x 4 = L. 8.967.700.000.

Workstation

Quattro Workstation Sun Microsystem UltraSparc IIi L. 60.000.000

Set_top_box

Dalle L. 300.000 a L. 500.000.

Memorie di massa a basso costo (DVD/tape)

- DVD Recorder L. 1.430.000 cad x 4 = L. 5.720.000;
- Registratori DAT RAM External 8 GB L. 1.350.000 x 4 = L. 5.400.000;
- DVD jukebox Pioneer DRM-5004X 500-Disc (4 lettori 32x) :
  L. 27.000.000 x 4 = L. 108.000.000

Schede di compressione MPEG2

Circa L. 6.000.000 x 4 = L. 24.000.000

Software

Quattro licenze Oracle Video Server L. 20.000.000 x 4 = L. 80.000.000

Quattro licenze Transit L. 5.000.000 x 4 = L. 20.000.000

Quattro licenze Oracle8 L. 4.000.000 x 4 = L. 16.000.000