

## Audio e video

### Introduzione

L'uso di pagine *Web* che includono *media* continui, come audio e video, è ormai molto frequente ed è diffuso in numerose tipologie di servizi e attività che attraverso il *Web* si realizzano o si completano. A titolo di esempio si possono citare i siti di testate d'informazione televisive, che tipicamente offrono la possibilità di fruire dei notiziari attraverso sistemi di *video on demand*, o i sistemi di *e-learning* che offrono materiale didattico ottenuto preregistrando lezioni in aula.

Entrambi questi *media* sono di tipo continuo, ovvero sono definiti attraverso valori che cambiano nel tempo. La quantità di informazioni necessarie per rappresentare un *media* continuo cresce in funzione del tempo e per questo motivo sia l'audio che il video sono gestiti in molti casi come flussi di informazioni, piuttosto che come *file*. Questa meccanica di trasmissione, nota come *streaming*, è molto efficace nel limitare l'attesa dovuta allo scaricamento completo dei *file*. Non è però da sola sufficiente a rendere audio e video fruibili attraverso il *Web* (o più in generale attraverso *Internet*). Allo *streaming* devono infatti essere affiancati meccanismi di compressione molto efficaci che consentano di ridurre sensibilmente la larghezza di banda richiesta per la trasmissione dei flussi.

La contemporanea realizzazione di entrambe queste meccaniche, disponibile ormai attraverso l'uso di diverse piattaforme *software*, ha reso l'inclusione di oggetti continui nelle pagine *Web*, una attività relativamente semplice da realizzare e che offre la possibilità di arricchire in modo significativo servizi e contenuti offerti dai singoli siti. In presenza di elementi multimediali occorre nuovamente sottolineare l'importanza del rispetto delle principali regole di accessibilità del *Web*.

### Audio

I suoni sono vibrazioni meccaniche che si propagano attraverso l'etere sotto forma di un'onda longitudinale di compressioni e rarefazioni dell'aria. L'audio in forma analogica è dunque un'onda e la sua riproduzione in forma digitale avviene tipicamente attraverso una specifica componente del PC, la scheda sonora. La stessa scheda sonora si occupa poi della riproduzione dell'audio digitale che il PC gestisce.

La scheda sonora è dunque in realtà un dispositivo di *input/output*:

- In *input*, collegata ad un microfono, campiona i suoni dall'esterno trasformando il segnale audio analogico esterno in uno digitale che arriva al PC. In questo caso funge da *Analogue to Digital Converter* (ADC).
- In *output*, collegata alle casse o alla cuffia, trasforma il segnale audio digitale del PC in uno analogico udibile all'esterno. In questo caso funge da *Digital to Analogue Converter* (DAC).

Così come per le immagini, anche per il suono esistono sistemi di creazione di audio di sintesi. Per esempio un suono digitale di sintesi si può ottenere interfacciando una tastiera elettronica al *computer*.

L'audio è un *media* continuo e dunque la digitalizzazione avviene sulla base di:

- campionamento: il segnale è misurato a intervalli discreti e il numero di campioni in un intervallo di tempo è detto *sampling rate*;
- quantizzazione: i valori possibili sono fissati in livelli di quantizzazione tipicamente a intervalli regolari.

### Digitalizzazione dell'audio

La digitalizzazione del suono viene definita da due parametri, la frequenza di campionamento e il

sistema di quantizzazione.

Il campionamento è determinante poiché il suono è un segnale continuo e in particolare il formato analogico è un'onda. Le onde sonore si distinguono per frequenza in:

- Infrasuoni da 0 a 20 Hz.
- Udibile (da orecchio umano): da 20 Hz a 22 KHz.
- Ultrasuoni: da 22 KHz a 1 GHz.
- Ipersuoni da 1 GHz a 10 THz.

Il parlato copre un sottoinsieme specifico dell'udibile, quello che va (circa) da 600 Hz a 5 KHz. Utilizzando la frequenza di *Nyquist*, introdotta in precedenza, si possono facilmente calcolare le frequenze di campionamento corrispondenti all'udibile e al parlato che sono di 44.000 Herz (ovvero 44 kHz) per l'udibile e di 4.000 Herz (ovvero 4 kHz) per il parlato. Campionare a 44 KHz significa misurare il segnale 44.000 volte ogni secondo.

La discretizzazione avviene rappresentando ogni campione con un valore numerico. Maggiore è il numero di valori possibili e maggiore è il numero di *bit* necessari per rappresentarlo. Nella digitalizzazione del suono il numero tipico di livelli di quantizzazione è:

- 256 valori, per rappresentare i quali sono necessari 8 *bit*. La qualità non è massima e questo tipo di scelta viene fatta o per rappresentare la voce umana o per trasmettere il suono attraverso la rete.
- 65536 valori, per rappresentare i quali sono necessari 16 *bit*. La qualità è quella massima percepibile ed è nota come qualità CD, perché è quella utilizzata dai CD audio.

## Dimensioni

Il suono è un *media* continuo e la misura della dimensione del flusso audio digitale è data in bps (*bit* per secondo). In questo modo la dimensione dei corrispondenti *file* può essere fatta moltiplicando il valore in bps per la durata del suono. Considerando la frequenza di campionamento e la discretizzazione possiamo calcolare la dimensione del flusso per poi verificare se esistono le condizioni per trasportare l'audio non compresso attraverso la rete.

Con la voce un primo calcolo approssimativo si può fare nel modo seguente: le frequenze sono in un intervallo di circa 4 kHz e raddoppiandole (per *Nyquist*) si ottiene un campionamento a 8 kHz. Consideriamo un *byte* di codifica per ogni campione (8 *bit*), la codifica del flusso è pari a  $8 \times 8 = 64$  Kbps.

Con la musica (stereo ad alta fedeltà) il calcolo va invece fatto nel modo seguente: le frequenze sono in un intervallo di 22.05 kHz e raddoppiandole (per *Nyquist*) si ottiene un campionamento a 44.1 kHz. Consideriamo due *byte* di codifica per ogni campione (16 *bit*) e due canali (per avere l'effetto stereo), la codifica del flusso è pari a  $1.411.200 \text{ bps} = 176.400 \text{ Bps}$

Con la qualità stereo e alte fedeltà il risultato è che un brano di 5 minuti occupa circa 53 *Gbyte* e questo giustifica il fatto che nei CD audio siano contenuti circa 70 minuti di musica.

Il *bandwidth* richiesto è, per la trasmissione in rete, irrealistico. Dovremo quindi introdurre formati compressi per l'audio e meccaniche di trasporto che consentano di fruire di audio attraverso *Internet*.

## MP3

Una famiglia di sistemi di compressione (*lossy* e *source encoding*) di particolare successo è quello proposto da MPEG (*Moving Pictures Experts Group*) che si occupa di sviluppare e standardizzare

algoritmi di compressione audio e video. Il lavoro del comitato MPG è diviso in fasi indicate da numeri arabi (1,2,4,7). Il nome **MPEG** Audio indica gli algoritmi di compressione audio, che si distinguono in tre livelli (*Layer* I, II e III) di crescente complessità e *performance*. Il *layer* III è comunemente detto **MP3**. Quindi **MP3** indica il terzo *layer* di codifica della parte audio di **MPEG-1** ed **MPEG-2**.

Lo *standard* **MP3** è meglio identificato dalla sigla **ISO-MPEG** Audio *Layer-3* (IS 11172-3 e IS 13818-3).

**MP3** è un algoritmo di compressione con perdite: dopo la decodifica non si ottiene il *file* di origine, ma un *file* che viene percepito in modo analogo dall'utente. L'algoritmo alla base di **MP3** è dunque di tipo percettivo e ha lo scopo di produrre un segnale che suoni come l'originale per un ascoltatore umano, pur essendo differente come forma d'onda.

**MP3** consente di scegliere il livello di compressione da applicare all'audio e in particolare codifica tre livelli:

- bassa compressione, alta qualità, con flusso a tra 192 e 300 Kbps;
- media compressione, media qualità, con flusso a 128 Kbps (è il livello più usato);
- alta compressione, bassa qualità, con flusso fino a 16 Kbps.

Il confronto con 1.411.200 bps dell'audio digitale in formato non compresso fa emergere evidentemente le motivazioni che sono alla base del successo della compressione **MP3**. In particolare la codifica a media compressione viene utilizzata come *standard* nello scambio di *file* attraverso la rete e consente, tra l'altro di memorizzare quasi 100 brani musicali su un CD, sullo stesso tipo di supporto che col formato CD audio contiene 72 minuti di musica.

## Video

Con video si intende un sistema di riproduzione di immagini in movimento. L'occhio percepisce una sequenza sufficientemente veloce di immagini statiche come un movimento continuo a causa di un fenomeno di persistenza delle immagini sulla retina (noto come **Persistence of Vision**, POV). Il movimento del video è dunque una illusione ottica in cui il sistema riproduce una sequenza di immagini statiche (comunemente chiamate fotogrammi) e la velocità con cui le immagini si susseguono è maggiore di quella che l'occhio umano è in grado di seguire (circa una nuova immagine ogni decimo di secondo).

Nel digitalizzare il video si ripropongono tutte le tematiche relative alle immagini (risoluzione spaziale e cromatica), assommate a quelle dell'audio. Inoltre occorre considerare che le sequenze di immagini vengono prodotte a intervalli di tempo regolari e che si deve digitalizzare un numero di immagini al secondo sufficiente per ingannare l'occhio umano. La velocità con cui i fotogrammi si susseguono è misurata in *frame* al secondo, o *frame per second* o fps (in realtà è una frequenza per cui si misura anche in Hz). Se la velocità non è sufficiente si possono avere due fenomeni:

- Movimento non fluido: c'è percezione del movimento ma con scarsa continuità.
- Video lento: non c'è percezione del movimento ma dei singoli fotogrammi.

Misure tipiche per il *frame rate* sono le seguenti: teleconferenza 10 fps (video lento, movimento a scatti), film muto 16 fps (movimento non fluido), film sonoro 24 fps, televisione 25-30 fps, televisione ad alta definizione 50-60 fps.

Nelle tecnologie di riproduzione video a tubo catodico, il *refresh* sequenziale (riga per riga, dall'alto verso il basso, da sinistra verso destra) delle immagini, può produrre sfarfallio. Per evitarlo, si può interlacciare la riproduzione video, visualizzando le righe in alternanza, prima le pari, poi le dispari.

## Digitalizzazione del video

Il video digitale può essere visto come una risorsa a due dimensioni:

- i *frame*, ciascuno dei quali deve essere codificato come una immagine *bitmap*, ovvero come una griglia di punti  $n \times m$ .
- Il tempo (che scandisce la sequenza dei *frame*).

Il *file* video deve dunque codificare tante immagini, un certo numero per secondo, per quanti secondi dura il video. Affiancato al video è solitamente codificato anche il canale audio che viene registrato in parallelo.

Il sistema colorimetrico tipico del video è quello **YUV Luminance-Chrominance** che permette di sfruttare il fatto che l'occhio umano è più sensibile alla variazione della luce che a quella del colore (ci sono più coni, che percepiscono la luce che non bastoncelli, che percepiscono il colore).

Per meglio comprendere le dimensioni tipiche di un flusso video, proviamo a fare i calcoli in un caso specifico: consideriamo i fotogrammi con dimensioni in punti: 640 x 480 e profondità del colore 3 *byte* (*true color*) ovvero 24 *bit*. Supponiamo di usare un *frame rate* di 30 fotogrammi al secondo per un filmato di 2 minuti (ovvero 120 secondi). Il flusso risultante è pari a 640 X 480 X 3 X 30 Bps ovvero a circa 27 *MegaBytes* al secondo. Il *file* di due minuti occupa complessivamente circa 3.3 *GigaBytes*

## MPEG

La necessità di utilizzare tecniche di compressione è in questo caso ancora più evidente che nel caso dell'audio. I meccanismi di compressione più utilizzati per il video sono tutti basati sulla codifica **MPEG Video**, definito dal *Moving Picture Experts Group*, lo stesso ente che sviluppa **MP3**.

La prima versione di **MPEG Video**, **MPEG 1**, era stata studiata per produrre un flusso compresso di circa 1.5 Mbit/s in modo da poter memorizzare su un CD-ROM (la cui capacità è di 638 *MBytes*) circa 56 minuti di filmato audio-video. L'obiettivo di **MPEG 1** era quello di ottenere una qualità del filmato paragonabile a quella di un videoregistratore VHS.

Le versioni attualmente in uso di **MPEG** (**MPEG 2**, utilizzato nella produzione dei DVD Video e **MPEG 4**, che viene sfruttato anche dai sistemi DiVx), ottengono flussi che sono:

- a parità di capacità richiesta, di maggiore qualità;
- a parità di qualità, con minore capacità richiesta.

Tutte le versioni di **MPEG** (inclusa la 1) utilizzano meccanismi percettivi per comprimere ed in particolare usano uno spazio dei colori **YUV** in cui l'immagine di luminanza è codificata per intero, mentre per la crominanza è usato un insieme ridotto di valori.

La compressione opera su entrambe le dimensioni del video rimuovendo:

- La **Ridondanza spaziale**: ogni singolo fotogramma può essere considerato come una immagine a qualità fotografica e viene quindi compresso utilizzando **JPEG**.
- La **Ridondanza temporale**: i fotogrammi successivi sono spesso molto simili e si differenziano per particolari ridotti e circoscritti. Questa condizione è determinata dalla stessa meccanica con cui è costruito il video (quella del *framing* e della POV). Possono quindi essere individuate meccaniche di compressione che sfruttano le relazioni tra i diversi fotogrammi, per rimuovere informazioni ridondanti.

Entrambe le fasi coinvolgono algoritmi *lossy*, di tipo *source encoding* e basati su considerazioni di tipo percettivo. Ulteriori informazioni sull'argomento sono reperibili attraverso la **Bibliografia**.

## Streaming

L'audio e il video, in quanto *media* continui, producono *file* la cui dimensione è proporzionale alla durata del *media*. Nonostante i criteri di compressione introdotti consentano di ridurre sensibilmente la larghezza di banda necessaria alla trasmissione di audio e di video, esistono molte applicazioni che non richiedono o, a volte, non supportano lo scaricamento completo di un *file* e operano invece sul flusso di dati.

In molti casi l'attesa per un *download* completo è insostenibile:

- A causa della dimensione del *file* e dell'attesa che deriverebbe dallo scaricamento: per esempio, scaricare di un intero film (tipicamente, di notevoli dimensioni in *byte*) può richiedere ore.
- A causa di requisiti di tempo reale o di *liveness* (trasmissione dal vivo): per esempio, *Internet-Radio* e *Internet-TV* vogliono raggiungere prima possibile i propri ascoltatori e/o telespettatori.

In questi casi si utilizza un sistema di trasmissione basato sui flussi, che viene chiamato *streaming*. Il meccanismo di base dello *streaming* si può riassumere nel seguente modo:

1. L'*host* sorgente inizia la trasmissione.
2. L'*host* destinazione riceve il primo pacchetto dalla sorgente e aspetta un certo periodo durante il quale accumula pacchetti che arrivano a destinazione.
3. Terminato il periodo d'attesa, l'*host* destinazione comincia il *playout* dei pacchetti accumulati.
4. La trasmissione continua con le modalità descritte sino al termine.

La dimensione del *buffer*, e quindi il risultante tempo di attesa per il riempimento, sono tipicamente stimati dall'applicazione *client* al momento dell'inizio della trasmissione. In realtà *Internet* non può in generale garantire condizioni stabili, per cui la larghezza di banda disponibile può, rispetto a questo valore iniziale, aumentare o diminuire (ovvero le prestazioni della rete possono migliorare o peggiorare). Se la larghezza di banda si riduce il *buffer* non è sufficiente a compensare la riduzione del flusso e può accadere che la trasmissione si interrompa. Tipicamente l'applicazione *client* ricomincia a bufferizzare (come se fosse al punto due della sequenza sopra descritta). L'utente percepisce discontinuità della comunicazione e il tempo di attesa. Se invece la larghezza di banda aumenta, non ci sono effetti immediatamente percepibili e la percezione dell'utente sarà di un flusso continuo audio e/o video.

Su questo argomento è disponibile un **approfondimento**.

## Piattaforme di streaming

Nel caso del *Web* l'*host* sorgente è un *server* di *streaming* (non necessariamente collocato sulla stessa macchina che fa da *server* HTTP) e l'*host* destinazione è il *client* di *streaming*, che si integra tipicamente col *browser*. La richiesta avviene tramite il *Web*, ovvero come richiesta HTTP da un *browser* ad un *server* HTTP, che avvia la comunicazione di *streaming* tra *client* e *server* nel modo seguente:

- Il *Web browser* richiede una risorsa che in realtà è un meta *file*, ovvero un *file* che descrive il *media* di cui verrà fatto lo *streaming*.
- Il *Server* HTTP risponde inviando il meta *file*.
- Il *Browser* lancia l'appropriato *Player* e gli passa il meta *file*.
- Il *Player*, seguendo le indicazioni del meta *file*, avvia lo *streaming*.

*Player* e *server* di *streaming* concordano le modalità di trasmissione e a tutt'oggi non esiste un protocollo che sia interoperabile, per cui per fruire dello *streaming* offerto da un certo *server* occorre dotarsi del *client* opportuno offerto di solito gratuitamente dal produttore dei *software server*. I più diffusi sistemi di *streaming* sono:

- *Apple Quicktime*: *QuickTime* è un marchio *Apple* utilizzato per una *suite* di prodotti che include un formato multimediale, diversi *player* multimediali con il supporto per lo *streaming* e alcune piattaforme *server*, di cui una (*Darvin Streaming Server*) *open source*. Maggiori informazioni sulla *suite Quicktime* sono reperibili attraverso la **Bibliografia**.
- *Microsoft Windows Media Technologies*: Tecnologie *Microsoft* per la fruizione di multimedia, sia in *streaming* che non. Sono fornite con il sistema operativo, rispettivamente *client* per il *player* e *server* per i servizi di *streaming*. Maggiori informazioni sui prodotti *Microsoft Media* sono reperibili attraverso la **Bibliografia**.
- *RealNetworks*: produce numerose soluzioni per produrre e distribuire multimedia. In particolare esistono diverse versioni del *player* e del *server*, tra cui anche una ad-hoc per le piattaforme mobili. Maggiori informazioni sulle soluzioni *Real* sono reperibili attraverso la **Bibliografia**.

Tutte le piattaforme supportano *streaming* audio e video con meccanismi che consentono adattività alla larghezza di banda disponibile. *Real* e *Quicktime* sono multiplatforma, *Windows Media* è per *Windows*.

## Multimedia continui nel Web

Gli elementi multimediali continui (audio e video) vengono inseriti nella pagina *Web* attraverso il *tag OBJECT* introdotto già nel trattare l'inserimento di immagini in *HTML*.

Lo schema generale d'uso di *OBJECT* è il seguente:

```
<OBJECT data=URLRisorsaMultimediale type=tipoMIME>
---- alternativa al playout ----
</OBJECT>
```

Gli attributi di *OBJECT* (tutti facoltativi) correlati all'inserimento di *media* continui sono:

- **TITLE**: contiene informazioni per la consultazione relative all'oggetto su cui è definito.
- **HEIGHT**: ridefinisce l'altezza dell'oggetto.
- **WIDTH**: ridefinisce la larghezza dell'oggetto.
- **TYPE**: tipo di contenuto *MIME* dell'oggetto. Ovviamente il riferimento al tipo deve essere fatto in accordo con le politiche specifiche di ciascuna delle piattaforme *server*.
- **DATA**: specifica la sorgente di dati ovvero l'*URL* in cui è memorizzato l'oggetto multimediale. Ovviamente il riferimento al *file* o al *metafile* deve essere fatto in accordo con le politiche specifiche di ciascuna delle piattaforme *server*.

La porzione di codice che segue inserisce a titolo di esempio un filmato **MPEG**.

```
<OBJECT title="Introduzione alla programmazione a oggetti" data="lezione1.mpeg"
type="application/mpeg">
La slide mostra, dal basso verso l'alto, i tre livelli: (1) <span lang="en"gt;
hardware </span>, (2) sistema operativo e (3) applicazioni.
</OBJECT>
```

Il filmato incluso nella pagina mostra una porzione di lezione sulla programmazione ad oggetti ed in particolare il docente sta illustrando una *slide* in cui è incorporata una figura. Il testo alternativo a commento, si riduce in questo caso al commento della *slide* ed è inserito tra *<OBJECT>* e *</OBJECT>*.

OBJECT offre dunque un meccanismo implicito per specificare la rappresentazione di oggetti alternativi e ciascuna dichiarazione incorporata di OBJECT può ricorsivamente definire altri tipi di contenuto alternativi. Se un *browser* non è in grado di riprodurre l'OBJECT più esterno perché non ha il *plugin* necessario, allora tenta di riprodurre i contenuti, che a loro volta possono essere costituiti da un altro elemento OBJECT.

Per esempio possiamo utilizzare una definizione di OBJECT in cascata per offrire più contenuti alternativi al filmato MPEG, nel modo seguente:

```
<!-- Prova il video in formato MPEG -->
<OBJECT title="Introduzione alla programmazione a oggetti" data="lezione1.mpeg"
type="video/mpeg">
<!-- Oppure prova immagine in formato JPEG -->
<OBJECT data="slide1.jpg" type="image/jpeg">
<!-- Altrimenti usa il Text Equivalent -->
La figura mostra, dal basso verso l'alto, i tre livelli: (1) <span lang="en">
hardware </span>, (2) sistema operativo e (3) applicazioni.
</OBJECT>
</OBJECT>
```

Questo frammento di codice induce il *browser* a riprodurre, se può, l'MPEG con la videolezione. Se non supporta il *player* MPEG, può riprodurre soltanto l'immagine della *slide*. Alternativamente se non supporta le immagini (o se è usato attraverso *screen reader*), può fornire all'utente la descrizione testuale alternativa. L'esempio non considera volutamente la traccia audio della lezione, che è invece inserita nella porzione di codice seguente, in modo da:

- costruire una alternativa al video, composta da *slide* e audio insieme.
- Costruire una alternativa testuale alla *slide*.
- Costruire una alternativa testuale all'audio (sottotitoli).

```
<!-- Prova il video in formato MPEG -->
<OBJECT title="Introduzione alla programmazione a oggetti" data="lezione1.mpeg"
type="video/mpeg">
<!-- Oppure prova immagine in formato JPEG e audio associato -->
<OBJECT data="slide1.jpg" type="image/jpeg">
<!-- Altrimenti usa il Text Equivalent -->
La figura mostra, dal basso verso l'alto, i tre livelli: (1) <span lang="en">
hardware </span>, (2) sistema operativo e (3) applicazioni.
</OBJECT>
<OBJECT data="commento1.mp3" type="audio/mp3">
<!-- Altrimenti usa il Text Equivalent come sottotitolo-->
Buongiorno, sono Simone Martini, insegno linguaggi e paradigmi di programmazione
all'Università di Bologna e oggi ci intratteniamo su una lezione che tratterà di
linguaggi orientati agli oggetti ....
</OBJECT>
</OBJECT>
```

Questo tipo di rappresentazione offre naturale supporto all'applicazione di meccaniche che favoriscano l'accessibilità della pagina. Maggiori dettagli sull'elemento OBJECT e sui suoi attributi sono reperibili attraverso la **Bibliografia**.

## Conclusioni

Scopo di questa breve trattazione è stato quello di introdurre i *media* continui, audio e video, cercando di presentare i problemi dimensionali legati alla trasmissione di questo tipo di dati. I *media* continui sono infatti caratterizzati dal produrre *file* che sono proporzionali alla durata in tempo del *media*. A ciò va aggiunto che entrambi i segnali (audio e video) sono complessi da rappresentare e quindi anche in termini di larghezza di banda (ovvero in modo indipendente dal tempo), la

trasmissione di questo tipo di *media* è difficoltosa.

Due sono i principali meccanismi messi in opera per ottenere *media* digitali continui che possono essere inseriti in una pagina *Web*:

- la compressione, che mira a ridurre la richiesta in termini di larghezza di banda;
- lo *streaming*, che ha lo scopo di trattare il *media* continuo come un flusso, eliminando le attese dovute a *download* di interi *file*.

Infine è stato trattato il *tag* OBJECT che consente l'inserimento di *media* continui nella pagina *Web*. Ulteriori informazioni sugli argomenti trattati possono essere reperite attraverso la **Bibliografia**.