

# LEZIONE 9

# CARICAMENTO DI CONTENUTI ESTERNI

## TIPO DI CONTENUTI MULTIMEDIALI

- In Flash posso caricare contenuti multimediali presenti su disco. A seconda dei contenuti utilizzerò oggetti e sistemi di controllo diversi:
  - Filmati Flash
  - Immagini di tipo bitmap
  - File audio
  - Video

# IMMAGINI BITMAP

- Le immagini bitmap vengono caricate utilizzando la classe MovieClip normalmente utilizzando MovieClipLoader per controllare il caricamento
- Flash Player versione 7 è in grado di visualizzare solo immagini in formato jpeg non interallacciate
- Flash Player versione 8 è in grado di visualizzare immagini in formato jpeg interallacciato e non interallacciate, immagini in formato GIF (con eventuale trasparenza) e immagini in formato PNG (con eventuale trasparenza)

# FILMATI FLASH

- I filmati flash vengono caricati utilizzando la classe MovieClip.
- Possono essere ottimizzati:
  - Verificare la qualità del sonoro
  - Verificare la qualità delle immagini bitmap
  - Trasformare gli oggetti vettoriali molto complicati in immagini
- Per default i filmati Flash sono eseguiti progressive downloading, cioè l'esecuzione inizia non appena caricato il primo frame.

## FILE VIDEO

- Oggetti Video e NetConnection
- File in formato **FLV** (**FL**ash **V**ideo)
- Codecs Sorenson Spark (flash 6 e 7) e On2 VP6 (solo flash 8).
- Calcolo del bit rate:
  - $(\text{Bytes} * 8) / \text{durata (in secondi)}$ .

## FILE AUDIO

- Oggetto Sound
- File in formato MP3 cbr (constant bit rate)
- Calcolo del bit rate (vale anche per il video):
  - $(\text{Bytes} * 8) / \text{durata (in secondi)}$ .

# CARICAMENTO ED ESECUZIONE DI UN FILMATO FLASH (SWF)



## IL PLAYER: CARICAMENTO

- Il filmato verrà caricato in una **MovieClip** utilizzando il metodo **MovieClip.loadMovie()**
- Verrà controllato lo stato del caricamento, usando i metodi **MovieClip.getBytesLoaded()** e **MovieClip.getBytesTotal()**.

## IL PLAYER: ESECUZIONE

- Verrà controllato lo stato dell'esecuzione attraverso le proprietà **MovieClip.\_currentframe** e **MovieClip.\_totalframes**.
- Una proprietà **autoplay** stabilirà se l'esecuzione del filmato sarà automatica o partirà a richiesta dell'utente
- Una proprietà **loop** stabilirà se l'esecuzione del filmato verrà ripetuta indefinitamente e si fermerà all'ultimo frame.

## IL PLAYER: LIMITI

- Il filmato verrà eseguito al frame rate del filmato che lo carica e non a quello per cui è stato creato.
- Per semplicità la dimensione del filmato riprodotto verrà assunta come fissa e conosciuta.
- Si assumerà che sulla linea temporale del filmato non ci siano comandi che ne modifichino il flusso d'esecuzione.

## IL PLAYER: COMANDI

- Comando **Play**. Verrà realizzato utilizzando il metodo **MovieClip.Play()**.
- Comando **Pause**. Verrà realizzato utilizzando il metodo **MovieClip.Stop()**.
- Comando **Stop**. Verrà realizzato utilizzando il metodo **MovieClip.GotoAndStop( 1)**.
- L'esecuzione potrà saltare ad un punto arbitrario utilizzando il metodo **MovieClip.GotoAndPlay(...)**.

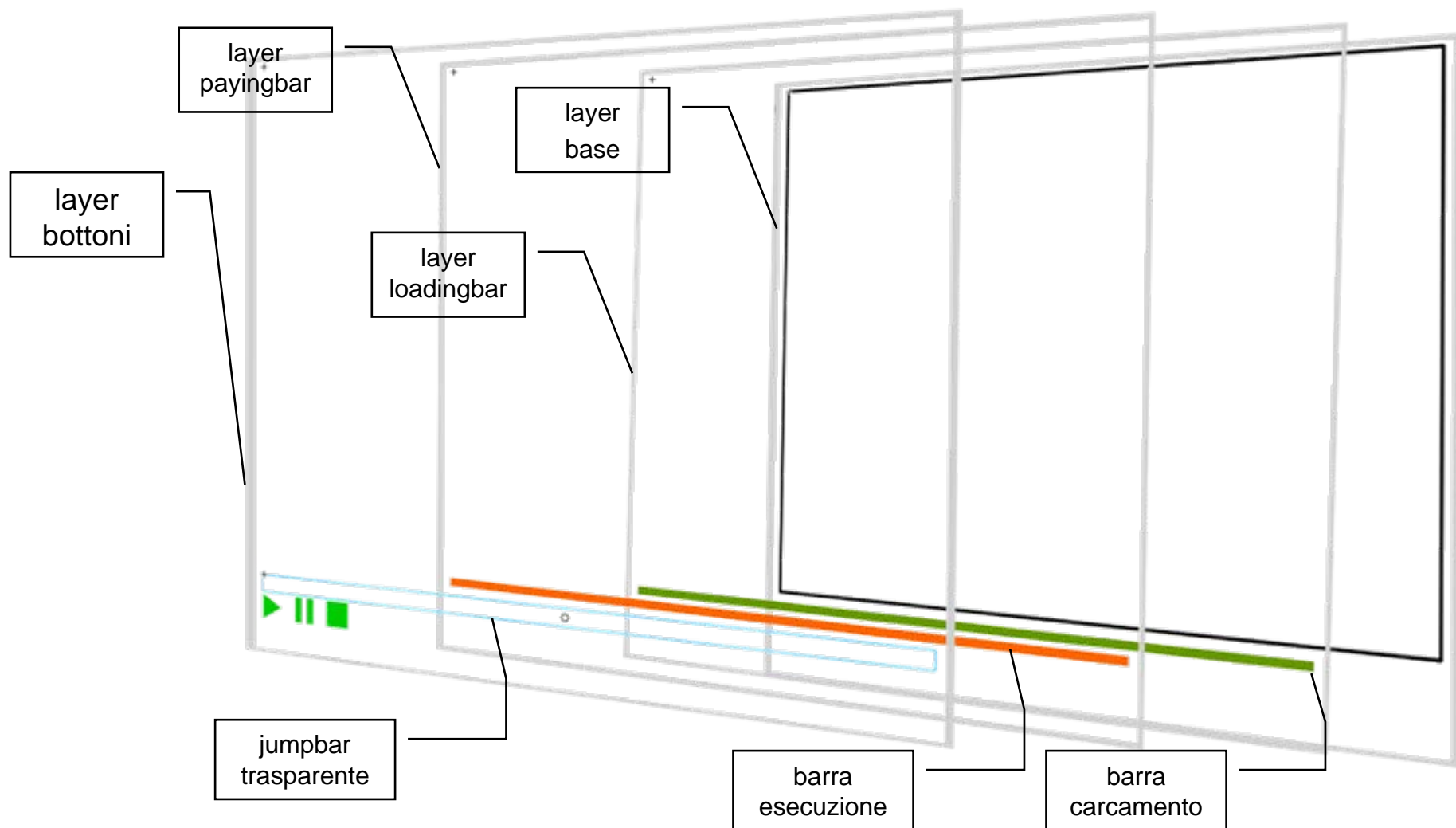
## IL PLAYER: INTERFACCIA

- Un rettangolo indicherà l'area in cui verrà caricato il filmato.
- Una barra indicherà lo stato del caricamento
- Una seconda barra indicherà lo stato di avanzamento dell'esecuzione
- Tre bottoni saranno associati ai comandi **Play**, **Pause** e **Stop**.
- Cliccando sulla barra si potrà saltare ad un punto arbitrario del filmato.

## IL PLAYER: GLI ELEMENTI

- Una MovieClip per caricare il filmato. Verrà creata dinamicamente: ***filmato\_mc***
- Una barra di caricamento (tipo MovieClip): ***loadingbar\_mc***
- Una barra indicatore di esecuzione (tipo MovieClip): ***playingbar\_mc***
- Tre bottoni associati ai comandi **Play**, **Pause** e **Stop**. (tipo Button): ***play\_btn***, ***pause\_btn***, ***stop\_btn***.
- Una barra invisibile per saltare ad un punto arbitrario del filmato: ***iump\_mc*** (tipo MovieClip).

# COSTRUZIONE INTERFACCIA



# IL PLAYER: LE PROPRIETÀ

```
private var filmato_mc:MovieClip; //MovieClip in cui viene
                                   //caricato il filmato
private var loadingbar_mc:MovieClip; //Caricamento
private var playingbar_mc:MovieClip; //Esecuzione

//bottoni
private var jump_mc:MovieClip; //Salto
private var play_btn:Button;
private var stop_btn:Button;
private var pause_btn:Button;

private var init_movie:Boolean;

public var loop:Boolean = false;
public var autostart:Boolean = true;
```



# IL PLAYER: I METODI PRIVATI

```
private function init () {  
    filmato_mc = this.createEmptyMovieClip("filmato_mc",this.getNextHighestDepth());  
    jump_mc.enabled = false;  
    play_btn.enabled = false;  
    stop_btn.enabled = false;  
    pause_btn.enabled = false;  
  
    play_btn.onRelease = function () {  
        this._parent.playClip();  
    }  
    stop_btn.onRelease = function () {  
        this._parent.stopClip();  
    }  
    pause_btn.onRelease = function () {  
        this._parent.pauseClip();  
    }  
    jump_mc.onPress = function () {  
        var my_percent:Number = this._xmouse/this._width;  
        this._parent.jumpTo(my_percent);  
        trace(my_percent);  
    }  
}
```

# IL PLAYER: I METODI PRIVATI

```
private function playClip () {  
    play_btn.enabled = false;  
    stop_btn.enabled = true;  
    pause_btn.enabled = true;  
    jump_mc.enabled = true;  
    filmato_mc.play();  
  
}
```

```
private function stopClip () {  
    play_btn.enabled = true;  
    stop_btn.enabled = false;  
    pause_btn.enabled = false;  
    jump_mc.enabled = true;  
    filmato_mc.gotoAndStop(1);  
  
}
```

# IL PLAYER: I METODI PRIVATI

```
private function pauseClip () {  
    play_btn.enabled = true;  
    stop_btn.enabled = false;  
    pause_btn.enabled = false;  
    jump_mc.enabled = true;  
    filmato_mc.stop();  
}
```

```
private function jumpTo (f:Number) {  
    play_btn.enabled = false;  
    stop_btn.enabled = true;  
    pause_btn.enabled = true;  
    jump_mc.enabled = true;  
    filmato_mc.gotoAndPlay(Math.round(f *  
filmato_mc._totalframes));  
}
```

# IL PLAYER: I METODI PRIVATI

```
public function SWF_player ( ) {  
    init();  
}
```

# IL PLAYER: I METODI PRIVATI

```
public function loadClip (url:String) {
    init_movie = true;
    this.onEnterFrame = function () {
        if (init_movie) {
            if (filmato_mc._framesloaded > 0) {
                init_movie = false;
                if (autostart) {
                    playClip();
                } else {
                    stopClip();
                }
            }
        }
        loadingbar_mc._xscale = (filmato_mc.getBytesLoaded()/filmato_mc.getBytesTotal()) * 100;
        playingbar_mc._xscale = (filmato_mc._currentframe / filmato_mc._totalframes) * 100;
        if (filmato_mc._currentframe == filmato_mc._totalframes) {
            if (loop) {
                filmato_mc.gotoAndPlay(1);
            } else {
                stopClip();
            }
        }
    }
    filmato_mc.loadMovie(url);
}
```