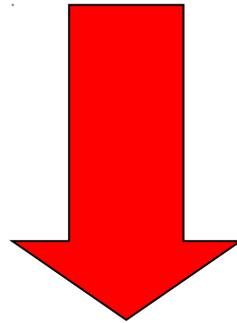


Compressione!!



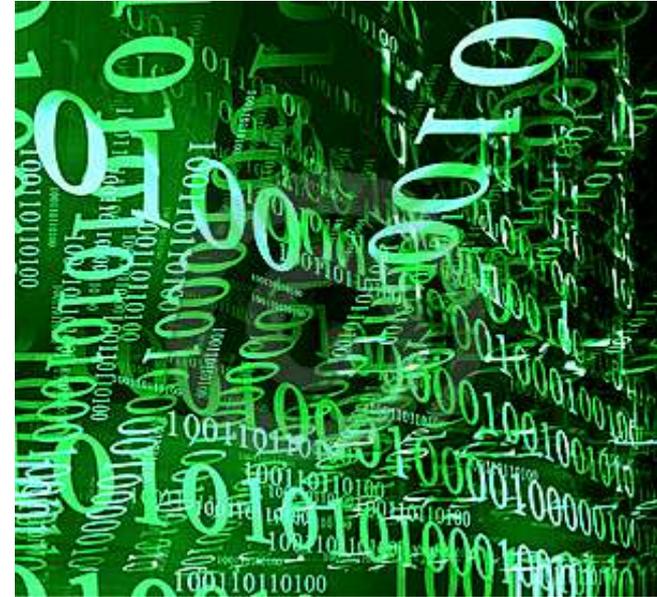
Compressione!!

**Compressione di dati:
testi, immagini, suoni e
video**

B. Cipolla

Compressione: motivazioni

- Ridurre le **dimensioni** (il “peso”) di *oggetti digitali* per
- Risparmiare
 - spazio occupato
 - tempo di trasmissione
- A spese di
 - tempo e risorse per la **compressione** e, ovviamente, la **decompressione**



Se...



- Se esistessero....
 - memorie economiche con capacità immensa e ...
 - reti di trasmissione dati con velocità infinita
- Non ci sarebbe bisogno di comprimere le informazioni
- però...

Ad esempio

- Nel 1985
- con un “enorme” disco rigido da 5 MB,
(pagato l'equivalente di circa 7000 Euro di oggi!)
- si usava “Stacker” (programma di
compressione **lossless**)
- per “aumentare” la capacità del disco
- poi..

Tipi di Compressione

- **Senza** perdita (**lossless**)
- **Con** perdita (lossy)
 - di **informazione**



Compressione senza perdita

Senza Perdita

- **De-**comprimendo si ri-ottiene *e-sat-ta-men-te* l'informazione originale
- Usata per testi, programmi, documenti, **molto raramente** per oggetti multimediali.
- Usata dove la **perdita** di informazioni sarebbe *inaccettabile*

Esempi di compressione *senza* perdita

- Programmi:
 - WinZIP
 - WinRAR
- Compressione **dei dischi (NTFS)** di Windows
- Compressione v.44 (modem)
- Pochi **codec lossless** per video
- **Codec Audio** FLAC
- Immagini (disegni) in GIF
- altri



CO-DEC

Codificatore e
DECodificatore

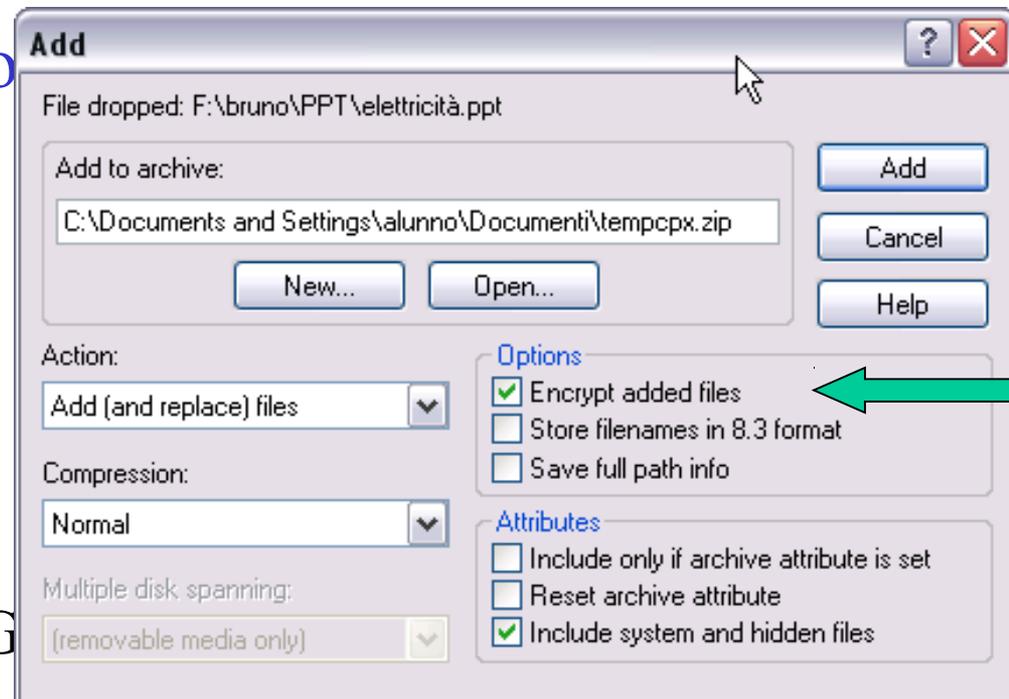
In pratica è un metodo, algoritmo, programma
per comprimere e decomprimere
l'informazione

WinRAR



Winzip e Winrar

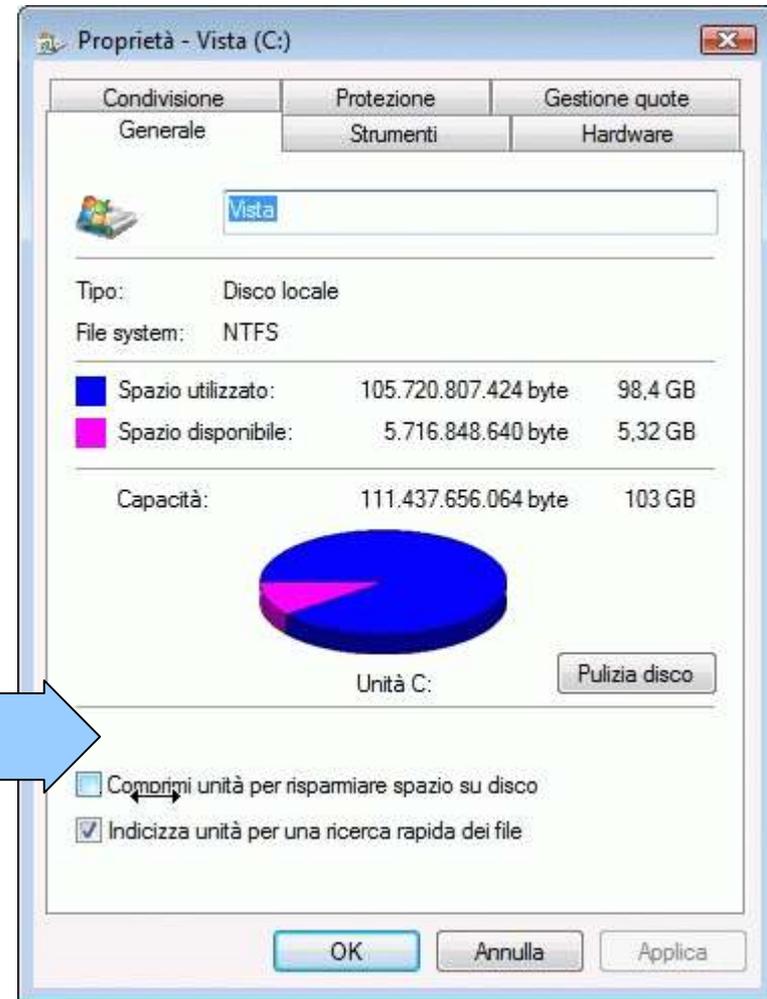
- Shareware
- Comprimono, ma sono usati anche per
 - impaccare più files
 - proteggere con passwo
 - Cifrare
 - ecc.



V1.42: G

Compressione automatica dischi di windows

- Figlia di Stacker e Doublespace
- Solo per **NTFS**!
- Si seleziona nelle proprietà
- Opera automaticamente.
- Meglio acquistare un disco più capiente! (costano poco)



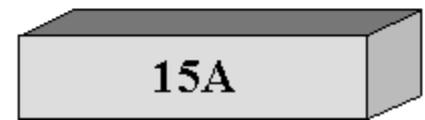
Tecniche di compressione *Lossless*

Run Length Encoding (RLE) (ripetizioni)

- RLE rimpiazza occorrenze consecutive di un simbolo con il **simbolo** seguito dal **numero di volte** che è ripetuto.
- Per esempio, la stringa 111110000003355 potrebbe essere rappresentata da 15063252
- Utile dove i **simboli** sono **ripetuti** molte volte, per esempio immagini con **aree con pixels uguali** (cartoni)
- esempio: **immagini GIF**



AAAAAAAAAAAAAAAAAAAA



15A

Relative Encoding

- Tecnica di trasmissione in cui vengono trasmesse le differenze fra un valore ed quello che lo precede
- Per esempio 15106433003 verrebbe trasmesso come 1+4-4-1+6-2-1+0-3+0+3
- Se i valori successivi sono “vicini” fra di loro... bastano meno bits per la codifica

Codifica Huffman:

(abbreviazioni)

- Assegna **codici a lunghezza variabile** ai simboli così da usare codici “corti” per **simboli frequenti** e viceversa, (anziché usare codici tutti della stessa lunghezza)
- Comprimendo testi, **per esempio**, si usano codici a lunghezza variabile al posto dei Bytes ASCII e ai caratteri più comuni, ad es. spazio e “t” vengono associati codici brevi
- La codifica Huffman è efficace se i dati sono **dominati** da un piccolo numero di simboli

Codifica aritmetica

- Huffman è efficiente, ma è **ottimale** solo se le **probabilità** di occorrenza dei simboli sono potenze di due.
- La codifica Aritmetica non ha questa restrizione ed è più efficiente, (e più complicata)

Codifica Lempel-Ziv (dizionario)

- Usa un **dizionario** di sequenze di simboli (tabella di compressione)
- Quando c'è una sequenza, viene rimpiazzata dalla sua posizione nel dizionario
- Esistono variazioni sul tema, (come viene gestito il dizionario)
- In un testo una sequenza può essere una sillaba, una parola o anche una frase intera

guarda **quante** sequenze ripetute!

Codifica Lempel Ziv

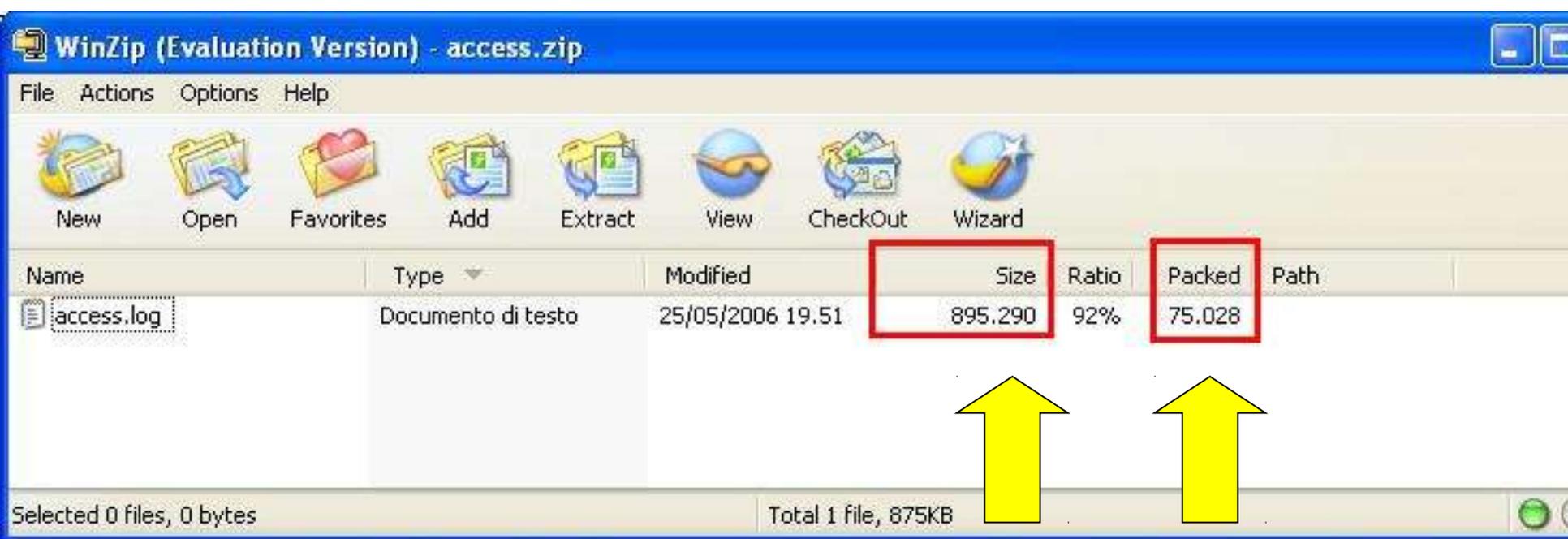


- Usata per esempio da winzip
- **Diabolicamente efficace** nel caso di **sequenze** (magari lunghe) ripetute spesso
- esempio del **file di log di un web server**

```
access.log - Blocco note
File Modifica Formato Visualizza ?
82.104.96.17 - - [25/May/2006:19:15:38 +0200] "GET /juke/dc/Cheru.gif HTTP/1.1" 200 866
82.104.96.17 - - [25/May/2006:19:15:52 +0200] "GET /juke/sc HTTP/1.1" 301 315
82.104.96.17 - - [25/May/2006:19:15:53 +0200] "GET /juke/sc/ HTTP/1.1" 200 2416
82.104.96.17 - - [25/May/2006:19:15:54 +0200] "GET /juke/sc/baufoto.jpg HTTP/1.1" 200 9
82.104.96.17 - - [25/May/2006:19:16:08 +0200] "GET /juke/ni HTTP/1.1" 301 315
82.104.96.17 - - [25/May/2006:19:22:32 +0200] "GET /juke/ni/ HTTP/1.1" 200 3191
82.104.96.17 - - [25/May/2006:19:22:34 +0200] "GET /juke/ni/axl4.jpg HTTP/1.1" 200 1325
82.104.96.17 - - [25/May/2006:19:22:34 +0200] "GET /juke/ni/jhon.jpg HTTP/1.1" 200 2891
82.104.96.17 - - [25/May/2006:19:22:37 +0200] "GET /juke/ni/tommy.jpg HTTP/1.1" 200 306
82.104.96.17 - - [25/May/2006:19:22:41 +0200] "GET /juke/ni/bjmi.ram HTTP/1.1" 200 36
82.104.96.17 - - [25/May/2006:19:22:52 +0200] "GET /juke/ni/bjmi.rm HTTP/1.1" 404 291
82.104.96.17 - - [25/May/2006:19:23:08 +0200] "GET /juke/ni/bjmi.rm HTTP/1.1" 404 291
82.104.96.17 - - [25/May/2006:19:23:18 +0200] "GET /juke/ni/lzsh.ram HTTP/1.1" 200 36
82.104.96.17 - - [25/May/2006:19:23:21 +0200] "GET /juke/ni/lzsh.rm HTTP/1.1" 200 19963
192.168.0.2 - - [25/May/2006:19:51:48 +0200] "GET /Sound%20Effects HTTP/1.1" 301 323
192.168.0.2 - - [25/May/2006:19:51:48 +0200] "GET /Sound%20Effects/ HTTP/1.1" 200 2077
192.168.0.2 - - [25/May/2006:19:51:48 +0200] "GET /icons/blank.gif HTTP/1.1" 200 148
192.168.0.2 - - [25/May/2006:19:51:48 +0200] "GET /icons/folder.gif HTTP/1.1" 200 225
192.168.0.2 - - [25/May/2006:19:51:48 +0200] "GET /icons/back.gif HTTP/1.1" 200 216
192.168.0.2 - - [25/May/2006:19:51:53 +0200] "GET /Sound%20Effects/Miscellaneous/ HTTP/1.1" 301 323
192.168.0.2 - - [25/May/2006:19:51:53 +0200] "GET /icons/sound2.gif HTTP/1.1" 200 221
```

Fattore di compressione **oltre 11!**

- Ovviamente, **senza perdita!!**





Compressione con perdita per oggetti multimediali



Compressione con perdita (lossy)

- De-comprimendo NON si ri-ottiene esattamente l'originale
- però...
- **Può** non esserci una **apprezzabile** differenza perché...
- ...vengono **soppresse** informazioni non essenziali
- Si può ***decidere quanto*** comprimere
- Usata quasi sempre per **oggetti multimediali** (immagini, audio, video ...)

Lossy: esempio 1: MP3 (con perdita)

- **Musica** CD (non compressa)
 - circa 1500 Kbit/sec
- compressa in **MP3**:
 - da 32 a 320 Kbps, **solitamente** 128Kbps
- fattore di compressione *medio* **circa 12**
- **nessuna apprezzabile perdita di qualità sopra i 90-100Kbps** (ascoltare per credere!)

Lossy: esempio 2 telefonata (con perdita)

- Suoni di tipo **vocale**.
- Conversazione telefonica digitale non compressa (ISDN)
 - **64Kbps**
- compressione audio fatta dal **cellulare GSM**
 - GSM 06.10
 - **13 Kbps**
 - **Fattore 5, nessuna perdita apprezzabile di qualità**

Compressione immagini (foto)

- Immagini (JPEG) fattore tipico 10 (variabile)
- perdita di dettaglio

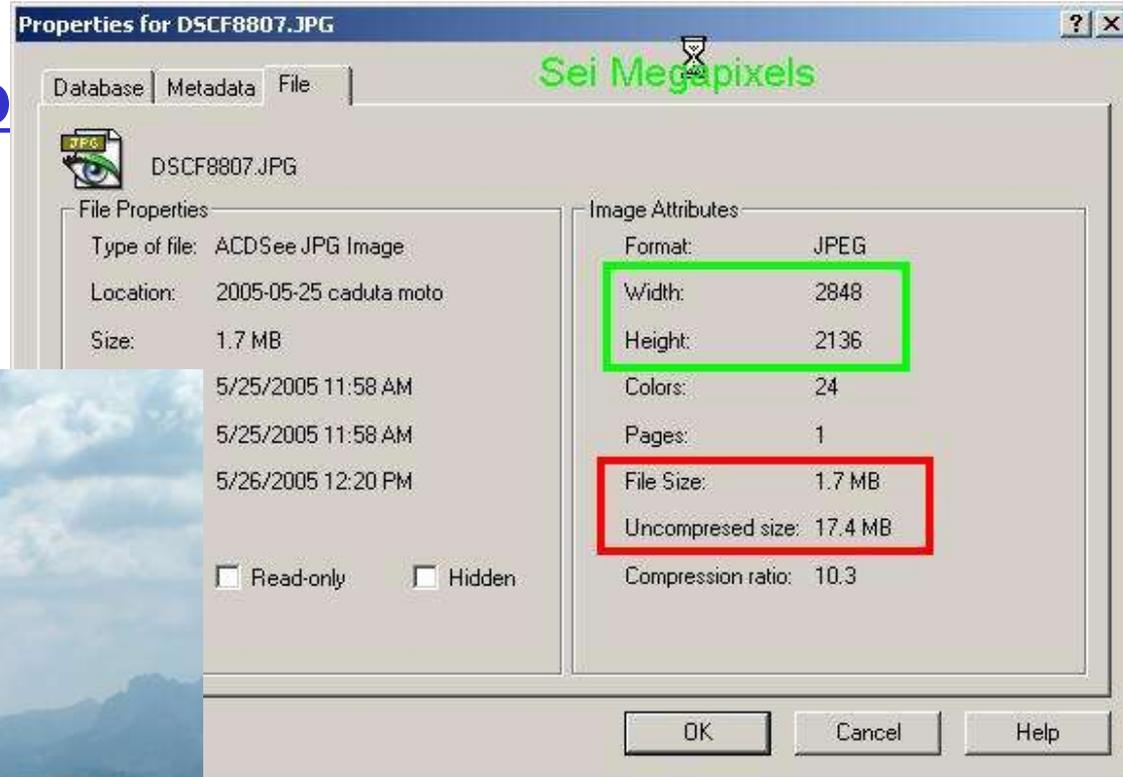


Foto: esempio 3

- Immagine da **un megapixel** (oggi una foto tipica occupa sui 15MP)
- **Non** compressa (formato BMP o RAW)
 - Occupa 3 Megabytes (3 bytes per ogni pixel)
- Compressa in **Jpeg**
 - occupa circa 100-150Kbytes
 - fattore 20-30!
- Compressione variabile! (la decidi tu)



JPEG qualità 10% - 3,2 Kb



JPEG qualità 50% - 6,7 Kb



JPEG qualità 90% - 30,2 Kb



JPEG qualità 100% - 87,7 Kb

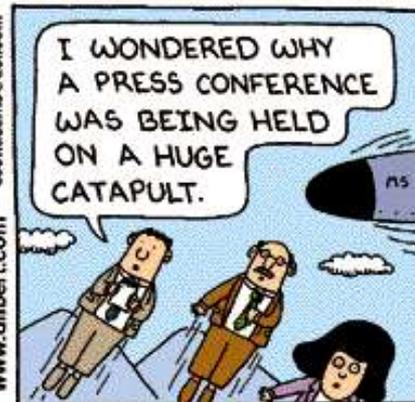
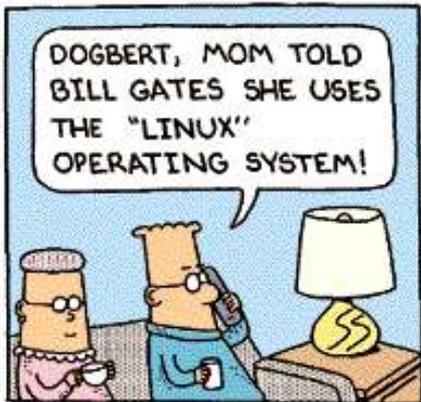
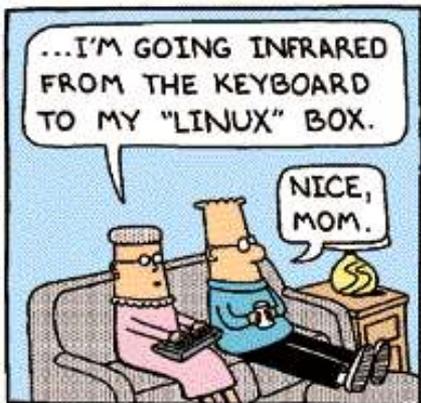


Immagini disegni, fumetti

- Compressione **GIF** (perdita di colori)
- OK per immagini con **pochi colori ed ampie aree dello stesso colore**

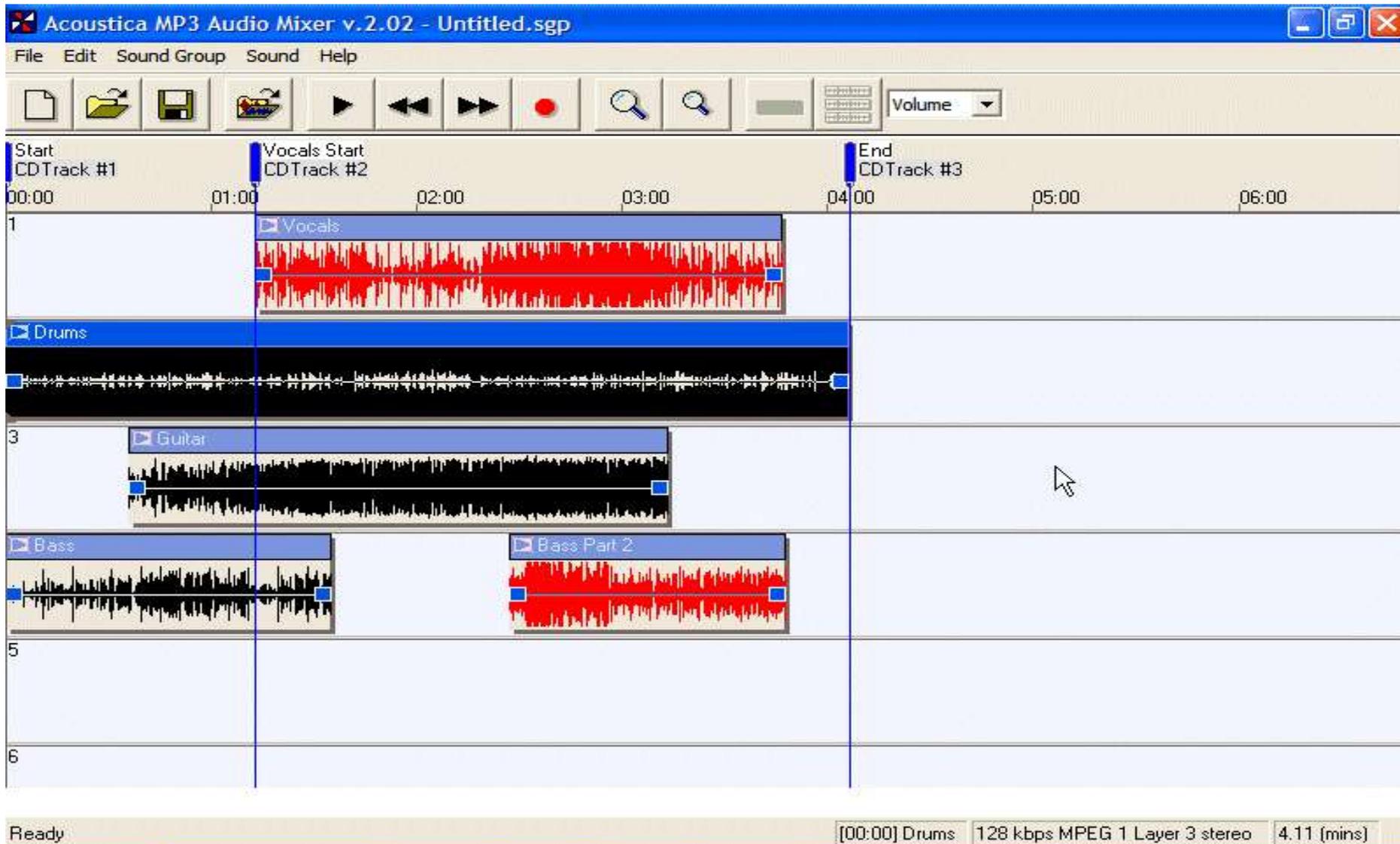


Esempi GIF



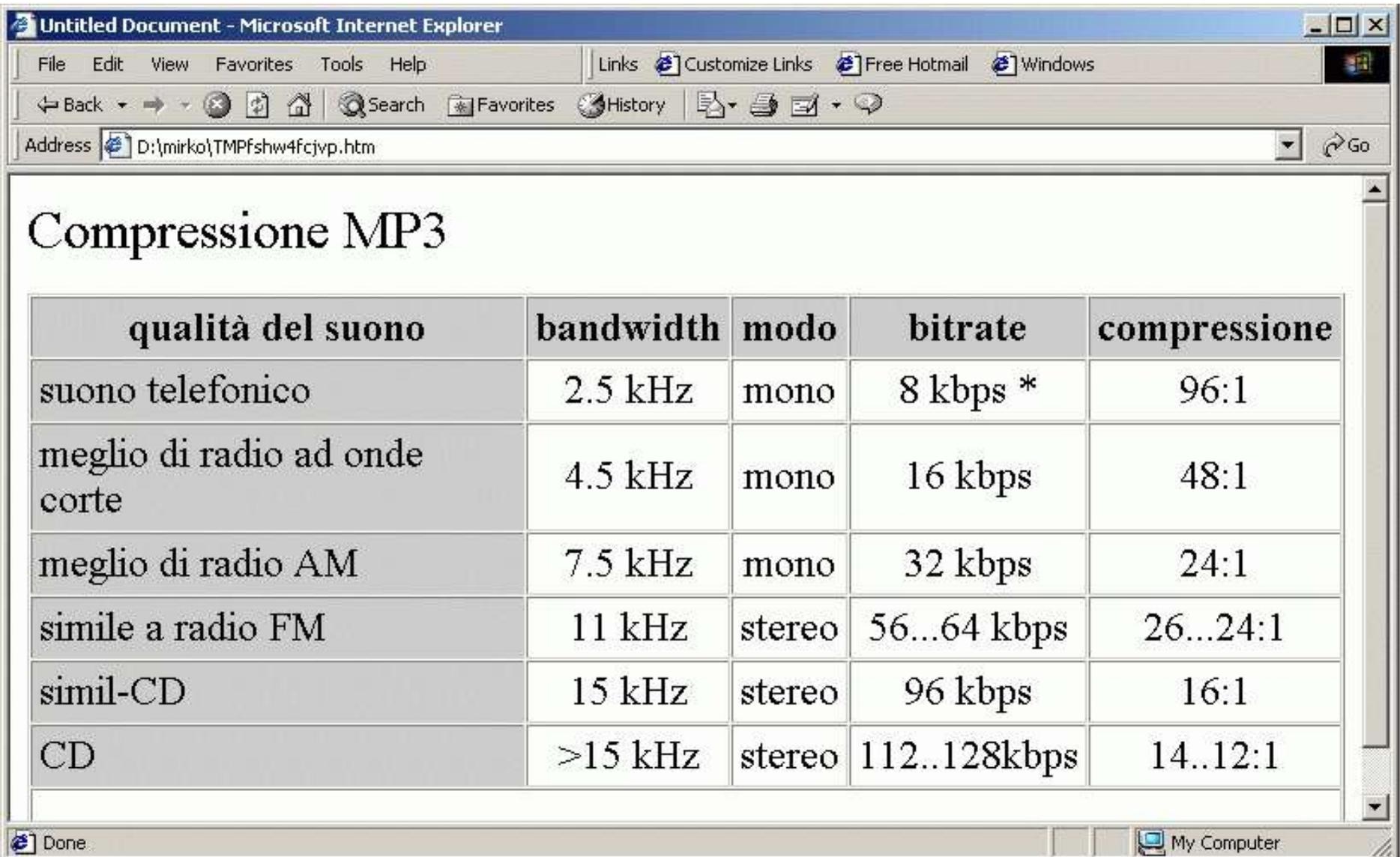
Audio: fattori di compressione

- **Suoni** (MP3) fattore 10... ma anche 20 o 30



Qualità audio di MP3

(il purista usa VBR)



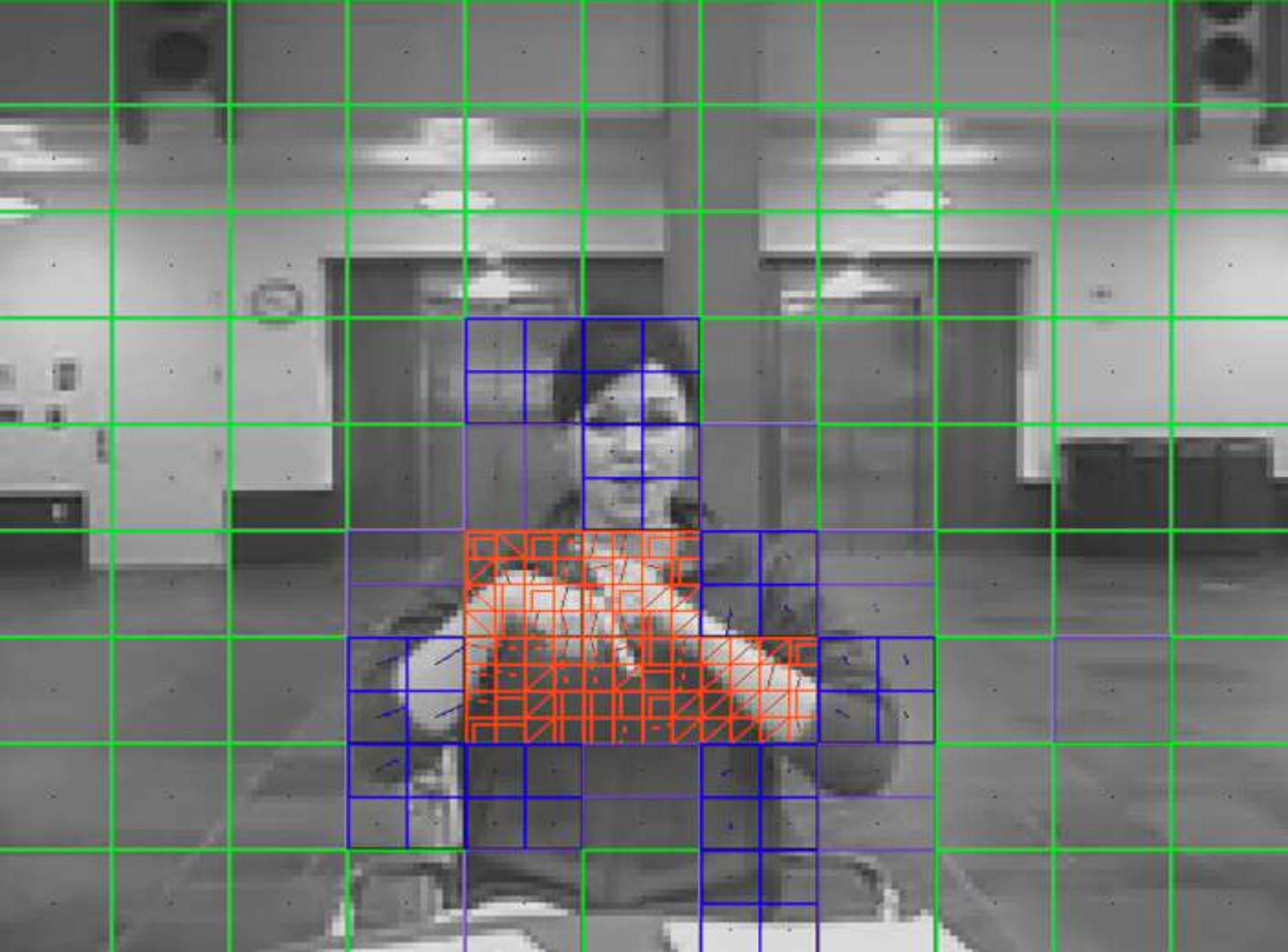
The screenshot shows a Microsoft Internet Explorer window with the address bar displaying 'D:\mirko\TMPfshw4fcjvp.htm'. The main content area contains the title 'Compressione MP3' and a table with five columns: 'qualità del suono', 'bandwidth', 'modo', 'bitrate', and 'compressione'. The table lists various audio quality levels from 'suono telefonico' to 'CD'.

qualità del suono	bandwidth	modo	bitrate	compressione
suono telefonico	2.5 kHz	mono	8 kbps *	96:1
miglior di radio ad onde corte	4.5 kHz	mono	16 kbps	48:1
miglior di radio AM	7.5 kHz	mono	32 kbps	24:1
simile a radio FM	11 kHz	stereo	56...64 kbps	26...24:1
simil-CD	15 kHz	stereo	96 kbps	16:1
CD	>15 kHz	stereo	112..128kbps	14..12:1

Compressione *video*

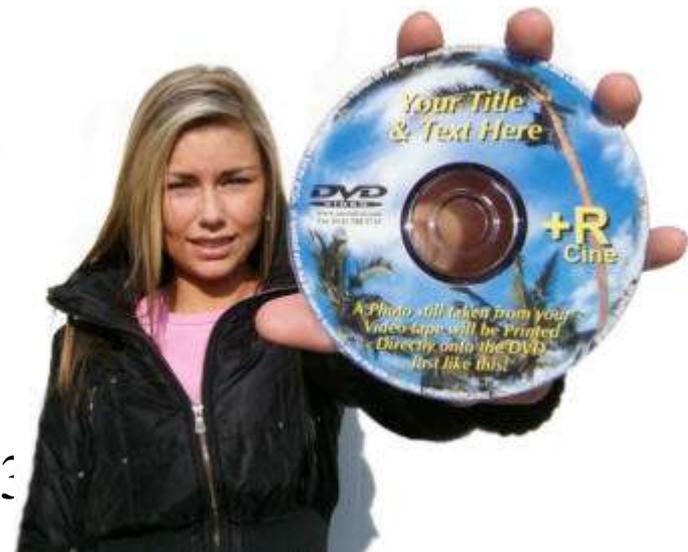
- Fattore tipico 100!!!!
- Compressione **per differenze** fra fotogrammi
- I fotogrammi consecutivi **sono molto simili**
- Fotogrammi scomposti in “piastrelle” quadrate 8*8 o 16*16 pixels o anche rettangolar (chiamate **macroblocchi**)
- ogni **fotogramma** viene rapportato ai
 - precedenti
 - **o... successivi!!**





Esempio IV: DVD video

- **Tipico** film in DVD
 - un'ora e 40 minuti cioè **150.000 fotogrammi**
 - 720 per 576 pixels **l'uno** (circa 400.000 pixel)
 - con 24 bits di colore (ovvero 3 Bytes per pixel)
- Dimensioni totali (non compresso) **180 Gigabytes!!**
- compresso in Mpeg**2** diventano **6-8 Gigabytes**
- In MPEG**4** basta circa **1 GB**



Un caso particolare: il Powerpoint

- Una tipica presentazione PPT
- è un documento “di testo” che ...
- **contiene** anche oggetti multimediali (immagini ed altro)
- Se “obesa” diventa ingestibile.
- esistono **programmi appositi** (Powershrink per esempio)
- che comprimono gli oggetti multimediali all’interno (senza perdita apprezzabile di qualità)



Open



Shrink



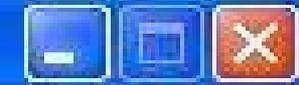
Shrink&Mail



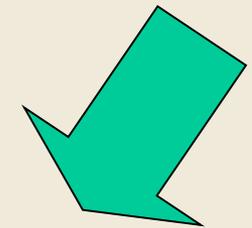
Settings



Wizard



1 file(s) shrunked



Original file(s):
25,34 Mb

Shrunked file(s):
1,61 Mb

Total space saved:
23,73 Mb



OK

Tipica **insensatezza**

- Di un utente **sprovveduto**
- Comprimere con winzip o winrar
- Oggetti multimediali già compressi come:
 - MP3
 - Jpeg
 - Video
- Se sono già compressi cosa li comprimi a fare?
- Prova anche tu e poi dimmi...

RI-compressione

- Si usa per ridurre **ulteriormente** le dimensioni di oggetti multimediali **già compressi**
- Esempi
 - Portare una canzone MP3 da 128Kbps a 32Kbps (diventa grande un quarto ma cala la qualità)
 - Diminuire la **qualità** Jpeg di una **foto**
 - Diminuirne le **dimensioni** in pixel (tipica foto digitale 8-12 Megapixels, schermo PC da 500K a 1.5Megapixels)
 - Per usarla su E-mail o pubblicarla su internet
 - Ricomprimere un **filmato** per poterlo visualizzare sul cellulare (codec 3GP) o fare upload su internet.

Esercizi

Esercizi

- Compressione di
 - Scarica testi da <http://www.liberliber.it>.
 - comprimi la divina commedia o un altro testo “grosso”
 - prova a comprimere con Winzip o Winrar
 - Fogli elettronici
 - Documenti Word processing
 - Presentazioni PPT
 - documenti PDF
 - Immagini (jpeg etc.)
 - MP3

Esercizi

- Crea una immagine **bitmap** col paint
- Salvala anche come GIF e come Jpeg
- Ci sono differenze visibili?
- Comprimi il tutto con Winzip o Winrar e analizza le differenze



Esercizi



- Riesci a ricomprimere con il **registratore di suoni** (quello di windows XP) un file MP3?
- Cerca su Internet un programma, magari **freeware**, che ricomprima musica MP3
- Magari su download.com?
- Magari “audacity”?

Esercizi:

- Scarica da Internet:
 - una piccola [immagine GIF](#)
 - Un GIF [animato](#)
 - Un [testo](#) di un centinaio di KB
 - Una immagine JPG di un centinaio di KB
 - Un foglio elettronico Excel
- Scegli fra winzip e winrar e..
 - Comprimili insieme
 - Analizza i vari rapporti di compressione.
 - Proteggili con password
 - Cifrali
 - Inviarmi il tutto come [allegato via email](#)

Compressione: FINE!

Diapositive extra

Misure di compressione

- *bit/pixel*: indica il numero medio di bit con cui ogni pixel dell'immagine è codificato; valori tipici sono compresi tra 0.2 e 1.5;
- *rapporto di compressione*: è il rapporto tra il numero di byte impiegati per codificare l'immagine di input e quelli impiegati per codificare l'immagine compressa.

Codificare **senza** perdita o **con** perdita di informazione

- Metodi non lossy
 - RLC;
 - Huffmann;
 - Aritmetico;
-
- Metodi lossy
 - trasformate;
 - a dizionario.

Immagine: mappa di pixel - 1

Il modo più semplice per codificare un'immagine digitale è quello di suddividerla in una griglia di piccoli punti ai quali associare un codice numerico che ne rappresenta il colore.

Queste “mappe di pixel” hanno il vantaggio di poter venire facilmente elaborate, ma hanno anche lo svantaggio di occupare una grande quantità di spazio.

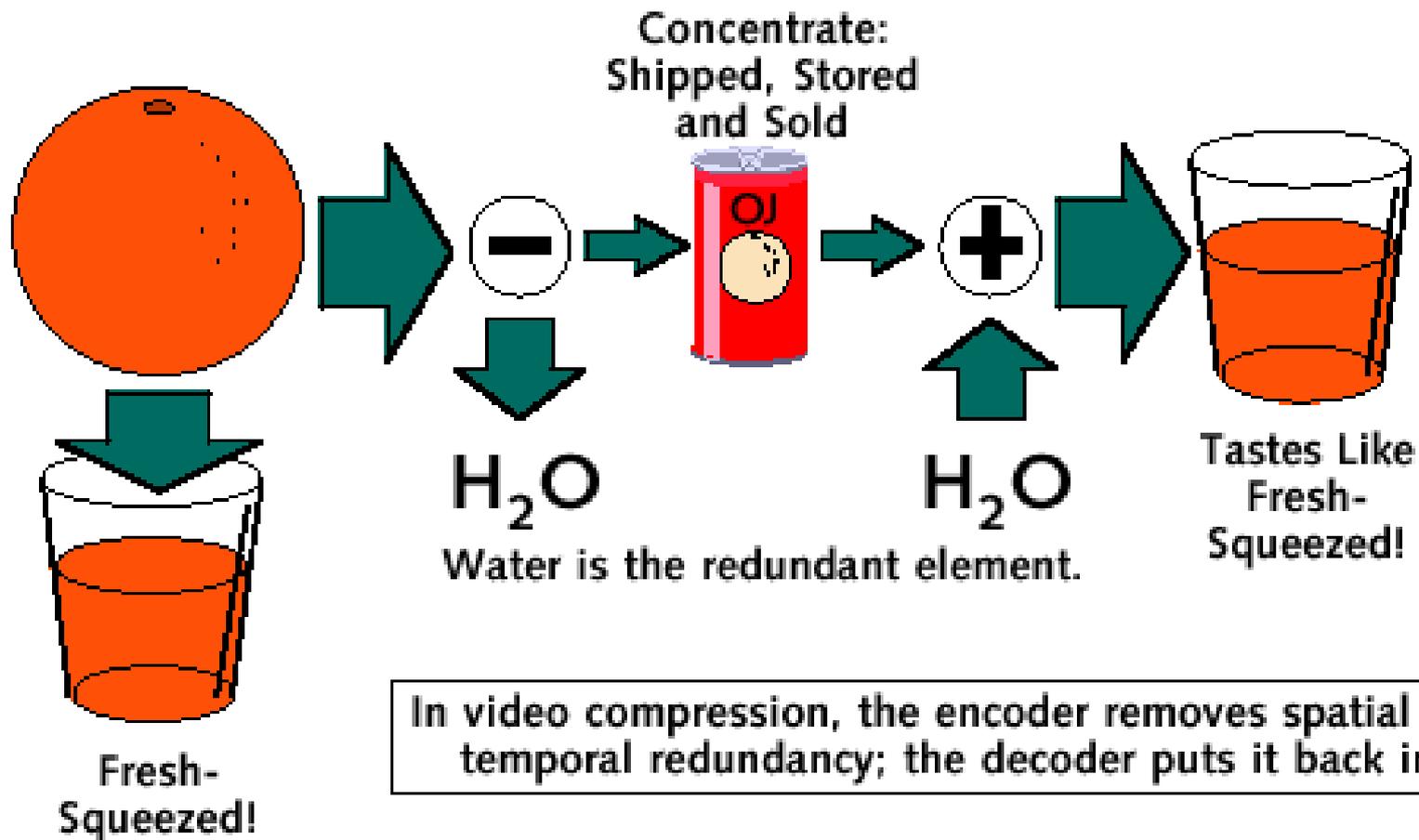
Immagine: mappa di pixel - 2

Elenco di alcuni formati tipici di **immagini**, e quanta memoria (Bytes) occupa **un** fotogramma:

- QCIF: 160 x 120
- CIF: 320 x 240
- VGA: 640 x 480
- NTSC: 720 x 486
- Workstation: 1280 x 1024
- HDTV: 1920 x 1080
- 35mm slide: 3072 x 2048

8 bit	24 bit
19.200	57.600
76.800	230.400
307.200	921.600
349.920	1.049.760
1.310.720	3.932.160
2.073.600	6.220.800
6.291.456	18.874.368

COdifica e DECodifica immagini audio e video
simile al succo concentrato d'arancia...

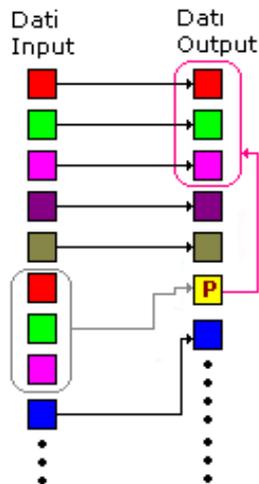


Algoritmi di Compressione

- Compressione Mediante Dizionari -

LZ1

Le occorrenze, successiva alla prima, vengono sostituite con un puntatore definito dalla coppia $\langle \text{distanza, lunghezza} \rangle$



LZ2

Si costruisce un elenco delle stringhe che sono state rilevate (*dizionario*) e si sostituiscono le ricorrenze con l'indice ad esse associato

