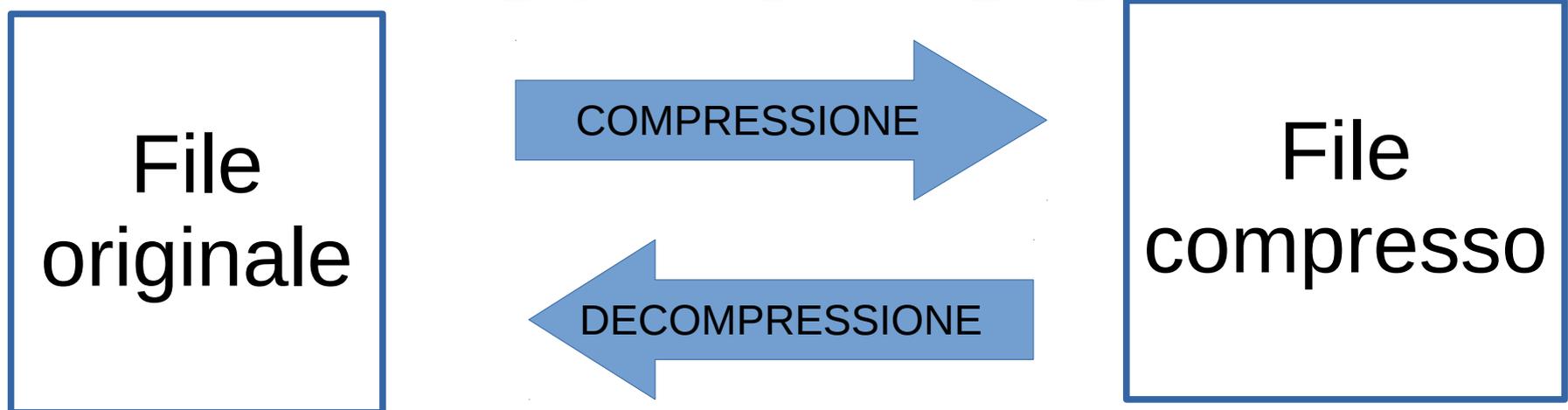


Compressione

- Cos'è la compressione?
- A cosa serve?
- Compressione senza e con perdita di qualità
- Alcuni algoritmi: RLE, Huffman
- Compressione delle immagini
 - Il formato JPG
- Compressione di file audio
 - Il formato MP3
- Compressione di filmati
 - Il formato MPEG-4

Cos'è la compressione?

- E' una tecnica di elaborazione dei dati attraverso la quale, per mezzo di opportuni algoritmi, si riduce il numero di bit necessari a rappresentare un dato.
- In altri termini, per rappresentare un dato, tramite la compressione utilizza un numero di bit inferiore a quello che si userebbe per rappresentarlo con il metodo “convenzionale”
- L'algoritmo usato deve essere reversibile!





Compressione

A cosa serve?

- A ridurre lo spazio occupato dai dati su un supporto di memorizzazione
- A ridurre il tempo impiegato nella trasmissione dei dati

Compressione

Svantaggi:

- Tempo necessario per comprimere/decomprimere
- Necessità di decomprimere un file compresso prima di usarlo
- Maggior sensibilità agli errori: un solo bit errato può rendere inutilizzabile il file

NB Non esiste un algoritmo che riduca la dimensione di qualsiasi file!

Compressione

Esempio: quanta memoria può richiedere un'enciclopedia multimediale?

500.000 pagine di testo (2 KB per pagina): **1GB**

3.000 immagini a colori (in media $640 \times 480 \times 24$ bit = 1MB): **3 GB**

500 mappe (in media $640 \times 480 \times 16$ bit = 0.6 MB): **300 MB**

60 minuti di suono stereo (176KB/sec.): **600 MB**

30 animazioni di durata media di 2 minuti ($640 \times 320 \times 16$ bit x 16 fps = 6.5 MB/s): **23 GB**

50 video, di durata media di 1 minuto ($640 \times 480 \times 24$ bit x 30 frame/sec = 27.6 MB/s): **83 GB**

Totale: 111 GB

Compressione

Utilizzando la compressione:

Testo **2:1** → **0.5 GB**

Immagini a colori **15:1** → **0.2 GB**

Mappe **10:1** → **0.03 GB**

Suono stereo **6:1** → **0.1 GB**

Animazioni **50:1** → **0.47 GB**

Video **50:1** → **1.66 GB**

Totale: 2.96 GB (rispetto a 111 GB!), quindi circa 40:1

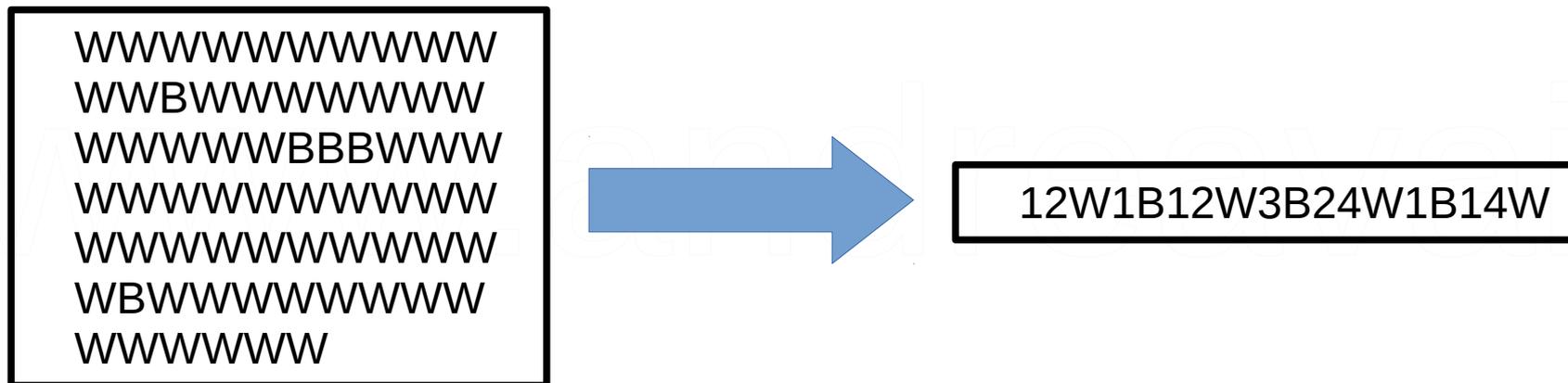
Compressione lossless e lossy

2 tipi di compressione:

- Lossless (senza perdita di qualità): i dati di partenza vengono ricostruiti identici in seguito alla decompressione
 - Esempi: Zip, TIFF/LZW, PNG...
- Lossy (con perdita di qualità): si alterano i dati del file originale prima di comprimerlo per ottenere una compressione migliore
 - Esempi: JPEG, MP3

Compressione

Algoritmo RLE (Run-Length Encoding)



18 Byte invece di 67 (compression ratio 3.72 : 1)



Compressione

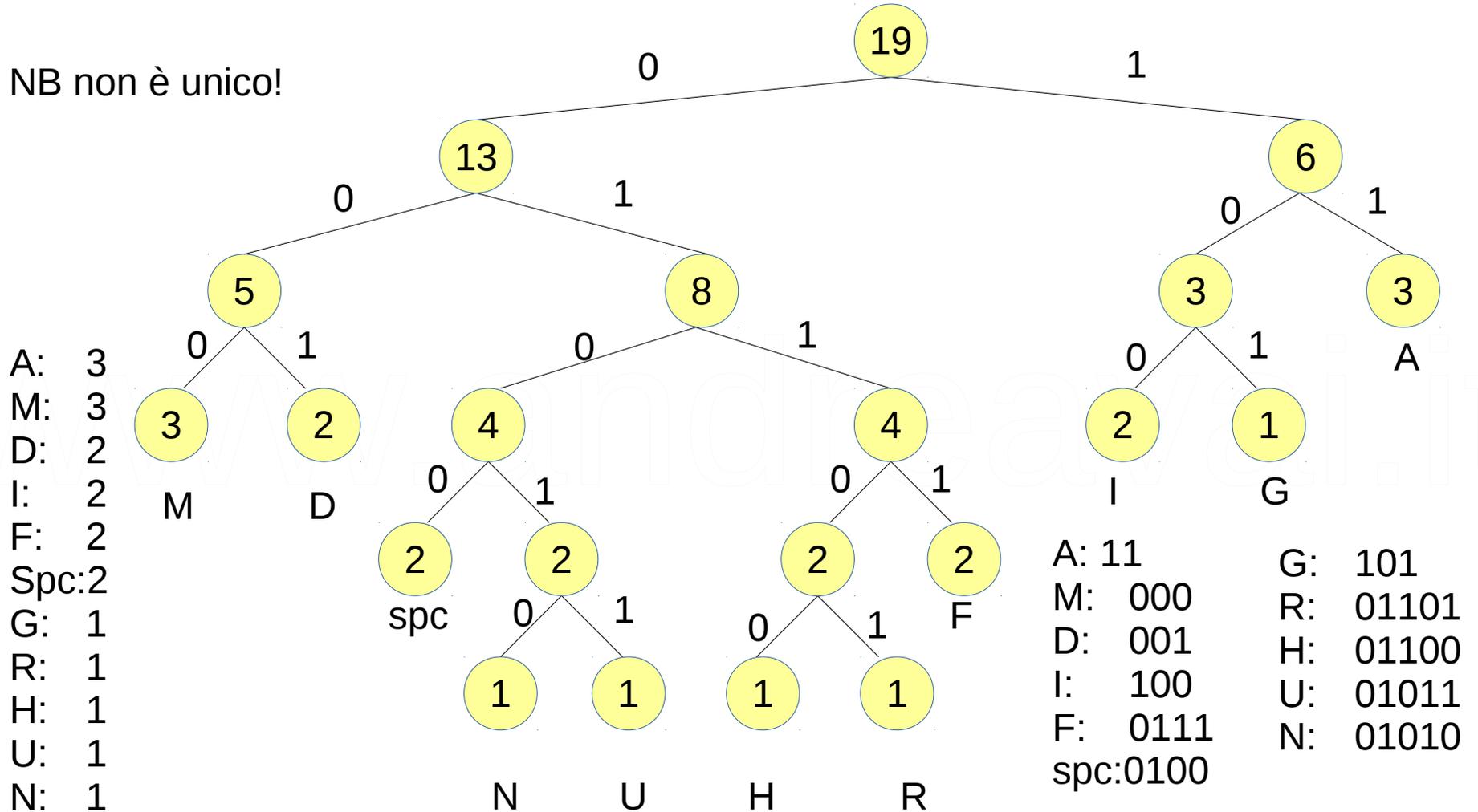
Algoritmo di Huffman

- Codifica con un numero inferiore di bit le informazioni più frequenti, e con un numero maggiore quelle meno frequenti
- Per comprenderlo meglio si sfruttano i diagrammi di Huffman

Compression

Esempio: "DIAGRAMMA DI HUFFMAN": (19 Byte)

NB non è unico!



Tot: 19 (152 bit)



Tot: $2 \times 3 + 3 \times 3 + 3 \times 2 + 3 \times 2 + 4 \times 2 + 4 \times 2 + 3 \times 1 + 5 \times 1 + 5 \times 1 + 5 \times 1 + 5 \times 1 = 56$ bit

Compressione

Immagini: il formato JPG

- Lossy
- Se un pixel ha un colore simile a quelli adiacenti lo sostituisce con un pixel dello stesso colore di quelli adiacenti (o del colore che rende migliore la compressione)



Compressione

Audio: il formato MP3

- Lossy
- Riduce o elimina le frequenze normalmente non avvertite o poco percepite dall'orecchio umano

Compressione

Filmati: il formato MPEG-4

- Lossy
- Alla compressione di immagini (spaziale) e audio unisce una compressione temporale, tramite keyframe e compressione differenziale (ovvero, vengono memorizzate solo le variazioni rispetto al frame di riferimento)