



# CSS - parte 2

Corso di Web Design

Fabio Pittarello, Università Ca' Foscari Venezia - DAIS [pitt@unive.it](mailto:pitt@unive.it)

**Nota:** il materiale contenuto in questo documento è disponibile solo per uso interno nell'ambito del corso di Web Design.

## Gli attributi tipografici

---

- Windows, Macintosh e Unix hanno diversi set di font, diverse risoluzioni predefinite e diverse tecnologie di rendering (a pixel interi, con antialias moderato, Cleartype).
- Molti metodi di dimensionamento dei font (anche in CSS) hanno prodotto soprattutto nel passato e continuano a produrre (a causa di vari fattori tra cui bug di implementazione) risultati differenti nei diversi sistemi operativi.

## Unità di misura per specificare la dimensione lineare con CSS

---

px	Pixel	Le unità in pixel corrispondono all'unità minima di colore visualizzabile sul monitor
pt	Punto (Point)	Misura tipografica tradizionale; ci sono circa 72 punti per ogni pollice (inch)
pc	Pica	Misura tipografica tradizionale; corrisponde a 1/6 di pollice
em	Em	Unità di misura relativa che corrisponde alla larghezza della lettera M maiuscola nel carattere utilizzato.
ex	Ex	Unità di misura relativa che corrisponde all'altezza della lettera x minuscola nel carattere utilizzato (circa la metà di un em).
in	Pollici (Inches)	Unità di misura standard negli USA.
mm	Millimetri	
cm	Centimetri	

## Specificare la grandezza del carattere

---

- Tutte le unità di misura viste precedentemente possono essere utilizzare per specificare la grandezza del carattere, anche se la preferenza dovrebbe essere accordata a quelle unità di misura pensate per il mezzo informatico (**pixel**, **em** ed **ex**).
- Oltre a queste unità di misura è possibile specificare la grandezza del carattere in **percentuale** (es. 50%) rispetto alla grandezza del font predefinito del browser.
- Un'altra possibilità, che richiede di rinunciare ad una parte dell'espressività nella specifica della dimensione, ma che garantisce altri vantaggi (vedi slides successive) è data dall'uso delle cosiddette **keywords**.

## Utilizzare i pixel - I

---

- Funzionano correttamente su browser vecchi e nuovi, sia per la definizione dei font che per la definizione di altre caratteristiche di presentazione come i margini, dando la possibilità di definire in modo preciso relazioni visuali
- Unica eccezione sono alcune vecchissime versioni di Netscape e alcune versioni di Opera (es. Opera 5 per Macintosh), che possono visualizzare dimensioni leggermente inferiori rispetto a quelle specificate; questo problema è risolto nelle versioni 7 di Opera.
- Il pixel, inteso comunemente dalla maggior parte dei designer, come unità di misura assoluta, viene intesa nelle specifiche CSS come unità relativa, dal momento che la dimensione effettiva di visualizzazione dipende dalla risoluzione del monitor.

## Utilizzare i pixel - II

---

- Il problema dell'uso del pixel è legato a Internet Explorer per Windows. Anche se la specifica CSS non dichiara che i caratteri le cui dimensioni sono espresse in font non devono essere ridimensionabili, la scelta implementativa di questo browser stabilisce che la dimensione di un testo fissata in pixel non può essere modificata dall'utente.
- IE 5/Macintosh ha proposto un'interessante soluzione di questo problema attraverso la tecnologia TextZoom, che consente di ridimensionare tutto il contenuto della pagina con vari livelli di scala. La tecnologia è stata successivamente adottata dalla quasi totalità dei browser (Mozilla, Opera, ecc.) tranne che dal browser più diffuso: IE/Windows. Solo la versione 7 di IE per Windows permette di riscalare il carattere espresso in pixel.

## Utilizzare i pixel – III

---

- Da questo punto di vista IE/Windows costituisce il maggior ostacolo per il design di pagine che siano contemporaneamente coerenti su piattaforme diverse e contemporaneamente accessibili. Il problema è stato risolto solo per la versione 7 del browser, recentemente introdotta e non ancora molto diffusa.
- In conclusione, data la diffusione delle versioni precedenti del browser, non è possibile utilizzare la misura in pixel ottenendo al contempo siti con caratteri ridimensionabili per tutti gli utenti.
- Quali possibilità per gli utenti di IE6 Windows per ridimensionare i caratteri stabiliti con l'unità di misura in pixel? Ci sono solo soluzioni (molto) parziali, come scrivere un foglio di stile personalizzato per IE6/Windows e associarlo in base a meccanismi di riconoscimento del browser

## Utilizzare gli em e le percentuali

---

- Utilizzando gli em e le percentuali seguiamo la direzione proposta dalle linee guida sull'accessibilità, fornendo una presentazione scalabile e omogenea (a patto che la grandezza del carattere tipografico di default sia omogenea per tutti i browser; nonostante scelte diverse fatte in passato dai diversi produttori, possiamo dire che ora questo requisito sia sostanzialmente vero)
- Nel passato alcuni bug implementativi facevano sì che si ottenevano risultati omogenei dal punto di vista della presentazione sui diversi browser solo se il valore specificato era pari o maggiore a quello della dimensione predefinita. In caso contrario (es. specificando un valore di 0.5em) si potevano ottenere risultati diversi (per approfondimenti si veda [http://www.thenoodleincident.com/tutorials/box\\_lesson/font/](http://www.thenoodleincident.com/tutorials/box_lesson/font/))

# Le parole chiave (keywords)

- Sette parole chiave definite da CSS1 per controllare le dimensioni del carattere:
  - xx-small
  - x-small
  - small
  - medium
  - large
  - x-large
  - xx-large
- Un vantaggio fondamentale nell'uso delle parole chiave si verifica nel caso di tag nidificati:
  - se si assegna un valore di grandezza di carattere utilizzando come riferimento un tag esterno (rispetto alla gerarchia specifica dei tag), questo valore verrà trasmesso anche ai contenuti dei tag nidificati; questo comportamento si verificherà sia che si usino le percentuali, gli ems o le keywords
  - tuttavia nel caso di un'ulteriore specifica di grandezza di carattere riferita ad un tag nidificato:
    - nel caso di utilizzo di em o percentuali le due indicazioni verranno composte
    - nel caso di utilizzo di keywords le indicazioni di grandezza riferite al tag esterno non verranno applicate ai contenuti del tag nidificato; per questi verrà la grandezza di carattere fissata per questo tag
- Per maggior chiarezza si veda l'esempio delle slides successive

# Le parole chiave (keywords)

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
6 <title>Test dimensione caratteri</title>
7 <style type="text/css">
8 <!--
9 #main {font-size: 1.5em;}
10 #main2 {font-size: 150%;}
11 #main3 {font-size: x-large;}
12 -->
13 </style>
14 </head>
15 <body>
16 <div id="main">
17 <p>Lorem ipsum sit amet 1.5em</p>
18 <div id="content">
19 <p>Lorem ipsum sit amet</p>
20 <p id="note">Parvus</p>
21 </div>
22 </div>
23 <div id="main2">
24 <p>Lorem ipsum sit amet 150%</p>
25 <div id="content2">
26 <p>Lorem ipsum sit amet</p>
27 <p id="note2">Parvus</p>
28 </div>
29 </div>
30 <div id="main3">
31 <p>Lorem ipsum sit amet x-large</p>
32 <div>
33 <p>Lorem ipsum sit amet</p>
34 <p id="note3">Parvus</p>
35 </div>
36 </div>
37 </body>
38 </html>
```

Lorem ipsum sit amet 1.5em

Lorem ipsum sit amet

Parvus

Lorem ipsum sit amet 150%

Lorem ipsum sit amet

Parvus

Lorem ipsum sit amet x-large

Lorem ipsum sit amet

Parvus

In questo esempio vengono applicate le regole di stile utilizzando come unità di misura, nell'ordine, em, % e keywords

Le regole applicate a selettori id associato a tag div vengono trasmessi ai livelli inferiori della gerarchia dei tag

# Le parole chiave (keywords)

```
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
5 <title>Test dimensione caratteri</title>
6 <style type="text/css">
7 <!--
8 #main {font-size: 1.5em;}
9 #content {font-size: 2em;}
10 #note {font-size: 1em;}
11 #main2 {font-size: 150%;}
12 #content2 {font-size: 200%;}
13 #note2 {font-size: 100%;}
14 #main3 {font-size: x-large;}
15 #content3 {font-size: xx-large;}
16 #note3 {font-size: medium;}
17 -->
18 </style>
19 </head>
20 <body>
21 <div id="main">
22 <p>Lorem ipsum sit amet 1.5em</p>
23 <div id="content">
24 <p>Lorem ipsum sit amet 2em</p>
25 <p id="note">Parvus 1em</p>
26 </div>
27 </div>
28
29 <div id="main2">
30 <p>Lorem ipsum sit amet 150%</p>
31 <div id="content2">
32 <p>Lorem ipsum sit amet 200%</p>
33 <p id="note2">Parvus 100%</p>
34 </div>
35 </div>
36
37 <div id="main3">
38 <p>Lorem ipsum sit amet x-large</p>
39 <div id="content3">
40 <p>Lorem ipsum sit amet xx-large</p>
41 <p id="note3">Parvus medium</p>
42 </div>
43 </div>
44
45 </body>
46 </html>
```

Lorem ipsum sit amet 1.5em

Lorem ipsum sit amet 2em

Parvus 1em

Lorem ipsum sit amet 150%

Lorem ipsum sit amet 200%

Parvus 100%

Lorem ipsum sit amet x-large

Lorem ipsum sit amet xx-large

Parvus medium

In questo esempio vengono aggiunte ulteriori regole di stile, applicandole a selettori id associato a tag div nidificati all'interno dei precedenti.

Il rendering degli elementi della pagina varia a seconda delle unità di misura utilizzate:

- nel caso di em e % avviene una composizione;
- nel caso delle keywords ha la prevalenza la regola associata al tag più vicina ad un determinato contenuto e che lo nidifica.

## Le parole chiave: linee guida CSS2.1

- La tabella illustra le linee guida stabilite dalla specifica CSS per i produttori di browser per associare il fattore di scala alle diverse parole chiave, alle intestazioni XHTML e alle dimensioni dei font ottenibili attraverso l'uso del deprecato attributo XHTML font size.
- Nota: IE Win fino alla versione 5.5 era caratterizzato da un mapping sbagliato tra keywords e fattore di scala, portando a risultati di rendering diversificati. Per risolvere questo problema si ricorreva ad hack di vario tipo, tra cui il metodo Fahrner, che consisteva fondamentalmente nella dichiarazione di alcune regole di stile che in alcuni browser interrompevano la lettura di parte del codice CSS successivo. Il problema non sussiste più dalla versione 6 del browser.

CSS absolute-size values	xx-small	x-small	small	medium	large	x-large	xx-large	
scaling factor	3/5	3/4	8/9	1	6/5	3/2	2/1	3/1
HTML headings	h6		h5	h4	h3	h2	h1	
HTML font sizes	1		2	3	4	5	6	7

## Il metodo Fahrner: un esempio di applicazione

---

- IE Win 5.5 associava la keyword `small` (anzichè `medium`) alla grandezza predefinita del carattere. Per ottenere un risultato omogeneo per tutti i browser si utilizzava il seguente procedimento:
- veniva dichiarata una regola di stile che permetteva di ottenere la grandezza desiderata in IE 5.X (ma non per i browser corretti):
- si inserivano nel codice CSS due regole `voice-family` (assolutamente prive di significato nel contesto d'uso) che IE 5.X non comprendeva e che causavano (per IE 5.X ed Opera 6.X) l'interruzione della lettura del codice successivo fino alla chiusura della parentesi grafa
- veniva dichiarato un secondo valore di keyword per la regola `font-size` che sovrascriveva la precedente per tutti i browser in grado di leggerla (tutti tranne IE 5.X ed Opera 6.X)
- infine per passare la regola corretta anche ad Opera 6.X (che comprendeva comunque correttamente le keywords ma era soggetto allo stesso bug di interpretazione delle regole `voice-family` di IE 5.X) veniva dichiarato a seguire un ulteriore selettore CSS2 di maggiore specificità, comprensibile da Opera ma non da IE 5.X

```
p      {font-size: x-small; /* per IE */
      voice-family: "\"}\"";
      voice-family: inherit;
      font-size: small; /* regola non interpretata da IE 5.X */
      }
html>p {font-size: small;}
```

## Le parole chiave (keywords)

---

- Le parole chiave sono basate sulle impostazioni predefinite dell'utente, come succede per gli em e per le percentuali.
- Tuttavia, a differenza di quello che succede per gli em e per le percentuali, se l'utente modifica l'impostazione del carattere di default, la maggior parte dei browser applica una regola di salvaguardia, impedendo al carattere di scendere sotto la soglia di leggibilità, fissata a 9px.
- In questo caso dunque l'uso delle parole chiave **x-small** e **xx-small** avrà come effetto la visualizzazione di testi appiattiti su un unico valore (appunto, **9 px**) evitando di dar luogo a dimensioni di carattere non leggibili.
- Questo limite non è fissato per gli em e per le percentuali, lasciando al designer il compito di verificare eventuali problemi di leggibilità nell'uso dei caratteri più piccoli.

# Proprietà dei caratteri

---

- I fogli di stile offrono una possibilità di controllo simile a quella che si trova nel desktop publishing.
- Font-family
  - Si possono specificare font specifici o famiglie generiche di font, separate da virgole
    - `h2 {font-family: Helvetica, sans-serif}`
- Font-style
  - E' possibile scegliere tra normal, italic e oblique
    - `h2 {font-style: italic}`
- Font-variant
  - Maiuscoletto
    - `h2 {font-variant: small-caps}`
- Font-weight
  - Specifica il peso del font; si possono utilizzare termini descrittivi (normal, bold, bolder, lighter) o valori numerici tra 100 a 900, a intervalli di 100)
    - `h2 {font-weight: 800}`
    - `h2 {font-weight: bold}`

# Proprietà dei caratteri

---

- Font-size
  - È possibile specificare la dimensione del testo, utilizzando 4 metodi
    - Length (numero con unità di misura scelta tra em|ex|px|pt|pc|mm|cm|in)
      - `h2 {font-size: 36pc}`
    - Percentage (numero %); percentuale della dimensione ereditata. Nell'esempio sottostante il font è il 140% più grande del testo normale
      - `h1 {font-size: 140%}`
    - Absolute size (xx-small|x\_small|small|medium|large|x-large|xx-large); ci si riferisce ad una tabella delle dimensioni fissata nel browser
      - `h2 {font-size: x-large}`
    - Relative size (larger|smaller); la misura è relativa a quella indicata per l'elemento padre o per l'elemento figlio nella gerarchia
      - `h2 {font-size: larger}`
- Font
  - E' una forma abbreviata per specificare tutte le caratteristiche del font in una sola regola; l'ordine è importante, anche se non occorre che siano presenti tutti i valori
    - `h2 {font: peso stile variante dimensione/altezza nome dei fonts}`
    - `h2 {font: bold 12em Helvetica, sans-serif}`



## CSS – Specificare i valori delle proprietà

---

- selettore {proprietà: valore}
- Specificare il colore
  - Per nome
    - La specifica CSS1 prevede 16 nomi di colore (aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, purple, red, silver, teal, white, yellow)
      - Esempio h4 {color: olive}
  - Per valore RGB
    - Utilizzando 6 cifre esadecimali (da 0 a f)
      - Esempio h4 {color: #2288ee}
    - Utilizzando 3 cifre esadecimali (è utilizzabile quando ognuna delle 3 coppie di valori RGB è costituita dallo stesso valore ripetuto)
      - Esempio h4 {color: #2c1}
    - Specificando 3 cifre decimali (da 0 a 255)
      - Esempio h4 {color: rgb(255,0,255)}
    - Specificando 3 valori percentuali (da 0% a 100% per ogni componente additiva; è possibile usare valori decimali)
      - Esempio h4 {color: rgb(0%,0%,15.2%)}

## Caratteri, immagini raster e unità di misura - I

---

- **Pixel:** è l'unità minima di informazione di un'immagine raster (cioè definita attraverso una griglia rettangolare di elementi) ed è associata ad un'informazione cromatica il cui range possibile è definito dalla profondità colore (espressa in bit; es. 8 bit, 24 bit, ecc.)
- **Risoluzione di un'immagine:** dà un'indicazione sul numero di unità informative minime in una certa dimensione lineare fisica
- **PPI** (pixel per inch) = numero di pixel per unità di lunghezza (pollice)
- **PPC** (pixel per cm) = numero di pixel per unità di lunghezza (centimetro)
- **Risoluzione di acquisizione** = numero di pixel utilizzati nell'acquisizione / dimensione reale lineare dell'immagine da acquisire
- **Risoluzione di visualizzazione** = numero di pixel visualizzati / dimensione reale lineare del monitor

## Caratteri, immagini raster e unità di misura - II

---

- I monitor a multiscansione permettono di ottenere un range di risoluzioni diverse, in contrapposizione ai primi monitor che permettevano una risoluzione unica
- E' tutttavia molto diffusa l'idea che i monitor abbiano una risoluzione di 72 dpi.
- L'origine di questa erronea convinzione sta nel fatto che effettivamente i primi monitor a colori Apple utilizzavano una risoluzione a 72 dpi (la scelta era dovuta al forte legame con il mondo del desktop publishing: in un monitor a 72 dpi il pixel ha la stessa dimensione (misurata sullo schermo) del punto tipografico (pt), unità di misura corrispondente a 1/72 di pollice; quindi le dimensioni di visualizzazione su monitor di un carattere o di un layout corrispondevano alle stesse dimensioni di stampa.
- Oggi questa convenzione non è più rispettata nemmeno da Apple, ma molti continuano a riferirsi al valore di 72dpi come valore tipico per l'output su monitor.

## Il metodo Fahrner

---

- Permette di utilizzare in modo corretto le parole chiave superando i diversi problemi dei browser
- Metodologia
  - Nascondere gli stili dai browser 4.0 attraverso @import (i browser 4.0 non comprendono questa direttiva)
  - Regola di Tantek per servire falsi valori a IE5.X Windows e valori corretti ai browser conformi alle specifiche
  - Regola CSS per proteggere Opera
- Controindicazioni
  - Se l'utente ha diminuito le dimensioni del carattere del browser il testo può risultare poco leggibile:
    - Es. Utenti Windows che impostano le preferenze di lettura a small anzichè medium
    - Es. 2 Utenti Macintosh che reimpostano le preferenze a 12px/72ppi (o comunque qualunque altra impostazione al di sotto di 16px/96ppi)
    - Es. 3 Utenti di Chimera che non hanno come dimensione predefinita 16px/96 ppi
  - In altre parole, le parole chiave utilizzate con la metodologia Fahrner assicurano che il carattere non scenda mai sotto ai 9 px, ma se l'utente ha impostato una dimensione predefinita molto piccola medium sarà molto piccolo e tutte le altre dimensioni saranno più piccole appiattendosi come caso estremo tutte a 9 px.

## Le parole chiave: problemi

---

- Le sette parole chiave non erano implementate correttamente nei primi browser:
  - **Netscape 4** le ignora completamente
  - **Netscape 4.5 e superiori e IE3** le interpretano a dimensioni illeggibili (Netscape 4.5 segue le raccomandazioni W3C stabilendo che ogni dimensione successiva doveva essere 1.5 volte la precedente, ma il fattore era troppo elevato; successivamente il W3C modificò la propria raccomandazione portando il fattore ad un più ragionevole 1.2; in CSS2.1 il fattore di scala varia per fornire una maggiore leggibilità, vedi slide successiva)
  - **IE4 Windows, IE5 Windows e IE 5.5 Windows** implementano le parole chiave, mappando le sette parole chiave sulle sette dimensioni dei font di Netscape; ma associano la parola chiave **small** anziché la parola chiave **medium** alla grandezza di font predefinita (il ragionamento alla base di questa diversa implementazione è che small è la terza parola chiave dal basso (xxsmall - xsmall - small > ...), come lo è font size = 3, che corrisponde alla grandezza del carattere di default). **Quindi specificando in un foglio di stile la dimensione a small questi browser visualizzeranno il carattere con una dimensione maggiore rispetto ai browser che rispettano le specifiche.**
  - **IE 5 Macintosh, Netscape 6 e successivi, Mozilla e IE 6** interpretano correttamente le parole chiave

## Caratteri, immagini raster e unità di misura - III

---

- **DPI** (dot per inch) = numero di punti di inchiostro per unità di lunghezza
- Unità di misura utilizzata per dispositivi di stampa che devono presentare informazioni su una superficie fisica (un foglio di carta). Dà un'indicazione del numero di punti stampabili dal dispositivo per una certa dimensione lineare fisica.
- **IMPORTANTE:** per rappresentare un pixel su carta vengono in genere utilizzati  $n$  punti di inchiostro, aggregati secondo la tecnica del dithering

## Breve storia del font su web - I

---

- Tim- Berners-Lee non incluse nessun controllo tipografico: i designer usavano scorrettamente tag strutturali per ottenere effetti di presentazione
- 1995 – Siti commerciali in crescita: Netscape introduce <font> con specifica della dimensione e Microsoft aggiunge la specifica del tipo di carattere (face). In questo periodo per Windows la dimensione di base predefinita è pari a 16 px/96ppi, per Macintosh è pari a 12px/72 ppi.
- Le prime implementazioni CSS causarono problemi. Molti designer scelsero di usare dimensionamento in pt (punto tipografico) per definire i caratteri, con il risultato di produrre caratteri illeggibili nel mondo Macintosh che utilizza un numero di pixel inferiori nella conversione (es 7pt per Windows risulta in 9 pixel, su Macintosh risulta in 7 pixel).

Si potevano utilizzare in alternativa em e percentuali, ma l'applicazione era condizionata dalle carenti implementazioni di CSS e dalle diverse dimensioni dei caratteri di default a cui tali unità di misura fanno riferimento.

L'unica misura che funzionava allo stesso modo era il pixel (ma questo non era ridimensionabile su IE per Windows – causando problemi di accessibilità; per quanto riguarda IE per Windows solo dalle versione 7 è possibile il ridimensionamento).

## Breve storia del font su web - II

---

- 1999 – Standardizzazione della dimensione predefinita; 16px/96ppi, adottata da Netscape 6+, Mozilla, IE5+/Macintosh, IE/Windows.
- Vantaggio -> Stessi risultati con punti, em, percentuali e parole chiave fino al momento in cui utente non avesse cambiato le preferenze.
- Una parte di utenti (coloro che utilizzavano interfacce impostate a 72ppi), disturbati da dimensioni di caratteri percepite come troppo grandi, riportarono i browser alla vecchia impostazione. I cambi di impostazione ebbero l'effetto di produrre risultati diversi nei diversi browser e piattaforme, accentuati anche da alcuni problemi di implementazione dei browser più diffusi (problemi per l'utilizzo di em, percentuali e (in maniera meno accentuata) parole chiave).
- Ancora una volta il pixel rimaneva l'unica unità di misura che permetteva risultati omogenei (nel senso che un pixel è un pixel su tutte le piattaforme). Il limite di questo approccio era ancora legato alle diverse dimensioni effettive con cui i font in pixel venivano visualizzati nei diversi sistemi a causa delle diverse risoluzioni di visualizzazione.

## Breve storia del font su web - III

---

- A questo va aggiunto che la scelta di standardizzazione del 1999 non fu adeguatamente comunicata, con il risultato che un numero considerevole di designer continuarono a produrre siti nei quali si combinavano riconoscimento del browser e conseguente associazione di un foglio di stile specifico (spesso basato su punti tipografici), quando la standardizzazione permetteva ora di ottenere risultati uguali con un unico foglio di stile uguale per tutti.
- Alcuni browser emergenti (Chimera, Safari) hanno complicato ulteriormente la situazione stabilendo una dimensione predefinita diversa, pari a 14 pixel.

## Nota sulla visualizzazione dei caratteri tipografici nelle piattaforme Mac e Windows

---

- Un font di 12 pixel visualizzato su un display a 72 ppi (standard per i primi Macintosh) mostra un carattere alto circa 0.4 cm. Lo stesso accade per un font di 16 pixel visualizzato su un display a 96 ppi (standard per i sistemi Windows).  
Quindi in teoria tutto dovrebbe funzionare dal punto di vista della dimensione effettiva misurabile sulla superficie del display, ma la qualità con cui il font viene presentato varia a seconda del numero dei pixel impiegati, fino a situazioni in cui i pixel non bastano più a garantire la leggibilità.
- L'allineamento delle due piattaforme a 16pixel/96ppi si propose di risolvere questo problema, aumentando il numero di pixel per descrivere il font per Macintosh allineandolo a Windows (e quindi raggiungendo anche l'obiettivo di una uniformità di partenza).
- Ma gli utenti Macintosh con schermo a 72 ppi (anziché 96 ppi) trovarono la cosa sgradevole, dettata dal fatto che ora si vedeva a 72 ppi un font pensato per essere visualizzato con una risoluzione più elevata. Quindi per questi utenti fu naturale diminuire la dimensione del font per compensare. Questo accadde non solo per utenti Macintosh, ma anche che per gli utenti Windows che avevano interfacce impostate a 72 dpi.