

# Introduzione ad HTML

## prima parte

---

Fabio Vitali



# Introduzione

Oggi esaminiamo in breve:

- ◆ La storia e le caratteristiche di HTML
- ◆ La struttura di un documento HTML
- ◆ I tag di testo e blocco e i link
- ◆ Tabelle
- ◆ Entità predefinite

# Introduzione ad HTML (1)

Il linguaggio HTML è un tipo di documenti SGML (*esiste un DTD di HTML*).

HTML viene usato dai browser WWW per visualizzare documenti ipertestuali. Tramite HTML è possibile realizzare documenti con una semplice struttura, con aspetti grafici anche sofisticati, con testo, immagini, oggetti interattivi e connessioni ipertestuali ad altri documenti

Fino ad oggi i browser si sono preoccupati poco della correttezza sintattica o strutturale dei documenti HTML. Questo significa che tra un documento HTML *visualizzabile da un browser WWW* ed un documento HTML *corretto* esistono differenze anche sensibili.

# Introduzione ad HTML (2)

È normale associare un significato strutturale agli elementi definiti in un DTD. HTML associa anche significati grafici agli elementi che definisce. Cioè dà istruzioni più o meno precise su come rendere graficamente gli elementi che definisce.

Questo porta anche ad abusi della resa grafica che a noi interessano poco. Per noi la resa grafica finale, in assenza di linguaggi di stile appropriato, è secondaria.

HTML non forza strutture gerarchiche nei suoi documenti. Inoltre in HTML i vincoli di contenimento tra elementi sono pochi e piuttosto ovvi. I browser WWW sono ancora più lassisti, a questo proposito, del linguaggio stesso.

# Storia di HTML (1)

HTML è esistito in varie versioni dal 1989 ad oggi:

- ◆ 0.9 (?): il linguaggio di HTML disponibile sul browser WWW aveva strutture di base per intestazioni, paragrafi e stili base, più ovviamente il tag A che ne costituiva la novità.
- ◆ 1.0 (1992): I primi browser shareware e freeware al mondo (il più importante di questo periodo fu Arena, ma esistevano anche MacWeb ed altri) implementavano alcune versioni di HTML leggermente diverse. Tra queste ebbe un certo successo la proposta HTML+. Viene introdotto il tag IMG e il supporto per il GIF.
- ◆ 2.0 (1994): La prima versione veramente nota di HTML. Questa è quella implementata su Mosaic, da cui deriverà Netscape. E' la prima versione ad essere formalizzata su un DTD SGML, invece che ispirarsi vagamente ad SGML. Introduce i form.

# Storia di HTML (2)

- ◆ 3.0 (1995): Questa versione non è mai stata ufficialmente approvata. Durante la sua discussione vennero proposte molte aggiunte. Alcune di queste vennero implementate prima di raggiungere un consenso (tabelle), altre (ad es. supporto per la matematica) mai prese in considerazione.
- ◆ 3.2 (1997): Quando divenne chiaro che i browser non avrebbero supportato tutto il 3.0, si lavorò per generarne un sottoinsieme su cui ci fosse consenso, e che tenesse conto delle aggiunte proprietarie dei vari produttori. Questa versione include tabelle, applet, script e altre migliorie, ma non i frame, sebbene Netscape e Microsoft le avessero già implementate fin dal 1995.
- ◆ 4.0 (1997): supporto per l'internazionalizzazione, per gli style sheet, per i frame, tabelle molto più ricche, il tag OBJECT, ecc.
- ◆ 4.01 (dic. 1999): contiene alcune minime variazioni e correzioni.

# Storia di HTML (3)

- ◆ XHTML1.0: Nel 1998 parte l'iniziativa di riformulare HTML come applicazione di XML, piuttosto che di SGML.

Il 26 gennaio 2000 esce la prima recommendation del W3C, XHTML 1.0, che è una semplice riformulazione di HTML 4 in termini di XML, senza nessuna introduzione di nuove forme.

XHTML però identifica anche un percorso di evoluzione verso la creazione di una famiglia di tipi di documenti che estendano localmente o semplifichino XHTML per una vasta gamma di usi e device.

# HTML 4

Una prima recommendation è del 18 dicembre 1997, una revisione del 24 aprile 1998. Il 24 dicembre 1999 esce HTML 4.01, che contiene alcuni cambiamenti editoriali e precisazioni.

La diffusione dei fogli di stile ha liberato i web designer dall'obbligo di forzare i tag HTML ad assumere uno scopo tipografico e decorativo.

Però alcuni tag esistono solo allo scopo di fornire istruzioni tipografiche di base. Per compatibilità col passato, tuttavia, si è deciso di mantenere il supporto per alcuni elementi che non hanno più senso con l'uso dei fogli di stile.



# I DTD di HTML 4

Per esigenze di compatibilità con il passato, HTML 4 è composto di tre strutture (DTD) alternative:

- ◆ Transitional DTD (detto anche loose): contiene l'intero linguaggio ammesso per HTML, inclusi quegli elementi "deprecati" che vengono mantenuti per compatibilità col passato.
- ◆ Strict DTD: contiene i soli elementi di HTML che non vengono influenzati dall'uso degli style sheet (ma sono escluse le tabelle, che non sono gestite da CSS).
- ◆ Frameset DTD: un semplicissimo DTD per quei documenti in cui al posto di BODY si usano i tag dei frame.

# Criteri di sviluppo (1)

HTML 4.0 estende HTML 3.2 con meccanismi per i fogli di stile, gli script, i frame, oggetti embedded, criteri di internazionalizzazione, tabelle più ricche e miglioramenti ai form.

Questi sono i criteri di sviluppo più importanti:

- ◆ **Internazionalizzazione** (*Internationalization* o *I18N*): l'adozione dei meccanismi necessari per il supporto di linguaggi e notazioni di tutto il mondo, e per la creazione di documenti contenenti linguaggi misti.
- ◆ **Accessibilità**: l'adozione dei meccanismi necessari per il supporto delle esigenze degli utenti con limitazioni fisiche (visive, uditive, etc.).

# Criteri di sviluppo (2)

- ◆ **Tabelle sofisticate:** l'adozione di meccanismi necessari per creare tabelle ancora più sofisticate delle precedenti.
- ◆ **Documenti composti:** l'adozione dei meccanismi necessari per inserire (embed) in maniera generalizzata oggetti di ogni possibile media all'interno di una pagina HTML.
- ◆ **Style sheet:** l'adozione di meccanismi per specificare in maniera precisa e sofisticata la resa tipografica di una pagina senza appesantire la gestione del contenuto.
- ◆ **Scripting:** l'adozione dei meccanismi necessari per realizzare sul client degli oggetti attivi, in grado di eseguire computazioni locali (ad esempio, per pre-verificare la correttezza delle informazioni inserite in un form).

# Usare HTML 4 (1)

## Separazione di struttura e presentazione

- ◆ Via via che HTML tende ad assomigliare al suo antenato SGML, molti dei suoi aspetti presentazionali vengono sostituiti o affiancati da altri meccanismi. Questo porta in particolare a "deprecare" gli aspetti più presentazionali di HTML (ad esempio, l'elemento FONT), e a proporre meccanismi alternativi, più indipendenti e sofisticati (ad esempio, gli style sheet).

## Considerare l'accessibilità universale al Web

- ◆ Via via che si considera il supporto di un numero maggiore di utenti, maggiori saranno le differenze tra di essi di cui tenere conto: esigenze linguistiche specifiche, minorazioni fisiche, architetture diverse, modalità diverse di fruizione delle pagine richiedono gli autori di considerare appropriatamente le specifiche esigenze di tutti.

# Usare HTML 4 (2)

## Aiutare i browser con il rendering incrementale

- ◆ Immagini, oggetti embedded, tabelle complesse rendono complicato (e quindi lento) il meccanismo di impaginazione dei documenti HTML. L'adozione di misure per permettere la visualizzazione incrementale delle caratteristiche del documento favorisce una velocità percepita di visualizzazione utile per il buon successo delle proprie pagine.

## Internazionalizzazione (I18N)

- ◆ Il supporto per l'internazionalizzazione passa attraverso 4 meccanismi:
  - ✦ Il set di caratteri (UCS -8, 16, 32).
  - ✦ L'attributo lang
  - ✦ L'attributo dir
  - ✦ Il tag BDO (di-directional override)

# Principali aspetti di HTML

- Tipi di dati: colori, lunghezze, URL ecc.
- Struttura dei documenti HTML
- Elementi per testo e liste
- Link, oggetti inclusi ed immagini
- Tabelle
- Fogli di stile e script
- Form
- Frame
- Aspetti presentazionali (font, allineamenti, HR)

# Colori

In molte situazioni (sfondi, caratteri, ecc.) è possibile specificare un colore. HTML fornisce due modi per farlo:

- ◆ **Codice RGB prefissato da un carattere di hash.** Si usano due caratteri esadecimali ciascuno per esprimere la quantità di Rosso, Giallo e Blu del colore (00 significa assenza, FF significa presenza massima). E' possibile descrivere 4096 colori diversi.
- ◆ **Nome:** sono definiti 16 nomi di colori: black, silver, gray, white, maroon, red, purple, fuchsia, green, lime, olive, yellow, navy, blue, teal, aqua. N.B. Microsoft definisce 256 nomi di colori, che IE accetta, ma che Netscape ignora. Questo causa incompatibilità tra i browser.

Tuttavia HTML 4.01 deprecia l'uso esplicito di colori nel documento HTML, e suggerisce di usare dei fogli di stile.

```
<font color="#FF0000">testo in rosso</font>
```

```
<body bgcolor="#008080">sfondo teal</body>
```

```
<td bgcolor="yellow">sfondo giallo</td>
```

# Lunghezze

HTML usa le lunghezze per tutti gli oggetti con una presenza sulla pagina di dimensioni determinate (immagini, tabelle, frame, ecc.) Si usano tre tipi di lunghezze:

- ◆ **Pixel:** una dimensione in punti di schermo.
- ◆ **Percentuali:** una dimensione in proporzione alla dimensione dello schermo
- ◆ **Multi-lunghezze:** una sequenza, separata da virgole, di valori di lunghezza. In questo caso può essere usato il carattere ‘\*’, che divide lo spazio restante in parti uguali (dopo aver tolto le lunghezze esplicite in pixel e percentuale).



# URL assoluti e relativi (1)

Un URL individua una risorsa esterna a cui il documento fa riferimento (es. immagini, link, script, fogli di stile, oggetti multimediali, ecc.)

Un URL assoluto specifica senza default tutte le parti rilevanti dell'indirizzo della risorsa:

`http://www.sito.com/dir1/dir2/name.html#pippo`

- ◆ Protocollo: (`http://`): il protocollo di connessione
- ◆ Nome di dominio: il computer su cui risiede il server Web
- ◆ Nome locale: l'insieme di directory e il nome del documento
- ◆ Frammento: la parte interna al documento in cui voglio individuare la risorsa.

# URL assoluti e relativi (2)

Un URL relativo, invece, non specifica alcune parti a partire da sinistra. Queste parti si assumono specificate per default sulla base del documento visualizzato (detto base)

Se la base è

<http://www.sito.com/dir1/dir2/pippo.html#pluto>

... allora...

- `/dir3/paperino.html` corrisponde a

<http://www.sito.com/dir3/paperino.html>,

- `minnie.html` corrisponde a

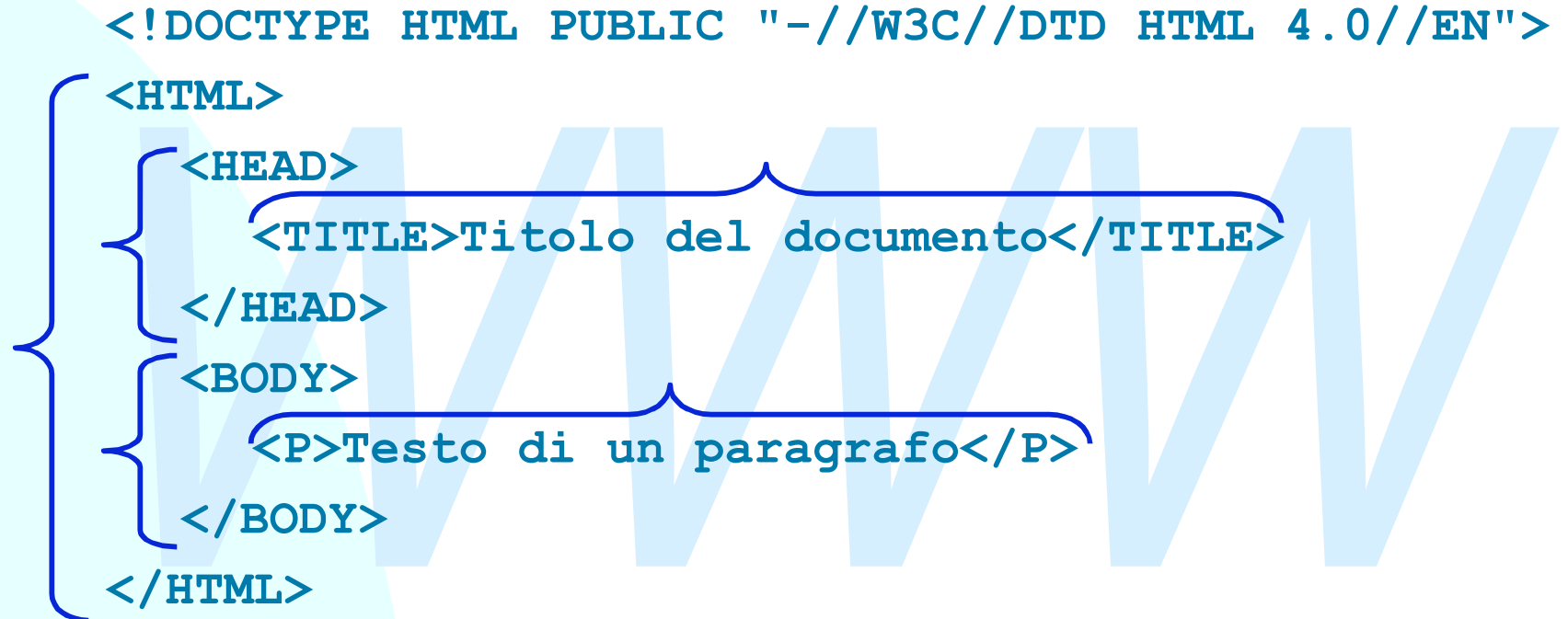
<http://www.sito.com/dir1/dir2/minnie.html>

- `#nonnapapera` corrisponde a

<http://www.sito.com/dir1/dir2/pippo.html#nonnapapera>

# La struttura di un documento HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">  
<HTML>  
  <HEAD>  
    <TITLE>Titolo del documento</TITLE>  
  </HEAD>  
  <BODY>  
    <P>Testo di un paragrafo</P>  
  </BODY>  
</HTML>
```

The diagram illustrates the structure of an HTML document. It shows the following code: 

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">  
<HTML>  
  <HEAD>  
    <TITLE>Titolo del documento</TITLE>  
  </HEAD>  
  <BODY>  
    <P>Testo di un paragrafo</P>  
  </BODY>  
</HTML>
```

Blue curly braces are used to group the code into sections. A large brace on the left groups the entire document structure from `<HTML>` to `</HTML>`. A smaller brace groups the `<HEAD>` section, which includes the `<TITLE>` tag. Another brace groups the `<BODY>` section, which includes the `<P>` tag.

# Tag di struttura

HTML: La radice dell'albero.

- ◆ Attributi: I18N, version (attributo implicito, deprecato)

HEAD: Informazioni globali sull'albero

- ◆ Attributi: I18N

BODY: Il contenuto vero e proprio del documento

- ◆ Attributi: I18N, core (identificazione e stile dell'elemento), events (script associati ad eventi utente o sistema)
- ◆ Attributi deprecati:
  - ✦ background: l'URI di un'immagine di background
  - ✦ bgcolor: il colore dello sfondo
  - ✦ text, link, alink, vlink: il colore del testo, del link non attraversato, del link già attraversato, e del link in corso di attraversamento.

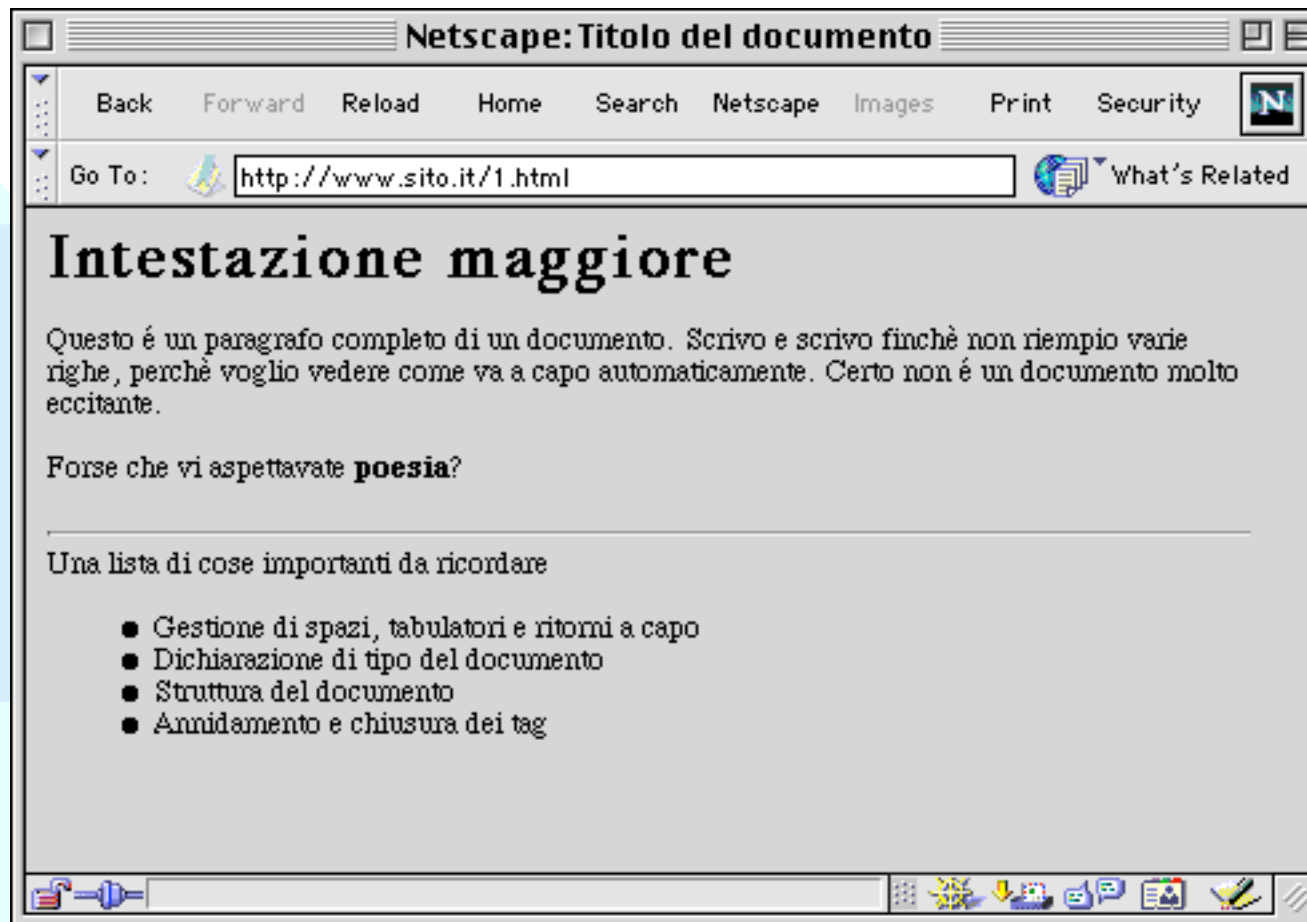
# Un primo esempio di HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
  <HEAD>
    <TITLE>Titolo del documento</TITLE>
  </HEAD>
  <BODY>
    <H1>Intestazione maggiore</H1>
    <P>Questo &acute; un paragrafo completo di un
      documento. Scrivo e scrivo finch&egrave; non
      riempio varie righe, perch&egrave; voglio
      vedere come va a capo automaticamente. Certo
      non &acute; un documento molto eccitante.</P>
    <P>Forse che vi aspettavate <B>poesia</B>? </P>
    <HR>
    <P>Una lista di cose importanti da ricordare:</P>
    <UL>
      <LI>Spazi, tabulatori e ritorni a capo</LI>
      <LI>Dichiarazione di tipo del documento</LI>
      <LI>Struttura del documento</LI>
      <LI>Annidamento e chiusura dei tag</LI>
    </UL>
  </BODY>
</HTML>
```

A seguire: Il risultato su Netscape

21/37

# Il risultato su Netscape



# Tag di carattere (inline)

I tag *fontstyle* specificano informazioni di rendering. Non tutti sono deprecati, ma si suggerisce di usare gli stili.

- ◆ TT (TeleType, font monospaziato, ad es. Courier), I (corsivo), B grassetto, U (sottolineato - deprecato), S e STRIKE (testo barrato - deprecato), BIG, SMALL (testo più grande e più piccolo)

I tag *phrase* (di fraseazione o idiomatici) aggiungono significato a parti di un paragrafo.

- ◆ EM (enfasi), STRONG (enfasi maggiore), DFN (definizione), CODE (frammento di programma), SAMP (output d'esempio), KBD (testo inserito dall'utente), VAR (variabile di programma), CITE (breve citazione), Q (citazione lunga), ABBR e ACRONYM (abbreviazioni ed acronimi), SUP e SUB (testo in apice e in pedice), BDO (bidirectional override), SPAN (generico elemento inline)

# Tag di blocco

I tag di blocco definiscono l'esistenza di blocchi di testo che contengono elementi inline.

- ◆ P (paragrafo), DIV (generico blocco), PRE (blocco preformattato), ADDRESS (indicazioni sull'autore della pagina), BLOCKQUOTE (citazione lunga)

Alcuni blocchi hanno ruolo strutturale

- ◆ H1, H2, H3, H4, H5, H6 (intestazione di blocco)

Alcuni blocchi hanno ruolo di lista:

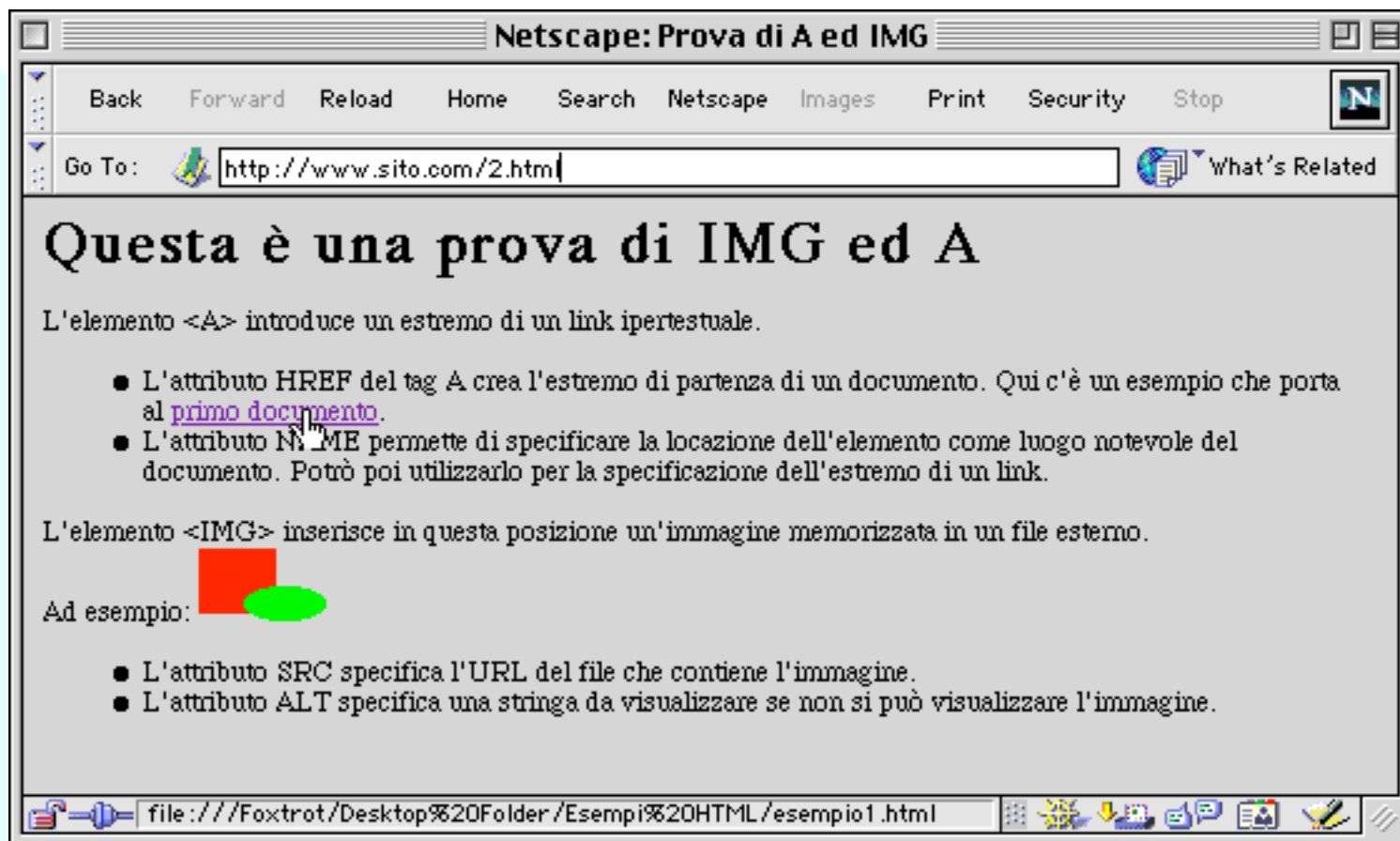
- ◆ UL, OL, DIR, MENU, DL
- ◆ LI (list item per UL, OL, DIR e MENU), DT e DD (definition term e definition data per DL)



# Link ed immagini

```
<BODY>
  <H1>Questa &grave; una prova di IMG ed A</H1>
  <P>L'elemento &lt;A&gt; definisce gli estremi dei link. </P>
  <UL>
    <LI>L'attributo HREF del tag A crea l'estremo di partenza
      di un documento. Qui c'&grave; un esempio che porta
      al <A HREF="1.html">primo documento</A>. </LI>
    <LI>L'attributo NAME specifica quello come luogo di
      destinazione di un link. Ad esempio:
      <A NAME="prova">dell'estremo di un link</A>. </LI>
  </UL>
  <P>L'elemento &lt;IMG&gt; inserisce in questa posizione
    un'immagine posta in un file esterno. <BR>Ad esempio:
    <IMG SRC="esempio.gif" ALT="Un rettangolo ed un'ellisse">
  <UL>
    <LI>L'attributo SRC specifica l'URL del file che contiene
      l'immagine. </LI>
    <LI>L'attributo ALT specifica una stringa da visualizzare
      se non si pu&grave; visualizzare l'immagine. </LI>
  </UL>
</BODY>
```

# Il risultato su Netscape



# Link

I link sono definiti attraverso l'elemento A (anchor interno al documento) e LINK (relazione tra documenti). Alcuni attributi:

- ◆ HREF: specifica una destinazione. Quindi `<A HREF="xx">` è un estremo di partenza. La destinazione è specificata come un URL (Universal Resource Locator). Gli URL sono o assoluti (`"http://www.cs.unibo.it/~fabio/index.html"`) o relativi (`"esempio2.html"`).
- ◆ NAME: specifica un nome utilizzabile come destinazione puntuale di un link. Es: `<A HREF="esempio2.html#prova">click </A>` conduce al frammento `<A NAME="prova">blah blah</A>` del documento "esempio2.html"
- ◆ REL e REV: definiscono un tipo di relazione (e la relazione inversa) che esiste tra un documento e l'altro grazie al link. Esempi: Alternate, Stylesheet, Start, Next, Prev, Contents, Index, Copyright, ecc.
- ◆ TARGET: con i frame, permette di specificare in quale frame avverrà la visualizzazione del link. Esistono alcuni target predefiniti.

# Immagini

Le immagini inline sono definite attraverso l'elemento IMG. Formati tipici: JPEG, GIF, PNG. Alcuni attributi:

HTML: `<IMG SRC="pippo.gif" ALT="Foto di pippo">`

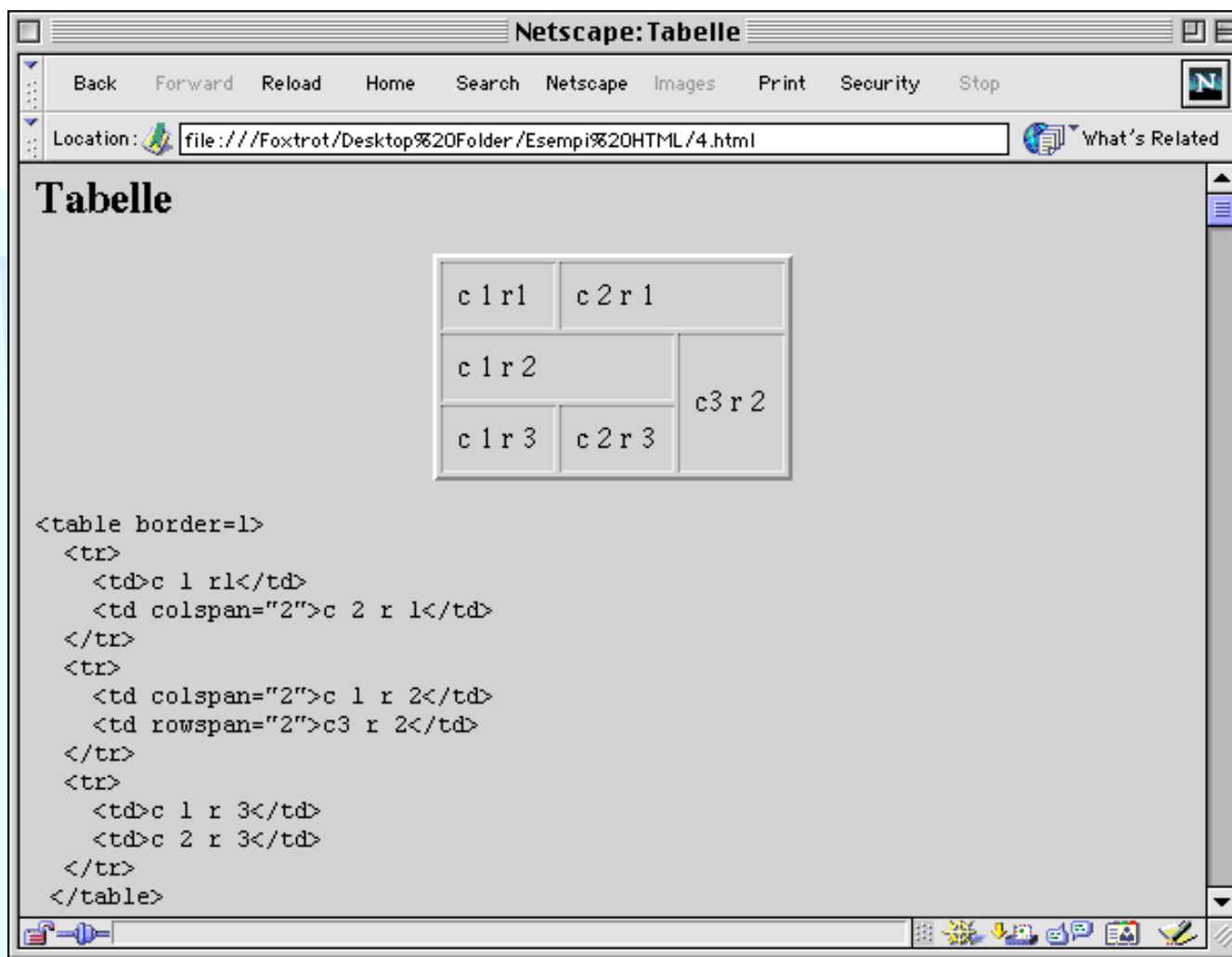
XHTML: ``

- ◆ SRC (obbligatorio): l'URL del file contenente l'immagine.
- ◆ ALT: testo alternativo in caso di mancata visualizzazione dell'immagine
- ◆ NAME: un nome usabile per riferirsi all'immagine
- ◆ USEMAP: indica che l'immagine è una mappa client-side
- ◆ ISMAP: indica che l'immagine è una mappa server-side
- ◆ WIDTH: forza una larghezza dell'immagine.
- ◆ HEIGHT: forza una altezza dell'immagine.
- ◆ ALIGN, BORDER, VSPACE, HSPACE: deprecati, specificano il rendering.

# Tabelle

- Le tabelle vengono specificate riga per riga.
- Di ogni riga si possono precisare gli elementi, che sono o intestazioni o celle normali.
- Una tabella può anche avere una didascalia, un'intestazione ed una sezione conclusiva.
- E' possibile descrivere insieme le caratteristiche visive delle colonne.
- Le celle possono occupare più righe o più colonne.

# Un esempio di tabella



The screenshot shows a Netscape browser window titled "Netscape: Tabelle". The address bar displays the file path: `file:///Foxrot/Desktop%20Folder/Esempi%20HTML/4.html`. The page content is titled "Tabelle" and features a table with the following structure:

c 1 r 1	c 2 r 1	
c 1 r 2		c 3 r 2
c 1 r 3	c 2 r 3	

Below the table, the HTML source code is displayed:

```
<table border=1>
  <tr>
    <td>c 1 r 1</td>
    <td colspan="2">c 2 r 1</td>
  </tr>
  <tr>
    <td colspan="2">c 1 r 2</td>
    <td rowspan="2">c 3 r 2</td>
  </tr>
  <tr>
    <td>c 1 r 3</td>
    <td>c 2 r 3</td>
  </tr>
</table>
```

A seguire: Il codice della tabella

# Il codice della tabella

```
<table border=2 align=center>
  <tr>
    <td>c 1 r1</td>
    <td colspan="2">c 2 r 1</td>
  </tr>
  <tr>
    <td colspan="2">c 1 r 2</td>
    <td rowspan="2">c3 r 2</td>
  </tr>
  <tr>
    <td>c 1 r 3</td>
    <td>c 2 r 3</td>
  </tr>
</table>
```

# I tag delle tabelle (1)

## TABLE

Introduce una tabella, ed è composto di una etichetta, una descrizione di colonne, una intestazione, una conclusione ed un corpo. Attributi:

- ✦ SUMMARY: una descrizione testuale del contenuto della tabella per browser non grafici
- ✦ ALIGN, WIDTH: allineamento e larghezza della tabella rispetto alla pagina. Deprecati
- ✦ CELLPADDING, CELLSPACING: spazio tra cella e testo e tra cella e cella, espresso in pixel.
- ✦ FRAME, RULES, BORDER: tipo di bordo associato alla tabella o alle singole celle.



# I tag delle tabelle (2)

## CAPTION

Introduce una etichetta per la tabella.

## THEAD, TFOOT e TBODY

La testa, la coda e il corpo della tabella. Tutti e tre contengono righe di tabella (elementi TR)

## COLGROUP, COL

Permettono di descrivere l'aspetto di colonne e gruppi di colonne della tabella. Attributi:

- ✦ SPAN: il numero di colonne a cui assegnare le proprietà
- ✦ WIDTH: la larghezza in pixel della colonna
- ✦ ALIGN, VALIGN: allineamento orizzontale e verticale del contenuto della colonna

# I tag delle tabelle (3)

## TR

Introduce una riga di una tabella. Attributi:

- ◆ ALIGN, VALIGN: allineamento orizzontale e verticale del contenuto della tabella

## TD, TH

Una cella di dati o di intestazione. Attributi:

- ◆ WIDTH, HEIGHT (deprecati): larghezza ed altezza della cella
- ◆ NOWRAP (deprecato): esclude la formattazione in paragrafo del contenuto
- ◆ ALIGN, VALIGN: allineamento orizzontale e verticale del contenuto della colonna
- ◆ COLSPAN, ROWSPAN: estende la cella a coprire più colonne o più righe della tabella.

# Entità in HTML

HTML definisce un certo numero di entità per quei caratteri che sono:

- proibiti perché usati in HTML (<, >, &, “, ecc.)
- proibiti perché non presenti nell'ASCII a 7 bit.
- Ad esempio:

◆ amp	&	quot	“
◆ lt (less than)	<	gt (greater than)	>
◆ reg	®	nbsp (non-breaking space)	
◆ Aelig	Æ	Aacute	Á
◆ Agrave	À	Auml	Ä
◆ aelig	æ	aacute	á
◆ agrave	à	auml	ä
◆ ccedil	ç	ntilde	ñ
ecc.			

# Conclusioni

Oggi abbiamo parlato di

- ◆ La storia di HTML
- ◆ Tipi di dati comuni (colori, lunghezze)
- ◆ La struttura di un documento HTML
- ◆ I tag di testo e blocco
- ◆ Link
- ◆ Tabelle
- ◆ Entità predefinite

# Riferimenti

## *Wilde's WWW, capitolo 7*

### Altri testi:

- D. Raggett, A. Le Hors, I. Jacobs, *HTML 4.01 Specification*, W3C Recommendation 24 December 1999, <http://www.w3.org/TR/html401>