

Intro ad HTML

Fabio Vitali
20 marzo 2000



Introduzione

Oggi esaminiamo in breve:

- ◆ La struttura di un documento HTML
- ◆ I tag di testo e blocco
- ◆ Link, oggetti, immagini e mappe
- ◆ Form
- ◆ Tabelle
- ◆ Frame
- ◆ Entità predefinite



La struttura di base 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>
```



Tag di struttura

HTML: La radice dell'albero.

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

HEAD: Informazioni globali sull'albero

- ◆ Attributi: I18N, profile (la locazione di un profilo di metainformazioni sul documento)

BODY: Il contenuto vero e proprio del documento

- ◆ Attributi: I18N, core, events, onload, onunload
- ◆ Attributi deprecati: background, bgcolor, text, link, alink, vlink



Un primo esempio di HTML

```
<BODY>
```

```
<H1>Intestazione maggiore</H1>
```

```
<P> Questo &acute; un paragrafo completo di un documento.  
Scrivo e scrivo finch&grave; non riempio varie righe,  
perch&grave; 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>Gestione di spazi, tabulatori e ritorni a capo</LI>
```

```
<LI>Dichiarazione di tipo del documento</LI>
```

```
<LI>Struttura del documento</LI>
```

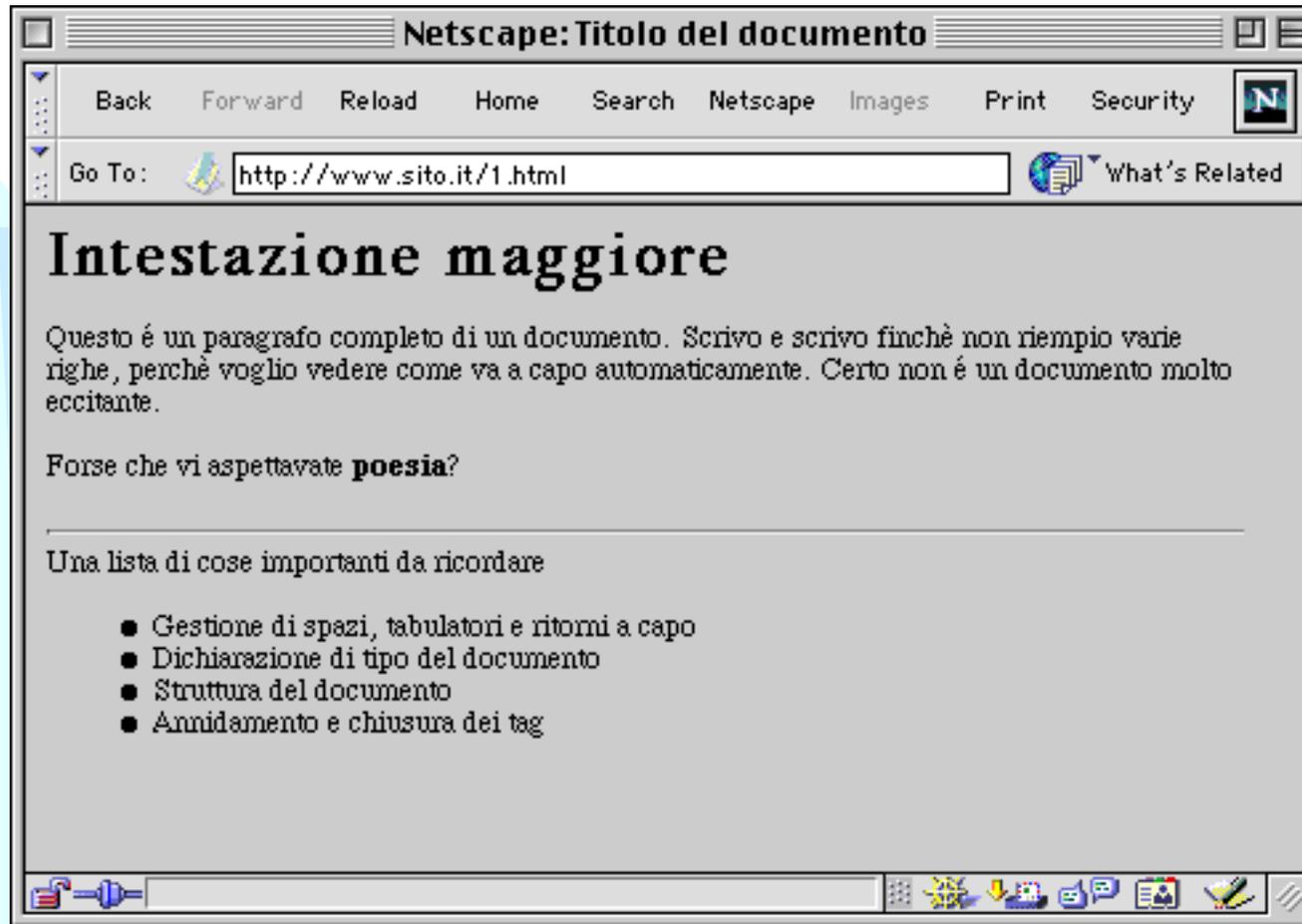
```
<LI>Annidamento e chiusura dei tag</LI>
```

```
</UL>
```

```
</BODY>
```



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)

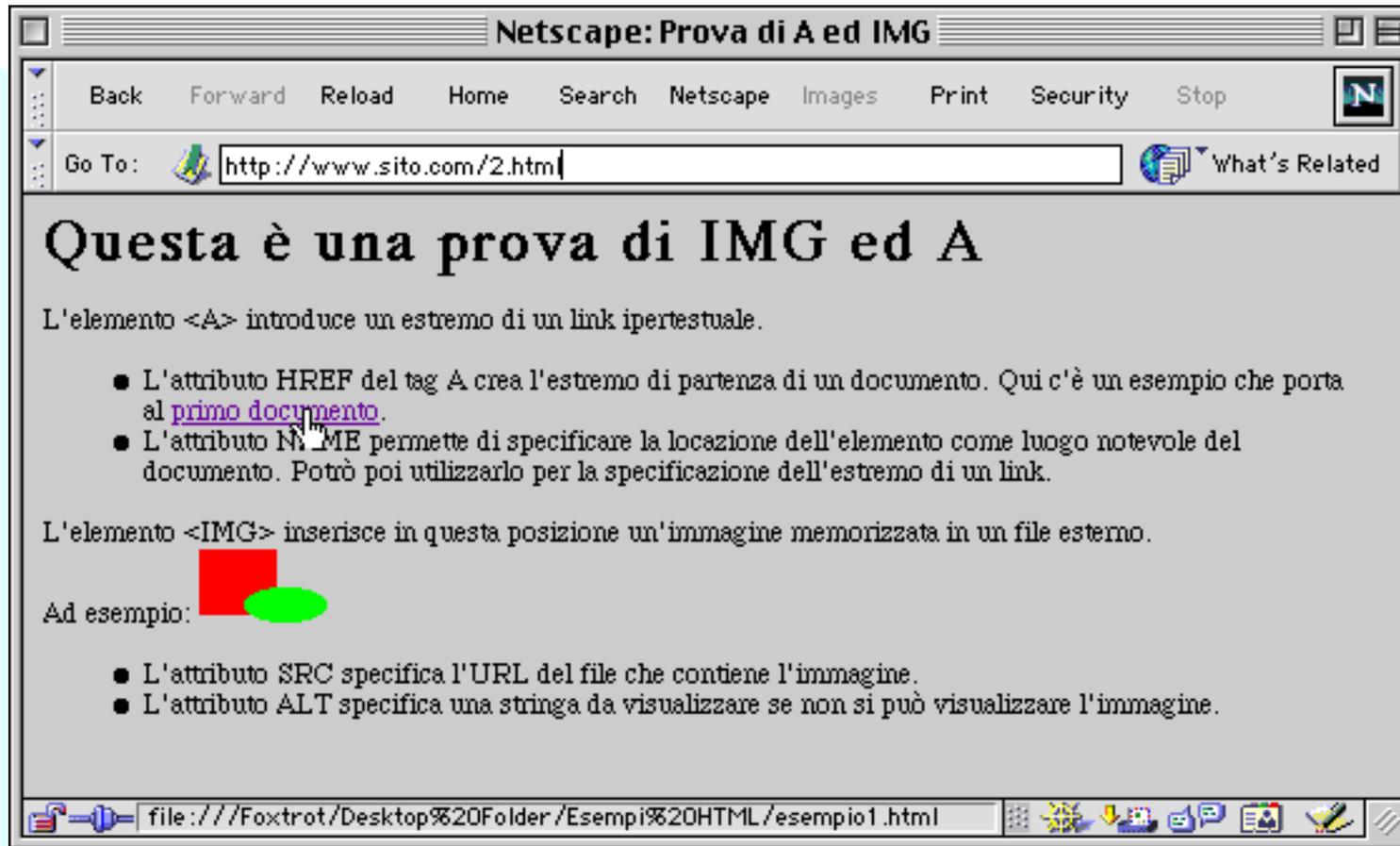


Link ed immagini

```
<BODY>
<H1>Questa &egrave; 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'&egrave; 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&ograve; 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 `` 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: `click ` conduce al frammento `blah blah` 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:

- ◆ 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: Può forzare una larghezza dell'immagine.
- ◆ HEIGHT: forza una altezza dell'immagine.
- ◆ ALIGN, BORDER, VSPACE, HSPACE: deprecati, specificano il rendering.



Mappe, mappe server-side

Un'imagemap (o mappa) è un'immagine (il cui contenuto è ignoto ad HTML) in cui alcune aree vengono specificate e a cui viene associata una destinazione ipertestuale. L'effetto è di aver creato un'immagine in cui, a seconda della locazione in cui si clicca, si va in una destinazione diversa.

- ◆ **Mappe server-side:** deprecated: la lista delle aree è sul server, e si usa il tag IMG con l'attributo ISMAP immerso dentro ad un tag A:

```
<a href="/imagemap/pippo">  
    
</a>
```

Quando l'utente clicca su una parte dell'immagine, il link specificato viene attivato, e il browser manda le coordinate del click. Un'applicazione server-side identifica la destinazione corretta e la spedisce in risposta.



Mappe client side (1)

- **Mappe client-side:** la corrispondenza tra click e area viene fatta sul browser; l'elenco delle zone viene spedito nel documento HTML come tag AREA dentro al tag MAP:

```
<IMG SRC=immagine.gif USEMAP="#pippo">
... altra roba ...
<MAP NAME="pippo">
  <AREA shape="rect" coord="0,0,100,100" href="doc1.html">
  <AREA shape=circle coord="100,50,100" href="doc2.html">
  ... altre aree ...
</MAP>
... altra roba ...
```



Mappe client side (2)

Le mappe client-side si basano su due tag:

- ◆ **MAP:** identifica un blocco di aree che costituiscono la mappa di un'immagine. Attributi:
 - ◆ Name: I valore dell'attributo USEMAP dell'immagine. E' un URI!!!
- ◆ **AREA:** identifica un'area dell'immagine, a cui associare una destinazione ipertestuale. Attributi:
 - ◆ Shape: la forma dell'area (default, rect, circle, poly)
 - ◆ Coords: le coordinate dell'area
 - ◆ Href: la destinazione connessa con quell'area
 - ◆ Nohref: la specifica che l'area non ha destinazioni (senza valore)



Oggetti esterni

L'inserimento nelle pagine HTML di blocchi di dati esterni è stato un problema sentito fin dall'inizio. Il tag IMG voleva fornire supporto per immagini GIF e JPEG.

Per includere oggetti di vario tipo, ogni browser ha inventato le proprie estensioni: la Sun il tag APPLET, la Netscape il tag EMBED, Internet Explorer i tag DYNASRC e BGSOUND, senza contare IMG, svolgono più o meno lo stesso servizio.

In HTML 4 si è introdotto il tag OBJECT per uniformare tutti i tipi di oggetti esterni. Il tag APPLET è mantenuto ma deprecato, mentre il tag IMG è mantenuto per l'alta diffusione che ha.



I tag OBJECT e PARAM

OBJECT

Include un oggetto esterno. Il contenuto del tag viene visualizzato dai browser che non sanno utilizzare l'oggetto.

Attributi:

- ◆ CLASSID, CODEBASE, DATA: l'URL del codice, la base del documento che costituisce l'oggetto incluso, e i dati che bisogna attribuirgli
- ◆ ARCHIVE: la locazione di classi, dati ecc. che risultino più veloci da accedere delle rispettive fonti originarie

PARAM

Un parametro che il browser passa all'oggetto in fase di inizializzazione. Attributi:

- ◆ NAME: il nome dell'attributo
- ◆ VALUE: il valore dell'attributo
- ◆ TYPE: il tipo del valore dell'attributo



Form

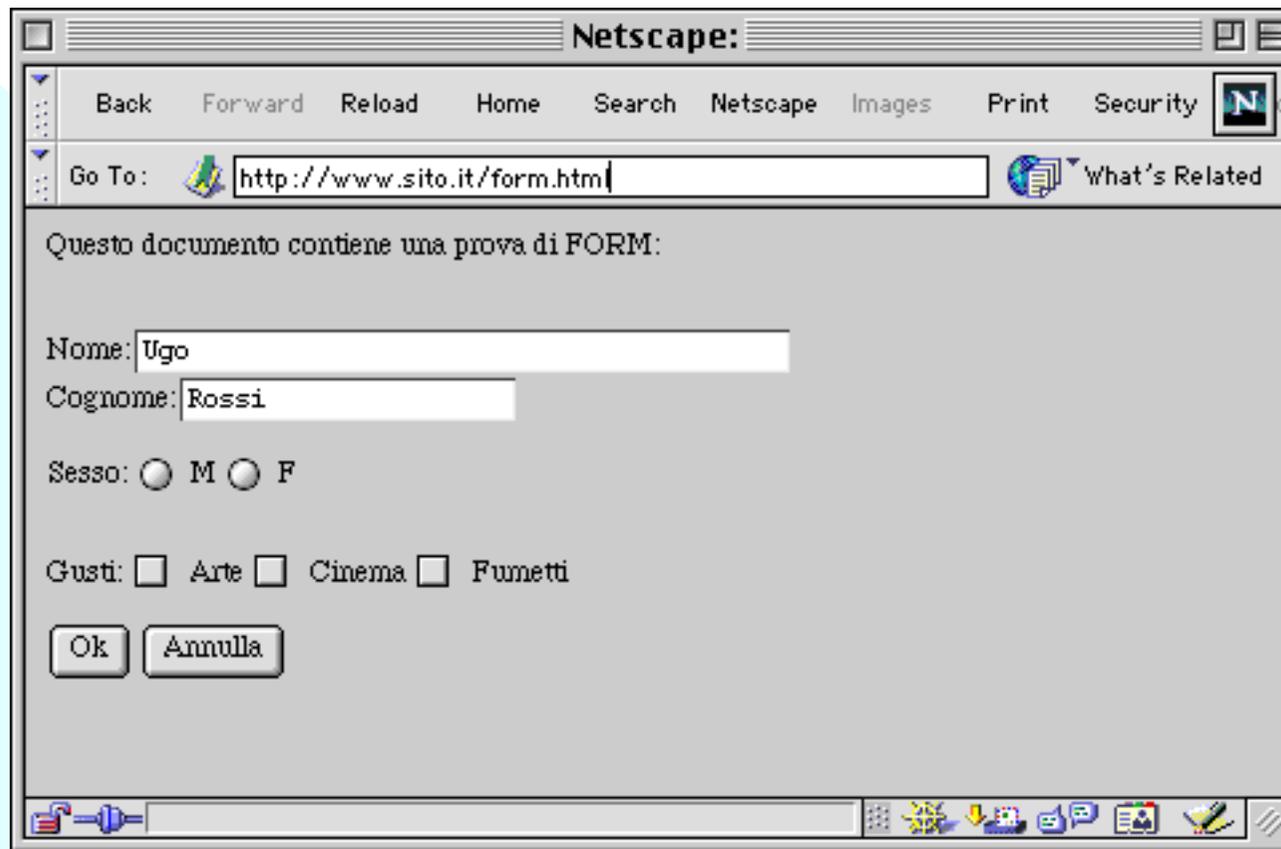
Con i FORM si utilizzano le pagine HTML per inserire valori che vengono poi elaborati sul server. I FORM sono legati al CGI (Common Gateway Interface):

Il browser raccoglie dati con un form dall'utente. Crea una connessione HTTP con il server, specificando una ACTION (cioè un applicazione che funga da destinatario) a cui fare arrivare i dati. Il destinatario riceve i dati, li elabora e genera un documento di risposta, che viene spedito, tramite il server HTTP, al browser.

I controlli tipati e nominati vengono usati per l'inserimento dei dati nei form: campi di inserimento dati, pulsanti, bottoni radio, checkbox, liste a scomparsa, ecc.



Esempio di form



The image shows a screenshot of a Netscape browser window. The title bar reads "Netscape:". The menu bar includes "Back", "Forward", "Reload", "Home", "Search", "Netscape", "Images", "Print", and "Security". The address bar shows "Go To: http://www.sito.it/form.htm". The main content area displays the text "Questo documento contiene una prova di FORM:" followed by a form with the following fields and options:

Nome:

Cognome:

Sesso: M F

Gusti: Arte Cinema Fumetti

Buttons:

The browser's status bar at the bottom contains various icons for navigation and utility.



Il codice della form

```
<P>Questo documento contiene una prova di FORM:</P>
```

```
<FORM METHOD="GET" ACTION="http://www.sito.it/cgi-bin/a.pl">
```

```
<BR>Nome:<INPUT TYPE="text" NAME="Nome" VALUE="Ugo" SIZE=40>
```

```
<BR>Cognome:<INPUT TYPE="text" NAME="Cognome" VALUE="Rossi">
```

```
<P>Sesso: <INPUT TYPE="radio" NAME="Sesso" VALUE="M"> M
```

```
<INPUT TYPE="radio" NAME="Sesso" VALUE="F"> F </P>
```

```
<BR>Gusti:
```

```
<INPUT TYPE="checkbox" NAME="Gusti" VALUE="Arte"> Arte
```

```
<INPUT TYPE="checkbox" NAME="Gusti" VALUE="Cinema"> Cinema
```

```
<INPUT TYPE="checkbox" NAME="Gusti" VALUE="Fumetti"> Fumetti
```

```
<P><INPUT TYPE=submit NAME="Submit" VALUE="Ok">
```

```
<INPUT TYPE=reset NAME="Cancel" VALUE="Annulla">
```

```
</FORM>
```



I tag del form (1)

FORM

Raggruppa tutti i controlli che debbono raccogliere data per un destinatario. Attributi:

- ◆ ACTION: l'URL dell'applicazione server-side che riceverà i dati
- ◆ METHOD: il metodo HTTP che deve essere usato per i dati
- ◆ NAME: un nome univoco per il form
- ◆ ENCTYPE: il metodo di codifica dei dati da usare. Per default si usa `'application/x-www-form-urlencoded'`: i codici non alfanumerici sono sostituiti da `'%HH'` (HH: codice esadecimale del carattere), gli spazi sono sostituiti da `'+'`, i nomi dei controlli sono separati da `'&'`, e il valore è separato dal nome da `'='`).



I tag del form (2)

INPUT

Raccoglie la maggior parte dei tipi di controllo disponibili in un form. Attributi:

- ◆ TYPE: rappresenta il tipo di controllo da istanziare: text, password, checkbox, radio, submit, reset, file, hidden, image, button
- ◆ NAME: il nome che verrà passato all'applicazione server-side insieme al valore inserito
- ◆ VALUE: il valore di default (per i controlli di testo) o selezionato (per i controlli di tipo bottone) che viene presentato all'utente e passato all'applicazione server-side
- ◆ SIZE e MAXLENGTH: dimensione prevista e lunghezza massima (per i testi)
- ◆ DISABLED, CHECKED: per i bottoni, radio e checkbox.



I tag del form (3)

TEXTAREA

Area di inserimento testo. Attributi (oltre a NAME):

- ✦ ROWS: il numero di righe previste
- ✦ COLS: il numero di colonne previste

SELECT, OPTGROUP

Una lista di opzioni e sotto-opzioni. Attributi (oltre a NAME):

- ✦ SIZE: il numero di opzioni visibili
- ✦ MULTIPLE: Vero se la lista permette selezioni multiple.

OPTION

Una opzione in un SELECT o in un OPTGROUP.

- ✦ SELECTED: vero se l'elemento è selezionato all'avvio
- ✦ LABEL: scritta visibile all'utente
- ✦ VALUE: valore passato all'applicazione server-side

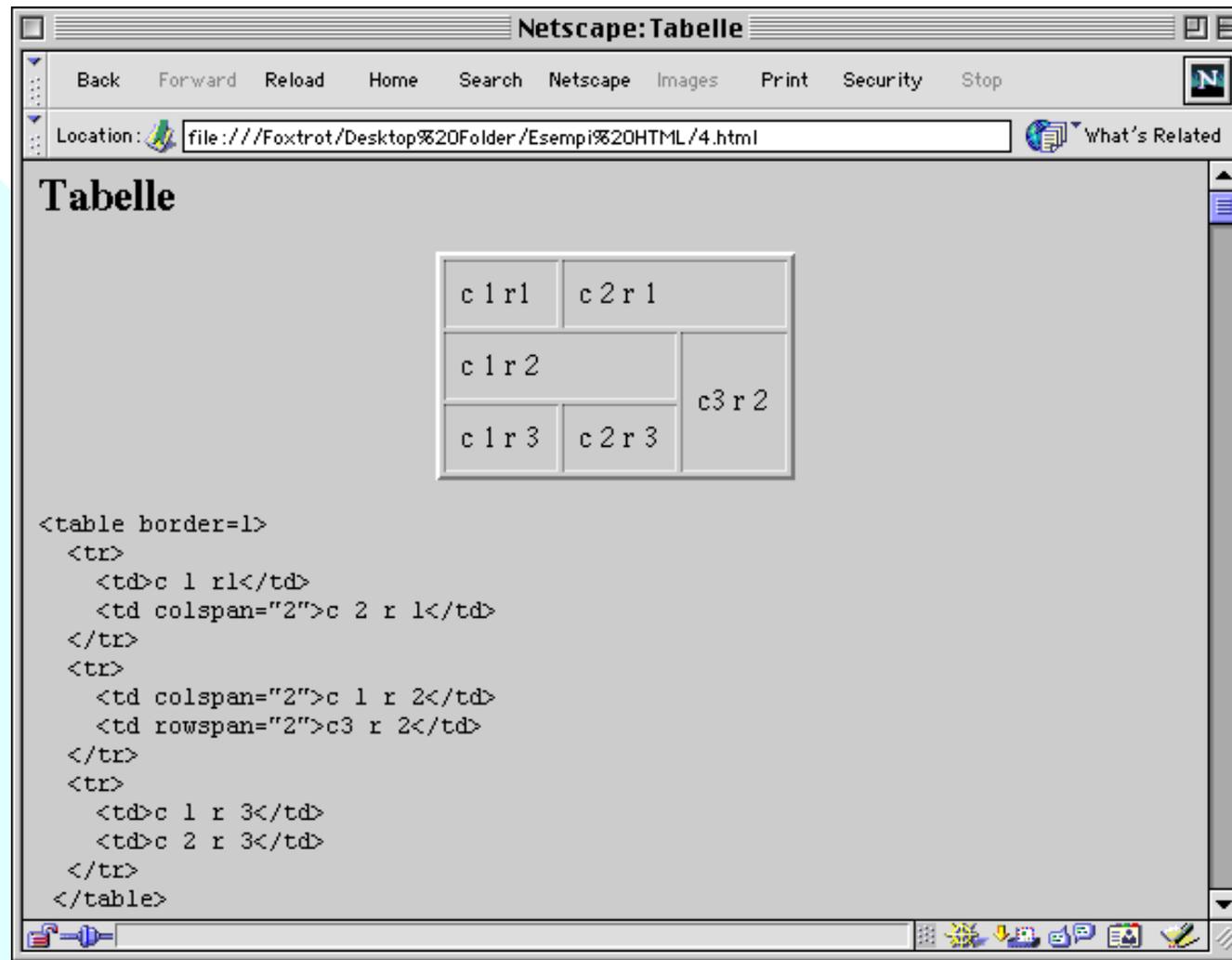


Tabelle

- 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:///Foxtrot/Desktop%20Folder/Esempi%20HTML/4.html`. The main content area shows 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 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>
```



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 colonna. Attributi:

- ✦ ALIGN, VALIGN: allineamento orizzontale e verticale del contenuto della colonna

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.



I tag di HEAD (1)

HEAD contiene delle informazioni che sono rilevanti per tutto il documento. Esse sono:

- ◆ TITLE: il titolo del documento
- ◆ BASE: l'URL che deve essere considerato come base, invece dell'URL del documento, per gli URL relativi
- ◆ LINK: eventuali link su tutto il documento, ad esempio per esplicitare macrostrutture di documenti o per indicare la posizione dei fogli di stile o degli script
- ◆ SCRIPT, STYLE: librerie di script e di stili posizionate nell'intestazione vengono raggruppate in posizione facilmente identificabile.



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.

◆ 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.			



I frame

I frame servono per dividere la finestra in più zone indipendenti ed associare a ciascuna di loro un documento diverso.

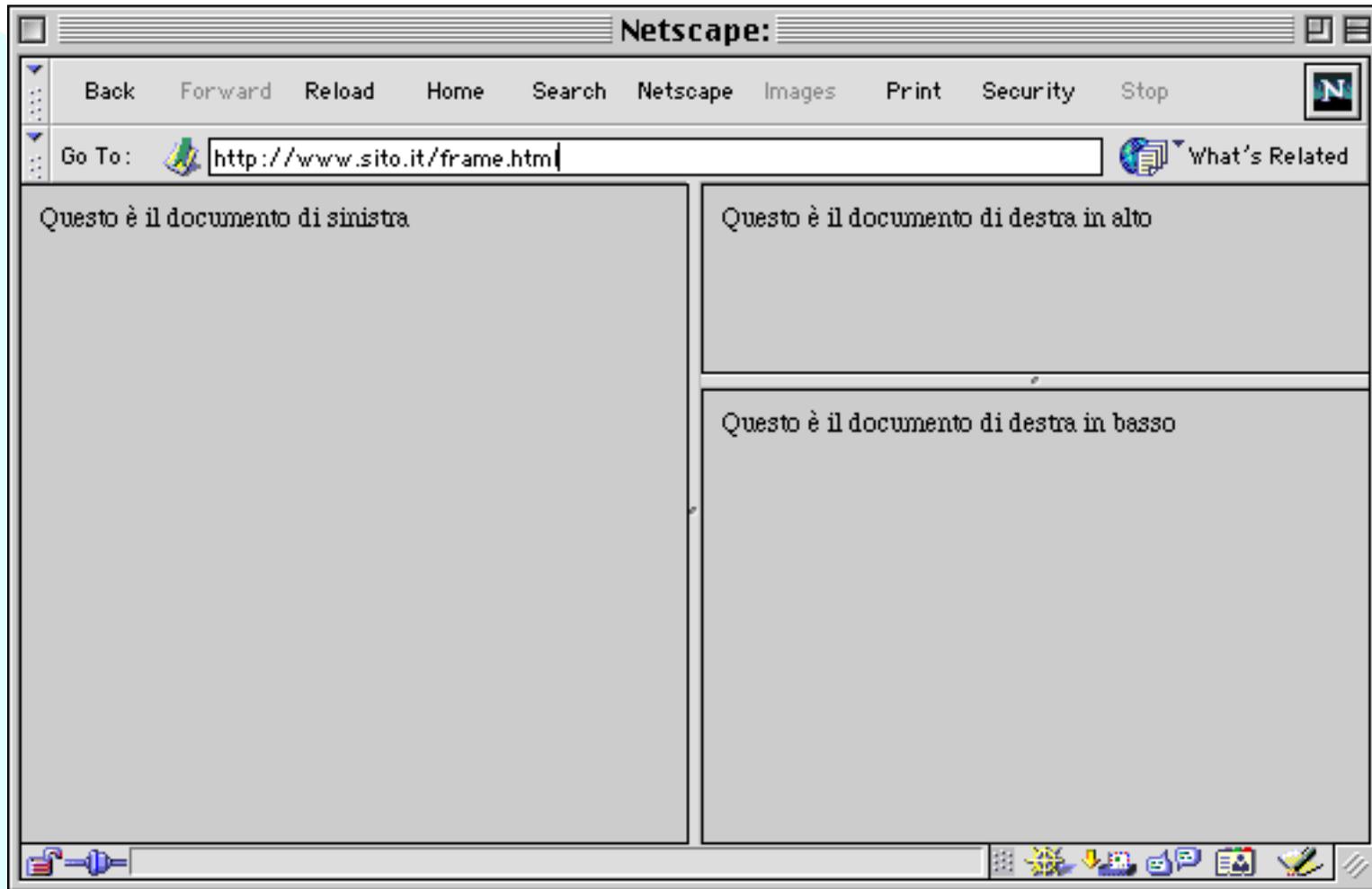
Ogni zona (o frame) ha un nome, che viene usata per specificare in quale zona viene visualizzato la destinazione di un link.

Il tag FRAMESET introduce una serie di frame o verticali o orizzontali. FRAMESET può contenere annidati altri FRAMESET (per cambiare l'orientamento).

La definizione dei link A cambia per rendere possibile precisare il nome di una zona o di una finestra come destinazione del documento.



Un esempio di frame



Il codice del frame

```
<HTML>
  <FRAMESET COLS="50%,*">
    <FRAME SRC="f2.html" NAME="sinistra">
      <FRAMESET ROWS="30%,*">
        <FRAME SRC="f3.html" NAME="destra alto">
        <FRAME SRC="f4.html" NAME="destra basso">
      </FRAMESET>
    </FRAMESET>
  </FRAMESET>
  <NOFRAMES>
  <BODY>
    <P>Testo per browser che non supportano i frame</P>
  </BODY>
</HTML>
```



I tag dei frame (1)

FRAMESET

Introduce un gruppo di frame. Attributi:

- ✦ ROWS, COLS: la disposizione delle righe o delle colonne del frameset. Il valore è una lista di numeri separati da virgole. Ogni numero identifica la dimensione di un frame. Espresso o in pixel, o in percentuale, o con un asterisco (divisione equa del rimanente spazio)

<FRAMESET ROWS="30,30%,*,*">: tre righe, una alta 20 pixel, una alta il 30 per cento dello spazio rimanente, la terza e la quarta si dividono lo spazio rimanente.

NOFRAMES

Blocco di dati da visualizzare nel caso non si sappiano visualizzare i frame.



I tag dei frame (2)

FRAME

Introduce un frame. Attributi:

- ✦ SRC: l'URL del documento da visualizzare nel frame.
- ✦ NAME: il nome del frame, da usare nel target dei link <A>
- ✦ FRAMEBORDER, MARGINWIDTH, MARGINHEIGHT: misure di visualizzazione dei margini dei frame
- ✦ NORESIZE, SCROLLING: controllano il ridimensionamento e la possibilità di scrolling del frame.

IFRAME

Inserisce un frame all'interno di un documento HTML normale (non diviso in frame). Attributi:

- ✦ Gli stessi di frame
- ✦ ALIGN e WIDTH: controllano la posizione e la dimensione del frame rispetto alla pagina.



Conclusioni

Oggi abbiamo parlato di

- ◆ La struttura di un documento HTML
- ◆ I tag di testo e blocco
- ◆ Link, oggetti, immagini e mappe
- ◆ Form
- ◆ Tabelle
- ◆ Frame
- ◆ 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>

