

Altri blocchi e componenti lineari particolari

Inserzioni particolari	443
Riquadri	444
Copia di porzioni del documento	446
Copia di piè di pagina	447
Copia di immagini	447

Dopo la descrizione di elementi di uso abbastanza semplice, conviene concentrare l'attenzione su altri elementi importanti con funzioni speciali.

Inserzioni particolari

Sono disponibili diversi elementi di importanza minore. Si tratta di `'br'`, `'hr'`, `'newpage'`, `'bottompage'`, `'heightrequired'` e `'navlink'`. I primi due emulano gli elementi corrispondenti dell'HTML, interrompendo una riga e inserendo una linea orizzontale rispettivamente.

L'elemento `'newpage'` richiede un salto pagina, se il tipo di composizione lo consente.

L'elemento `'bottompage'` serve per definire un gruppo di blocchi di testo da rappresentare nella parte bassa della pagina, nella composizione per la stampa. In pratica, si usa `'bottompage'` per delimitare informazioni legali nella seconda pagina relativa dei volumi:

```
<tomeheading>Bla bla bla</tomeheading>

<bottompage>
  <p>Copyright &copy; Pinco Pallino...</p>

  <p>Bla bla bla...</p>
</bottompage>
```

L'elemento `'heightrequired'` serve nella composizione per la stampa, a garantire che sia disponibile una certa quantità di spazio (un'altezza minima prima della fine della pagina), in mancanza del quale viene inserito un salto pagina. Questo elemento serve per rimediare agli errori di composizione che compaiono di tanto in tanto.

Tabella u68.2. Inserzioni varie.

Elemento	Descrizione
<code>br</code>	Elemento vuoto che manda a capo il testo, da usare in un contesto lineare.
<code>hr</code>	Elemento vuoto che inserisce una riga orizzontale di separazione. Può essere usato solo tra un blocco e l'altro.
<code>newpage</code>	Elemento vuoto che richiede un salto pagina, se il contesto lo consente. Può essere usato solo tra un blocco e l'altro.
<code>bottompage</code>	Elemento contenente blocchi che richiede una rappresentazione alla base della pagina nella composizione per la stampa.
<code>heightrequired height="altezza"</code>	Elemento vuoto che serve a richiedere espressamente la presenza di una certa quantità di spazio prima della fine della pagina. Si tratta evidentemente di un elemento da usare tra un blocco e l'altro. L'attributo <code>'height'</code> serve a specificare l'altezza minima richiesta.

Elemento	Descrizione
navlink	Elemento contenente una stringa da usare come riferimento alla pagina in cui viene collocato, per la navigazione HTML. Tale riferimento viene inserito in tutte le pagine HTML risultanti dalla composizione.

L'elemento **'navlink'** consente di aggiungere nella composizione HTML un riferimento ipertestuale fisso, in tutte le pagine, allo scopo di raggiungere facilmente la posizione in cui l'elemento stesso viene inserito. Si osservi l'esempio seguente:

```
<h1>
Indice analitico
</h1>

<navlink>indice analitico</navlink>

<printindex index="main">

</index>
```

Si tratta dell'inserimento dell'indice analitico, con l'aggiunta di un riferimento ipertestuale fisso nelle pagine della composizione HTML.

Figura u68.4. Esempio di una pagina HTML prodotta dalla composizione di un sorgente contenente un riferimento aggiuntivo per la consultazione, denominato **'indice analitico'**.

[successivo] [precedente] [inizio] [fine] [indice generale] [indice analitico] [volume] [parte]

Capitolo 3. Standard

Attorno ai sistemi operativi che si rifanno al modello di Unix, si sono definiti degli standard importanti. Vengono qui annotati alcuni riferimenti a proposito di questi standard; tuttavia, si tenga presente che questo è sempre un campo in evoluzione e nulla è definitivo.

3.1 Linguaggio C

Il linguaggio C è quello su cui si basano i sistemi Unix; l'evoluzione dei sistemi Unix va di pari passo con quella del suo linguaggio.

- Brian W. Kernighan, Dennis M. Ritchie, *The C programming language*, prima edizione, Prentice-Hall 1978

Questo è il primo documento che definisce il linguaggio C, per quello che oggi è noto come «K&R C», ovvero il linguaggio C di Kernighan e Ritchie. Di questo libro esiste una seconda edizione, del 1988, rivista secondo le convenzioni in corso di definizione dallo standard ANSI C.

<http://cm.bell-labs.com/cm/cs/cbook/>.

[...]

Dovrebbe essere possibile fare riferimento a questa pagina anche con il nome standard.htm

[successivo] [precedente] [inizio] [fine] [indice generale] [indice analitico]

Riquadri

Alm1 consente di inserire nel documento dei riquadri, a cui si associa una numerazione separata rispetto alle sezioni, che eventualmente possono essere resi fluttuanti nel testo. Questi riquadri sono ottenuti con l'elemento **'object'**.

Figura u68.5. Sintassi semplificata per l'uso dei riquadri.

```
object [id="ancora"] [pos="fixed|float"]
| [sep="none|rule|border"] [split="0|1"]
| [printedfontsize="dimensione"]
|-- [caption]
| `-- testo_lineare
| `-- blocco...
```

L'elemento **'object'** può contenere una didascalia, delimitata dal-

l'elemento **'caption'**, che a sua volta contiene testo lineare; quindi può contenere blocchi di vario tipo, compresi dei blocchi speciali che possono apparire solo al suo interno (come nel caso dell'elemento **'asciart'**). Nell'insieme, il riquadro può essere bordato o meno, può essere fisso o fluttuante, può essere separato tra le pagine oppure può essere un blocco unico. L'esempio seguente mostra un caso tipo:

```
<object sep="border" pos="float" id="a2-mio-riquadro-di-prova"
printedfontsize="0.9em">
<caption>
Riquadro <objectref>. Avvio di un disco esterno e ritardo nel
kernel.
</caption>
<p>Per l'avvio di nanoLinux installato in un disco USB è importante
considerare che tra le opzioni del kernel deve essere prevista la
presenza di <samp>setupdelay</samp>, a cui si assegna un numero
intero che rappresenta un ritardo in secondi prima dell'innesto del
file system principale. Questa opzione è presente solo nel kernel
realizzato per nanoLinux e consente di avviare un disco USB senza
bisogno di un disco RAM. Questa opzione è già presente nel file
<file>boot/grub/menu.lst</file> per le voci riferite a dischi di
questo tipo, con un ritardo di <num>5</num> s.</p>
</object>
```

Il risultato della composizione di questo esempio, si può osservare nel riquadro successivo:

Riquadro u68.8. Avvio di un disco esterno e ritardo nel kernel.

Per l'avvio di nanoLinux installato in un disco USB è importante considerare che tra le opzioni del kernel deve essere prevista la presenza di **'setupdelay'**, a cui si assegna un numero intero che rappresenta un ritardo in secondi prima dell'innesto del file system principale. Questa opzione è presente solo nel kernel realizzato per nanoLinux e consente di avviare un disco USB senza bisogno di un disco RAM. Questa opzione è già presente nel file **'boot/grub/menu.lst'** per le voci riferite a dischi di questo tipo, con un ritardo di 5 s.

Si osservi che le opzioni definite attraverso gli attributi dell'elemento **'object'** non possono convivere sempre in tutte le condizioni. In particolare, se il riquadro viene bordato attraverso l'elemento **'object'** stesso, non è possibile ottenere che il contenuto si possa separare tra le pagine. A questo proposito, si può osservare invece che elementi come **'pre'**, **'verbatimpre'** e **'syntax'**, si possono bordare e separare tra le pagine, ma in tal caso, se vengono inseriti in un elemento **'object'**, questo deve risultare non bordato, lasciando il compito della bordatura agli elementi contenuti.

Tabella u68.9. Riquadri.

Elemento	Descrizione
object [id="ancora"] [pos="fixed float"] [sep="none rule border"] [split="0 1"] [printedfontsize="dimensione"]	Involucro di un riquadro. L'attributo 'id' consente di mettere un'ancora di riferimento; l'attributo 'pos' consente di rendere fluttuante il riquadro; l'attributo 'sep' consente di definire un bordo esterno; l'attributo 'printedfontsize' consente di definire la dimensione del carattere normale da usare nel riquadro; l'attributo 'split' consente di stabilire se il riquadro debba rimanere unito o possa essere suddiviso.
caption	Contiene la didascalia, in forma di testo lineare.

Il corpo del carattere «normale» che si inserisce all'interno di un riquadro, può essere controllato con l'attributo **'printedfontsize'**, oppure, in modo generale, nell'intestazione con un elemento **'printedfontsize'**, come nell'esempio seguente:

```
<head>
  <admin>
  ...
  <printedfontsize type="object">3,5mm</printedfontsize>
  ...
</admin>
...
</head>
```

Se non si indica questa informazione, il carattere viene ridotto leggermente rispetto a quello del corpo normale del testo; se invece si vuole mantenere un carattere uguale a quello del contesto esterno, basta usare l'attributo `'printedfontsize'` indicando una dimensione pari a un quadratone, come nell'esempio seguente:

```
<object sep="border" pos="float" printedfontsize="1em">
...
</object>
```

Non si deve confondere il riquadro costituito dall'elemento `'object'` con la cornice dell'elemento `'frame'`. L'elemento `'frame'` (sezione u0.2) serve per mettere in evidenza una nota breve, mentre l'elemento `'object'` delimita un contenuto autonomo che potrebbe avere una didascalia.

Copia di porzioni del documento

« Alcune porzioni del documento che si scrive con Alml, possono essere copiate in posizioni successive. Ciò si ottiene con gli elementi `'copy'` e `'paste'`.

Gli elementi `'copy'` e `'paste'` possono essere usati sia in un contesto che richiede l'uso di blocchi, sia quando il contesto è lineare; di conseguenza, il loro contenuto può essere fatto di blocchi o di testo lineare.

Tabella u68.12. Copia di porzione del documento.

Elemento	Descrizione
<code>cut cut="area_di_memoria"</code>	Delimita la porzione di documento da accumulare nell'area denominata come indicato con l'attributo <code>'copy'</code> . Il contenuto dell'elemento non appare nella composizione finale.
<code>copy copy="area_di_memoria"</code>	Delimita la porzione di documento da accumulare nell'area denominata come indicato con l'attributo <code>'copy'</code> .
<code>paste paste="area_di_memoria"</code>	Si tratta di un elemento vuoto che inserisce in quel punto quanto accumulato nella voce indicata con l'attributo <code>'paste'</code> .

L'esempio seguente serve ad accumulare alcuni paragrafi in un'area di memoria denominata `'commenti'`:

```
<p>Bla bla bla bla...</p>
<copy copy="commenti">
  <p>Che sciocchezze che si scrivono negli esempi...</p>
  <p>Cosa si può aggiungere di più?</p>
</copy>
<p>Bla bla bla.</p>
```

Nell'esempio successivo, si recupera quanto accumulato in precedenza nell'area di memoria `'commenti'`:

```
<p>Ecco i commenti fatti fino a questo punto:</p>
<frame>
  <paste paste="commenti">
</frame>
```

Si osservi che l'elemento `'copy'` accumula blocchi o testo lineare in memoria, ma questi rimangono visibili normalmente nella composizione finale; al contrario, `'cut'` accumula soltanto, senza mostrare il suo contenuto. Inoltre, si osservi che l'elemento vuoto `'paste'` recupera quanto accumulato fino a quel punto; se in seguito, nel corso del documento si usano ancora gli elementi `'cut'` e `'copy'` per

accumulare nella stessa area di memoria, questa viene espansa ulteriormente e con un successivo elemento `'paste'` si ottiene tutto, anche quanto già incollato in precedenza.

La copia avviene utilizzando il codice del sistema di composizione finale e ciò ha, come effetto collaterale, il pregio di mantenere inalterata la numerazione degli elementi `'object'`, dove i riferimenti automatici, puntano correttamente ai riquadri originali.

Il difetto di questo sistema di copia sta nell'impossibilità di incollare prima ciò che nel documento appare dopo.

Il fatto che gli elementi `'cut'` e `'copy'` possano essere usati indifferentemente in un contesto a blocchi o lineare e che possano contenere indifferentemente questo e quello, implica che il loro utilizzo richieda accortezza. In particolare, non si devono accumulare nella stessa area di memoria dei blocchi assieme a dei componenti lineari; inoltre, l'elemento `'paste'` va usato nel contesto appropriato al contenuto dell'area di memoria che si vuole incollare. Evidentemente, il sistema di controllo SGML non è in condizione di individuare errori di utilizzo di questo tipo.

Copia di piè di pagina

« Può essere interessante la copia di una nota a piè di pagina, per poi riprodurla tale e quale in altre posizioni. Si osservi l'esempio seguente:

```
<p>Bla bla bla bla<copy copy="nota"><footnote>Il classico testo di nessuna importanza</footnote></copy> bla bla bla...</p>
<p>Di nuovo bla bla bla bla<paste paste="nota"> bla bla bla...</p>
```

In questo modo, si intende avere una sola nota a piè di pagina, per entrambe le posizioni: sia la prima volta, dove si vede l'uso dell'elemento `'footnote'`, sia dopo, quando viene incollato il contenuto dell'area di memoria `'nota'`. In pratica, la nota che si legge alla fine del capitolo è una sola e i riferimenti alla nota, sono sempre allo stesso numero di nota.

Copia di immagini

« È possibile utilizzare gli elementi `'cut'`, `'copy'` e `'paste'` anche per accumulare delle immagini che si inseriscono con gli elementi `'*img'`. Il vantaggio di questo sta nel fatto che nella composizione finale, viene prodotto un solo file contenente l'immagine stessa. Pertanto, ciò può essere molto utile per rappresentare delle icone ricorrenti nel documento.

