

Sgmltexi: contenuti



Paragrafi	1458
Indici e riferimenti incrociati	1458
Delimitazione di parole e di frasi	1463
Delimitazione di blocchi di testo	1465
Elenchi e tabelle	1468
Inserzioni	1472
Definizioni	1473
Codice condizionato e codice letterale in base alla composizione 1478	
Problemi	1481

Dopo la struttura generale, il sorgente Sgmltexi si articola generalmente in elementi che possono essere classificati sommariamente in blocchi e in testo interno a un blocco. Nei DTD comuni si utilizzano frequentemente le entità parametriche ‘%**block**;’ e ‘%**inline**;’, per definire questi due grandi raggruppamenti. Nel DTD di Sgmltexi si usa la stessa convenzione e in questo senso vanno interpretate tali sigle nelle tabelle riassuntive.

A titolo di esempio, un blocco è qualcosa di simile a un paragrafo, un elenco, una tabella; un elemento interno alla riga è fatto per contenere del testo, eventualmente assieme a delle enfatizzazioni di qualche genere. Di solito, anche se questo fatto non può valere in generale, un elemento interno alla riga è fatto per contenere testo

o altri elementi dello stesso genere; al contrario, un elemento che costituisce un blocco, può contenere altri blocchi, oppure del testo interno alla riga.

Il DTD di Sgmltexi non prevede elementi che possano contenere testo interno alla riga o blocchi a scelta, come accade invece nell'HTML.

Paragrafi

«

I blocchi di testo più comuni sono dei paragrafi, delimitati dall'elemento '**p**', il quale può apparire con un rientro iniziale o meno, a seconda dell'uso dell'attributo '**indent**'. I paragrafi, compresi quelli centrati che si ottengono con l'elemento '**center**', contengono testo o altri elementi interni alla riga.

Tabella u91.1. Paragrafi con Sgmltexi.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
p	%inline;	Blocco di testo elementare, riconducibile al paragrafo.
<i>indent</i>	Attributo	Rientro prima riga: ' on ', ' off '.
center	%inline;	Blocco di testo centrato: '@ center '.

Indici e riferimenti incrociati

«

Sgmltexi mette a disposizione diversi elementi il cui scopo è quello di permettere delle inserzioni per generare degli indici o

dei riferimenti incrociati, riproducendo i comandi equivalenti di Texinfo.

Le voci degli indici analitici vengono inserite attraverso un gruppo di elementi vuoti: ‘**cindex**’, ‘**findex**’, ‘**vindex**’, ‘**kindex**’, ‘**pindex**’, ‘**tindex**’ e ‘**userindex**’. Tutti questi elementi hanno lo stesso attributo ‘**entry**’, che serve a specificare la voce da inserire nell’indice relativo. In particolare, l’elemento ‘**userindex**’ ha in più l’attributo ‘**name**’ per specificare l’indice al quale si vuole fare riferimento.

Questi elementi possono essere usati solo dopo la dichiarazione di una sezione (un titolo di qualunque tipo, dal tomo in giù), ma prima del testo normale che ne seguirebbe. Per esempio così:

```
<h1>IP protocol history</h1>
<cindex entry="IP protocol">
<cindex entry="history">

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

La tabella u91.3 riassume brevemente l’uso di questi elementi.

Tabella u91.3. Voci degli indici analitici.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
cindex	Vuoto	Voce dell’indice analitico normale.
<i>entry</i>	Attributo	Voce da inserire nell’indice.
findex	Vuoto	Voce dell’indice analitico delle funzioni.
<i>entry</i>	Attributo	Voce da inserire nell’indice.
vindex	Vuoto	Voce dell’indice analitico delle variabili.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<i>entry</i>	Attributo	Voce da inserire nell'indice.
kindex	Vuoto	Voce dell'indice analitico dei tasti premuti.
<i>entry</i>	Attributo	Voce da inserire nell'indice.
pindex	Vuoto	Voce dell'indice analitico dei programmi.
<i>entry</i>	Attributo	Voce da inserire nell'indice.
tindex	Vuoto	Voce dell'indice analitico dei tipi di dati.
<i>entry</i>	Attributo	Voce da inserire nell'indice.
userindex	Vuoto	Voce di un indice analitico definito dall'utilizzatore.
<i>entry</i>	Attributo	Voce da inserire nell'indice.
<i>name</i>	Attributo	Sigla identificativa dell'indice definito dall'utente.
printindex	Vuoto	Inserisce l'elenco delle voci dell'indice specificato.
<i>name</i>	Attributo	Sigla identificativa dell'indice.

Ogni indice analitico si distingue in base a una sigla di due lettere. Gli indici analitici già previsti da Texinfo hanno una sigla fissa, mentre tutte le altre combinazioni possono essere usate per gli indici stabiliti dall'utilizzatore. La tabella u91.4 riassume le sigle degli indici standard, la cui conoscenza è necessaria per poter usare correttamente l'elemento '**printindex**' allo scopo di riprodurre l'elenco

dell'indice relativo.

Tabella u91.4. Sigle identificative degli indici analitici standard.

Sigla	Descrizione
cp	Indice analitico normale.
ky	Indice analitico dell'uso della tastiera.
pg	Indice analitico dei programmi.
fn	Indice analitico delle funzioni.
vr	Indice analitico delle variabili.
tp	Indice analitico dei tipi di dati.

Gli elementi utilizzati per realizzare dei riferimenti incrociati sono vuoti e sono sempre interni alla riga di testo. Tutte le informazioni necessarie sono passate attraverso attributi. Dal momento che questi elementi rispecchiano fedelmente i comandi equivalenti di Texinfo, viene mostrata solo la tabella u91.5, senza entrare nel dettaglio del significato di ognuno di loro.

Tabella u91.5. Riferimenti incrociati.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
anchor	Vuoto	Comando '@anchor' di Texinfo.
<i>id</i>	Attributo	Stringa di identificazione dell'ancora.
xref	Vuoto	Comando '@xref' di Texinfo.
<i>id</i>	Attributo	Nodo o ancora a cui si fa riferimento.
<i>name</i>	Attributo	Nome del riferimento.
<i>title</i>	Attributo	Titolo o argomento a cui si fa riferimento.
<i>info</i>	Attributo	Nome del file Info.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<i>ptitle</i>	Attributo	Titolo dell'edizione stampata.
ref	Vuoto	Comando '@ref' di Texinfo.
<i>id</i>	Attributo	Nodo o ancora a cui si fa riferimento.
<i>name</i>	Attributo	Nome del riferimento.
<i>title</i>	Attributo	Titolo o argomento a cui si fa riferimento.
<i>info</i>	Attributo	Nome del file Info.
<i>ptitle</i>	Attributo	Titolo dell'edizione stampata.
pxref	Vuoto	Comando '@pxref' di Texinfo.
<i>id</i>	Attributo	Nodo o ancora a cui si fa riferimento.
<i>name</i>	Attributo	Nome del riferimento.
<i>title</i>	Attributo	Titolo o argomento a cui si fa riferimento.
<i>info</i>	Attributo	Nome del file Info.
<i>ptitle</i>	Attributo	Titolo dell'edizione stampata.
inforef	Vuoto	Comando '@inforef' di Texinfo.
<i>id</i>	Attributo	Nodo o ancora a cui si fa riferimento.
<i>name</i>	Attributo	Nome del riferimento.
<i>info</i>	Attributo	Nome del file Info.
uref	Vuoto	Comando '@uref' di Texinfo.
<i>uri</i>	Attributo	Indirizzo URI a cui si fa riferimento.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<i>name</i>	Attributo	Nome del riferimento.
<i>replace</i>	Attributo	Testo di rimpiazzo da mostrare.
email	Vuoto	Comando '@email' di Texinfo.
<i>email</i>	Attributo	Indirizzo di posta elettronica.
<i>name</i>	Attributo	Titolo o descrizione dell'indirizzo.

In particolare, è opportuno osservare che l'attributo 'id' degli elementi 'hn', 'partheadng' e 'tomeheadng', è un'ancora a cui possono puntare tutti i vari tipi di riferimenti incrociati disponibili (tranne 'uref' e 'email' che puntano a degli URI).

L'esempio seguente mostra come usare l'elemento 'pxref' in modo molto semplice:

```
<p>Sgmltexi crea automaticamente il nodo Top. Come già spiegato in precedenza, (<pxref id="top node menu">), il menù può essere...</p>
```

Delimitazione di parole e di frasi

Un certo numero di elementi serve a delimitare parole o frasi, per qualche motivo. Il DTD di Sgmltexi è molto permissivo, in modo tale che ogni elemento di questi può contenere qualunque altro elemento interno alla riga di testo. Ciò è stato fatto per assicurare la massima compatibilità con Texinfo, ma in futuro potrebbero essere poste delle piccole limitazioni.

La tabella u91.7 elenca questi elementi, assieme a 'kbdinputstyle', che si usa per specificare lo stile di

rappresentazione del contenuto dell'elemento **'kbd'**.

Tabella u91.7. Delimitazione di parole e frasi.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
code	<code>%inline;</code>	Comando <code>'@code'</code> di Texinfo.
kbd	<code>%inline;</code>	Comando <code>'@kbd'</code> di Texinfo.
kbdinputstyle	Vuoto	Comando <code>'@kbdinputstyle'</code> di Texinfo.
<i>style</i>	Attributo	Stile: <code>'code'</code> , <code>'example'</code> , <code>'distinct'</code> .
key	<code>%inline;</code>	Comando <code>'@key'</code> di Texinfo.
samp	<code>%inline;</code>	Comando <code>'@samp'</code> di Texinfo.
var	<code>%inline;</code>	Comando <code>'@var'</code> di Texinfo.
env	<code>%inline;</code>	Comando <code>'@env'</code> di Texinfo.
file	<code>%inline;</code>	Comando <code>'@file'</code> di Texinfo.
command	<code>%inline;</code>	Comando <code>'@command'</code> di Texinfo.
option	<code>%inline;</code>	Comando <code>'@option'</code> di Texinfo.
dfn	<code>%inline;</code>	Comando <code>'@dfn'</code> di Texinfo.
cite	<code>%inline;</code>	Comando <code>'@cite'</code> di Texinfo.
acronym	<code>%inline;</code>	Comando <code>'@acronym'</code> di Texinfo.
url	<code>%inline;</code>	Comando <code>'@url'</code> di Texinfo.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
emph	%inline;	Comando '@emph' di Texinfo.
strong	%inline;	Comando '@strong' di Texinfo.
sc	%inline;	Comando '@sc' di Texinfo.
roman	%inline;	Comando '@r' di Texinfo.
italic	%inline;	Comando '@i' di Texinfo.
bold	%inline;	Comando '@b' di Texinfo.
typewriter	%inline;	Comando '@t' di Texinfo.

Viene mostrato un esempio molto semplice dell'uso dell'elemento '**strong**':

```
<p><strong>Pinco Pallino</strong> è un uomo molto vecchio...</p>

<p><strong>Tizio Tizi</strong> ha studiato tecnologia delle
comunicazioni...</p>
```

Delimitazione di blocchi di testo

Alcuni elementi servono a delimitare blocchi di testo, o un tipo particolare di testo interno alle righe. Il DTD di Sgmltexi è molto permissivo per assicurare la massima compatibilità con Texinfo, ma in futuro potrebbero essere poste delle piccole limitazioni.

La tabella u91.9 elenca questi elementi, assieme a '**pre**', che permette di inserire del testo preformattato, e a '**exdent**', utilizzato all'interno di '**pre**' per ottenere delle righe che sporgono verso l'esterno.

Tabella u91.9. Delimitazione di blocchi di testo.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
exdent	%inline;	Comando '@exdent' di Texinfo.
pre	%inline;	Testo preformattato.
quotation	%block;	Comando '@quotation' di Texinfo.
display	%block; o 'pre'	Comando '@display' di Texinfo.
smalldisplay	%block; o 'pre'	Comando '@smalldisplay' di Texinfo.
example	%block; o 'pre'	Comando '@example' di Texinfo.
smallexample	%block; o 'pre'	Comando '@smallexample' di Texinfo.
flushleft	%inline;	Comando '@flushleft' di Texinfo.
flushright	%inline;	Comando '@flushright' di Texinfo.
lisp	%block; o 'pre'	Comando '@lisp' di Texinfo.
smalllisp	%block; o 'pre'	Comando '@smalllisp' di Texinfo.
cartouche	%block; o 'pre'	Comando '@cartouche' di Texinfo.
format	%block; o 'pre'	Comando '@format' di Texinfo.
smallformat	%block; o 'pre'	Comando '@smallformat' di Texinfo.
texinfo		Codice Texinfo incorporato.

In generale, l'uso di questi elementi è molto semplice, come si può vedere in questo caso:

```
<example>
<p>Bla bla bla...</p>
<p>Bla bla bla...</p>
</object>
```

L'esempio seguente, invece, mostra l'uso dell'elemento **'pre'**, allo scopo di incorporare del testo preformattato, pur continuando a espandere le macro SGML:

```
<example>
<pre>
#!/usr/bin/perl
while ($line = &gt;STDIN&gt;)
{
    chomp $line;
    print ("$line\r\n");
}
</pre>
</object>
```

In aggiunta, si può delimitare il contenuto dell'elemento **'pre'** per poterlo scrivere in modo letterale:

```
<example>
<pre>
<![CDATA[
#!/usr/bin/perl
while ($line = <STDIN>)
{
    chomp $line;
    print ("$line\r\n");
}
]]>
</pre>
</example>
```

Elenchi e tabelle



Elenchi e tabelle, sono blocchi di testo. La gestione di Texinfo per ciò che riguarda queste strutture, è abbastanza speciale. Qui viene riassunto tutto nella tabella u91.13, che però richiede la conoscenza dei comandi di Texinfo corrispondenti.

Tabella u91.13. Elenchi e tabelle.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
itemize	<code>'item'</code> , <code>'itemx'</code> , <code>%block</code> ;	Comando <code>'@itemize'</code> di Texinfo.
<i>mark</i>	Attributo	Segno usato al posto del pallino iniziale.
enumerate	<code>'item'</code> , <code>'itemx'</code> , <code>%block</code> ;	Comando <code>'@enumerate'</code> di Texinfo.
<i>start</i>	Attributo	Valore iniziale dell'elenco numerato.
table	<code>'item'</code> , <code>'itemx'</code> , <code>%block</code> ;	Comando <code>'@table'</code> di Texinfo.
<i>emphasis</i>	Attributo	Enfasi della colonna descrittiva: <code>'asis'</code> , <code>'code'</code> , <code>'samp'</code> , <code>'var'</code> , <code>'kbd'</code> , <code>'file'</code> .
vtable	<code>'item'</code> , <code>'itemx'</code> , <code>%block</code> ;	Comando <code>'@vtable'</code> di Texinfo.
<i>emphasis</i>	Attributo	Enfasi della colonna delle variabili: <code>'asis'</code> , <code>'code'</code> , <code>'samp'</code> , <code>'var'</code> , <code>'kbd'</code> , <code>'file'</code> .
ftable	<code>'item'</code> , <code>'itemx'</code> , <code>%block</code> ;	Comando <code>'@ftable'</code> di Texinfo.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<i>emphasis</i>	Attributo	Enfasi della colonna delle funzioni: ‘ asis ’, ‘ code ’, ‘ samp ’, ‘ var ’, ‘ kbd ’, ‘ file ’.
item	%inline; o vuoto.	Comando ‘@item’ di Texinfo.
itemx	%inline; o vuoto.	Comando ‘@itemx’ di Texinfo.
multitable		Comando ‘@multitable’ di Texinfo.
columnfraction	‘.n’.	Colonna larga 0,n volte lo spazio totale.
columnexample	Testo puro.	Colonna larga tanto quanto l’esempio.
raw	%inline;, ‘ tab ’.	Riga di una tabella.
tab	Vuoto	Separatore tra una colonna e la successiva.

Vengono mostrati alcuni esempi, a cominciare da un elenco non numerato:

```
<itemize mark="#">
<item>
  <p>Primo elemento dell'elenco.</p>
<item>
  <p>Secondo elemento.</p>
</itemize>
```

In questo caso, si ottiene un elenco puntato di due sole voci, dove al posto del pallino usuale, appare il simbolo ‘#’. Sostituendo l’elemento ‘**itemize**’ con ‘**enumerate**’, si ottiene un elenco numerato:

```
<enumerate start="3">
<item>
  <p>Primo elemento dell'elenco.</p>
<item>
  <p>Secondo elemento.</p>
</enumerate>
```

In questo caso, si fa in modo che il primo dei due elementi abbia il numero tre. L'elenco descrittivo si ottiene attraverso l'elemento **'table'**, dove gli elementi **'item'** contengono le voci relative. Si osservi l'esempio:

```
<table emphasis="code">
<item>ls</item>
<itemx>dir</itemx>
  <p>Elenco del contenuto della directory.</p>
<item>cd</item>
  <p>Cambia directory.</p>
</table>
```

Si intende così che l'elemento **'itemx'** serve quando un elemento dell'elenco è composto da più di una voce.

Le tabelle, intese come quelle a cui si è abituati di solito, sono gestite attraverso l'elemento **'multitable'**. Questo, prima dell'indicazione delle righe che compongono la tabella, richiede di specificare quante sono le colonne e quanto larghe devono essere. Per questo, all'inizio occorre utilizzare una serie di elementi **'columnfraction'**, oppure **'columnexample'**, attraverso i quali si specificano proprio queste larghezze (in percentuale o attraverso un testo di esempio). L'esempio seguente mostra il caso di una tabella le cui colonne sono state definite in modo percentuale:

```

<multitable>
<columnfraction>.30</columnfraction>
<columnfraction>.70</columnfraction>
<raw><strong>Parametro LOC</strong>
<tab><strong>Posizione corrispondente</strong>
</raw>
<raw>h
<tab>posizione attuale
</raw>
<raw>t
<tab>superiore
</raw>
<raw>b
<tab>inferiore
</raw>
<raw>p
<tab>pagina
</raw>
</multitable>

```

In alternativa, dato che la larghezza delle colonne dipende proprio dai titoli, si potrebbe fare così:

```

<multitable>
<columnexample>Parametro LOC</columnexample>
<columnexample>Posizione corrispondente</columnexample>
<raw><strong>Parametro LOC</strong>
<tab><strong>Posizione corrispondente</strong>
</raw>
<raw>h
<tab>posizione attuale
</raw>
<raw>t
<tab>superiore
</raw>
<raw>b
<tab>inferiore
</raw>
<raw>p
<tab>pagina
</raw>
</multitable>

```

In entrambi i casi, lo scopo è quello di ottenere uno specchietto si-

mile a quello che segue. Si osservi che non ci sono didascalie e nemmeno esiste la possibilità di collocare dinamicamente la tabella.

Parametro	LOC	Posizione corrispondente
h		posizione attuale
t		superiore
b		inferiore
p		pagina

Inserzioni

«

Alcuni elementi sono difficilmente classificabili in gruppi particolari. Qui, vengono distinti in due raggruppamenti: quelli interni alle righe e quelli che rappresentano dei blocchi. A questi corrispondono le tabelle u91.20 e u91.21.

Tabella u91.20. Inserzioni interne alle righe.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
dmn	#PCDATA	Comando '@dmn' di Texinfo.
math	#PCDATA	Comando '@math' di Texinfo.
footnote	%inline;	Comando '@footnote' di Texinfo.
image	Vuoto	Comando '@image' di Texinfo.
<i>name</i>	Attributo	Nome del file da inserire, senza estensione.
<i>width</i>	Attributo	Ampiezza dell'immagine.
<i>height</i>	Attributo	Altezza dell'immagine.
whole	%inline;	Comando '@w' (previene l'interruzione di riga).
br	Vuoto	Comando '@*' (interruzione di riga).

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
dh	Vuoto	Comando '@-' (separazione facoltativa).
hyphenation	Vuoto;	Comando '@hyphenation' di Texinfo.
<i>words</i>	Attributo	Elenco di parole separate in sillabe.

Tabella u91.21. Inserzione di blocchi.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
sp	Vuoto	Comando '@sp' di Texinfo.
<i>lines</i>	Attributo	Quantità di righe da saltare (un numero intero).
page	Vuoto	Comando '@page' di Texinfo.
group	%block;	Comando '@group' di Texinfo.
need	Vuoto	Comando '@need' di Texinfo.
<i>mils</i>	Attributo	Millesimi di pollice richiesti.

Definizioni

Texinfo prevede un grande numero di comandi per la descrizione di definizioni di vario genere. Queste «definizioni» vanno intese generalmente come dei modelli sintattici. È un po' difficile comprendere bene quando usare questa o quella forma di definizione; per cui occorre studiare la documentazione di Texinfo.

Tutte le forme di definizione si dichiarano attraverso un elemento provvisto di diversi attributi. Questo elemento contiene generalmente la descrizione del modello, in una serie di blocchi di testo, ma in particolare potrebbe contenere la descrizione degli argomenti, all'interno dell'elemento **'args'**, comune a tutte le definizioni che ne hanno.

Tabella u91.22. Definizioni.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
args	%inline;	Argomenti di una definizione.
deffn	'args', %block;	Comando '@deffn' di Texinfo.
<i>cat</i>	Attributo	Categoria della funzione.
<i>name</i>	Attributo	Nome della funzione.
deffnx	Vuoto	Comando '@deffnx'. Attributi come 'deffn'.
defun	'args', %block;	Comando '@defun' di Texinfo.
<i>name</i>	Attributo	Nome della funzione.
defunx	Vuoto	Comando '@defunx'. Attributi come 'defun'.
defmac	'args', %block;	Comando '@defmac' di Texinfo.
<i>name</i>	Attributo	Nome della macro.
defmacx	Vuoto	Comando '@defmacx'. Attributi come 'defmac'.
defspec	'args', %block;	Comando '@defspec' di Texinfo.
<i>name</i>	Attributo	Nome di uno <i>special form</i> .
defspecx	Vuoto	Comando '@defspecx'. Attributi come 'defspec'.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
defvr	%block;	Comando '@defvr' di Texinfo.
<i>cat</i>	Attributo	Categoria della variabile.
<i>name</i>	Attributo	Nome della variabile.
defvrx	Vuoto	Comando '@defvrx'. Attributi come 'defvr'.
defvar	%block;	Comando '@defvar' di Texinfo.
<i>name</i>	Attributo	Nome della variabile.
defvarx	Vuoto	Comando '@defvarx'. Attributi come 'defvar'.
defopt	%block;	Comando '@defopt' di Texinfo.
<i>name</i>	Attributo	Nome dell'opzione.
defoptx	Vuoto	Comando '@defoptx'. Attributi come 'defopt'.
deftypefn	'args', %block;	Comando '@deftypefn' di Texinfo.
<i>cat</i>	Attributo	Categoria.
<i>type</i>	Attributo	Tipo di dati.
<i>name</i>	Attributo	Nome.
deftypefnx	Vuoto	Comando '@deftypefnx'. Attributi come 'deftypefn'.
deftypefun	'args', %block;	Comando '@deftypefun' di Texinfo.
<i>type</i>	Attributo	Tipo di dati.
<i>name</i>	Attributo	Nome.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<code>deftypefunx</code>	Vuoto	Comando <code>'@deftypefunx'</code> . Attributi come <code>'deftypefun'</code> .
<code>deftypevr</code>	<code>%block;</code>	Comando <code>'@deftypevr'</code> di Texinfo.
<i>cat</i>	Attributo	Categoria.
<i>type</i>	Attributo	Tipo di dati.
<i>name</i>	Attributo	Nome.
<code>deftypevr_x</code>	Vuoto	Comando <code>'@deftypevr_x'</code> . Attributi come <code>'deftypevr'</code> .
<code>deftypevar</code>	<code>%block;</code>	Comando <code>'@deftypevar'</code> di Texinfo.
<i>type</i>	Attributo	Tipo di dati.
<i>name</i>	Attributo	Nome.
<code>deftypevar_x</code>	Vuoto	Comando <code>'@deftypevar_x'</code> . Attributi come <code>'deftypevar'</code> .
<code>args</code>	<code>%inline;</code>	Argomenti di una definizione.
<code>defcv</code>	<code>%block;</code>	Comando <code>'@defcv'</code> di Texinfo.
<i>cat</i>	Attributo	Categoria.
<i>class</i>	Attributo	Classe.
<i>name</i>	Attributo	Nome.
<code>defcv_x</code>	Vuoto	Comando <code>'@defcv_x'</code> . Attributi come <code>'defcv'</code> .
<code>defivar</code>	<code>%block;</code>	Comando <code>'@defivar'</code> di Texinfo.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<i>class</i>	Attributo	Classe.
<i>name</i>	Attributo	Nome.
defivarx	Vuoto	Comando '@defivarx'. Attributi come 'defivar'.
deftypeivar	%block;	Comando '@deftypeivar' di Texinfo.
<i>class</i>	Attributo	Classe.
<i>type</i>	Attributo	Tipo.
<i>name</i>	Attributo	Nome.
deftypeivarx	Vuoto	Comando '@deftypeivarx'. Attributi come 'deftypeivar'.
defop	'args', %block;	Comando '@defop' di Texinfo.
<i>cat</i>	Attributo	Categoria.
<i>class</i>	Attributo	Classe.
<i>name</i>	Attributo	Nome.
defopx	Vuoto	Comando '@defopx'. Attributi come 'defop'.
defmethod	'args', %block;	Comando '@defmethod' di Texinfo.
<i>class</i>	Attributo	Classe.
<i>name</i>	Attributo	Nome.
defmethodx	Vuoto	Comando '@defmethodx'. Attributi come 'defmethod'.
deftypemethod	'args', %block;	Comando '@deftypemethod' di Texinfo.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
<i>class</i>	Attributo	Classe.
<i>type</i>	Attributo	Tipo.
<i>name</i>	Attributo	Nome.
deftypemethodx	Vuoto	Comando '@deftypemethodx'. Attributi come 'deftypemethod'.
deftp	'args', %block;	Comando '@deftp' di Texinfo.
<i>cat</i>	Attributo	Categoria.
<i>name</i>	Attributo	Nome.
deftpx	Vuoto	Comando '@deftpx'. Attributi come 'deftp'.

Ecco un esempio molto semplice:

```
<deffn cat="Command" name="sgmltexi">
  <args>[<var>options</var>]... <var>sgml_source</var></args>

  <p>This is the front-end for the SGML to Texinfo system.</p>

</deffn>
```

La composizione in formato Info genera il risultato seguente:

```
- Command: sgmltexi [OPTIONS]... SGML_SOURCE
  This is the front-end for the SGML to Texinfo system.
```

Codice condizionato e codice letterale in base alla composizione

«

Texinfo ha la possibilità di selezionare del codice in dipendenza del tipo di composizione finale. In SGML si possono fare cose simili attraverso le sezioni marcate, ma non si tratta della stessa cosa. Per

questa ragione, Sgmltexi include alcuni elementi speciali corrispondenti ai comandi che servono a Texinfo per selezionare il codice, consentendo anche di inserire pezzi di codice letterale.

Tabella u91.25. Codice condizionato e codice letterale in base alla composizione.

Elemento o attributo	Contenuto	Descrizione o corrispondenza con Texinfo
ifinfo	<code>%inline;</code>	<code>'@ifinfo'</code> ... <code>'@end ifinfo'</code>
ifinfoblock	<code>%block;</code>	<code>'@ifinfo'</code> ... <code>'@end ifinfo'</code>
iftex	<code>%inline;</code>	<code>'@iftex'</code> ... <code>'@end iftex'</code>
iftexblock	<code>%block;</code>	<code>'@iftex'</code> ... <code>'@end iftex'</code>
ifhtml	<code>%inline;</code>	<code>'@ifhtml'</code> ... <code>'@end ifhtml'</code>
ifhtmlblock	<code>%block;</code>	<code>'@ifhtml'</code> ... <code>'@end ifhtml'</code>
ifnotinfo	<code>%inline;</code>	<code>'@ifnotinfo'</code> ... <code>'@end ifnotinfo'</code>
ifnotinfoblock	<code>%block;</code>	<code>'@ifnotinfo'</code> ... <code>'@end ifnotinfo'</code>
ifnottex	<code>%inline;</code>	<code>'@ifnottex'</code> ... <code>'@end ifnottex'</code>
ifnottexblock	<code>%block;</code>	<code>'@ifnottex'</code> ... <code>'@end ifnottex'</code>
ifnohtml	<code>%inline;</code>	<code>'@ifnohtml'</code> ... <code>'@end ifnohtml'</code>
ifnohtmlblock	<code>%block;</code>	<code>'@ifnohtml'</code> ... <code>'@end ifnohtml'</code>
tex	<code>#PCDATA</code>	<code>'@tex'</code> ... <code>'@end tex'</code>
html	<code>#PCDATA</code>	<code>'@html'</code> ... <code>'@end html'</code>
texinfo	<code>#PCDATA</code>	Codice Texinfo.

È importante osservare che `'ifinfo'`, `'iftex'`, `'ifhtml'`, `'ifnotinfo'`, `'ifnottex'` e `'ifnohtml'`, sono elementi interni alla riga di testo, che contengono lo stesso genere di cosa. Al contrario, `'ifinfoblock'`, `'iftexblock'`, `'ifhtmlblock'`, `'ifnotinfoblock'`, `'ifnottexblock'` e `'ifnohtmlblock'`, sono blocchi che contengono altri blocchi. Questa distinzione è necessaria per evitare problemi nella definizione del documento SGML (nel DTD).

In particolare, gli elementi `'tex'`, `'html'` e `'texinfo'`, sono fatti per contenere testo letterale solitamente racchiuso tra `'<![CDATA['` e `']>'`.

L'elemento `'texinfo'` non ha un comando equivalente in Texinfo, perché rappresenta del codice Texinfo. Si osservi l'esempio seguente:

```
<p>The letter <texinfo>@ubaraccent{o}</texinfo> is a special...</p>
```

Usando questo elemento, potrebbe essere necessario forzare l'interpretazione letterale anche da parte dell'SGML. In tal caso, il contenuto dell'elemento può essere racchiuso come si vede qui:

```
<p>The letter <texinfo><![CDATA[@ubaraccent{o}]]></texinfo> is a...
```

Il caso particolare dell'esempio non mostra una situazione in cui sia indispensabile l'interpretazione SGML letterale, tuttavia questo è il modo quando succede tale circostanza.

Viene mostrato un altro esempio nell'uso di codice letterale specifico per il tipo di composizione. L'intenzione è quella di mostrare un'espressione matematica molto semplice: $123 + 10^{-1}$.

```
<p><tex><![CDATA[ $123+10^{-1}$ ]]></tex>
<html><![CDATA[123+10<sup>-1</sup>]]></html>
<ifinfo>123+10^-1</ifinfo>
= 12.3</p>
```

Si potrebbe notare una sorta di incoerenza nell'uso degli elementi letterali, assieme a **'ifinfo'**, il cui scopo è solo quello di essere preso in considerazione quando la composizione produce il formato Info. Il fatto è che gli altri due elementi letterali, oltre che contenere codice letterale per il tipo rispettivo di composizione, sono implicitamente elementi condizionali. Dal momento che la composizione Info non può prevedere una codifica letterale speciale, l'unico modo per integrare le varie parti è quello di usare **'ifinfo'** per rappresentare in qualche modo l'espressione, anche in questo caso.

Problemi

Texinfo, come TeX e *roff, distingue i blocchi di testo in quanto separati da una o più righe vuote. In tal modo, la distinzione tra blocchi di testo e testo interno alle righe, è solo una questione di spazio verticale. Per esempio, il pezzo seguente di un sorgente Texinfo, mostra tre ambienti del tipo **'@ifcomposizione'**, che sono parte dello stesso blocco di testo, ovvero lo stesso paragrafo.

```
La composizione attuale è
@iftex
TeX
@end iftex
@ifhtml
HTML
@end ifhtml
@ifinfo
Info
@end ifinfo
e si può vedere che...
```

In una situazione differente, questi ambienti possono diventare blocchi isolati di testo, come si vede qui:

```
La composizione attuale è:

@iftex
TeX
@end iftex

@ifhtml
HTML
@end ifhtml

@ifinfo
Info
@end ifinfo

Si può vedere che...
```

Con un sistema SGML, questa confusione di ruoli non è desiderabile, oltre che essere difficile da realizzare. Questo è il motivo per cui Sgmltexi distingue tra ‘*@ifcomposizione*’ o ‘*@ifnotcomposizione*’, e ‘*@ifcomposizioneblock*’ o ‘*@ifnotcomposizioneblock*’.

Sgmltexi cerca di mantenere le interruzioni di riga contenute all’interno del sorgente SGML, ma per questo ci sono delle conseguenze nell’uso degli ambienti condizionali, del tipo interno alle righe. Ciò dipende dal fatto che necessariamente occorre aggiungere delle in-

terruzioni aggiuntive. Si supponga di voler scrivere qualcosa come ciò che segue:

```
<p>La composizione attuale
è <iftex>TeX</iftex><ifhtml>HTML</ifhtml><ifinfo>Info</ifinfo>, per cui
si sa cosa comporta questo fatto.</p>
```

Ci si aspetta che i marcatori di apertura e di chiusura vengano rimpiazzati aggiungendo anche le interruzioni di riga appropriate. Ma se fosse così, il risultato sarebbe quello seguente, in cui ciò che prima era testo interno alla riga diventa invece un blocco separato:

```
La composizione attuale
è
@iftex
TeX
@end iftex

@ifhtml
HTML
@end ifhtml

@ifinfo
Info
@end ifinfo
, per cui
si sa cosa comporta questo fatto.</p>
```

Per risolvere il problema, questi elementi intesi come ambienti condizionali interni alle righe, non introducono alcuna interruzione iniziale o finale che sia; rimane compito dell'autore il preoccuparsi di questo problema. Per questo, il sorgente di Sgmltexi deve essere scritto come si vede nell'esempio seguente, considerando anche che non c'è alcun modo di mettere la virgola dopo il nome del tipo di composizione.

```
<p>La composizione attuale è  
<iftex>TeX</iftex>  
<ifhtml>HTML</ifhtml>  
<ifinfo>Info</ifinfo>  
per cui si sa cosa comporta questo fatto.</p>
```

Lo stesso problema appare con gli elementi **'tex'** e **'html'**, ma in tal caso non c'è bisogno di qualificarne il contenuto, che si intende sempre come testo interno alle righe.

```
<p>  
<tex>  
$$ \chi^2 = \sum_{i=1}^N  
    \left (y_i - (a + b x_i)  
    \over \sigma_i\right)^2 $$  
</tex>  
</p>
```

Utilizzando un sistema SGML, l'inserzione di codice letterale per il tipo di composizione particolare che si utilizza, è da considerarsi come l'**ultima risorsa**. In altri termini, se sono necessari tali espedienti, è evidente che l'SGML è la scelta sbagliata per scrivere la propria documentazione.