



XFrames – A volte ritornano...

DARIO DE JUDICIBUS

Messe in soffitta con l'avvento dei fogli di stile, le famigerate cornici sono tornate, ma questa volta rinnovate e perfettamente integrate nell'architettura XML

Consideriamo un sito web di tipo tradizionale, come quelli nei quali ci troviamo spesso a navigare. Analizzandone la struttura è facile identificare all'interno di ogni singola pagina tutta una serie di elementi, ognuno dei quali ha caratteristiche ben precise quali la posizione, le dimensioni e, ovviamente, i contenuti. Alcuni di questi elementi possono essere statici, altri essere dinamici e magari in grado di interagire con il visitatore, di cambiare di dimensione o forma se la pagina viene allargata o ristretta, oppure di riadattare i propri contenuti alle nuove dimensioni. Ad esempio, in una struttura abbastanza comune, direi quasi "classica", potremmo avere in alto un'intestazione, magari con un logo, mentre in fondo alla stessa potrebbe essere presente una sezione contenente le note legali, i diritti d'autore, un eventuale indirizzo di posta elettronica dell'amministratore del sito e il solito contatore. A sinistra, poi, potremmo avere il classico menù di navigazione e a destra un'eventuale sezione per le novità, la pubblicità o quant'altro. Al centro, ovviamente, ci sarebbe il cuore dell'intero sito, ovvero le informazioni che si desidera pubblicare e che cambiano a seconda di come si interagisce con le altre sezioni.

È chiaro quindi che disegnare un sito web non vuole solo dire identificarne i contenuti nel suo complesso, ma anche progettarne la struttura, definirne ogni singolo elemento, specificandone posizione, dimensioni, comportamento, stile, correlazioni con gli altri elementi e, infine, ovviamente, anche quello che dovrà apparire al suo interno.

In passato questa strutturazione della pagina veniva fatta in due modi: o con le tabelle, o con le cornici (*frame*). Le tabelle hanno il grande vantaggio di poter costruire in modo molto semplice strutture complesse capaci di adattarsi al ridimensionamento della finestra del browser grazie agli algoritmi integrati, ma hanno anche lo svantaggio di rendere il codice HTML molto difficile da leggere e quindi da mantenere. Inoltre è necessario replicare in ogni pagina i vari elementi, come ad esempio i menù e le barre di navigazione, con il rischio di disallineare i diversi file a fronte di un'eventuale modifica di

uno qualsiasi di questi elementi. Le cornici consentono invece di utilizzare un singolo documento per ogni componente e di non dover gestire eventuali problemi legati al ridimensionamento della pagina, permettono cioè il riutilizzo del codice. In pratica, caricando un documento in ogni cornice, si simula una determinata struttura di pagina, ovvero una specifica configurazione di contenuti.

Purtroppo ogni documento all'interno di una cornice è indipendente dagli altri. Certo, è possibile attivare un'azione in una cornice e agire in questo modo sul contenuto di un'altra, ma ad esempio, se si preme il pulsante **INDIETRO** (*back*) del browser, non sempre si ritorna alla situazione precedente. Lo stesso dicasi se si cerca di ricaricare la pagina con il pulsante **AGGIORNA** (*reload, update o refresh*): il risultato può essere del tutto imprevisto e comunque non esiste alcun controllo sull'ordine nel quale vengono caricati i singoli documenti. Inoltre non è possibile definire un segnalibro che corrisponda a una specifica configurazione delle cornici, e questo vuol dire che anche i motori di ricerca hanno difficoltà con questi oggetti, dato che ragionano in termini di pagine e non di insiemi di cornici (*frameset*). Una conseguenza abbastanza consueta è che se si arriva a una certa pagina attraverso un motore di ricerca, ci si trova nell'impossibilità di navigare nel sito dato che di fatto si è caricato solo un pezzo dell'intera configurazione, ovvero un solo elemento della pagina. Analogamente può capitare di rimanere intrappolati in una cornice, ovvero di caricare un altro sito all'interno di una singola cornice piuttosto che visualizzarlo per intero nella finestra del browser. Un problema di questo tipo può portare a una situazione che ben si può rappresentare metaforicamente come un oggetto che si rifletta in una serie di specchi paralleli, ovvero un'immagine in un'immagine all'interno di un'immagine e così via.

Certo, esistono varie tecniche per ovviare all'intrappolamento o alle pagine "orfane". La maggior parte sono basate su JavaScript, ma si tratta di un'ulteriore complessità che va poi comunque gestita e mantenuta in tutto il sito. Inoltre l'elemento **noframes**, che dovrebbe permettere allo sviluppatore di gestire uno schema alternativo per quei browser che non gestiscono le cornici, non è quasi mai utilizzato e, se presente, si limita a constatare come non sia stato possibile visualizzare correttamente il sito.

Dato poi che ogni cornice permette di visualizzare un documento differente, è anche possibile per un malintenzionato dirottare (*hijacking*) uno dei puntatori a un sito differente e caricare così in una cornice un documento diverso da quello previsto, che magari gli assomiglia, ma inteso a veicolare altri contenuti attraverso il sito che nulla hanno a che vedere con quelli del sito originale. Esiste quindi un problema di sicurezza. In effetti, quando si carica una pagina web, la si identifica con un unico URI, che sia un singolo documento o una serie di documenti che vengono posizionati in due o più cornici. Il visitatore non ha alcuna visibilità di cosa effettivamente venga caricato: ad ogni cornice infatti può anche essere associato staticamente o dinamicamente un documento appartenente a un altro dominio.

Con l'avvento dei fogli di stile, gli sviluppatori di siti web hanno iniziato a realizzare le loro opere utilizzando le divisioni (**div**) piuttosto che tabelle e cornici. Il risultato è stato di avere un codice molto più leggibile e un comportamento della pagina molto più affidabile, anche se questo ha richiesto di sviluppare varie tecniche per gestire le incompatibilità dei vari *browser* nell'implementare gli standard del W3C, trasferendo di fatto la complessità del codice dai file HTML, XHTML e XML a quelli CSS. Non si tratta tuttavia di una soluzione perfetta, dato che le divisioni non permettono tecniche di impaginazione con sezioni a scorrimento indipendenti e richiedono la duplicazione degli stessi elementi in ogni pagina, come accadeva per le tabelle. A quest'ultimo problema si è ovviato grazie alla generazione del codice sul lato server (PHP, ASP, ColdFusion, ecc...), ma ovviamente non tutti hanno la possibilità di agire alla fonte, e comunque è richiesta una certa cura da parte del programmatore, dato che in questo modo si possono creare serie esposizioni di sicurezza. Il PHP, ad esempio, è un linguaggio molto potente, ma se non si fa attenzione nello scrivere il codice, permette all'utente smalzato di penetrare all'interno del sito e alterarne le caratteristiche.

In definitiva, quindi, il principio di costruire una pagina a partire da più di un documento ha tuttora una sua validità, anche se il modo con il quale attualmente si può realizzare non è assolutamente soddisfacente. Da qui è nata l'esigenza di ridefinire l'implementazione del modello a cornici sviluppando uno standard che fosse più coerente, robusto e in linea con le specifiche XML: le *XFrames*, appunto.

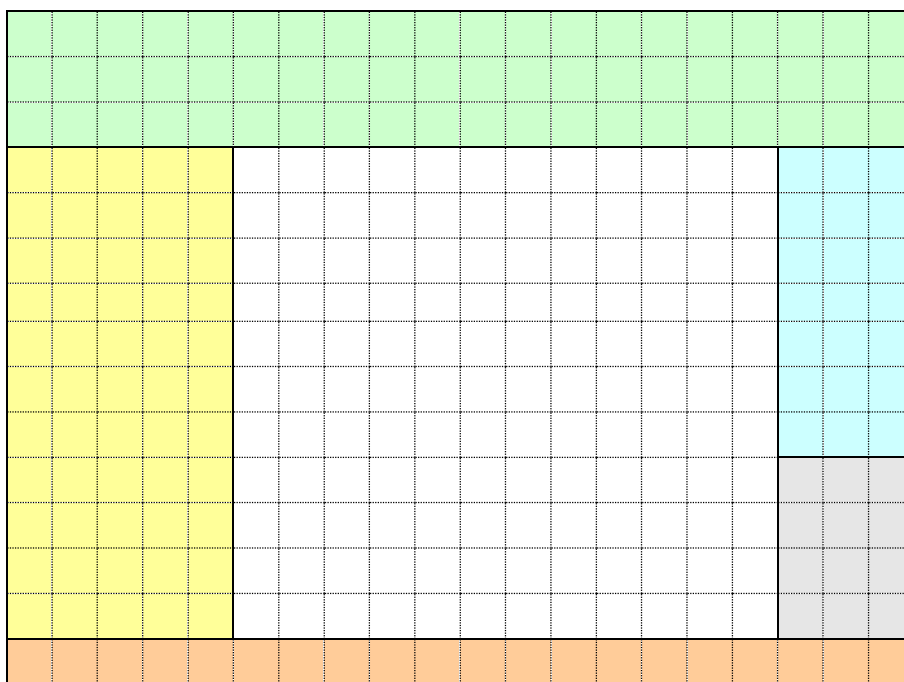


Figura 1 - Schema a cornici tradizionale

In pratica il modulo XFrames non è altro che un linguaggio per comporre in una singola vista due o più documenti di varia natura all'interno di una griglia nella quale ogni documento occupa una sezione di forma rettangolare.

La vista in questione mantiene il nome di *frameset*, o schema a cornici, anche se come marcatore questo termine è scomparso rispetto all'XHTML. Le varie sezioni possono poi essere composte rispetto alle altre in vari modi e quindi, non solo affiancate come nelle vecchie cornici, ma anche rappresentate come sottofinestre spostabili all'interno della finestra del *browser* (*window-like panes*) o come foglietti di un libretto a linguette (*tabbed panes*).

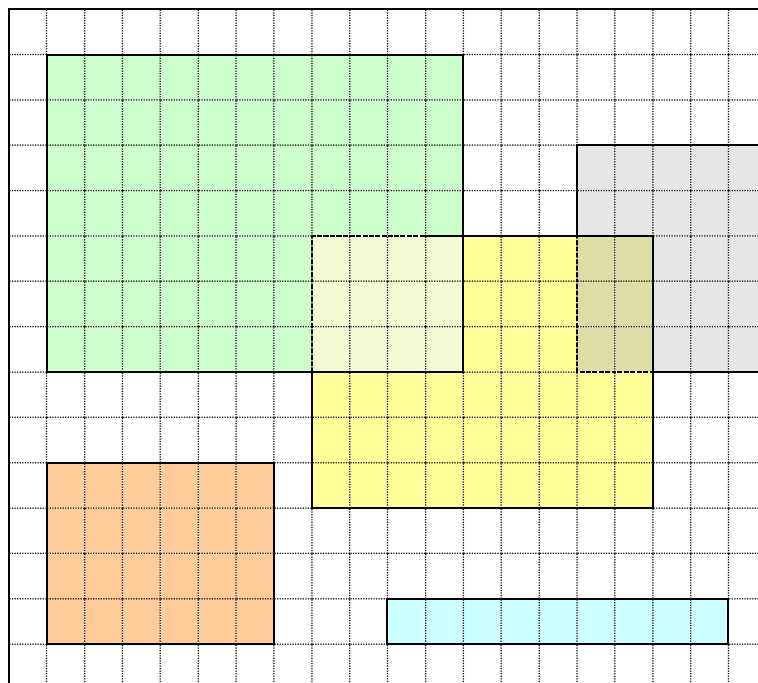


Figura 2 - Schema a cornici a finestre

Già il fatto che le varie sezioni si possano comporre in più modi offre allo sviluppatore un ulteriore grado di libertà nel disegno del sito. Ad esempio, una configurazione a libretto non richiede una barra di navigazione, essendo questa implicita nelle linguette associate ad ogni singola sezione.

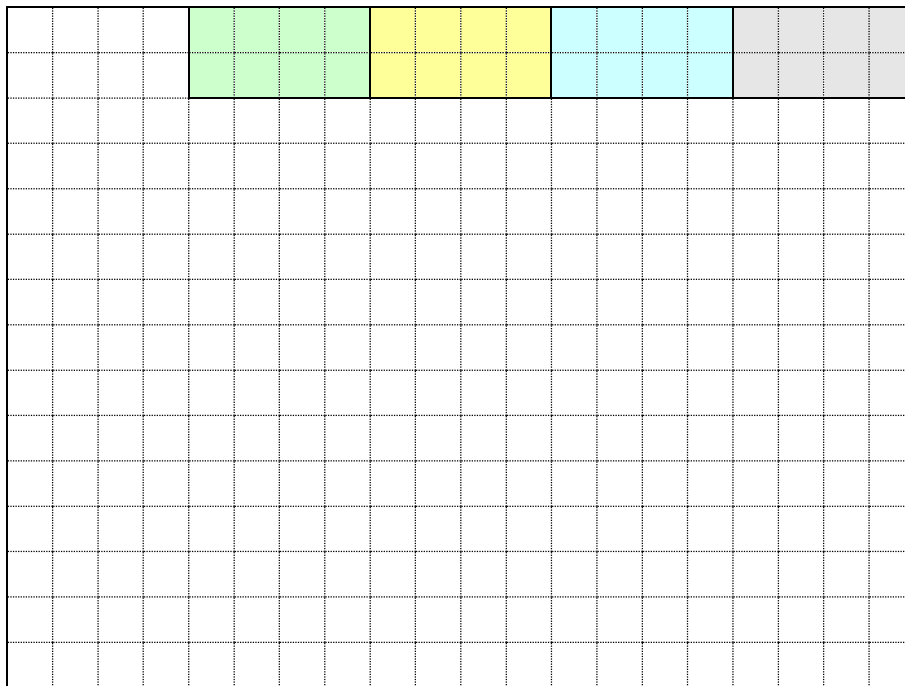


Figura 3 - Schema a cornici a libretto

Il modulo è piuttosto semplice. Gli elementi di questo linguaggio XML sono infatti solo sei: **frames**, **head**, **title**, **style**, **group** e **frame**.

Il primo elemento, **frames**, non è altro che la radice del documento e, a parte gli attributi tipici di ogni marcatore XML, presenta l'attributo facoltativo **xml:base**, che permette di risolvere eventuali URI relativi presenti all'interno del documento.

Il secondo documento, **head**, non è altro che la sezione di testa utilizzata per i metadati, così come esiste anche in HTML e XHTML. In particolare, questo elemento è usato per specificare un titolo (**title**) e un eventuale foglio di stile (**style**). Entrambi questi marcatori funzionano come i loro analoghi in XHTML, per cui non rappresentano nulla di nuovo sul piano sintattico. Su quello semantico, invece, già il fatto di poter dare un titolo e, soprattutto, di poter associare un foglio di stile allo schema nel suo complesso, rappresenta una novità. In particolare, il foglio di stile **non** serve a definire gli stili degli specifici documenti che saranno caricati nelle varie cornici, bensì a dare uno stile alle cornici stesse, ad esempio, definirne i margini o eventuali bordi. Facciamo un esempio:

```
<frames xmlns="http://www.w3.org/2002/06/xframes/">
  <head>
    <title>Portale dei portali</title>
    <style type="text/css">
      #testa {height: 10em;margin:0;border-bottom:2px solid red;}
      #menu, #notizie {width: 20%;}
      #piede {height: 4em; margin: 0.5em 2em; border-top:2px solid
red;}
    </style>
  </head>
```

```

<group compose="vertical">
  <frame xml:id="testa" source="titolo.xhtml"/>
  <group compose="horizontal">
    <frame xml:id="menu" source="menu.xhtml"/>
    <frame xml:id="corpo" source="benvenuti.xhtml"/>
    <frame xml:id="notizie" source="notizie.xhtml"/>
  </group>
  <frame xml:id="piede" source="diritti.xhtml"/>
</group>
</frames>

```

Come si può facilmente vedere, l'utilizzo degli stili permette di fare a meno di tutti quegli attributi che caratterizzavano le cornici nell'XHTML, come ad esempio **marginwidth**, **marginheight** e **frameborder**.

Il ruolo del marcatore **frameset** è ora preso da **group** il quale può contenere due o più elementi **frame**. A quest'ultimo, proprio in quanto adesso tutte le caratteristiche di una cornice possono essere definite con i fogli di stile, è rimasto un solo attributo, **source**, che ovviamente serve a specificare l'URI del documento da caricare nella cornice. Naturalmente ogni cornice o gruppo può essere referenziato tramite l'identificativo **xml:id**. In aggiunta **group** ha anch'esso la possibilità di specificare una base (**xml:base**) per gli eventuali URI relativi al suo interno che, ovviamente, si sostituisce a quella specificata in **frames**.

La composizione delle varie cornici in un gruppo è definita invece tramite l'attributo **compose** di **group**. Questo può assumere i seguenti valori: **vertical**, se le cornici sono impilate una sull'altra; **horizontal**, se sono affiancate; **single**, se sono sovrapposte in modo che solamente una alla volta sia visibile in un dato momento; **free**, se le varie cornici si comportano come finestre su un piano di lavoro. È inoltre possibile specificare un nome qualificato (QName) associato a una qualsivoglia altra configurazione. Ad esempio, si potrebbe immaginare una configurazione **x-free-tile** che corrisponde a una tassellatura della finestra del *browser* come avviene all'interno di un sistema operativo quando si attiva l'analogia funzione.

Vediamo qualche esempio:

```

<group compose="vertical">
  <group compose="horizontal">
    <frame xml:id="F1"/>
    <frame xml:id="F2"/>
  </group>
  <group compose="horizontal">
    <frame xml:id="F3"/>
    <frame xml:id="F4"/>
  </group>

```

F1	F2
F3	F4

```
<group compose="horizontal">
  <frame xml:id="F1"/>
</group>
<group compose="vertical">
  <frame xml:id="F2"/>
  <frame xml:id="F3"/>
</group>
</group>
```

F1	F2
	F3

```
<group compose="vertical">
  <frame xml:id="F1"/>
</group>
<group compose="horizontal">
  <frame xml:id="F2"/>
  <frame xml:id="F3"/>
</group>
</group>
```

F1	
F2	F4

```
<group compose="single">
  <frame xml:id="F1"/>
  <frame xml:id="F2"/>
  <frame xml:id="F3"/>
</group>
```

F1	F2	F3
F1		

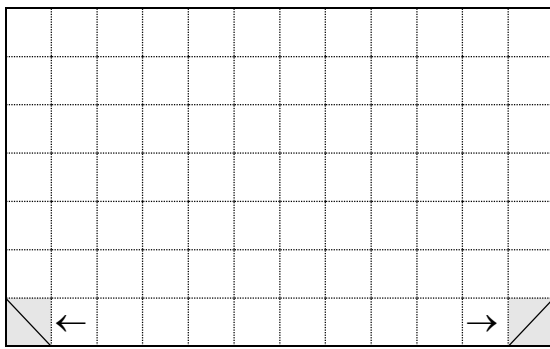
Da notare che, nel caso **compose** assuma il valore **single**, le specifiche non dicono nulla su come debba essere effettivamente realizzato il meccanismo che permette all'utente di passare da un documento all'altro. Si limitano semplicemente ad affermare che debba esistere. Che si usino le linguette, una barra di navigazione o un piccolo controllo in basso che permetta di "sfogliare" le singole pagine, è a discrezione dei disegnatori del *browser*.

[illegible]

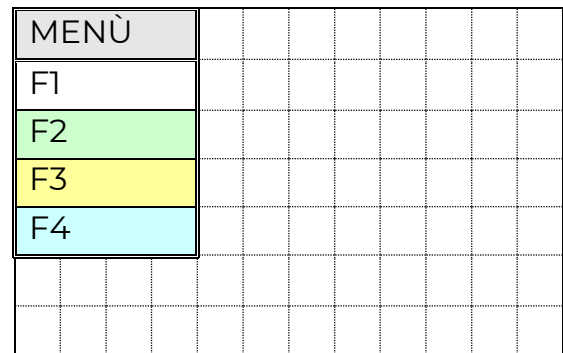
A libretto con linguette

[illegible]

Con barra di navigazione



Con i pulsanti di sfogliamento



Con un menù di navigazione

Figura 4 – Varie realizzazioni di cornici a composizione singola

Fin qui, sebbene *XFrames* offra sicuramente più funzionalità delle cornici classiche, non si vede ancora in che modo possa supplire agli svantaggi menzionati all'inizio di questo articolo. Per comprenderlo è necessario analizzare in che modo un documento venga associato a una specifica cornice.

Consideriamo il seguente insieme:

```
<group compose="horizontal">
  <frame xml:id="f1"/>
  <frame xml:id="f2" source="centro.xml"/>
  <frame xml:id="f3"/>
</group>
```

Come si può vedere ogni cornice ha un identificativo, ma solo una ha anche associato uno specifico documento. Se si caricasse lo schema in questione tramite il suo URI, ad esempio <http://www.esempio.info/schema3.xframes>, avremmo una pagina con al centro il documento `centro.xml` e i lati vuoti. Da notare l'estensione `xframes` per il file che contiene lo schema. Se però facciamo seguire l'URI dello schema dal carattere cancelletto e dalla parola riservata `frames` seguita da una lista di associazioni fra parentesi separate da virgole, ovvero

```
http://www.esempio.info/
schema3.xframes#frames\(f1=sinistra.xml,f3=destra.xml\)
```

allora verranno caricati anche i documenti `sinistra.xml` e `destra.xml` rispettivamente nella cornice di sinistra e in quella di destra. Non solo: è possibile anche ridefinire il documento centrale aggiungendo alla lista `f2=nuovocentro.xml`. In questo caso il documento specificato nell'URI avrà la priorità su quello codificato all'interno dello schema tramite l'attributo `source`. Da notare che se nello schema esiste più di una cornice con lo stesso identificativo, allora il documento sarà caricato **solo** nell'ultima, ovvero non è permesso usare lo stesso identificativo in più cornici per caricare un singolo documento contemporaneamente in più posizioni. Inoltre gli URI delle singole cornici sono risolti in

base al parametro **xml:base** eventualmente associato allo schema o a un gruppo di cornici oppure comunque all'URI dello schema.

Ma cosa succede a questo punto se in **centro.html** c'è un collegamento a un altro documento e l'utente ci clicca sopra? Dipende dal fatto se la cornice che contiene il documento abbia o meno un identificativo. Se ce l'ha, come nel caso visto sopra (**xml:id=f2**), allora il nuovo documento sarà caricato nella cornice con quel identificativo, altrimenti il documento sostituirà l'intero schema. Importante notare che nel momento stesso in cui un URI prende il posto di un altro all'interno di una cornice, nella barra degli indirizzi relativa alla pagina, l'URI dell'intero schema sarà modificato in modo che l'utente sia cosciente dell'avvenuto cambiamento. Ad esempio, se **nuovocentro.html** viene caricato al posto di **centro.html**, la barra degli indirizzi cambierà di conseguenza riportando la nuova struttura:

```
http://www.esempio.info/
```

```
schema3.xframes#frames(f1=sinistra.xml,f3=destra.xml,f2=nuovocentro.html)
```

Naturalmente è possibile che il collegamento specifichi esplicitamente una cornice bersaglio tramite il parametro **target**, ad esempio **target=f1**. In questo caso il documento in questione sarà caricato nella cornice specificata. Se l'identificativo bersaglio non si trova nello schema, ma questi è associato a sua volta a un altro schema, ovvero è stato caricato in una delle cornici di un'altra configurazione, il browser cercherà nel livello superiore l'identificativo in questione e continuerà così finché non l'abbia trovato o meno. In quest'ultimo caso una nuova finestra verrà aperta per caricarvi il documento richiesto.

È chiaro come questo meccanismo non solo permetta di rendere l'utente sempre cosciente di cosa sia stato caricato e dove, ma consente anche ai motori di ricerca di indicizzare interi schemi e soprattutto di non indicizzare singole cornici o comunque un documento al di fuori della sua cornice, dato che l'URI specifico non viene mai rappresentato come tale ma sempre all'interno di una lista. Ovviamente lo sviluppatore dovrà fare attenzione a non cambiare gli identificativi delle cornici se non strettamente necessario o a cambiare la struttura, pena l'obsolescenza degli URI già indicizzati, ovvero la generazione di collegamenti interrotti (*broken link*).

Le specifiche non dicono nulla sull'ordine di caricamento dei documenti nelle singole cornici. La questione è ancora aperta, ma è evidente come il nuovo meccanismo si presti a definire una qualche regola a riguardo. Ad esempio si potrebbero caricare i documenti nell'ordine fornito nella lista specificata nell'URI dopo **frames**, per cui i due URI seguenti corrisponderebbero allo stesso schema, agli stessi contenuti, ma a un ordine di caricamento differente:

```
http://www.esempio.info/
```

```
schema3.xframes#frames(f1=sinistra.xml,f2=nuovocentro.html,f3=destra.xml)
```

<http://www.esempio.info/>

`schema3.xframes#frames(f1=sinistra.xml,f3=destra.xml,f2=nuovocentro.html)`

Per quanto riguarda infine le dimensioni di ogni cornice, queste possono essere specificate direttamente tramite un eventuale foglio di stile, o calcolate automaticamente dal browser in base ai contenuti che saranno caricati nella specifica cornice. Le dimensioni specificate direttamente hanno ovviamente la precedenza su quelle calcolate a partire dai contenuti.

Con XFrames le cornici torneranno molto probabilmente a diventare uno dei meccanismi fondamentali nella strutturazione delle pagine web, soprattutto quando si può operare solo dal lato cliente. Se poi avranno effettivamente successo, questo lo vedremo non appena i browser inizieranno a incorporare queste nuove specifiche. Staremo a vedere.