



NITF Change request: Simplification of the model of the ‘definition list’ structure

Document history				Document URN: urn:iptc:workdoc:nm:0402:1
Revision	Issue Date	Pages	Author (revised by)	Remark
unrevised	2004-03-01	2	Laurent Le Meur, afp	
Rev a.				

Definition lists - <dl> tag –are more or less implemented in the same way as HTML definition lists, and can be inserted between two paragraphs. The ‘dl’ element contains a collection of ‘dt’ / ‘dd’ pairs; ‘dt’ stands for definition term, ‘dd’ stands for ‘definition data’, and both may be repeated inside a ‘dl’ element. ‘dt’ supports enriched text, and ‘dd’ contains enriched text in a text block.

```
<!ELEMENT dl (dt | dd)+>
<!ATTLIST dl %common-attributes; >
```

```
<!ELEMENT dt (%enriched-text;)*>
<!ATTLIST dt %common-attributes; >
```

```
<!ELEMENT dd (block)*>
<!ATTLIST dd %common-attributes; >
```

with:

```
<!ENTITY % enriched-text "#PCDATA | chron | classifier | copyrite | event | function | location | money | num | object.title | org | person | postaddr | virtloc | a | br | em | lang | pronounce | q">
```

and:

```
<!ELEMENT block ((%block.start;)? , (%block.content;)* , (%block.end;)?)>
<!ENTITY % block.content "p | hl2 | table | nitf-table | media | ol | ul | dl | bq | fn | note | pre | hr">
```

‘block’ is defined as

Partial description	A group of related containers.
Full description	This container can be used to organize a set of related containers, such as a sidebar.

The issue with this implementation is that it leads to a verbose syntax for simple definition lists, as ‘dd’ MUST contain a ‘block’ and a ‘p’ element:

```
<body.content>
  <hl2>Brian Laudrup</hl2>
  <dl>
    <dt>Name</dt>
    <dd><block><p>Brian Laudrup</p></block></dd>
    <dt>Country</dt>
    <dd><block><p>Denemark</p></block></dd>
    <dt>Date of birth</dt>
    <dd><block><p>22-02-1969</p></block></dd>
    <dt>Place of birth</dt>
    <dd><block><p>Vienna</p></block></dd>
  </dl>
</body.content>
```



HTML4 offers a leaner syntax, as the model for ‘dd’ allows for the inclusion of a simple string. This covers 90% of users’ needs.

```
<body>
  <h2>Brian Laudrup</h2>
  <dl>
    <dt>Name</dt>
    <dd>Brian Laudrup</dd>
    <dt>Country</dt>
    <dd>Danemark</dd>
    <dt>Date of birth</dt>
    <dd>22-02-1969</dd>
    <dt>Place of birth</dt>
    <dd>Vienna</dd>
  </dl>
</body>
```

Compared with other NITF elements with a similar usage, the mandatory presence of a block in ‘dd’ is also surprising. In a table, the model for ‘td’ or ‘th’ is (%enriched-text; | %block.content;)*, as is the model for ‘caption’. The same model applies for other mixed elements, like the list item ‘li’ or ‘media-caption’.

Many other NITF elements have a simple model of (%enriched-text;)*: this is the case for ‘dt’ (the definition term in a definition list), ‘h1’, ‘h2’, ‘p’, ‘q’, ‘a’, and several others.

‘block’ is only used in the three following elements: ‘body.content’, ‘bq’ (block quote) and ... ‘dd’. And currently no mixed element has a model that includes a ‘block’ element.

Consequently, the best model for ‘dd’ would seem to be either (%enriched-text)* or (%enriched-text; | %block.content;)*. The last model allows for the inclusion of tables or other useful structures, so it seems more powerful than the first one.

Even if the usage of a ‘block’ element in ‘dd’ seems like a burden, the need to make NITF evolve in a backward compatible way forces us to keep this feature.

Proposal: in the next minor revision of the NITF standard, revise the model of the ‘dd’ element in a backward compatible way, as:

```
<!ELEMENT dd (%enriched-text; | %block.content; | block)*>
<!ATTLIST dd %common-attributes; >
```

=== END of document ===