EXCHANGE TABLE MODEL DOCUMENT TYPE DEFINITION

SGML Open Technical Resolution TR 9503:1995

Harvey Bingham, Interleaf, Chair, Table Interchange Subcommittee, SGML Open

Revision date: 1996 May 8

Copyright © 1995, 1996 by SGML Open

Permission to reproduce parts or all of this information in any form is granted to SGML Open members provided that this information by itself is not sold for profit and that SGML Open is credited as the author of this information.

This SGML Open Technical Resolution defines an Exchange subset of the full CALS table model DTD described in SGML Open Technical Memorandum TM 9502:1995. This Exchange subset has been chosen as being a useful subset of the complete CALS table model such that, if an application's tables are tagged according to this subset, there is a high probability that the table will be interoperable among the great majority of SGML Open vendor products. See also the SGML Open Technical Research Paper TRP 9501:1995 on Table Interoperability: Issues for the CALS table model.

Note that the set of element and attribute declarations in the markup declaration module section of this document partially defines the Exchange table model. However, the model is not well-defined without the accompanying natural language description of the semantics (meanings) of these various elements, attributes, and attribute values. The semantic writeup, in the section following that containing the markup declaration module, must be used in conjunction with the element and attribute declarations.

Technical Resolution 9503:1995
Committee Draft: 1995 November 21
Final Draft Technical Resolution: 1996 January 12
Final Technical Resolution: 1996 May 8

Introduction

This SGML Open Technical Resolution defines an Exchange subset of the full CALS table model DTD described in SGML Open Technical Memorandum TM 9502:1995. This Exchange subset has been chosen as being a useful subset of the complete CALS table model such that, if an application's tables are tagged according to this subset, there is a high probability that the table will be interoperable among the great majority of SGML Open vendor products. See also the SGML Open Technical Research Paper TRP 9501:1995 on Table Interoperability.

Specific changes of substance between this Exchange Model and the CALS table model contained in the Example DTD published in MIL-HDBK-28001 (30 June 95) include:

1. allow content model for table to include title on the graphic+ form of table content when there are no tgroups;
2. change most explicit attribute defaults from value to #IMPLIED to allow value inheritance, possibly from a style sheet;
3. eliminate <entrytbl>;
4. eliminate <tfoot>;
5. eliminate <spanspec>;
6. eliminate the char and charoff attributes from <tgroup>;
7. eliminate the rotate attribute from <entry>;
8. eliminate references to security attributes.

Finally, this declaration set is assigned a Formal Public Identifier.

Note that the set of element and attribute declarations in the next section of this document partially defines the Exchange table model. However, the model is not well-defined without the accompanying natural language description of the semantics (meanings) of these various elements, attributes, and attribute values. The semantic writeup, in the section following that containing the markup declaration module, must be used in conjunction with the element and attribute declarations.

The following markup declaration module defines parameter entities for various parts of the model. This organization provides for more flexibility in using this module in different circumstances. However, it must be realized that overriding declarations of these parameter entities that enlarge the model (i.e., that define a superset of the model or a different model) may reduce practical interoperability. It is therefore recommended that the parameter entities be used primarily to define a subset of the default model. Any parameter entity redefinition other than to define a more constrained model than the default model must be done carefully with the knowledge of ones target toolset's capabilities and with the realization that interoperability may be reduced. The portion of the model where constraining changes only are reasonable is its geometric aspects, including some table attributes, and the elements and attributes of the table structure: tgroup through row and some entry attributes. The table content model and its exceptions other than table self-exclusion and the entry content model and its exceptions are candidates for change.

Exchange Table model DTD module

```
<!-- EXCHANGE TABLE MODEL DECLARATION MODULE -->

<!-- This set of declarations defines the Exchange Table Model as of the
     date shown in the Formal Public Identifier (FPI) for this entity.

     This set of declarations may be referred to using a public external
     entity declaration and reference as shown in the following two lines:

<!ENTITY % calstbls PUBLIC "-//SGML Open//DTD Exchange Table Model 19960430//EN">
%calstbls;

     If various parameter entities used within this set of declarations
     are to be given non-default values, the appropriate declarations
     should be given before calling in this package (i.e., before the
     "%calstbls;" reference).

     NOTE:  This set of declarations assumes a NAMELEN of 32 as is used in
     the standard CALS defined SGML declaration.
-->

<!-- This entity includes a set of element and attribute declarations
     that partially defines the Exchange table model.  However, the model
     is not well-defined without the accompanying natural language
     description of the semantics (meanings) of these various elements,
     attributes, and attribute values.  The semantic writeup, also available
     from SGML Open, should be used in conjunction with this entity.
-->

<!-- In order to use the Exchange table model, various parameter entity
     declarations are required.  A brief description is as follows:

     ENTITY NAME      WHERE USED              WHAT IT IS

     %bodyatt         In ATTLIST of:          Additional (non-table related)
                      table element(s)        attributes on the overall
                                              (wrapper) table element(s)

     %yesorno         In ATTLIST of:          An attribute declared value
                      almost all elements     for a "boolean" attribute

     %titles          In content model of:    The "title" part of the model
                      table element(s)        group for the table element(s)

     %paracon         In content model of:    The "text" (logical content)
                      <entry>                 of the model group for <entry>
```

```
%tbl.table.name  In declaration of:      The name(s) of the "table"
                 table element(s)        element(s)

%tbl.table-titles.mdl In content model of: The model group for the title
                 table elements(s)       part of the content model for
                                         table element(s)

%tbl.table-main.mdl In content model of: The model group for the main part
                 table elements(s)       (not including titles) of the
                                         content model for table element(s)

%tbl.table.mdl   In content model of:    The model group for the content
                 table elements(s)       model for table element(s),
                                         often (and by default) defined
                                         in terms of %tbl.table-titles.mdl
                                         and %tbl.table-main.mdl

%tbl.table.excep In content model of:    The exceptions for the content
                 table element(s)        model for table element(s)

%tbl.table.att   In ATTLIST of:          Additional attributes on the
                 table element(s)        table element(s)

%tbl.tgroup.mdl  In content model of:    The model group for the content
                 <tgroup>                model for <tgroup>

%tbl.tgroup.att  In ATTLIST of:          Additional attributes on the
                 <tgroup>                <tgroup> element

%tbl.tbody.att   In ATTLIST of:          Additional attributes on the
                 <tbody>                 <tbody> element

%tbl.colspec.att In ATTLIST of:          Additional attributes on the
                 <colspec>               <colspec> element

%tbl.hdft.name   In declaration of:      The name(s) of the table
                 head/foot element(s)    head and foot element(s)

%tbl.hdft.mdl    In content model of:    The model group for the content
                 head/foot element(s)    model for head/foot element(s)

%tbl.hdft.excep  In content model of:    The exceptions for the content
                 head/foot element(s)    model for head/foot element(s)

%tbl.hdft.att    In ATTLIST of:          Additional attributes on the
                 head/foot element(s)    head/foot element(s)
```

```
     %tbl.row.mdl     In content model of:    The model group for the content
                      <row>                   model for <row>

     %tbl.row.excep   In content model of:    The exceptions for the content
                      <row>                   model for <row>

     %tbl.row.att     In ATTLIST of:          Additional attributes on the
                      <row>                   <row> element

     %tbl.entry.mdl   In content model of:    The model group for the content
                      <entry>                 model for <entry>

     %tbl.entry.excep In content model of:    The exceptions for the content
                      <entry>                 model for <entry>

     %tbl.entry.att   In ATTLIST of:          Additional attributes on the
                      <entry>                 <entry> element

     This set of declarations will use the default definitions shown below
     for any of these parameter entities that are not declared before this
     set of declarations is referenced.
-->

<!-- These definitions are not directly related to the table model, but are
     used in the default CALS table model and may be defined elsewhere (and
     prior to the inclusion of this table module) in the referencing DTD. -->

<!ENTITY % bodyatt "">
<!ENTITY % yesorno 'NUMBER'  -- no if zero(s),
                                yes if any other digits value -->
<!ENTITY % titles  'title?'>
<!ENTITY % paracon '#PCDATA' -- default for use in entry content -->

<!--
The parameter entities as defined below change and simplify the CALS table
model as published (as part of the Example DTD) in MIL-HDBK-28001.  The
resulting simplified DTD has support from the SGML Open vendors and is
therefore more interoperable among different systems.

These following declarations provide the Exchange default definitions
for these entities.  However, these entities can be redefined (by giving
the appropriate parameter entity declaration(s) prior to the reference
to this Table Model declaration set entity) to fit the needs of the
current application.

Note, however, that changes may have significant effect on the ability to
interchange table information.  These changes may manifest themselves
in useability, presentation, and possible structure information degradation.
```

```
-->

<!ENTITY % tbl.table.name       "table">
<!ENTITY % tbl.table-titles.mdl "%titles;,">
<!ENTITY % tbl.table-main.mdl   "tgroup+">
<!ENTITY % tbl.table.mdl        "%tbl.table-titles.mdl; %tbl.table-main.mdl;">
<!ENTITY % tbl.table.excep      "-(%tbl.table.name)">
<!ENTITY % tbl.table.att        "
    pgwide      %yesorno;       #IMPLIED ">
<!ENTITY % tbl.tgroup.mdl       "colspec*,thead?,tbody">
<!ENTITY % tbl.tgroup.att       "">
<!ENTITY % tbl.tbody.att        "">
<!ENTITY % tbl.colspec.att      "">
<!ENTITY % tbl.hdft.name        "thead">
<!ENTITY % tbl.hdft.mdl         "row+">
<!ENTITY % tbl.hdft.excep       "">
<!ENTITY % tbl.hdft.att         "">
<!ENTITY % tbl.row.mdl          "entry+">
<!ENTITY % tbl.row.excep        "">
<!ENTITY % tbl.row.att          "">
<!ENTITY % tbl.entry.mdl        "(%paracon;)*">
<!ENTITY % tbl.entry.excep      "">
<!ENTITY % tbl.entry.att        "">

<!-- =====  Element and attribute declarations follow. =====  -->

<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % tbl.table.name       "table"
     ENTITY % tbl.table-titles.mdl "%titles;,"
     ENTITY % tbl.table-main.mdl   "tgroup+"
     ENTITY % tbl.table.mdl        "%tbl.table-titles; %tbl.table-main.mdl;"
     ENTITY % tbl.table.excep      "-(%tbl.table.name)"
     ENTITY % tbl.table.att        "
                         pgwide          %yesorno;       #IMPLIED "
-->

<!ELEMENT %tbl.table.name; - - (%tbl.table.mdl;) %tbl.table.excep; >

<!ATTLIST %tbl.table.name
        frame           (top|bottom|topbot|all|sides|none)      #IMPLIED
        colsep          %yesorno;                               #IMPLIED
        rowsep          %yesorno;                               #IMPLIED
        %tbl.table.att;
        %bodyatt;
>
```

```
<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % tbl.tgroup.mdl    "colspec*,thead?,tbody"
     ENTITY % tbl.tgroup.att    ""
-->

<!ELEMENT tgroup - O (%tbl.tgroup.mdl;) >

<!ATTLIST tgroup
        cols            NUMBER                                  #REQUIRED
        colsep          %yesorno;                               #IMPLIED
        rowsep          %yesorno;                               #IMPLIED
        align           (left|right|center|justify|char)        #IMPLIED
        %tbl.tgroup.att;
>

<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % tbl.colspec.att   ""
-->

<!ELEMENT colspec - O EMPTY >

<!ATTLIST colspec
        colnum          NUMBER                                  #IMPLIED
        colname         NMTOKEN                                 #IMPLIED
        colwidth        CDATA                                   #IMPLIED
        colsep          %yesorno;                               #IMPLIED
        rowsep          %yesorno;                               #IMPLIED
        align           (left|right|center|justify|char)        #IMPLIED
        char            CDATA                                   #IMPLIED
        charoff         NUTOKEN                                 #IMPLIED
        %tbl.colspec.att;
>

<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % tbl.hdft.name     "thead"
     ENTITY % tbl.hdft.mdl      "row+"
     ENTITY % tbl.hdft.excep    ""
     ENTITY % tbl.hdft.att      ""
-->

<!ELEMENT %tbl.hdft.name; - O (%tbl.hdft.mdl;)  %tbl.hdft.excep;>
```

```
<!ATTLIST %tbl.hdft.name;
        valign          (top|middle|bottom)                     #IMPLIED
        %tbl.hdft.att;
>

<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % tbl.tbody.att     ""
-->

<!ELEMENT tbody - O (row+)>

<!ATTLIST tbody
        valign          (top|middle|bottom)                     #IMPLIED
        %tbl.tbody.att;
>

<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % tbl.row.mdl       "entry+"
     ENTITY % tbl.row.excep     ""
     ENTITY % tbl.row.att       ""
-->

<!ELEMENT row - O (%tbl.row.mdl;) %tbl.row.excep;>

<!ATTLIST row
        rowsep          %yesorno;                               #IMPLIED
        valign          (top|middle|bottom)                     #IMPLIED
        %tbl.row.att;
>


<!--
     Default declarations previously defined in this entity and
     referenced below include:
     ENTITY % paracon           "#PCDATA"
     ENTITY % tbl.entry.mdl     "(%paracon;)*"
     ENTITY % tbl.entry.excep   ""
     ENTITY % tbl.entry.att     ""
-->

<!ELEMENT entry - O (%tbl.entry.mdl;) %tbl.entry.excep; >

<!ATTLIST entry
        colname         NMTOKEN                                 #IMPLIED
```

```
        namest          NMTOKEN                                 #IMPLIED
        nameend         NMTOKEN                                 #IMPLIED
        morerows        NUMBER                                  #IMPLIED
        colsep          %yesorno;                               #IMPLIED
        rowsep          %yesorno;                               #IMPLIED
        align           (left|right|center|justify|char)        #IMPLIED
        char            CDATA                                   #IMPLIED
        charoff         NUTOKEN                                 #IMPLIED
        valign          (top|middle|bottom)                     #IMPLIED
        %tbl.entry.att;
>
```

Semantic description for the Exchange table model

Note that the above set of element and attribute declarations in the previous section only partially defines the Exchange table model. The model is not well-defined without the accompanying natural language description of the semantics (meanings) of these various elements, attributes, and attribute values. This section containing that semantic writeup must be used in conjunction with the element and attribute declarations.

1 Element and attribute description for <table>

1.1 General

The <table> element markup identifies a table.

Values of attributes on the <table> element may at times be inherited to be used as defaults for its contained elements. Furthermore, a <table> element may have some style specification associated with it that may provide default values for some or all of its attributes. If a <table> element has no explicit specification for an attribute but does have an associated style sheet that gives a specification for this attribute, then this value shall be used as the value that is inherited from this element.

1.2 Optional attributes

1.2.1 frame outer table ruling

Describes position of outer rulings.

Declared value: sides (left and right), top (below title), bottom (after last <row>), topbot (both top and bottom), all (all of above), or none (none of above). The outer rulings appear in place of and in the space that would otherwise be taken by horizontal and vertical rulings on the outsides of those entries that appear at the edges of the table.

Default: IMPLIED (implies value from style specification if available, if not, implies “all”).

1.2.2 colsep column separators (vertical ruling)

Provides default for all <tgroup>s in this <table>. If a number other than zero, display the internal column rulings to the right of each <entry>; if zero, do not display them. Ignored for the rightmost column, where the frame setting applies.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED (implies value from style specification if available, if not, implies “1”).

1.2.3 rowsep row separators (horizontal ruling)

Provides default for all <tgroup>s in this <table>. If other than zero, display the internal horizontal row ruling below each <entry> ending in a <row>. If zero, do not display them. Ignored for the last <row> of the <table> (i.e., the last <row> of the last <tgroup> in this <table>), where the frame value applies.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED (implies value from style specification if available, if not, implies “1”).

1.2.4 pgwide make table span full page width

If zero, the maximum available width for the <table> is the (galley) width (possibly respecting current indents in force as specified by the style sheet) of the current column of the orient="port" page. If a specified value other than zero, the <table> spans the width of the entire page (possibly causing any previous multicolumn text on the page to be balanced and any extra processing associated with column balancing and page spanning to be performed). This attribute is ignored when orient="land" where all tables are treated as if pgwide="1".

Declared value: %yesorno; (NUMBER)

Default: IMPLIED (implies value from the style specification if available). In the absence of an explicit value (or one implied from the style specification), the system should attempt to format the table into a galley column if reasonable, based on explicit <colspec> colwidth values, galley column width, and possible table indentation implied by context. If any relative width specifications exist, they should continue to apply, but the unit proportion should be first based on the width uncommitted from the galley column.

In the above paragraph, “if reasonable” is intended to preclude a system generating columns so narrow that the entry content is obscured by awkward line folding, or clipping. For align=char, the content either side of the alignment character should also be considered. No explicit algorithm for determining when a table will fit is provided. Because of the uncertainty in meaning, interchange of a table should include an explicit value for pgwide.

1.2.5 %bodyatt; general body element attributes

Any of the attributes in the associated Attribute Set may be used with this element.

Default: As appropriate for each attribute in the set.

2 Element and attribute description for <tgroup>

2.1 General

Each <tgroup> effectively identifies a new portion of a <table>. The <colspec>s and the colnames in each <tgroup> are independent of any other <tgroup>. The <colspec>s in a <tgroup> apply to its <tbody>. The <colspec>s in a <tgroup> shall be in left-to-right column order. Explicit colnums in the different <colspec>s of a <tgroup> shall be unique, in ascending order left-to-right, in the range 1 to the number of <colspec>s, which is not to exceed the value of the cols attribute of <tgroup>.

The colnames in the different <colspec>s of a <tgroup> shall be unique. (A colname may have the same value as the colnum of the same <colspec> if restricted to digits.) The <colspec>s of <tbody> also apply to <thead>.

All <tgroup>s of a <table> shall have the same width, so the <table> frame can surround them uniformly. Each <thead> and <tbody> has that total width as well. For each <tgroup>, let the minimum <tgroup> width be the sum of the fixed portions of the set of <colspec> colwidths in that <tgroup> which should not exceed the indicated total width, determined from the <table> pgwide attribute. If pgwide="0", the target total width is the galley column width, possibly reduced by the current indents in force; if pgwide is non-zero, the target total width is the full page width. The minimum <table> width shall be the maximum of all the minimum <tgroup> widths.

The total width for any <tgroup> may be specified with some columns having fixed colwidths and others having proportional widths. When no proportional width are specified for a given

<tgroup>, the width of that <tgroup> shall be the sum of the fixed widths of the columns. When any proportional widths are specified, the unit proportion is determined by starting with the target table width determined from the <table> pgwide value non-zero for full width, "0" for galley column width (possibly affected by current indents). Reduce that by the sum of border widths and column ruling widths and the fixed colwidths to get the available proportional width. Sum the proportional factors N over all columns with proportional parts "N*" and including "1*" for each implicit colwidth when the value of the <tgroup> cols exceeds the number of <colspec>s in that <tgroup>. Then the unit proportion is the available proportional width divided by the sum of the proportionality factors. Note that the fixed widths for frame and colsep rulings are deemed to be small and system dependent, not essentially included as fixed parts of each colwidth in <colspec>.

If the table fixed widths take up too much of the indicated table width, then it is up to the output system to resolve the conflict. If there are multiple <tgroup>s in a single table with fixed widths (i.e., no proportional parts) such that these fixed widths differ from one another, then it is up to the output system to determine the interpretation.

Values of attributes on the <tgroup> element may at times be inherited to be used as defaults for its contained elements. Furthermore, a <tgroup> element may have some style specification associated with it that may provide default values for some or all of its attributes. If a <tgroup> element has no explicit specification for an attribute but does have an associated style sheet that gives a specification for this attribute, then this value shall be used as the value that is inherited from this element.

2.2 Required attribute

2.2.1 cols number of columns

Number of columns in the <tgroup>.

Declared value: NUMBER

2.3 Optional attributes

2.3.1 colsep column separators (vertical ruling)

Provides default for all <colspec>s in this <tgroup>. If other than zero, display the internal column rulings to the right of <entry>; if zero, do not display them. Ignored for the last (rightmost) column, where the frame setting applies. Possible default source for <colspec> or <entry>.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED (implies value from style specification if any, else from <table>)

2.3.2 rowsep row separators (horizontal ruling)

Provides default for <colspec>s in this <tgroup>. If other than zero, display the internal horizontal row ruling below each <entry>. If zero, do not display them. Ignored for the last (bottom) <row> of the <table> where the frame value applies. In the last <row> in any <tgroup> other than the last (or only) in the table, the regular <row> or <entry> rowsep specifications apply. Possible default source for <row> or <entry>.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED (implies value from style specification if any, else from <table>)

2.3.3 align horizontal alignment of table entry content

Text horizontal position within the column. Applies to text that is #PCDATA or other in-line elements, not further contained in another element with its own formatting style, such as paragraph, list, or annotation. Default source for <colspec> align. Possible default source for <entry>.

Declared value: left, (quad flush left), center (centered), right (quad flush right), justify (both quad left and right), or char (align text to the leftmost occurrence of the value of the non-null attribute char value.)

Default: IMPLIED (means use value from style specification if any, else "left".)

3 Element and attribute description for <colspec>

3.1 General

Specifies a column, a vertical portion of a <table>. The default values come from the <tgroup> or <thead> starting the current (enclosing) group. Each <colspec> is for a single column in left-to-right order, so it properly has a column number, colnum, implicitly in order starting from 1, and an optional column name (colname) by which it is known when referenced by any <entry>. Any references from <entry>s within the <thead> to colname, namest, or nameend refer to values defined by the set of <colspec>s in the containing <tgroup>. <Colspec>s from the containing <tgroup> apply to <thead> and <tbody>.

The number of columns should be determined by the cols attribute on the <tgroup> element, not by the number of <colspec>s actually defined. If the number in cols is larger than the number of <colspec>s, then additional <colspec>s of colwidth "1*" should be inferred. <Colspec>s can be numbered or unnumbered, and if numbered should be increasing in sequential order starting with 1 at the left. Unnumbered <colspec>s are interpreted as being numbered incrementally (one more than the previous column number), with the first <colspec> starting at 1. It is left up to the implementation how to handle any mismatch in <colspec> numbering, or a number of <colspec>s greater than cols. It is recommended that an authoring or editing implementation or any implementation that verifies the compliance of the table markup to this Resolution offer the option of producing a warning message when it encounters such markup. It is an error for an authoring or editing implementation to produce a table with such conflicting markup.

The char and charoff attributes are allowed on <colspec> as well as on <entry>. While an implementation should accept these attributes on either of these elements upon input, it may produce logically equivalent output that nevertheless differs with respect to these attributes and still be considered in compliance with the semantics of this resolution. If a table contains one or more <entry>s with values of either char or charoff that differ from the value for the same attribute on a <colspec> that logically contains the <entry>(s), it is left up to the implementation how this should be handled (i.e., the interpretation of such markup is beyond the scope of this model). It is recommended that an authoring or editing implementation or any implementation that verifies the compliance of the table markup to this Resolution offer the option of producing a warning message when it encounters such markup. It is an error for an authoring or editing implementation to produce a table with such conflicting markup.

3.2 Optional attributes

3.2.1 colnum column number

Number of column, counting from 1 at left of the table. The value of colnum is not useful to identify a column in an <entry>, so serves no functional purpose other than a consistency check on the order of the <colspec>s.

Declared value: NUMBER

Default: IMPLIED (colspec is the next one in order)

3.2.2 colname column name

Name of column, used to specify the position or horizontal span of columns in a row by reference in <entry> using colname, namest, and/or nameend.

The colname value could be the same as colnum, as its declared value is NMTOKEN, though there is no such requirement. The name space for colnames is different for each <tgroup>.

Declared value: NMTOKEN

Default: IMPLIED (colspec cannot be referenced without colname)

3.2.3 colwidth column width specification

Either proportional measure of the form number*, e.g., "5*" for 5 times the proportion, or "*" (which is equivalent to "1*"); fixed measure, e.g., 2pt for 2 point, 3pi for 3 pica. (Mixed measure, e.g., 2*+3pt, while allowed in the full CALS table model, is not supported in this Exchange model.) Coefficients are positive integers or fixed point numbers; for fixed point numbers, a leading (possibly 0) integer part is required, and implementations should support at least 2 decimal places. A value of "" [the null string] is treated as a proportional measure of "1*".

The fixed unit values are case insensitive. The standard list of allowed unit values is "pt" (points), "cm" (centimeters), "mm" (millimeters), "pi" (picas), and "in" (inches). The default fixed unit should be interpreted as "pt" if neither a proportion nor a fixed unit is specified.

Declared value: CDATA

Default: IMPLIED (means assume a proportional measure of "1*")

3.2.4 colsep column separators (vertical ruling)

Default for column ruling to the right of <entry>s starting in this column (within the <tgroup>).

Declared value: %yesorno; (NUMBER)

Default: IMPLIED, from <tgroup>.

3.2.5 rowsep row separators (horizontal ruling)

Default for row ruling below an <entry> starting in this column when there is neither a rowsep value on the <entry> nor on the <row> in which the <entry> occurs. Note that with a non-zero value for morerows on the entry, that ruling will be several rows lower.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED, from <tgroup>.

3.2.6 align horizontal alignment of table entry content

Text horizontal position within the column or spanning columns. Possible default source for <entry>

Declared value: left, (quad flush left), center (centered), right (quad flush right), justify (both quad left and right), or char (positioned by charoff).

Default: IMPLIED (means use value from <tgroup>)

3.2.7 char alignment character

Default source for <entry>s starting in this column. If align ="char", the value is the single alignment character source for any implied char values for <entry> immediately in this column. The char shall not be an SDATA entity. A value of "" (the null string) means there is no aligning character.

Declared value: CDATA

Default: IMPLIED, means "" (i.e., there is no aligning character).

3.2.8 charoff horizontal offset of alignment character when align=char

Default source for <entry>s starting in this column. For align="char" on an <entry> in the column, horizontal character offset is the percent of the current column width to the left of the (left edge of the) alignment character.

Declared value: NUTOKEN

Default: IMPLIED, means "50" (i.e., 50%)

4 Element and attribute description for <thead>

4.1 General

Identifies the heading <row>s of a <tgroup>, displayed as the first rows, and again at the top of any continuation after a physical break between <rows> in <tbody>.

The attributes apply from the set of <colspec>s of the containing <tgroup>.

4.2 Optional attributes

4.2.1 valign vertical alignment

Default text vertical positioning within the <entry>s. Provides default value for <row>s and <entry>s in <thead>.

Declared value: top, middle (approximately vertically centered), or bottom.

Default: IMPLIED (implies “bottom”).

5 Element and attribute description for <tbody>

5.1 General

Identifies the body of a <tgroup>.

5.2 Optional attributes

5.2.1 valign vertical alignment

Default Text vertical positioning within the <entry>s. Provides default value for <row> and <entry>s in <tbody>.

Declared value: top, middle (approximately vertically centered), or bottom.

Default: IMPLIED (implies “top”)

6 Element and attribute description for <row>

6.1 General

Identifies the row information in a <thead> or <tbody> element. Default values come from the enclosing <thead>, <tbody>, or <tgroup> attribute values for like-named attributes.

The number of columns consumed by the <entry>s in a row including their spans, and by columns encroached by an <entry> with morerows from a prior row of a <tgroup>, shall not exceed the cols attribute value in effect for this <tgroup>. It is left up to the implementation how to handle the situation when more than one <entry> would fill any column of a <row>, including by horizontal spanning or morerows vertical spanning. It is recommended that an authoring or editing implementation or any implementation that verifies the compliance of the table markup to this Resolution offer the option of producing a warning message when it encounters such markup. It is an error for an authoring or editing implementation to produce a table with such conflicting markup.

If the number of columns consumed is less than the number of columns specified in the <tgroup cols=N>, then missing <entry>s without straddling are implicitly present, and their colsep and rowsep rulings are inherited.

6.2 Optional attributes

6.2.1 rowsep Default for all <entry>s starting in this <row> that do not specify rowsep. If other than zero, display the internal horizontal row ruling below an <entry> in the <row>. If zero, do not display it. Rowsep is ignored for the last <row> of the table where the frame specification determines the ruling.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED, from the closest element in the inheritance path:

- <tgroup> then
- <table>.

6.2.2 valign vertical alignment

Text vertical positioning default for <entry>s in a <row>.

Declared value: top, middle (approximately vertically centered), or bottom.

Default: IMPLIED, from the closest element in the inheritance path:

- either <thead> or <tgroup>, whichever is the parent of this <row> then
- <tgroup>.

7 Element and attribute description for <entry>

7.1 General

Identifies an <entry> in a <row> of a <thead> or <tbody> in a <tgroup>.

Default values for colsep, rowsep, valign, align, char, and charoff may come from like-named attributes from the nearest element identified as influencing this <entry>. The values may come

from <table>, <tgroup>, <colspec> (defining namest or colname), <tbody>, or <row>. Note that <colspec> is not strictly in the element lineage.

The char and charoff attributes are allowed on <colspec> as well as on <entry>. While an implementation should accept these attributes on either of these elements upon input, it may produce logically equivalent output that nevertheless differs with respect to these attributes and still be considered in compliance with the semantics of this resolution. A table with one or more <entry>s with values of either char or charoff that differ from the value for the same attribute on any other entry in that logical column or from a <colspec> that logically contains the <entry>(s) does not conform to the model defined in this Resolution. It is left up to the implementation receiving such a table to determine how it should be handled. It is recommended that an authoring or editing implementation or any implementation that verifies the compliance of the table markup to this Resolution offer the option of producing a warning message when it encounters such markup.

An <entry> gets its defaults from its starting column. A row has no explicit <entry> in any column into which a vertical straddle occurs because of a morerows=N in an <entry> from a prior row. A spanning <entry> has no explicit repeated <entry>s in the columns to the right of the column identified by namest. An <entry> without specific starting column cannot span, and falls in the next non-straddled and non-spanned column.

The precedence for determining the column(s) for any <entry> is:

- span before individual column:
  1. namest through nameend
- individual column:
  1. namest
  2. colname if no namest
  3. implicit if neither namest or colname

The implicit resolution places the <entry> in the next available column.

There are various markup combinations which this Resolution defines as “erroneous” and referred to as “errors” in some of the following attribute descriptions. It is left up to the implementation how to handle each of these situations:

- any column referenced by name (either colname, namest or nameend) is not a colname of a <colspec> in that <tgroup>;
- any of those names are multiply defined in different <colspec>s of that <tgroup>;
- any portions of different <entry>s overlap each other;
- morerows vertical straddling extends beyond the rows appropriate to that <thead> or <tbody>;
- too many <entry>s occur in any <row>, exceeding the cols on the <tgroup>;
- any colnames used in namest and nameend to define a span in an <entry> refers to the namest column that is not to the left of the nameend column;
- an <entry> specifies a start column via <entry>’s namest or colname that is to the left of the column where the <entry> would be placed by default.

It is recommended that an authoring or editing implementation or any implementation that verifies the compliance of the table markup to this Resolution offer the option of producing a

warning message when it encounters such markup. It is an error for an authoring or editing implementation to produce a table with such markup.

If the content of an <entry> is too deep for the available depth of a page, what information is presented and how it is presented is left to be determined by the implementation.

7.2 Optional attributes

7.2.1 colname column name

Column name of <entry>. Ignore if namest is present. It is an error if colname is not defined in a <colspec> of the current <tgroup>.

Declared value: NMTOKEN

Default: IMPLIED (column(s) determined by namest or implicitly the next.)

7.2.2 namest name of starting column for this entry

Name of leftmost column of span. The value must be some colname in a <colspec> of the current <tgroup>. A namest on an entry with no nameend specification indicates the single column so named, though colname is usually used for such a specification. If neither namest nor colname occur, the <entry> goes in the next column in sequence to the right, skipping over any encroachment from a prior row via <entry morerows=N>. It is an error if the namest value is not defined in a <colspec> for the current <tgroup>.

Declared value: NMTOKEN

Default: IMPLIED (implies this attribute has no effect on spanning)

7.2.3 nameend name of ending column for this entry

Name of rightmost column of span. The value must be some colname in a <colspec> of the current <tgroup>. The column must be to the right of the column identified by namest. Nameend without namest is ignored.

Declared value: NMTOKEN

Default: IMPLIED (implies this attribute has no effect on spanning)

7.2.4 morerows number of additional rows in a vertical span

There shall be at least that many more rows in the appropriate <thead> or <tbody>. Any entries with morerows that would attempt to extend further downward is an error. The rowsep is determined when the entry is processed, in the top row if morerows is positive.

Declared value: NUMBER

Default: IMPLIED (interpret as “0”)

7.2.5 colsep column separators (vertical ruling)

If other than zero, display the internal vertical column ruling at the right of the <entry>; if zero, do not display it. Ignored for the last column of a <row>, where the frame setting applies. The colsep value is determined by the leftmost column of a spanning entry, even though its effect is on the last column of such spanning entry.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED, from the closest element in the inheritance path:

- <colspec> then

- <tgroup> then
- <table>.

Note that if a <tgroup> or <table> element has no explicit specification for this attribute but does have an associated style sheet that gives a specification for this attribute, then this value shall be used as the value that is inherited from this element.

7.2.6 rowsep row separators (horizontal ruling)

If other than zero, display the internal horizontal row ruling below the <entry>; if zero, do not display it. Ignored for the last <row> of <tgroup> where the table frame applies. Rowsep for the entry is determined in the topmost row in which the entry occurs, even though its effect may be morerows below.

Declared value: %yesorno; (NUMBER)

Default: IMPLIED, from the closest element in the inheritance path:

- <row> then
- <colspec> then
- <tgroup> then
- <table>.

Note that if a <tgroup> or <table> element has no explicit specification for this attribute but does have an associated style sheet that gives a specification for this attribute, then this value shall be used as the value that is inherited from this element.

7.2.7 align horizontal alignment of table entry content

Text horizontal position within the column or spanning columns. Applies to text that is #PCDATA or other inline elements not further contained in another element with its own formatting style, such as paragraph, list, or annotation though such elements could inherit their alignment from this value.

Declared value: left, (quad flush left), center (centered), right (quad flush right), justify (both quad left and right), or char (align on leftmost of char, positioned by charoff).

Default: IMPLIED (means use value from nearest applicable <colspec> of the nearest ancestral <tgroup>. If none, then “left”.)

7.2.8 char alignment character

If align="char", the value of the char attribute is the single alignment character on which the first to occur of this character in the <entry> is aligned. A value of "" (the null string) means there is no alignment character. If there is no alignment character or the alignment character does not occur in the <entry>, the <entry> right aligns to the left of the charoff position. The value of the char attribute shall not be an SDATA entity.

Declared value: CDATA

Default: IMPLIED, from <colspec> else there is no aligning character.

7.2.9 charoff horizontal offset of alignment character when align=char

When align="char" for this <entry>, horizontal character offset is the percent of the current column width to the left of the (left edge of the) alignment character.

Declared value: NUTOKEN

Default: IMPLIED, from <colspec> else "50" (i.e., 50%).

7.2.10 valign vertical alignment

Text vertical positioning within the <entry>.

Declared value: top, middle (approximately vertically centered), or bottom.

Default: IMPLIED from the closest ancestor with explicit specification of valign:

- <row> then
- <thead> or <tbody>.