



The DocBook Publishers Schema Version 1.1

Working Draft

5 April 2013

Specification URIs

This version:

<http://docs.oasis-open.org/docbook/publishers/publishers-wd01.html> (Authoritative)
<http://docs.oasis-open.org/docbook/publishers/publishers-wd01.pdf>
<http://docs.oasis-open.org/docbook/publishers/publishers-wd01.xml>

Previous version:

<http://docs.oasis-open.org/docbook/publishers/publishers.html>
<http://docs.oasis-open.org/docbook/publishers/publishers.pdf>
<http://docs.oasis-open.org/docbook/publishers/publishers.xml>

Latest version:

<http://docs.oasis-open.org/docbook/publishers/publishers.html> (Authoritative)
<http://docs.oasis-open.org/docbook/publishers/publishers.pdf>
<http://docs.oasis-open.org/docbook/publishers/publishers.xml>

Technical Committee:

OASIS DocBook TC [http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=docbook]

Chairs:

Norm Walsh (ndw@nwalsh.com), Mark Logic Corporation
Scott Hudson (scott.hudson@schneider-electric.com), Schneider Electric Industries SAS

Editors:

Scott Hudson (scott.hudson@schneider-electric.com), Schneider Electric Industries SAS
Nic Gibson (nicg@corbas.net), Individual
Dick Hamilton (hamilton@xmlpress.net), Individual

Additional artifacts:

This prose specification is one component of a Work Product which also includes:

- [XML Schemas](http://docs.oasis-open.org/docbook/publishers/v1.1/csd01/schemas/) [<http://docs.oasis-open.org/docbook/publishers/v1.1/csd01/schemas/>]

Related work:

This specification replaces or supersedes:

[DocBookPub-v1.0] *The DocBook Publishers Schema*. 27 August 2010. OASIS Committee Specification 01. <http://docs.oasis-open.org/docbook/specs/publishers-1.0-spec-cs-01.html>.

This specification is related to:

[DocBook-v5.1] *The DocBook Schema Version 5.1*. OASIS Committee Specification 01. <http://docs.oasis-open.org/docbook/specs/docbook-5.1-spec-os.html>.

Declared XML namespaces:

<http://docbook.org/ns/docbook>

Abstract:

For more than a decade, DocBook has provided a structured markup vocabulary for hardware and software documentation. DocBook is now widely used in both commercial and Open Source environments. DocBook has a very broad element set, and applies to much more than just technical documentation. The DocBook TC is engaged in evolving the suite of DocBook specifications. The community benefits from having a standard, open, interchangeable vocabulary in which to write structured content. DocBook has been, and will continue to be, designed to satisfy this requirement.

The OASIS DocBook Sub Committee for Publishers is chartered to develop and maintain official variants of DocBook in support of the publishing industry. Specifically, the subcommittee will focus on schema and stylesheet customizations to support: periodicals as regularly published technical notes or journals, book publishing (such as business, legal, medical, and other non-technical domains), educational textbooks, and other document types as appropriate for this industry.

The DocBook Publishers Schema is based on **[DocBook-v5.1]** and delivered in the normative RelaxNG Schema format. A DTD version will also be made available.

Status:

This **Working Draft** [<http://www.oasis-open.org/policies-guidelines/tc-process#dWorkingDraft>] (WD) has been produced by one or more TC Members; it has not yet been voted on by the TC or **approved** [<http://www.oasis-open.org/policies-guidelines/tc-process#committeeDraft>] as a Committee Draft (Committee Specification Draft or a Committee Note Draft). The OASIS document **Approval Process** [<http://www.oasis-open.org/policies-guidelines/tc-process#standApprovProcess>] begins officially with a TC vote to approve a WD as a Committee Draft. A TC may approve a Working Draft, revise it, and re-approve it any number of times as a Committee Draft.

Technical Committee members should send comments on this specification to the Technical Committee's email list. Others should send comments to the Technical Committee by using the "**Send A Comment** [http://www.oasis-open.org/committees/comments/index.php?wg_abbrev=docbook]" button on the Technical Committee's web page at <http://www.oasis-open.org/committees/docbook/>.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (<http://www.oasis-open.org/committees/docbook/ipr.php>).

Citation format:

When referencing this specification the following citation format should be used:

[DocBookPub-v1.1] *The DocBook Publishers Schema Version 1.1*. 5 April 2013. OASIS Committee Specification Draft wd01. <http://docs.oasis-open.org/docbook/publishers/publishers.html>.

Notices

Copyright © OASIS Open 2013. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at <http://www.oasis-open.org/who/intellectualproperty.php>.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of [OASIS](http://www.oasis-open.org) [http://www.oasis-open.org], the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <http://www.oasis-open.org/who/trademark.php> for above guidance.

Table of Contents

1 Introduction	5
1.1 Terminology	5
1.1.1 Key words	5
1.2 Normative References	5
1.3 Non-Normative References	6
2 The DocBook Publishers RELAX NG Schema	7
3 Changes to the DocBook Publishers schema v1.1	8
3.1 Epigraphs allow poetry	8
3.2 Allow admonitions	8
3.3 Stage Directions	8
3.4 Table Accessibility	10
4 Conformance	13

Appendixes

A Acknowledgements (Non-Normative)	14
B Accessible tables in DocBook 5.1 using the expanded CALS table model (Non-Normative)	15
B.1 Why do this?	15
B.2 Walk-through by example	15
B.2.1 Identify the table's headers	15
B.2.1.1 Identify column headers	15
B.2.1.2 Identify row headers	16
B.2.2 Explicitly associate table data with its headers	18
B.3 To do: Documentation	20
B.3.1 DocBook Schema / DocBook Publishers Schema	20
B.3.2 The Definitive Guide	21
B.3.2.1 db.cals.table and db.cals.informaltable	21
B.3.2.2 colspec	23
B.4 To do: XSL Transformations	23
C Revision History (Non-Normative)	24

1 Introduction

For more than a decade, DocBook has provided a structured markup vocabulary for hardware and software documentation. DocBook is now widely used in both commercial and Open Source environments. DocBook has a very broad element set and applies to much more than just technical documentation. The DocBook TC is engaged in evolving the suite of DocBook specifications. The community benefits from having a standard, open, interchangeable vocabulary in which to write structured content. DocBook has been, and will continue to be, designed to satisfy this requirement.

The OASIS DocBook Sub Committee for Publishers is chartered to develop and maintain official variants of DocBook in support of the publishing industry. Specifically, the subcommittee will focus on schema and stylesheet customizations to support: periodicals as regularly published technical notes or journals, book publishing (such as business, legal, medical, and other non-technical domains), educational textbooks, and other document types as appropriate for this industry.

The [DocBook Publishers subcommittee](mailto:docbook-publishers@lists.oasis-open.org) [mailto:docbook-publishers@lists.oasis-open.org] maintains the DocBook Publishers schema. Based on [\[DocBook-v5.1\]](#), the DocBook Publishers schema is normatively available as a [\[RELAX NG\]](#) Schema (with some additional Schematron assertions). A DTD version will also be made available, but is not the normative format.

The DocBook Technical Committee welcomes bug reports and requests for enhancement (RFEs) from the user community. The current list of outstanding requests is available through the [SourceForge tracker](https://sourceforge.net/tracker/?group_id=21935&atid=1037852) [https://sourceforge.net/tracker/?group_id=21935&atid=1037852] interface. This is also the preferred mechanism for submitting new requests.

The DocBook Publishers Schema v1.1 addresses the following requests for enhancement:

- [RFE 3168755](https://sourceforge.net/tracker/?func=detail&aid=3168755&group_id=21935&atid=1037852) [https://sourceforge.net/tracker/?func=detail&aid=3168755&group_id=21935&atid=1037852]: Epigraphs should allow poetry
- [RFE 3519024](https://sourceforge.net/tracker/?func=detail&aid=3519024&group_id=21935&atid=1037852) [https://sourceforge.net/tracker/?func=detail&aid=3519024&group_id=21935&atid=1037852]: Add admonitions to Publishers
- [RFE 3292583](https://sourceforge.net/tracker/?func=detail&aid=3292583&group_id=21935&atid=1037852) [https://sourceforge.net/tracker/?func=detail&aid=3292583&group_id=21935&atid=1037852]: Include mechanisms for table accessibility
- [RFE 3560731](https://sourceforge.net/tracker/?func=detail&aid=3560731&group_id=21935&atid=1037852) [https://sourceforge.net/tracker/?func=detail&aid=3560731&group_id=21935&atid=1037852]: Add stage direction markup

1.1 Terminology

1.1.1 Key words

The key words *must*, *must not*, *required*, *shall*, *shall not*, *should*, *should not*, *recommended*, *may*, and *optional* are to be interpreted as described in [\[RFC 2119\]](#). Note that for reasons of style, these words are not capitalized in this document.

1.2 Normative References

[DocBookPub-v1.0] *The DocBook Publishers Schema* 27 August 2010. OASIS Committee Specification 01. <http://docs.oasis-open.org/docbook/specs/publishers-1.0-spec-cs-01.html>

- [**DocBook-v5.1**] *The DocBook Schema Version 5.1* TBD OASIS Committee Specification 01 <http://docs.oasis-open.org/docbook/specs/docbook-5.1-spec-os.html>
- [**DocBook:TDG5.1**] *DocBook 5.1: The Definitive Guide* 26 December 2012 Norman Walsh, Richard Hamilton. <http://docbook.org/tdg51/en/>
- [**RELAX NG**] *RELAX NG Specification 3* December 2001 OASIS Committee Specification <http://relaxng.org/spec-20011203.html>
- [**RELAX NG Compact Syntax**] *RELAX NG Compact Syntax 21* November 2002 OASIS Committee Specification <http://relaxng.org/compact-20021121.html>
- [**XML**] *Extensible Markup Language (XML) 1.0 (Fourth Edition)* 16 August 2006 World Wide Web Consortium <http://www.w3.org/TR/REC-xml>
- [**XLink11**] *XML Linking Language (XLink) Version 1.1* 2005 World Wide Web Consortium <http://www.w3.org/TR/xlink11>
- [**RFC 2119**] *Key words for use in RFCs to Indicate Requirement Levels*, March 1997. S. Bradner. IETF (Internet Engineering Task Force) RFC 2119, <http://www.ietf.org/rfc/rfc2119.txt>
- [**RFC 3023**] *XML Media Types*, 2001. M. Murata, S. St. Laurent, D. Kohn. IETF (Internet Engineering Task Force) RFC 3023, <http://www.ietf.org/rfc/rfc3023.txt>
- [**DCMI Metadata Terms**] *DCMI Metadata Terms (DCMI Recommendation)* 14 January 2008 Dublin Core Metadata Initiative. All Rights Reserved. <http://www.dublincore.org/about/copyright>. <http://dublincore.org/documents/dcmi-terms/>

1.3 Non-Normative References

- [**SGML**] *ISO 8879:1986 Information processing -- Text and office systems -- Standard Generalized Markup Language (SGML)* 1986 JTC 1, SC 34
- [**W3C XML Schema**] *XML Schema Part 1: Structures* 2000 World Wide Web Consortium <http://www.w3.org/TR/xmlschema-1/>.
- [**W3C XML Datatypes**] *XML Schema Part 2: Datatypes* 2000 World Wide Web Consortium <http://www.w3.org/TR/xmlschema-2/>.
- [**Schematron**] *The Schematron Assertion Language 1.6* 2002-10-01 Rick Jelliffe Rick Jelliffe and Academia Sinica Computing Centre <http://www.ascc.net/xml/resource/schematron/Schematron2000.html>.

2 The DocBook Publishers RELAX NG Schema

The DocBook Publishers [RELAX NG Schema](http://docs.oasis-open.org/docbook/rng/) [http://docs.oasis-open.org/docbook/rng/] is distributed from the [DocBook site](http://www.oasis-open.org/docbook/) [http://www.oasis-open.org/docbook/] at [OASIS](http://www.oasis-open.org/) [http://www.oasis-open.org/]. DocBook is also available from the mirror on <http://docbook.org> [http://docbook.org/].

This subcommittee will submit additional enhancements back to the full DocBook standard as appropriate. The scope of this DocBook Publishers schema is publishing industry content. Broadly, this includes books, journals and other related publications. The DocBook Publishers schema has been written as a native RELAX NG grammar, based on [[DocBook-v5.1](#)]. This effort delivers on the following goals:

1. Build an official DocBook variant, based against the [[DocBook-v5.1](#)] schemas.
2. Address issues and enhancement requests that have arisen from experience with real-world DocBook implementations.
3. Add support for features specific to the publishing industry.

3 Changes to the DocBook Publishers schema v1.1

This version of the DocBook Publishers schema includes the following content model changes to [[DocBook-Pub-v1.0](#)] and Core DocBook. These are considered "official customizations".

3.1 Epigraphs allow poetry

Poetry is a common component of epigraphs in trade and academic publishing. The DocBook Publishers schema will allow poetry in epigraph.

The content model for epigraph will be overridden as follows:

```
db.epigraph =
  element epigraph {
    db.epigraph.attlist,
    db.epigraph.info,
    db.attribution?,
    (db.para.blocks | db.literallayout | db.extension.blocks)+
  }
```

3.2 Allow admonitions

The admonitions group was removed from the initial DocBook Publishers specification. Vocational textbook publishers have responded that admonitions are used fairly extensively. Admonitions should be allowed to avoid the workarounds that have been required so far (using sidebars with role attributes).

```
include "../docbook/src/admonitions.rnc"
```

3.3 Stage Directions

The DocBook Publishers schema supports dramatic markup, but there is no support for semantic markup of stage directions. the following elements have been added and allow the standard set of inlines:

1. Add stagedir to drama, linegroup to provide block level stage directions (between the speeches, lines, etc).
2. Add inlinestagedir to provide inline stage direction markup.

Content model:

```
div {
db.stagedir.attlist = db.common.attributes
db.stagedir.content = (db.para+)
db.stagedir =
## A stage direction.
element db:stagedir {db.stagedir.attlist, db.stagedir.content }
```



```

}

div {
  db.inlinestagedir.attlist = db.common.attributes
  db.inlinestagedir.content = (db.all.inlines | text)+
  db.inlinestagedir =
  ## inline stage direction.
  element db:inlinestagedir {db.inlinestagedir.attlist, db.inlinestagedir.content }
}

div {
  db.line.role.attribute = attribute role { text }
  db.line.attlist =
    db.line.role.attribute?
    & db.common.attributes
    & db.common.linking.attributes
  db.line =
    ## A line of dialogue or other form of communication.
    element line { db.line.attlist, (db.inlinestagedir | db.all.inlines | text)+ }
}

div {
  db.linegroup.role.attribute = attribute role { text }
  db.linegroup.attlist =
    db.linegroup.role.attribute?
    & db.common.attributes
    & db.common.linking.attributes
  db.linegroup =
    ## A grouping of lines in dialogue or other form of communication.
    element linegroup {
      db.linegroup.attlist, db.speaker*, (db.stagedir | db.line | db.all.blocks)+
    }
}

div {
  db.drama =
    ## A container for plays and other forms of dramatic presentation.
    element drama {
      db.drama.attlist,
      db.drama.info?,
      (db.stagedir | db.mediaobject | db.linegroup | db.line | db.all.blocks)+
    }
  db.drama.info = db._info.title.only
  db.drama.role.attribute = attribute role { text }
  db.drama.attlist =
    db.drama.role.attribute?
    & db.common.attributes
    & db.common.linking.attributes
}

```

3.4 Table Accessibility

In order to enhance accessibility for screen readers, the following changes have been made to the DocBook table model for [[DocBook-v5.1](#)] (and associated schemas):

1. Add rowheader to the colspec element. Indicates whether or not the entries in the respective column should be considered row headers.

Content model:

```
db.rowheader.attribute =
  ## Indicates whether or not the entries in the first column should be considered row headers.
  attribute rowheader {
    ## Indicates that entries in the first column of the table are functionally row headers
(analogous to the way that a thead provides column headers).
Applies when rowheader is used as a table attribute.
    "firstcol" |
    ## Indicates that entries of a column described via colspec are functionally row headers
(for cases with more than one column of row headers).
Applies when rowheader is used as a colspec attribute.
    "headers" |
    ## Indicates that entries in the first column have no special significance with respect to
column headers.
    "norowheader"
  }

db.colspec.attlist =
  db.colspec.role.attribute?
  & db.common.attributes
  & db.common.linking.attributes
  & db.colspec.colnum.attribute?
  & db.char.attribute?
  & db.colsep.attribute?
  & db.colspec.colwidth.attribute?
  & db.charoff.attribute?
  & db.colname.attribute?
  & db.rowsep.attribute?
  & db.align.attribute?
  & db.rowheader.attribute?
```

2. Allow scope attribute on entry. The scope attribute can contain one of "row" or "col" or "rowgroup" or "colgroup", as defined in [[DocBook-v5.1](#)].

Content model:

```
db.entry.scope.attribute =
## Specifies the scope of a header.
  attribute scope {
    ## Applies to the row
    "row"
    |
    ## Applies to the column
  }
```

```

"col"
|
## Applies to the row group
"rowgroup"
|
## Applies to the column group
"colgroup"
}

db.entry.attlist =
db.entry.role.attribute?
& db.common.attributes
& db.common.linking.attributes
& db.valign.attribute?
& db.char.attribute?
& db.colsep.attribute?
& db.charoff.attribute?
& db.entry.morerows.attribute?
& db.column-spec.attributes?
& db.rowsep.attribute?
& db.entry.rotate.attribute?
& db.align.attribute?
& db.entry.scope.attribute?

```

3. Allow headers attribute on entry. Identifies one or more entry elements that apply to its entry. The headers attribute contains an unordered set of unique space-separated tokens, each of which is an ID reference.

Example content markup:

```

<thead>
<row>
  <entry></entry>
  <entry xml:id="date1">12/12/2005</entry>
  <entry xml:id="date2">12/19/2005</entry>
  <entry xml:id="date3">12/26/2005</entry>
</row>
</thead>
<tbody>
<row>
  <entry xml:id="prog">Prognosis</entry>
  <entry headers="date1 prog">8,000</entry>
  <entry headers="date2 prog">7,000</entry>
  <entry headers="date3 prog">6,000</entry>
</row>
<row>
  <entry xml:id="act">Actual</entry>
  <entry headers="date1 act">11,123.45</entry>
  <entry headers="date2 act">11,012.34</entry>
  <entry headers="date3 act">10,987.64</entry>
</row>
</tbody>

```

Content model:

```
db.entry.headers.attribute =
  ## Specifies the entry or entries which serve as headers for this element.
  attribute headers { xsd:IDREFS }
db.entry.attlist =
  db.entry.role.attribute?
  & db.common.attributes
  & db.common.linking.attributes
  & db.valign.attribute?
  & db.char.attribute?
  & db.colsep.attribute?
  & db.charoff.attribute?
  & db.entry.morerows.attribute?
  & db.column-spec.attributes?
  & db.rowsep.attribute?
  & db.entry.rotate.attribute?
  & db.align.attribute?
  & db.entry.scope.attribute?
  & db.entry.headers.attribute?
```

4. Since there is no @summary in CALS, the table's summary should be allowed in a caption. Ensure that caption is allowed within table.

Content model:

```
element table {
  db.cals.table.attlist,
  db.cals.table.info,
  (db.alt? & db.indexing.inlines* & db.textobject*),
  (db.mediaobject+ | db.tgroup+),
  db.caption?
}
```

This change is informed by the following resources:

- <http://www.w3.org/html/wg/wiki/IssueTableHeaders>
- W3C HTML5 editor's draft (26 June 2012) scope: <http://dev.w3.org/html5/spec/the-th-element.html#the-th-element>
- W3C HTML5 editor's draft (26 June 2012) headers: <http://dev.w3.org/html5/spec/attributes-common-to-td-and-th-elements.html>
- Techniques for WCAG2.0: <http://www.w3.org/TR/WCAG20-TECHS/H43.html>
- W3C research on complex tables: <http://www.w3.org/html/wg/wiki/ComplexTables>
- WebAim Guide for accessible data tables: <http://webaim.org/techniques/tables/data>

Additional examples and guidance for Table Accessibility are provided in [Appendix B, Accessible tables in DocBook 5.1 using the expanded CALS table model \(Non-Normative\)](#).

4 Conformance

The DocBook Publishers schema is a valid [DocBook-v5.1] variant. Publications or documents shall be considered conformant to this specification if they validate against the DocBook Publishers schema (publishers.rnc or publishers.rng).

For purposes of naming and versioning, documents conformant to this specification shall identify themselves using the *version="5.1-variant publishers-1.1"* identifier, as described in <http://www.docbook.org/docs/how-to/#cust-naming>.

Extensions allow implementers to include features that are in demand by their customers. Also, extensions, often times, define new features that may migrate into future versions of this specification. However, the use of extensions can have a severe negative impact on interoperability.

Any extensions to this schema shall follow the principles and guidelines of the [DocBook-v5.1] and this Publishers specification; that is, the specifications shall be extended in a standard manner as described in <http://docbook.org/tdg5/en/html/ch05.html>.

For implementations and/or applications that contain extensions, those extensions shall be clearly described in supporting documentation, and the extensions shall be marked as such within the implementation/application.

This document's normative language is English. Translation into other languages is permitted.

Appendix A Acknowledgements (Non-Normative)

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants

- Nic Gibson, Individual
- Dick Hamilton, Individual
- Scott Hudson, Pelco by Schneider Electric (DocBook Publishers SC Chair, Editor)
- Nathalie Sequeira, Individual
- Norman Walsh, Mark Logic Corporation (DocBook TC Chair)

Appendix B Accessible tables in DocBook 5.1 using the expanded CALS table model (Non-Normative)

Usage guide to the expanded CALS table model in DocBook 5.1, with notes for documentation changes to enhance its usability.

B.1 Why do this?

The increasingly widespread usage of texts in their digital form has hugely positive implications for the inclusion of people with disabilities. Access to information is becoming much easier, especially for blind people, since digital texts can be read directly by a screen reader without any extra manipulation (e.g. conversion to Braille, scanning textbooks so they can be read aloud by corresponding software, etc.).

However, to guarantee that the blind user's experience is as smooth as possible, certain helpers must be in place to facilitate navigation and the correct interpretation of a text by screen readers. Structural markup of headings, lists and tables, for example, are central to this. Thus, DocBook innately already has much of what is needed in place to create texts that are universally accessible.

In the specific case of tables, it is vital to attribute table headers to the table data. Where this is usually achieved visually by distinct header styles, someone who cannot see these cues will need programmatic indications of which cells contain headers, and how they relate to the data presented in the table. Without these semantic markers, screen readers will read table data linearly, that is, from left to right, row by row, thus making it very hard to understand the data and its interrelations.

DocBook 5.1 and its official Publishers variant set out to offer this needed markup by expanding the CALS table model in such a way that these semantic relationships can be expressed and transferred to other formats the DocBook is converted to, very similarly to how the HTML table model already can achieve this.

The following walk-through shows the new possibilities of the CALS table model in use.

B.2 Walk-through by example

In marking up a CALS table for screen reader accessibility, there are two steps to be taken, depending on the table structure's complexity:

1. identify the table's headers
2. explicitly associate table data with its headers

B.2.1 Identify the table's headers

In very simple tables with only one row or column of headers, this mechanism alone is sufficient to allow screen readers to associate data with its respective headers.

B.2.1.1 Identify column headers

To identify column headers, the relevant rows are wrapped in the `thead` element.

Table B.1. example: a simple table with column headers

Mark's points	Peter's points	Cindy's points
11,123.45	11,012.34	10,987.64

```
<table frame="all">
  <title>example: a simple table with column headers</title>
  <tgroup cols="3">
    <thead>
      <row>
        <entry>Mark's points</entry>
        <entry>Peter's points</entry>
        <entry>Cindy's points</entry>
      </row>
    </thead>
    <tbody>
      <row>
        <entry>11,123.45</entry>
        <entry>11,012.34</entry>
        <entry>10,987.64</entry>
      </row>
    </tbody>
  </tgroup>
</table>
```

Without any markup of the table headers, a screen reader may just read the table's contents as:

Mark's points Peter's points Cindy's points 11,123.45 11,012.34 10,987.64.

With the headers marked up, screen readers will attribute the headers like this:

Mark's points 11,123.45, Peter's points 11,012.34, Cindy's points 10,987.64.

Clearly, the second variant is much more easily understood, and thus well worth the little effort required to appropriately markup the table in the DocBook source.

B.2.1.2 Identify row headers

DocBook now offers two alternatives to identify row headers in CALS tables:

- if only the first column contains row headers, set the `rowheader` attribute to the `firstcol` value in the `table` element.

Table B.2. example: a simple table with first row headers

Captions are allowed!

Mark's points	11,123.45
Peter's points	11,012.34
Cindy's points	10,987.64


```

<table frame="all" rowheader="firstcol">
  <title>example: a simple table with first row headers</title>
  <tgroup cols="2">
    <tbody>
      <row>
        <entry>Mark's points</entry>
        <entry>11,123.45</entry>
      </row>
      <row>
        <entry>Peter's points</entry>
        <entry>11,012.34</entry>
      </row>
      <row>
        <entry>Cindy's points</entry>
        <entry>10,987.64</entry>
      </row>
    </tbody>
  </tgroup>
  <caption><para>Captions are allowed!</para></caption>
</table>

```

- if more than one column contains row headers, set the `rowheader` attribute on the relevant `colspec` declarations to headers.

Table B.3. example: a table with two columns of row headers

points	Mark	11,123.45
	Peter	11,012.34
	Cindy	10,987.64

```

<table frame="all">
  <title>example: a table with two columns of row headers</title>
  <tgroup cols="3">
    <colspec colname="c1" colwidth="1.0*" rowheader="headers"/>
    <colspec colname="c2" colwidth="1.0*" rowheader="headers"/>
    <colspec colname="c3" colwidth="1.0*" />
    <tbody>
      <row>
        <entry morerows="2">points</entry>
        <entry>Mark</entry>
        <entry>11,123.45</entry>
      </row>
      <row>
        <entry>Peter</entry>
        <entry>11,012.34</entry>
      </row>
      <row>
        <entry>Cindy</entry>
        <entry>10,987.64</entry>
      </row>
    </tbody>
  </tgroup>
</table>

```

```

        </tbody>
    </tgroup>
</table>

```

Note: the examples here are mere constructions to demonstrate DocBook functionality options. In the wild, the first variant would be preferred to the second from an accessibility standpoint. Also, in the second example, the spanned entry "points" would have to include a `scope="rowgroup"` to make its attribution unmistakably clear (left out here for clarity's sake).

B.2.2 Explicitly associate table data with its headers

This step is necessary for more complex tables, for example containing row AND column headers, multiple column header rows, or spanned entries.

Analogously to the HTML model, DocBook CALS tables now offer two mechanisms to achieve this:

- `scope`

The `scope` attribute is used on table entries functioning as headers to explicitly define the range of entries they apply to (a row, a column, a row group, or a column group).

This method is best suited to tables of medium complexity.

Table B.4. example: assuring complex table accessibility using `scope`

	points	
	expected	actual
Mark	10,000	11,123.45
Peter	9,000	11,012.34
Cindy	10,000	10,987.64

```

<table frame="all" rowheader="firstcol">
<title>
    example: assuring complex table accessibility using <code>scope</code>
</title>
<tgroup cols="3">
    <colspec colname="c1" colwidth="1.0*" />
    <colspec colname="c2" colwidth="1.0*" />
    <colspec colname="c3" colwidth="1.0*" />
    <thead>
        <row>
            <entry morerows="1"></entry>
            <entry namest="c2" nameend="c3" scope="colgroup">points</entry>
        </row>
        <row>
            <entry scope="col">expected</entry>
            <entry scope="col">actual</entry>
        </row>
    </thead>

```

```

<tbody>
  <row>
    <entry scope="row">Mark</entry>
    <entry>10,000</entry>
    <entry>11,123.45</entry>
  </row>
  <row>
    <entry scope="row">Peter</entry>
    <entry>9,000</entry>
    <entry>11,012.34</entry>
  </row>
  <row>
    <entry scope="row">Cindy</entry>
    <entry>10,000</entry>
    <entry>10,987.64</entry>
  </row>
</tbody>
</tgroup>
</table>

```

As a basic table, it would be read by a screen reader like so:

points expected actual. Mark 10,000 11,123.45. Peter 9,000 11,012.34. Cindy 10,000 10,987.64.

which really does not make much sense at all. But by using the markup as shown above, a screen reader will be able to see order in the chaos and will read:

Mark points expected 10,000 Mark points actual 11,123.45. Peter points expected 9,000 Peter points actual 11,012.34. Cindy points expected 10,000 Cindy points actual 10,987.64.

While this leans slightly towards the verbose, it is also by far more clear, allowing for quick understanding even if the reader cannot see the table's spatial layout.

- `xml:id` and `headers`

With this dynamic duo, a very granular association of table data to its headers can be accomplished. The header entries are identified via `xml:id` attributes, which are then referenced in the `headers` attribute on data entries.

This method is best suited for very complex tables that cannot be otherwise simplified.

Table B.5. example: assuring complex table accessibility using headers

	points	
	expected	actual
Mark	10,000	11,123.45
Peter	9,000	11,012.34
Cindy	10,000	10,987.64



```

<table frame="all" rowheader="firstcol">
<title>
  example: assuring complex table accessibility using <code>headers</code>
</title>
<tgroup cols="3">
  <colspec colname="c1" colwidth="1.0*" />
  <colspec colname="c2" colwidth="1.0*" />
  <colspec colname="c3" colwidth="1.0*" />
  <thead>
    <row>
      <entry morerows="1"></entry>
      <entry namest="c2" nameend="c3" xml:id="pts">points</entry>
    </row>
    <row>
      <entry xml:id="exp" headers="pts">expected</entry>
      <entry xml:id="act" headers="pts">actual</entry>
    </row>
  </thead>
  <tbody>
    <row>
      <entry xml:id="name1">Mark</entry>
      <entry headers="name1 exp pts">10,000</entry>
      <entry headers="name1 act pts">11,123.45</entry>
    </row>
    <row>
      <entry xml:id="name2">Peter</entry>
      <entry headers="name2 exp pts">9,000</entry>
      <entry headers="name2 act pts">11,012.34</entry>
    </row>
    <row>
      <entry xml:id="name3">Cindy</entry>
      <entry headers="name3 exp pts">10,000</entry>
      <entry headers="name3 act pts">10,987.64</entry>
    </row>
  </tbody>
</tgroup>
</table>

```

B.3 To do: Documentation

To make the documentation clear concerning the different attribute values for rowheader depending on context, I suggest the following wording:

B.3.1 DocBook Schema / DocBook Publishers Schema

db.rowheader.attribute =

Indicates whether or not the entries in columns should be considered row headers
 attribute rowheader {

Indicates that entries in the first column of the table are functionally

```

    row headers (analogous to the way that a thead provides column headers).
    Applies when rowheader is used as a table attribute.
    "firstcol"
    |
    ## Indicates that entries of a column described via colspec are functionally
    row headers (for cases with more than one column of row headers).
    Applies when rowheader is used as a colspec attribute.
    "headers"
    |
    ## Indicates that entries in columns have no special significance
    with respect to column headers.
    "norowheader"
}

```

B.3.2 The Definitive Guide

Being as it is the definitive guide, it may be useful to include a usage guide along the lines of the one above. However, I could not find "the perfect spot" for this.

The same changes as indicated here for the Definitive Guide are to be made in the appropriate spots of the DocBook Publishers Definitive Guide.

B.3.2.1 [db.cals.table](http://www.docbook.org/tdg51/en/html/cals.table.html) and [db.cals.informatable](http://www.docbook.org/tdg51/en/html/cals.informatable.html)

rowheader Indicates whether or not the entries in the first column should be considered row headers

Enumerated values	Explanation
"firstcol"	Indicates that entries in the first column of the table are functionally row headers (analogous to the way that a thead provides column headers).
"headers"	Do not use in the table context. (For use in the context of colspec only.)
"norowheader"	Indicates that entries in the first column have no special significance with respect to column headers.

Also, the sample CALS table needs reworking. Because of the complex spanned entries it contains, the example currently in the Guide should use `xml:id / headers` to associate table data to its headers, like so (slightly modified to approximate it to real world use cases):

```

<article xmlns='http://docbook.org/ns/docbook'>
<title>Example table</title>

<table xml:id="ex.calstable" frame='all' rowheader="firstcol">
<title>Sample CALS Table</title>
<tgroup cols='5' align='left' colsep='1' rowsep='1'>
  <colspec colname='c1' />
  <colspec colname='c2' />

```

```

<colspec colname='c3' />
<colspec colname='c4' />
<colspec colnum='5' colname='c5' />
<thead>
  <row>
    <entry xml:id="col1" morerows="1">a1</entry>
    <entry xml:id="cols2-3" namest="c1" nameend="c2" align="center">
      Horizontal Span
    </entry>
    <entry xml:id="col4" morerows="1">a4</entry>
    <entry xml:id="col5" morerows="1">a5</entry>
  </row>
  <row>
    <entry xml:id="col2" headers="cols2-3">a2</entry>
    <entry xml:id="col3" headers="cols2-3">a3</entry>
  </row>
</thead>
<tfoot>
  <row>
    <entry xml:id="footrow" headers="col1">f1</entry>
    <entry headers="footrow cols2-3 col2">f2</entry>
    <entry headers="footrow cols2-3 col3">f3</entry>
    <entry headers="footrow col4">f4</entry>
    <entry headers="footrow col5">f5</entry>
  </row>
</tfoot>
<tbody>
  <row>
    <entry xml:id="datarow1" headers="col1">b1</entry>
    <entry headers="datarow1 cols2-3 col2">b2</entry>
    <entry headers="datarow1 cols2-3 col3">b3</entry>
    <entry headers="datarow1 col4">b4</entry>
    <entry headers="datarow1 datarow2 col5" morerows='1' valign='middle'>
      <para>Vertical Span</para>
    </entry>
  </row>
  <row>
    <entry xml:id="datarow2" headers="col1">c1</entry>
    <entry headers="datarow2 datarow3 cols2-3 col2 col3"
      namest="c2" nameend="c3"
      align='center' morerows='1' valign='bottom'>
      Span Both
    </entry>
    <entry headers="datarow2 col4">c4</entry>
  </row>
  <row>
    <entry xml:id="datarow3" headers="col1">d1</entry>
    <entry headers="datarow3 col4">d4</entry>
    <entry headers="datarow3 col5">d5</entry>
  </row>
</tbody>
</tgroup>
</table>

```

</article>

The resulting table then looking like this:

Table B.6. Sample CALS Table

a1	Horizontal Span		a4	a5
	a2	a3		
b1	b2	b3	b4	Vertical Span
c1	Span Both		c4	
d1			d4	
f1	f2	f3	f4	f5

B.3.2.2 colspec [<http://www.docbook.org/tdg51/en/html/colspec.html>]

rowheader

Indicates whether or not the entries in the specified column should be considered row headers

Enumerated values	Explanation
"firstcol"	Do not use in the <code>colspec</code> context. (For use in the context of <code>table</code> only.)
"headers"	Indicates that entries in the specified column are functionally row headers (analogous to the way that a <code>thead</code> provides column headers).
"norowheader"	Indicates that entries in the column have no special significance with respect to column headers.

B.4 To do: XSL Transformations

The following transformations are to be added:

- HTML
 - if the `colspec` for an entry's column specifies `rowheaders`, `entry --> th`
 - assure that empty entries in row-or column-header groups are transformed to empty `td`'s
 - transfer the `scope` and `headers` attribute into the HTML as is.
- ePub

How must tables be marked up in ePub to be screen reader accessible?
- FO

How must tables be marked up in PDF to be screen reader accessible?

 - transform `<caption>` elements
 - transform `colspec`-defined header columns (analogously to `firstcol`, as already implemented?)

Appendix C Revision History (Non-Normative)

Revision 0.6	5 April 2013	sch
Fixed typos reported by Loren Cahlander.		
Revision 0.5	29 March 2013	sch
Fixed content model of inlinestagedir to (db.all.inlines text)+		
Added Nathalie's <i>Accessible tables in DocBook 5.1 using the expanded CALS table</i> as an appendix.		
Revision 0.4	25 March 2013	sch
Added stagedir and inlinestagedir elements.		
<ul style="list-style-type: none">• RFE 3560731 [https://sourceforge.net/tracker/?func=detail&aid=3560731&group_id=21935&atid=1037852]: Add stage direction markup		
Revision 0.3	6 March 2013	sch
Incorporated feedback from Dick Hamilton. Simplified description for rowheader.		
Revision 0.2	22 February 2013	sch
Incorporated feedback and better table accessibility examples from Nathalie Sequeira.		
Revision 0.1	20 February 2013	sch
Fixed the following RFEs:		
<ul style="list-style-type: none">• RFE 3168755 [https://sourceforge.net/tracker/?func=detail&aid=3168755&group_id=21935&atid=1037852]: Epigraphs should allow poetry• RFE 3519024 [https://sourceforge.net/tracker/?func=detail&aid=3519024&group_id=21935&atid=1037852]: Add admonitions to Publishers• RFE 3292583 [https://sourceforge.net/tracker/?func=detail&aid=3292583&group_id=21935&atid=1037852]: Include mechanisms for table accessibility		