DocBook Version 5.1

Committee Specification Draft 01

Public Review Draft 01

11 Feb 2015

Specification URIs:

This version:

http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/docbook-v5.1-csprd01.html (Authoritative)
http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/docbook-v5.1-csprd01.xml
http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/docbook-v5.1-csprd01.pdf

Previous version:
None

Latest version:

http://docs.oasis-open.org/docbook/docbook/v5.1/docbook-v5.1.html (Authoritative)
http://docs.oasis-open.org/docbook/docbook/v5.1/docbook-v5.1.xml
http://docs.oasis-open.org/docbook/docbook/v5.1/docbook-v5.1.pdf

Technical Committee:
OASIS DocBook Technical Committee

Chair:
Norman Walsh (norman.walsh@marklogic.com), MarkLogic Corporation

Editor:
Norman Walsh (norman.walsh@marklogic.com), MarkLogic Corporation

Additional artifacts:

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

- RELAX NG Schemas accessible from http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/schemas/rng/
- Schematron Schemas accessible from http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/schemas/sch/
- An XML Catalog accessible from http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/schemas/catalog.xml
- An NVDL Schema accessible from http://docs.oasis-open.org/docbook/docbook/v5.1/csprd01/schemas/docbook.nvdl
- DocBook V4.x conversion tools accessible from http://docs.oasis-
Related work:

This specification replaces or supersedes:

  
  http://docbook.org/specs/docbook-5.0-spec-os.html

Declared XML namespaces:

  http://docbook.org/ns/docbook

Abstract:

DocBook is a general purpose [XML] schema particularly well suited to books and papers about computer hardware and software (though it is by no means limited to these applications).

The Version 5.1 release introduces assemblies for topic-oriented authoring. It also addresses a selection of bugs and feature requests.

The Technical Committee provides the DocBook 5.1 schema in other schema languages, including W3C XML Schema and an XML DTD, but the RELAX NG Schema is the normative schema.

Status:

This document was last revised or approved by the DocBook Technical Committee on the above date. The level of approval is also listed above. Check the current location noted above for possible later revisions of this document. This document is updated periodically on no particular schedule.

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" 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).

Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=docbook#technical.

Citation format:

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


Notices
1. Introduction

DocBook is general purpose XML schema particularly well suited to books and papers about computer hardware and software (though it is by no means limited to these applications).

The DocBook Technical Committee maintains the DocBook schema. Starting with V5.0, DocBook is normatively available as a [RELAX NG] Schema (with some additional [Schematron] assertions). W3C XML Schema and Document Type Definition (DTD) versions are also available.

The Version 5.1 introduces assemblies for topic-oriented authoring and addresses a selection of bugs and feature requests.

The DocBook Technical Committee welcomes bug reports and requests for enhancement (RFEs) from the user community. Please send comments and requests for enhancement to the DocBook comments list, docbook-comment@lists.oasis-open.org mailing list. Outstanding requests can be seen in the archives as well as in the SourceForge tracker interface.

1.1. Terminology

The key words must, must not, required, shall, shall not, should, should not, recommended, may, and optional in this OASIS Committee Specification; are to be interpreted as described in [RFC 2119].

1.2. Normative References


1.3. Non-Normative References


2. The DocBook RELAX NG Schema

The DocBook RELAX NG Schema (and associated non-normative schemas and tools) are distributed with this specification at the locations described above under “Additional artifacts”. DocBook is also available from the mirror on http://docbook.org/.

DocBook includes the following schemas:

- `docbook`, the DocBook V5.1-csprd01 schema.
- `docbookxii`, the DocBook XInclude V5.1-csprd01 schema. (This schema allows XInclude elements in many places.)
- `assembly`, the DocBook Assembly V5.1-csprd01 schema.

2.1. Assemblies

One modern school of thought on technical documentation stresses the development of independent units of documentation, often called topics, rather than a single narrative. Instead of writing something that DocBook users would easily recognize as a book consisting of a preface, several consecutive chapters, and a few appendices, the author (or authors) write a set of discrete topics covering different aspects of the system as if they were wholly independent.

In a typical online presentation system, for example the world wide web or online help, each topic is a page that stands alone. Except, of course, that just as no man is an island, no topic is completely unrelated to the other topics that are available.

From any given topic, there may be topics of obviously related interest. The nature of the relationships may vary. Some topics are related by physical proximity (if you're interested in the ink cartridges in a printer, you may also be interested in the print head), others by their procedural
nature (adding or replacing memory, adding or replacing a hard drive, or even changing the CPU are all topics that might logically follow a topic that describes how to open the computer case).

In a single narrative, it is the responsibility of the author to manage these relationships. He or she can reasonably assume that anyone reading chapter 4 has read chapters 1, 2, and 3. If the reader needs to be directed elsewhere, a cross reference can be used (for example, “for more information on paper jams, see Section 3.5, The Paper Path”).

In a topic-oriented system, authors are explicitly instructed to write independent units. No linear order can be assumed and many forms of explicit cross-reference are discouraged.

Documentation managers treat the library of available topics very much as programmers treat libraries of available functions. Just as any given program can pick and choose from the available libraries, the documentation for any given system can pick and choose from the available topics.

If you imagine a large documentation group managing the documentation for several related systems (different models of printer, different configurations of a software system, computers assembled from different components, etc.) it’s easy to see the appeal of topic-oriented authoring.

In a successful deployment, you might find a library of say 1,000 topics which, taken together, document five or six related systems, each of which uses 700-800 topics. Some topics are used in every system, many are used in several systems, and a small number of topics are unique to a specific system.

In order to make such a documentation platform functional, you need not only the individual topics, but also some sort of “map” or “assembly” file that describes which topics from the library are used, what relationships exist between them and, at least for print presentation, what linear order is to be imposed upon them.

DocBook uses an assemblies for this purpose.

3. Identifying DocBook Documents and Schemas

Historically, when DocBook was defined by a DTD, DocBook documents could be identified by the presence of standard public and/or system identifiers in the document type declaration. RELAX NG, the normative schema language for DocBook V5.0, does not provide any equivalent mechanism.

For systems that can make use of public identifiers, e.g., systems where the informative DTD is being used, the following public identifier should be used for DocBook V5.1: “-//OASIS//DTD DocBook V5.1//EN//XML”.

4. Conformance

This specification normatively defines DocBook V5.1 with a RELAX NG grammar and a set of Schematron assertions. A conformant DocBook V5.1 document must be valid according to both the grammar and the assertions.

The reference documentation describes additional constraints and processing expectations. A conformant DocBook V5.1 document should respect those constraints and anticipate those processing expectations.

5. Release Notes

See http://www.relaxng.org/ for a list of tools that can validate an XML document using RELAX NG.
Note that not all products are capable of evaluating the Schematron assertions in the schema.

A. The DocBook Media Type

This appendix registers a new MIME media type, “application/docbook+xml”.

1. Registration of MIME media type
application/docbook+xml

MIME media type name:
   application

MIME subtype name:
   docbook+xml

Required parameters:
   None.

Optional parameters:
   charset
   This parameter has identical semantics to the charset parameter of the application/xml media type as specified in [RFC 3023] or its successors.

Encoding considerations:
   By virtue of DocBook XML content being XML, it has the same considerations when sent as “application/docbook+xml” as does XML. See [RFC 3023], Section 3.2.

Security considerations:
   Several DocBook elements may refer to arbitrary URIs. In this case, consider the security issues of RFC 2396, section 7.

Interoperability considerations:
   None.

Published specification:
   This media type registration is for DocBook documents as described by [DocBook 5: TDG].

Applications which use this media type:
   There is no experimental, vendor specific, or personal tree predecessor to “application/docbook+xml”, reflecting the fact that no applications currently recognize it. This new type is being registered in order to allow for the deployment of DocBook on the World Wide Web as a first class XML application.

Additional information:
   Magic number(s):
There is no single initial octet sequence that is always present in DocBook documents.

**File extension(s):**

DocBook documents are most often identified with the extension “.xml”.

**Macintosh File Type Code(s):**

TEXT

**Person & email address to contact for further information:**

Norman Walsh, <ndw@nwalsh.com>.

**Intended usage:**

COMMON

**Author/Change controller:**

The DocBook specification is a work product of the DocBook Technical Committee at OASIS.

### 2. Fragment Identifiers

For documents labeled as “application/docbook+xml”, the fragment identifier notation is exactly that for “application/xml”, as specified in [RFC 3023] or its successors.

### B. Acknowledgements

The following individuals have participated in the creation of this specification and are gratefully acknowledged: Steve Cogorno, Gary Cornelius, Adam Di Carlo, Paul Grosso, Dick Hamilton, Nancy Harrison, Scott Hudson, Mark Johnson, Gershon Joseph, Jirka Kosek, Larry Rowland, Michael Smith, Robert Stayton (Secretary), Norman Walsh, (Chair, Editor).

### C. Revision History

#### 1. Changes in DocBook V5.1CR3

This release contains a bug fix.

1. **Fixed** issue #305; made navigational components optional in sect1.

#### 2. Changes in DocBook V5.1CR2

This release contains bug fixes and improvements over V5.1CR1.

1. **Use** final ITS 2.0 schemas.

2. **Fixed** issue #303; moved multimediaparam into the *data elements and allow the *data elements to be repeated.

3. **Added** RDFa Lite attributes to DocBook; removed the separate customization layer.
4. **Added** source for catalog.xml.

### 3. Changes in DocBook V5.1CR1

This release contains bug fixes and improvements over V5.0.

1. **Updated** the db4-upgrade script.
2. **Added** an RDFa Lite extension schema.
3. **Merged** ITS changes.
4. **Fixed** issue #300; added a class to see/seealso to handle the 'under' case.
5. **Fixed** issue #277; added a result element.
6. **Added** @its:version, improved better handling of extensibility.
7. **Merged** pull request #5 from kosek/master.
8. **Updated** ITS to support ITS 2.0
9. **Fixed** issue #298; don't allow secondary without primary in indexterm.
10. **Fixed** issue #295; allow navigation components at the beginnings of sections.
11. **Fixed** issue #293; removed spurious, duplicate 'other' value.
12. **Attempt** to implement the whole proposal for accessability attributes in CALS tables.
13. **Fixed** issue #293; allow admonitions in formal objects.
14. **Fixed**: issue #299; allow articles in sets.
15. **Added** scope attribute to CALS tables.
16. **Removed** format attribute from output element; the standard effectivity attribute outputformat can be used instead.
17. **Added** outputformat as an effectivity attribute.
18. **Added**: AltGr and Return to keycap class values.
19. **Renamed** fileref attribute to href in on resources in assemblies.
20. **Fixed** bug in Schematron assertions about XLInk, thanks to Hussein Shafie
21. **Fixed** issue #292; added pgwide to informalexample and informalequation.
22. **Made** info on structure and module optional in assemblies.
23. **Implemented** recent TC decisions about assemblies.
24. **Adopted** the recent proposals to add attributes/parameters to audio and video objects.
25. **Fixed** reference to broken pattern; make sure linking attributes are on areas.
26. **Fixed** issue #285; made content optional in components and sections.
27. **Allow** link in extendedlink, in preparation for arc and locator being removed in V6.0.
28. **Added** extendedlink changes to the V6.0 future use comments.
29. **Fixed** issue #289; allow multiple procedure elements in task.
30. **Fixed** issue #288; allow tag elements to nest
31. **Reworked** XLink attributes to support simple/extended links.
32. **Added** pattern for imagedata, SVG, and MathML content (so that it can be extended by the XInclude schema).
33. **Added** XInclude to images and equations; allow foreign, namespace-qualified attributes on the xi:include element.
34. **Fixed** issue #276; broaden content model of contrib.
35. **Fixed** issue #282; update HTML informaltable attributes.
36. **Fixed** issue #283; allow production to contain rhs+.
37. **Fixed** issue #284; support ISTC as a biblioid class.
38. **Attempt** to implement Larry's latest suggestions about assemblies.
39. **Fixed** issue #281; allow xi:include in set.
40. **Fixed** issue #280; added securitycontext and other to systemitem.
41. **Fixed** issue #279; allow dedication in article.
42. **Changed** Schematron namespace to official ISO Schematron URI.
43. **Allow** topic in chapter and appendix (as an alternative to narrative content) per May 2010 TC meeting.
44. **Fixed** content model of book and part to make topic an alternative, not part of the component mixture.
45. **Allow** the other major components of an assembly to be top level elements (so they can be stored in separate files, for example).
46. **Allow** an assembly without any structure elements.
47. **Tweak** assembly schemas.
48. **Allow** override element in assemblies.
49. **Generalized** toc/index to db.navigation.components in assembly structure and module for consistency
50. **Updated**: in assembly, if at least one resource is required, then at least one structure should be required as well.
51. **Removed** description attribute from assemblies (no content in attributes!); added some refpurpose documentation for attributes and attribute values.
52. **Added** refpurpose for type attribute.