

1. ATL Transformation Example

1.1. Example: BibTeXML → DocBook

The BibTeXML to DocBook example describes a transformation of a BibTeXML model to a DocBook composed document. BibTeXML [1] is an XML-based format for the BibTeX bibliographic tool. DocBook [2], as for it, is an XML-based format for document composition.

The aim of this transformation is to generate, from a BibTeXML file, a DocBook document that presents the different entries of the bibliographic file within four different sections. The first section provides the full list of bibliographic entries. The second section provides the sorted list of the different authors referenced in the bibliography. The third section presents the titles of the bibliography titled entries (in a sorted way). Finally, the last section provides the list of referenced journals (in article entries).

1.1.1. Metamodels

This transformation is based on a simplified BibTeXML metamodel which only deals with the mandatory fields of each BibTeX entries (for instance, author, year, title and journal for an article entry). The considered metamodel is presented in Figure 1, and provided, in km3 format [3], in Appendix I. It has been designed in such a way that it should be easily extensible to handle optional fields (with minor modifications).

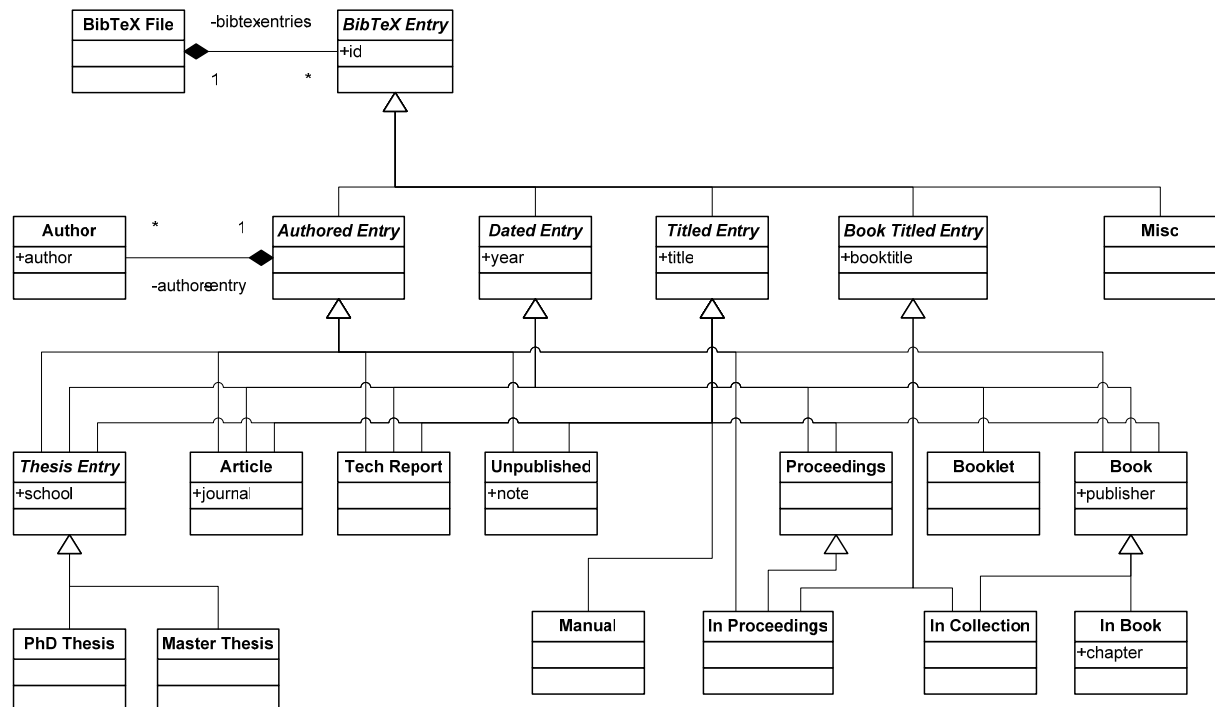


Figure 1. The BibTeXML metamodel

A bibliography is modeled by a BibTeX File element. This element is composed of BibTeX Entries which are each associated with an id. All entries inherit, directly or indirectly, of the abstract BibTeX

Entry element. The abstract classes Authored Entry, Dated Entry, Titled Entry and Book Titled Entry, as well as the Misc entry, directly inherit of BibTeX Entry.

There are 13 possible entry types: PhD Thesis, Master Thesis, Article, Tech Report, Unpublished, Manual, In Proceedings, Proceedings, Booklet, In Collection, Book, In Book and Misc. Concrete BibTeX entries inherits from some of these abstract classes according to their set of mandatory fields. An authored entry may have several authors (at least one). It is possible to note that the Misc entry has no mandatory field.

The transformation also relies on a limited subset of the DocBook definition. The metamodel considered here is described in Figure 2, and provided in Appendix II in km3 format.

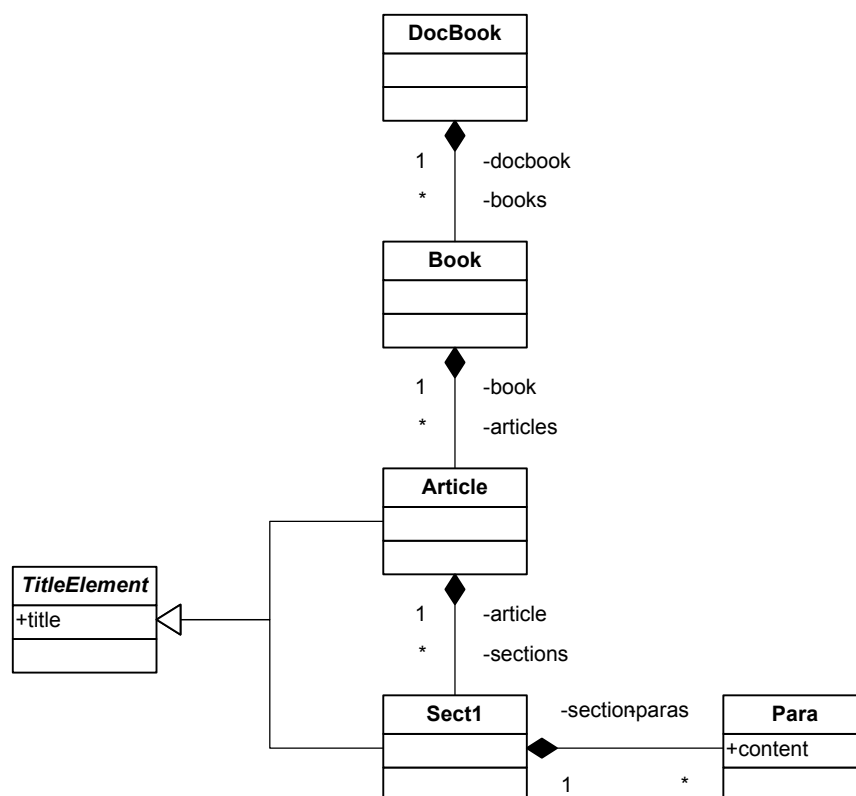



Figure 2. The DocBook metamodel

Within this metamodel, a DocBook document is associated with a DocBook element. Such an element is composed of several Books that, in their turn, are composed of several Articles. An Article is composed of Sections (“Sect1”) that are ordered. A Section is composed of Paragraphs (“Para”) that are also ordered within each section. Both Article and Section inherit from the TitledElement abstract class.

1.1.2. Rules Specification

These are the rules to transform a BibTeXML model to a DocBook model:

- For the root BibTeX File element, the following elements are created:
 - A DocBook element;
 - A Book element (linked to the DocBook);

	ATL TRANSFORMATION EXAMPLE	
	BibTeXML to DocBook	Date 04/03/2005

- An Article element (linked to the Book), which title is “BibTeXML to DocBook”;
- 4 Sections elements (linked to the Article), which titles respectively are
 - “References List” for Section 1
 - “Author List” Section 2
 - “Titles List” Section 3
 - “Journals List” Section 4
- For each *distinct* Author, a Paragraph is created
 - Its content is the author name;
 - It is linked to Section 2;
- For each untitled BibTeX Entry (entries that does not inherit from the Titled Entry class), a Paragraph is created
 - Its content contains all available information on the Entry;
 - It is linked to Section 1;
- For each Titled Entry, except Articles, the following elements have to be created
 - A Paragraph, linked to Section 1, that contains all information on the Titled Entry;
 - A Paragraph, linked to Section 3, that contains the title of the Entry (each title only appears once);
- For each Article, the following elements have to be created
 - A Paragraph, linked to Section 1, that contains all information on the Titled Entry;
 - A Paragraph, linked to Section 3, that contains the title of the Entry (each title only appears once);
 - A Paragraph linked to Section 4, which contains the name of the journal (each journal only appears once).

1.1.3. ATL Code

This ATL code for the BibTeXML to DocBook transformation consists of 4 helpers and 9 rules. Among helpers, `authorSet` provides a subset of Author elements in which a same author name appears only once. In the same way, `titledEntrySet` provides a subset of TitledEntry elements in which a same title appears only once. Finally, `articleSet` provides a subset of Article elements in which a same journal name appears only once. According to the type of each entry, the last helper, `buildEntryPara`, builds the content of the paragraph that will be associated, in the first section, with the entry.

The rule `Main` allocates the structure of the DocBook model. This rule makes use of the `“thisModule.resolveTemp(e, str)”` method (in outputs `“se3”` and `“se4”`). In the `“se3”` output, this method makes it possible to handle the output elements generated in the `‘titled_para’` output of rule `TitledEntry_NoArticle` (the one matching the `titledEntrySet` elements). The method is used in the same way in output `“se4”` to handle the output elements generated in the `‘journal_para’` output of rule `Article`. The rule `Author` allocates the DocBook paragraphs that compose the second section (list of the distinct authors).

Since, with ATL, a same source element cannot be matched more than once, contents of sections 1, 3 and 4 are generated by the 7 following rules: UntitledEntry, TitledEntry_Title_NoArticle, TitledEntry_NoTitle_NoArticle, Article_Title_Journal, Article_NoTitle_Journal, Article_Title_NoJournal and Article_NoTitle_NoJournal. The rule UntitledEntry allocates a part of the first section (for all untitled entries).

The rule TitledEntry_Title_NoArticle deals with titled entries that belong to the titledEntrySet set and are not Articles. It allocates the corresponding paragraphs in the first section, as well as the paragraphs of the third section that contains distinct titles. TitledEntry_NoTitle_NoArticle is a similar rule that applies to entries that do not belong to the titledEntrySet set and are not Articles. As a consequence, it only creates paragraph in the first section.

The rule Article_Title_Journal allocates paragraphs in the first, the third and the fourth sections. This rule considers Article entries that belong both to articleSet and titledEntrySet sets. Article_NoTitle_Journal is a similar rule that only applies to articles that belong to the articleSet set, but not to the titledEntrySet one. It therefore creates paragraphs in the first and the fourth sections, but not in the third one. In the same way, Article_Title_NoJournal applies to articles that belong to the titledEntrySet set, but not to the articleSet one. As a consequence, it only creates paragraphs in the first and the third sections. Finally, Article_NoTitle_NoJournal deals with articles that belong neither to the titledEntrySet set, nor to the articleSet one. It creates a single paragraph in the first section.

```
module BibTeX2DocBook;
create OUT : DocBook from IN : BibTeX;

-----
-- HELPERS -----
-----

-- This helper builds the set of distinct authors referenced in the input BibTeX
-- model.
-- Built set is sorted by author name.
-- RETURN: Sequence(BibTeX!Author)
helper def: authorSet : Sequence(BibTeX!Author) = BibTeX!Author.allInstances()-
>iterate(e; ret : Sequence(BibTeX!Author) = Sequence {} |
    if ret->collect(e | e.author)->includes(e.author) then
        ret
    else
        ret->including(e)
    endif
)->sortedBy(e | e.author);

-- This helper builds the set of distinct titles referenced in the input BibTeX
-- model.
-- Built set is sorted by title.
-- RETURN: Sequence(BibTeX!TitledEntry)
helper def: titledEntrySet : Sequence(BibTeX!TitledEntry) =
BibTeX!TitledEntry.allInstances()->iterate(e; ret : Sequence(BibTeX!TitledEntry) =
Sequence {} |
    if ret->collect(e | e.title)->includes(e.title) then
        ret
    else
        ret->including(e)
    endif
)->sortedBy(e | e.title);

-- This helper builds the set of distinct journals referenced in the input BibTeX
-- model.
-- Built set is sorted by journal name.
-- RETURN: Sequence(BibTeX!Article)
```

```

helper def: articleSet : Sequence(BibTeX!Article) = BibTeX!Article.allInstances()-
>iterate(e; ret : Sequence(BibTeX!Article) = Sequence {} |
    if ret->collect(e | e.journal)->includes(e.journal) then
        ret
    else
        ret->including(e)
    endif
->sortedBy(e | e.journal);

-- This helper builds a string containing all information on a given BibTeXEntry.
-- Content of the generated string depends on the entry type.
-- IN: BibTeX!BibTeXEntry
-- RETURN: Sequence(BibTeX!Author)
helper context BibTeX!BibTeXEntry def: buildEntryPara() : String =
    '[' + self.id + ']'
    + ' ' + self.oclType().name
    + (if self.oclIsKindOf(BibTeX!TitledEntry) then ' ' + self.title else ''
endif)
    + (if self.oclIsKindOf(BibTeX!AuthoredEntry)
        then self.authors->iterate(e; str : String = '' | str + ' ' + e.author)
        else ''
        endif)
    + (if self.oclIsKindOf(BibTeX!DatedEntry) then ' ' + self.year else '' endif)
    + (if self.oclIsKindOf(BibTeX!BookTitledEntry) then ' ' + self.booktitle else
'' endif)
    + (if self.oclIsKindOf(BibTeX!ThesisEntry) then ' ' + self.school else ''
endif)
    + (if self.oclIsKindOf(BibTeX!Article) then ' ' + self.journal else '' endif)
    + (if self.oclIsKindOf(BibTeX!Unpublished) then ' ' + self.note else ''
endif)
    + (if self.oclIsKindOf(BibTeX!Book) then ' ' + self.publisher else '' endif)
    + (if self.oclIsKindOf(BibTeX!InBook) then ' ' + self.chapter.toString() else
'' endif)
    ;

-----
-- RULES -----
-----

-- Rule 'Main'
-- This rule generates the structure of the DocBook model from a BibTeXFile element
rule Main {
    from
        bib : BibTeX!BibTeXFile
    to
        doc : DocBook!DocBook (
            books <- boo
        ),
        boo : DocBook!Book (
            articles <- art
        ),
        art : DocBook!Article (
            title <- 'BibTeXML to DocBook',
            sections_1 <- Sequence{se1, se2, se3, se4}
        ),
        se1 : DocBook!Sect1 (
            title <- 'References List',
            paras <- BibTeX!BibTeXEntry.allInstances()->sortedBy(e | e.id)
        ),
        se2 : DocBook!Sect1 (
            title <- 'Authors list',

```

```
        paras <- thisModule.authorSet
    ),
    se3 : DocBook!Sect1 (
        title <- 'Titles List',
        paras <- thisModule.titledEntrySet->collect(e |
thisModule.resolveTemp(e, 'title_para'))
    ),
    se4 : DocBook!Sect1 (
        title <- 'Journals List',
        paras <- thisModule.articleSet->collect(e |
thisModule.resolveTemp(e, 'journal_para'))
    )
}

-- Rule 'Author'
-- This rule generates a section_2 paragraph for each distinct author.
rule Author {
    from
        a : BibTeX!Author (
            thisModule.authorSet->includes(a)
        )
    to
        p1 : DocBook!Para (
            content <- a.author
        )
}

-- Rule 'UntitledEntry'
-- This rule generates a section_1 paragraph for each untitled entry.
rule UntitledEntry {
    from
        e : BibTeX!BibTeXEntry (
            not e.oclIsKindOf(BibTeX!TitledEntry)
        )
    to
        p : DocBook!Para (
            content <- e.buildEntryPara()
        )
}

-- Rule 'TitledEntry_Title_NoArticle'
-- [titledEntrySet contains a subset of TitledEntry, so that each title
-- appears only once in the set]
-- For each "no article" titled entry that belongs to titledEntrySet,
-- this rule generates:
-- * a section_1 paragraph;
-- * a section_3 paragraph.
rule TitledEntry_Title_NoArticle {
    from
        e : BibTeX!TitledEntry (
            thisModule.titledEntrySet->includes(e) and
            not e.oclIsKindOf(BibTeX!Article)
        )
    to
        entry_para : DocBook!Para (
            content <- e.buildEntryPara()
        ),
        title_para : DocBook!Para (
            content <- e.title
        )
}
```

```
-- Rule 'TitledEntry_NoTitle_NoArticle'
-- [titledEntrySet contains a subset of TitledEntry, so that each title
-- appears only once in the set]
-- For each "no article" titled entry that does not belong to titledEntrySet,
-- this rule generates:
-- * a section_1 paragraph;
rule TitledEntry_NoTitle_NoArticle {
  from
    e : BibTeX!TitledEntry (
      not thisModule.titledEntrySet->includes(e) and
      not e.oclIsKindOf(BibTeX!Article)
    )
  to
    entry_para : DocBook!Para (
      content <- e.buildEntryPara()
    )
}


-- Rule 'Article_Title_Journal'
-- [titledEntrySet contains a subset of TitledEntry, so that each title
-- appears only once in the set]
-- [articleSet contains a subset of Article, so that each journal
-- appears only once in the set]
-- For each article entry that belongs to articleSet, this rule generates:
-- * a section_1 paragraph;
-- * a section_3 paragraph;
-- * a section_4 paragraph.
rule Article_Title_Journal {
  from
    e : BibTeX!Article (
      thisModule.titledEntrySet->includes(e) and
      thisModule.articleSet->includes(e)
    )
  to
    entry_para : DocBook!Para (
      content <- e.buildEntryPara()
    ),
    title_para : DocBook!Para (
      content <- e.title
    ),
    journal_para : DocBook!Para (
      content <- e.journal
    )
}

-- Rule 'Article_NoTitle_Journal'
-- [titledEntrySet contains a subset of TitledEntry, so that each title
-- appears only once in the set]
-- [articleSet contains a subset of Article, so that each journal
-- appears only once in the set]
-- For each article entry that belongs to articleSet, this rule generates:
-- * a section_1 paragraph;
-- * a section_4 paragraph.
rule Article_NoTitle_Journal {
  from
    e : BibTeX!Article (
      not thisModule.titledEntrySet->includes(e) and
      thisModule.articleSet->includes(e)
    )
  to
    entry_para : DocBook!Para (
      content <- e.buildEntryPara()
    )
}
```

```
    ),
    journal_para : DocBook!Para (
        content <- e.journal
    )
}

-- Rule 'Article_Title_NoJournal'
-- [titledEntrySet contains a subset of TitledEntry, so that each title
-- appears only once in the set]
-- [articleSet contains a subset of Article, so that each journal
-- appears only once in the set]
-- For each article entry that belongs to articleSet, this rule generates:
-- * a section_1 paragraph;
-- * a section_3 paragraph.
rule Article_Title_NoJournal {
    from
        e : BibTeX!Article (
            thisModule.titledEntrySet->includes(e) and
            not thisModule.articleSet->includes(e)
        )
    to
        entry_para : DocBook!Para (
            content <- e.buildEntryPara()
        ),
        title_para : DocBook!Para (
            content <- e.title
        )
}

-- Rule 'Article_NoTitle_NoJournal'
-- [titledEntrySet contains a subset of TitledEntry, so that each title
-- appears only once in the set]
-- [articleSet contains a subset of Article, so that each journal
-- appears only once in the set]
-- For each article entry that belongs to articleSet, this rule generates:
-- * a section_1 paragraph;
rule Article_NoTitle_NoJournal {
    from
        e : BibTeX!Article (
            not thisModule.titledEntrySet->includes(e) and
            not thisModule.articleSet->includes(e)
        )
    to
        entry_para : DocBook!Para (
            content <- e.buildEntryPara()
        )
}
```


	ATL TRANSFORMATION EXAMPLE	
	BibTeXML to DocBook	Date 04/03/2005

I. BibTeXML metamodel in km3 format

```

package BibTeX {

class BibTeXFile {
    reference entries[*] container : BibTeXEntry;
}

class Author {
    attribute author : String;
}

abstract class BibTeXEntry {
    attribute id : String;
}

abstract class AuthoredEntry extends BibTeXEntry {
    reference authors[1-*] container : Author;
}

abstract class DatedEntry extends BibTeXEntry {
    attribute year : String;
}

abstract class TitledEntry extends BibTeXEntry {
    attribute title : String;
}

abstract class BookTitledEntry extends BibTeXEntry {
    attribute booktitle : String;
}

class Article extends AuthoredEntry, DatedEntry, TitledEntry {
    attribute journal : String;
}

class TechReport extends AuthoredEntry, DatedEntry, TitledEntry {
}

class Unpublished extends AuthoredEntry, TitledEntry {
    attribute note : String;
}

class Manual extends TitledEntry {
}

class Proceedings extends DatedEntry, TitledEntry {
}

class InProceedings extends Proceedings, AuthoredEntry, BookTitledEntry {
}

class Booklet extends DatedEntry {
}

```



**ATL
TRANSFORMATION EXAMPLE**

BibTeXML to DocBook

Date 04/03/2005

```
class Book extends AuthoredEntry, DatedEntry, TitledEntry {
  attribute publisher : String;
}

class InCollection extends Book, BookTitledEntry {
}

class InBook extends Book {
  attribute chapter : Integer;
}


class Misc extends BibTeXEntry {
}

abstract class ThesisEntry extends AuthoredEntry, DatedEntry, TitledEntry
{
  attribute school : String;
}

class PhDThesis extends ThesisEntry {
}

class MasterThesis extends ThesisEntry {
}

package PrimitiveTypes {
  datatype String;
  datatype Integer;
}
```

	ATL TRANSFORMATION EXAMPLE	
	BibTeXML to DocBook	Date 04/03/2005

II. DocBook metamodel in km3 format

```

package DocBook {

    class DocBook {
        reference books [1-*] ordered container: Book;
    }

    class Book {
        reference articles [1-*] ordered container: Article;
    }

    abstract class TitledElement {
        attribute title : String;
    }


    class Article extends TitledElement {
        reference sections_1 [1-*] ordered container : Sect1;
    }

    class Sect1 extends TitledElement {
        reference paras [1-*] ordered container: Para;
    }

    class Para {
        attribute content : String;
    }
}

package PrimitiveTypes {
    datatype String;
}

```

	ATL TRANSFORMATION EXAMPLE	
	BibTeXML to DocBook	Date 04/03/2005

References

- [1] BibTeXML, BibTeX as XML markup, <http://bibtexml.sourceforge.net/>.
- [2] DocBook: The Definitive Guide. Norman Walsh. O'Reilly & Associates, Inc. October 1999.
- [3] KM3: Kernel MetaMetaModel. Available at <http://dev.eclipse.org/viewcvs/indextech.cgi/~checkout~/gmt-home/doc/atl/index.html>.