

# ConT<sub>E</sub>Xt

**title       : Module Documentation**  
**subtitle   : Bibliographies**  
**author     : Taco Hoekwater**  
**date       : June 22, 2006**



# Contents

1	Introduction	1
1.1	General overview	1
2	Setup commands	2
2.1	Global settings: <code>\setuppublications</code>	2
2.2	How the entries are formatted: <code>\setuppublicationlist</code>	3
2.3	Setting citation options: <code>\setupcite</code>	4
2.4	Setting up BibTeX: <code>\setupbibtex</code>	5
2.5	Borrowing publications: <code>\usepublications</code>	6
2.6	Legacy database support	6
3	Citations	6
3.1	Default and explicit citations	7
3.1.1	Citation types	7
3.2	Citations with local setups	8
3.3	Invisible citations	8
4	Placing the publication list	8
5	The bbl file	9
5.1	Defining a publication	9
5.1.1	Complex fields	10
5.1.2	Simple fields	10
6	Defining a publication type layout	11
7	References	12

## 1 Introduction

The bibliographic module (`t-bib.tex`) takes care of references to publications and the typesetting of publication lists, as well as providing an interface between BibTeX and ConTeXt. This manual documents version 2006.06.16.

The bibliographic subsystem consists of the main module `t-bib.tex`; four BibTeX styles (`cont-xx.bst`); and a set of example configuration files (`bibl-xxx.tex`) that set up specific formatting styles for both the citations and the list of references.

### 1.1 General overview

A typical input file obeys following structure:

1. A call to `\usemodule[bib]`.
2. Optionally, a few setup commands for the bibliographic module.
3. A number of definitions of publications to be referenced in the main text of the article. The source of these definitions can be a combination of:
  - The `\jobname.bbl` file (automatically read at `\starttext`)
  - extra bbl files
  - a file or inline macros before `\starttext`These possibilities will be explained below. For now, it is only important to realize that of all these definitions have to be known *before* the first citation in the text.
4. `\starttext`
5. The body text, with a number of `\cite` and `\nocite` commands.

6. The list of publications, called using the command `\placepublications` or the command `\completepublications`.
7. `\stoptext`

## 2 Setup commands

Bibliographic references use a specific ‘style’, a collection of rules for the use of `\cite` as well as for the formatting that is applied to the publication list. The ConT<sub>E</sub>Xt bibliographic module expects you to define all of these style options in one single file of which the names starts with the prefix `bibl-`.

Unlike the normal situation in L<sup>A</sup>T<sub>E</sub>X, this style *also* includes the formatting of the items themselves. Because of this, the `.bbl` file is set up as a database of entries with fields.

### 2.1 Global settings: `\setuppublications`

The most important user-level command is `\setuppublications`. Most of the options to this command should be set by the bibliography style file, but a few of them are of immediate interest to the casual user as well.

Like all setup commands, thus command should be given before `\starttext`, as it sets up global information about the bibliographic references used in the document. ConT<sub>E</sub>Xt needs this information in order to function correctly.

```
\setuppublications [...,*,...]
```

<b>*</b>	<b>alternative</b>	=	ams	<u>apa</u>	apa-de	apa-fr	aps	num	num-fr	ssa								
	<b>refcommand</b>	=	author	authornum	authoryear	<u>authoryears</u>	key	num	serial	page	short							
			type	year	data	doi	url											
	<b>sorttype</b>	=	<u>cite</u>	bbl														
	<b>criterium</b>	=	all	<u>cite</u>														
	<b>numbering</b>	=	yes	<u>no</u>	short	bib												
	<b>autohang</b>	=	yes	<u>no</u>														
	<b>monthconversion</b>	=	COMMAND															

alternative	This gives the name of a bibliography style. The chosen style defines the other default options, the options given in this documentation are the defaults as they are set up by the ‘apa’ style. When this argument is given, the newly set style is read in first, before the other options are processed. Thus allowing you to override specific settings from the chosen style.
refcommand	the default option for <code>\cite</code>
sorttype	How the publications in the final publication list should be sorted. ‘cite’ means: by the order in which they were first cited in your text. ‘bbl’ tells the module to keep the relative ordering in which the publication definitions were found The current default for apa is ‘cite’
criterium	Whether to list only the referenced publications or all of them. If this value is ‘all’, then if ‘sorttype’ equals ‘cite’, this means that all referred-to publications are listed before all others, otherwise (if ‘sorttype’ equals ‘bbl’) you will just get a typeset version of the used database(s). The default for apa is ‘used’
numbering	Whether or not the publication list should be labelled and if so, how. <b>yes</b> uses the item number in the publication list as label. <b>short</b> uses the short label. <b>bib</b> uses the original number in the BibT <sub>E</sub> X database as a label. Anything else turns

	labelling off.
	The default for apa is ‘no’
autohang	Whether or not the hanging indent should be re-calculated based on the real size of the label. This option only applies if numbering is turned on.
	The default is ‘no’.
monthconversion	The presentation form of any month field, if it is entered in the database as a numeric value. The default is to typeset the number without any conversion

## 2.2 How the entries are formatted: `\setuppublicationlist`

```

\setuppublicationlist [...,*...]
```

<b>*</b>	<code>totalnumber</code>	=	NUMBER
	<code>samplesize</code>	=	TEXT
	<code>alternative</code>	=	<code>a b c ... none command</code>
	<code>criterium</code>	=	SECTION local previous <u>current</u> all
	<code>pagenumber</code>	=	yes <u>no</u>
	<code>artauthor</code>	=	<code>\normalauthor \normalshortauthor \invertedauthor</code> <code>\invertedshortauthor</code>
	<code>author</code>	=	<code>\normalauthor \normalshortauthor \invertedauthor</code> <code>\invertedshortauthor</code>
	<code>editor</code>	=	<code>\normalauthor \normalshortauthor \invertedauthor</code> <code>\invertedshortauthor</code>
	<code>namesep</code>	=	TEXT
	<code>lastnamesep</code>	=	TEXT
	<code>finalnamesep</code>	=	TEXT
	<code>firstnamesep</code>	=	TEXT
	<code>juniorsep</code>	=	TEXT
	<code>vonsep</code>	=	TEXT
	<code>surnamesep</code>	=	TEXT
	<code>authoretallimit</code>	=	NUMBER
	<code>authoretaltext</code>	=	TEXT
	<code>authoretaldisplay</code>	=	NUMBER
	<code>artauthoretallimit</code>	=	NUMBER
	<code>artauthoretaltext</code>	=	TEXT
	<code>artauthoretaldisplay</code>	=	NUMBER
	<code>editoretallimit</code>	=	NUMBER
	<code>editoretaltext</code>	=	TEXT
	<code>editoretaldisplay</code>	=	NUMBER
	<code>inherits from \setuplist</code>		

The list of publications at the end of the article is comparable with a sequence of normal ConT<sub>E</sub>Xt ‘list items’ that behaves much like the list that defines the table of contents. *In previous versions, it was in fact implemented as a ‘normal’ ConT<sub>E</sub>Xt list, but this is no longer the true.*

The module defines a set of extra options. These option names are static, they do *not* change to follow the selected ConT<sub>E</sub>Xt interface language.

The first two options provide default widths for ‘autohang’:

`totalnumber` The total number of items in the following list (used for autohang).  
`samplesize` The longest short label in the list (used for autohang)

The other extra options are needed to control micro–typesetting of things that are buried deep within macros. There is a separate command to handle the larger layout options (`\setuppublicationlayout`, explained below), but the options here are the only way to make changes in the formatting used for editors’, authors’, and article authors’ names.

## Bibliographies

<code>editor</code>	command to typeset one editor in the publication list.
<code>author</code>	command to typeset one author in the publication list.
<code>artauthor</code>	command to typeset one article author in the publication list.
<code>namesep</code>	the separation between consecutive names (either editors, authors or artauthors).
<code>lastnamesep</code>	the separation before the last name in a list of names.
<code>firstnamesep</code>	the separation following the fistname or inits within a name in the publication list.
<code>juniorsep</code>	likewise for ‘junior’.
<code>vonsep</code>	likewise for ‘von’.
<code>surnamesep</code>	likewise for surname.

The commands after ‘editor’ e.g. are predefined macros that control how a single name is typeset. The four supplied macros provide formatting that looks like this:

```
\invertedauthor      von Hoekwater jr Taco
\invertedshortauthor von Hoekwater jr T
\normalauthor        Taco von Hoekwater jr
\normalshortauthor   T von Hoekwater jr
```

As you can see in the examples, there is a connection between certain styles of displaying a name and the punctuation used. Punctuation in this document has been set up by the ‘apa’ style, and that style makes sure that `\invertedshortauthor` looks good, since that is the default command for ‘apa’ style. (Keep in mind that the comma at the end of the author will be inserted by either ‘namesep’ or ‘lastnamesep’.)

In case you are not happy with the predefined macros; it is quite simple to define one of these macros yourself, it is a simple macro with 5 arguments: firstnames, von-part, surname, inits, junior.

For example, here is the definition of `\normalauthor`,

```
\def\normalauthor#1#2#3#4#5%
  {\bibdoif{#1}{#1\bibalternative{firstnamesep}}%
   \bibdoif{#2}{#2\bibalternative{vonsep}}%
   #3\bibalternative{surnamesep}%
   \bibdoif{#5}{#5\unskip}}
```

but commands can be a lot simpler, like this:

```
\def\surnameonly#1#2#3#4#5{#3}
\setuppublicationlist[editor=\surnameonly]
```

The following options are initialized depending on the global settings for ‘numbering’ and ‘autohang’:

<code>width</code>	Set to the calculated width of the largest label, but only if autohang is ‘yes’
<code>distance</code>	Set to 0pt, but only if autohang is ‘yes’
<code>numbercommand</code>	A command given in ‘setuppublications’ if numbering is turned on, otherwise empty.
<code>textcommand</code>	Set to a macro that outdents the body text if numbering is turned off, otherwise empty

## 2.3 Setting citation options: `\setupcite`

The `\cite` command has a lot of alternatives, as could be seen above in the setting of ‘refcommand’. And these alternatives have their own options:

```

\setupcite [...1,...] [...2,...]

1  author  year  authoryear  authoryears  key  serial  page  short  type  data  doi  url  num
2  andtext   = TEXT
   otherstext = TEXT
   pubsep    = TEXT
   lastpubsep = TEXT
   compress  = yes no
   inbetween  = TEXT
   left      = TEXT
   right     = TEXT

```

`andtext`      separation between two authors (for `\cite [author]` styles)  
`otherstext`   text used for ‘et.al.’ (for `\cite [author]` styles)  
`pubsep`       separator between publication references in a `\cite` command.  
`lastpubsep`   same, but for the last publication in the list.  
`left`         left side of a `\cite` (like `[`)  
`inbetween`    the separator between parts of a single citation.  
`right`        right side of a `\cite` (like `]`)  
`compress`     Whether `\cite` should try to compress it’s argument list.

Not all options apply to all types of `\cite` commands. E.g. ‘compress’ does not apply to the citation list for all options of `\cite`, since sometimes compression does not make sense or is not possible. The ‘num’ version compresses into a condensed sorted list, and the various ‘author’ styles try to compress all publications by one author, but e.g. years are never compressed.

Likewise, ‘inbetween’ only applies to three types: ‘authoryear’ (a space), ‘authoryears’ (a comma followed by a space), and ‘num’ (where it is ‘--’ (an endash), the character used to separate number ranges).

## 2.4 Setting up BibT<sub>E</sub>X: `\setupbibtex`

BibT<sub>E</sub>X bibliographic databases are converted into `.bbl` files, and the generated file is just a more T<sub>E</sub>X-minded representation of the full database(s).

The four `.bst` files do not do any actual formatting on the entries, and they do not subset the database either. Instead, the *entire* database is converted into T<sub>E</sub>X-parseable records. About the only thing the `.bst` files do is sorting the entries (and BibT<sub>E</sub>X itself resolves any ‘STRING’ specifications, of course).

The module will read the created `\jobname.bbl` file and select the parts that are needed for the current article.

```

\setupbibtex [...,*,...]

*  database = FILE
   sort     = title author short no

```

`database`    List of bibtex database file names to be used. The module will write a very short `.aux` file instructing BibT<sub>E</sub>X to create a (possibly very large) `\jobname .bbl` file, that will be `\input` by the module (at `\starttext`).  
`sort`        How the publications in the BibT<sub>E</sub>X database file should be sorted.  
             The default here is ‘no’ (`cont-no.bst`), meaning no sorting at all. ‘author’ (`cont-au.bst`) sorts alphabetically on author and within that on year, ‘title’ (`cont-ti.bst`) sorts al-

phabetically on title and then on author and year, and ‘short’ (`cont-ab.bst`) sorts on the short key that is generated by BibTeX.

For now, you need to run BibTeX by hand to create the `\jobname.bbl` file (`texutil` will hopefully do this for you in the future).

You may want to create the `\jobname.bbl` yourself. The `.bbl` syntax is explained below. There is no default database of course, and you do not *have* to use one: it is perfectly OK to just `\input` a file with the bibliographic records, as long as it has the right input syntax. Or even to include the definitions themselves in the preamble of your document.

The most efficient calling order when using BibTeX is:

```
texexec --once myfile
bibtex myfile
texexec myfile
```

Texexec should be smart enough to recognize how many runs are needed in the final part, but it seems it sometimes does one iteration too few. So you might have to call `texexec` one last time to get the page references correct. Numbered references always need at least one run more than author, year references, because the final number in the reference list is usually not decided upon yet at the moment the `\cite` command is encountered.

## 2.5 Borrowing publications: `\usepublications`

It is also possible to instruct the module to use the bibliographic references belonging to another document. This is done by using the command `\usepublications[files]`, where `files` is a list of other ConTeXt documents (without extension).

```
\usepublications [...]  
  
* FILE
```

To be precise, this command will use the `.bbl` and `.tuo` files from the other document(s), and will therefore not work if these files cannot be found (the `.tuo` file is needed to get correct page references for `\cite[page]`).

## 2.6 Legacy database support

Old BibTeX databases tend to contain L<sup>A</sup>T<sub>E</sub>X-specific commands and, especially, command–definitions. To make it easier to handle these databases, a support module that defines a simplified version of L<sup>A</sup>T<sub>E</sub>X’s `\newcommand` is shipped alongside the `bib` module. You can load this support code by adding

```
\usemodule[bibltx]
```

to your document preamble.

## 3 Citations

Citations are normally handled through the `\cite` command.

`\cite` has two basic appearances:



## 3.1 Default and explicit citations

```
\cite [...] [...]
      OPTIONAL
1  author year authoryear authoryears key serial page short type doi data url num
2  REFERENCE
```

The single-argument form executes the style-defined default citation command. This is the preferred way of usage, since some styles might use numeric citations while others might use a variation of the (author,year) style.

The two-argument form allows you to manually select the style you want.

### 3.1.1 Citation types

Following is the full list of recognized keywords for `\cite`, with a short explanation where the data comes from. Most of the information that is usable within `\cite` comes from the argument to `\startpublication`. This command is covered in detail below.

All of these options are *valid* in all publication styles, since ConTeXt always has the needed information available. But not all of these are *sensible* in a particular style: using numbered references if the list of publications itself is not numbered is not a good idea, for instance. Also, some of the keys are somewhat strange and only provided for future extensions.

First, here are the simple ones:

author	(Hoekwater)	(from ‘a’)
doi	[ <a href="http://dx.doi.org/">http://dx.doi.org/</a> ]	(from ‘d’)
key	[me]	(from ‘k’)
serial	[1]	(from ‘n’)
short	[TH2006]	(from ‘s’)
type	[manual]	(from ‘t’)
year	(2006)	(from ‘y’)
url	[ <a href="http://contextgarden.net/Bibliography">http://contextgarden.net/Bibliography</a> ]	(from ‘u’)

Keep in mind that ‘n’ is a database sequence number, and not necessarily the same number that is used in the list of publications. For instance, if ‘sorttype’ is cite, the list will be re-ordered, but the ‘n’ value will remain the same. To get to the number that is finally used, use

num [1] (this is a reference to the sequence number used in the publication list)

If the list of publications is not numbered visually, there will still be a number available.

Three of the options are combinations:

authoryear	Hoekwater (2006)	(from ‘a’ and ‘y’)
authoryears	(Hoekwater, 2006)	(from ‘a’ and ‘y’)
authornum	Hoekwater [1]	(from ‘a’ and ‘num’)
data	The data content of the entry	

Taco Hoekwater *ConTeXt Publication Module, The user documententation*. 14p. In case you didn’t know: it’s the document you are reading now.

And the last one is a page reference to the page where the the entry is typeset within the publication list.

page [12] (a page reference)

## 3.2 Citations with local setups

```
\cite [...,\u1,...] [...\u2,...]
      OPTIONAL
1  alternative = author year authoryear authoryears key serial page short type doi data
      url num
      extras = TEXT
      inherits from \setupcite
2  REFERENCE
```

The arguments in this form are inherited from `\setupcite`, except for `extras`. The argument of ‘`extras`’ is typeset at the end of the reference, but before a potential ‘`right`’, so it can be used for e.g. page or chapter specifiers.

## 3.3 Invisible citations

```
\nocite [...,*...]
* REFERENCE
```

This command registers the references in the argument list, but does not generate typeset material. It can be used to force certain entries from the database to appear in the typeset list of publications.

## 4 Placing the publication list

To typeset the list of publications, use `\completepublications` or `\placepublications` at the location in your text where you want the publication list to appear. As is normal in ConT<sub>E</sub>Xt, `\placepublications` gives you a raw list, and `\completepublications` a list with a heading.

The default for the publication list is to contain only the ‘locally’ referenced items, so if you want to use your own heading instead of the default one, you most likely want to call `\placepublications` with an explicit criterium, like so:

```
\placepublications[criterium=all]
```

The module uses the following defaults for the generated head:

```
\setupheadtext[en] [pubs=References]
\setupheadtext[nl] [pubs=Literatuur]
\setupheadtext[de] [pubs=Literatur]
\setupheadtext[it] [pubs=Bibliografia]
\setupheadtext[sl] [pubs=Literatura]
\setupheadtext[fr] [pubs=Bibliographie]
```

These (or new ones) can be redefined as needed.

## 5 The bbl file

A typical bbl file consists of one initial command (`\setuppublicationlist`) that sets some information about the number of entries in the bbl file and the widths of the labels for the list, and that command is followed by a number of appearances of `\startpublication ... \stoppublication`

The full appearance version of `\cite` accepts a number of option keywords, and we saw earlier that the argument of the `\startpublication` command defines most of the things we can reference to. This section explains the precise syntax for `\startpublication`.

Each single block defines one bibliographic entry. I apologise for the use of single-letter keys, but these have the advantage of being a) short and b) safe w.r.t. the multi-lingual interface.

```
\startpublication [...,*,...] ... \stoppublication

*   k = TEXT
    a = TEXT
    y = TEXT
    n = TEXT
    s = TEXT
    t = TEXT
    u = TEXT
    o = TEXT
```

Here is the full example that has been used throughout this document:

```
\startpublication[k=me,
                  t>manual,
                  a=Hoekwater,
                  y=2006,
                  s=TH2006,
                  n=1,
                  u=http://contextgarden.net/Bibliography]
\author[]{{Taco}[T.]}{Hoekwater}
\title{\CONTEXT\ Publication Module, The user documententation}
\pubyear{2006}
\note{In case you didn't know: it's the document you are reading now}
\pages{14}
\stoppublication
```

### 5.1 Defining a publication

The list of commands that is allowed to appear between `\startpublication` and `\stoppublication` is given below.

Order within an entry is irrelevant, except for the relative ordering within each of the three commands that might appear more than once: `\artauthor`, `\author` and `\editor`.

Most of these are ‘normal’ Bib<sub>T</sub><sub>E</sub><sub>X</sub> field names (in lowercase), but some are extra special, either because they come from non-standard databases that I know of, or because the bst file has pre-processed the contents of the field.

### 5.1.1 Complex fields

The three fields that contain names are extra special, because they have more than one argument. These are: `\artauthor`, `\author` and `\editor`. At the moment, these commands require exactly 5 arguments (of which two look like optional arguments, but they are not)

<code>\artauthor[#1]{#2}[#3]{#4}{#5}</code>	AUTHOR	For an author of any publication that appears within a larger publication, like an article that appears within a journal or as part of a proceedings.
<code>\author[#1]{#2}[#3]{#4}{#5}</code>	AUTHOR	The author of a standalone publication, like a monograph.
<code>\editor[#1]{#2}[#3]{#4}{#5}</code>	EDITOR	The editor of e.g. an edited volume.

### 5.1.2 Simple fields

Rather a large list, this is caused by the desire to support as many existing BibTeX databases as possible. Please note that a few of the fields have names that are not the same as in BibTeX, because a 1 on 1 mapping causes conflicts with predefined macro names in ConTeXt.

<code>\abstract</code>	ABSTRACT	just text.
<code>\annotate</code>	ANNOTATE	just text.
<code>\arttitle</code>	TITLE	The title of a partial publication (one that has <code>\artauthors</code> ).
<code>\assignee</code>	ASSIGNEE	Assigned person for a patent
<code>\bibnumber</code>	NUMBER	
<code>\bibtype</code>	TYPE	See the BibTeX documentation for it's use. This is <i>not</i> related to the type of entry that is used for deciding on the layout.
<code>\biburl</code>	URL	Location on the internet.
<code>\chapter</code>	CHAPTER	the chapter number, if this entry is referring to a smaller section of a publication. It might actually be a part number or a (sub)section number. The field <code>\bibtype</code> (above) differentiates between these.
<code>\city</code>	CITY	city of publication.
<code>\comment</code>	COMMENT	just text.
<code>\country</code>	COUNTRY	country of publication.
<code>\crossref</code>	CROSSREF	A cross-reference to another bibliographic entry. It will insert a citation to that entry, forcing it to be typeset as well.
<code>\day</code>	DAY	Date of publication (for a patent)
<code>\dayfiled</code>	DAYFILED	Filing date for a patent
<code>\doi</code>	DOI	Document Object Identifier
<code>\edition</code>	EDITION	The edition.
<code>\eprint</code>	EPRINT	E-print information
<code>\howpublished</code>	HOWPUBLISHED	
<code>\isbn</code>	ISNB	isbn number (for books)
<code>\issn</code>	ISSN	issn number (for journals)
<code>\issue</code>	ISSUE	issue number (for journals)
<code>\journal</code>	JOURNAL	The journal's name.
<code>\keyword</code>	KEYWORD	just text (for use in indices).
<code>\keywords</code>	KEYWORDS	just text (for use in indices).
<code>\lang</code>	LANGUAGE	The language of the current bibliographic record
<code>\month</code>	MONTH	Month of publication
<code>\monthfiled</code>	MONTHFILED	Filing month for a patent
<code>\names</code>	NAMES	just text (for use in indices).
<code>\nationality</code>	NATIONALITY	Nationality information for a patent

<code>\note</code>	NOTE	just text (this is the ‘standard’ BibTeX commentary field).
<code>\notes</code>	NOTES	just text.
<code>\organization</code>	ORGANIZATION	Like institute, but for e.g. companies.
<code>\pages</code>	PAGES	Either the number of pages, or the page range for a partial publication. The ‘t’ key to startpublication will decide automatically what is meant.
<code>\pubname</code>	INSTITUTION, PUBLISHER, SCHOOL	Publisher or institution name.
<code>\pubyear</code>	YEAR	Year of publication. Within this command, the BibTeX bst files will sometimes insert the command <code>\maybeyear</code> , which is needed to make sure that the bbl file stay flexible enough to allow all styles of formatting.
<code>\revision</code>	REVISION	Release version
<code>\series</code>	SERIES	Possible book series information.
<code>\size</code>	SIZE	Size in KB of a PDF file (this came from the NTG Maps database)
<code>\thekey</code>	KEY	See the BibTeX documentation for it’s use. This is <i>not</i> related to the key used for citing this entry.
<code>\title</code>	TITLE, BOOKTITLE	The title of a book.
<code>\volume</code>	VOLUME	Volume number for multi-part books or journals.
<code>\yearfiled</code>	YEARFILED	Filing year for a patent

When the `\lang` field’s content is a full word instead of a two-letter code, correct processing depends on an auxiliary command `\setbiblanguage`, to be used like this:

```
\setbiblanguage{English}{en}
```

The first argument is a literal `\lang` argument, the second argument has to be a two-letter language abbreviation understood by ConTeXt.

Adding in one of your own fields is reasonably simple:

```
\newbibfield[mycommand]
```

This will define `\mycommand` for use within a publication (plus `\bib@mycommand`, it’s internal form) as well as the command `\insertmycommand` that can be used within `\setuppublicationlayout` to fetch the supplied value (see below).

## 6 Defining a publication type layout

Publication style files of course take care of setting defaults for the commands as explained earlier, but the largest part of a such a publication style is concerned with specifying layouts for various types of publications.

The command that does the work is `\setuppublicationlayout`:

```
\setuppublicationlayout [...1,...] {...2.}

1  article book booklet conference electronic inbook incollection inproceedings manual
   mastersthesis misc patent periodical phdthesis proceedings standard techreport
   unpublished
2  TEXT
```

## Bibliographies

The first argument that is a publication (BibTeX entry) type, and all publications that have this type given as argument to the ‘t’ key of `\startpublication` will be typeset by executing the commands that appear in the group following the command.

For example, here is a possible way to typeset an article: from `bibl-apa`:

```
\setuppublicationlayout[article]{%
  \insertartauthors{ }{\insertthekey{ }{ }}%
  \insertpubyear{ }{ }\unskip.%
  \insertarttitle{\bgroup }\egroup. }{%
  \insertjournal{\bgroup \it}\egroup}
  {\insertcrossref{In }{ }}%
  \insertvolume
  {, }
  {\insertissue{ }{ }}\insertpages{: }{ }{ }%
  {\insertpages{, pages }{ }{ }%
  \insertnote{ }{ }%
  \insertcomment{ }{ }%
}
```

For every command in the long list given in the previous paragraph, there is a corresponding `\insertxxx` command. (As usual, `\author` etc. are special: they have a macro called `\insertxxxs` instead). All of these `\insertxxx` macros use the same logic:

```
\insertartauthors{<before>}{<after>}{<not found>}
```

Sounds easy? It is! But it is also often tedious: database entries can be tricky things: some without issue numbers, others without page numbers, some even without authors. So, you often need to nest rather a lot of commands in the `<not found>` section of the ‘upper’ command, and `\unskip` and `\ignorespaces` are good friends as well.

Incidentally, the distributed `bibl-xxx` files define layouts for the ‘standard’ publication types that are defined in the example bibliography that comes with BibTeX. The list of possible types is in no way limited to that list, but it provides a reasonable starting point.

## 7 References

- [1] Taco Hoekwater *ConTeXt Publication Module, The user documententation*. 14p. In case you didn’t know: it’s the document you are reading now.