User's Guide to the IdxTEX Program

Version 2.0

Abstract

The IdxTEX program is used to automate the generation of an Index in a LATEX document. It uses the .IDX file generated by the `\makeindex` command to create a file which is `\input` in the document to generate the Index. Version 2.0 improves the visual appearance of the Index and adds support for page ranges, index cross references, and the generation of a Master Index.

R L Aurbach
CR&DS MIS Group
Monsanto Company
800 N. Lindbergh Blvd.
St. Louis, MO 63167
(314)–694–5453

April 19, 1987

[File: IdxTeX.QMS]

User's Guide to the IdxTEX Program

The Index Problem

The LaTeX text formatting program is an extremely versatile tool for the generation of high-quality documents. However, its handling of an Index is incomplete.

LaTeX Index Processing

To make an Index using LaTeX, the writer includes the `\makeindex` command in the preamble of the document. This causes an auxiliary file (with file type .IDX) to be generated. For every `\index` command in the document, a `\indexentry` command is written to the auxiliary file, containing the text supplied in the `\index` command and the page number of the page on which the `\index` command occurred.

It is then the writer's responsibility to use the auxiliary file to create the series of commands needed to format an appropriate index and to insert these commands into the document at the appropriate place.

The entire process is described in Sections 4.5 and C.10.5 of the LaTeX *User's Guide and Reference Manual*.

The IdxTEX Solution

While this procedure is capable of generating an effective, high-quality Index, a substantial amount of work is required to convert the auxiliary .IDX file into the appropriate set of commands. The IdxTEX program was written to automate this manual part of the process.

IdxTEX uses the .IDX auxiliary file to generate a new file with file type .IND which contains all of the necessary commands to create an attractive, two-column index. This .IND file may then be included in the document at the appropriate place to generate the Index.

The IdxTEX program

- Generates all of the commands necessary to build the Index, including the `\begin{theindex}` and `\end{theindex}` commands.
- Generates an alphabetized, two-column index.
- Supports three levels of indexing — index items, index subitems, and index sub-subitems.
- Supports special formatting of ranges of consecutive page numbers, to enhance the visual appearance of the Index.
- Supports highlighting effects such as boldface, italics, etc., in the index items, while maintaining proper alphabetization of entries.
- Supports highlighting effects (boldface, italics, and underlining) of page numbers.
- Supports the use of LATEX commands, including `\verb` commands, in the index items, while maintaining proper alphabetization of entries.
- Supports references within one index entry to other index entries.
- Supports the generation of a Master Index for a set of documents.

Using IdxTEX

The IdxTEX program is easy to use. However, like BIBTEX, there are a number of steps necessary to generate the index and to get it included properly in your document.

Command Definition

The IdxTEX program is a foreign DCL command.[1]

Preparing the Document

Include the `\makeindex` command in the preamble area of the document source file (i.e., between the `\documentstyle` and `\begin{document}` commands). Also, include the appropriate `\index` commands (as described below) in the source file.

[1] On the Monsanto BB1T VAXcluster, the IdxTeX symbol is defined by issuing the CRLSETUP command.

Generate the IDX File

Use LaTeX to generate an initial form of the document. LaTeX will generate an auxiliary file which has the same filename as the main document file and a filetype of .IDX.

Run IdxTeX

Use IdxTeX to generate the formatted index file.

```
$ IdxTeX   file  [/TOC={ ARTICLE | REPORT }]
```

where "file" is the filename of the .IDX file. IdxTeX generates a file with the same filename as the .IDX file and a filetype of .IND.

The `/TOC` qualifier is optional. If you specify it, you must specify either the value `ARTICLE` or the value `REPORT`. The /TOC qualifier is used to add an entry into the Table of Contents for the Index. The standard document styles (*article* and *report*) do not automatically make a Table of Contents entry for the Index. If you want the Index to be listed in the Table of Contents, you **must** use the /TOC qualifier. The value `ARTICLE` should be used with documents which use the *article* document style or one of its derivatives. The value `REPORT` should be used with documents which use the *report* or *book* document styles.

Note

A number of document styles have been developed which automatically include a Table of Contents entry for the Index[2]. You **should not** use the /TOC qualifier with one of these document styles. If you do, the Index will be listed in the Table of Contents twice.

Generate the Document

Insert the command `\input{file.IND}` into your document source file just before the `\end{document}` command.

Finally, run the document through LaTeX twice (to make sure that the Table of Contents includes the page reference for the Index). Your document is now complete, including a formatted Index.

Special Features

The IdxTeX program attempts to perform all of the operations needed to generate an Index which you can be proud of. Of course, the quality of the Index is a function of the amount of effort you put into your writing — the program can only generate index entries for items you tell it to index for you. But if you do a good job of generating index entries, IdxTeX should do a good job formatting them for you.

[2]Examples of these styles include the Monsanto *pamphlet*, *manual*, and *memo* styles.

Index Generation

The .IND file generated by IdxTEX contains all of the commands necessary to generate the index. The file contains a `\begin{theindex}` command. This sets up the Index environment according to your current `\documentstyle`. For example, in the *article* `\documentstyle`, the index is an unnumbered section. In the *report* `\documentstyle`, the index is an unnumbered chapter.

The last line in the file is a `\end{theindex}` command. This command ends the Index environment. Between these lines are found all of the commands needed to generate the Index.

Alphabetical Index

The index is generated in an alphabetized, two-column format by major index entry. Special characters such as "\" don't participate in the alphabetization, so a term like `\begin` is indexed under "B", not under "\".

For clarity, the group of index entries which begin with one letter of the alphabet are set off from the index entries beginning with the next letter of the alphabet with vertical spacing and a heading. See the index of this document for examples.

Index Levels

Items are inserted into the Index by using the `\index` command. For example, to index the term "entry", enter `\index{Entry}` at the appropriate place in the text of your document. The information entered as the argument to `\index` will appear as specified (special visual effects are discussed below). The page number on which the `\index` command appeared will be displayed.

This approach generates a simple Index. That is, all terms are on the same footing. It is often convenient, however, to have terms appear as subindices under an index term. For example, if you were discussing "Commands", you might wish to have separate index entries for "Add Command" and "Change Command". However, in addition, you should create an index entry for "Commands" and place "Add" and "Change" as subindices under it. This technique generates an index which is more helpful for the person looking for information.

IdxTEX actually supports three such levels of indexing — a main index item, a subindex item, and a subsubindex item. In analogy to the indexing capabilities of the RUNOFF program, the "`>`" symbol is used to separate the index term from the subindex term and the subindex term from the subsubindex term in the `\index` command.

For example,

```
\index{Top Level Item>SubItem>SubSubItem}
```

generates such three-level deep index entry in the index. Look at the index of this document to see it.

There are two important points to note here.

- First, each `\index` command generates a single page number reference in the index. The page number is associated with the lowest level of the index specified. Therefore, in this example, the page number is associated with the "SubSubItem" entry, not any of the higher entries. If you want page numbers at all levels, you must use several `\index` commands.

  All subitems are displayed in alphabetical order below their top-level item. All subsubitems are displayed in alphabetical order below their second-level item.
- Second, the `>` symbol used to separate items of different levels is normally a math symbol. As such, it is possible that you will want to index a math expression which contains it. This is okay, because IdxTEX is sensitive to the use of `$` as a command to enter or leave math mode and will not consider `>` a level separator unless it is *not* in math mode.

While IdxTEX allows you to be pretty wordy in your index terms (there is a practical limit of about 110 characters for the text contained in an `\index` command), your index will be most effective if you select short and concise index terms. People usually scan an index quickly, looking for something that looks like what they are interested in. A lengthy exposition in an index term makes this difficult.

Item Highlighting

Sometimes it is desirable to make a word or phrase (or even an entire index entry) stand out in the Index. This is referred to as Item Highlighting, and is an effective technique to make the Index more useful, if not overdone.

Use the normal LATEX commands to generate the desired highlighting. For example, the following commands appear in this document.

```
\index{Highlighting>A {\bf boldface} entry}
      \index{Highlighting>An {\em italic\/} entry}
```

Look for the result in the index!

From the standpoint of IdxTEX, the item, subitem, and subsubitem are processed as different text elements. Therefore, if you wish to highlight more than one level of the index, you must handle each one independently. That is,

```
\index{{\bf One>Two>Three}}
```
WRONG!

is incorrect, but

```
\index{{\bf One}>{\bf Two}>{\bf Three}}
```
CORRECT!

will produce the desired result.

Page Number Highlighting

Sometimes, when you are indexing the same term in many places in a document, you may wish to help the reader decide which is the primary reference and which are secondary references. One way of doing this is to provide visual highlighting of important page numbers.

IdxTEX provides a mechanism to cause a page number of a reference to be printed in **boldface**, in *italics*, or underlined. There are three special characters recognized by IdxTEX **when they appear as the first character of an `\index` command argument**.

- The `^` command is used to make the page number reference appear in **boldface**.
- The `~` command is used to make the page number reference appear in *italics*.
- The `_` command is used to make the page number reference appear underlined.

For example, in this document, the following `index` commands appear.

```
\index{^Page Numbers>boldface}
\index{~Page Numbers>italics}
\index{_Page Numbers>underlined}
```

Look at the index to see the results!

Note

These special characters must appear as the first character of an `\index` command argument. If they appear after a `>` (i.e., as the first character of a subindex entry or a subsubindex entry), they will not work. In fact, they will cause a LATEX error!

Special LATEX Commands

Occasionally (such as in this document), it is desirable to include LATEX commands in the index in an index entry to create special effects or because the command itself is being referenced. IdxTEX

supports this — just type the entries as you would expect them to appear.

The trick is that IdxTEX tries to alphabetize your entries correctly. This means that it must figure out how to alphabetize LATEX commands, whose displayed images are different than the character strings which generate them.

The following rules attempt to indicate how IdxTEX treats LATEX commands when it finds them.

- Accents are ignored when figuring out how to spell a term. This means, for example, that "`se\~{n}or`" is treated as if it were spelled "`senor`".
- Text contained in "`\verb`" or "`\verb*`" commands are spelled as if the command were not present. For example, the index entry "`\verb+\Pi+`" is treated as if it were spelled "`\Pi`".
- Commands which affect font size or type style are ignored when figuring out how something is spelled. For example, "`\sc term`" is treated as if it were spelled "`term`". Note, however, that this applies only to spelling. The term will be displayed as you want it.
- Case is ignored in spelling considerations. Terms like "large" and "Large" are treated as identical. Only one reference (with two page numbers) will appear — the first form seen will be used.
- IdxTEX ignores the grouping commands "{", "}", and "$" when placing a term in alphabetical order. On the other hand, the sequences which generate explicit characters (i.e., "`\{`", "`\}`", and "`\$`") are handled as if they were the characters themselves — the "\" is ignored for spelling. For example, the term "`{\bf The \{ Command}`" will be alphabetized as if it were spelled "`The { Command`".
- All "\" characters are ignored. The term "`\begin`" will be located in the Index as if it began with the "b", not the "\".

Page Ranges

IdxTEX supports two types of page ranges — implicit and explicit.

An implicit page range is one which is discovered during index processing. For example, if it turns out that a particular item is indexed on three consecutived pages, then this is an implicit range.

An explicit page range is one in which you specifically indicate that an item being referenced is discussed on a particular page and succeeding pages.

Implicit Page Ranges

Suppose that you index the term "Command" a number of times within your document and it turn out (when you use LaTeX to format it) that the references occur on pages 10, 15, 16, 17, 20, and 21. IdxTeX will recognize that there are two implicit page ranges here and will display the reference in the index as

Command, 10, 15–17, 20–21

This processing occurs automatically — you need not do anything special to get it to occur (except, of course, insert the `\index` commands where they are needed).

In some document styles, chapter-oriented page numbering is used. From the standpoint of IdxTeX, a chapter oriented page number is any page number which consists of a string (which is usually numeric) followed by one or more dashes, followed by a page number. For example, `2--6` and `Ref--12` are typical chapter oriented page numbers.

Since the IdxTeX program parses and recognizes page numbers in this format as well as simple numeric page numbers, it is able to correctly build ranges for chapter oriented documents. For example, if references to "Command" occurred on pages 2–6, 3–7, 3–8, 3–9, and 4–10, then IdxTeX would build an index entry of the form

Command, 2–6, 3–7 to 3–9, 4–10

So far, we have considered page numbers which have no visual highlights. If one or more of the page number references in a range has a highlight, the range is broken up into sub-ranges, so that each sub-range has the same visual highlight. For example, if references to "Noise" occurred on pages 20 through 26, with the references on pages 22 and 23 flagged as underlined and the references on pages 24 and 25 flagged as boldface, then IdxTeX would generate an index entry of the form

Noise, 20–21, <u>22–23</u>, **24–25**, 26

Explicit Page Ranges

Another form of page range occurs when a single reference indicates that a particular topic is covered on a particular page and following pages. In this kind of reference, only one index

item is needed — for the first page referenced. By convention, the letters ff follow that page reference to mean "and following".

To specify this type of reference, use the `#` symbol as the first character in the index entry (just as if it were a special type of page highlight). For example, the reference `\index{#Page Ranges>Explicit}` was placed at the beginning of this section. Look to the index to see the result.

There are two special features of this notation.

- Explicit page ranges take precedence over implicit page ranges. That is, if an explicit range entry occurs within an implicit range, then the whole implicit range is condensed into a single explicit reference. For example, if the reference to "Command" was made on pages 20 through 25 and the reference on page 23 was explicit, then the output would have the form

  Command, 20ff

  rather than.

  Command, 20–22, 23ff, 24–25

- Explicit page ranges take precedence over all page highlighting. If

  ```
  \index{~Command}\index{#Command}
  ```

  occurs, for example, then the italic reference is ignored.

Cross References

It is not uncommon in an index to want to reference a term in a variety of different ways, so that people will be easily able to find the reference they need. This is often a lot of work, however, because you don't normally want to repeat a big list of `\index` commands for every place where a term is referenced. Instead, a great deal of effort can be saved if you can index the term thoroughly once, and then *cross reference* it in all its synonyms.

For example, suppose you are describing the syntax of a command. You index the various forms of the command under the term "Syntax". However, you also want to place an alternate entry in the index under "Command Syntax" in case people look there first. Obviously, you'd rather not repeat all of your index commands twice to get a complete set of references. Instead, you'd like to **refer** the reader who looks up "Command Syntax" to try "Syntax" for the information.

IdxTEX supports this capability. The special symbol "`&`" is used too indicate the beginning of a cross reference, which is used **instead of the page reference** in the entry. In our example above, entering `\index{Command Syntax&Syntax}` produces the desired result — look at the index.

You can combine cross references with page references. For example,

```
\index{Combining References}
\index{Combining References&Cross References}
```

contains a page reference and a cross reference, as is shown in the index.

For convenience, the notation `\index{aaa&bbb>ccc>ddd}` generates a cross reference to `bbb, ccc, ddd`, so you can use the same syntax in your cross references as you do in your index terms themselves. Note that the text following the "`&`" must obey all of the rules associated with index entries in general.

You can include as many cross references as you like. For example, the sequence

```
\index{aaa&bbb} \index{aaa&ccc} \index{aaa&ddd}
```

is perfectly acceptable. In the index, it generates

aaa, *see*
- bbb
- ccc
- dddD

d

e

Also note that a cross-reference entry does not, in itself, contain any page number information. Therefore, the index entries containing cross referencing may appear anywhere. For example, you could build a basic index in your document, then add all the appropriate cross reference entries later, all in one place in your text. Note, however, that cross references in index entries are treated like simple text — IdxTEX does not check to make sure that a cross reference entry actually exists. So be careful when creating them, to avoid misleading your reader.

Master Index Processing

A Master Index is a document in a set of documents which contains a complete index of all terms indexed in **any** of the other volumes of the document set. IdxTEX provides support for the automatic generation of a Master Index.

Prepare the Individual Indices

The first step is to run all of the other documents in the document set through LaTeX, so that all of the `.IDX` files are generated. These files are used by IdxTeX to format the Master Index.

Build the .MDX File

IdxTeX must be made aware of which `.IDX` files to use to build the Master Index. In addition, it is important to define a *label* which is associated with each volume in the Master Index, so that the reader can figure out which volume to look to find the reference.

Both of these requirements are handled by a new auxiliary file called the `MDX` File. The format of this file is quite simple. For each volume which you wish to be included in the Master Index, insert a line in the `.MDX` file of the form

```
\usefile{label}{idx-file}
```

Here, "*label*" is the label which you wish to be associated with the volume in the Master Index and "*idx-file*" is the file specification of the associated `.IDX` file which contains the relevant information.

For example, suppose you have a document set containing four documents — a User's Guide, a Reference Manual, an Operations Manual, and an Installation Guide. You wish to build an Introduction document which contains a Master Index of the document set. In this case, you might build an `.MDX` file named `INTRO.MDX` which contains the following lines

```
\usefile{User}{user_guide}
\usefile{Ref}{ref_manual}
\usefile{Operator}{op_manual}
\usefile{Install}{install}
```

This example assumes that the `.IDX` file for the User's Guide is `USER_GUIDE.IDX`, etc. When IdxTeX processes the Master Index, it uses a default file specification of `SYS$DISK:[].IDX` with each file specified in the `.MDX` file.

Run IdxTeX

Next, process the `.MDX` file with IdxTeX.

```
$ IDXTEX intro /MASTER
```

The /MASTER qualifier indicates that IdxTeX is to prepare a Master Index. In this case, the filename parameter of the IdxTeX command is the specification of the `.MDX` file. A default file specification of `SYS$DISK:[].MDX` is used.

IdxTeX will read the specified `.MDX` file and will read and process each of the `.IDX` files specified in it. It will generate a master index

output file which has the same name as the `.MDX` file, but with the `.MND` filetype. In this example, IdxTEX will generate the file `INTRO.MND`.

Build the Master Index

The final step in building the Master Index is to include a `\input` command to incorporate the `.MND` file into the final volume and to run LATEX on it, just as in the normal index case.

In our example, include the command

```
\input{intro.mnd}
```

in the file `INTRO.TEX` and process this file using LATEX as usual.

The result will be a Master Index, which contains all of the index information contained in the volume set, formatted so that users will easily be able to find it. In particular, if a term is referenced in more than one volume of the volume set, its references for each volume are clearly separated from each other to make it clearer which volumes are associated with which page references.

In short, IdxTEX now makes it as easy to generate an attractive Master Index to a set of documents as it does to generate an Index to each volume of the set.

Index

— T —

Contents