index-sort-as

Contents

index-sort-as 1

index-sort-as

The <index-sort-as> element specifies a sort phrase under which an index entry would be sorted.

This element gives an author the flexibility to sort an index entry in an index differently from how its text normally would be sorted. The common use for this is to disregard insignificant leading text, such as punctuation or words like ″the″ or ″a″. For example, the author may want <data> to be sorted under the letter D rather than the left angle bracket (<). An author may want to include such an entry under both the punctuation heading and the letter D, in which case there can be two index entry directives differentiated only by the sort order.

Certain languages may have special sort order needs. For example, Japanese index entries might be written partially or wholly in kanji, but need to be sorted in phonetic order according to its hiragana/katakana rendition. There is no reliable automated way to map written to phonetic text: for kanji text, there can be multiple phonetic possibilities depending on the context. The only way to correctly sort Japanese index entries is to keep the phonetic counterparts with the written forms. The phonetic text would be presented as the sort order text for indexing purposes.

The <index-sort-as> element's content is logically augmented by the textual content of its parent <indexterm> element to produce the effective sort key (i.e., the textual content acts as a secondary sort field), so two indexterms with different content but the same <index-sort-as> value would never merge into a single index entry.

An <index-sort-as> element provides sort key information for the indexterm that is its parent; therefore, in a multiple level indexterm, the index-sort-as only affects the level in which it occurs.

It is an error if there is more than one index-sort-as child for a given indexterm. An implementation may (but need not) give an error message, and may (but need not) recover from this error condition by ignoring all but the last index-sort-as.

This is an example of an index entry for <data> that will be sorted as ″data″:

```
<indexterm>&lt;data&gt;<index-sort-as>data</index-sort-as></indexterm>
```

Contains:

(text data or keyword or option or parmname or apiname or cmdname or msgnum or varname or wintitle or term or data or data-about or foreign or unknown) *(any number)*

Note: The actual contains or contained-by information displayed here may differ slightly, depending on whether the element is used in a map, bookmap, stand-alone topic, or composite file (ditabase).

Contained by:

indexterm

Inheritance:

+ topic/index-base indexing-d/index-sort-as

Attributes

Name	Description	Data Type	Default Value	Required?
keyref	The keyref attribute is reserved for future use. The OASIS DITA Technical Committee expects to finish designing this attribute in the next release of the DITA standard.	NMTOKEN	#IMPLIED	No
class	*Not for use by authors. If an editor displays class attribute values, do not edit them.* The class attribute supports specialization. Its predefined values allow DITA tools to work correctly with ranges of related content.	CDATA	*Default value differs for each element*	No
%univ-atts; (%select-atts;, %id-atts;, %localization-atts;)	A set of related attributes, described at %univ-atts;	parameter entity	*PE not applicable*	Not applicable
%global-atts; (xtrf, xtrc)	A set of related attributes, described at %global-atts;	parameter entity	*PE not applicable*	Not applicable