

quarto-index cross-reference conflict probe

Marking one term two different ways

`makeindex` rejects two marks that share an index key and a printed page but carry different encapsulations, and Quarto turns that rejection into a failed render. The filter never emits such a pair: every mark of a key more than one mark describes differently emits one agreed command, which the index tool merges into a single entry. It reports what it composed, once per entry. This fixture holds one instance of each shape that has to be composed, and two that do not.

A plain mark on kappa against a cross-reference two pages below it. The two plain marks are deliberately on different printed pages and the cross-reference mark on a third, so the printed locator count says which marks contributed one: two means the cross-reference mark contributed none, which is what this extension documents, and three would mean it did.

The second plain mark on the contested key, a page later: kappa.

The cross-reference mark on the contested key, a page later still: kappa.

A **see=** on lambda against a **see-also=** on lambda.

A plain mark on tau against a **see-also=** on tau — the pairing **kappa** does not carry, since which attribute is written changes which encapsulation the back-end emits.

Two **see=** marks naming DIFFERENT targets, which contest the key without either being a plain mark: **upsilon** and **upsilon**.

A both-attributes mark, whose single encapsulation is neither a bare **see** nor a bare **see also**, against a plain mark: **phi** and **phi**.

A contested key BELOW the top level, so the repair is exercised where the cross-reference hangs off a sub-entry rather than off a top-level term: **psi** and **omega**.

A contested key DEEPER than the back-end stores, carrying a sort key. The fold makes the third level a join of several, and an entry with no key of its own for that level files under the text it prints. Contesting the key must not move it: it has to file exactly where the same entry files uncontested, and only what it prints may change. **theta** and **eta**.

The uncontested twin of that entry — the same depth, and a sort key of the same shape, whose third level repeats the level's own text — which is what says where the contested one had to file: **iota**.

A contested key whose target carries every character this extension escapes. Folding a target out of the encapsulation channel and into the entry's printed text moves it into a field the index tool reads for its own operators too, so the escaping is probed there and not only where it already was: **chi** and **chi**.

Two marks that agree, which **makeindex** folds together and which must NOT be reported: **mu** and **mu**.

A term marked plainly twice, which is the ordinary case and must not be reported either: **nu** and **nu**.

Cross-reference targets that do and do not resolve

An HTML index links a cross-reference to its target when that target is an entry in the same index. Whether it is one is decided on the parsed level list, never on the string a reader sees — and the two can look identical, since levels are joined with **:** and a level may contain a colon itself.

A real two-level entry to point at: **pi**.

A single-level target that prints exactly like that two-level entry, and must therefore stay plain text: **rho**.

The genuine two-level target, which must be a link: **sigma**.

Entries the targets name

Each target above other than `sigma`'s and `rho`'s names one of these entries, marked with no visible text. `sigma`'s names the entry `pi` marks, and `rho`'s stays unresolved on purpose.

Index

% & # _ { } \ ~ ^ \$ @ | ! " < >, 5

Afar, 5

Another Way, 5

Aye Two, 5

Bee Two, 5

chi, *see* % & # _ { } \ ~ ^ \$ @ | ! " < >, 4

Deep

Level, *see* Shallow, 4

Elsewhere, 5

Elsewhere Again, 5

Here, 5

kappa, *see* Elsewhere, 1, 2

lambda, *see* Here; *see also* There

mu, *see* Same

Note

on birds, 4

nu, 4

One Way, 5

phi, *see* Aye Two; *see also* Bee Two, 4

rho, *see* Note: on birds

Same, 5

Shallow, 5

sigma, *see* Note: on birds

tau, *see also* Elsewhere Again, 4

There, 5

Tree

Branch

Cedar, Dogwood, *see* Afar, 4

Maple, Holly, 4

upsilon, *see* One Way; *see* Another Way