

# UserTagDesign

## User Tagging Design

### Data Set

```
A10: title = Lucene in Action (LIA)
A11: title = Lucene in Action Deux
A12: title = Solr Flare in Action
A13: title = Practical Perl

erik tagged
  A10 with "lucene"
  A11 with "lucene", "solr"
  A12 with "ruby", "solrflare"
  A13 with

yonik tagged
  A10 with
  A11 with "lucene", "solr", "excellent"
  A12 with "solr"
  A13 with "foo"
```

### Use Cases

#### U-addUserTag

Allow a user to tag a book

- Example: Allow erik to tag A10 with "lucene"

#### U-delUserTag

Allow a user to remove a tag from a book, or all instances of a tag they have used.

#### U-delUser

Remove all tags that were added by a specific user

#### U-tagFacet

Show number of books tagged with each tag, restricted by users current search results and filters.

- Example: user submits a search of "title:lucene", and the resulting tag counts are "lucene(2), solr(2), excellent(1)"  
Notice that the count is number of books with that tag, not the number of tags... there are 3 "lucene" tags on the books, but only 2 books are tagged "lucene".

#### U-userFacet

Show number of books tagged tagged by each user.

- Example: user submits a search of "title:lucene", and the resulting tag counts are "erik(2), yonik(1)"  
Notice that the count is number of books tagged, not the number of tags on books.

#### U-tagSuggest

When a user is tagging a book, allow them to type in the first few letters and then give a dropdown list of existing tags to choose from. Sort by tag popularity, optionally show counts.

Tag popularity: number of users using that tag, or number of books with that tag? Either could work if necessary,

- Example1: user types in "so" into the textbox when tagging a book, and

they are automatically shown "solr(2), solrflare(1)" (uses #books tagged)

- Example2: user types in "so" into the textbox when tagging a book, and

they are automatically shown "solr(3), solrflare(1)" (uses #tag instances)

## U-tagNarrow

User selects an existing tag to narrow their search results by.  
Any displayed results (including facet counts) must have all tags that have been selected by the user.

- Example: narrow search results by the tag "solr"

## U-userTagNarrow

Show all books a specific user tagged with a specific tag, or restrict search results by the same.

- Example: restrict matches to books with erik's "solr" tag => restricts to A11

## U-tagNarrowSuggest

Allow the user to narrow their search results by typing in a tag instead of selecting it from a list. When the user has typed one or two letters, automatically pop up a list of tags starting with that prefix. Optionally sort tags by number of books it applies to in the current search results.

- Example: search "title:lucene", user types "so" and is presented with solr(1)

## U-userNarrow

Restrict books to those tagged by a specific user.

- Example: search "title:\*" restrict to books tagged by erik => A10,A11,A12
- Example2: search "title:\*", facet by tag, restrict to books tagged by erik:  
facet counts={lucene(2),solr(2),ruby(1),solrflare(1),excellent(1)}  
(note that this does \*not\* restrict shown tag counts to erik's tags)

## U-userTagsNarrow

Restrict \*tags\* to those of a specific user.

- Example: search "lucene", facet by tag, restrict to erik's tags:  
facet counts={lucene(2),solr(1)}

## U-userNarrowMulti

Restrict books to those tagged by a specific users.

- Example: search "title:\*" restrict to books tagged by erik or yonik => A10,A11,A12,A13
- Example2: search "title:\*" restrict to books tagged by erik and yonik => A10,A11,A12

## U-tagRelevance

When searching for a specific tag, increase the relevance of books that have more instances of that tag.

- Example: search for tag "lucene" and show A11 before A10

## U-tagTimeliness

Restrict to tags added in the last year (or time period)

## Machine Tags or Triple Tags

<http://www.flickr.com/groups/api/discuss/72157594497877875>

## Tag Hierarchies

# Implementations

## Flat Schema #1

Add tags directly to the documents as a single user/tag token.

## U-addUserTag

```
add to A10, field utag=~erik#lucene" // single token
add to A10, field utag2=~erik", "#lucene" // two tokens, added via copyField with a tokenizer that splits the
original
```

Alternative:

```
add to A10, field utag="erik#lucene" // or "erik lucene", single token
add to A10, field user="erik" // via copyField
add to A10, field tag="lucene" // via copyField
```

The latter looks simpler, but the former allows phrase queries to match different components of a tag with a single query. A Lucene [PhraseQuery](#) across multiple fields would also work for the latter if this capability is needed.

## Relevancy Calculations for Tags

To leverage Relevancy calculations, you'd include the tag as part of the regular fulltext search (q), vs. just adding it as a filter (fq).

If multiple users have tagged a document with "lucene", then that field's density for the term will be higher, so Relevancy will also tend to be higher. However, another document with only 1 tag, which happens to be 'lucene', will likely still rank higher than a heavily tagged document with only 40% of the tags equal to 'lucene', given Lucene's default relevancy formulas.

More advanced relevancy models would need more sophisticated implementations, for example perhaps a custom Similarity class.

### U-delUserTag

```
remove A10.utag=~erik#lucene"
```

### U-delUser

```
q="utag:~erik*", get set of documents, remove all tags starting with ~erik
```

### U-tagFacet

```
q="title:lucene" facet.field=utag2 facet.prefix=#
```

### U-userFacet

```
q="title:lucene" facet.field=utag2 facet.prefix=~
```

### U-tagSuggest

- Example1: facet.field=utag2 facet.prefix=#so
- Example2: not easily doable... would require more work within solr to count up tf's

### U-tagNarrow

```
fq=utag2:#solr
```

### U-userTagNarrow

- fq=utag:~erik#solr
- OR fq="utag2:~erik #solr"

### U-tagNarrowSuggest

```
q=title:lucene facet.field=utag2 facet.prefix=#so
```

### U-userNarrow

```
q=title:* fq=utag2:~erik
```

### U-userTagsNarrow

```
q=lucene fq=utag2:~erik facet.field=utag facet.prefix=~erik
```

### U-userNarrowMulti

- Example: q=title:\* fq=utag2🙄~erik OR ~yonik)
- Example2: q=title:\* fq=utag2🙄+~erik +~yonik)

U-tagRelevance

q=utag2:#lucene

U-tagTimeliness

??? reserve another prefix for fields like time