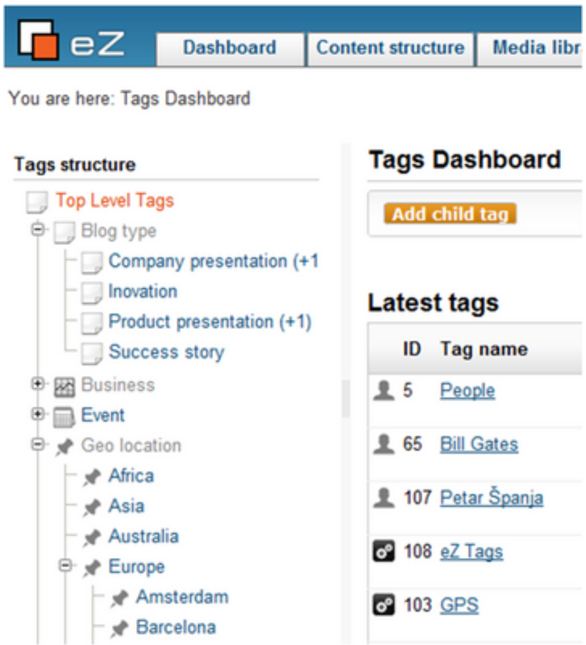
The eZ Tags extension

The eZ Tags extension
Overview
What to do with it
Installing eZ Tags
Using eZ Tags
Usage scenarios
Features

Overview



ID	Tag name
5	People
65	Bill Gates
107	Petar Španja
108	eZ Tags
103	GPS

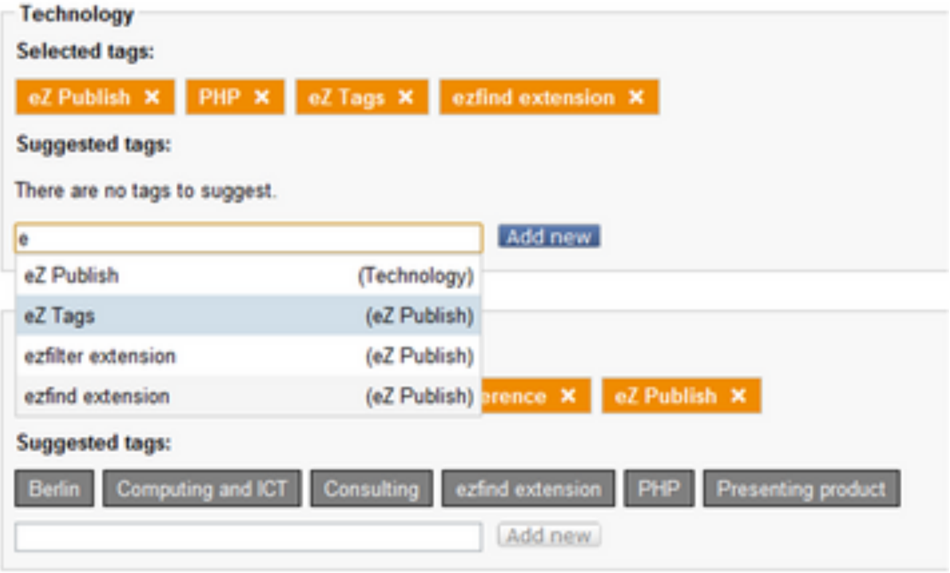
eZ Tags is an **eZ Publish extension for taxonomy management** and easier classification of content objects. We basically took the standard ezkeyword data type and enhanced it to the next level by:

- Organizing tags (keywords) in a tree
- Developing a “sexy” looking tag input interface
- Implementing a management interface for creating/deleting/renaming/merging tags
- Inventing tag suggestion powered by eZ Find
- Adding tag view for web interface
- Adding extended attribute filter for content list/tree fetch

The result is an extension that is not only able to replace the ezkeyword data type, but can be used for all taxonomies, including:

- Closed classifications which are usually predefined
- Open classifications like user tags (usually referred to as “folksonomy”)
- Combination of both

What to do with it



There are no tags to suggest.

Here are some examples on what you can do with eZ Tags:

1. Replace the ezkeyword data type. Migration should be straightforward as database schema is very similar. You will get the hierarchy and management which is missing in ezkeyword.
2. Replace the closed classification based on ezselection or ezobjectrelation(s) data type. You will get much easier input interface, easier management & maintenance, better performance
3. Be able to change from open to closed and vice versa when ever you need
4. Provide a better user experience to your editors
5. Create dynamic pages based on tagged content

Installing eZ Tags

eZ Tags requires eZ Publish version 4.3 and above to function properly. Versions 4.2 and below are not supported mainly because they don't have eZ JSCore extension in distribution files, and eZ Tags relies heavily on it. Other than that, there may also be some incompatibilities with APIs in mentioned eZ Publish versions, since eZ Tags is not tested on them. Also, eZ Tags uses eZ Find for suggesting tags and fetching related tags, so those features will not be available if eZ Find is not installed and activated.

For the most part, installing eZ Tags is a straightforward process. Download the stable release version from http://projects.ez.no/eztags and unpack it into extension directory of your eZ Publish installation. You need to add two tables in your database, so run the following command from root of your web, replacing the example values with the real ones:

```
mysql -u "user" -p"password" -h"host" "database" <
extension/eztags/sql/mysql/schema.sql
```

After that, activate the extension in your site.ini, regenerate the autoload array and clear the caches. This simple process is enough to have eZ Tags working. However, to gain access to all the functionality and speed, there are a few more steps you can do. These steps are optional, but highly recommended:

- To speed up loading of the tags tree in admin interface, you need to symlink index_treemenu_tags.php file from the extension, to root directory of your web (note the dot at the end of the command):

  ```
  ln -s extension/eztags/index_treemenu_tags.php .
  ```

- After that, add the following in your .htaccess file, just below "RewriteEngine On" line:

  ```
  RewriteRule tags/treemenu/? index_treemenu_tags.php RewriteRule
  ^index_treemenu_tags\.php - [L]
  ```

- To enable anonymous users to actually see the content that is related to a particular tag, allow anonymous access to "view" view of "tags" module
- Finally, if you wish to use tags suggestion and see related tags on tag view page, install and activate eZ Find extension and then edit the file extension/ezfind/java/solr/conf/schema.xml, add the following block of text and then restart Tomcat and reindex:
  - Inside <schema> element add:

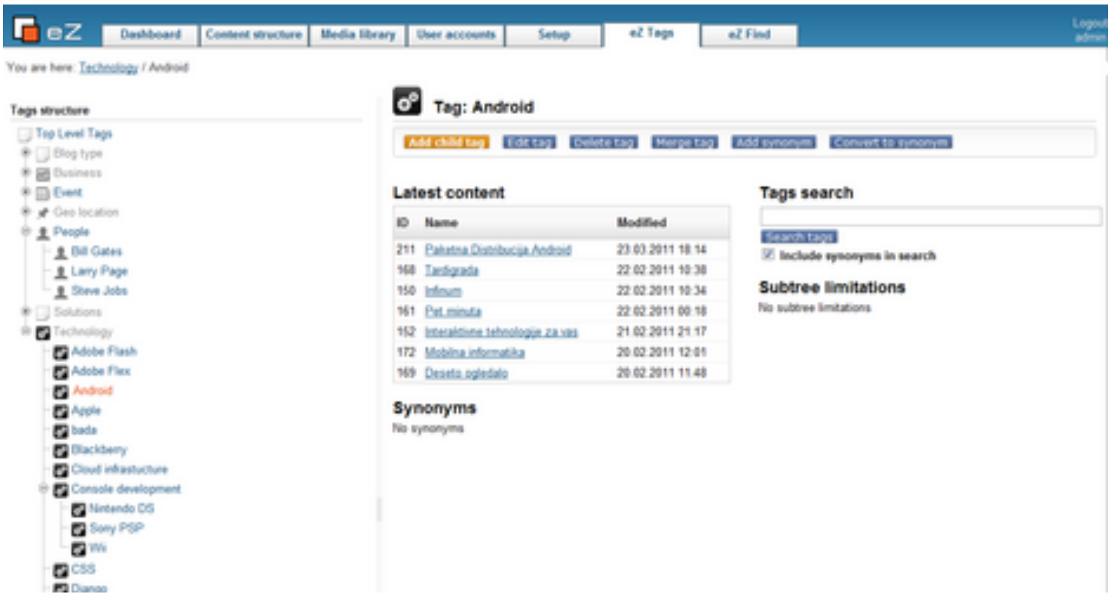
```
<copyField source="*_lk" dest="ezf_df_tags" /> <copyField
source="*_k" dest="ezf_df_tags" />
```

- Inside <fields> element add:

```
<field name="ezf_df_tags" type="text" indexed="true"
stored="true" multiValued="true" termVectors="true" />
```

Using eZ Tags

There are two sides to using eZ Tags extension. First one is in content object edit interface and the second one is a separate tab in administration where you can manage your tags (search tags, add new ones, delete them, merge two tags into one, add/edit/delete the tag synonyms and convert tags to synonyms). Image below is a screenshot of tags admin interface:



Tag: Android

Latest content

ID	Name	Modified
211	Paketna Distribucija Android	23.03.2011 18:14
168	Tardigrada	22.02.2011 10:38
150	Infinum	22.02.2011 10:34
161	Pet minuta	22.02.2011 00:18
152	Interaktivne tehnologije za vas	21.02.2011 21:17
172	Mobilna informatika	20.02.2011 12:01
169	Deseta ogledalo	20.02.2011 11:48

Synonyms

No synonyms

Tags search

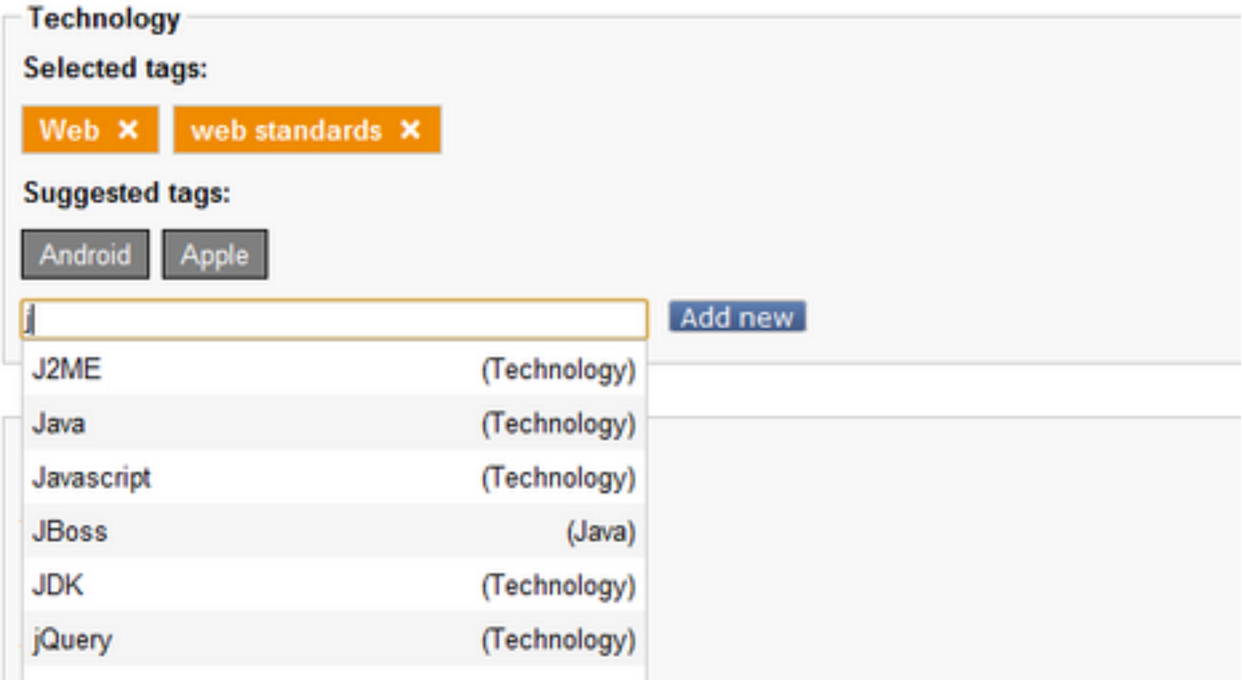
☑ Include synonyms in search

Subtree limitations

No subtree limitations

Before adding tags to your objects, you need to edit your classes and add the “Tags” attribute to them. When adding the data type to your class, you can choose if this particular attribute can use all available tags in the database, or just a particular subtree. This can be done by selecting one of the tags in tag tree popup window. If you do select one of the tags, only tags below the selected one can be added to the content object attribute. If you select the root of the tree, object attribute can use all the available tags.

Image below illustrates how content object attribute edit interface looks like:



Orange items (changed to blue in version 1.0 final) represent tags that are currently present in your object attribute and adding them is easy. Just type anything into the available input field, autocomplete will be activated to show you matching tags, and after that, just click on one of the results.

If you find yourself in need to add the tags not already present in the database, enter tag name into input field and click “Add new” button which will show you a popup window to place your tag into tree hierarchy. After selecting a parent tag in popup window, it will automatically close and the tag will be added to your object.

Grey items represent tags that were suggested to you based on your current selection of tags in the attribute and tags present in other objects as your current selection.

Usage scenarios

There are several scenarios on how to use eZ Tags.

Here are some general scenarios:

1. The most obvious is to replace standard ezkeyword data type used for open taxonomies. It basically means to let users add tags on their own. Important thing is that the web admin can always rearrange the tags for better usage:
    a. For SEO: more context words in tag links (e.g. /tags/view/Places/UK/London/BigBen)
    b. For cross linking: parent tag linking on tag pages
    c. For easier management
    d. Less duplicated entries by merging and synonyms
2. Setup a closed taxonomy with more classification where tags will be inserted in bulk. Users can only select predefined tags. Important thing is that it is very easy to manage tags if there are some changes after deployment. Also, as tags can be structured in tree hierarchy, search can be done by tag context (parent line)
3. The mix of both previous scenarios. It is possible to simply open up a closed taxonomy or close the open one. Depends on the situation

Features

Version 1.1 was released June 2010. The main focus of 1.1 was to add different fetches so developers can easily access all tags through their templates and PHP code, raising the value of the extension as tags now behave just as content objects and nodes. You can do with them whatever you want, organize them in whatever way you see fit, and act upon the model in your code.

Four fetches were added and they mimic the built in eZ Publish content fetches:

- Tags/list and tags/list_count fetches behave just like content/list and content/list_count. They return a list of child tags and their count just below a specific tag ID, respectively. You can use those fetches to build navigation structures, for example.
- Tags/tree and tags/tree_count, which can be used to fetch absolutely all tags below a specific tag ID, just like content/tree and content/tree_count fetches do for content nodes.

Version 1.2 is a mix of a maintenance changes and bug fixes, while still adding a couple of useful features:

- eZ Tags data type now fully supports class serialization
- A small but valuable feature is "tags" attribute in data type content (example: $node.data_map.eztags.content.tags) which allows you to get a list of all tags in your content object attribute
- Administration interface got a handy new feature that displays the table of all child tags at the bottom of the tags/id page.
- Tags/view module view in the user part of the site was initially conceived to display only the latest objects that were related to a specific tag. It is now extended to display all objects, through pagination support
- Included extended attribute filter got a small upgrade and now you can fetch nodes/objects that have one or more tags you specify, and is not limited to one tag as before
- eZ Tags data type got two new options to better match the requirements of your taxonomy/folksonomy rules:
    - You can now limit the number of tags that can be added to content object attribute. eZ Tags content object attribute edit interface supports that setting in a streamlined way. When you add the maximum possible number of tags, all controls for adding additional tags simply disappear and reappear again as you remove tags from your attributes

  - When you use some of the tags as a closed taxonomy, sometimes there's a requirement to not be able to add the root tag of of your tag sub tree to your objects, as it only serves as a container and doesn't have any real value to you. You can now set the option not to display that tag in object edit interface so your editors cannot add it to your object, thus respecting the rules you set for your data model. This option was available in the initial release of eZ Tags, however, hidden in the ini file and applied system wide, not per class attribute

- We have included a handy script that can be used to convert the content of ezkeyword data type to eZ Tags, which allows you to migrate easily from ezkeyword to eZ:

```
$ sudo -u apache php extension/eztags/bin/php/convertezkeyword.php --help

Converts ezkeyword datatype content to eztags datatype content.

Since the script would require as many publish operations as there are
translations per each object, the script will not republish the objects, but
rather update the current version of currently published objects. Because of
that, you will need to take care of clearing relevant caches, reindexing and
so on.

php extension/eztags/bin/php/convertezkeyword.php \
  --from-attr-id=123 --to-attr-id=456 --parent-tag-id=42

General options:

  -h,--help             display this help and exit

  -q,--quiet            do not give any output except when errors occur

  -s,--siteaccess       selected siteaccess for operations

  -d,--debug...         display debug output at end of execution

  -c,--colors           display output using ANSI colors (default)

  --no-colors           do not use ANSI coloring

  --logfiles            create log files

  --no-logfiles         do not create log files (default)

  -v,--verbose...       display more information

  -r,--allow-root-user  Allows the script to be run by the root user

Options:

  --from-attr-id=VALUE   Specifies source class attribute ID.

  --to-attr-id=VALUE     Specifies destination class attribute ID.

  --parent-tag-id=VALUE  Specifies where in tags tree new tags are located
```