

Developing Drupal site with native i18n support

Ilya Gruzinov
(ilia gruzynov)
(Илья Грузинов)

Table of contents

1. Model/Concept
2. Used Modules
3. Interface translation
4. Node translations
5. Views & queues
6. Paths & menus
7. Language negotiation

Model / Concept

- Few languages
- Default site language is not English.
- We use nodequeues for ordered content
- We use views
- We want language switcher on each page. when language is switched we must stay on this page but use another language if possible

Used modules

1. Locale
2. Content translation(translation)
3. i18n
 1. Multilingual content(i18n_node)
 2. Block languages(i18n_block)
 3. Field translation(i18n_fields)
 4. Menu translation(i18n_menu)
 5. Path translation(i18n_path)
4. Views
5. Nodequeue
6. Pathauto
7. Transliteration

t() is translate

t('Some text') will return 'Some text' in current language or if translation is absent original text.

```
t('Hello @username', array('@username' => $user->name), 'de');  
//Hallo Bob or Hello Bob
```

Where *t()* can/must be used

- Templates
- Modules which return some text for displaying

Where *t()* was used

- In all drupal core modules
- In most of other modules which can be downloaded from drupal.org

How to translate 'Some text'

- Via translation interface. It is easy.

Multilingual site

Node 1
(en)

Node 2
(de)

Node 3
(de)

Multilingual site with translations

Node 1
(en)

Node 2
(de)

Node 3
(de)

Node 5
(de)

Node 6
(en)



Create Article


Title *

Article creation

Body ([Edit summary](#))


Just add new node with selected language

Text format

[More information about text formats](#) 

- Web page addresses and e-mail addresses turn into links automatically.
 - Allowed HTML tags: <a> <cite> <blockquote> <code> <dl> <dt> <dd>
- Lines and paragraphs break automatically.

Language

English 

Save

Preview



Home

Home

Search input field with magnifying glass icon

Navigation

► [Add content](#)

Languages

- English
- Deutsch
- Русский

Article creation

[View](#) [Edit](#) [Translate](#) [Nodequeue](#)

Submitted by [admin](#) on Sun, 09/11/2011 - 17:36

Just add new node with selected language

Language

English

Add new comment

Your name [admin](#)

Subject

Text input field for subject

Comment *

Large text area for comment with a scrollbar

Translations of *Article creation*

VIEW

EDIT

TRANSLATE

NODEQUEUE

[Home](#) » [Article creation](#)

Translations of a piece of content are managed with translation sets. Each translation set has one source post and any number of translations in any of the [enabled languages](#). All translations are tracked to be up to date or outdated based on whether the source post was modified significantly.

LANGUAGE	TITLE	STATUS	OPERATIONS
English (source)	Article creation	Published	edit
German	n/a	Not translated	add translation
Russian	n/a	Not translated	add translation

SELECT TRANSLATIONS FOR *ARTICLE CREATION*

Alternatively, you can select existing nodes as translations of this one or remove nodes from this translation set. Only nodes that have the right language and don't belong to other translation set will be available here.

German	<input type="text"/>	<input type="radio"/>
Russian	<input type="text"/>	<input type="radio"/>

Update translations

Article erstellen

Title *

Article creation

Body [\(Edit summary\)](#)


Just add new node with selected language

Textformat Filtered HTML 

[Weitere Informationen über Textformate](#) 

- Internet- und E-Mail-Adressen werden automatisch umgewandelt.
- Zulässige HTML-Tags: `<a>` `` `` `<cite>` `<blockquote>` `<code>` `` `` `` `<dl>` `<dt>` `<dd>`
- HTML - Zeilenumbrüche und Absätze werden automatisch erzeugt.

Sprache

Deutsch 

Speichern

Vorschau

Translations of *Article creation*

VIEW

EDIT

TRANSLATE

NODEQUEUE

[Home](#) » [Article creation](#)

Translations of a piece of content are managed with translation sets. Each translation set has one source post and any number of translations in any of the [enabled languages](#). All translations are tracked to be up to date or outdated based on whether the source post was modified significantly.

LANGUAGE	TITLE	STATUS	OPERATIONS
English (source)	Article creation	Published	edit
German	Artikel Schöpfung	Published	edit
Russian	n/a	Not translated	add translation

SELECT TRANSLATIONS FOR *ARTICLE CREATION*

Alternatively, you can select existing nodes as translations of this one or remove nodes from this translation set. Only nodes that have the right language and don't belong to other translation set will be available here.

German	<input type="text" value="Artikel Schöpfung [nid:8]"/>	<input type="radio"/>
Russian	<input type="text"/>	<input type="radio"/>

Update translations

Deutsch



Home

Home



Navigation

▶ [Add content](#)

Languages

- English
- [Deutsch](#)
- [Русский](#)

Article creation

View

Edit

Translate

Nodequeue

Submitted by [admin](#) on Sun, 09/11/2011 - 17:36

Just add new node with selected language

Language

English

[Deutsch](#)

Add new comment

Your name [admin](#)

Add simple queue

LIST

SETTINGS

SIMPLE QUEUE

Simple queues have just one subqueue. Nodes put into a queue are added to the back of the queue; when a node is added to a full queue, the node in the front of the queue will be popped out to make room.

Title *

Enter the name of the queue

Internationalization

Treat translation nodes as a single node

Manually manage translated nodes

Treating translations as a single node allows users to add, remove and manipulate a node in the queue regardless of which translation is acted upon; nodequeue will only act on the original node. When manually managing translation nodes, Nodequeue will ignore the relationship between node translations; each translation node must be added to the queue separately and will occupy a separate spot in the queue. Changing this setting will **not** update content that is already in the queue.

Submit

Add relationships

Search

Filter

Content translation

- Content translation: Source translation
The source that this content was translated from.
- Content translation: Translations
Versions of content in different languages.

Selected: Content translation: Translations

Add and configure relationships

Cancel

Configure Relationship: Content translation: Translations

For

Versions of content in different languages.

Identifier

Edit the administrative label displayed when referencing this relationship form filters, etc.

Require this relationship

Enable to hide items that do not contain this relationship

Translation option

The translation options allows you to select which translation or translations in a translation set join on. Select "Current language" or "Default language" to join on the translation in the current or default language respectively. Select a specific language to join on a translation in that language. If you select "All", each translation will create a new row, which may appear to cause duplicates.

▶ [MORE](#)

Apply (all displays)

Cancel

Remove

Configure field: Content: Title

For

The content title.

Relationship

Do not use a relationship

Translations

Enable to create a label for this field.

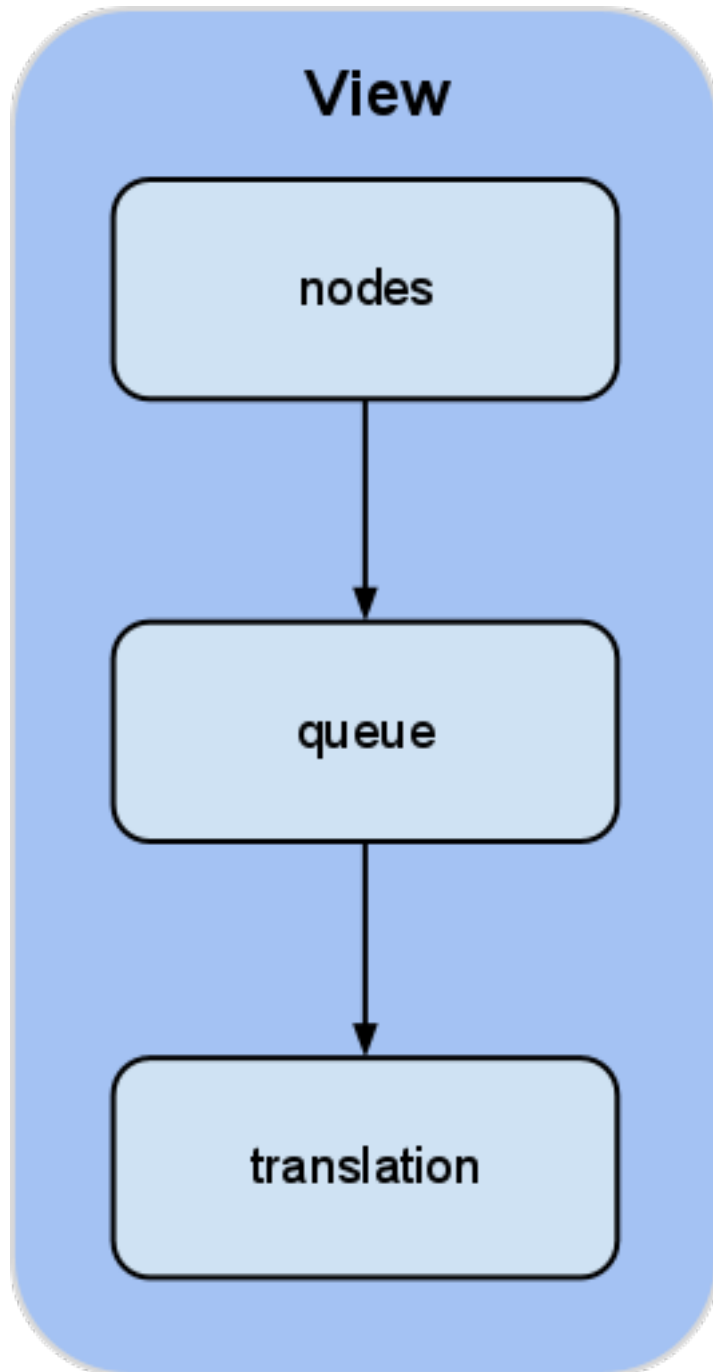
Exclude from display

Enable to load this field as hidden. Often used to group fields, or to use as token in another field.

Link this field to the original piece of content

Enable to override this field's links.

▶ [STYLE SETTINGS](#)



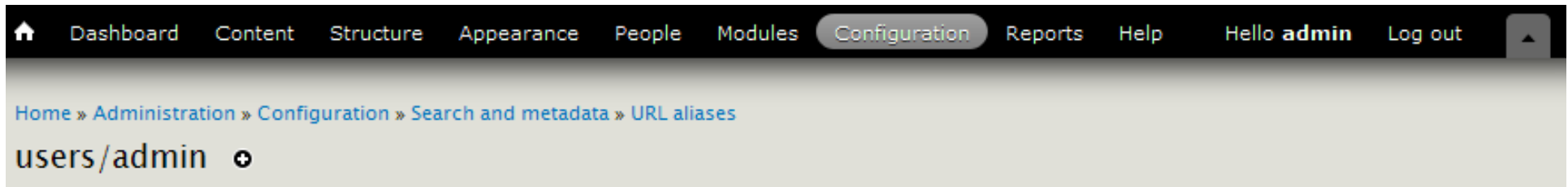
- Lets create node in english
- Put to queue
- Look to our view
- **Node is absent here.**
- Lets add translation to this node in any language
- Node appear in english version of view.
- Lets remove all translations except original english version.
- Node **still** in view.

tnid !

```
1. <?php
2.
3. /**
4.  * hook_enable() implementation
5.  */
6. function i18n_node_tnid_auto_enable() {
7.     $content_types = node_type_get_types();
8.     $queued = array();
9.     foreach ($content_types as $name => $value) {
10.         if (i18n_node_type_enabled($name)) {
11.             $queued[] = $name;
12.         }
13.     }
14.     if ($queued) {
15.         db_update('node')
16.             ->expression('tnid', 'nid')->condition('type', $queued, 'IN')->condition('tnid', '0',
17.             '=')->execute();
18.     }
19.
20. /**
21.  * hook_node_insert() implementation
22.  */
23. function i18n_node_tnid_auto_node_insert($node) {
24.     if (i18n_node_type_enabled($node->type)) {
25.         db_update('node')
26.             ->expression('tnid', 'nid')->condition('nid', $node->nid, '=')->execute();
27.     }
28. }
```

Path

- Translated nodes have own i18ned alias
- We can add alias for each language for each internal path



Language

German

A path alias set for a specific language will always be used when displaying this page in that language, and takes precedence over path aliases set for *All languages*.

Existing system path *

http://i18n.local/

Specify the existing path you wish to alias. For example: node/28, forum/1, taxonomy/term/1.

Path alias *

http://i18n.local/

Specify an alternative path by which this data can be accessed. For example, type "about" when writing an about page. Use a relative path and don't add a trailing slash or the URL alias won't work.

Save

Delete

```
1. function mm_urls_map() {
2.     static $map= array(
3.         'services/' => array(
4.             'fr' => 'nos-services/'
5.             'de' => 'leistungen/'
6.         ),
7.         'methods/' => array(
8.             'fr' => 'nos-methodes/',
9.             'de' => 'methoden/'
10.        ),
11.    );
12.    return $map;
13. }
14. function mm_url_inbound_alter(&$path, $original_path, $path_language) {
15.     global $language;
16.     $lang = $language->language;
17.     $map = mm_urls_map();
18.     foreach ($map as $from => $to) {
19.         if (!empty($to[$lang]) && substr($path, 0, strlen($to[$lang])) == $to[$lang]) {
20.             $path = $from . substr($path, strlen($to[$lang]));
21.         }
22.     }
23. }
24. function mm_url_outbound_alter(&$path, $original_path, $path_language) {
25.     global $language;
26.     $map = mm_urls_map();
27.     $lang = (!is_array($original_path) || !array_key_exists('language', $original_path)) ?
    !empty($original_path['language']) : $language->language;
28.     foreach ($map as $from => $to) {
29.         if (!empty($to[$lang]) && substr($path, 0, strlen($from)) == $from) {
30.             $path = $to[$lang] . substr($path, strlen($from));
31.         }
32.     }
33. }
```

Languages +

LIST

DETECTION AND SELECTION

Define how to decide which language is used to display page elements (primarily text provided by Drupal and modules, such as field labels and help text). This decision is made by evaluating a series of detection methods for languages; the first detection method that gets a result will determine which language is used for that type of text. Define the order of evaluation of language detection methods on this page.

User interface text language detection

Order of language detection methods for user interface text. If a translation of user interface text is available in the detected language, it will be displayed.

[Show row weights](#)

DETECTION METHOD	DESCRIPTION	ENABLED	OPERATIONS
+ URL	Determine the language from the URL (Path prefix or domain).	<input checked="" type="checkbox"/>	Configure
+ Session	Determine the language from a request/session parameter.	<input checked="" type="checkbox"/>	Configure
+ User	Follow the user's language preference.	<input checked="" type="checkbox"/>	
+ Browser	Determine the language from the browser's language settings.	<input checked="" type="checkbox"/>	
+ Default	Use the default site language (English).	<input checked="" type="checkbox"/>	

Save settings

```
1. function mm_language_negotiation_info_alter(array &$language_providers) {
2.   if (isset($language_providers[LANGUAGE_NEGOTIATION_DEFAULT])) {
3.     $language_providers[LANGUAGE_NEGOTIATION_DEFAULT]['callbacks']['language'] = '_mm_locale_negotiation_en';
4.     $language_providers[LANGUAGE_NEGOTIATION_DEFAULT]['description'] = t('Use the @language_name.',
array('@language_name' => 'English'));
5.     $language_providers[LANGUAGE_NEGOTIATION_DEFAULT]['file'] = drupal_get_path('module', 'mm') .
'/locale_negotiation.inc';
6.   }
7. }
8. function mm_init() {
9.   if ($_SERVER["REQUEST_METHOD"] == "GET") {
10.    module_load_include('inc', 'mm', 'locale_negotiation');
11.    if (_mm_locale_negotiation_used()) {
12.      $languages = language_list('enabled');
13.      $languages = $languages[1];
14.      $langcode = locale_language_from_browser($languages);
15.      $language = $languages[$langcode ? $langcode : 'en'];
16.      $url = url(drupal_is_front_page() ? '' : $_GET["q"], array('language' => $language));
17.      header("Location: $url", TRUE, $code);
18.      exit;
19.    }
20.  }
21. }
22. //----- locale_negotiation.inc -----
23. function _corp_common_locale_negotiation_en() {
24.   $used = &drupal_static(__FUNCTION__);
25.   $used = TRUE;
26.   return 'en';
27. }
28. function _corp_common_locale_negotiation_used() {
29.   $used = &drupal_static('_corp_common_locale_negotiation_en');
30.   return $used;
31. }
```

Developing Drupal site with native i18n support

- You know enough now
- Just do it!

ilya.gruzinov@volcanoideas.com