



DRUPALCAMP
SPAIN 2015
JEREZ DE LA FRONTERA

Drupal 8 Entity API





About Me





About Me

Drupal™

[Drupal Homepage](#)[Your Dashboard](#)[Logged in as tstoeckler](#)[Log out](#)

Download & Extend

[Download & Extend Home](#)[Drupal Core](#)[Distributions](#)[Modules](#)[Themes](#)

Libraries API

[View](#)[Edit](#)[Version control](#)[Revisions](#)[Automated Testing](#)

Posted by [sun](#) on *May 18, 2009 at 2:51pm*

The common denominator for all Drupal modules/profiles/themes that integrate with libraries.



About Me

Configuration Translation module

- Gábor Hojtsy 'Gábor Hojtsy' <http://drupal.org/user/4166>
- **Tobias Stöckler** 'tstoeckler' <https://drupal.org/user/107158>
- Vijayachandran Mani 'vijaycs85' <https://drupal.org/user/93488>



About Me

Shortcut module

- Tobias Stöckler 'tstoeckler' <https://www.drupal.org/user/107158>
- Jibran Ijaz 'jibran' <https://www.drupal.org/user/1198144>



DRUPALCAMP
SPAIN 2015
JEREZ DE LA FRONTERA

Drupal 8 Entity API





Entities in Drupal 6





Entities in Drupal 7

```
array_keys(entity_get_info()) === array(  
  'comment',  
  'node',  
  'file',  
  'taxonomy_term',  
  'taxonomy_vocabulary',  
  'user',  
);
```

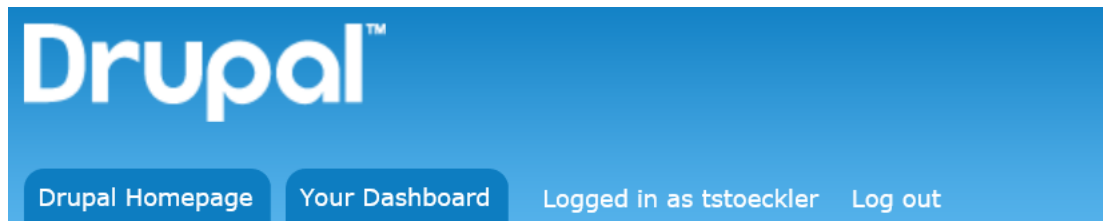



Problem: Properties vs. Fields

```
$node->title === 'The node title';  
is_array($node->body) === TRUE;  
$node->body[LANGUAGE_NONE][0] === 'The node body';  
$node->body['und'][0] === 'The node body';
```



Problem: Translatability



Download & Extend

[Download & Extend Home](#) [Drupal Core](#) [Distributions](#) [Modules](#) [Themes](#)

Title

[View](#) [Version control](#) [Revisions](#) [Automated Testing](#)

Posted by [plach](#) on *September 25, 2004 at 12:04pm*

While working on the new [content translation system](#) for Drupal 7, we (the Drupal c team) faced the need to convert node titles to the Field API in order to make nodes translatable.



Problem: Code Duplication

```
comment_save($comment);  
node_save($node);  
file_save($file);  
taxonomy_term_save($term);  
taxonomy_vocabulary_save($vocabulary);  
user_save($account);
```



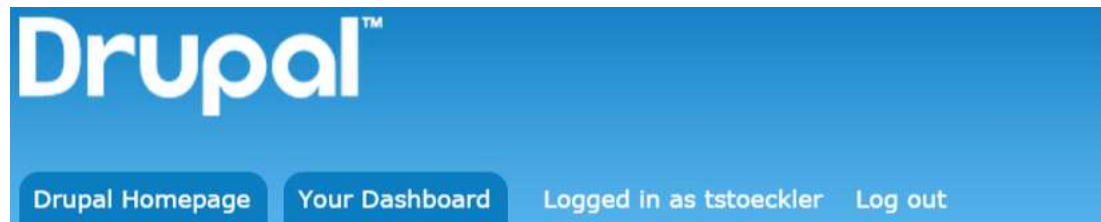
DRUPALCAMP
SPAIN 2015
JEREZ DE LA FRONTERA

Drupal 8 Entity API





Solution



Download & Extend

[Download & Extend Home](#) [Drupal Core](#) [Distributions](#) [Modules](#) [Themes](#)

Entity API

[View](#) [Version control](#) [Revisions](#) [Automated Testing](#)

Posted by [fago](#) on *November 7, 2009 at 2:37pm*

This module extends the entity API of Drupal core in order to provide a unified way with entities and their properties. Additionally, it provides an entity CRUD controller helps simplifying the creation of new entity types.



Entity Classes

```
class Node implements NodeInterface {}

function do_something(EntityInterface $node) {
    $nid = $node->id();
    $uuid = $node->uuid();
    $title = $node->label();
    $node_type_name = $node->bundle();

    $node->save();
    $node->delete();
}
```



Entity Classes

```
function do_something(NodeInterface $node) {  
    $title = $node->getTitle();  
    $node_type_name = $node->getType();  
}  
  
$new_node = Node::create([  
    'type' => 'article',  
    'title' => 'abc',  
]);  
$old_node = Node::load($nid);
```



Entity Classes

```
/**
 * Defines the node entity class.
 *
 * @ContentEntityType (
 *   id = "node",
 *   label = @Translation("Content"),
 * )
 */
class Node extends ContentEntityBase implements NodeInterface {
```




Config Entities





Entities in Drupal 7

```
array_keys(entity_get_info()) === array(  
  'comment',  
  'node',  
  'file',  
  'taxonomy_term',  
  'taxonomy_vocabulary',  
  'user',  
);
```



Entities in Drupal 8

Content

- Aggregator feed
- Aggregator feed item
- Comment
- Contact message
- Content
- Custom block
- Custom menu link
- File
- Shortcut link
- Taxonomy term
- User

Configuration

- Action
- Base field override
- Block
- Comment type
- Contact form
- Content Language Settings
- Content type
- Custom block type
- Date format
- Entity form display
- Entity view display
- Field
- Field storage
- Form mode
- Image style

Configuration

- Language
- Menu
- Migration
- RDF mapping
- Responsive image style
- Role
- Search page
- Shortcut set
- Taxonomy vocabulary
- Text Editor
- Text format
- Tour
- View
- View mode



Entities in Drupal 8





Entity Handlers

```
*   handlers = {
*     "storage" = "Drupal\node\nodeStorage",
*     "view_builder" = "Drupal\node\nodeViewBuilder",
*     "access" = "Drupal\node\nodeAccessControlHandler",
*     "views_data" = "Drupal\node\nodeViewsData",
*     "form" = {
*       "default" = "Drupal\node\nodeForm",
*       "delete" = "Drupal\node\nodeDeleteForm",
*       "edit" = "Drupal\node\nodeForm"
*     },
*     "route_provider" = {
*       "html" = "Drupal\node\nodeRouteProvider",
*     },
*   },
```



The Entity Manager

```
$entity_manager = \Drupal::entityManager();  
$storage = $entity_manager->getStorage('node');  
$access_handler = $entity_manager->getAccessControlHandler('node');  
$views_data = $entity_manager->getHandler('node', 'views_data');
```



Entity Classes

```
function do_something(NodeInterface $node) {  
    $title = $node->getTitle();  
    $node_type_name = $node->getType();  
}  
  
$new_node = Node::create([  
    'type' => 'article',  
    'title' => 'abc',  
]);  
$old_node = Node::load($nid);
```



Entity Classes

```
function do_something(NodeInterface $node) {  
    $title = $node->getTitle();  
    $node_type_name = $node->getType();  
}  
  
$storage = $entity_manager->getStorage('node');  
$new_node = $storage->create([  
    'type' => 'article',  
    'title' => 'abc',  
]);  
$old_node = $storage->load($nid);
```




Entity Query

```
$query = \Drupal::entityQuery('node');  
$ids = $query  
  ->condition('title', 'About', 'STARTS_WITH')  
  ->condition('created', 637200000, '>')  
  ->execute();  
$nodes = Node::loadMultiple($ids);
```



Entity Query

```
$query = $storage->getQuery();  
$ids = $query  
  ->condition('title', 'About', 'STARTS_WITH')  
  ->condition('created', 637200000, '>')  
  ->execute();  
$nodes = $storage->loadMultiple($ids);
```



Configuration Translation

Add German translation for *Article content type*



[Home](#) » [Administration](#) » [Structure](#) » [Content types](#) » [Article](#) » [Translate](#)

Name

Article

Name

Description

Use *articles* for time-sensitive content like news, press releases or blog posts.

Description

Explanation or submission guidelines

(Empty)

Explanation or submission guidelines

Save translation



Configuration Translation

▼ PAGER

▼ PAGED OUTPUT, FULL PAGER

▼ TAGS

Previous page link text < previous

Previous page link text

Next page link text next >

Next page link text

First page link text « first

First page link text

Last page link text last »

Last page link text

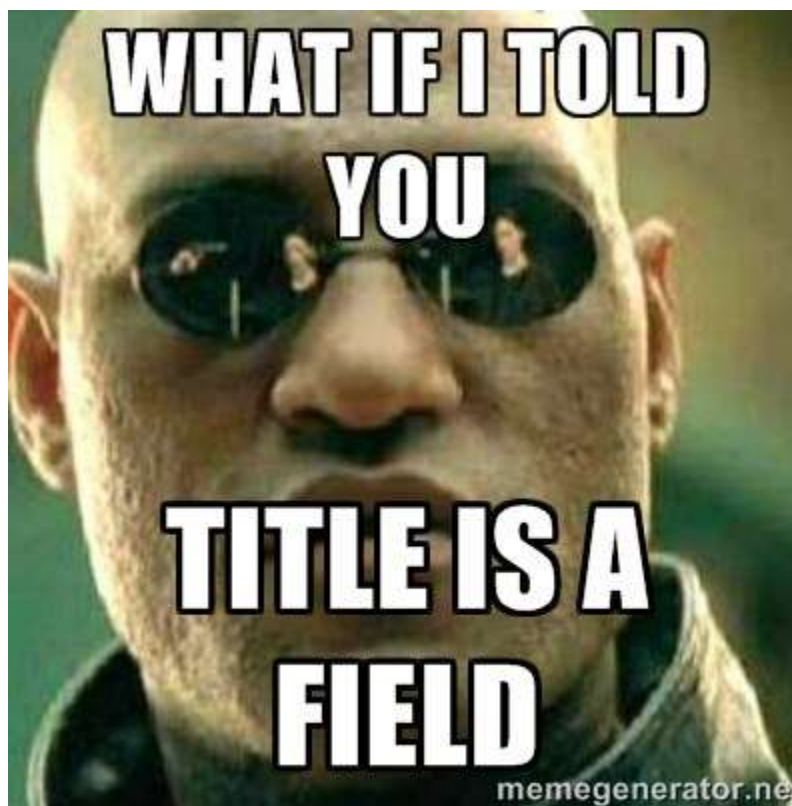


Configuration Translation

```
$storage = $entity_manager->getStorage('node_type');  
$node_type = $storage->load('article');  
$description = $node_type->getDescription();  
$description = $node_type->get('description');
```



Content Entities





Content Entities

Create *German translation of DrupalCamp Spain* ☆

[Home](#)

Title *

Body ([Edit summary](#))

B *I*



 Quellcode

DrupalCamp Spain is in the beautiful city of Jerez this year.



Content Entities

Manage form display ☆

[Edit](#)
[Manage fields](#)
[Manage form display](#)
[Manage display](#)
[Translate content type](#)

[Home](#) » [Administration](#) » [Structure](#) » [Content types](#) » [Article](#)

Content items can be edited using different form modes. Here, you can define which fields are shown and hidden when *Article* displayed in each form mode.

FIELD	WEIGHT	PARENT	WIDGET
Title	<input type="text" value="0"/>	<input type="text" value="- None -"/>	<input type="text" value="Textfield"/>
Body	<input type="text" value="1"/>	<input type="text" value="- None -"/>	<input type="text" value="Text area with a summary"/>
Tags	<input type="text" value="3"/>	<input type="text" value="- None -"/>	<input type="text" value="Autocomplete (Tags style)"/>
Image	<input type="text" value="4"/>	<input type="text" value="- None -"/>	<input type="text" value="Image"/>



Fields

```
['comment', 'datetime', 'file', 'image', 'link', 'list_float', 'list_integer',  
'list_string', 'path', 'text', 'text_long', 'text_with_summary', 'boolean',  
'changed', 'created', 'decimal', 'email', 'entity_reference', 'float',  
'integer', 'language', 'map', 'password', 'string', 'string_long',  
'timestamp', 'uri', 'uuid'];
```



Fields

```
$fields = $entity_manager->getFieldDefinitions('node', 'article');  
$fields['title']->getType() === 'string';  
$fields['body']->getType() === 'text_long';  
$fields['image']->getType() === 'image';
```



Fields

```
public static function baseFieldDefinitions(EntityTypeInterface $entity_type) {  
    $fields['title'] = BaseFieldDefinition::create('string')  
        ->setLabel(t('Title'))  
        ->setRequired(TRUE)  
        ->setTranslatable(TRUE)  
        ->setRevisionable(TRUE);  
    return $fields;  
}
```



Fields

```
class Gallery {  
  public static function baseFieldDefinitions(EntityTypeInterface $entity_type) {  
    $fields['images'] = BaseFieldDefinition::create('image')  
      ->setLabel(t('Images'))  
      ->setTranslatable(TRUE)  
      ->setRevisionable(TRUE)  
      ->setCardinality(FieldStorageDefinitionInterface::CARDINALITY_UNLIMITED)  
      ->setSetting('uri_scheme', 'private');  
    return $fields;  
  }  
}
```



Fields

General

- Boolean
- Comments
- Date
- Email
- Link

Number

- List (float)
- List (integer)
- Number (decimal)
- Number (float)
- Number (integer)

Reference

- Content
- File
- Image
- Taxonomy term
- User
- Other...

Text

- Text (plain, long)
- Text (formatted, long, with summary)
- List (text)
- Text (formatted)
- Text (formatted, long)
- Text (plain)



Working with Entities

```
$node->getTitle();  
$node->get('title')->value;  
$node->title->value;
```



Working with Content Entities

```
$node->get('tags')->target_id;  
$node->tags->value;
```

```
$node->get('tags')->first()->target_id;  
$node->tags[0]->target_id;
```

```
$node->get('tags')->get(2)->target_id;  
$node->tags[2]->target_id;
```



Working with Content Entities

```
$english_title = $node->getTitle();  
  
$translation = $node->getTranslation('de');  
  
$german_title = $translation->getTitle();
```




Working with Content Entities

```
$english_title = $node->getTitle();  
  
$node = $node->getTranslation('de');  
  
$german_title = $node->getTitle();
```



Configuration Translation

```
$storage = $entity_manager->getStorage('node_type');  
$node_type = $storage->load('article');  
$description = $node_type->getDescription();  
$description = $node_type->get('description');
```



Entity Storage

Drupal 8

Verify requirements

Overview

Review updates

Run updates

Review log

Drupal database update

The version of Drupal you are updating from has been automatically detected.

▼ 1 PENDING UPDATE

node entity type

- Update the *Content* entity type.

Apply pending updates



Thanks!