

Creating a Custom Drupal 8 Field Formatter

...

Michael Anello



Copyright 2016 Anello Consulting, Inc.

Michael Anello

- DrupalEasy/Anello Consulting, Inc. (drupaleasy.com)
- ultimike on drupal.org (drupal.org/u/ultimike)
- @ultimike

Concepts

- Hooks
- Object-orientation
- PSR-4
- Plugins
- Annotations
- Services
- Drupal Console
- Symfony and other external projects
- Composer

Concepts - Drupal Console

- A command-line tool built for Drupal 8.
- Similar to Drush, but evolved for Drupal 8.
- Provides:
 - Module "scaffolding" functionality
 - Cache rebuilding
 - Module download/install/uninstall
 - Much more

Concepts - Drupal Console

- <https://drupalconsole.com/>
- Extends Symfony Console
- Easy to install on Mac OS and Linux.

Hands-on Example

- HTML5 Audio module
 - Custom field formatter for fields of type "link".
 - Drupal Console provides the scaffolding.
 - We add code and a Twig template file to make it all work.
 - This module is the same one as used on [DrupalEasy.com](https://drupaleasy.com) (podcasts).

Hands-on Example

- Output will look something like this:

```
<audio controls preload="metadata">
  <source src="{link field url 1}" type="{link field url 1 type}">
  <source src="{link field url 2}" type="{link field url 2 type}">
  <source src="{link field url ...}" type="{link field url ... type}">
</audio>
```

Hands-on Example

- We will use Drupal Console to generate scaffolding.
- We will write just a little bit of code and a Twig template file.

Let's begin!

Step 1

- `drupal generate:module`
 - Drupal Console will prompt for details.
 - Review results.
 - Enable module.

Step 2

- `drupal generate:plugin:fieldformatter`
 - Drupal Console will prompt for details.
 - Review results.
 - Methods.

Step 3

- Configure D8 site to use the formatter
 - Add a new, multivalued "Audio URL" field of type "Link" to the "article" content type.
 - Set it to use our new formatter.
 - Create a new node of type "article".
 - <http://hackerpublicradio.org/eps/hpr1941.ogg>
 - <http://hackerpublicradio.org/eps/hpr1941.mp3>

Step 4

- Create template file
 - /templates/audio-tag.html.twig

```
<audio controls preload="metadata">
  {% for source in sources %}
    <source src="{{ source.src }}" type="{{ source.mimetype }}" />
  {% endfor %}
</audio>
```

Step 5

- Implement hook_theme in .module file

```
/**
 * Implements hook_theme().
 */
function html5_audio_theme() {
  return array(
    'audio_tag' => array(
      'variables' => array('sources' => NULL),
    ),
  );
}
```

Step 6

- Connect the dots
 - Add code to the viewElements() method of the formatter class.
 - Messages the field data.
 - Outputs variables requested by the template.

Step 6 (continued)

```
/**
 * {@inheritdoc}
 */
public function viewElements(FieldItemListInterface $items, $langcode) {
    $elements = [];

    // Render all field values as part of a single <audio> tag.
    $sources = array();
    foreach ($items as $delta => $item) {
        // Get the mime type.
        $mimetype = \Drupal::service('file.mime_type.guesser')->guess($item->uri);
        $sources[] = array(
            'src' => $item->uri,
            'mimetype' => $mimetype,
        );
    }

    // Put everything in an array for theming.
    $elements[] = array(
        '#theme' => 'audio_tag',
        '#sources' => $sources,
    );

    return $elements;
}
```

Step 7

- We're done!
 - Well - as always, rebuild caches...

Questions?

- Want the code?
 - https://github.com/ultimike/html5_audio
- This example (and two others) are part of DrupalEasy's "Introduction to Drupal 8 Module Development" workshop an 12-week Drupal Career Online program.