

DRUPALDELPHIA 2016

DRUPAL BEYOND THE
BROWSER: USING DRUPAL TO
POWER APPS AND
TOUCHSCREENS

BLUECADET

- > **MARK LLOBRERA - TECHNOLOGY DIRECTOR @
BLUECADET**
 - > Twitter: @dirtystylus
- > **PUTRA ROEUNG - SENIOR DEVELOPER @
BLUECADET**
 - > Twitter: @putrabon



HEAT TREATING THE HIGH HOUSE

WHAT IS HEAT TREATING?
If you look towards the bridge to your left, you can see a tall and narrow building. This is the High House, or No. 3 Treatment Facility. Its tall shape accommodated long steel beams. Here gun barrels and other long steel pieces were heated and cooled to improve the metal's properties. Annealing, a type of heat treating, softens the steel, relieves internal stresses, and improves the grain. The result is a better combination of strength, elasticity, and plasticity.

EXPERIENCE

EACH PIECE IS PLACED
in a vertical furnace and heated. Excess gases from the blast furnaces are burned as the fuel source for this heat. The steel must stay at temperature for several hours.

WORKERS "QUENCH"
or cool the steel quickly in water or oil. The steel's properties change depending on how fast it cools.

2

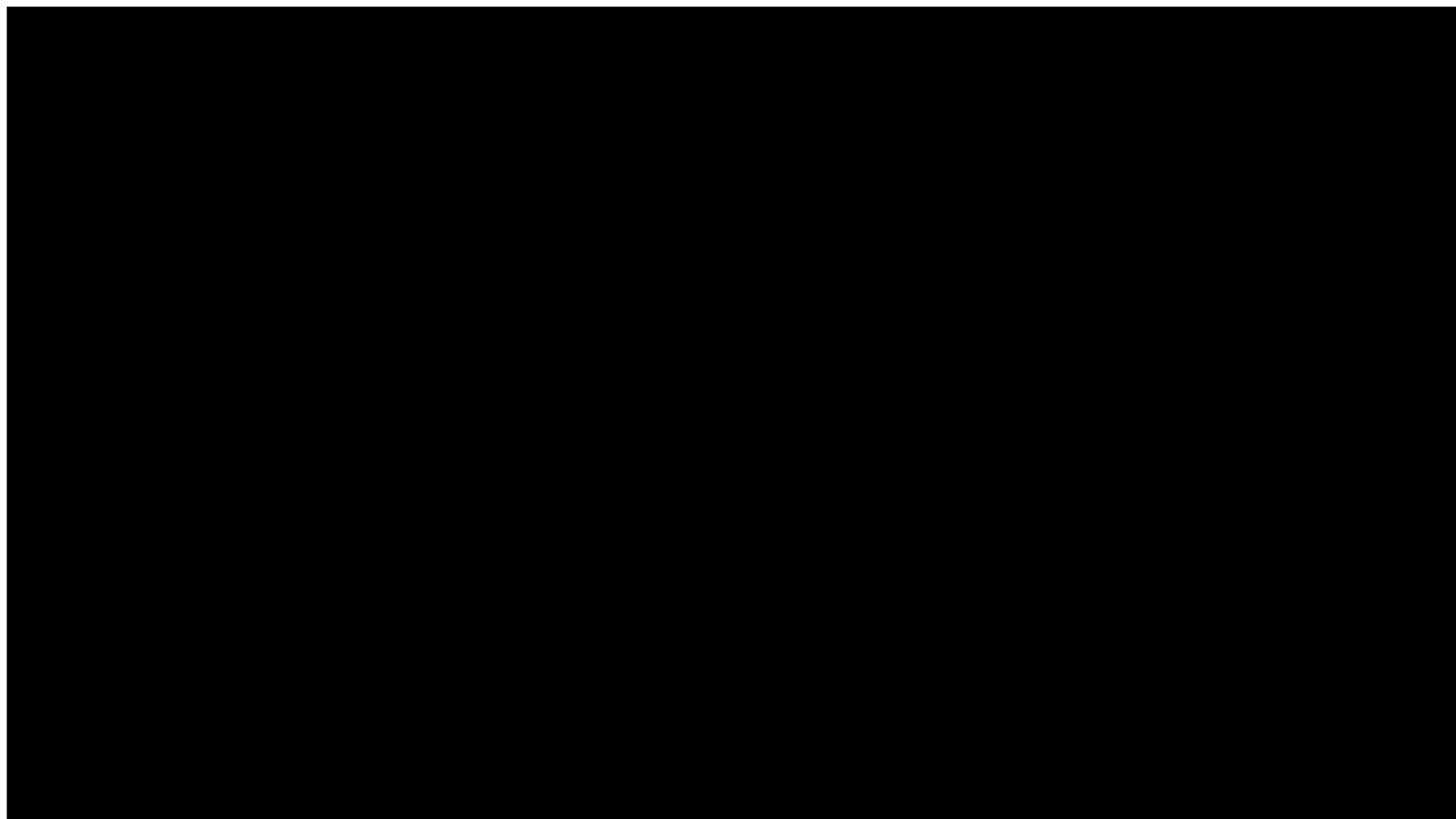
3



OUTLINE

- > STORY TIME: HOW DID WE GET HERE?
 - > CASE STUDY: FIELD MUSEUM
 - > CASE STUDY: VAN GOGH
- > RECIPE: DRUPAL 7 + SERVICES
 - > DRUPAL 8 DEMO

A STORY: CONTENT MANAGEMENT THE HARD WAY



[HTTP://BLUECADET.COM/WORK/NATIVE-AMERICAN-VOICES-THE-
PEOPLE-HERE-AND-NOW/](http://bluecadet.com/work/native-american-voices-the-people-here-and-now/)

A STORY: CONTENT MANAGEMENT THE HARD WAY

- > NATIVE AMERICAN VOICES EXHIBIT AT THE PENN MUSEUM
 - > HUNDREDS OF ARTIFACTS (~400)
 - > CONTENT MANAGEMENT WITH STATIC XML AND LOCAL FILES

4GIFs.com



LITTLEPAWZ.TUMBLR.COM

**XML IS NOT MEANT TO BE
HAND-AUTHORED. IT'S AN
OUTPUT ARTIFACT.**

- A SMARTER PERSON THAN ME

PAIN = SIGNAL

CONNECTING THE DOTS

- › WEB STUFF OVER HERE. LOTS OF CMS/WEBSITE BUILDS
- › APPLICATIONS OVER THERE (IOS. ANDROID. CINDER)

START SMALL

- > WE BUILT A CMS USING SOMETHING THAT RHYMES WITH 'BIRD DRESS'
 - > JSON, NOT XML
 - > PREPROCESSING, NOT 'LIVE'

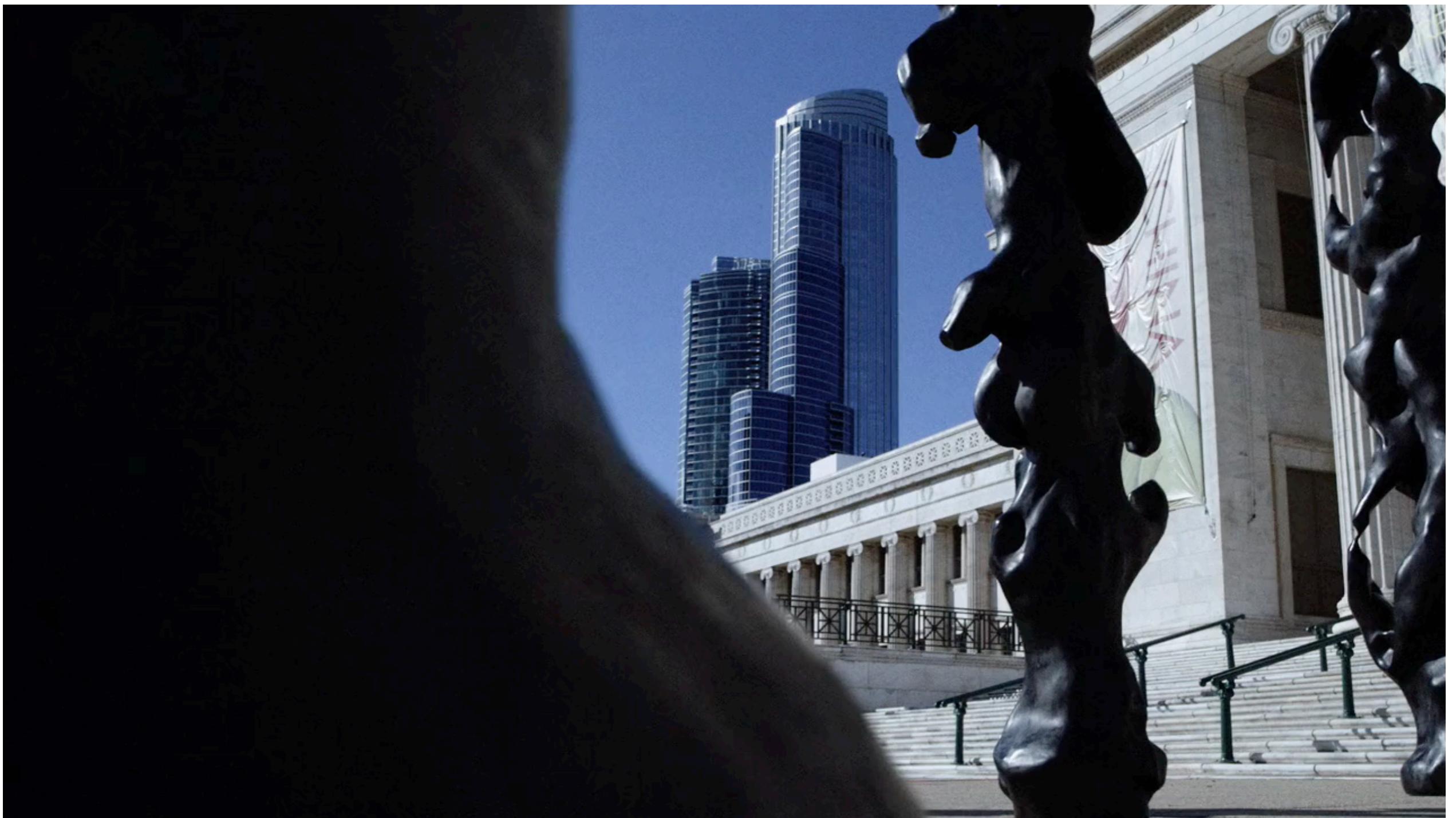
CASE STUDY: FIELD MUSEUM



DAVID THOMPSON ([HTTPS://WWW.FLICKR.COM/
PHOTOS/39023889@N00/52377145](https://www.flickr.com/photos/39023889@N00/52377145))

CASE STUDY: FIELD MUSEUM

- › DIGITAL ORIENTATION SCREENS FOR FINDING OUT INFORMATION ON EXHIBITS, EVENTS, AMENITIES
- › TWELVE 55" SCREENS THROUGHOUT THE MUSEUM
- › CLIENT TEAM WAS FAMILIAR WITH DRUPAL, SO WE LOOKED INTO WHAT WE COULD DO WITH DRUPAL



[HTTPS://VIMEO.COM/160897482](https://vimeo.com/160897482)

CASE STUDY: FIELD MUSEUM

- › DRUPAL 7. SERVICES MODULE PLUS SERVICES VIEWS
- › SCREENS NEEDED TO DOWNLOAD CONTENT LOCALLY. USED BASH SCRIPTS TO RSYNC JSON AND IMAGES FROM SERVER TO LOCAL NETWORK.

CASE STUDY: FIELD MUSEUM

CUSTOM MODULE TO "MARK" FEEDS THAT NEEDED UPDATING

- > **TODO: TRY NODEJS**
- > APP ONLY WANTED TO SYNC JSON DATA/IMAGES FOR NEW CONTENT
- > SERVICES VIEWS COULD CREATE OUTPUT FOR THE LAST CREATED/MODIFIED NODE OF EACH TYPE
- > **BUT: DELETING A NODE WOULD NOT BE REFLECTED**

CASE STUDY: FIELD MUSEUM

CUSTOM MODULE TO "MARK" FEEDS THAT NEEDED UPDATING

- `hook_node_presave()`
- `hook_node_delete()`
- `hook_taxonomy_term_presave()`
- `hook_taxonomy_term_delete()`

CASE STUDY: FIELD MUSEUM

CUSTOM MODULE TO "MARK" FEEDS THAT NEEDED UPDATING

```
function field_dos_feed_status_node_presave($node) {  
  switch ($node->type) {  
    case 'amenity':  
      variable_set('amenities_timestamp', time());  
      break;  
    case 'behind_scenes_item':  
      variable_set('behind_scenes_items_timestamp', time());  
      break;  
    case 'collection_highlight':  
      variable_set('collection_highlights_timestamp', time());  
      break;  
    ...  
    default:  
      break;  
  }  
}
```

CASE STUDY: FIELD MUSEUM

CUSTOM MODULE TO "MARK" FEEDS THAT NEEDED UPDATING

```
function field_dos_feed_status_menu_object() {  
  $feeds = array(  
    array('amenities' => variable_get('amenities_timestamp', '')),  
    array('amenity-categories' => variable_get('amenity_categories_timestamp', '')),  
    array('behind-the-scenes-items' => variable_get('behind_scenes_items_timestamp', '')),  
    ...  
    array('touchscreen-stations' => variable_get('touchscreen_stations_timestamp', '')),  
  );  
  return $feeds;  
}
```

CASE STUDY: FIELD MUSEUM

CUSTOM MODULE TO "MARK" FEEDS THAT NEEDED UPDATING

[

```
{"amenities":1449786054},  
{"amenity-categories":1450716038},  
{"behind-the-scenes-items":1449606828},  
{"collection-highlights":1449607778},  
{"exhibit-items":1451915487},  
{"itineraries":1449608094},  
{"locations":1451915570},  
{"schedule":1454972584},  
{"touchscreen-stations":1439735864}
```

]

CASE STUDY: FIELD MUSEUM

FRONT END

- > ANGULARJS (WAIT BUT I THOUGHT YOU SAID NO BROWSERS)
- > TWEENMAX ([HTTP://GREENSOCK.COM/TWEENMAX](http://greensock.com/tweenmax))

CASE STUDY: FIELD MUSEUM

FRONT END



CASE STUDY: FIELD MUSEUM

FRONT END

- > SINGLE TOUCH
- > TYPOGRAPHY IN CINDER IS STILL . . . DEVELOPING

CASE STUDY:



CASE STUDY: VAN GOGH'S BEDROOMS

- > INSTALLATION BRINGING THREE VERSIONS OF VAN GOGH'S BEDROOM TOGETHER
 - > PROJECTION / TOUCHSCREEN



[HTTPS://VIMEO.COM/155694123](https://vimeo.com/155694123)

CASE STUDY: VAN GOGH'S BEDROOMS

- > CMS ORIGINALLY IN WORDPRESS FOR INITIAL BUILD. CLIENT STANDARDIZED ON DRUPAL SO WE REBUILT THE CMS
- > DRUPAL 7. SERVICES AND SERVICES VIEWS TO PROVIDE JSON
- > CINDER (C++) ([HTTPS://WWW.LIBCINDER.ORG](https://www.libcinder.org)) APPLICATION

D7 RECIPE

- › D7: INSTALL SERVICES MODULE AND SERVICES VIEWS
- › DRUPALIZE.ME SERIES: [HTTPS://DRUPALIZE.ME/VIDEOS/INTRODUCTION-BUILDING-SERVICES-DRUPAL-7-SERIES?P=1487](https://drupalize.me/videos/introduction-building-services-drupal-7-series?P=1487)

D7 RECIPE

CREATE A VIEW WITH A SERVICES DISPLAY

Displays

The screenshot shows the 'Displays' configuration page for a view named 'Services'. The top navigation bar includes 'Page', 'Services' (which is selected), and '+Add'. There is also a 'edit view name/description' dropdown.

The main configuration area is divided into several sections:

- Title:** Display name: Services
- Format:** Unformatted list
- Fields:** Title (title), Overview (overview), Image (image), Map Popup Image (map_popup_image)
- Services settings:** Path: /exhibit-items, Access: Permission | View published content
- Advanced:** Header (Add), Footer (Add), Pager (Use pager: Display all items | All items, More link: No)

A sidebar on the left lists available services: Attachment, Block, Date browser, Entity Reference, Feed, Page, Services, and System. The 'Services' option is currently selected.

D7 RECIPE

ADD FIELDS. ASSIGN VALUE KEYS

For This services (override) ▾

Appears in: node:exhibit_item.

Set custom value key

Custom value key

hall_name

Place a colon after the label

Exclude from display

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

Formatter

Text

Apply (this display) Cancel Remove

D7 RECIPE

CONFIGURE SERVICE

Edit Server Authentication Resources Export

Machine-readable name of the endpoint *

The endpoint name can only consist of lowercase letters, underscores, and numbers.

Server *

REST

Select the server that should be used to handle requests to this endpoint.

Path to endpoint *

Debug mode enabled
Useful for developers. Do not enable on production environments

Authentication

Session authentication

Choose which authentication schemes that should be used with your endpoint. If no authentication method is selected all requests will be done by an anonymous user.

Save

D7 RECIPE

SELECT RESPONSE FORMATS

Edit Server Authentication Resources Export

REST

Response formatters *

bencode
 json
 jsonp
 php
 xml

Select the response formats you want to enable for the rest server.

Request parsing *

application/json
 application/vnd.php.serialized
 application/x-www-form-urlencoded
 application/xml
 multipart/form-data
 text/xml

Select the request parser types you want to enable for the rest server.

Save

D7 RECIPE

SELECT RESOURCES (YOUR VIEW)

Edit Server Authentication Resources Export

Resources

Select the resource(s) or methods you would like to enable, and click *Save*.

<input type="checkbox"/> Resource	Settings
<input checked="" type="checkbox"/> ▾ amenities	
CRUD operations	
<input checked="" type="checkbox"/> index	View: Amenities
<input checked="" type="checkbox"/> ▾ amenity-categories	
CRUD operations	
<input checked="" type="checkbox"/> index	View: Amenity Categories
<input type="checkbox"/> ▶ amenity-categories-es	
<input checked="" type="checkbox"/> ▾ behind-the-scenes-items	
CRUD operations	
<input checked="" type="checkbox"/> index	

D7 RECIPE

OUTPUT

```
▼ [ ▼ { "category": [ "Special Exhibitions" ], "feature_on_homepage": "1", "hall_name": "Fay and Daniel Levin Hall", "homepage_image": "sites/default/files/styles/homepage_attract/public/11B.%20Clay%20Vase%20Detail_DOS.jpg?itok=0ED_Ewe", "image": "sites/default/files/styles/exhibit_item_main/public/9.%20Gold%20Gorgon%20Head_DOS.jpg?itok=kcUfog6y", "location_id": [ "main_20" ], "node_id": "578", "overview": "<p>Journey through 5,000 years of Greek history and culture and meet some of the earliest rulers, warriors, and heroes of the ancient world in <em>The Greeks\u2014Agamemnon to Alexander the Great</em>. </p>\n", "title": "The Greeks\u2014Agamemnon to Alexander the Great" }, ▼ { "category": [ "Human Cultures", "Special Exhibitions" ], "feature_on_homepage": "0", "hall_name": "Webber Gallery", "homepage_image": "sites/default/files/styles/homepage_attract/public/Spooning%20couple%202.jpg?itok=zhU6nB18" } ]
```

Toggle Formatting



DRUPAL 8

SO WHAT DO WE GET OUT OF THE BOX?

- > RESTFUL WEB SERVICES (REST)
- > SERIALIZATION (SERIALIZATION)
- > HYPERTEXT APPLICATION LANGUAGE (HAL)
- > HTTP BASIC AUTHENTICATION (BASIC_AUTH)

3 WAYS TO CREATE A REST API WITH DRUPAL 8

- › **OPTION: #1 - USING DRUPAL 8 CORE REST RESOURCES**
- › **OPTION: #2 - USING VIEW REST EXPORTS**
- › **OPTION: #3 - CREATE CUSTOM REST ENDPOINT**

OPTION: #1 - USING DRUPAL 8 CORE REST RESOURCES

RESOURCE NAME	PATH	DESCRIPTION	OPERATIONS
Action	/entity/action/{action}		<button>Enable</button>
Base field override	/entity/base_field_override/{base_field_override}		<button>Enable</button>
Block	/entity/block/{block}		<button>Enable</button>
Comment	/comment/{comment}		<button>Enable</button>
Comment type	/entity/comment_type/{comment_type}		<button>Enable</button>
Contact form	/contact/{contact_form}		<button>Enable</button>
Contact message	/entity/contact_message/{contact_message}		<button>Enable</button>
Content type	/entity/node_type/{node_type}		<button>Enable</button>
Custom block	/block/{block_content}		<button>Enable</button>
Custom block type	/entity/block_content_type/{block_content_type}		<button>Enable</button>
Custom menu link	/admin/structure/menu/item/{menu_link_content}/edit		<button>Enable</button>

DEMO

OPTION #1

PROS

- > STRAIGHT OUT OF THE BOX
- > REQUIRES ALMOST NO SETUP
- > NO CUSTOM CODE NECESSARY

CONS

- > ABSOLUTELY NO FLEXIBILITY
- > LACKS ABILITY TO SET CUSTOM PARAMETERS
- > UNABLE TO LIMIT OUTPUT

OPTION: #2 - USING VIEW REST EXPORTS

Displays

The screenshot shows the 'Displays' configuration page for a 'REST export' view. The view is titled 'REST export' and has a path of '/api/artworks'. It uses the 'Serializer' format and has 'None' as its title. The view includes fields for Title, Node ID (nid), mkey (mkey), Credit Line (credit_line), Dimensions (dimensions), Changed (changed), Path (node_path), Publishing status (= Yes), and Type (= Artwork). The 'Content' field type is selected. The 'Edit view name/description' and 'View REST export' buttons are visible at the top right. The 'ADVANCED' section contains settings for Contextual filters, Relationships, and other machine-specific details.

Display name: REST export

TITLE
Title: None

FORMAT
Format: Serializer / Settings
Show: Fields / Settings

FIELDS
Content: Title
Content: Node ID (nid)
Content: mkey (mkey)
Content: Credit Line (credit_line)
Content: Dimensions (dimensions)
Content: Changed (changed)
Content: Path (node_path)

FILTER CRITERIA
Content: Publishing status (= Yes)
Content: Type (= Artwork)

SORT CRITERIA

PATH SETTINGS
Path: /api/artworks
Access: Permission | View published content

HEADER
The selected display type does not use header plugins

FOOTER
The selected display type does not use footer plugins

NO RESULTS BEHAVIOR
The selected display type does not use empty plugins

PAGER
Items to display: Full | Paged, 20 items

ADVANCED

CONTEXTUAL FILTERS

RELATIONSHIPS

OTHER
Machine Name: rest_export_1
Administrative comment: None
Use aggregation: No
Query settings: Settings
Caching: Tag based

DEMO

OPTION #2

PROS

- › IT'S STRAIGHT OUT OF THE BOX
- › MOST DEVELOPERS ARE FAMILIAR WITH VIEWS
- › MANAGE YOUR CONFIGS WITHIN THE UI

CONS

- › MORE FLEXIBILITY, BUT STILL LIMITED IN VARIOUS AREAS
 - › NO WAYS TO SET CUSTOM PARAMETERS
 - › AUTHENTICATION ISSUES

OPTION: #3 - CREATE CUSTOM REST ENDPOINT

```
use Drupal\rest\ResourceResponse;

/**
 *
 * @RestResource(
 *   id = "artwork_review",
 *   label = @Translation("Artwork reviews"),
 *   uri_paths = {
 *     "canonical" = "/reviews/{id}"
 *   }
 * )
 */

class ArtworkReviewResource extends ResourceBase {

  /**
   * Responds to GET requests.
   *
   * Returns a review entry for the specified ID.
   *
   * @return \Drupal\rest\ResourceResponse
   *   The response containing the log entry.
   *
   * @throws \Symfony\Component\HttpKernel\Exception\HttpException
   */
  public function get($id = NULL) {
    if ($id) {
      $record = db_query("SELECT * FROM {reviews} WHERE id = :rid", array(':rid' => $id))->fetchAssoc();
      if (!empty($record)) {
        return new ResourceResponse($record);
      }
    }
  }
}
```

PROS

- > PROVIDES MOST FLEXIBILITY
- > TRANSFORMABLE OUTPUT (SPECIAL ENTITIES)
- > EASIER TO MANAGE VERSIONS

CONS

- > REQUIRES REASONABLE PROGRAMMING KNOWLEDGE

POWER OF DECOUPLED ARCHITECTURES

- CONTENT MANAGEMENT CAN STAY STABLE
- (MULTIPLE) FRONTENDS CAN CHANGE. PROTOTYPE IN ONE LANGUAGE. BUILD IN ANOTHER
- IF YOU'RE ALREADY USING DRUPAL. THE BARRIER TO ENTRY IS LOW
- USE THE ENDPOINTS FOR ANY DEVICES

BLUECADET

- > **MARK LLOBRERA - TECHNOLOGY DIRECTOR @
BLUECADET**
 - > Twitter: @dirtystylus
- > **PUTRA ROEUNG - SENIOR DEVELOPER @
BLUECADET**
 - > Twitter: @putrabon

GITHUB

> [HTTPS://GITHUB.COM/DIRTYSTYLUS/
DRUPALDELPHIA-2016](https://github.com/dirtystylus/drupalphiladelphia-2016)