

Deploy your contents with
Entity Share



Who am I? Florent Torregrosa

Drupal Tech expert at [Smile](#).

Using Drupal since 2011.

Some contributions:

- Maintainer/Co-maintainer on 30+ projects
- Event co-organizer
- Patches
- French translations

[Grimreaper](#) on Drupal.org.

Summary

1. Why sharing content?
2. Why the Entity Share module?
3. Entity Share's architecture
 - a. JSON:API usage
 - b. Main modules and concepts
 - c. Import configuration
 - d. Ecosystem
 - e. Known problems and limitations
 - f. Perspectives and roadmap
4. Demo



1: Why sharing content?

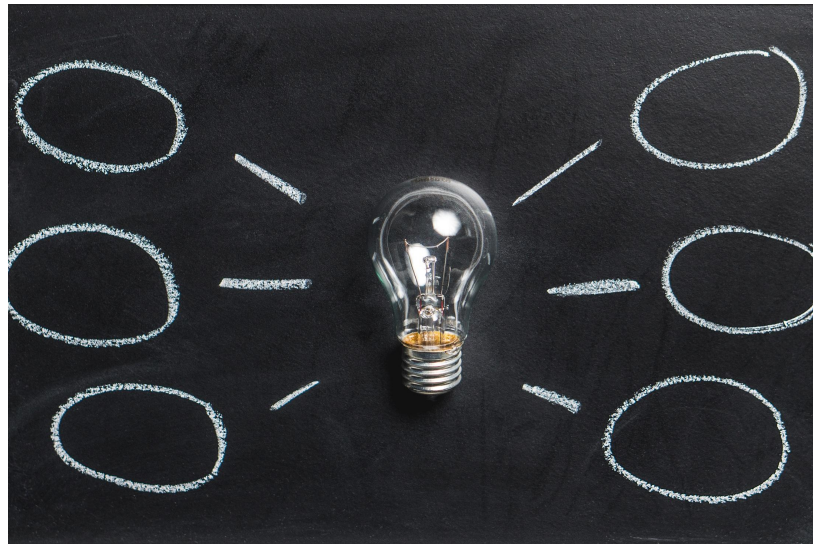


1: Why sharing content?

Deploy content on multiple websites.

Either for:

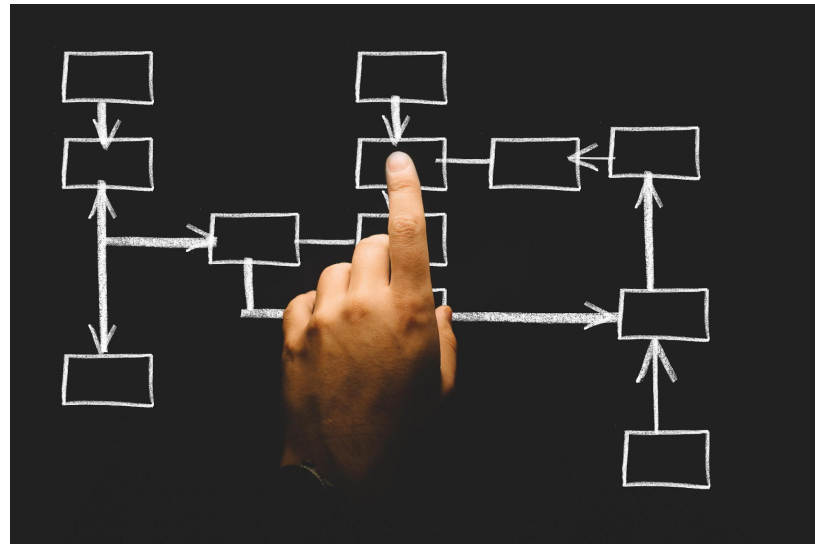
- project architecture with content hub
- cross-communications between different sites



1: Why sharing content?

In industrialization process

- Deploy content from preproduction to production (content staging), same as the content hub use case
- Retrieve content from production to development environments



2: Why the Entity Share module?



2: Why the Entity Share module?

Date context: **end of 2016**

Webfactory module: based on core Rest webservice, linked to the websites deployment feature (content sharing usage biased).

Deploy ecosystem: unstable, bound to workspace.

Acquia Content Hub: 3rd party, paid subscription.

Entity Pilot module: 3rd party, paid subscription (not known at this period).



3: Entity Share's architecture



3: Entity Share's architecture

A: JSON:API usage



- Perfect to provide entities listing
- Easy to use and adapt (with JSON:API Extras)
- In Core

Entity Share provides an UI on top of JSON:API to allow one website to retrieve content from another website.



3: Entity Share's architecture

B: Main modules and concepts



2 main sub-modules (both may be enabled on the same site).



3: Entity Share's architecture

B: Main modules and concepts



Entity Share Server:

- Enable on the website that will provide the content
- Provides the channels system (inspired from the Webfactory module):
 - Prepare a JSON:API endpoint URL to call by the client website:
 - entity type
 - bundle
 - language
 - filters
 - sorts
 - Which users can access the channel (roles and/or specific users)
- Exposes some server website content structure to ensure client website can retrieve data



3: Entity Share's architecture

B: Main modules and concepts



Back to site Manage admin Dev

Content Structure Appearance Extend Configuration People Reports Help

Edit *Es en*

Home » Administration » Configuration » Web services » Entity Share » Channel entities

Label *
Es en
Label for the channel.
Match name: es_en

Entity type *
Content

Bundle *
Entity share test

Language
English

Max size *
50
The JSONAPI's page limit option to limit the number of entities per page.

GROUPS

Add a new group

ID	CONJUNCTION	PARENT GROUP	OPERATIONS
There is currently no group for this channel.			

FILTERS

Add a new filter

ID	PATH	OPERATOR	VALUE	GROUP	OPERATIONS
There is currently no filter for this channel.					

SEARCHES

Add a new search

⚠ The label of the entity if it exists is automatically searchable. Do not add a search for that.

ID	PATH	LABEL	OPERATIONS
There is currently no search for this channel.			

SORTS

Add a new sort

ID	PATH	DIRECTION	WEIGHT	OPERATIONS
There is currently no sort for this channel.				

ACCESS

☒ All users with the Access channels list permission.

Authorized roles
☐ Administrator
Only roles with the Access channels list permission are listed.

Authorized users
☐ admin
Only users with the Access channels list permission are listed.

Save Delete

Example of channel



2022
DRUPAL DEV DAYS
April 4-8th Ghent, Belgium

3: Entity Share's architecture

B: Main modules and concepts



Entity Share Client:

- Enable on the website that will pull the content
- Allows to set the websites on which to connect to
 - Authentication method is plugin based for extensibility
 - The [Key](#) module is supported for proper handling of credentials
- Allows to configure “import config” to control how content is imported
- Allows to pull content:
 - Form (individual content or channel)
 - Drush commands (channel)
- Stores the information of which content had been pulled from which website and when in “Import entity status” content entities



3: Entity Share's architecture

B: Main modules and concepts



During the import:

1. Check if an entity exists with this UUID
 - a. If no entity is found, create a new one
 - b. If an entity already exists, create or update the translation regarding the language in the JSON data
2. Store the UUID in the processed entities list to avoid infinite loop

More details when we will see the import process plugins in the next chapter.



3: Entity Share's architecture

B: Main modules and concepts



default-web-entity-share.docker.localhost/admin/config/services/entity_share/remote/site_1/edit?destination=/admin/config/service

Back to site Manage admin Devel

Content Structure Appearance Extend Configuration People Reports Help

Edit Site 1

Home » Administration » Configuration » Web services » Entity Share » Remote entities

Label *

Site 1 Machine name: site_1

Label for the remote website.

URL *

https://site1-web-entity-share.docker.localhost

The remote URL. Example: http://example.com

Authorization methods

☒ Basic Auth

☐ OAuth2

☐ Header

☐ Anonymous

Credential provider

Local storage

STORED LOCALLY

Username

admin

Password

.....

⚠ With the Basic Auth authorization method you need to ensure that the *HTTP Basic Authentication* module is enabled on the server website.

Save Delete

Example of a remote website configuration.



DRUPAL DEV DAYS

April 4-8th Ghent, Belgium

3: Entity Share's architecture

B: Main modules and concepts



default-web-entity-share.docker.localhost/admin/content/entity_share/pull

Back to site Manage admin Devel

Content Structure Appearance Extend Configuration People Reports Help

Pull entities

Content Comments Entity Share Files Media

Pull entities Import statuses

Home » Administration » Content » Pull entities

Import configuration*

Default

Remote website*

Site 1

Channel*

Es en

Search

The search (CONTAINS operator) will occur on the following fields:

- Label

Synchronize entities Import the whole channel Import asynchronously Subscribe Unsubscribe

<input type="checkbox"/> LABEL	TYPE	BUNDLE	LANGUAGE	REMOTE ENTITY CHANGED DATE	STATUS	POLICY
<input type="checkbox"/> Es test new title (View local)	Content	Entity share test	English	Friday, April 1, 2022 - 19:07	Entities synchronized	Default

Synchronize entities Import the whole channel Import asynchronously Subscribe Unsubscribe

The pull form on the client website.



2022
DRUPAL DEV DAYS
April 4-8th Ghent, Belgium

3: Entity Share's architecture

B: Main modules and concepts



default-web-entity-share.docker.localhost/admin/content/entity_share/import_status

Back to site Manage admin Devel

Content Structure Appearance Extend Configuration People Reports Help

Entity import statuses

Content Comments Entity Share Files Media

Pull entities Import statuses

Home » Administration » Content » Pull entities

ID	ENTITY UUID	ENTITY ID	LANGUAGE	LINK TO ENTITY	ENTITY TYPE	BUNDLE	REMOTE	CHANNEL	LAST IMPORT	POLICY	OPERATIONS
1	8b2b8434-8314-4258-a03e-8130de4dfbf4	1	English	Canis_lupus_arctos.jpg	File	File	Site 1	es_en	April 1, 2022 - 19:05:36	Default	Delete
2	fe85046d-2e33-4064-be8f-8f8801895273	1	English	Es test new title > Paragraphs	Paragraph	Entity share test	Site 1	es_en	April 1, 2022 - 19:08:29	Default	Delete
3	fe854055-08c8-4e71-850f-d7c2d5ee87fa	1	English	Es test new title	Content	Entity share test	Site 1	es_en	April 1, 2022 - 19:08:29	Default	Delete

Example of listing of entity import statuses.



2022
DRUPAL DEV DAYS
April 4-8th Ghent, Belgium

3: Entity Share's architecture

C: Import configuration



The version 3 (8.x-3.x) introduced two concepts/plugin system:

- Import config and import process plugins
- Import policies plugins

With the addition of the “Import entity status” content entities, it allowed to solve bugs inherent to the previous architecture and helped to implement new features in a maintainable way. Providing an extension system.



3: Entity Share's architecture

C: Import configuration



Import configuration:

- Configure what happens and how during the import
- Unified way to control this behavior between UI and CLI
- Different behavior depending on your needs

Import process plugins:

- Inspired from Search API processor plugins
- Steps during import that triggers methods of enabled plugins
- Improved DX



DRUPAL DEV DAYS
April 4-8th Ghent, Belgium

3: Entity Share's architecture

C: Import configuration



Back to site | Menu | Admin | Search

Content | Structure | Appearance | Extensions | Configuration | People | Reports | Help

Edit Default

Home » Administration » Configuration » Web services » Entity Share » Import config entities » Default

Label *
Default
Label for the import config. Machine name: default

Max size *
50
The JSON API's page limit option to limit the number of entities per page.

ENABLED

- ☒ **Block field block content**
Import block contents from block fields. Require the 'Block field (Block field only) (Entity Share)' field enhancer enabled on both client and server websites.
- ☐ **Changed time**
Set the changed time to changed time from remote data.
- ☒ **Default data processor**
Define import policy and general JSON data preparation to have Entity Share import working.
- ☒ **Embedded entity**
Import embedded entities from text formatted fields. Require the 'Embedded entities (formatted text field only) (Entity Share)' field enhancer enabled on both client and server websites.
- ☐ **Entity reference**
Handle entity reference fields.
- ☐ **Language fallback**
Allow to set the language of the imported entity.
- ☒ **Link internal content**
Import internal content from link fields. Require the 'UUID for link (link field only) (Entity Share)' field enhancer enabled on both client and server websites.
- ☒ **Physical file**
When importing a file entity, also import the physical file.
- ☒ **Prevent update processor**
Prevent update of an already imported entity if the entity import status has the 'Create only' policy.
- ☐ **Revision**
Create new revision.
- ☒ **Skip already imported entities**
If the entities haven't changed on Remote, this plugin prevents saving of the entities.

PROCESSOR ORDER

PREPARE ENTITY DATA Show row weights	IS ENTITY IMPORTABLE Show row weights	PREPARE IMPORTABLE ENTITY DATA Show row weights	PROCESS ENTITY Show row weights	POST ENTITY SAVE Show row weights
<ul style="list-style-type: none">+ Default data processor+ Prevent update processor+ Skip already imported entities		<ul style="list-style-type: none">+ Default data processor+ Block field block content+ Embedded entity+ Link internal content	<ul style="list-style-type: none">+ Physical file+ Entity reference	<ul style="list-style-type: none">+ Default data processor

Processor settings

Block field block content Enabled	Maximum recursion depth -1 The maximum recursion depth, -1 for unlimited. When reaching max recursion depth, referenced entities are set if the entity already exists on the website.
Default data processor Enabled	
Embedded entity Enabled	
Entity reference Enabled	
Link internal content Enabled	
Physical file Enabled	

[Save](#) [Delete](#)

Example of import config configuration.



2022
DRUPAL DEV DAYS
April 4-8th Ghent, Belgium

3: Entity Share's architecture

C: Import configuration



Import policies:

- Declared as YAML plugins
- Allow to “mark” imported content in the related “import entity status” entity
- So then contrib or custom code can react depending on the import policy
- In Entity Share:
 - Default: no special behavior
 - Create only: use the related import processor to skip updating already imported entities
 - Locked editing: disable the edit form on the client website



3: Entity Share's architecture

C: Import configuration



It allowed to:

- Fix status detection from the server website
- Select recursion depths for entity reference
- Skip synchronization of imported entities (CLI only before)
- Select revision creation
- Prevent edition of shared entities
- Allow shared entities to be updated on client website
- Pull content in a language not enabled
- Parse RTE/link/block fields to get referenced entities like for entity reference fields
- And many more improvements!



3: Entity Share's architecture

D: Ecosystem



Entity Share Async (sub-module): to mark content to be pulled later by a queue during cron execution.

Entity Share Diff (sub-module): provides a basic diff feature on the pull form.

Entity Share ECA (sub-module): integration with the [ECA](#) module.

Entity Share Lock (sub-module): prevents to edit content once imported.

[Entity Share Cron](#): provides an UI to configure frequency of automated pull of channels. For more complex usage, use custom code.

[Entity Share Websub](#): allows the client websites to “subscribe” for specific content. When the content is changed, the server website will send a notification and all subscriber websites will (almost) instantly pull the changed content.

[And more!](#)



3: Entity Share's architecture

E: Known problems and limitations



- Intensive usage of JSON:API Extras to avoid core limitations or handle special fields.
- Implementation relies on some JSON:API internal classes ([#2939827](#))
- To avoid side effects, config entities and users are not handled.
- Push form ([#2856715](#)):
 - Impossible to PATCH translations (JSON:API)
 - Impossible to PATCH file field (JSON:API)



3: Entity Share's architecture

F: Perspectives and roadmap



Stable release:

- Path alias update handling ([#3107278](#))
- Infinite loop in special fields ([#3265613](#))

V4:

- D10 preparation ([#3248631](#), [#3251411](#))
- User sharing ([#3175111](#))





Thank you



To our wonderful sponsors, our awesome community and fantastic volunteers!

Platinum sponsors



Thank you



Gold sponsors



wieni_



Silver sponsors

AMPLEXOR



Acquia



2022
DRUPAL DEV DAYS
April 4-8th Ghent, Belgium

Stay in touch



#ddd2022 on Drupal slack



@drupaldevdays



/drupaldevdays



2022
DRUPAL DEV DAYS
April 4-8th Ghent, Belgium