

A Migration for All Seasons: Getting Data into Drupal with Custom Migrations



Matthew Messmer
Drupal Architect
at Clarity Partners

MIGRATION DEMO REPO

[https://github.com/matthewmessmer/
migration-demo](https://github.com/matthewmessmer/migration-demo)

Overview

What we will cover today

- **Use cases for migrations**
- **Contrib modules and drush commands**
- **Writing custom source and process plugins**

OUR USE CASE:

**Get data into Drupal
and reduce the time spent
on content entry**

Why use the Migrate API?

- 1. Data is usually interconnected with references**
- 2. I'm usually loading the site with many types of content at the same time**
- 3. Built in system for tracking which content is new and which is updated**
- 4. Built in system for rolling back (deleting) content**
- 5. Can be used once and then removed without causing a problem**
- 6. Can use migration map table as reverse lookup for Drupal nid/tid**

When are migrations a good fit? Or when are migrations a bad fit?

Points to consider:

- **Level of effort vs other data entry methods**
- **Does the data have a fixed pattern?**
- **Do you need real time data?**

Other potential pitfalls:

- **Migrating content with lots of references from spreadsheets can be trouble**
- **Migrating WYSIWYG fields can be problematic because of special codes**
- **Need to write code**

CONTRIB MODULES

Drupal Core's Migrate Module

- Contains the base functionality for migrations

Migrate Plus (https://www.drupal.org/project/migrate_plus)

- Allows import/export of migration config as yml
- Allows migrations from apis and files (xml, json, etc)

Migrate Tools (https://www.drupal.org/project/migrate_tools)

- Provides additional drush integration
- Provides a `--sync` option when running migrations
- Provides the `MigrateExecutable` class that lets us run migrations in code

Migrate Source CSV (https://www.drupal.org/project/migrate_source_csv)

- Provides a CSV source plugin

DRUSH COMMANDS

Drush Commands

drush ms

- Lists your migrations and their status

drush mim migration_name

- Import the migration
- Useful flags are `--group`, `--limit`, `--idlist`, `--update`, `--execute-dependencies`

drush mr migration_name

- Rollback (delete) all content created previously

drush mrs migration_name

- Reset a migration to idle

drush mmsg migration_name

- Shows messages when a migration encounters a problem

CREATING MIGRATIONS

Extract – Transform – Load (ETL)

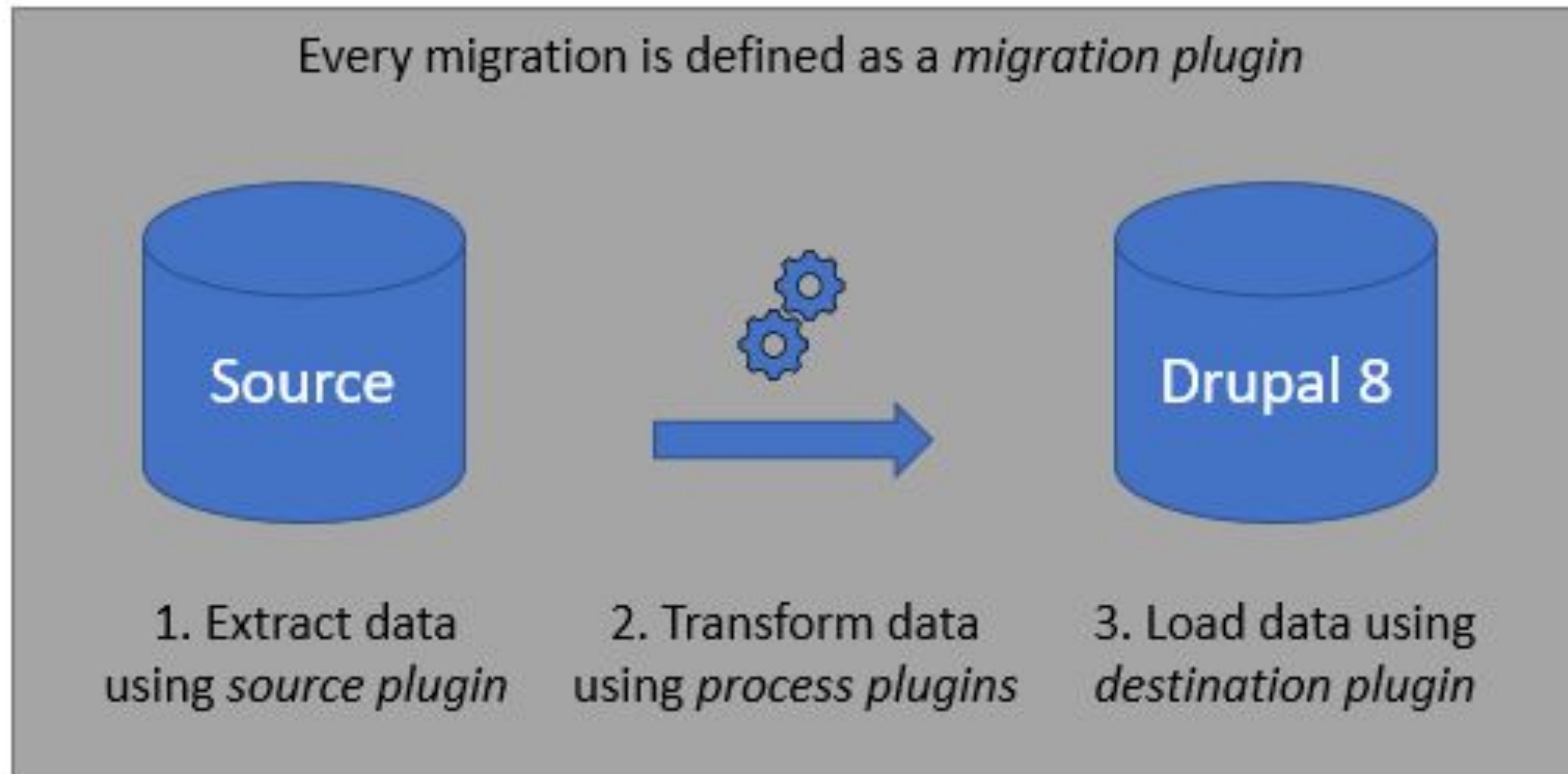


Image source: <https://www.drupal.org/docs/drupal-apis/migrate-api/migrate-api-overview>

Readymade Process Plugins

- <https://www.drupal.org/docs/8/api/migrate-api/migrate-process-plugins/list-of-core-migrate-process-plugins>
 - **default_value**
 - **format_date**
 - **static_map**
 - **migration_lookup**
 - **skip_on_empty**
- <https://www.drupal.org/docs/8/api/migrate-api/migrate-process-plugins/list-of-process-plugins-provided-by-migrate-plus>
 - **skip_on_value**
 - **entity_lookup**

Migration Demo

Parks Migration

- Node migration from JSON API
- Custom source plugin which calls secondary api to load additional data

Staff Migration

- Node migration from CSV with linked migrations to create media and file entities
- Custom process plugin to only migrate value if field is empty

Ratings Migration

- Taxonomy migration from CSV linked to Movies migration

Movies Migration

- Node migration from JSON API
- Custom Migrate Plus fetcher plugin which allows dynamic filtering
- Prepare row hook to dynamically set field value

LIVE DEMO

fingers crossed

PLEASE PROVIDE YOUR FEEDBACK!

mid.camp/6893

**THANK YOU,
QUESTIONS?**

m.messmer@claritypartners.com

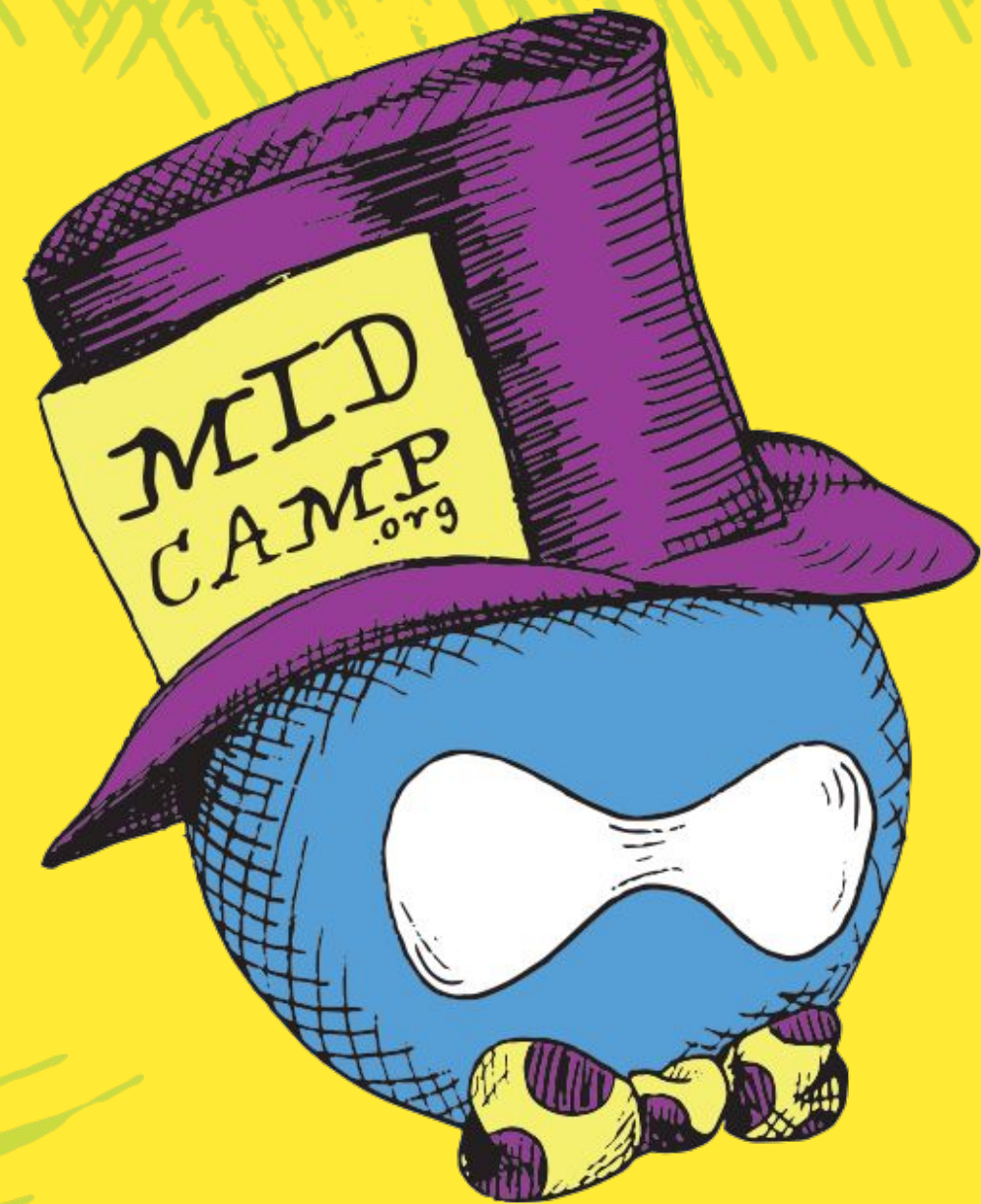


CONTRIBUTION DAY

Friday 10am to 4pm

You don't have to know code to give back!

New Contributor training 10am to Noon
with **AmyJune Hineline** of [Opensource.com](https://opensource.com)



Drupal Coffee Exchange at MidCamp

Bring a bag of beans,
Leave with a bag of beans.



That's how it works.

MidCamp is in Chicago April 26–28