

Rearchitecting for Drupal 8

By frob

- Welcome to the re-architecting for Drupal 8 talk
- You should be able to follow along by visiting the link below.

In case you are wondering



- Prolog <-- **You are here in the session**
- Why look at Drupal 8
- Custom LMS Drupal 7 architecture
 - Features
 - Content strategy
 - Modules used

- Prolog <-- **You are here in the session**
- Custom development
- Why look at Drupal 8
- Considerations Building with Drupal 8
 - Custom LMS Drupal 7 architecture
 - Architectural differences
 - Features
 - Content strategy
 - Modules used
 - Module availability
 - Custom development
 - Specifically What needs to change
- UI Changes
- Module availability
- Specifically What needs to change

Hello My Name Is Frank

- Hello my name is Frank Robert Anderson
- I am a Father, and Technology Enthusiast.
- Web development since high school
- *Online my name is frob (IRC, d.o, github)*
- Been working in Drupal professionally for just shy of 10 years.
- *On Twitter I am @frobdfas*
- I have worked almost exclusively in the Education industry
- *My Blog is www.frobiovox.com*
- for nearly 4 years.
- Online you can find me as frob
- *I work for Clarity Innovations Inc.*
 - IRC
 - github
 - drupal.org
- Twitter frobdfas
- blog
- website
- I work for Clarity Innovations.



Clarity Innovations

- Clarity Innovations is a professional services firm based in Portland, Oregon focused on providing K-12 and higher education technology consulting to non-profits, schools, and corporations.
- Our 25 employees combine leading edge technology and design skills with direct experience in the classroom and university.
- We develop solutions for our clients including strategic consulting, professional development and content creation, and engineering solutions such as apps, web apps, and websites to help improve the process and practice of teaching and learning.

Welcome! Begin here to view videos and participate in activities supporting your work with the Contexts for Learning Mathematics. Or you may wish to visit [Digging Deep](#), to study more intensely with us. Enjoy, and don't be shy about letting us hear from you in the [Forum](#)!

School Announcements

This is a test school for Clarity Innovations' internal use.

CFLM UNIT SUPPORT

DIGGING DEEP COURSES

Grades K to 3

Minilessons and Representation

Questioning and Conferring

Grades 3 to 5

Deepening My Math

Filling Out the Landscape

Grades 4 to 6

The Role of Context

Proof

 FORUMS

Why do this?

- Why
- The big question is D8 or D7, this is a question that I have to ask myself every time I am asked to do an estimate.
- How do we pick which version of Drupal to use?
- Let me **GET THIS OUT THE WAY** I will not be going over hard numbers for building with one version over the other. I will give you the only answer that I can give knowing what I know about your project's requirements. IT DEPENDS.

- I create with Drupal 7.
- I want to create with Drupal 8.
- I want to know the differences between creating with Drupal 7 vs 8.
- I would like to know how I can leverage the new features of Drupal 8 in complicated information systems.

- I create with Drupal 7.
- I want to create with Drupal 8.
- I want to know the differences between creating with Drupal 7 vs 8.
- I would like to know how I can leverage the new features of Drupal 8 in complicated information systems.

Lets go from:



To:



Custom LMS Architecture

Learning Management System

An LMS is software that allows for the organization and delivery of courses or training programs.

- LMS Architecture
- Need to deal with some prerequisite information so we all understand the goals of the project that I used as my test bed.
- What is an LMS? An LMS is a Learning Management System.
- It is software that allows for the organization and delivery of courses or training programs. If you have used buildamodule.com or drupalize.me then you have used a LMS.

Traditional LMS

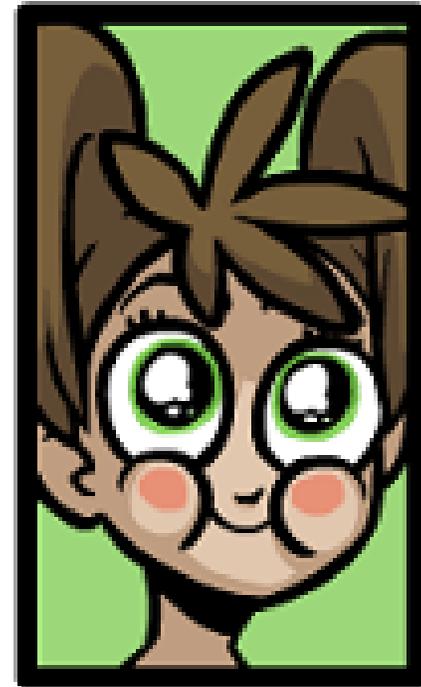
- Curriculum or Lesson plan
 - Unit or Module
 - Lesson/Project/Assessment/Mini-lessons

- Traditional LMS is based on a traditional course or lesson plan
 - Curriculum or Lesson plan
 - Collection of Units or Modules
 - Collection of Lesson/Project/Assessment/Mini-lessons

Our custom LMS is structured

- Module -> Menu
 - Segments -> Nodes or Content Type
 - Components -> Paragraphs Items

- Our custom LMS is structured as
 - Module -> Menu
 - Segments -> Nodes
 - Components -> Paragraphs Items
 - Lesson plan equivalents are built with select fields on the segment nodes.



- Basic Drupal Architecture
- In order to make sure we are all on the same page moving forward I need to talk about some basic Drupal Architecture. If this is review for some of you, I am sorry for the unskippable tutorial level.

Basic Drupal Content Architecture

- Entities
- Fields
- Variables

- Basic Drupal Architecture
- In order to make sure we are all on the same page moving forward I need to talk about some basic Drupal Architecture. If this is review for some of you, I am sorry for the unskippable tutorial level.
- Don't worry I will mostly just be talking about Entities, Fields, and Variables.

Entities

Entities are things in Drupal; things do not have to be nodes.

- Entities are things in Drupal, starting in drupal 7 we got content entities.
- In Drupal 6 custom things where typically nodes or totally custom
- In Drupal 7 the entity-field system brought the CCK into core without making everything a node
- Drupal 8 has a fully fledged entity api which is derived from the capabilities of the contrib entity api module for drupal 7.
- Users, Taxonomy Terms, and Nodes, are all Entities

Bundles and Fields

- A bundle is a collection of fields instances on an Entity
- A field instance is a place for complex data in a Entity Bundle
- An Entity can have Bundles to allow for multiple collections of fields instances on a single type of entity.
- A property is simple storage for Entity level data
 - Field instances are unique to that bundle.
 - An Entity has properties which span all bundles.
 - created date
 - node status
 - who created are all properties
 - A field instance is a place for complex data in a Entity Bundle
 - A property is simple storage for Entity level data

Properties

Properties are linked to an Entity, such as Node. Fields are linked to the Bundle, such as Content Type.

Properties are stored on the entity's table and fields are stored in their own table.

- Properties are linked to an Entity, such as Node. Fields are linked to the Bundle, such as Content Type.
- Properties are stored on the entity's table and fields are stored in their own table.

Variables

Variables are what Drupal 7 provides for configuration --when it isn't a custom table.

- Variables are what Drupal 7 provides for configuration -- when it isn't a custom table.

LMS Features

Support Backwards Design philosophy

- Site Features
- Traditional LMS
 - Collection of Curriculum or Lesson plans
 - Collection of Unit or Module
 - Collection of Lesson/Project/Assessment/Mini-lessons
- Curriculum Designer will often use a Backwards Design philosophy
- Starts Standards/Topical they wish to teach

Goals of lesson planning

- Engage
- Explore
- Explain
- Elaborate
- Evaluate

- Using the 5 e's as the goals of lesson planning
 - Engage
 - Explore
 - Explain
 - Elaborate
 - Evaluate

LMS Lesson Goals

- Teaching
- Application
 - Thus Goals for Lesson Plan Features
- Assessment
 - Activate Prior knowledge
 - Modeling or Teaching
 - Practice
 - Application
 - Assessment
- Our custom LMS is structured as
 - module -> menu
 - segments -> nodes
 - components -> paragraphs items

Drupal 7 Modules Used

- Modules Used
- [*Frank's list of modern site building tools*](#)
- I gave a talk on modern Drupal 7 development at a meetup in Portland last year, and I am not going to go into great detail here. I wrote a blog post about it and if you would like more details I suggest you look there.
- I will go over the modules that this site used (in its core functionality), what they do.
- Later I will go over how they relate to Drupal 8.
- Remember when building a site that the output that drupal give us is always a suggestion. The important part of site building is giving the content team everything they need, you can leave it up to the developers and themers to make sure that the content gets displayed properly.

Paragraphs

Paragraphs allows us to create collections of fields. Each collection type is its own bundle and can have its own fields.

- Paragraphs -- If you are familiar with field collection then this module functions very similarly, with the main exception of the user being able to pick the bundle or paragraph type when creating content.

Entity Form

Named eForm in Drupal 8. Allows the use of the field api to create webform and have the submissions be entities.

- Entity Form -- In Drupal 8 this module was renamed to eform. The cool part of this module is that it allows us to use the field api to create webforms. Unlike the webform module, the form submissions are entities. This gives us flexibility use in views or displaying them with multiple view modes.

Eva

Solves the problem of putting views into entities as pseudo-fields.

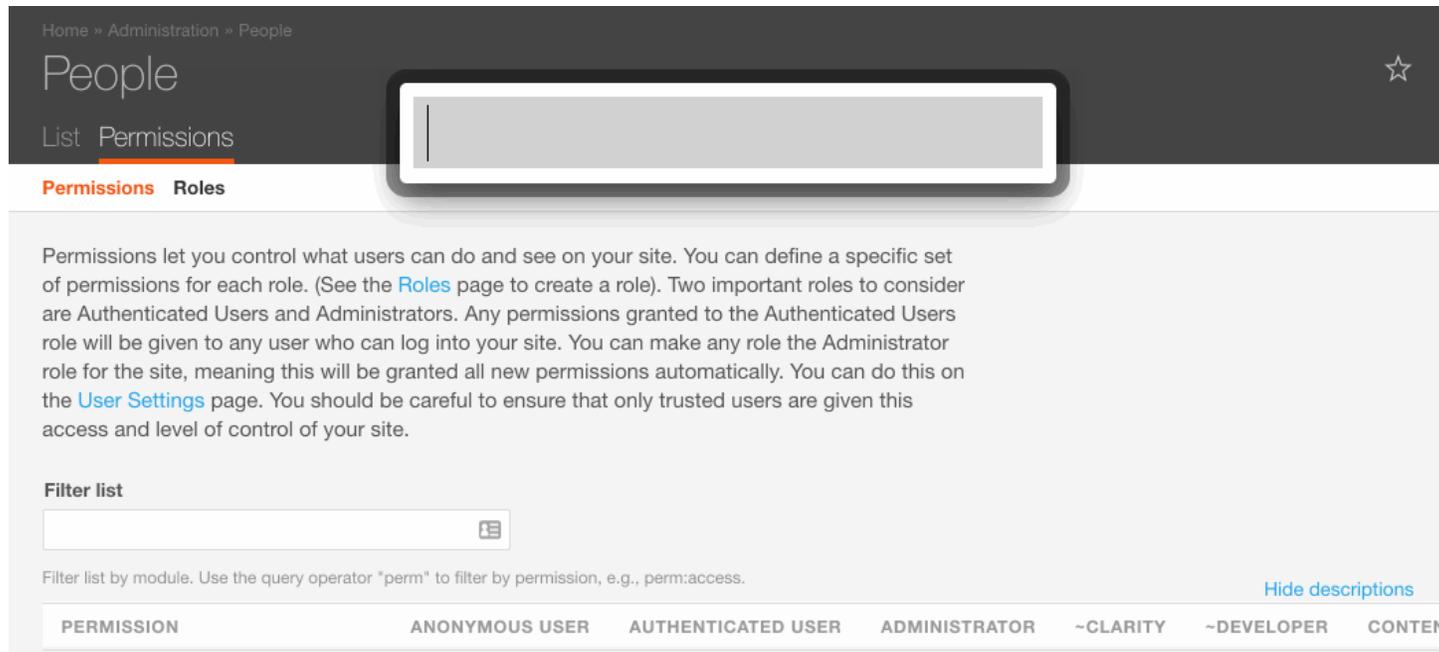
- EVA -- Entity View Attachment. Allows us to embed views into fields. Handy for things like Taxonomy term views or anywhere an ID can be used as an argument in a view.

Flag

This module allow users to flag things. Largest use is probably the "Flag this as inappropriate" functionality.

- Flags -- This module allow users to flag entities. Largest use is probably the "Flag this as inappropriate" functionality.
- The last two modules don't have much to do with this site per-se but they are fantastic modules that will really help a site-builder.

Coffee



Home » Administration » People

People

List Permissions

Permissions Roles

Permissions let you control what users can do and see on your site. You can define a specific set of permissions for each role. (See the [Roles](#) page to create a role). Two important roles to consider are Authenticated Users and Administrators. Any permissions granted to the Authenticated Users role will be given to any user who can log into your site. You can make any role the Administrator role for the site, meaning this will be granted all new permissions automatically. You can do this on the [User Settings](#) page. You should be careful to ensure that only trusted users are given this access and level of control of your site.

Filter list

Filter list by module. Use the query operator "perm" to filter by permission, e.g., perm:access. [Hide descriptions](#)

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	-CLARITY	-DEVELOPER	CONTEN
------------	----------------	--------------------	---------------	----------	------------	--------

Coffee works like Unity for Ubuntu or Spotlight search for Mac, or the Start Menu in Windows. Just start typing and a list of options will be presented. Pick the option and it will take you there.

- Coffee -- Speed up your Drupal navigation by typing where you want to go. Much like unity for Ubuntu or Spotlight for Mac or Start for Windows, this module allows us to zip around our site's admin interface with ease.

Speedboxes

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	-CLARITY	-DEVELOPER	CONTENT
Backup and Migrate						
Access Backup and Migrate Access the Backup and Migrate admin section.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Perform a backup Back up any of the available databases.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Access backup files Access and download the previously created backup files.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Delete backup files Delete the previously created backup files.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Restore the site Restore the site's database from a backup file.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
				Check all	Uncheck all	Check reverse

Check more than one box with a click and drag mechanic.

- Speedboxes -- Allows us to check multiple checkboxes at the same time. Super useful for setting up permissions.

A word on Blocks

- This is not a block heavy site
- It works mostly as an app with the blocks only really being used to place active menus on the screen.
- This is mainly because there are not many reusable pieces to the site. If it isn't a page level thing then it really isn't a part of this site.
- This site doesn't use blocks, but when I use blocks I build them with BEAN.
- Bean gives us all the functionality of building content types with nodes except with blocks instead.
- I normally add a content view for beans at /admin/content/blocks and I disable the default drupal block management stuff, with the exception of block placement because that is still necessary.
- But again, this site didn't make heavy use of blocks.





- I don't always use blocks, but when I do I use the bean module

LMS Content Strategy

Arguably, the most important part of a site.

- Content Strategy
- Now that's out of the way we can talk about the site's content strategy.

13 paragraphs types and 1 content type

Three where interactive

- Text response
- Checkbox list
- Drag and Drop

- In the end our site had 13 paragraphs item bundles for 1 content type
- Three where interactive
 - Text response
 - Checkbox list
 - Drag and Drop

Text response



Donna taking inventory.

If you were to sit and confer with Donna, what would you ask? You can record your thoughts here.

Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Aenean lacinia bibendum nulla sed consectetur. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Aenean lacinia bibendum nulla sed consectetur. Curabitur blandit tempus porttitor.

After saving your responses, you can share and then see your colleagues' work.



Checkbox list

xt?

After viewing the video, select all the statements that you believe to be true.

- 1. Most likely the teacher has already taught the students to count, and she now gives them a problem to apply what they have been taught.
- 2. The teacher sees the difference between giving a word problem and developing a context.
- 3. The number of the amounts to be inventoried and the objects chosen have been carefully selected by the teacher and she chooses which children to send to each area.
- 4. The teacher never tells the children to go to the area and count; she only describes the situation and invites the students to help her.
- 5. The teacher has used the context to hook the children and get them engaged.
- 6. The children can imagine the situation.

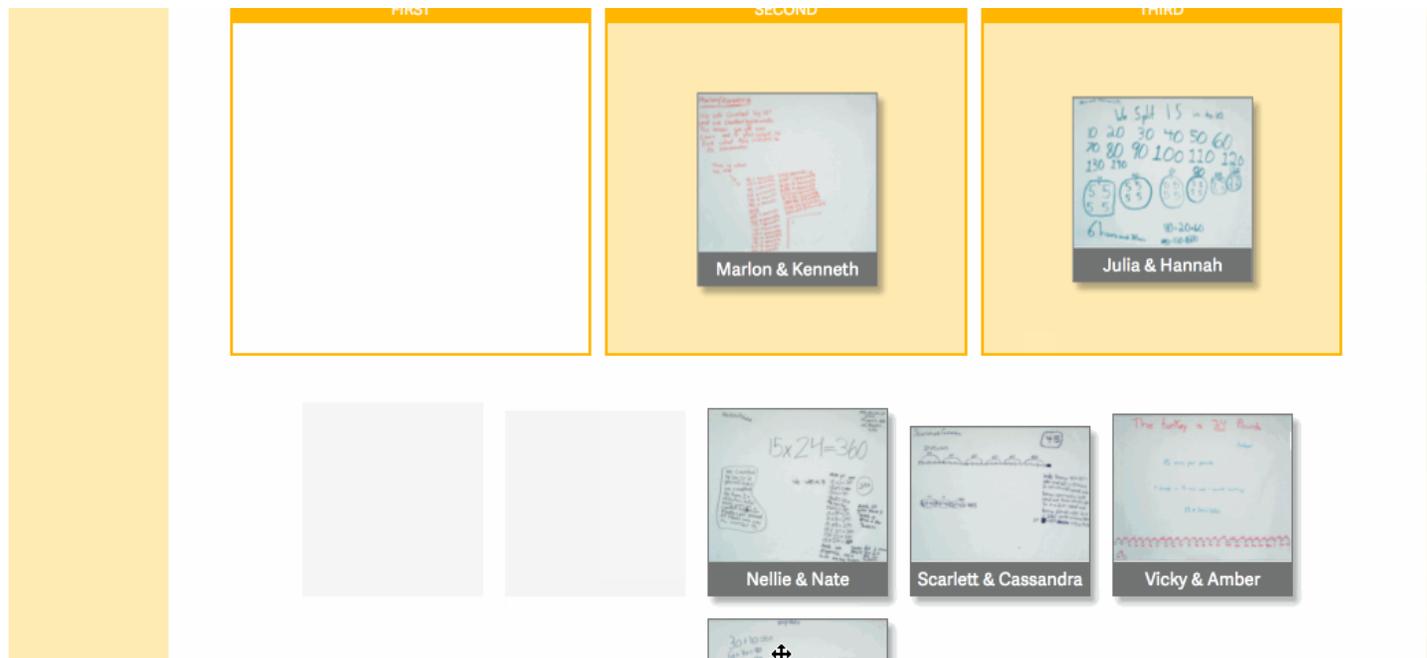
 **SAVE**

Drag and Drop

The interface displays six sticky notes on a light gray background. The top row contains three notes, each with a yellow header bar and a dark gray footer bar. The first note is labeled 'FIRST' and 'May & Rafe'. The second note is labeled 'SECOND' and 'Marlon & Kenneth'. The third note is labeled 'THIRD' and 'Julia & Hannah'. The bottom row contains three notes: 'Nellie & Nate', 'Scarlett & Cassandra', and 'Vicky & Amber'. A mouse cursor is positioned over the bottom right corner of the first note in the top row.

The rest are multimedia/content.

Everything can be *combined* to form larger whole components.



- Four were used for custom inter and intra module navigation
- The rest were videos, image, and text.
- all can be combined to form larger whole components.

Linking the EntityForm Submission to the Paragraph item

Entity	Machine Name
--------	--------------

Paragraph Type	text_response
----------------	---------------

- Linking the EntityForm Submission to the Paragraph item, each interactive paragraphs item had a corresponding entity form with an entity reference back to the paragraphs item that displayed the form. We used the paragraphs item's bundle machine name to pick the bundle of the EntityForm to display.
- We put this all together, using Drupal's ingenious render array system that doesn't care if it is displaying a form, or content, or anything themable, we were able to HULK smash the entityforms into the paragraphs item displays and save any users response and know what they were responding to and who was responding to it.

Evaluation

Sharing via Flags on the paragraphs items

Explain your thinking about your work sample selections, and your choice of sequence. Save your work, then decide if you would like to share your responses with your colleagues.

Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Aenean lacinia bibendum nulla sed consectetur. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer posuere erat a ante venenatis dapibus posuere velit aliquet. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Aenean lacinia bibendum nulla sed consectetur. Curabitur blandit tempus porttitor.

After saving your responses, you can share and then see your colleagues' work.

- We also had to allow for self evaluation. Our client wanted users to be able to share their responses and learn from the responses of others. So we used flags that let the users share their responses, in turn if they share their response then they see their peers' responses and then they can edit their old response and save a new one.

I selected Vicky and Amber's first because of the skip counting they used and discussed earlier in the unit.

I would start with Nellie and Nate to show that 15 can be two ways (15 and 15 = 30). I would then move to Scarlett and Casandra to show another way to make 30.

I would start with Vicky and Amber, then Scarlett and Amber, then May and Rafe to discuss how regrouping could be formalized.

- Our initial expectation was that we would have to build out all the content in a spreadsheet and at one point I was

building a migration to import all the content from a CSV.

Choosing the Paragraphs module gave our content editors an interface that allowed them to build the content in the site as we were adding functionality.

Custom Development

- Custom Development
- The meat of how we are putting this together is in the `hook_entity_view_alter` implementation and the `hook_form_alter` implementation.

Pass the build to custom function

```
if (!empty($build['#bundle'])) {  
  $bundle = $build['#bundle'];  
  if (function_exists("_component_{ $bundle }_alter")) {  
    // We need to pass build in an array in order to tri  
    // into passing the build by reference.  
    call_user_func("_component_{ $bundle }_alter", array(&  
  }  
}
```

- Pass the build to custom function.
- Remember we used the bundle type machine names to add the correct entity form types? We do that with `call_user_func` which calls a function based on the name we give it. We check beforehand if the function exists, this way it is extensible without having to modify too much existing code. So if we want to add another interactive element we can do so without modifying anything to do with our current interactive elements.

Text response callback

```
/**
 * Implements custom _component_TYPE_alter().
 */
function _component_text_response_alter(&$build) {
  if (isset($build[0]['#entity']) && !empty($build[0]['#entity'])) {
    module_load_include('inc', 'entityform', 'entityform.admin');
    $entity_form_name = 'text_submissions';
    $component_id = $build[0]['#entity']->item_id;
    $entity = $build[0]['#entity'];
    $form = entityform_empty_load($entity_form_name, $entity, $component_id);
    $form['field_component_submission']['#attributes']['class'] = array(
      'field-component-submission',
      'text-response-submission'
    );
    $form_rendered = drupal_render($form);
    $build[0]['#entity']->item_id = $component_id;
  }
}
```

- Text response callback
- In the case of the text response this checks for a valid callback and runs this code to include the entity form.
- Then in the form_alter we change the actual form options or the text area label to use the correct text.
- Things get complicated at this point, because ajax.
 - We have to ajaxify each of the paragraph items. I'm not going to go into code here because that would just be boring, and it is better documented elsewhere.
 - An interesting thing about saving a response is, only after it is saved can we make the paragraphs item sharable. Then after it is shared only then can we make show the shared responses. All of these things required heavy development for Drupal 7, and will likely also require lots of custom work for Drupal 8 too.

Considerations Building a LMS with Drupal 8

- Architectural differences
- UI Changes
- Module availability
- Specifically What needs to change
 - Architecture
 - Custom Development
 - Content strategy

Considerations Building with Drupal 8

- Architectural differences
- UI Changes
- Module availability
- Specifically What needs to change
 - Architecture
 - Custom Development
 - Content strategy

General Drupal 8 Considerations

- Setup the devel module
- It is now cache rebuilding
- Modules cannot be disabled
- General Drupal 8 Considerations
 - devel kint debugging DO NOT USE KUMO it doesn't work on real objects
- Enable Developer Mode
 - Beware of online sources
 - no longer cache clear now cache rebuilding
 - No more module disable, uninstall or die
 - Enable developer mode
<https://www.drupal.org/node/2598914>
 - Nice thing is that there is lots of information available already for Drupal8 the problem is that there is lots of bad information available for Drupal 8. Check your sources and do not expect everything you read to actually work. Even d.o documentation can be out of date. poll module example.

and isn't easy to export and sync and the placement of the block is configuration and is easy to sync. This can lead to embarrassing deployment gaffs. Where the placement of a block that doesn't exist is possible. Here is a good article about avoiding that.

<http://valuebound.com/resources/blog/how-to-sync-blocks-between-different-environments-a-drupal-8-website>

- Display Modes
 - Content View Modes
 - If you used the entity view modes modules or display suite to provide view modes for content types and such, then you should already be familiar with this. If you don't know what view modes are, they are ways for providing different ways to display the same content. So for a node we could have a teaser or a full view mode or a grid view mode. Basically anything that the design called for. These existed for content in Drupal 7 but they required a contrib module such as entity view modes or the display suite module to configure them without code. Drupal 8 gave a full fledged UI to work with.
 - Form View Modes
 - All that stuff I said about being able to display content in different configurable ways now also applies to forms with Drupal 8. So, as an example, let say you have a very complicated user registration form because your client wants to collect a bunch of information about the user, but none of it is required. Now you can create a new simplified form for adding users. You can do this in the UI in Drupal 8.
- New field types (entity reference, link, telephone)

Things to google (many have been backported to Drupal 7)

- Composer support
 - Things to google (many have been backported to Drupal 7)
- ◦ Composer support
- ◦ Toolbar update (Navbar in Drupal 7)
- ◦ Toolbar update (Navbar in Drupal 7)
- ◦ Responsive Image Styles (Picture Module in Drupal 7)
- ◦ Responsive Image Styles (Picture Module in Drupal 7)
- ◦ File Entities
- ◦ RESTful api
- ◦ File Entities
- ◦ Experimental Core Modules
- ◦ RESTful api
 - Content Moderation
- ◦ Workflows
- ◦ Experimental Core Modules
 - Field Layout
- ◦ Content Moderation
 - Migrate Drupal UI
- ◦ etc
 - Place Blocks <- outside in
- Switching to semver should allow for smaller backward compatible changes.
- Testing infrastructure should ensure backward compatible changes are backward compatible. The DA infrastructure team has made this a reality..

UI Changes

Good News! Not much has changed significantly in the UI. I guess that could be bad news too.

- This is going to be brief, mainly because the UI changes are minimal.

Operations

NAME	DESCRIPTION	OPERATIONS
Article	Use <i>articles</i> for time-sensitive content like news, press releases or blog posts.	Manage fields
Basic page	Use <i>basic pages</i> for your static content, such as an 'About us' page.	Manage fields
Segment		Manage fields Manage form display Manage display Edit Devel Delete

New Drop down thingy for selecting what we are doing to a thing.

- One change is Operations now go in a drop down with the maybe most common one first? Really I don't know how that was chosen so if someone here does know please tell me afterward.

Form Display Settings

Multiple entity form settings on the Manage form display page, this includes field widget settings.

Manage form display ☆

Edit Manage fields Manage form display Manage display Devel

Default suggestions

Home » Administration » Structure » Content types » Page

Content items can be edited using different form modes. Here, you can define which fields are shown and hidden when *Basic page* content is edited in each form mode, and define how the field form widgets are displayed in each form mode.

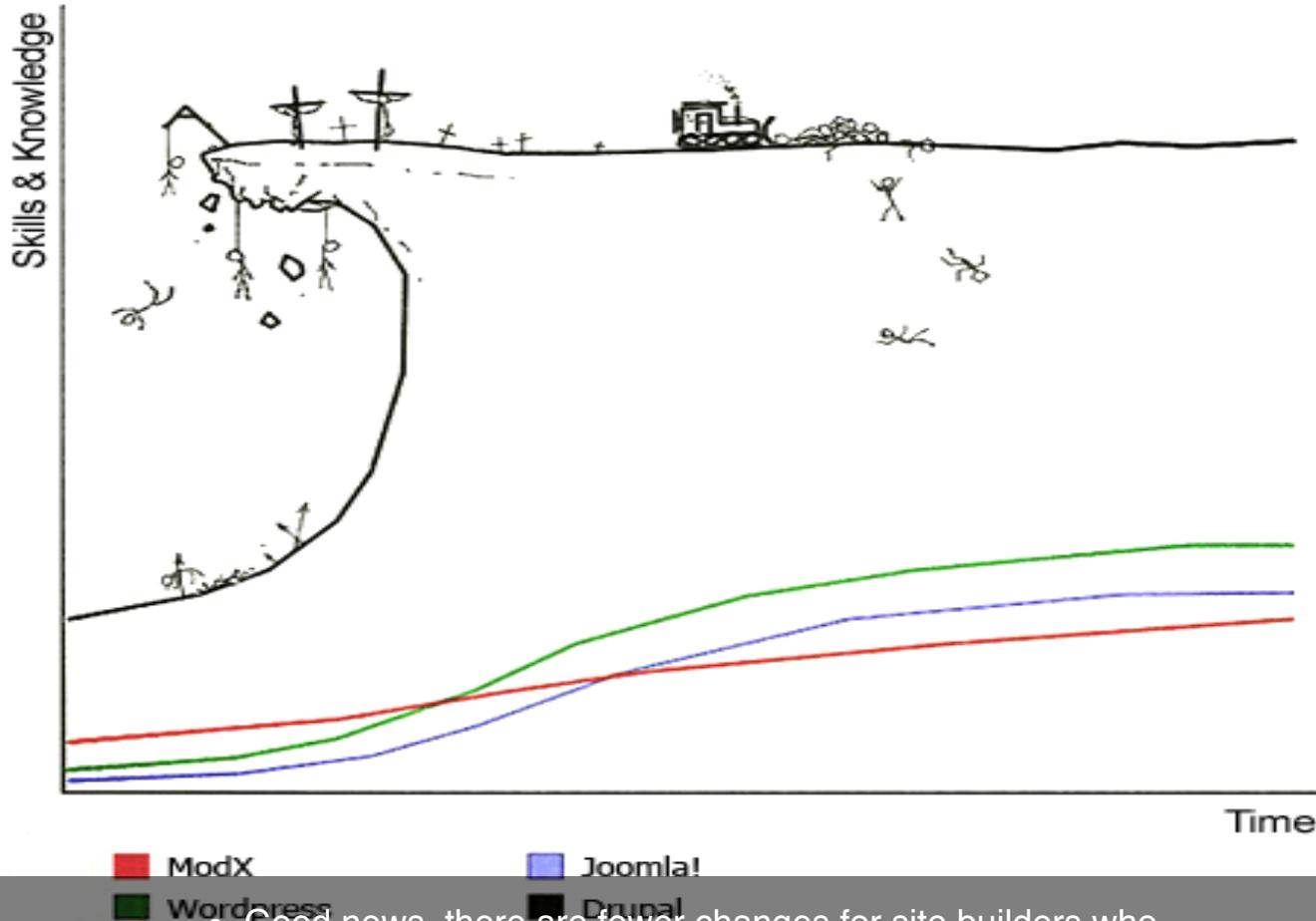
Show row weight

FIELD	WIDGET
✚ Title	Textfield <input type="text" value="Textfield"/> Textfield size: 60 ⚙
✚ Authored by	Autocomplete <input type="text" value="Autocomplete"/> Autocomplete matching: Contains Textfield size: 60 Textfield size: 60 ⚙
✚ Authored on	Date, time, and timestamp <input type="text" value="Date, time, and timestamp"/> ⚙

- Another change you will find between D7 and D8 I have already hinted at with the Form View. If you have an alternate node form then you will find the option for editing that form view in "Manage Form Display" section. This also means that controls for the form widgets are configured on the form view display settings and not the "Manage Fields" page. Manage Fields is now entirely about the field and its storage.

Good news about learning curves

Learning curve for popular CMS



- Good news, there are fewer changes for site builders who are coming from Drupal 7 to worry about. Bad news is for someone coming from not Drupal the perceived Drupal learning curve could be intimidating.

Module Availability



- Module Availability
- Lets acknowledge the elephant in the room.

module works well enough for this experiment, but if this where a full client project we would have to explore some options.

- Find an alternative
 - Custom Entities are an option
 - Comment module
 - Contact module extras
 - Relation module
- **Big problem**
 - EntityForm is not supported on Drupal 8
 - Didn't bring this module up in the modules used because we didn't use it, but this is a really cool module. This module has the ability to supplant the functionality of flag and organic groups. It adds the concept of entity endpoints which allow us to make fieldable relations between entities. In the example of this LMS that entity would have all the fields for the response, an endpoint for the paragraphs item (renamed paragraph revision for Drupal 8), and an endpoint for the user.
 - Comment module
 - Contact module extras
 - Relation module
 - I have used this module for saving user state in js apps and for keeping track of users completion status for resources in custom Professional Development portals.
 - Fund the development of a stable release
 - While I do encourage finding alternatives, at this state in Drupal 8 development I would encourage us to choose the final option because frankly aside from the comment module none of the above options are any more stable than eform and more importantly Drupal 8 needs more stability in contrib.

- Second problem
 - Paragraphs doesn't want me to reference paragraphs
 - It is possible, but requires use of the EXPERIMENTAL field widget.

- In this case the problem wasn't the module but my own unfamiliarity and over confidence. The answer was staring me in the face.
- Second problem
 - Paragraphs doesn't want me to reference paragraphs
 - It is possible, but requires use of the EXPERIMENTAL field widget. The main reason this is EXPERIMENTAL is that the paragraphs module is not mean to be used to reuse content, that is more of a function of blocks. I spent more time on this than I would have liked, due to some of Drupal 8's UI changes that I didn't fully understand. But it turned out to be a non-problem.
 - In this case the problem wasn't the module but my own unfamiliarity and over confidence. The answer was staring me in the face.



 Clarity Innovations • So we do have an elephant, but it is just a baby elephant.

- What needs to change?
- not much
- This site works more like a an app than a traditional site, this means we have slightly different expectations when choosing a Framework.
- Drupal was chosen for its scaleability, strong user centric or permissions based content model, high quality contrib space, and familiarity. As I said before I have been doing drupal professionally for close to 10 years, so I am very familiar with the project, community, and workflow. However, I am constantly looking into other technology and I have built things with other tech when Drupal isn't the best choice, so this isn't a statement coming from the island.
- Drupal 8 has lots of new Features but the biggest advances have been in underlying architecture (with the content/config entity system) and development workflow (with the switch to Semantic Versioning).

Architecture & Content Strategy

Mostly the same

- Same Features
- Same Modules
- Same Architecture
- Same Strategy
 - Architecturally the site remains the same
 - The site has the same features and the same modules are available to provide us with the same architecture and content strategy.
 - It won't always be like this. But if you put the work into making sure you have a good content strategy in the first place it will be less likely that you will have to completely re-architect a site when you move platforms. This is why content strategy is important.

Custom Development

Biggest Changes

- The largest change is the addition of developing a stable release for whatever we use for the response saving.

Custom Development

This is where the largest changes where made.

- Custom development
- This is where the largest changes where made, but maybe not where it was expected.

Not Much

- We still use `hook_entity_view_alter`
 - We still use `hook_form_alter`
-
- In Drupal 7 we accomplished the HULK SMASH part, where we smash the form from the entity form into the display of the paragraphs item, with a `hook_entity_view_alter` implementation coupled with a `hook_form_alter` implementation. In Drupal 8 with its Object Oriented Architecture I was able to accomplish it in the exact same way.



- "But Wait" you say, "Drupal 8 is OOP we aren't supposed to put stuff into the .module file anymore. That is just there for decoration!"
- Oh no, the joy of modifying aspects of the program with hooks is not gone in D8. Hooks are good things, and even though Drupal is want to change things with new version and add modern development practices to Drupal, Drupal also isn't going to ditch what is likely the most flexible pattern of php development just because it stepped in some OOP.

Okay, really lots

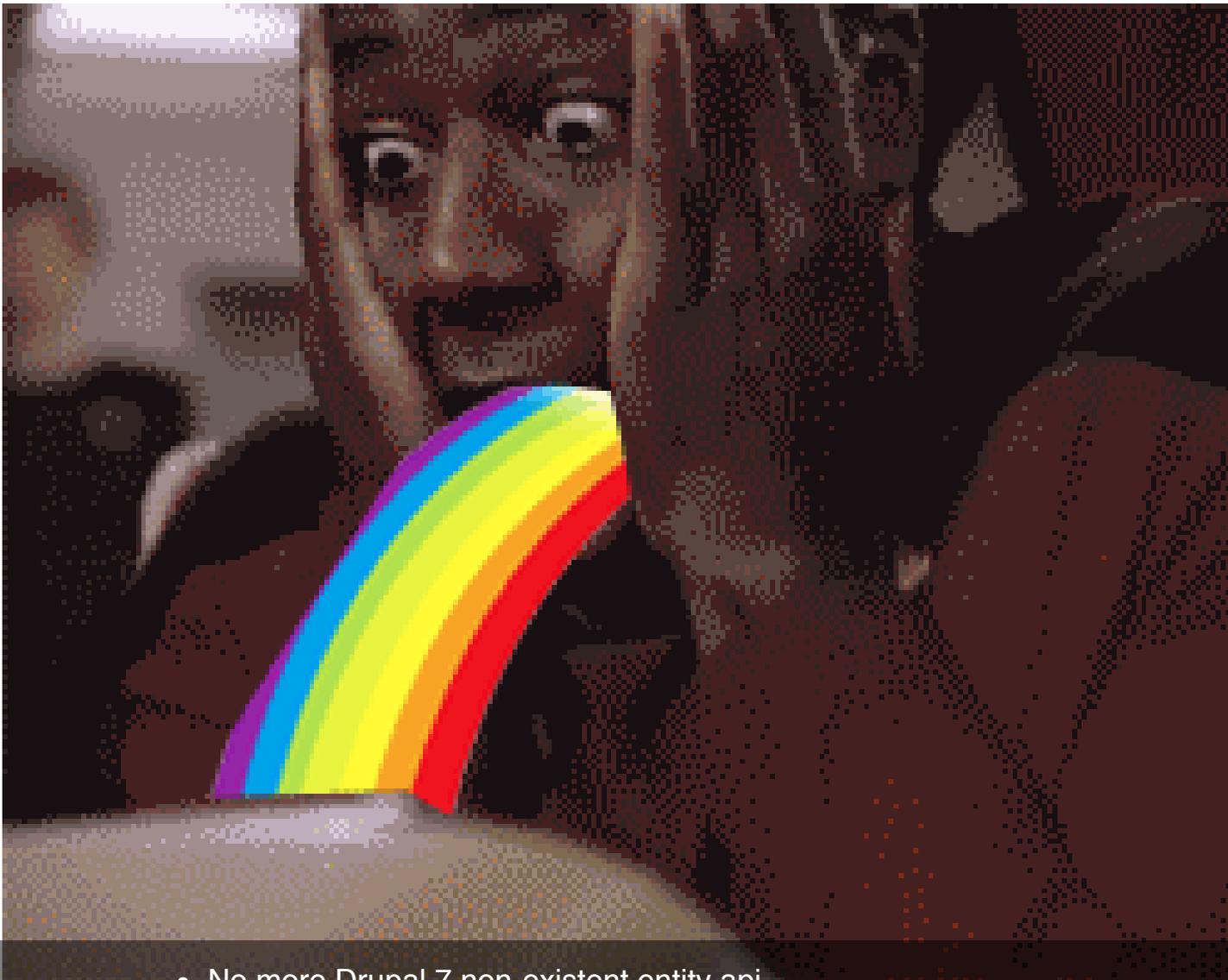
- The Object Oriented Architecture is designed to make common things easier.
- For example. But all good!
- When building this with Drupal 7, we had to reimplement parts of the form builder from the entity forms module in order to display entity forms programatically. This was likely due to the incomplete nature of the entity api in Drupal 7.
- In Drupal 8 the entity api is far more flushed out. We still use Entity Field Queries to get the EntityForm Submission for the current user and the current paragraph revision, but this time Drupal has an api for creating the empty EntityForm submission object if the user has never submitted. And it has an interface for building forms for Entities, new or old. Thus we can get a form for a new entity in the same way we get the form for the old entity, with all the right fields filled in.

Drupal 8 version:

```
// This creates a new eform submission.
$form_submission = \Drupal::entityTypeManager()
  ->getStorage('eform_submission')
  ->create(['type' => 'multiple_choice_question']);
// This loads an existing eform submission.
$submitted_eform = \Drupal::entityTypeManager()
  ->getStorage('eform_submission')->load(2);
// Either way we build the form with a call to the entity.form_builder
$form = \Drupal::service('entity.form_builder')
  ->getForm($submitted_eform);
// Then we append it to the render array.
$build[] = $form;
```

- To build a new entity, this could be a node, eform_submission, or comment, We use the Drupal::entityTypeManager class and call the create method or to load an existing one we use the same class with the load method and the entity id.
- Then we use the entity form builder service to build the form for the entity. I tested this with nodes as well. It is the same generic interface for all content entities.

- No more Drupal 7 non-existent entity api
- Drupal 8 has intelligent use of OOP that makes site building easier
- Over 100 lines of code is reduced to under 10



- No more Drupal 7 non-existent entity api
- Drupal 8 has intelligent use of OOP that makes site building easier
- Over 100 lines of code is reduced to under 10
- Mind Blown

- Why look at Drupal 8
- Custom LMS Drupal 7 architecture
- Considerations Building a with Drupal 8
 - Architectural differences
 - UI Changes
 - Module availability
 - Specifically What needs to change
- Conclusion <-- **Achievement unlocked**



Achievement unlocked
Sat through my DrupalCon Session

Don't take stuff so seriously, remember...



You are here.

When are sprints?

April 24-27: Sprint Lounge at Baltimore Convention Center

April 28: **Sprint Day** - General Sprints, Mentored Core Sprint, **First-Time Sprinter** Workshop at Baltimore Convention Center

We also provide a 24-hour Sprinter Lounge that opens on April 24th at 7:00pm and will close on Thursday, April 27th at midnight. The Sprinter Lounge will be located at the **Hilton Baltimore (401 West Pratt Street) in room Peale A-C.**

- Based on community feedback and input from the Sprint Leads, we understand the need for 'shorter sprints with greater support', and as a result will **not** be hosting Extended Weekend Sprints at this DrupalCon.
- When are sprints
 - April 24-27: Sprint Lounge at Baltimore Convention Center
 - April 28: Sprint Day - General Sprints, Mentored Core Sprint, First-Time Sprinter Workshop at Baltimore Convention Center
 - We also provide a 24-hour Sprinter Lounge that opens on April 24th at 7:00pm and will close on Thursday, April 27th at midnight. The Sprinter Lounge will be located at the Hilton Baltimore (401 West Pratt Street) in room Peale A-C.

Don't forget to give me feedback

node/17242