Cornell DrupalCamp 2017

October 19-20, 2017

Cornell University Ithaca, NY

camp.drupal.cornell.edu

For Starters:
Creating CU Bear, a Drupal 8 Starter Kit

Alison McCauley   Anthony Adinolfi   Nazrin Tingstrom
CIT/Custom Development Team, Cornell University

Background / Goals / Needs

- Why bother with any of this?
- What did we hope to achieve?

Background / Goals / Needs

- Why bother with any of this?
- What did we hope to achieve?

★ Save resources!
  - Lower our price tag
  - Build more Cool Stuff

Background / Goals / Needs

- Why bother with any of this?
- What did we hope to achieve?

★ Save resources!
  - Lower our price tag
  - Build more Cool Stuff
★ Maintenance!

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created…

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created...

★ Option 1: A codebase that you create an instance of and maintain a connection with -- it is your upstream -- you get your core updates AND config updates from there.

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created...

★ Option 1: A codebase that you create an instance of and maintain a connection with -- it is your upstream -- you get your core updates AND config updates from there.

★ Option 2: A snapshot starter-kit that you use to spin up a site, but then diverge from / customize / you're no longer dependent on / etc.

Fun with Drupal 8

★ Related terminology:
  ○ Drupal 7 "distributions and install profiles"

Fun with Drupal 8

★ Related terminology:
  ○ Drupal 7 "distributions and install profiles"
  ○ Drupal 8 distributions -- like Panopoly, Open Atrium, Lightning
  ○ Drupal 8 install profile (like our starter-kit)

Fun with Drupal 8

★ Related terminology:
- ○ Drupal 7 "distributions and install profiles"
- ○ Drupal 8 distributions -- like Panopoly, Open Atrium, Lightning
- ○ Drupal 8 install profile (like our starter-kit)
- ○ Drupal 8 "profile type" of composer package (behaves like distro)

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created…

★ Option 1: A codebase that you create an instance of and maintain a connection with -- it is your upstream -- you get your core updates AND config updates from there.
★ Option 2: A snapshot starter-kit that you use to spin up a site, but then diverge from / customize / you're no longer dependent on / etc.

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created...

★   Option 1: A codebase that you create an instance of and maintain a connection with -- it is your upstream -- you get your core updates AND config updates from there.
★   Option 2: A snapshot starter-kit that you use to spin up a site, but then diverge from / customize / you're no longer dependent on / etc.

Our considerations:
-   We wanted a composer-managed codebase (docroot concern, among other things)
    -   Pantheon hosted / composer limitation (also mention Pantheon "custom upstreams")

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created…

★ Option 1: A codebase that you create an instance of and maintain a connection with -- it is your upstream -- you get your core updates AND config updates from there.
★ Option 2: A snapshot starter-kit that you use to spin up a site, but then diverge from / customize / you're no longer dependent on / etc.

Our considerations:
- We build custom sites / none of our sites are identical.
  - Sites *will* diverge from the base.
- We hand over ownership of sites to customers, we don't "lock down" config.
  - Sites *will* diverge from the base.

Fun with Drupal 8

If you're building a Thing from which other Drupal 8 sites will be created...

★ Option 1: A codebase that you create an instance of and maintain a connection with -- it is your upstream -- you get your core updates AND config updates from there.

★ Option 2: A snapshot starter-kit that you use to spin up a site, but then diverge from / customize / you're no longer dependent on / etc.

What's inside!

- Config management, for initial site setup, via config_installer.
- Content types:
  - Basic page
  - Event
  - Gallery (images or videos)
  - News
  - Slideshow (i.e. "campaign slider")
  - Spotlight
  - Also: Entity browser ctype: two browser entities: files, nodes
- Paragraphs
- Taxonomy (vocabs):
  - Categories
  - Tags

What's inside!

- Basic Views (news, events, spotlights)
- Drupal search
- Localist (CU Calendar) integration for Events
- Roles: anonymous, authenticated, administrator
- Authentication and authorization via SSO (CUWebAuth/simpleSAMLphp) and AD groups. (Disabled by default.)
- Misc:
    - LinkIt
    - Basic URL alias patterns
    - Chosen + Chosen library
    - Coffee

Theme Approach (1 of 2)

- Bootstrap 3 base framework to meet customer requests and help ease of sharing.
- Bootstrap contrib Drupal base theme plus a Cornell-branded subtheme.
- Subsequent projects will be a subtheme of the Cornell-branded base.
- Cornell branding options are implemented as Theme Settings plugins.
- The Cornell base also customizes and overrides Bootstrap CSS to suit our front-end development approach and our stance on accessibility.
- Modular markup and CSS for Drupal components (News, Events, Spotlight, etc…)
  - Not Bootstrap-specific for future portability.
  - Not Drupal-specific, Views-specific, or Content Type-specific.
  - Instead, standard field types and reusable patterns so it can grow easily as new content types are identified.

Theme Approach

- Completely custom carousel and image gallery implementations.
  - Bootstrap Carousel does not meet our accessibility standards (even in v4).
  - Other popular carousels also fall short along with image gallery options.
  - No acceptable contrib solutions were found for Drupal 8.
  - Our solution's front-end: custom Javascript and CSS, responsive and accessible.
  - Back-end: "Slideshow" and "Gallery" Content Types with image fields, a Paragraph type for the carousel, and Twig templates to connect with the front-end scripts.
- *CKEditor customization still pending, but our D7 approach will help inform this.*

Demo

Backstage pass

Started with `drupal-composer/drupal-project` ...

- Upgraded Drupal core to 8.4.x (ps this was a nightmare; newer major version of symfony, broke drush integration, so much pain)
- Custom themes
  - Registered on packagist
  - Included in project via composer

Backstage pass

Started with `drupal-composer/drupal-project` …

- Upgraded Drupal core to 8.4.x (ps this was a nightmare; newer major version of symfony, broke drush integration, so much pain)
- Custom themes
    - Registered on packagist
    - Included in project via composer

Backstage pass

Started with `drupal-composer/drupal-project` …

- Upgraded Drupal core to 8.4.x (ps this was a nightmare; newer major version of symfony, broke drush integration, so much pain)
- Custom themes
  - Registered on packagist
  - Included in project via composer

Backstage pass

Started with `drupal-composer/drupal-project` ...

- Some `pantheon-systems/example-drops-8-composer` (for GH + CircleCI + Pantheon integration)
- Also, drupal-composer/drupal-scaffold is a thing -- something about composer + drupal core…
- Added config and modules and whatever from demo / dev site…
  - config_installer! https://www.drupal.org/project/config_installer

! SPEAKING OF WHICH !

Backstage pass

- A lot of the real work was figuring out what should be in the starter kit -- what ctypes, what media/file things, what views, what if any taxonomy things, what modules…
- We built a demo site, managing our dependencies with composer, and configuring All Of The Stuffs -- then exported the configuration into a config_installer install profile.

@todo

- Need automated testing of creating a project from the starter kit -- maybe using Lando to run tests "locally within circleCI."

@todo

- Need automated testing of creating a project from the starter kit -- maybe using Lando to run tests "locally within circleCI."
- Oh, and, automated tests overall :)
  - Our CircleCI integration is functional, the foundation is there for behat tests, buuuuut we don't have any actual tests…

@todo

- Need automated testing of creating a project from the starter kit -- maybe using Lando to run tests "locally within circleCI."
- Oh, and, automated tests overall :)
  - Our CircleCI integration is functional, the foundation is there for behat tests, buuuuut we don't have any actual tests…
- . . .
  - Text formats & WYSIWYG config
  - Finish Localist integration (CU Events calendar)
  - Finish SSO (CUWebAuth) -- and set up a non-admin authenticated role
  - People directory
  - TWIG TWIG TWIG

The Last Slide (??)

Important thing to consider, but doesn't have to be decided up front-- **the relationship to be between source project and child projects**.

★ In D7 terms: Do you want a (D7) install profile scenario, where the project is used to spin up a thing, and that's it, **OR**, a (D7) distro scenario, where changes from the source project come down to the child repos?

The Last Slide (??)

Important thing to consider, but doesn't have to be decided up front-- **the relationship to be between source project and child projects**.

★ In D7 terms: Do you want a (D7) install profile scenario, where the project is used to spin up a thing, and that's it, **OR**, a (D7) distro scenario, where changes from the source project come down to the child repos?

Remember: The starter kit we have is the former.

Remember: drupal-profile composer package type
★ Drupal 7 distro behavior in the Brave New composer-managed Drupal 8 World

The end!

https://github.com/CU-CommunityApps/gold-bear

^^ links to "Gold Bear" dev site and CircleCI instance

https://github.com/CU-CommunityApps/cubear-starter-kit

^^ links to theme repos are in cubear-starter-kit README

@alisonjo2786 everywhere; alison@cornell.edu

nazrin.tingstrom@cornell.edu

ama39@cornell.edu