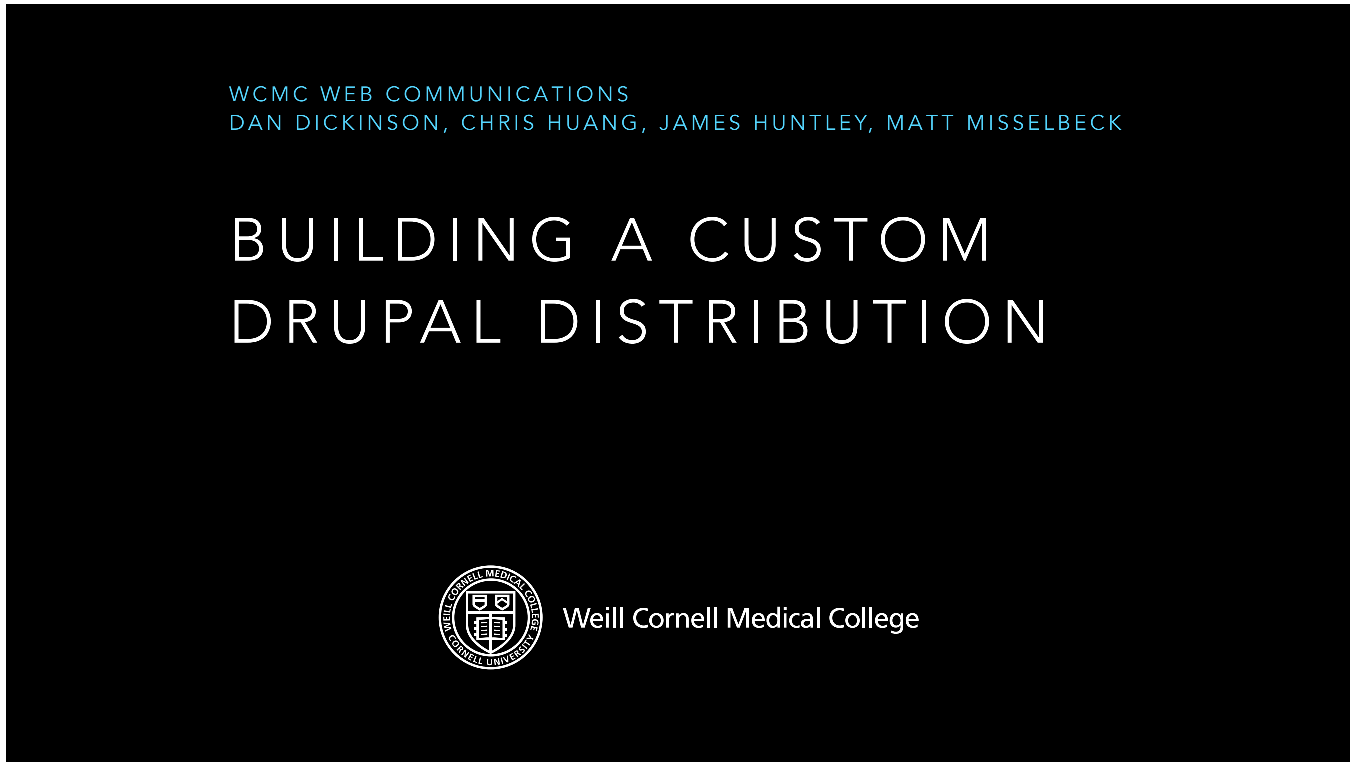
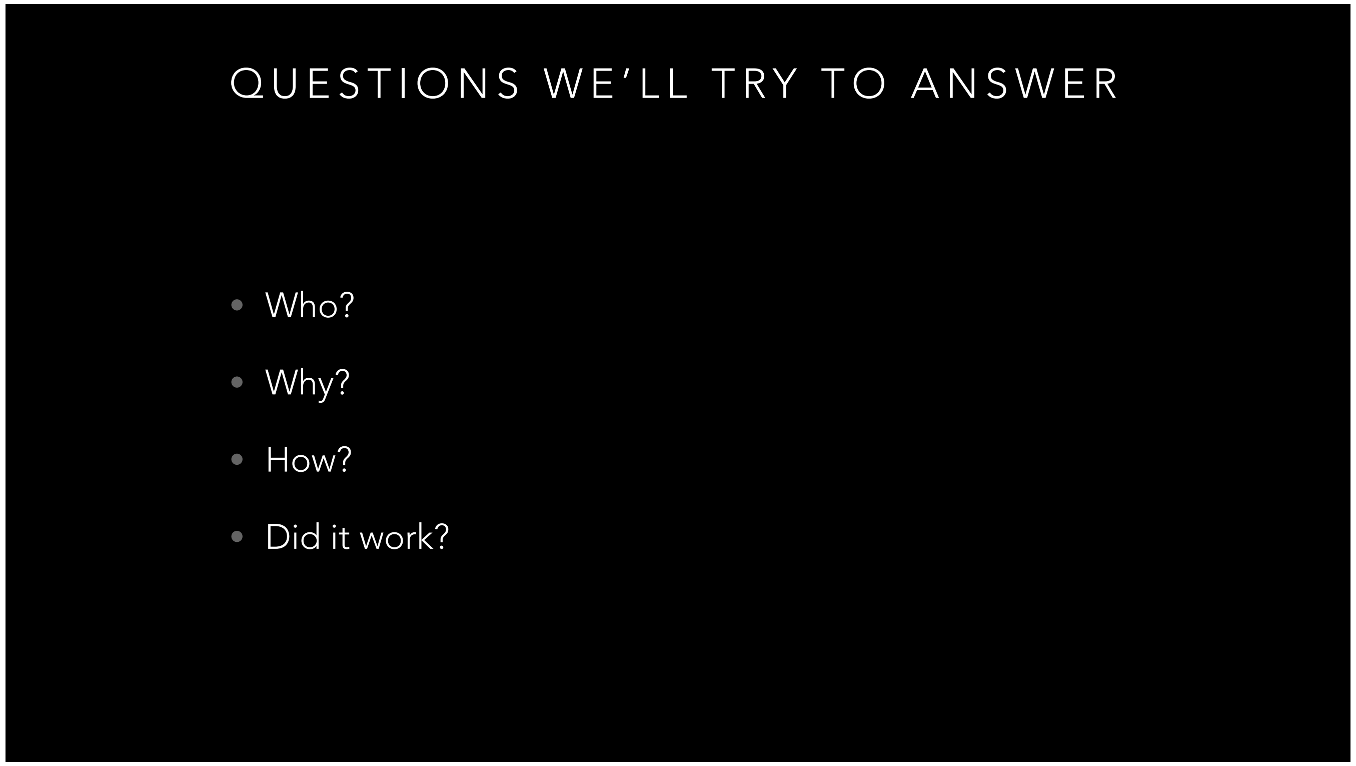


WCMC WEB COMMUNICATIONS
DAN DICKINSON, CHRIS HUANG, JAMES HUNTLEY, MATT MISSELBECK

BUILDING A CUSTOM DRUPAL DISTRIBUTION

Weill Cornell Medical College

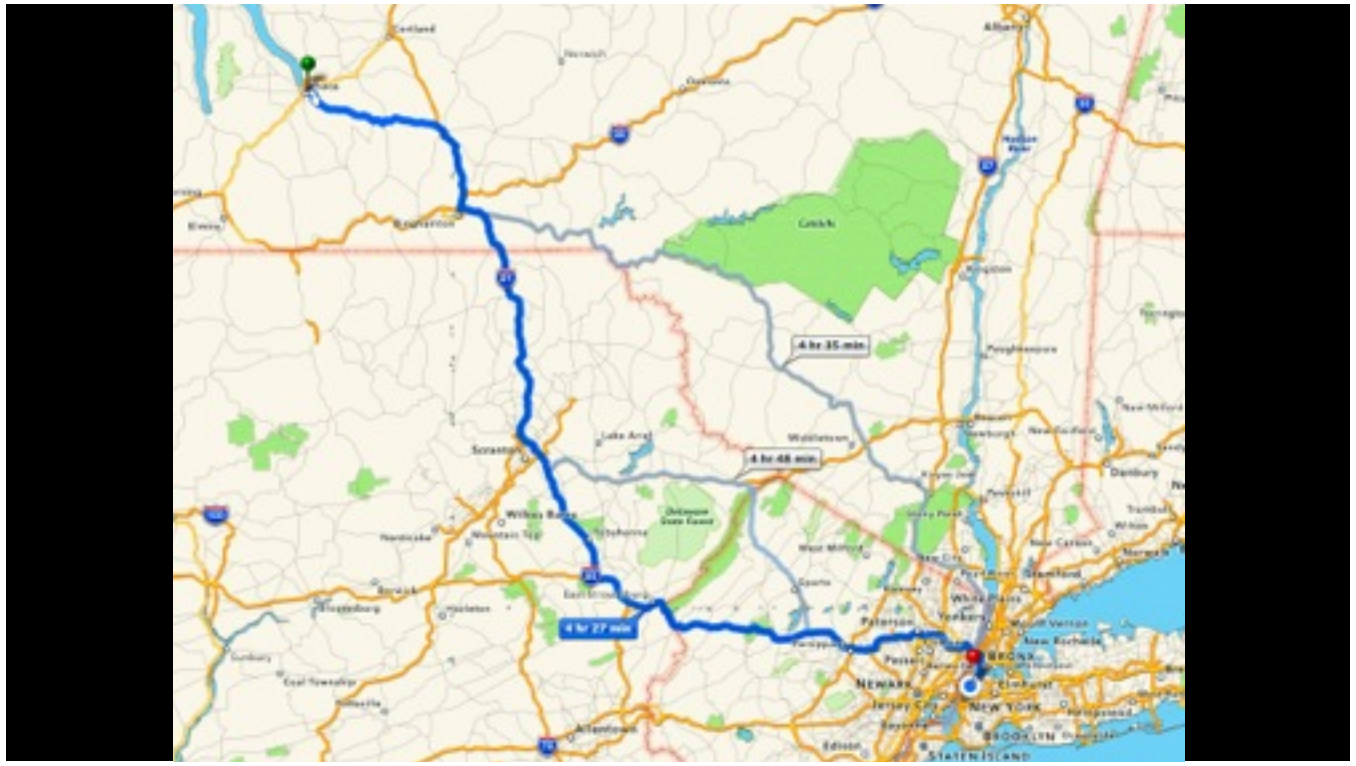


QUESTIONS WE’LL TRY TO ANSWER

- Who?
- Why?
- How?
- Did it work?

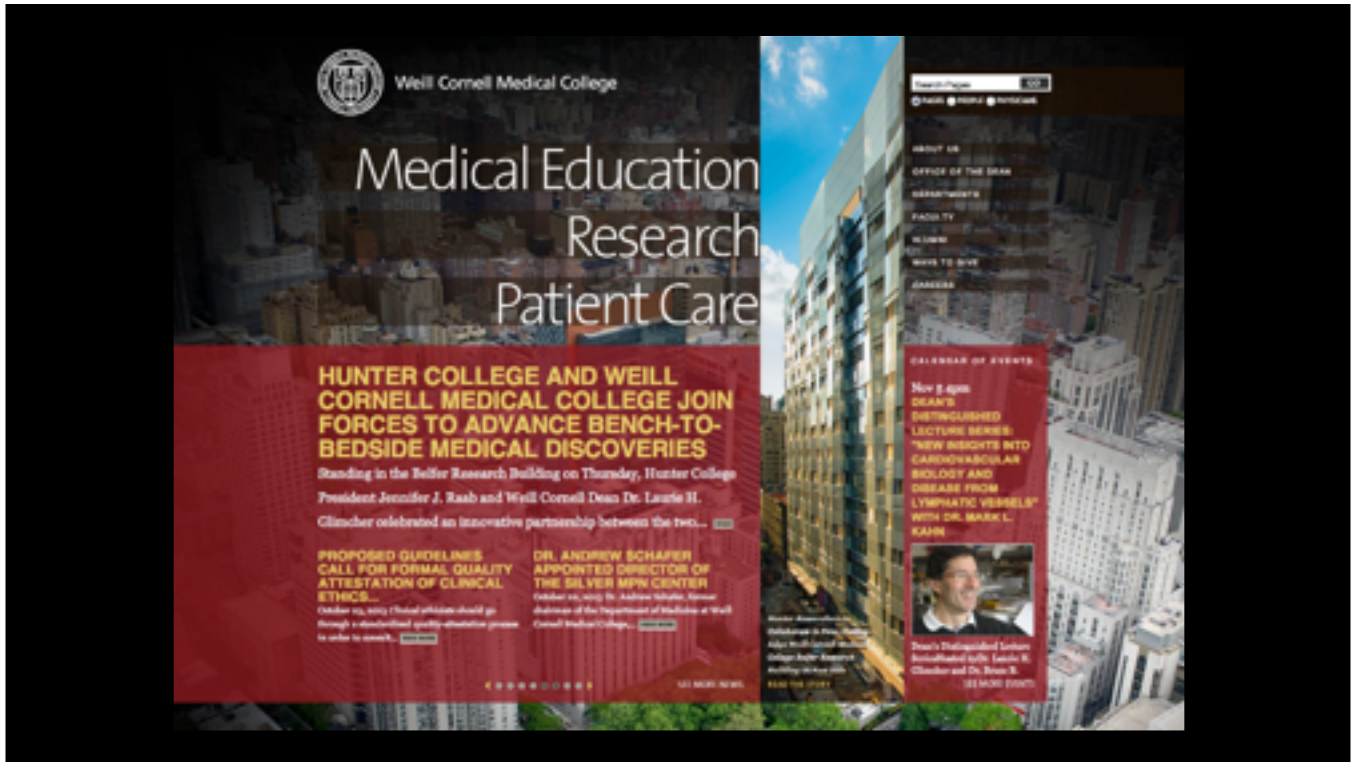
(DD) - Early. Coffee hasn’t kicked in.

- You have questions, we have answers



(DD) - Distance

- Introductions



HUNTER COLLEGE AND WEILL CORNELL MEDICAL COLLEGE JOIN FORCES TO ADVANCE BENCH-TO-BEDSIDE MEDICAL DISCOVERIES

Standing in the Belfer Research Building on Thursday, Hunter College President Jennifer J. Raab and Weill Cornell Dean Dr. Laurie H. Glimcher celebrated an innovative partnership between the two...

PROPOSED GUIDELINES CALL FOR FORMAL QUALITY ATTESTATION OF CLINICAL ETHICS...

October 23, 2013 Clinical ethicists should go through a standardized quality-attestation process in order to assess... [illegible]

DR. ANDREW SCHAFER APPOINTED DIRECTOR OF THE SILVER MPN CENTER

October 22, 2013 Dr. Andrew Schafer, former chairman of the Department of Medicine at Weill Cornell Medical College... [illegible]

Nov 5 4pm
DEAN'S DISTINGUISHED LECTURE SERIES: "NEW INSIGHTS INTO CARDIOVASCULAR BIOLOGY AND DISEASE FROM LYMPHATIC VESSELS" WITH DR. MARK L. KAHN

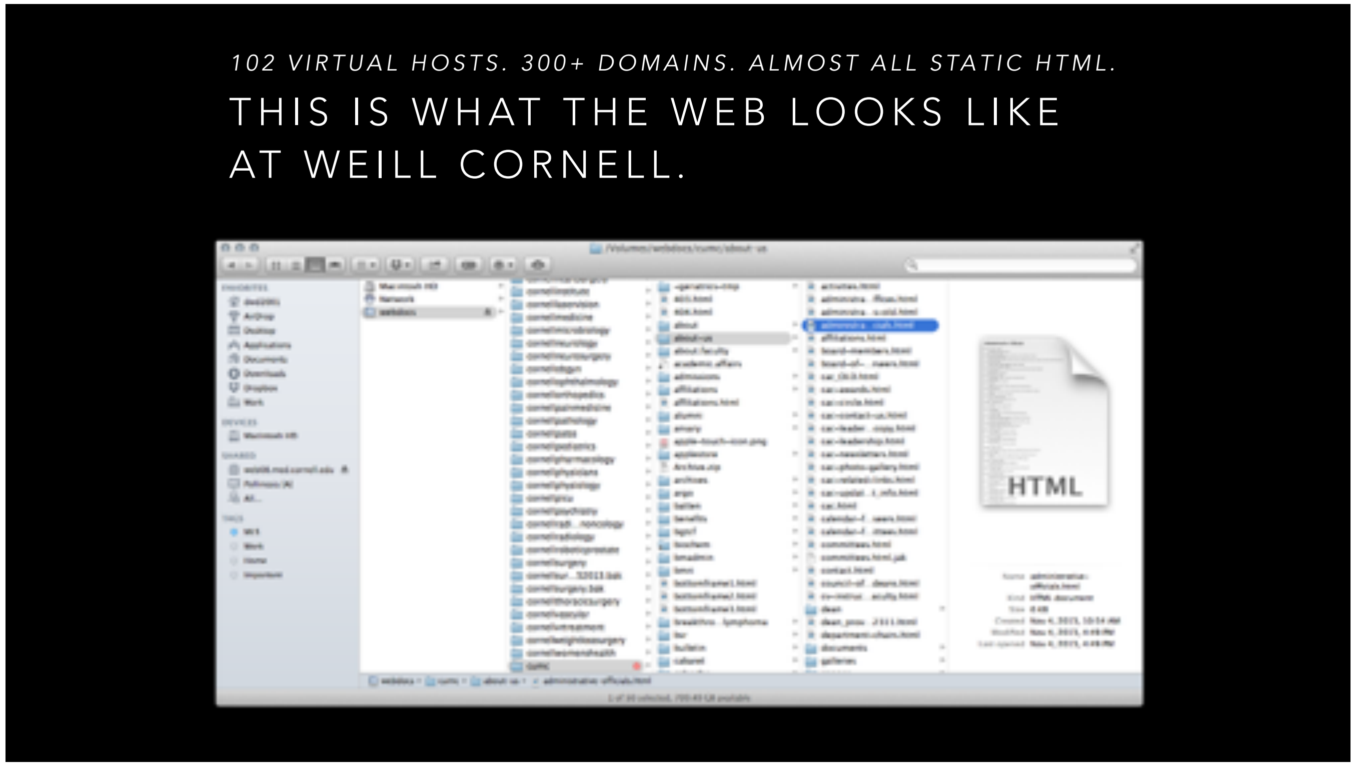
Dean's Distinguished Lecture Series Hosted by Dr. Laurie H. Glimcher and Dr. Bruce B. [illegible]

(DD) - Web Communications team, a web production shop located within the IT department at Weill Cornell.

- responsible for design and development of web sites for org units for all three missions of the college

- cost recovery

(DD) - Three stage decision process spanning over a year.

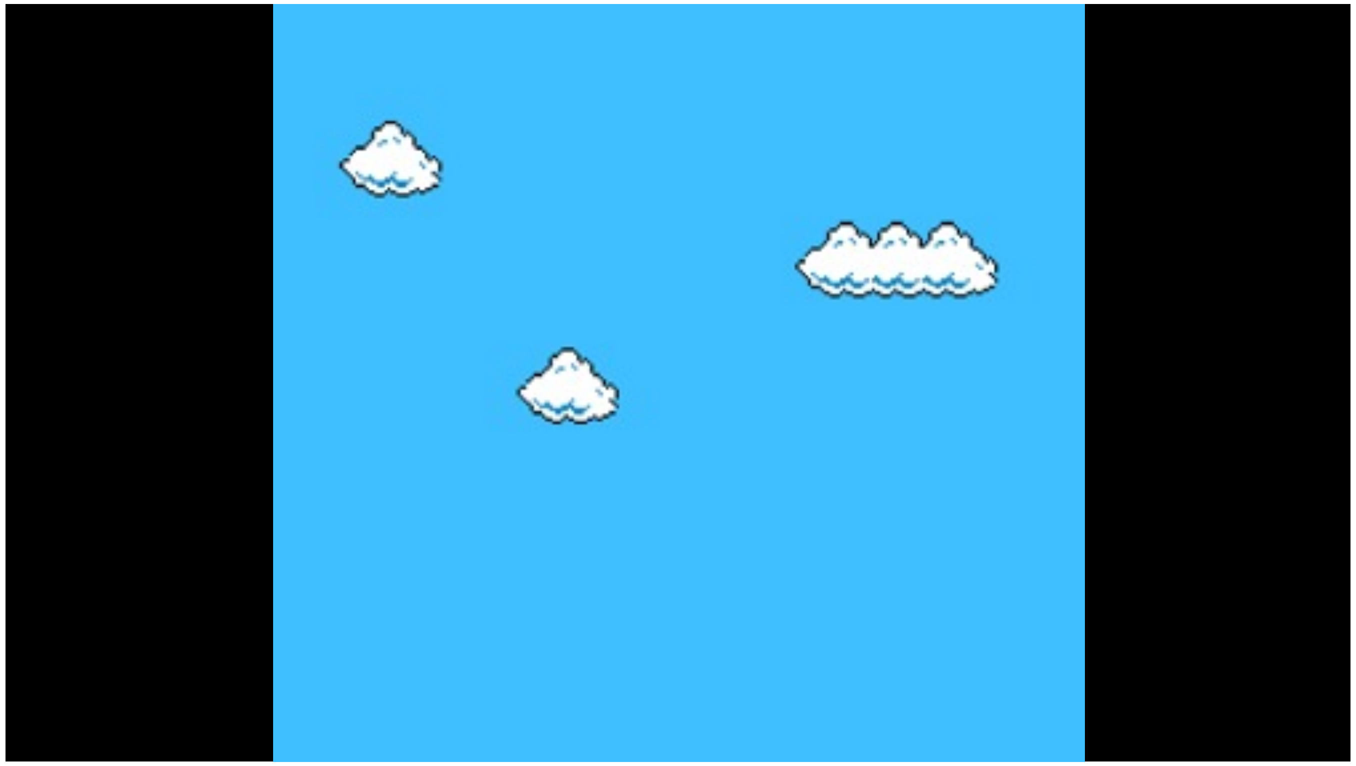


102 VIRTUAL HOSTS. 300+ DOMAINS. ALMOST ALL STATIC HTML.

THIS IS WHAT THE WEB LOOKS LIKE AT WEILL CORNELL.

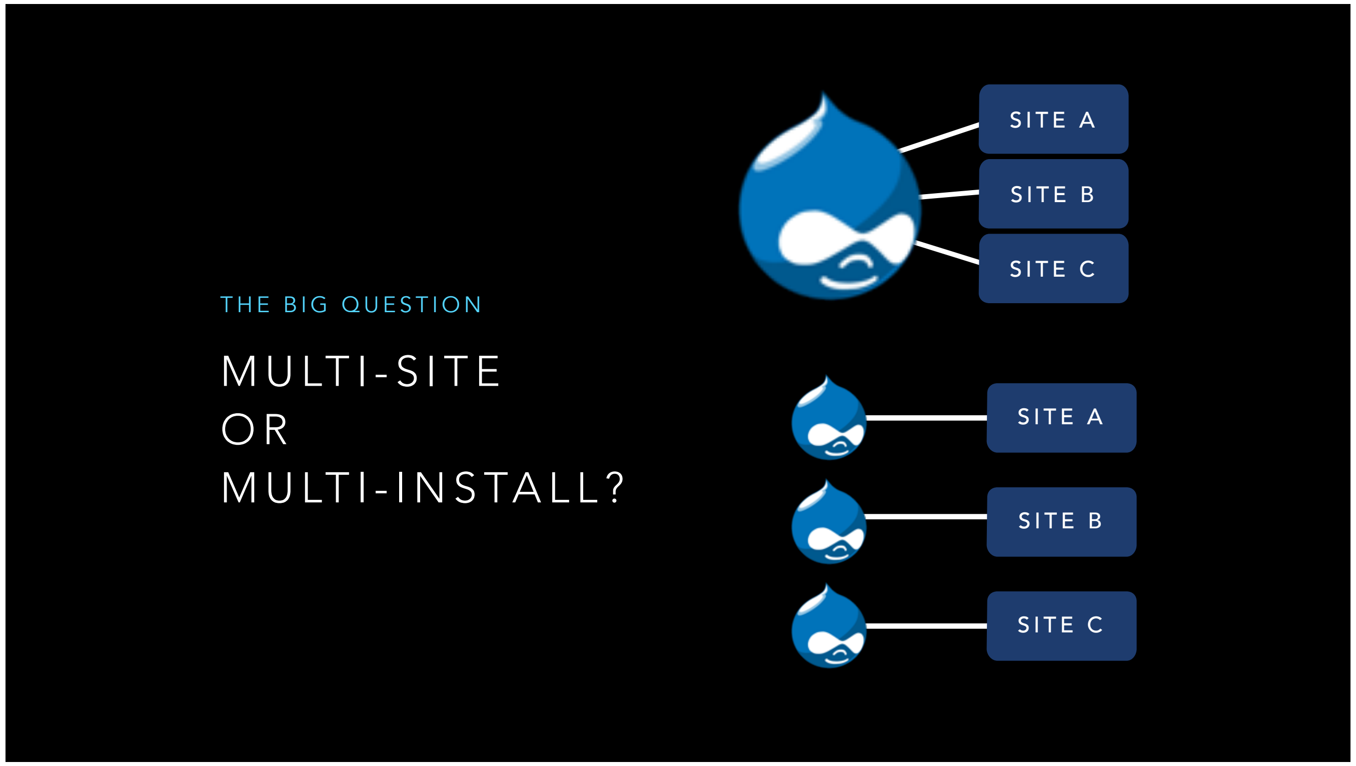
(DD) This is, sadly, a pretty recent photo of our web server. Over 100 virtual hosts chugging away on a dual-node Apache cluster. Almost entirely static HTML, with some lightweight CMS systems spitting out…static HTML. One enterprise CMS system spitting out…static HTML.

We’ve known for a while we needed to go to something more modern, and after some market research in early 2012, we settled on Drupal.



(DD)

- Hosting -> to *the cloud*.
- It’s a lot of sites, a lot of development, a lot of moving parts
- in a perfect world we’d have a whole team of Drupal-knowledgable LAMP sysadmins

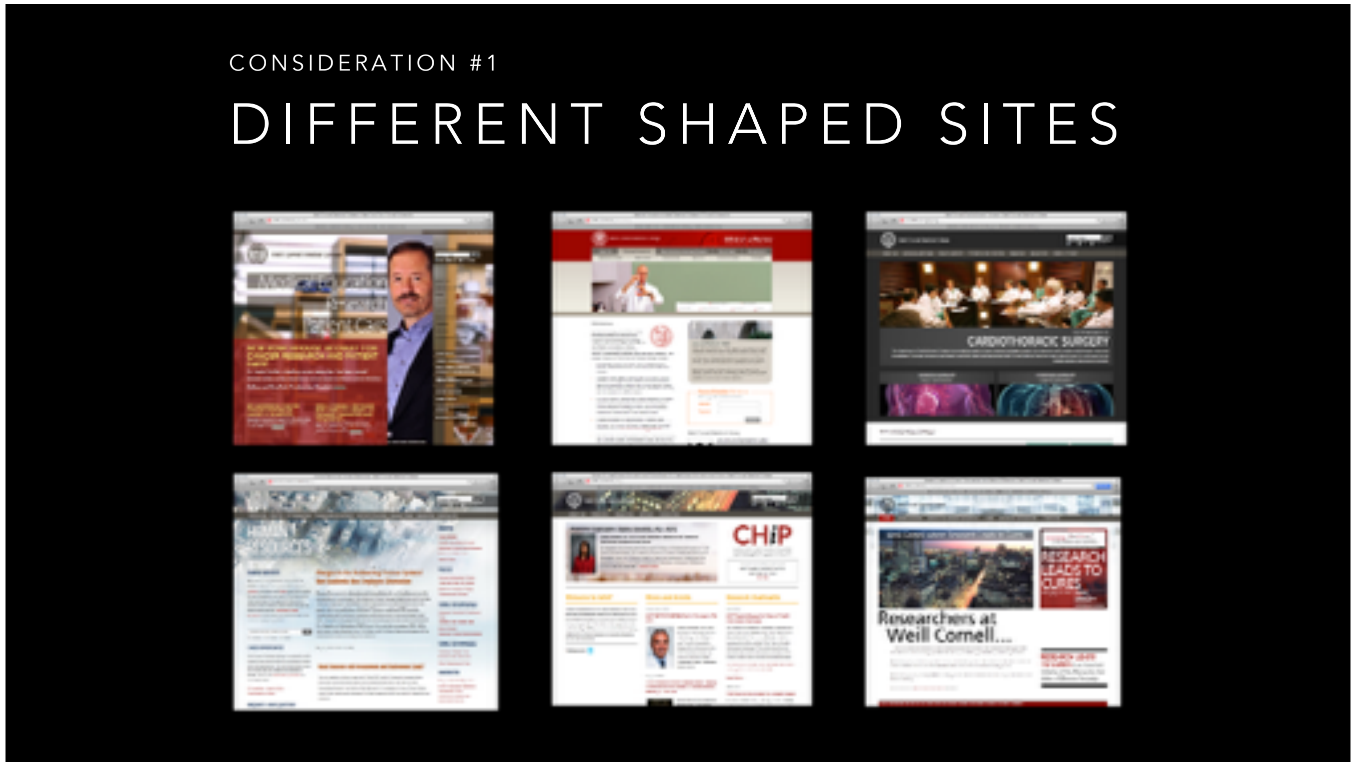


THE BIG QUESTION

MULTI-SITE
OR
MULTI-INSTALL?

(DD)

- Our vendor search took us to both Acquia and Pantheon
- A significant fork in the road: do we go with multi-site, and minimize the number of installations, or multi-install, and keep it one site to one codebase.
- needed to make sure we standardized site form and function
- something to help drive down our implementation times (and eventual cost to departments).
- spent a lot of time with this, and it came down to five key points.

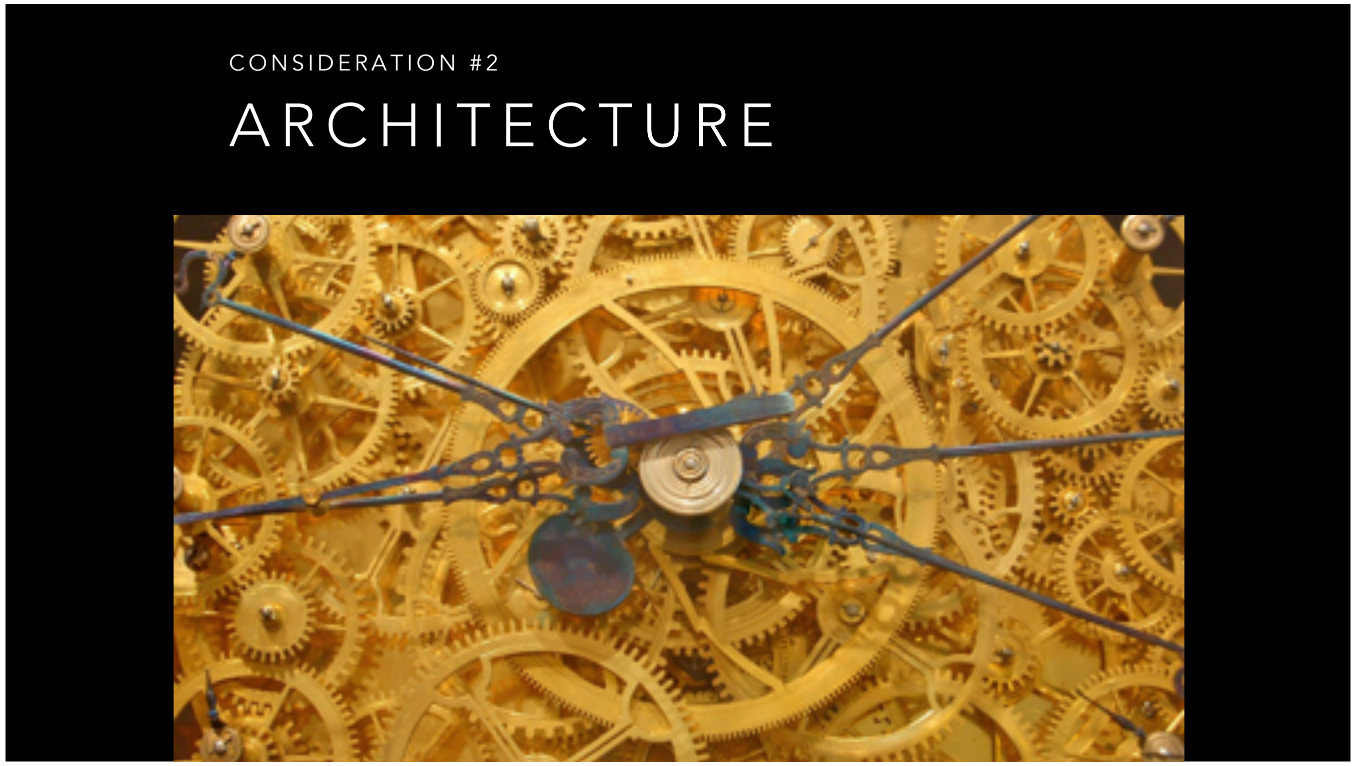


CONSIDERATION #1

DIFFERENT SHAPED SITES

(JH)

- Old model was built off of several different website models
- Built for different reasons at different times (.edu homepage different from CT surgery page; CT surgery is different than HR, HR different than Undergraduate program site, etc.)
- Differences seemed to be more in content regardless of audience/goals
- Different audiences should result in varying goals and thus alternate methods to target those goals (last time WCMC homepage was built, 21 target audiences were identified
- Some overlaps but those methods require varying functions
- The general multi-site page on drupal.org says to not use multisite if you have different functioning sites, so this was nearly enough on its own.



CONSIDERATION #2

ARCHITECTURE

(CH)

- part of the reason we wanted to go to the cloud was to avoid having to spend a lot of time on server architecture and tuning.
- multi-site: have to carve out sites in existing production installs, carefully considering resource usage and how many sites that particular piece of hardware could endure.
- different clients and different business drivers made this an area of uncertainty.

-multi-install: click a button and there’s a new site 5 minutes later. It doesn’t share resources, it doesn’t require long term resource planning (outside of a hosting tier). Pretty painless.

CONSIDERATION #3

LETTING OTHERS IN

(DD)

- We are not the only people who need to build websites for WCMC, but we are the ones setting standards.
- How do we let the anonymous (but very happy) Freelance Web Designer in to work on the site she got contracted for, while letting the Technically Savvy Research Lab build their own - while we go about our business?
- multi-site: everyone's working off the same codebase. Release, rights management, upgrading, debugging site outages all get much harder as the number of devs increase. Big increase in management overhead.

- Multi-install: everyone gets their own access, and they can't impact everyone else's site. Push down from upstream source repository when there's an update, which can be accepted by whoever's maintaining the site.

CONSIDERATION #4

UPDATING CORE

(CH) Fourth. Core.

This may seem counter-intuitive - wouldn’t updating core just a few times be easier than updating it on more, smaller installs?

It was definitely a concern, but then we considered what that upgrade process would actually look like:

Multi-site: schedule multiple sites down at once, throw entire install into maintenance mode, apply necessary updates to sites one by one, do a lot of testing.

Multi-install: schedule independently, update independently, less opportunity to break multiple sites with a single

CONSIDERATION #5

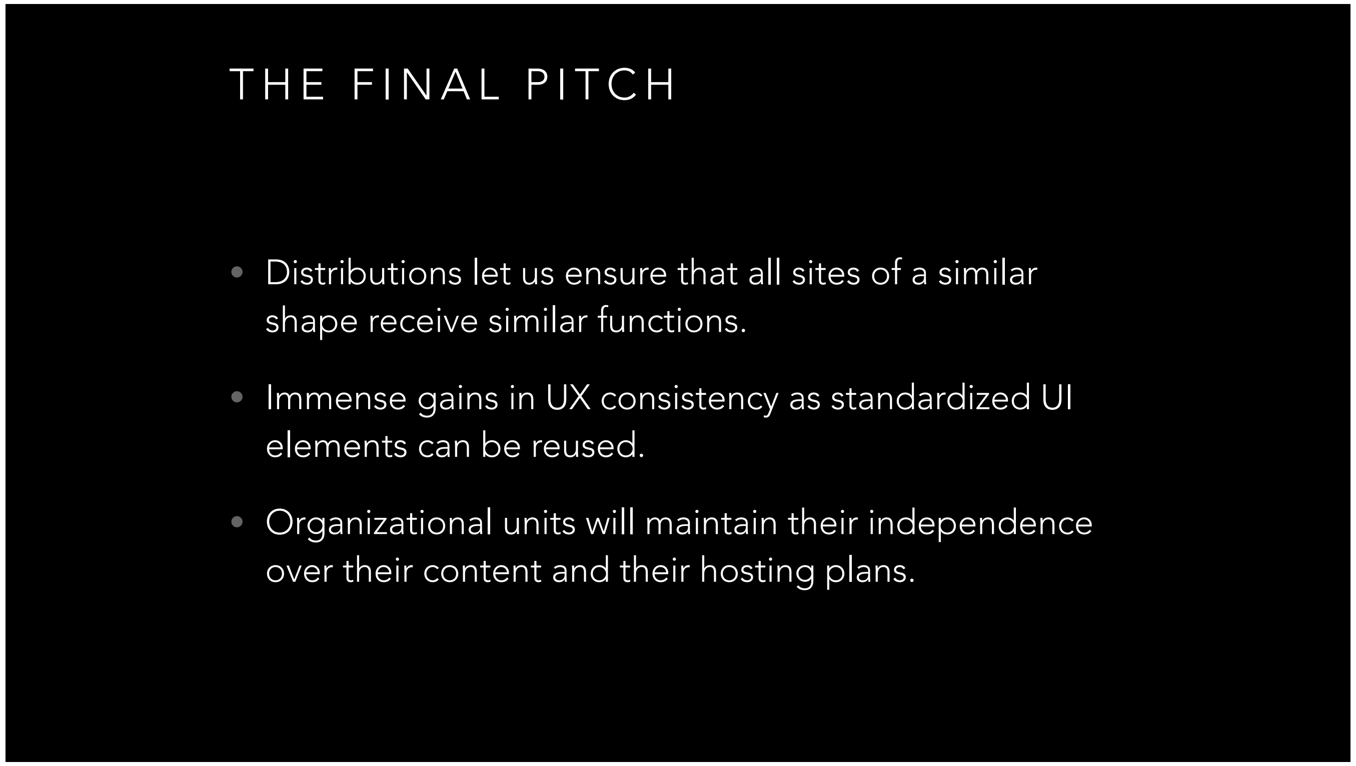
FINANCES

FEDERAL RESERVE NOTE
AB 39048518 L
B2
THIS NOTE IS LEGAL TENDER FOR ALL DEBTS, PUBLIC AND PRIVATE
Treasurer of the United States.
SERIES 1996
100
FRANKLIN

(DD) Finally, finances. We operate in a cost-recovery model - but not that I'm not saying "profit" here. We're not trying to buy yachts for the team - what's critical here was a model that makes sense for passing back to users. And this has a lot to do with how the vendors structure their offerings.

In a multi-site model, you're paying for servers with a fixed capacity. So take the cost of the server, add optional support packages - well, maybe, it depends on the site - divide by the number of sites that are on it, scale by expected traffic and empty tenant slots, multiply by the shared annual infrastructure costs and our overhead, and… ugh.

In multi-install model: you pay for a site with a fixed capacity. There's still overhead and base plan costs, but it's a very understandable unit for business folks. Here is your site, it costs X a year.

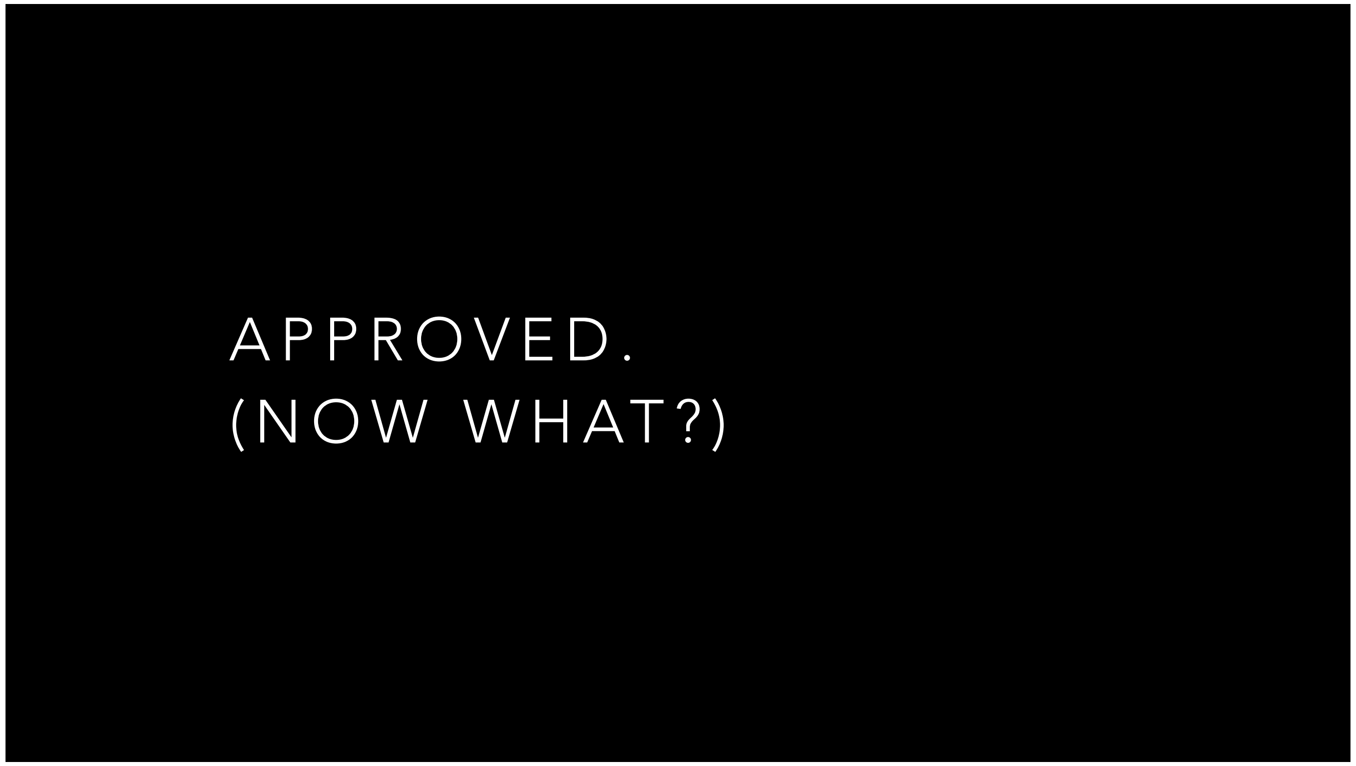


THE FINAL PITCH

- Distributions let us ensure that all sites of a similar shape receive similar functions.
- Immense gains in UX consistency as standardized UI elements can be reused.
- Organizational units will maintain their independence over their content and their hosting plans.

(DD) So all that said, we pitched multi-install instead of multi-site.

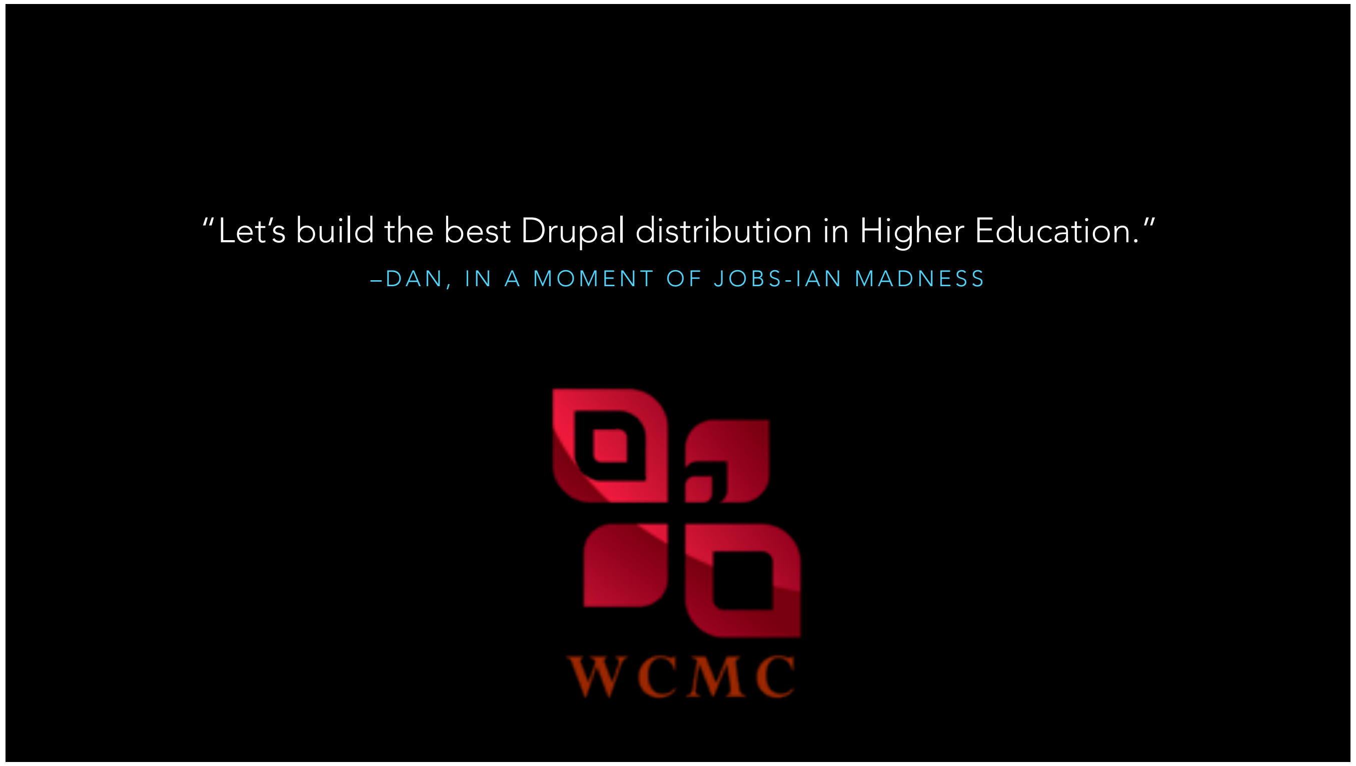
The overarching theme was that we wanted to benefit from a solid foundation while maintaining independence in the right spots. We still knew we could gain consistency and standardization that most people associate with multi-site if we spent the time on building the distribution as a product.



APPROVED.
(NOW WHAT?)

(DD) The bosses bought in.

Now came the hard part. What do we do?



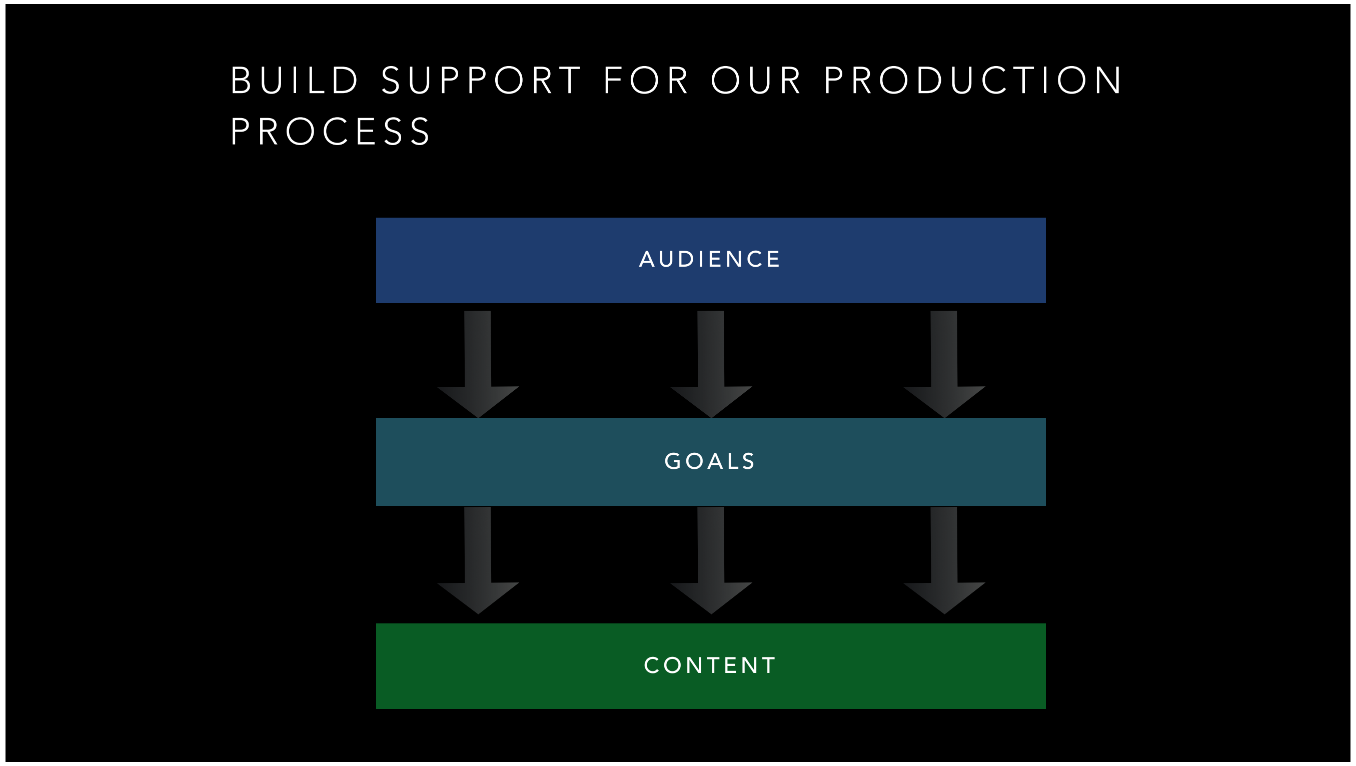
"Let's build the best Drupal distribution in Higher Education."

–DAN, IN A MOMENT OF JOBS-IAN MADNESS

WCMC

(DD) I may have gotten a little visionary and made a grandiose statement like "best Drupal distribution".

But that was our kickoff, aiming high for what we've called WCMC Drupal. And we wanted to make sure we built in a few things.



BUILD SUPPORT FOR OUR PRODUCTION PROCESS

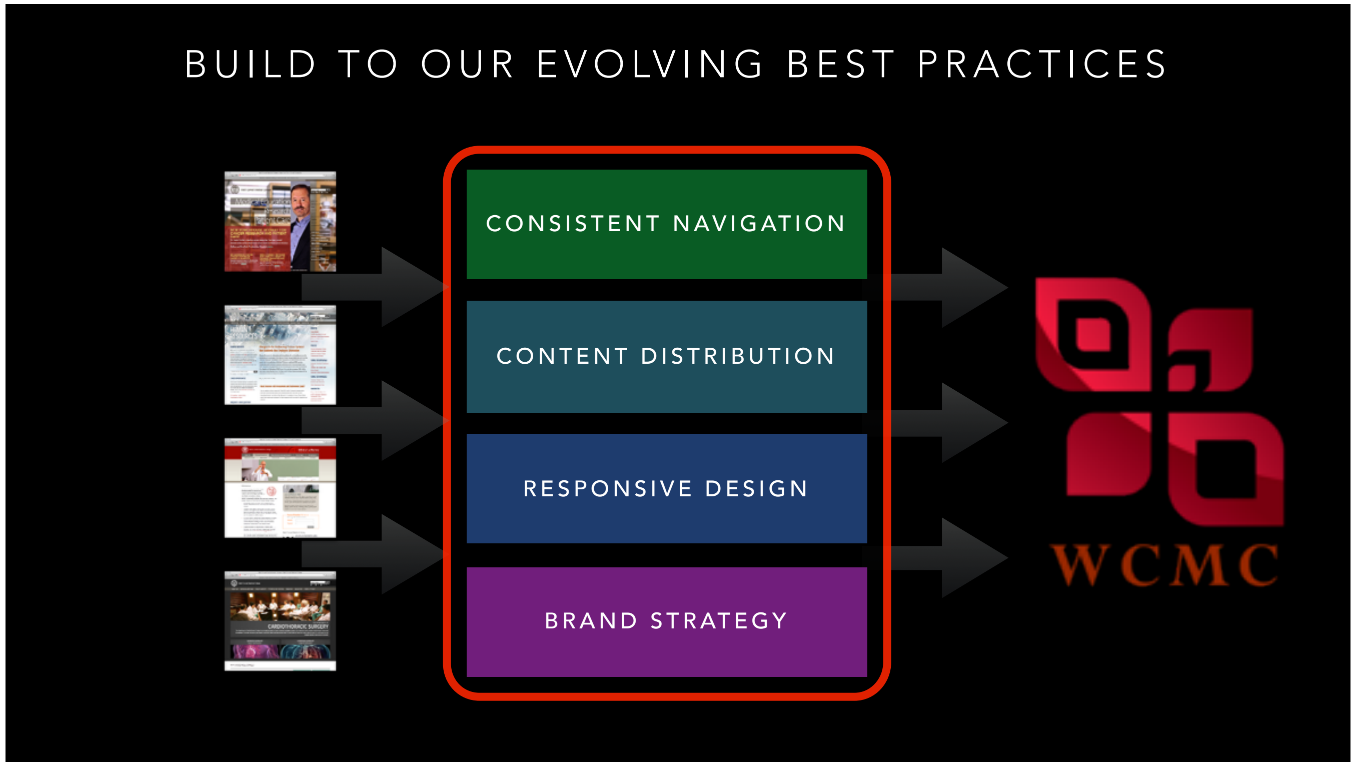
AUDIENCE

GOALS

CONTENT

(JH)

- Refocus site builds on users – Audience -> Goals -> Content (in that order)
- Tell this to every client – [broken record]
- We needed our distribution to support this
- Initial step was to have an intuitive design system for consistent and optimal UX through intuitive UI



BUILD TO OUR EVOLVING BEST PRACTICES

CONSISTENT NAVIGATION

CONTENT DISTRIBUTION

RESPONSIVE DESIGN

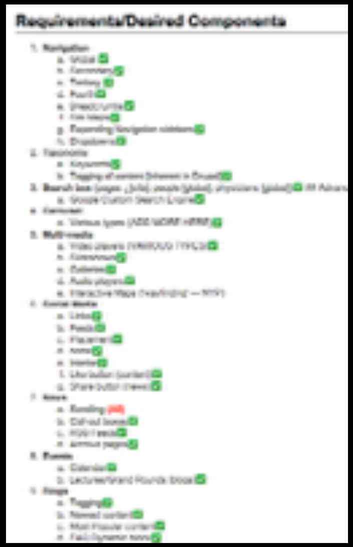
BRAND STRATEGY

WCMC

(JH)

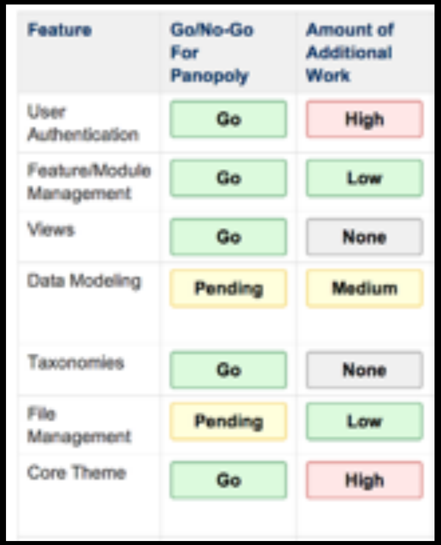
- Consistent Navigation:
  - Multiple Doorways to the site
  - Allow a user to explore (not get lost)
  - Improvement over inverted L
  - Must lend to responsive
- Content Distribution:
  - Rethink how content is positioned within website
  - Use homepage to drive users to functions (global functions)
  - How do users digest text and images
  - Types of media (are there consistencies we should estate?)

BUILD IN KEY FEATURES



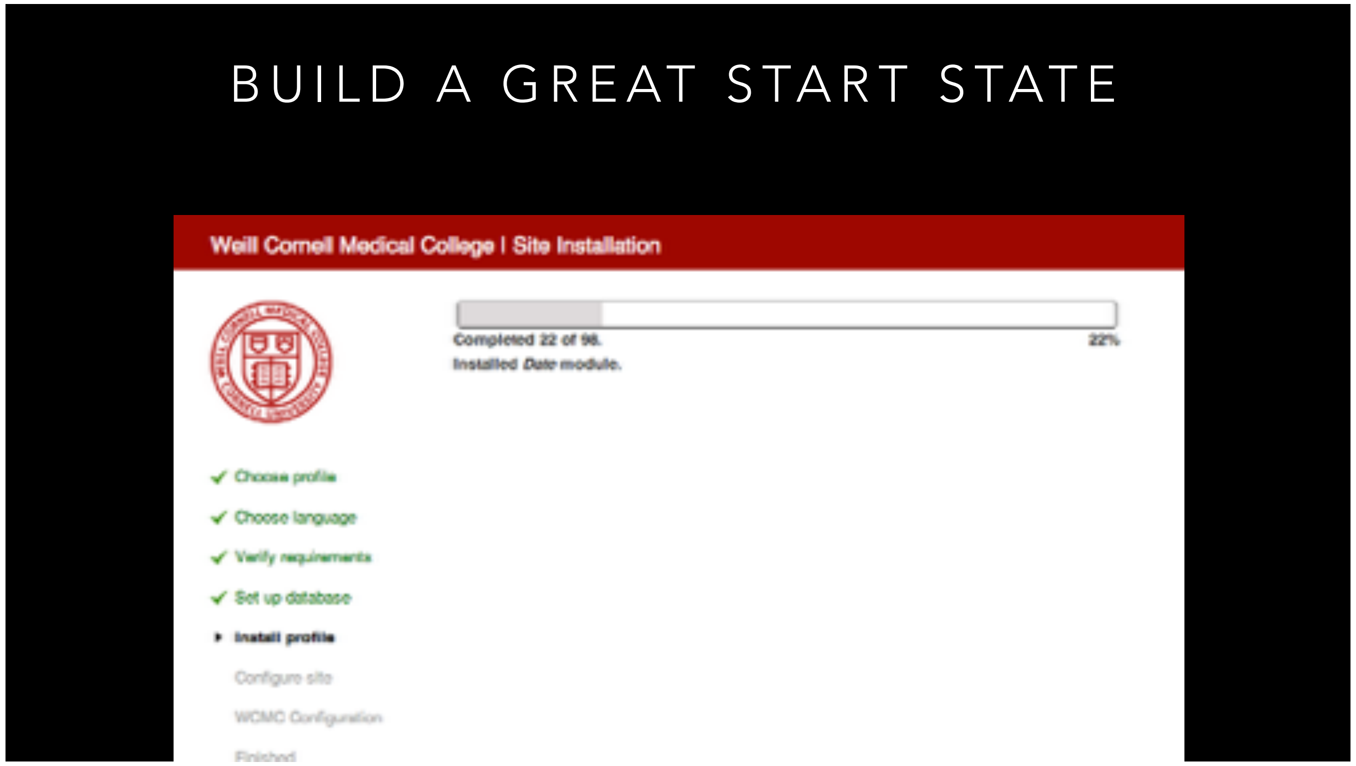
Requirements/Desired Components

1. **Navigation**
   a. Global ✓
   b. Secondary ✓
   c. Tertiary ✓
   d. Fourth ✓
   e. Breadcrumbs ✓
   f. Site Maps ✓
   g. Expanding Navigation sidebars ✓
   h. Dropdowns ✓
2. Taxonomies
   a. Keywords ✓
   b. Tagging of content (inherent in Drupal) ✓
3. **Search box** (pages, [illegible], people (global), physicians (global)) ✓ All Advan
   a. Google Custom Search Engine ✓
4. **Carousel**
   a. Various types (ADD MORE HERE) ✓
5. **Multi-media**
   a. Video players (VARIOUS TYPES) ✓
   b. Slideshows ✓
   c. Galleries ✓
   d. Audio players ✓
   e. Interactive Maps (wayfinding — ???)
6. Social Media
   a. Links ✓
   b. Feeds ✓
   c. Placement ✓
   d. Icons ✓
   e. [illegible] ✓
   f. Like button (content) ✓
   g. Share button (news) ✓
7. **News**
   a. Bundling (??)
   b. Call-out boxes ✓
   c. RSS feeds ✓
   d. Archive pages ✓
8. **Events**
   a. Calendar ✓
   b. Lecture/Grand Rounds block ✓
9. **Blogs**
   a. Tagging ✓
   b. Related content ✓
   c. Most Popular content ✓
   d. [illegible] ✓



Feature	Go/No-Go For Panopoly	Amount of Additional Work
User Authentication	Go	High
Feature/Module Management	Go	Low
Views	Go	None
Data Modeling	Pending	Medium
Taxonomies	Go	None
File Management	Pending	Low
Core Theme	Go	High

(CH)



BUILD A GREAT START STATE

Weill Cornell Medical College | Site Installation

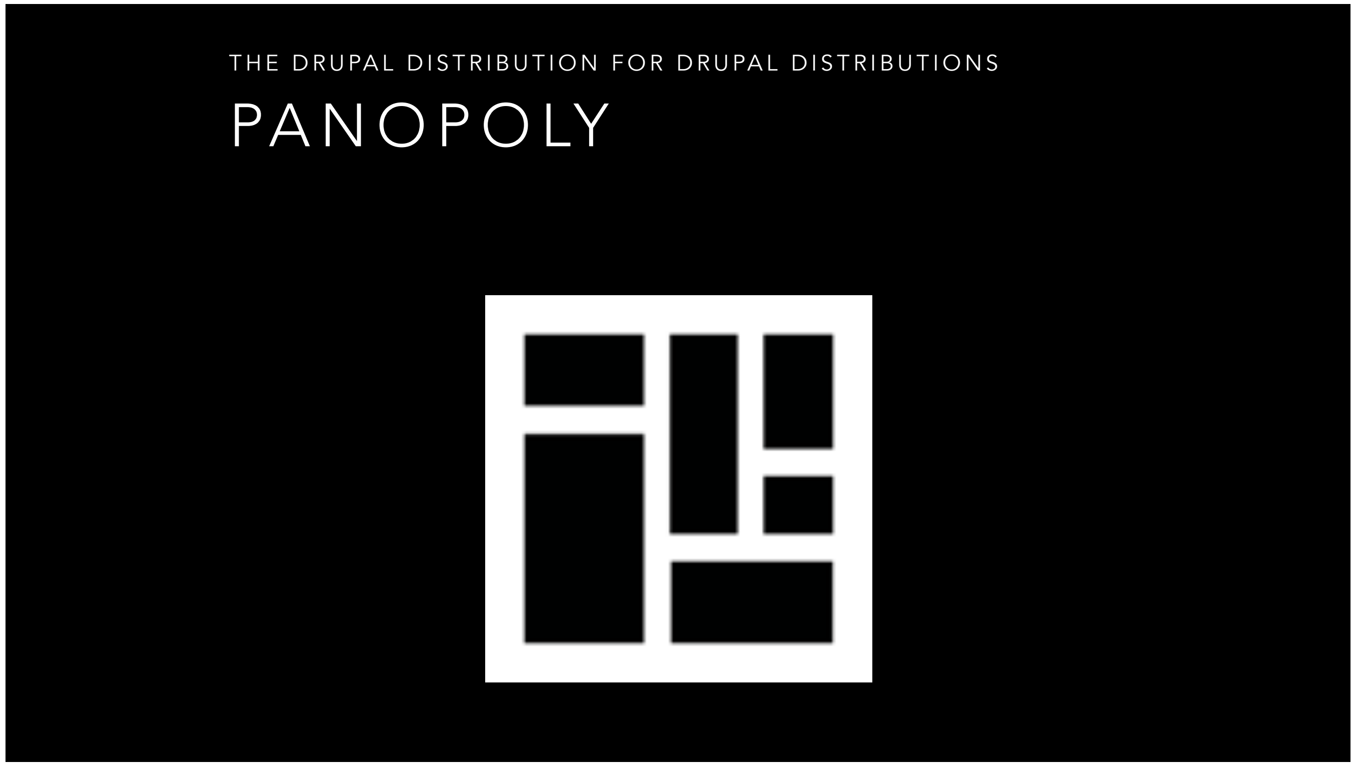
Completed 22 of 98. 22%

Installed *Date* module.

- ✓ Choose profile
- ✓ Choose language
- ✓ Verify requirements
- ✓ Set up database
- ▸ **Install profile**
- Configure site
- WCMC Configuration
- Finished

(MM) Spinning up a vanilla Drupal site can leave you with lots of base configuration work still to do to make the site usable and ready for production.

We wanted that out of the box. We wanted someone to get as close as possible to a great WCMC website without doing anything. So all those functions on the last page, we wanted to make them easily available



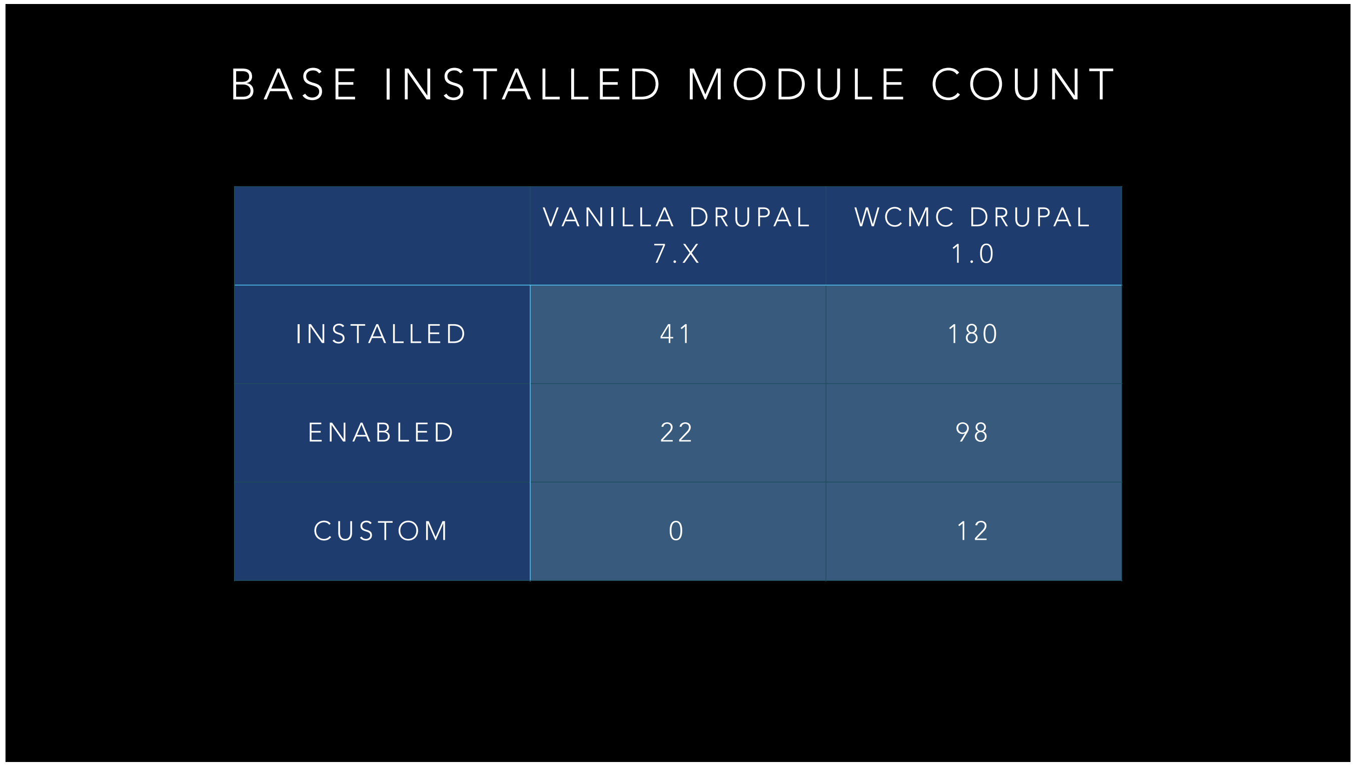
THE DRUPAL DISTRIBUTION FOR DRUPAL DISTRIBUTIONS

PANOPOLY

(MM) One of the primary decisions we made during development was to use the Panopoly framework as the base for our distribution.

Panopoly provides a curated set of best practice contributed modules. <https://drupal.org/node/1717680>

It ties together a robust panels-based workflow for the placement and display of content, nice utilities such as media management tools, and provides a vastly superior administration experience for both the site builder and end user.

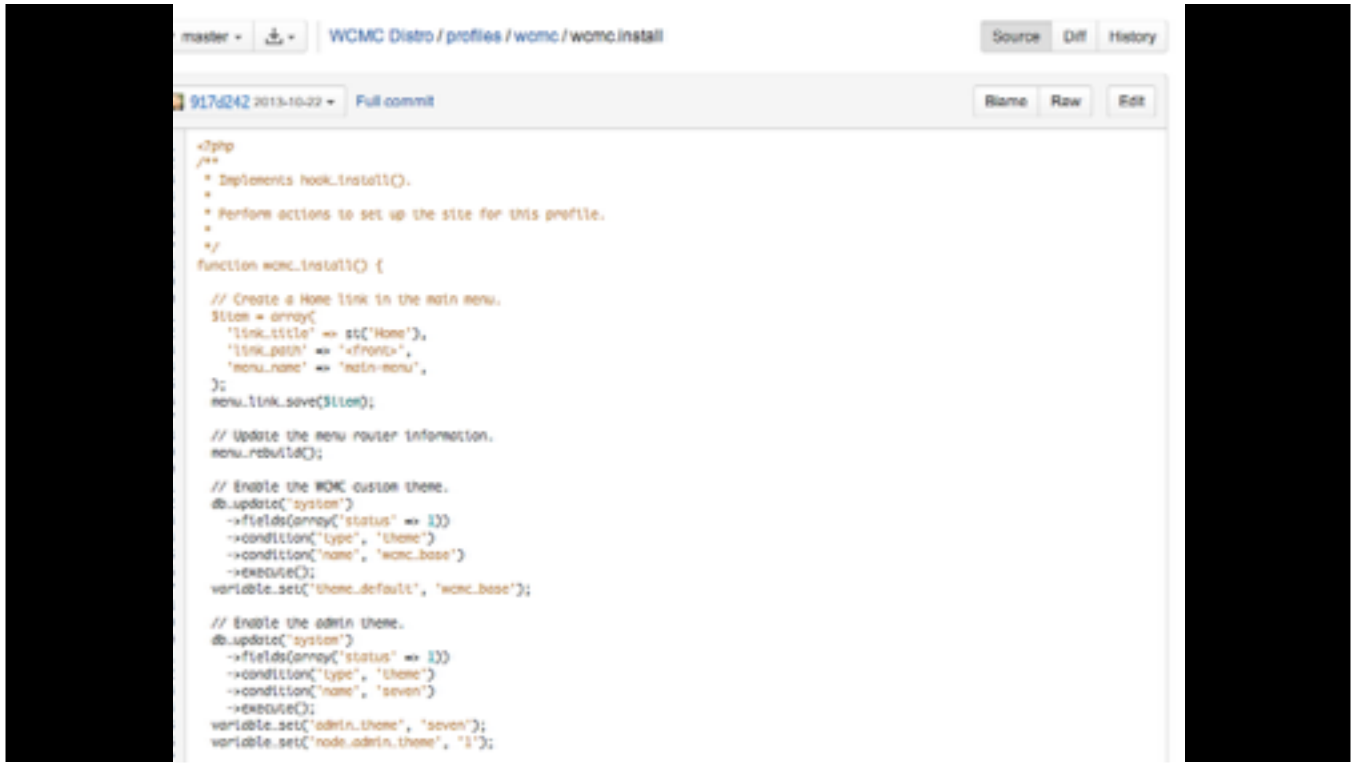


BASE INSTALLED MODULE COUNT

	VANILLA DRUPAL 7.X	WCMC DRUPAL 1.0
INSTALLED	41	180
ENABLED	22	98
CUSTOM	0	12

(MM)

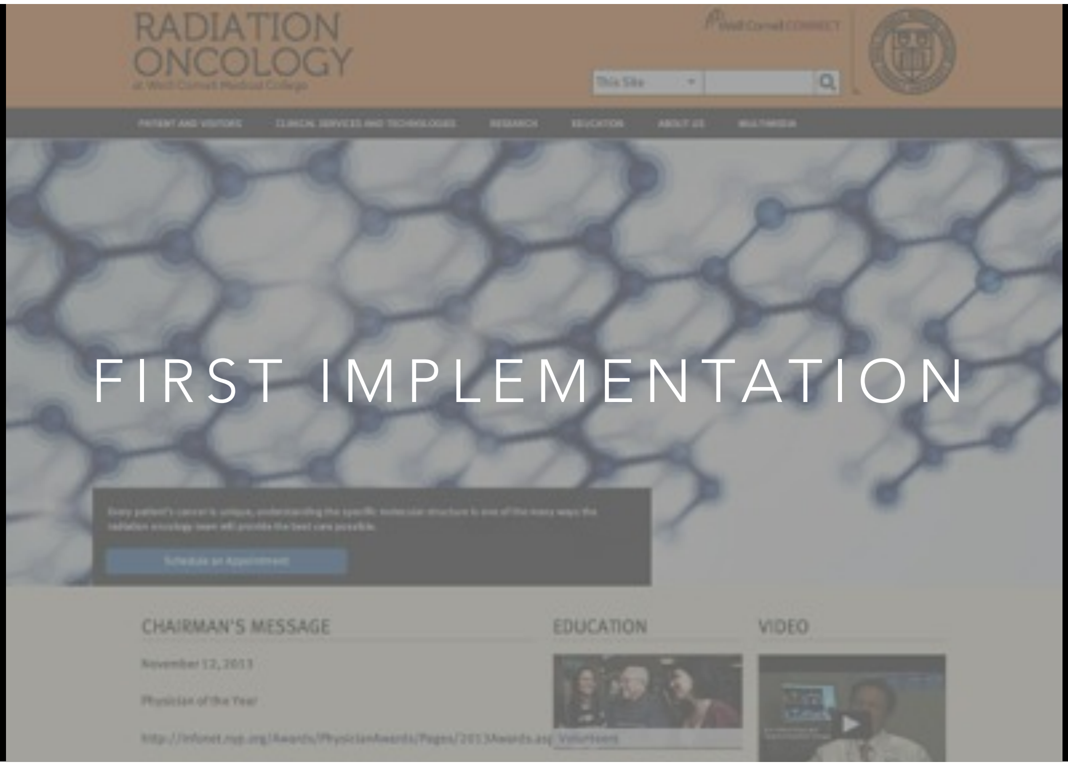
- Plugged in our own code by creating a custom base theme and a set of modules and custom panes.
- Used features wherever we could.
- This is a lot, but the number should reduce over time (as there’s some example modules in the count)



```php
<?php
/**
 * Implements hook_install().
 *
 * Perform actions to set up the site for this profile.
 *
 */
function wcmc_install() {

  // Create a Home link in the main menu.
  $item = array(
    'link_title' => st('Home'),
    'link_path' => '<front>',
    'menu_name' => 'main-menu',
  );
  menu_link_save($item);

  // Update the menu router information.
  menu_rebuild();

  // Enable the WCMC custom theme.
  db_update('system')
    ->fields(array('status' => 1))
    ->condition('type', 'theme')
    ->condition('name', 'wcmc_base')
    ->execute();
  variable_set('theme_default', 'wcmc_base');

  // Enable the admin theme.
  db_update('system')
    ->fields(array('status' => 1))
    ->condition('type', 'theme')
    ->condition('name', 'seven')
    ->execute();
  variable_set('admin_theme', 'seven');
  variable_set('node_admin_theme', '1');
```

(MM) With a foundation set, the next step was to develop a profile to handle most of the configuration steps upon installation.

The installation profile installs and configures our modules, creates user roles and permissions, and applies our base theme.



FIRST IMPLEMENTATION

Every patient's cancer is unique, understanding the specific molecular structure is one of the many ways the radiation oncology team will provide the best care possible.

CHAIRMAN'S MESSAGE

November 12, 2013

Physician of the Year

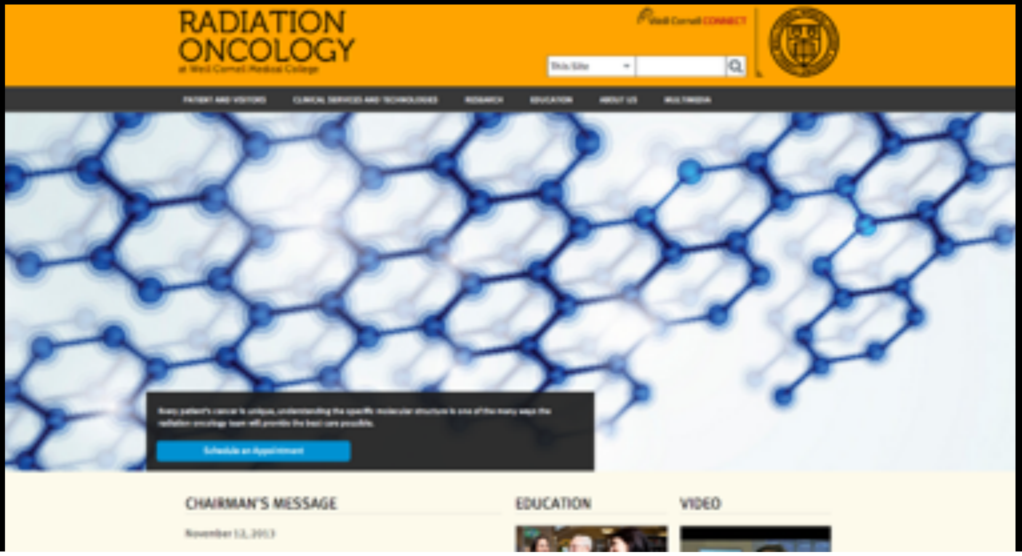
http://infonet.nyp.org/Awards/PhysicianAwards/Pages/2013Awards.asp

(JH)

- We took the dive. So it begins. Planning turned to action.
- Radiation Oncology was first department

CORNELLRADIATIONONCOLOGY.ORG

RADIATION ONCOLOGY

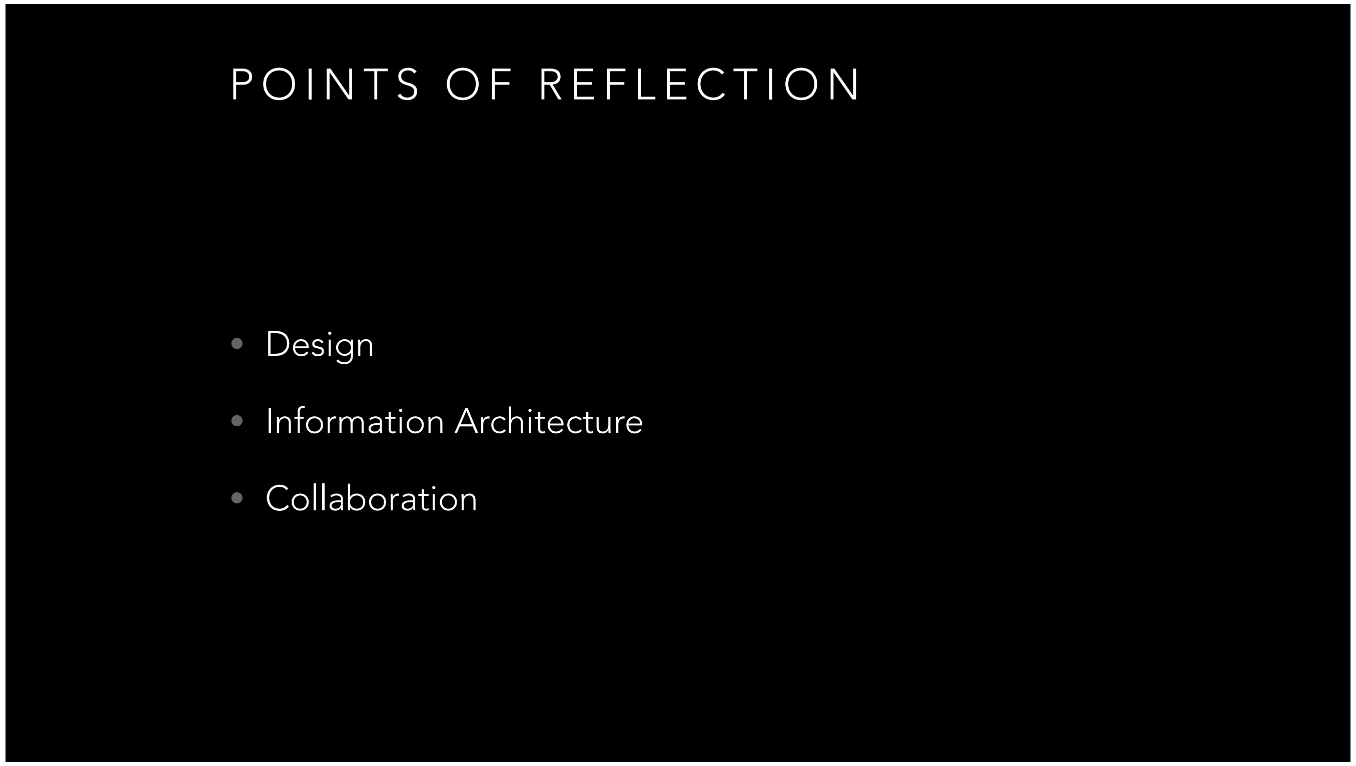


Every patient's cancer is unique, understanding the specific molecular structure is one of the many ways the radiation oncology team will provide the best care possible.

November 12, 2013

(JH)

- Resulting site captured all of our core exercises in site integration
- Learning as we built it
- Building for the new CMS, we were fine-tuning our distro as we built functions that would be eventually wrapped into it
- Our processes were historically built around static sites and required rethinking
- Required a repaired production strategy between us and the client that was more 360 degrees
- Production time was extended for R&D, settling production methods and normal production
- Wasn't a hard sell to clients – it was beautiful.

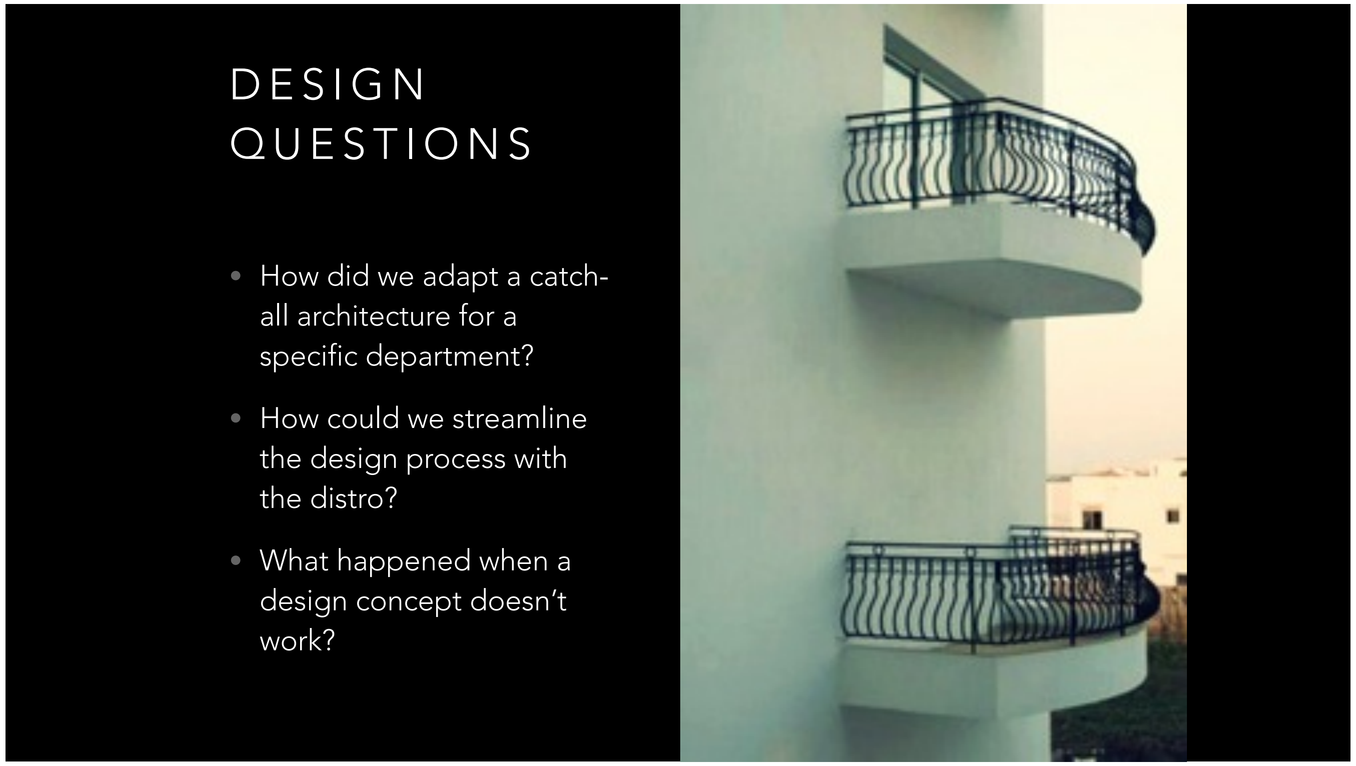


POINTS OF REFLECTION

- Design
- Information Architecture
- Collaboration

(JH)

- Following launch, we started to do some lessons learned in regard to:
- Design
- IA
- Collaboration (between the team and between the team and the client)

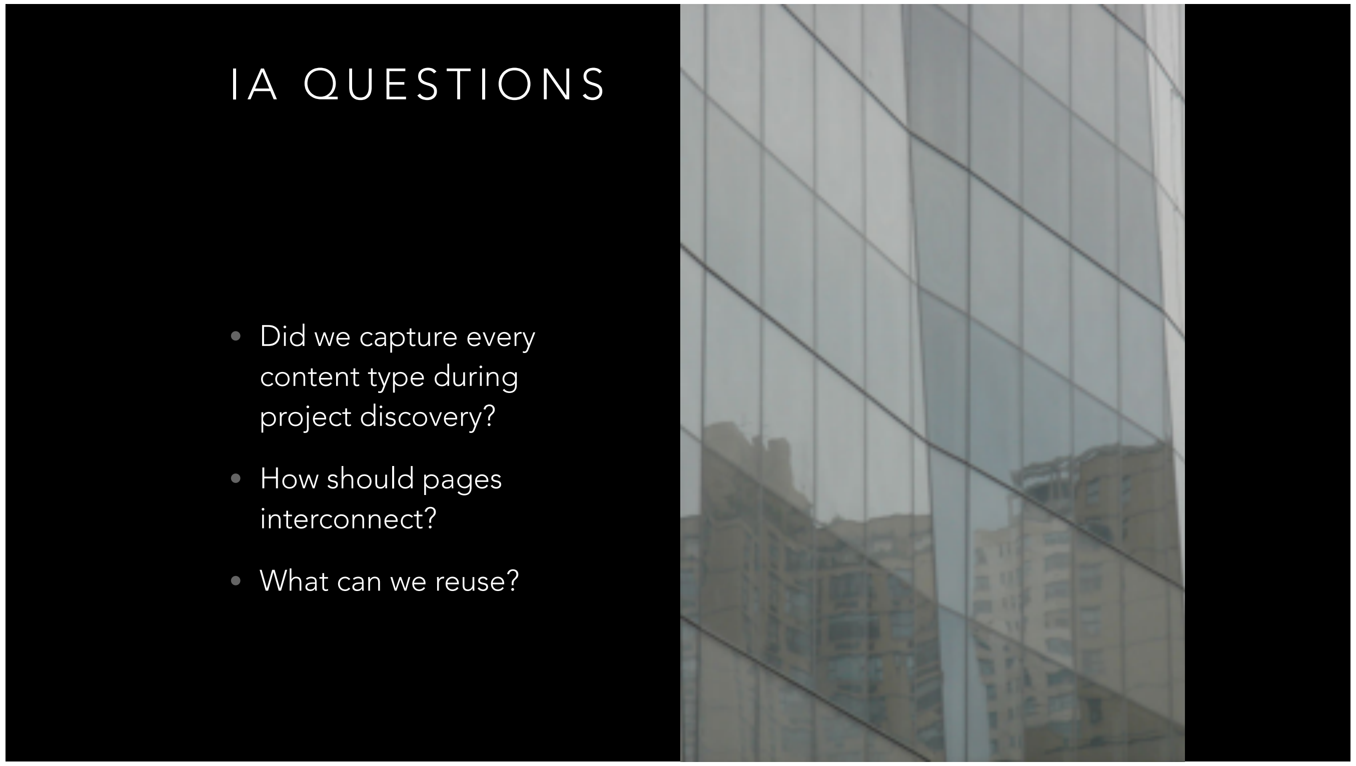


DESIGN QUESTIONS

- How did we adapt a catch-all architecture for a specific department?
- How could we streamline the design process with the distro?
- What happened when a design concept doesn't work?

(JH)

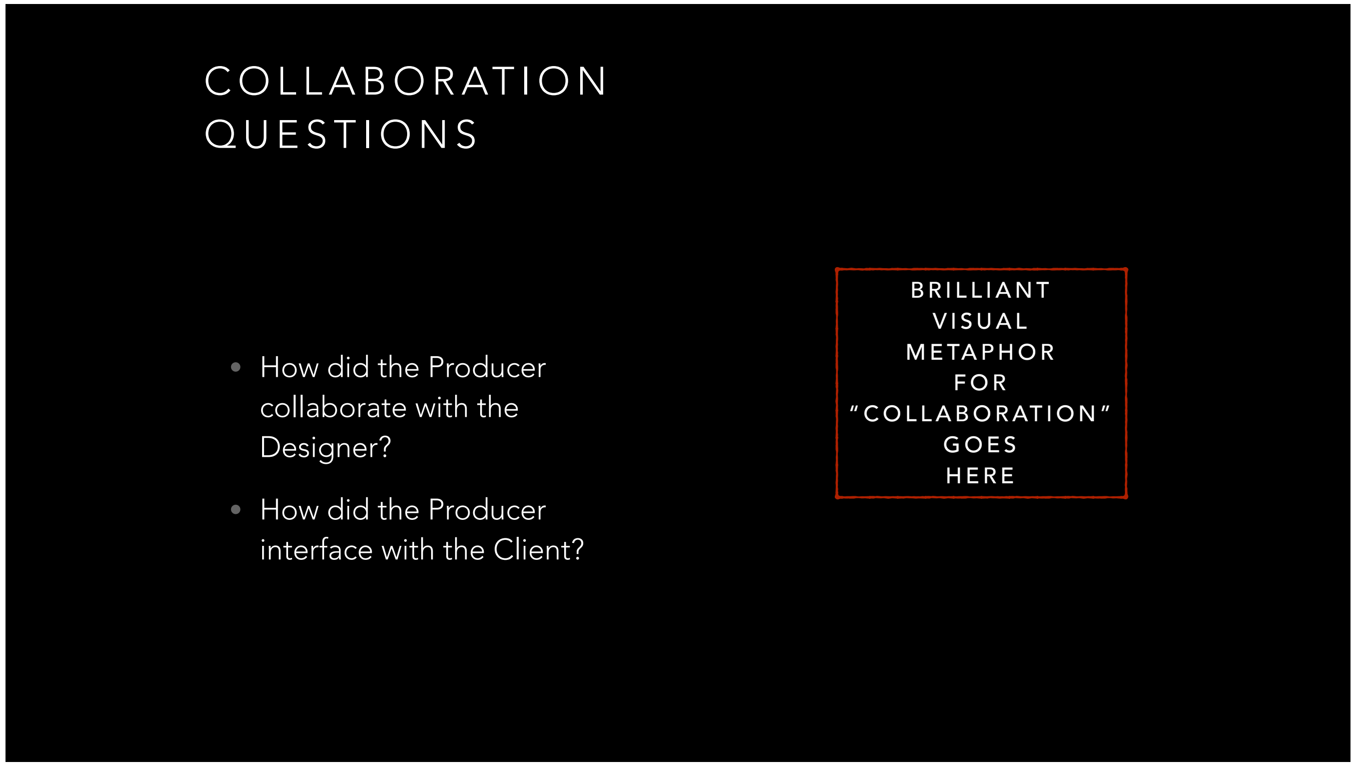
- How we adapt a catch-all architecture for a specific department? Answer bullets below
  - We had to communicate the optimizations and desired 'limitations' (navigation tree) to the client and demo – and demo again
  - Looked at their content first and got them involved in the process more so than in the past
- Streamline design process with a distro? Answers bullets below
  - We found that we did not collect enough information up front (content types)
  - Going back to our content strategy: we hadn't though through how the content types could interact fully from homepage to interior and laterally between pages
  - More communication and information gathering in the planning phase
  - Rethink timelines and how to layer milestones



IA QUESTIONS

- Did we capture every content type during project discovery?
- How should pages interconnect?
- What can we reuse?

(MM)



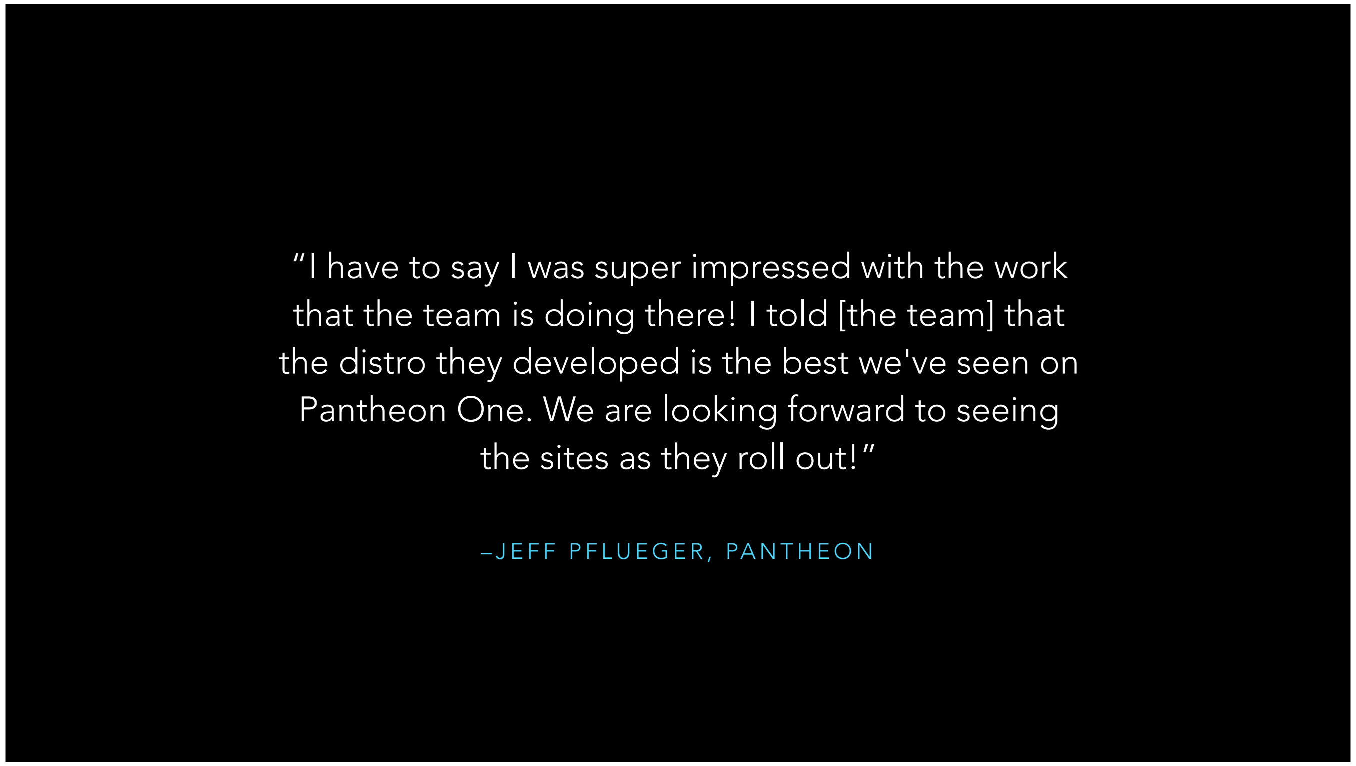
COLLABORATION QUESTIONS

- How did the Producer collaborate with the Designer?
- How did the Producer interface with the Client?

BRILLIANT VISUAL METAPHOR FOR “COLLABORATION” GOES HERE

(JH)

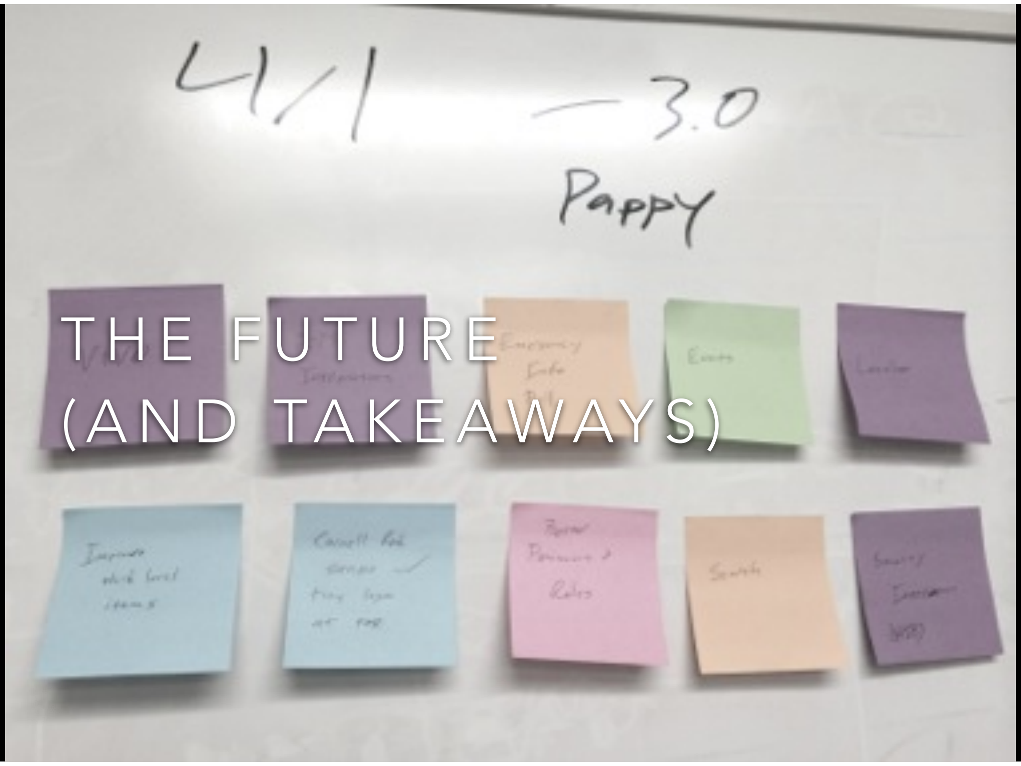
- How did the Producer Collaborate with Designer?
    - Rethink the roles as historically the designer was one-man-army (now we have a specialist)
    - Information gathering on the front end
    - Review of design with producer to ensure it takes advantage of the CMS
    - Understanding where we can turn a new design concept and bake it into our future distr.
- How did Producer Interface with Client?
    - Understanding client involves understanding their audiences and goals and shaping our future distros’s for that (example, student, research, mission sites)
    - Producer needed to be more involved directly with client – ask the required questions
    - Content efficiencies and standardizing that content receipt



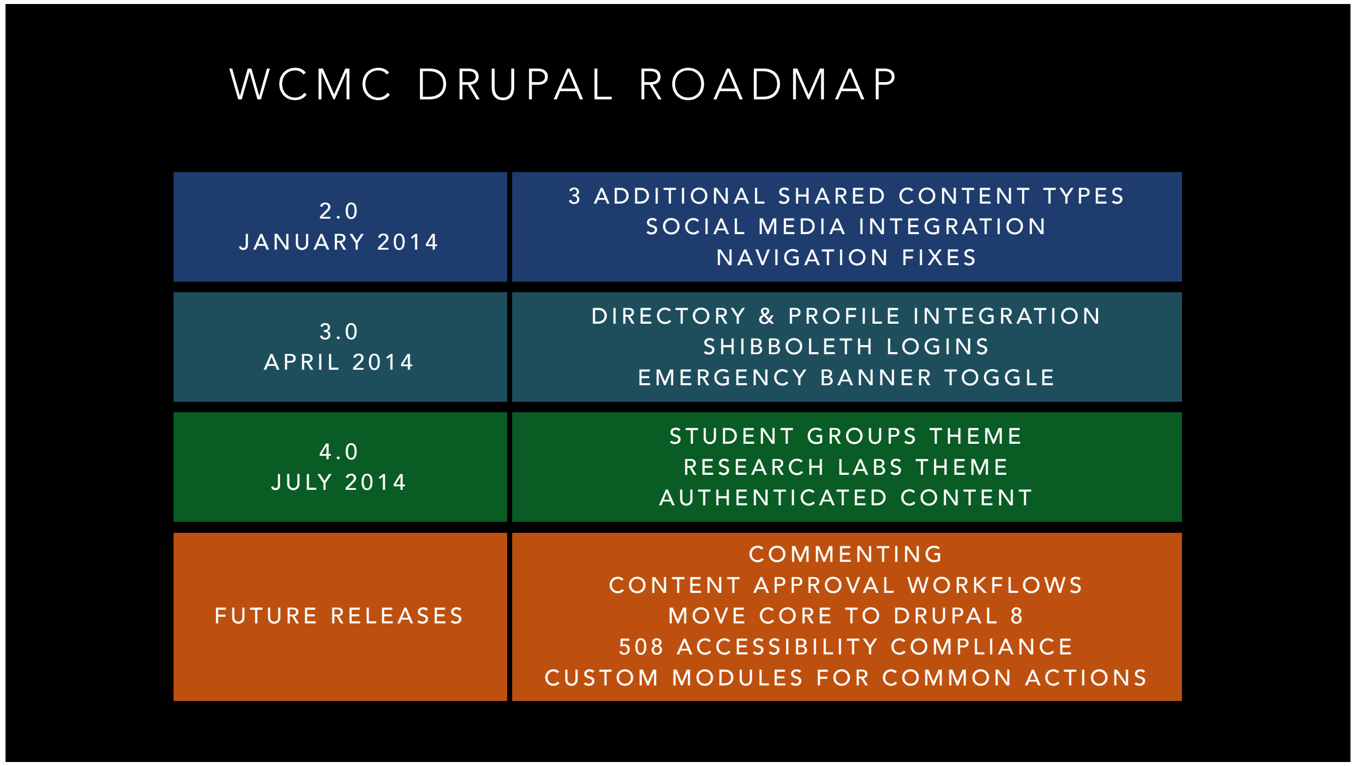
> “I have to say I was super impressed with the work that the team is doing there! I told [the team] that the distro they developed is the best we've seen on Pantheon One. We are looking forward to seeing the sites as they roll out!”
>
> –JEFF PFLUEGER, PANTHEON

(DD) We got a small bit of a lift around the time this project was closing, when we had a site visit from our reps at Pantheon and they provided some praise that we were quite honored to receive.

So I guess my Jobs-esqe quote wasn’t TOO ridiculous.



THE FUTURE (AND TAKEAWAYS)



WCMC DRUPAL ROADMAP

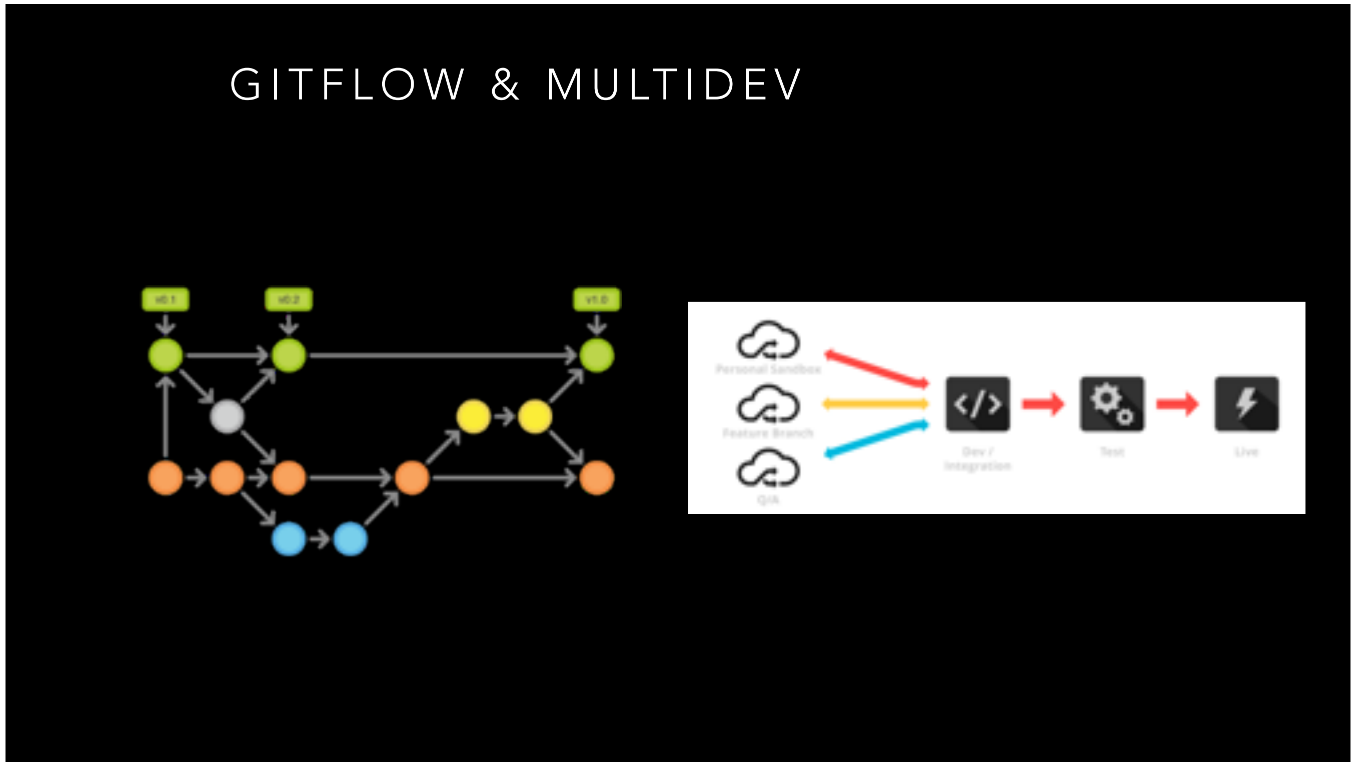
2.0 JANUARY 2014	3 ADDITIONAL SHARED CONTENT TYPES SOCIAL MEDIA INTEGRATION NAVIGATION FIXES
3.0 APRIL 2014	DIRECTORY & PROFILE INTEGRATION SHIBBOLETH LOGINS EMERGENCY BANNER TOGGLE
4.0 JULY 2014	STUDENT GROUPS THEME RESEARCH LABS THEME AUTHENTICATED CONTENT
FUTURE RELEASES	COMMENTING CONTENT APPROVAL WORKFLOWS MOVE CORE TO DRUPAL 8 508 ACCESSIBILITY COMPLIANCE CUSTOM MODULES FOR COMMON ACTIONS

We're planning quarterly releases (we're code naming them after delicious whiskeys) to keep adding functions to our base. And again, we can push down these updates from our master repository to individual sites as the time is right.

Briefly:

2.0 will feature some additional content types we couldn't get to in the initial release - a strong news module, a representation of a person, and such.

3.0 is really about integration with other systems. We have two faculty profile systems and a new online directory that we want to be able to pull data from, and those are critical to sharing of content.



GITFLOW & MULTIDEV

Gitflow is being used to get features that are currently in development for the distro into site development pipelines, so that we don’t have to wait for a full release if something is ready sooner.

MultiDev is something we’re not currently using, but are planning on it for the larger sites to ensure that multiple developer branches can be worked on simultaneously.

BUILD A DISTRO IF...

- You're a production shop.
- You expect to build a lot of independent sites but need a common base platform.
- You have a common set of modules you need to roll out for every site.

DON'T BUILD A DISTRO IF...

- You work on a single site.
- You have a couple of sites, all run by the same group and operating in a similar manner.
- You can't devote time to continued development.

IF YOU DO BUILD ONE…

- Use Panopoly as a distribution to build off of.
- Do your research for design & features before you start building sites.
- Plan it like a product.

THANKS FOR LISTENING.

QUESTIONS?