



for before the session:

This session is for people who want to know more about sprints and mentoring, who are mentors or want to be mentors, and to show sprint organizing is something you can do.

You are in the right spot if you are now or one day might be a mentor.

If that doesn't sound like you, it is ok to pick a different session. We wont mind. :)

If you have questions, write them down or tweet them so you remember them. We will have time for questions at the end of the session.

[ZenDoodles]

Our goals:

- Introduce contribution training resources

- Introduce all the stuff happening at this event, and prep you (yes you!) to participate.

- Give you the resources to plan a contribution sprint or training at your own user group, company, or Drupal event.

Our slides and a handout are linked from our session page

<http://portland2013.drupal.org/node/2433>

The urls, and resources, we mention during the talk are linked from there, so you don't have to try and write them down.

Who are we?



[add1sun, xjm YesCT, ZenDoodles]

We each introduce ourselves.

photos, first names, user names, and companies

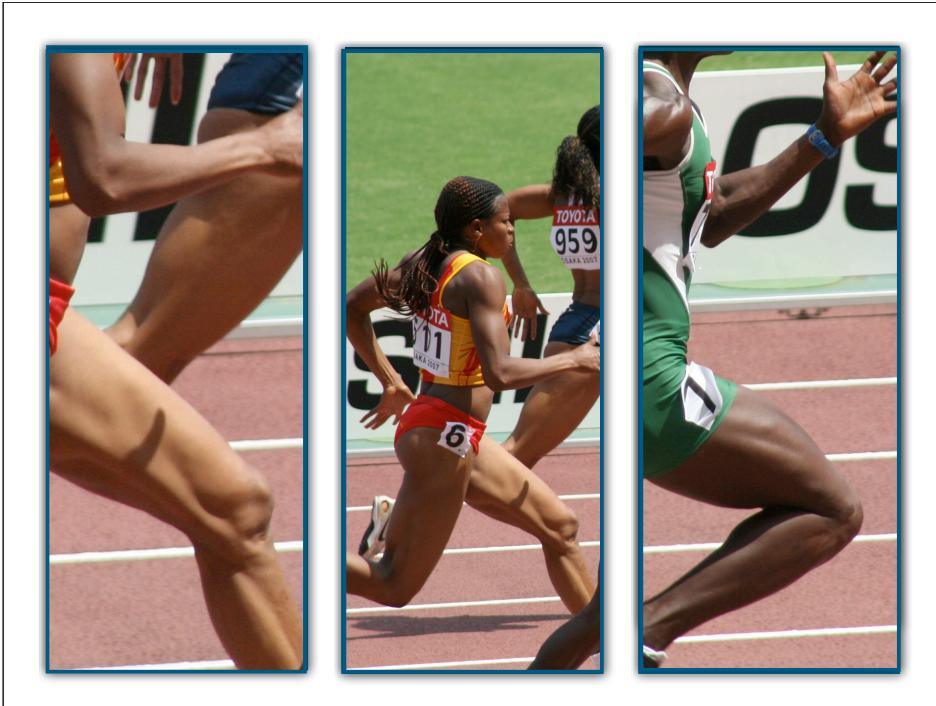
[yesct]

Hi. I am Cathy Theys, YesCT on drupal.org. I work at comm-press based in Hamburg Germany doing community contribution and blogging about contribution.

I am a regular mentor
at the weekly core office hours
in irc.
I organized the Global Sprint Weekend that was in March.

[ZenDoodles] We've led nine live core mentoring sprints at Drupal events in the past year, and Cathy also organized the global sprint weekend in March.

Also, if you look around the room, everyone wearing a green Portland shirt is going to help with the sprint here at Portland.



[ZenDoodles]

@todo make order and images match those in final remember in slide 49

Jess: The case for core contribution

Addi: Contribution solutions

Cathy: Planning a sprint

Me: This week's sprint and the lead-up to it.

Cathy: How to be an effective mentor and what to do next.

Then Questions...

If you only get three things out of the next hour, we hope you'll feel confident with

- **Finding (and using) resources**

- Including the tools to help you plan and execute a successful sprint, and maybe how to find some of the people who can help you along the way.

- **Plan to deal with development environment configuration**

- We don't expect you to know how to set up a dev stack on every system, but we hope we can provide a couple of strategies for addressing these inevitable issues.

- **Prepare a task list in advance.**

- Issue queue gardening is a skill. Your mentoring will be so much easier when you understand how to break it down and make that list.

But first, just some basics, so we're all on the same page...



What is a sprint?



[ZenDoodles]

A sprint is a focused contribution event.

The case for core contribution



[xjm]

So I'm going to talk a bit about the case for core contribution, or the problems we're trying to solve.

I'd like to begin by telling the story of how I got involved, because I think I faced the same challenges a lot of people do.

Core is the best way to learn



[xjm]

Two years ago, I was trying to upgrade the Taxonomy Access Control module to Drupal 7 for a client. I knew nothing about Drupal 7 at the start, but with some help I was able to get about 80% of the way through it before I got stuck. I talked to some other people about it, and we came to the conclusion that we needed a new alter hook, `hook_field_widget_form_alter()`. So we're like, okay, we're going to do this, we're going to file a Drupal core issue asking for this hook. (gesture)

So we posted the issue, and... I kid you not, sun had rolled a patch for this two hours later, like I came back from lunch and there was a patch there. But it was failing tests, and I didn't know how to fix it. By this point my port of TAC is a couple weeks late for my contract, and so I started asking, you know, what can I do to make this patch get in faster? What can I trade? And people suggested working on other core issues. My reaction is, okay, so, I'm not a core developer; I don't know what you expect me to do. I don't know anything about this stuff.

As fate would have it, two things happened that same week.

The first was that chx asked in IRC for someone to update a patch, just taking out an extra blank line that he'd accidentally added. And I go, huh. I could totally do THAT. So I did it. I downloaded the patch, applied it, made this one small change, and then uploaded a new patch.

And it was such a tiny, silly thing. chx could of course have made the change himself in like thirty second flat. But instead of doing that, he gave me the opportunity to take that first step, to cross that line in my head.

The second thing that happened that same week is that the issue summary functionality was deployed on Drupal.org. For anyone who's not familiar, the main post of a Drupal.org issue (before the comments) is editable. The idea is that anyone who's looking at an issue can collaborate to update it, to describe the current status. So instead of having to read through fifty or 100 comments to figure out what's going on, you can just read the summary.

So I figured that was something I could do as well, and I started writing issue summaries. Each day, I took one Drupal core issue, read the whole thing, researched things in the issue I didn't understand, and then wrote up a concise, up-to-date summary to save everyone else time.

A bit after this the patch for `hook_field_widget_form_alter()` was RTBCed, and was just stuck near the end of a big backlog of RTBC issues. RTBC by the way stands for "Reviewed and Tested by the Community," meaning it's ready for a core maintainer to look at it. I switched to writing summaries for RTBC issues, because I figured that by saving the core maintainers time reading these issues, I'd also help myself in the long run, because the less time they had to spend making heads or tails of a patch, the sooner they'd get to mine.

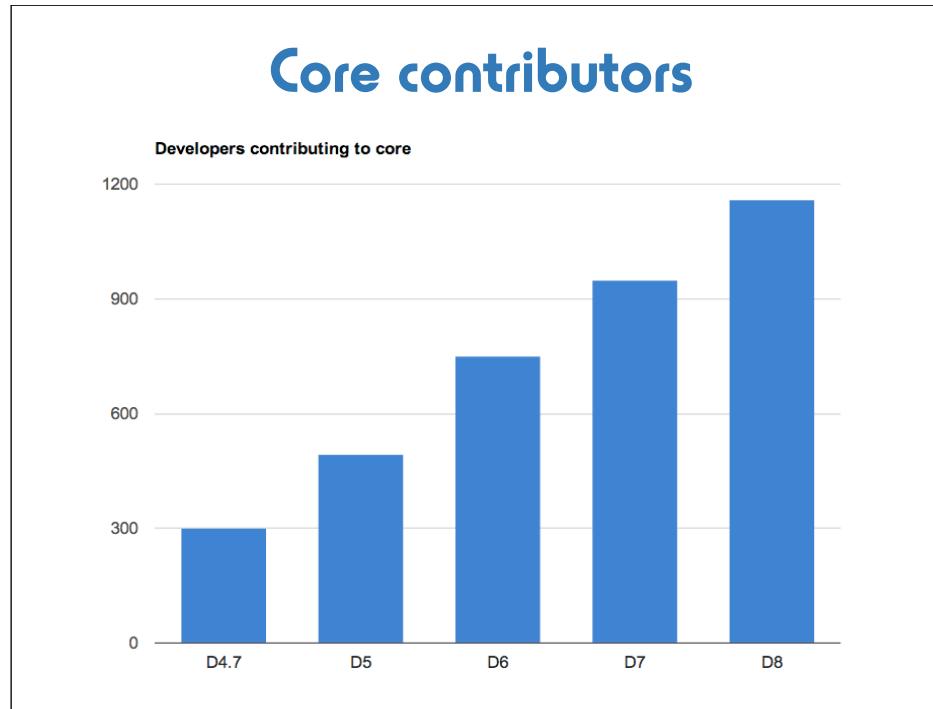
A bunch of stuff happened as a result of me writing those summaries, beyond helping my patch get in. One thing was that every now and then I'd come across an issue that I actually knew how to fix. And thanks to the encouragement chx had given me, I started submitting patches myself.

Another thing is that I figured out that there were a lot other Drupalists in the same situation as me, who wished they could contribute to these issues, but just didn't know where to start. So I tried to get other people to help write issue summaries too, and that ended up being one of the seeds for me for the core mentoring initiative.

And finally, completely by accident, I taught myself about core. By reading an issue every day, researching what I didn't understand, and then trying to explain it to other people, I learned a LOT. I started writing automated tests for bugs, and fixing more and more of those bugs myself. I started reviewing other people's code. And it was so entirely different from working on a contributed module. Instead of just hacking on things until they worked, I learned how to do them the right ways.

So that's why I think core is the best way to learn.

1. You learn a lot about Drupal
2. You get more feedback and interact with more people than in contrib
3. You're helping others at the same time
4. And you get this ownership in Drupal, this empowerment to solve problems you encounter.



[xjm]

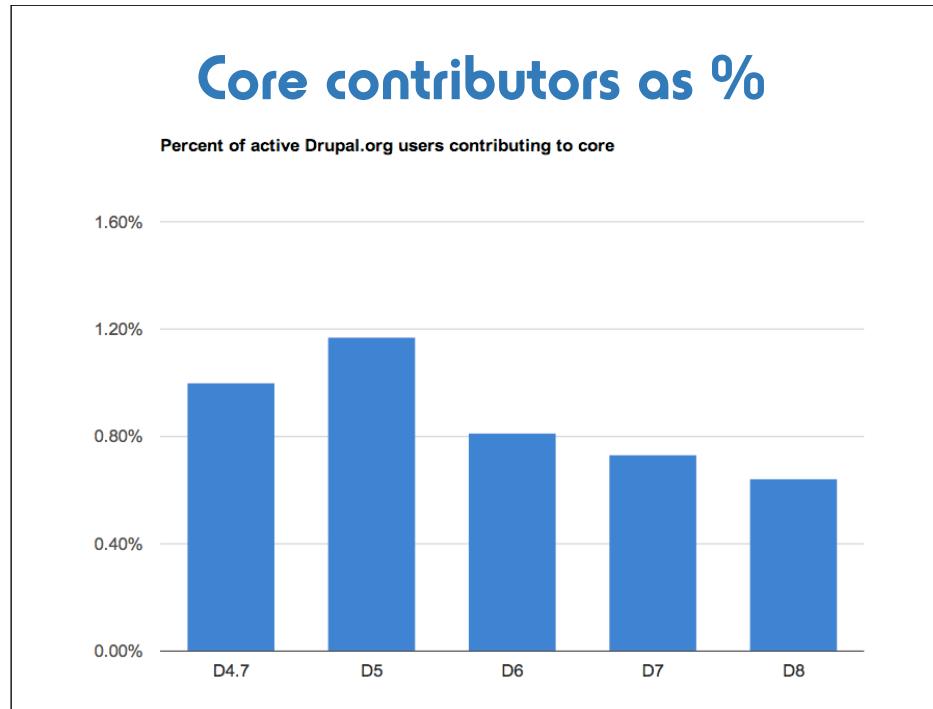
So let's look a bit at who contributes to Drupal core.

This graph shows the number of developers contributing to core for the past five major releases.

As you can see, the number increases steadily,

from about 300 in Drupal 4.7 to over 1200 already for Drupal 8--this graph is from a month ago--and Drupal 8's not even done yet.

Lots of people to learn from.



[xjm]

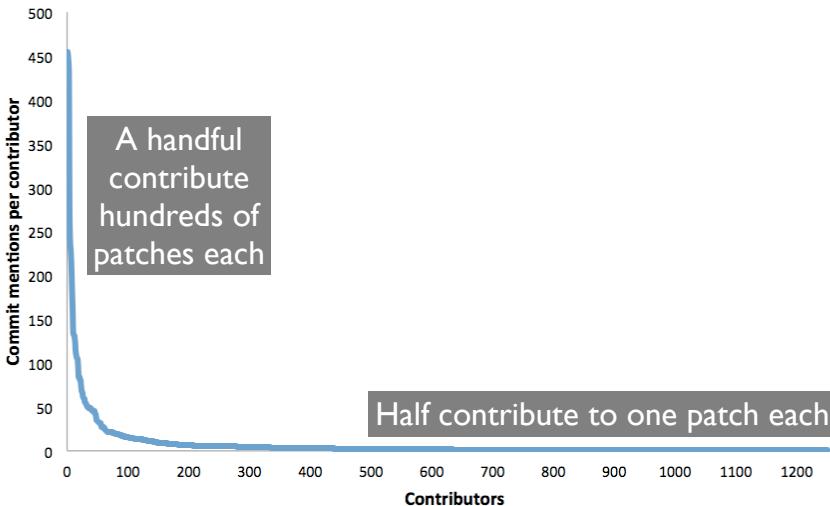
Something to keep in mind, though, is that Drupal's adoption is increasing faster than the number of core contributors.

As a percentage of the active users on Drupal.org, the number of core developers is actually dropping.

This is counting only active Drupal.org accounts,
people that logged in at least once during the year each version was released.

And we know that only a fraction of Drupal users even have Drupal.org accounts.

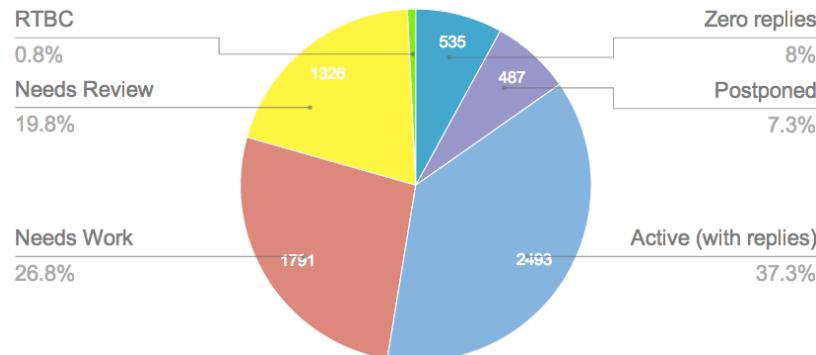
D8 patches per contributor



The workload isn't evenly distributed, either; most of these 1200 developers are what we call the "long tail." About half of the 1200 people have contributed to just one patch, while the top dozen or so contributors have contributed to hundreds of issues each.

Over 7000 Drupal 8 issues

Drupal 8 issue statuses



[xjm]

And finally, there's a large burden on existing regular contributors.

Right now there's over 7000 issues filed against Drupal 8 alone.

Core affects all of us, right?

If we rely on only a few hundred people to do the work on most of those issues, that's clearly a problem.

Nearly half of these issues are active, meaning there's no patch in the issue yet.

Over five hundred of them have zero replies.

And almost 20% of them are sitting at needs review.

One thing that's an ongoing challenge in Drupal core is a shortage of reviewers.

Ideally that slice would be much, much smaller

most of those issues should be marked needs work, and some might be RTBC

but a lot of them probably don't even apply anymore because they've been waiting for reviews for a month or longer

So there's a few things that need to happen, right?

We need to reduce the burden on this subset of core contributors.

We need to, ideally, recruit more core developers who will work on more issues.

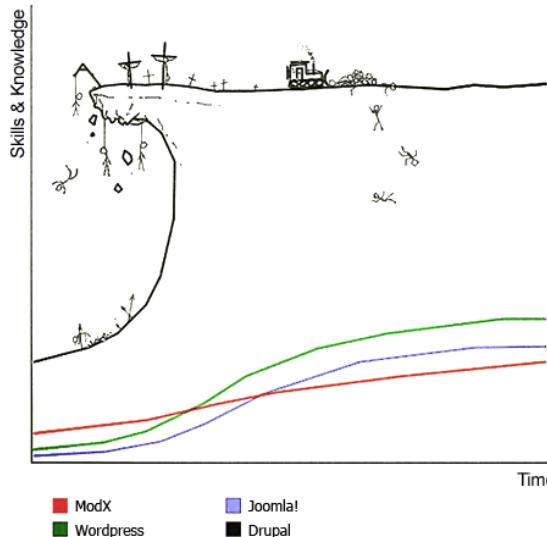
We need to do something about the issues with no replies.

And finally, we especially need to do something about the review gap, and make it easier to get good reviews faster.

(For anyone who's interested in that topic, by the way, Cathy and Andrea are doing a core conversation on patch reviews tomorrow.)

First steps can be difficult

Learning curve for popular CMS



[xjm]

It's hard, though, to grow the number of doers in core.

The first steps especially can be difficult, like they were for me.

Some of you have probably seen this picture before.

Some mock learning curves for different content management systems,
with Drupal's learning curve represented as an insurmountable cliff.

It's a joke, but it might resonate for anyone who tried to learn core development, especially a few years back.

There are, frankly speaking, barriers to core contribution.

The first, the most obvious, is this "I'm not smart enough" thing that I went through.

- Because of Drupal's famed complexity and flexibility,
- and maybe a bit because of our rockstar culture,
- we get the impression that there's nothing "little old me" can do.

Another thing is that core processes are hard to grok and more rigorous than contrib.

Also, it's unfortunately not entirely rare for people to have bad first experiences when they try to contribute,

- either unfriendly reviews, or worse, no reviews at all.

- This shortage of reviewers I mentioned means that someone's first ever patch
might go unlooked-at for months or years.

- That sucks, right? Who's going to try a second time when that happens?

Finally, people have difficulty finding tasks that they can do successfully.

- There's so many thousands of issues, so many things to learn, so much jargon.
- It's hard to know where to start.

Our goal in the core mentoring initiative is to give everyone a way up that cliff. Maybe even wear it down to a nice foothill. If you have Drupal site building experience, we want to make contributing back approachable to you.

Contribution solutions!

**Help people find tasks to do and
get the tools to do them**



[xjm]

So, our solutions focus on helping people find the task they can do, and getting the tools to do them.

The tasks



[xjm]

In the core mentoring program, we've focused on helping people find tasks and guiding them through the contribution process.

One issue, many tasks

[View](#) [Edit](#) [Outline](#)

Posted by [xjm](#) on May 4, 2013 at 4:50pm

Project:	Drupal core
Version:	8.x-dev
Component:	taxonomy.module
Category:	bug report
Priority:	normal
Assigned:	Unassigned
Status:	needs work
Issue tags:	needs accessibility review , Needs issue summary update , Needs manual testing , Needs tests

↑
“Needs” tags indicate the tasks to be done

[xjm]

This was my big realization back in 2011:

One issue actually includes many tasks.

We don't get that far with one rockstar programmer fixing a bug by herself; core is collaborative.

You don't have to write a patch all by yourself to help an issue along.

It's not always obvious to people what these tasks are, though,

so to make it easier to find tasks we've introduced a set of issue tags that identify tasks that need to be done for an issue.

Needs manual testing, Needs reroll, Needs issue summary update, and so on.

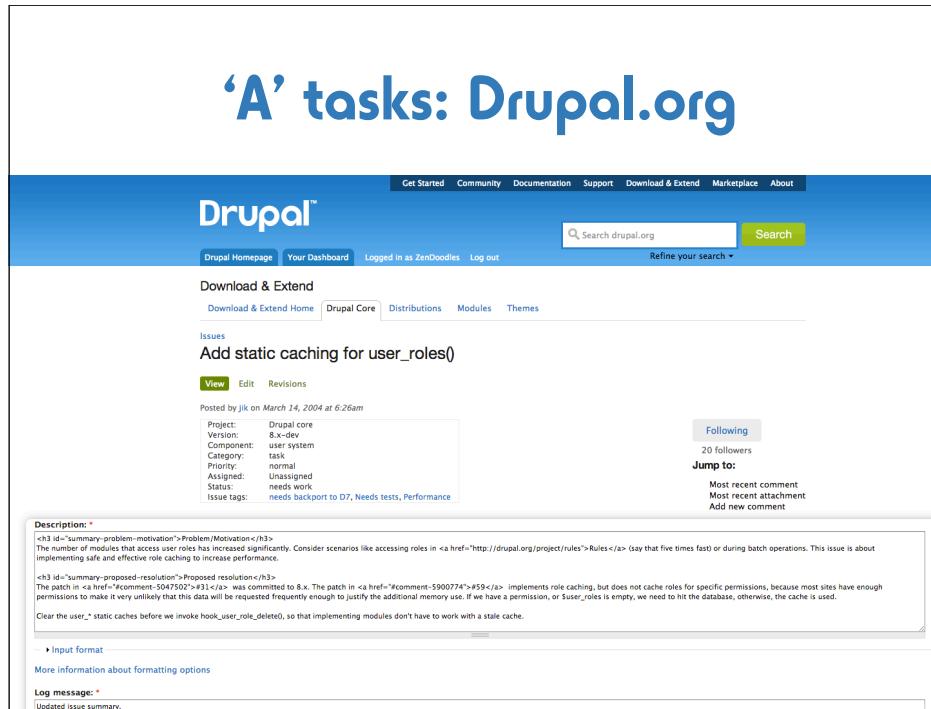
And it'd be totally awesome, by the way, if everyone in this room got in the habit of adding those tags to issues when appropriate.

The screenshot shows the 'Contributor tasks instructions' page on the Drupal.org website. The page has a blue header with the Drupal logo and a navigation bar with links like 'Get Started', 'Community', 'Documentation', 'Support', 'Download & Extend', 'Marketplace', and 'About'. Below the header is a search bar. The main content area has a 'Community' tab selected, showing sub-links for 'Community Home', 'Getting Involved' (which is highlighted in green), 'Chat', 'Mailing Lists', 'Member Directory', and 'Forum'. The main content title is 'New contributor tasks'. It includes a 'View' button, 'Edit' link, 'Outline' link, and 'Revisions' link. A note says 'Last updated April 18, 2013.' Below this is a paragraph about contributing to the Drupal community. A bulleted list follows: 'Contributing makes you feel good', 'Your contributions are valued', 'Many of the tasks are fun to do', 'The more you contribute, the more you learn', and 'Contributing makes you a more integrated part of the Drupal community'. A sidebar on the right is titled 'Page status' and says 'No known problems'. Below it is a 'Getting Involved Guide' section with a list of links: 'Ways to get involved', 'Talk with the community', 'Get a Drupal.org account', 'New contributor tasks' (which is expanded to show 'New contributor tasks: Anyone' and 'New contributor tasks: Non-English language speakers'), and 'English language speakers'. At the bottom of the page is a dark grey footer bar with the URL 'http://drupal.org/contributor-tasks'.

[xjm]

We also worked on some documentation in the Drupal.org handbook that explains how to do each of these tasks, to demystify the process.

I really recommend checking out this section of the handbook after the session is over. It's at drupal.org slash contributor dash tasks.



[xjm]

Finally, for the core mentoring initiative, we picked out the task types that work the best during IRC mentoring and sprints, and grouped them by the tools you need to do them. This helps us figure out what we can start someone with based on their past experience and skillset.

The first group, the “A” tasks in core mentor parlance, just require a user account on Drupal.org. These include:

- writing summaries for issues
- documenting the steps to reproduce an issue
- and triaging a reported issue to verify it, make sure it's filed correctly, and look for duplicates.

It's important to keep in mind that these are not necessarily quick or easy tasks. Sometimes writing a good issue summary takes a lot of work, and you get more out of it if you put more in.

‘B’ tasks: Git

```
commit ae0984fb711db81fd2157d189bddd105a9cc19
Author: catch <catch@35733.no-reply.drupal.org>
Date: Mon Apr 22 22:57:38 2013 +0100

Issue #1942346 by swentel: Convert Field API variables to OML.
commit 2d6ed6c862d3e4ea7e651f4f4518e0ed4df5f6f62fd
Author: Alex Pott <alex.o.pott@googlemail.com>
Date: Mon Apr 22 19:07:20 2013 +0100

Issue #1955286 by droplet | nod.: JSHint translation_entityO.
[tesla:drupal | Tue 08:41:18] $ git branch
7.x
8.x
actions
* else_if
  field
    frontpage
[tesla:drupal | Tue 08:41:28] $ git rebase origin/8.x
Current branch else_if is up to date.
[tesla:drupal | Tue 08:41:36] $ git --stat origin/8.x
[tesla:drupal | Tue 08:41:36] $ git diff --stat node/NodeStorageController.php | 2 +-
1 file changed, 1 insertion(<), 1 deletion(-)
[tesla:drupal | Tue 08:41:50] $ git show
commit 10cad61ac515d3dc763b334fb38fc5273b61282
Author: xjm <xjm65776.no-reply.drupal.org>
Date: Tue Apr 23 08:38:22 2013 -0500

  Correct else if to elseif in the node storage controller.

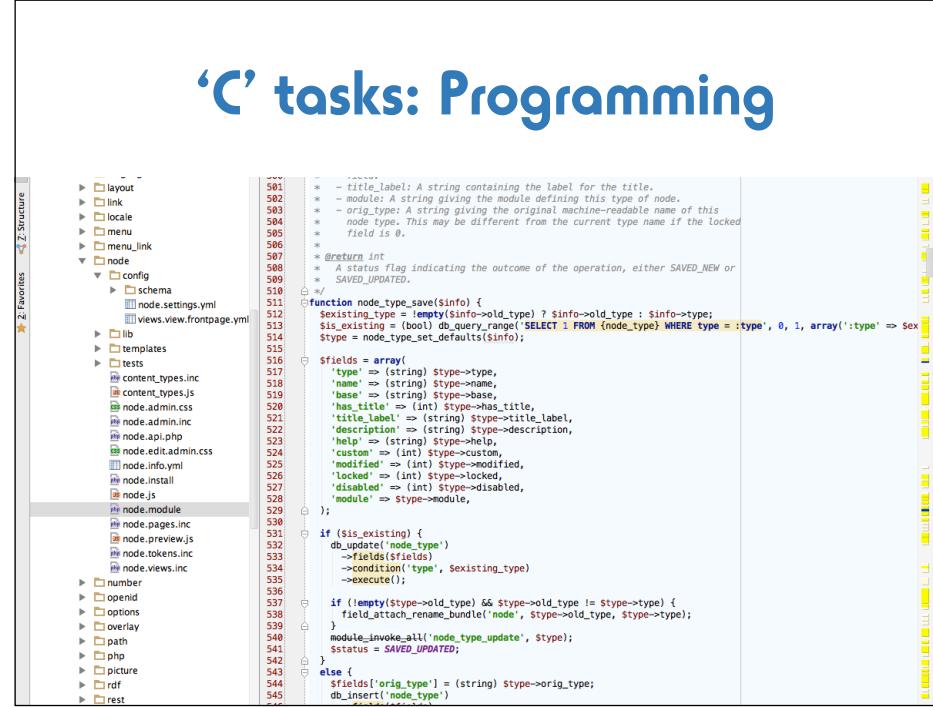
diff --git a/core/modules/node/lib/Drupal/node/NodeStorageController.php b/core/modules/node/lib/Drupal/node/NodeStorageController.php
index c9e832e..0f7c921 100644
--- a/core/modules/node/lib/Drupal/node/NodeStorageController.php
+++ b/core/modules/node/lib/Drupal/node/NodeStorageController.php
@@ -103,7 +103,7 @@ protected function invokeHook($hook, EntityInterface $node) {
  * @param $hook
  * @param $node
  */
-if ($hook == 'insert' || $hook == 'update') {
+elseif ($hook == 'insert' || $hook == 'update') {
  node_invoke($node, $hook);
}
-else if ($hook == 'predelete') {
+elseif ($hook == 'predelete') {
  // 'delete' is triggered in 'predelete' is here to preserve hook ordering
  // from Drupal 7.
```

[xjm]

The “B” tasks require using git, our version control system.
This includes small cleanups like that one-line fix chx asked me to do,
as well as

- manually testing things
- creating documentation patches
- and rerolling patches so that they apply to the latest HEAD

‘C’ tasks: Programming



The screenshot shows a code editor interface with a sidebar containing a file structure. The structure includes 'layout', 'link', 'locale', 'menu', 'menu_link', 'node' (which is expanded to show 'config', 'schema', 'views.settings.yml', 'views.view.frontpage.yml'), 'lib', 'templates', and 'tests'. Under 'tests', there are files like 'content_types.inc', 'content_types.js', 'node.admin.css', 'node.admin.inc', 'node.api.php', 'node.edit.admin.css', 'node.info.yml', 'node.install', 'node.js', 'node.module', 'node.pages.inc', 'node.preview.js', 'node.tokens.inc', and 'node.views.inc'. The main pane displays a snippet of PHP code:

```
501 * - title_label: A string containing the label for the title,
502 * - module: A string giving the module defining this type of node,
503 * - orig_type: A string giving the original machine-readable name of this
504 *   node type. This may be different from the current type name if the locked
505 *   field is 0.
506 *
507 * @return int
508 * A status flag indicating the outcome of the operation, either SAVED_NEW or
509 * SAVED_UPDATED.
510 */
511 function node_type_save($info) {
512   $existing_type = $info->old_type ? $info->old_type : $info->type;
513   $db = db_select('node_type')
514     ->fields($info->type)
515     ->condition('type', $existing_type)
516     ->execute();
517   $type = $info->type;
518   $type->type => $existing_type;
519   $type->name => $info->type;
520   $type->base => $info->type;
521   $type->has_title => (int) $info->has_title;
522   $type->title_label => (string) $info->title_label;
523   $type->description => (string) $info->description;
524   $type->help => (string) $info->help;
525   $type->locked => (int) $info->locked;
526   $type->disabled => (int) $info->disabled;
527   $type->module => $info->module;
528 }
529 */
530 if ($is_existing) {
531   db_update('node_type')
532     ->fields($info->type)
533     ->condition('type', $existing_type)
534     ->execute();
535
536 if (!empty($type->old_type) && $type->old_type != $type->type) {
537   field_attach_rename_bundle('node', $type->old_type, $type->type);
538 }
539
540 module_invoke_all('node_type_update', $type);
541
542 } else {
543   $status = SAVED_UPDATED;
544   $info->orig_type = (string) $type->orig_type;
545   db_insert('node_type')
546     ->fields($info->type)
547     ->execute();
548 }
549
550
551 }
```

[xjm]

And finally, the “C” group of tasks require programming skill.

- writing patches,
- writing automated tests,
- doing code reviews,
- and writing API change notifications.

The screenshot shows the homepage of drupalofficehours.org. At the top, there's a large blue header with the text "Task curation: drupalofficehours.org". Below the header is a navigation bar with tabs for "Tasks", "Review", "Mentors", and "Participants". The main content area has a blue sidebar on the left with the text "Drupal core mentoring" and "We've Got Issues™". The main content area contains a message about joining core contribution mentoring on Tuesdays and Wednesdays, and a note about listing mentors on your Drupal.org profile. Below this is a section titled "Active tasks" with a button to "Add a task". There are three filter dropdowns: "Task level" (set to "Any"), "Task type" (set to "Any"), and "Issue priority" (set to "Any"). A "Priority tasks" table follows, showing three rows of tasks:

Available	Task	Participants	Drupal.org issue	Priority	Status	Version	Action
No	Create patch	daviddagino Mentor: jpm	1983340: Change field UI so that adding a field is a separate task	major	active	8.x-dev	edit
+ Yes	Draft change notification		1901670: [Change notice] Start using PHPUnit for unit tests	critical	active	8.x-dev	edit
+ Yes	Expand test coverage	jeana	1007746: Reordering fails with more than 100 items in a menu 610076: Add a primary key to the	major	needs work	8.x-dev	edit

[xjm]

We also built a website to assign people to tasks, because the Drupal.org issue queues don't really do what we need them to.

The site is drupalofficehours.org.

We use it to coordinate both sprints and core mentoring in IRC.

It provides a curated list of tasks for core issues.

Having this site

- Filters the noise of the core queue

- Helps match contributor skills to available tasks

- Allows us to target important issues

- And, most importantly, matches contributors with mentors

We'll explain how to use this website at the BoF following the session.

The tools



[Addi]

Solution part B: Get people the tools they need to do the tasks

Working with the Community

- ▶ Drupal.org account
- ▶ Issue queue
- ▶ Local AMP stack
- ▶ Git
- ▶ Internet Relay Chat (IRC)



[Addi]

- For A level tasks: account, issue queue, and perhaps an AMP stack for reproducing
 - For B and C levels: A + AMP and Git
- IRC is so helpful for getting help and working with others in real-time
- keeps people connected after a sprint

The screenshot shows the homepage of The Drupal Ladder website. The header features a large blue title "The Drupal Ladder" and the URL "drupalladder.org". Below the header is a blue banner with a white background area containing a logo of a person climbing a ladder into a blue water drop, with the text "Drupal Ladder" above it. A red rectangular box contains the message "This website is under active development. All content is a work in progress.". To the right of the message are login fields for "Username" and "Password", with links for "Create new account" and "Request new password" below them. A "Log in" button is also present. Below the banner is a navigation menu with links: "About", "Drupal Core Ladder", "Ladders", "Lessons", "Resources", "Chat", "Get involved", and "D8 Initiatives Calendar". The main content area has a white background and displays the heading "About the Drupal Ladder project" and the subtext "The Goal of the Drupal Ladder is to have 1% of the Drupal Community contributing to core by 2014". There are two blue links at the bottom of this section: "About drupalladder.org" and "How to run a learn sprint".

[Addi]

Created for user group meetups (short sprint periods)

- Learn sprint lessons (essentially tool lessons)
- Resources for issue sprints



[Addi]

Doesn't follow the ladder directly, but uses some of the lessons.

Covers 2 main areas, hands-on, to prepare people for getting down to work:

- Communication (IRC & Issue Queues)
 - Dev Environment (AMP & Git; what are they; why you need them; install them)
- Full curriculum for how to run this workshop yourself (will improve over the summer)

Planning a sprint



[YesCT]

I'm going to talk about what it takes to plan a sprint.

Mentors, location, and publicity.

Selecting sprint tasks in advance,

and scheduling a time

after the sprint

for doing **task reviews**.

Depending on the **size of your sprint**, there could be a lot of planning to do.

For example, the four of us have been working with the Drupal Association to plan this friday's sprint since January.

Smaller sprints need less planning.

Here are the **essentials** for when you plan your sprint.

Planning: Mentors



[YesCT]

You can host a sprint and be the mentor. Will you need additional mentors?

Estimate the number of people that might attend.

A ratio of 1 mentor to 8 participants keeps mentors busy, but not too busy. So if having more than 8 attendees, perhaps recruit mentors.

On the day of your sprint, if there are not enough mentors. No worries: pair experienced people with new people.

You don't have to do it all. When someone is having trouble with git, find someone just a bit more experienced.

Ask them to help and move to sit next to the person having trouble.

It builds the confidence of that on-the-spot new mentor to see they can help others. Mentoring new mentors is really rewarding.

Planning: Location



Global Sprint Weekend: Oak Park

Photo: YesCT

[YesCT]

Make sure your location has:

wifi

outlet strips

food: people will leave if food is not there, and everyone will lose an hour and half of sprint time.

A simple way of dealing with food is to get a menu, order in.

When reserving a location for my local sprint during the March SprintWeekend, I asked about wifi (they had it), outlet strips (they didn't. so we would bring some). I found a sponsor for food. But what I didn't ask about was tables. It was a meeting room that doubled as a yoga studio. And, the day before the sprint, I was picking up the key, and they asked if we would like tables. !!! I said yes. :) Phew! I'm glad they asked.

Be sure to confirm your space has what you need.

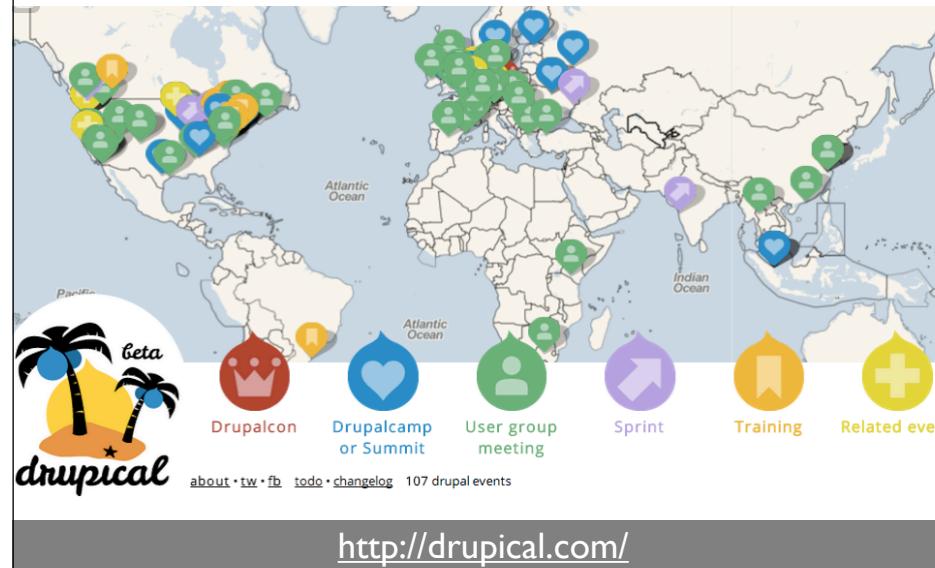
We ordered in lunch (only about \$80 to feed us all).

This is a picture of us at 2 tables with lunch evident.

We were on **both sides of rectangular tables facing each other.**

That or **round tables** work best to encourage interaction.

Planning: Publicity



[YesCT]

To get people to show up for your sprint, publicize it

drupical.com is an independent site, with realtime mapping of drupal events. It's useful for planning vacations around drupal events. :) drupical, like tropical, but for drupal. Your sprint can be on drupical.

This image is a sample of what the drupical.com map looks like with different colored pins, marking events all over the world. Sprint pins are purple. How do you get *your* sprint pin here? Make an event on groups.drupal.org. That not only tells people in your local groups.drupal.org group about the event, it makes it show on the drupical.com map automatically.

Additionally or as an alternative make an event on **meetup.com**. Its a good way for people to find your event that are not checking the drupal group.

When spreading the word about your sprint, in your announcements, say who is welcome at the sprint, usually that is:
anyone with site building experience, and also more regular contributors.

In your publicity, Remind attendees to bring...

what you need them to bring:

laptop for sure

maybe

outlet strip, snacks?

Planning: Development environments



DrupalCon Munich

Photo: Pedro Lozano

[YesCT]

On the day, people will have trouble getting set up to work on drupal 8. Plan for this.

We know from past camps and cons that although macs are popular, they aren't the only flavor of dev environments that shows up at a sprint. Other environments are equally popular.

Find people who have experience with setting up each: Linux, Mac, Windows, and ask them to come to your sprint.

Also, **encourage people to set up their dev environments before the sprint day.**

Send a reminder for them to do that at home, include resources that help them set up their environment. We have some linked from our resources handout.

What is a dev environment? Sure, it's a AMP Stack: apache, mysql and php.

It's also:

a shell, an editor, an irc client, git, dreditor.

At a minimum people need:

a web browser,

an account on d.o.,

dreditor to make d.o effective,

and **irc** to communicate with people while sprinting.

Some tasks can be done with just that minimal set of tools.

When doing your task selection,

select a few tasks that might just need their d.o account.

Also, simplytest.me rocks.

Simplytest.me helps people attending your sprint do more tasks, like test a patch or write steps to reproduce, without a local *AMP stack.

For other tasks, a local web server is needed. Sprint attendees are welcome to use any set up they want. Their favorite VM flavor, anything.

But for attendees that do not have anything yet,

an *AMP stack what we have new people set up.

We have lots of experience with setting up new contributors and we have found it's very helpful to have them set up their local AMP stack, like MAMP or WAMP. They can test it by trying to serve up a simple hello.html file and info.php file, ... and "then" git drupal and install that. Otherwise, people get confused.

Selecting sprint tasks		
A: Drupal.org		
Name	Where to find issues	Instructions
Issue summary	Issues tagged with "Needs issue summary update" <small>🔗</small>	Write an issue summary <small>🔗</small>
Steps to reproduce	Issues tagged with "Needs steps to reproduce" <small>🔗</small>	Document STR <small>🔗</small>
Triage and find duplicates	Issues with 0 replies <small>🔗</small>	Verify a reported issue <small>🔗</small>
B: Git		
Name	Where to find issues	Instructions
Backport	Issues with status "Patch (to be ported)" <small>🔗</small>	Backport a Drupal Core patch <small>🔗</small>
Create docs patch	Documentation issues <small>🔗</small>	Create a Drupal core patch <small>🔗</small>
Improve patch docs/code style	Issues marked CNR or CNW <small>🔗</small>	Improve patch coding style and documentation <small>🔗</small>
Manual testing	Issues tagged with "Needs manual testing" or "Needs screenshot" <small>🔗</small>	Manually test a patch for a Drupal issue <small>🔗</small>
Reroll	Issues tagged with "Needs reroll" <small>🔗</small>	Re-rolling patches <small>🔗</small>
C: Programmer		
Name	Where to find issues	Instructions
Write tests	Issues tagged with "Needs tests" <small>🔗</small>	Write a Drupal core automated test <small>🔗</small>
Create patch	Novice issues <small>🔗</small>	Create a Drupal core patch <small>🔗</small>
Draft change notification	Issues tagged with "Needs change notification" <small>🔗</small>	Write a change record <small>🔗</small>
Patch review	Issues marked CNR <small>🔗</small>	Review a Drupal core patch <small>🔗</small>

<http://core.drupalofficehours.org/task-info>

[YesCT]

This is a table of the task types
that Jess talked about earlier,
organized by the tools needed to do them.
It's at: core.drupalofficehours.org/task-info

The link is in the resources.

It's helpful when you are selecting sprint tasks.

Selecting tasks is the bare minimum planning you should do ahead of time.

It is essential for a smooth sprint.

Select tasks **no more than three days** in advance. If too far in advance, the tasks might not be relevant by the time of your sprint.

Be sure to pick a few tasks in each tool set:

Drupal.org,

git,

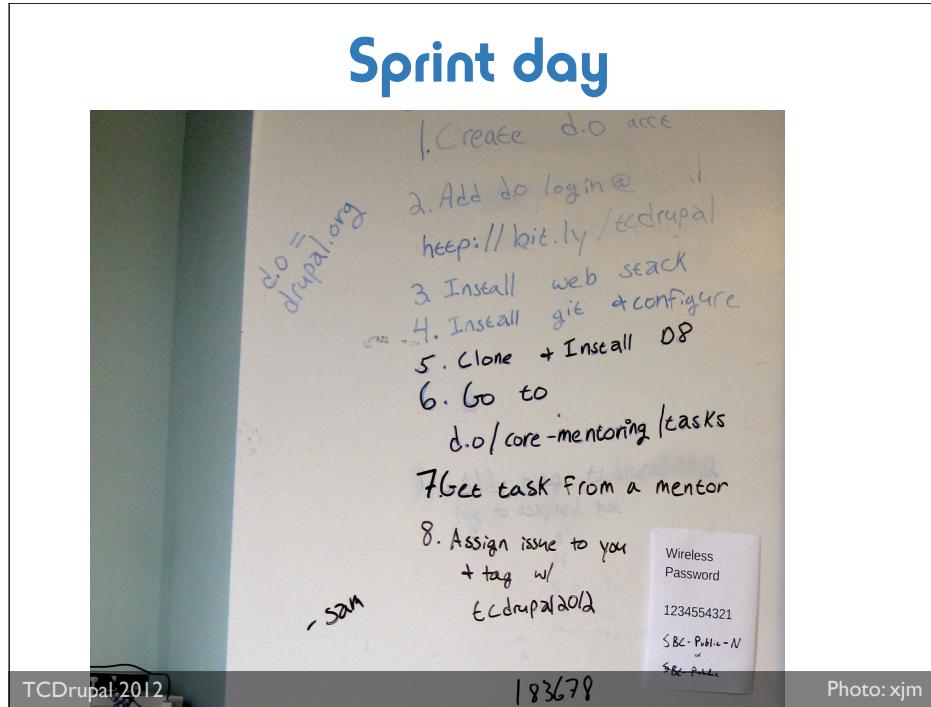
programming.

If you make your own table,
it's helpful to have what this table does:

the task type,

where to find issues (d.o search for status, or needs tags), and
instructions for completing them (like those contributor task doc pages).

Sprint day



[YesCT]

This todo list on a white board is typical of what gets posted on the day of a sprint. It has instructions for participants to:

- . create account on d.o
- . add their username to a google doc
- . install a web stack
- . install git
- . clone and install d8
- . get a task from a mentor
- . assign issue to themself, tag it with their sprint tag
- . it also has the wireless password

In general, on the day,

have signage so people know they are in the right location.

Welcome each person who comes, and introduce yourself.

Post a link to instructions for them (either a doc you wrote up or one from our resources), and include the wifi and password.

Get to know the attendees, asking them questions, and then help them choose tasks that are good matches to them.

Take pictures, and write down who is there.

After the sprint: Task review

<http://core.drupalofficehours.org/tasks-review>

Task level	Task type	Issue priority	Quick review?		
Task	Participant	Mentor	Drupal.org issue	Priority	Status
Create sub-issues for router conversions	jberges	valthebald	1971384 : [META] Convert page callbacks to controllers	major	active
Improve patch docs/code style	friveroll	disasm	1971206 : Cleanup of Book Manager Service	normal	active
Update patch	shanethehat	xjm	1939082 : Convert theme_status_messages() to Twig	normal	needs work
Manual testing	leochid	shyamala	1847314 : Reduce the dependency on javascript for the toolbar to display properly	normal	needs review
Issue summary	kstonef	Kay V	1007746 : Reordering fails with more than 100 items in a menu	major	needs work
Steps to reproduce	no_angel	YesCT	1945226 : Add language selector on menus	normal	active
Issue summary	kstonef	Kay V	1007746 : Reordering fails with more than 100 items in a menu	major	needs work
Triage and find duplicates	capuleto	Cottser	1912474 : Unable to pre-populate db in settings.php	normal	active
Backport	chrisjlee	Cottser	1261846 : Document 1MB maximum size limit for cache_set()	normal	needs work

[YesCT]

Feedback is really important to making new contributors stick around.
 For someone new, getting no feedback makes them feel like their work is going into a black hole.
 Getting timely feedback makes them feel valued.

Often the mentors make a special effort to either do reviews of tasks that happened during their sprint, or recruit people to help do the reviews.

Here is a screenshot of the tasks review page
 on the drupal office hours site
 that is handy if task selection was done using that tool. core.drupalofficehours.org/tasks-review

Otherwise you'll probably have your own list with similar info:
 Task, issue number, participant.

Or, if they tagged tasks on drupal.org with your sprint tag,
 you can use d.o search to generate a list of issues that had tasks worked on from your sprint to give feedback on.

For our Get Involved sprint on Friday, we have a review fest scheduled for one week later,
 Saturday and Sunday, June 1st and June 2nd
 in irc in #drupal-contribute.
 We'll tweet about it. :)

The Review Core Convo tomorrow goes into more detail on giving and getting good reviews on patches. But since a sprint task is not always making a patch. Mentors will give feedback on tasks like posting screenshots, or give feedback on ... a review. So sprint reviews are more focused on giving feedback to the task, and not so much on just reviewing patches.

DrupalCon Portland

...so much left to do!



[ZenDoodles]

Cathy and Steph have done an amazing job herding all the cats to make this sprint awesome, but there's so much left to do. This is going to be the biggest sprint day ever, so if you help out with this sprint, you're in great shape for doing your own sprint as well

DrupalCon Portland

...so much left to do!

Today	4:15 pm	Task entry BoF I in B110
Wednesday	2:15 pm	Development environment BoF
Thursday	10:45 am	Task entry BoF II
Friday	9:00 am	Workshop and mentored sprint
	7:00 pm	Mentor dinner
Next Week	See Cathy	Review fest in IRC



<http://drupal.org/core-mentoring/portland>

[ZenDoodles]

Cathy and Steph have done an amazing job herding all the cats to make this sprint awesome, but there's so much left to do. This is going to be the biggest sprint day ever, so if you help out with this sprint, you're in great shape for doing your own sprint as well

Sprint Day

Friday



[ZenDoodles]

We're expecting this one to be the biggest ever!
We are anticipating 700 sprinters...



[ZenDoodles]

Morning mentors need to show up early. Maybe we should get together for breakfast. See me after the session...



Some sprinters want training

[ZenDoodles] Some will want training...

these are the people new to contributing who don't know where to start. They may not have their development environment set up, or aren't familiar with the community tools. They're sometimes anxious and timid.



Some sprinters want coaches

[ZenDoodles] Some want a more hands-on approach. Usually they've got some experience with Drupal, or even development in another framework. Generally people who understand some of the tools.

They know they want to help, but aren't confident in their ability to find and complete core tasks. They may need a little extra help with things like "the Drupal way", what to do next on an issue, or version control

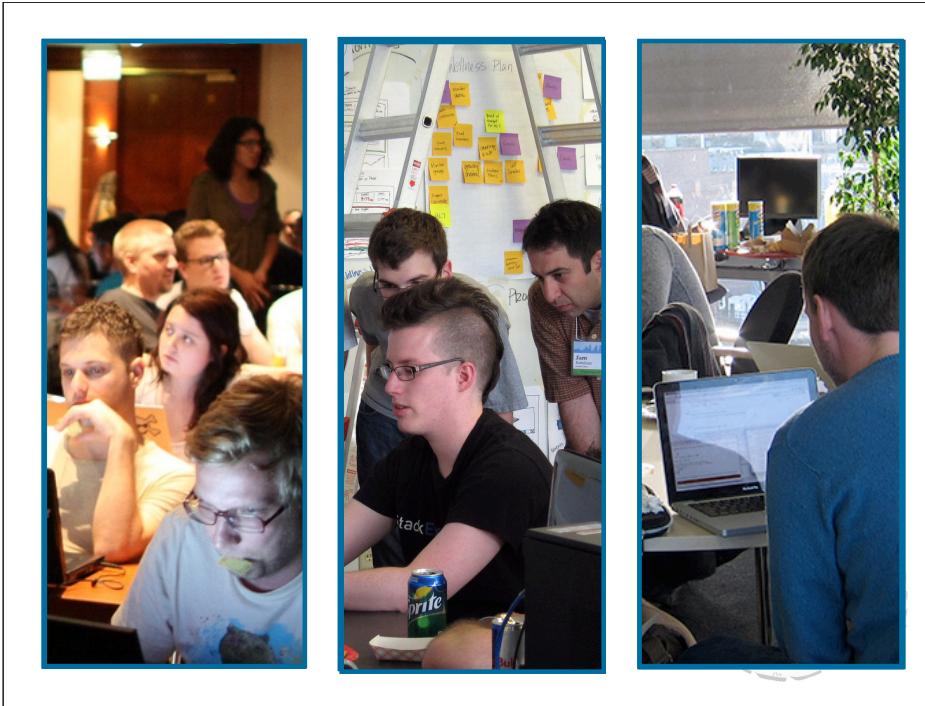


[ZenDoodles] ...and some won't need anything from us.

(I thought about saying some sprinters want cookies...)

You'll probably recognize a few of these haggard warriors. In general, they are exhausted from traveling, a week of giving and/or attending sessions, impromptu (and planned sprints), core conversations, and all the other fun of the con. They may have even been to a party or two along the way.

All they really want is power, internet access, and a place to work with their ~~nefarious henchmen~~ collaborators... Oh, and maybe some coffee and snacks. Pringles and wifi... what more could you ask for?



[ZenDoodles] Three parts, three audiences

Training people go to the Workshop.

The folks who want coaches go to the Mentored Sprint.

Experienced contributors will be in the Main sprint.

What roles mentors can play (see also <http://portland2013.drupal.org/volunteer/sprints>)

- morning and afternoon shifts

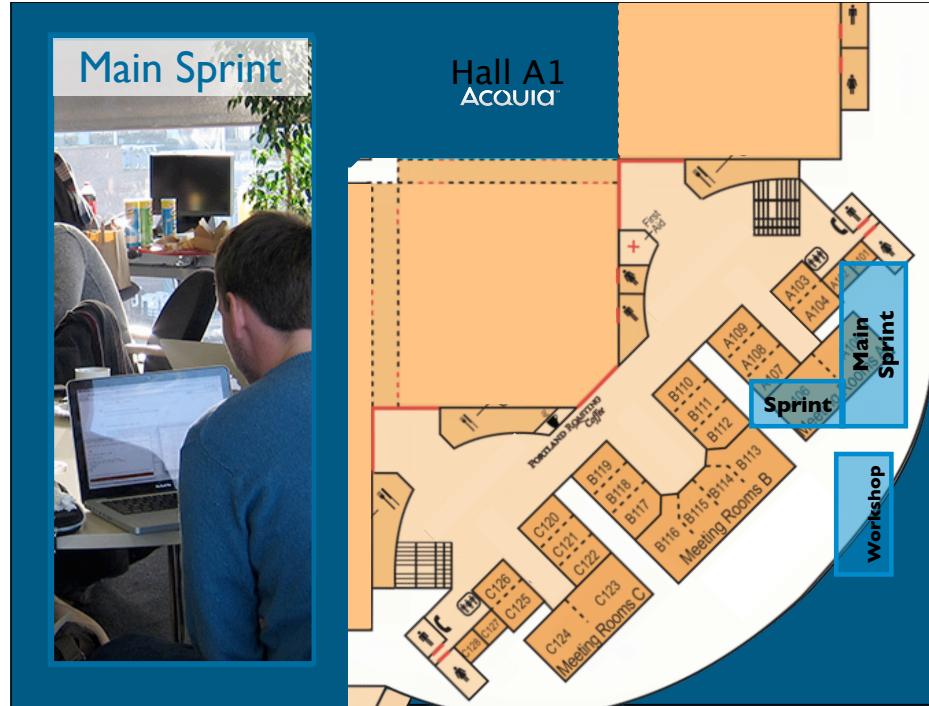
- helping with training

- helping with dev stacks

- helping participants select and work on tasks

- reviewing tasks during and after the sprint

- (also, LFM for BoFs)



[ZenDoodles]

C123-124 COMMUNITY TOOLS WORKSHOP

B117-119 - GET INVOLVED WITH CORE SPRINT

B113-116 - CONTRIBUTION SPRINTS

A105-A106 - SPRINT OVERFLOW

How to be an effective mentor



[YesCT]

You know what to do to plan your own sprint,
you know what's happening this week and
on friday ...

here's how to do it well.

Collaborate



DrupalCon Munich

Photo: Pedro Lozano

[YesCT]

This photo from a drupalcon sprint shows one of the best things at a sprint, two people sitting next to each other at a table, working together!

As a sprint organizer or mentor you might be feeling like you need to handle every question that comes up at your sprint.

But

When I was at Drupalcon Denver, I was not planning the sprint, or even a mentor (yet). But I had trouble with drush in my local dev environment, and so did someone else near me. So we decided to sit at the same table and work on it together. Then, we thought others must be having the same problem too, so we asked if the mentors would announce on the speaker that anyone with trouble with drush come sit at our table: the getting drush to work table. It worked, regular people collaborating to solve a problem.

So, Don't assume that you have to solve every problem by yourself

Other mentors or sprint attendees might have the answer!

At the sprints we've led

there have been several occasions where we couldn't figure out answer, but another sprinter could.

You don't have to know everything--just know who to ask or how to find out.

Expose your reasoning



[YesCT]

Participants think you are magically brilliant... :)

You might be magical or brilliant, ...

show them

they can do what you do.

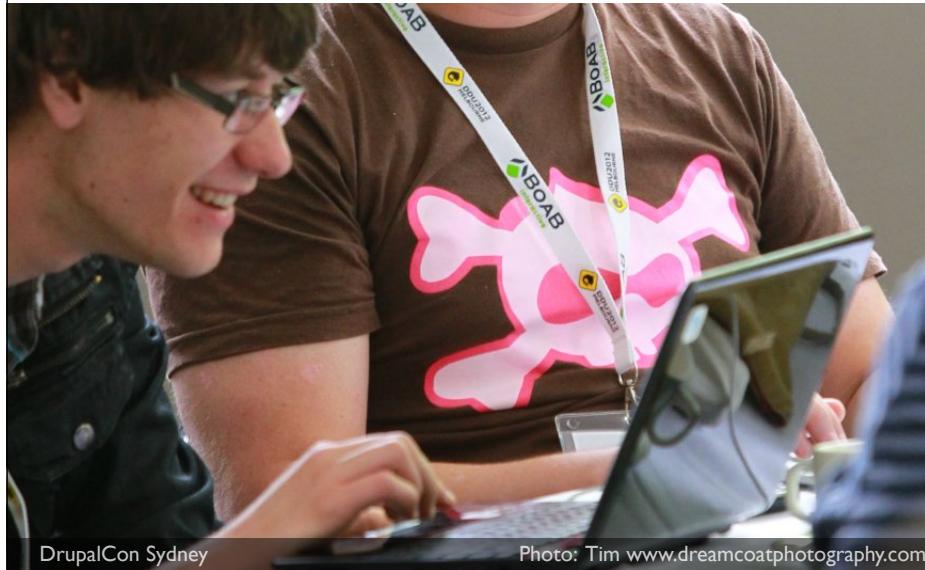
When you mentor, expose your reasoning
by asking participants questions
and revealing your process when you give answers.

Show someone how to look up the answer (api.drupal.org, d.o handbook, contributor task docs, other issues, core gates documentation page, factoids.. ;)

It's not magic, it's knowing where to find the answers.

This builds their confidence that they will be able to do this in the future without you right there.

Match task to participant



DrupalCon Sydney

Photo: Tim www.dreamcoatphotography.com

[YesCT]

A task matched well to the participant will leave them smiling.

Assess their language and communication skills--are they a good candidate for change notices, issue summaries, docs patches? Some love to work on clear wording, **and** have the language skills needed. Others will be better suited for other kinds of tasks.

Are they already familiar with git? Rerolls might work well for them.

Are they a developer at their day job? Give them a small coding change to do to just get used to the core way of making a patch and interdiff.

Give participants a first task that they can succeed at. A simple task!

This will give them confidence to stay engaged and keep trying.

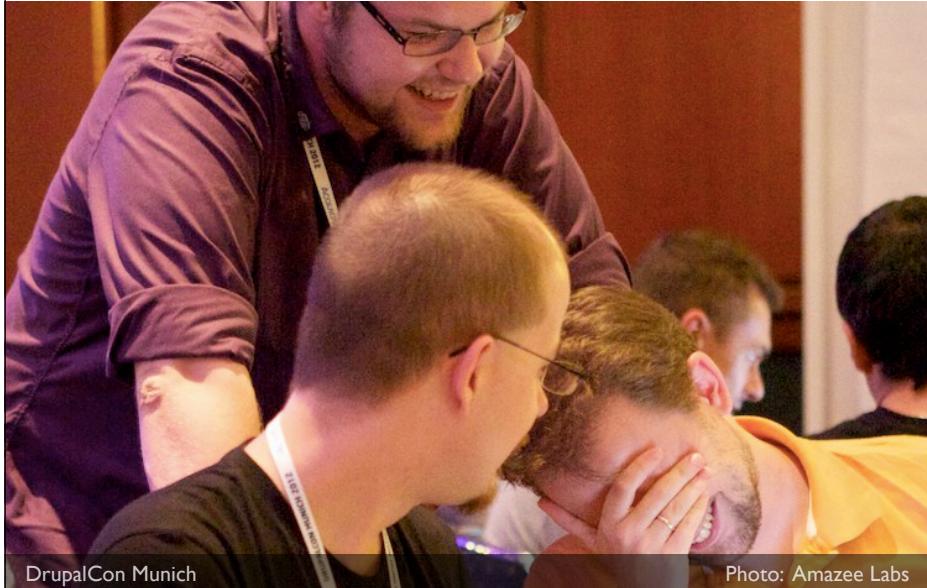
A simple task is the best first task,
even for people with a lot of skills

Scan the issue first, to make sure it's a nice issue,
not controversial or epic.

If the issue was confusing,
move on and find a different task.

Match the task to the participant,
make their first task one they will succeed at,
and they will keep coming back.

Full frontal nicety



[YesCT]

Be nice!

Treat people like peers and remind yourself that you were once new too.

During the sprint, ask people how things are going--
some participants might be heads-down working,
but others might be shy of asking questions.

You approaching them gives them a chance to describe what they are stuck on.

Make sure people understand it's okay to try a different task
if the one they have isn't working for them.

What's next for you?



[YesCT]

What's next for you?

Your mission

- ▶ Help lead a sprint
- ▶ Tell us about it --
@drupalmentoring



[YesCT]

Your mission is to lead a sprint!

Help on friday or
at home, ...
or both! :)

We still have room for mentors, come to the BoF right after this and we will get you set up.

Let us know what you are planning,
if you want help,
and
let us know how it went.

We'll collect data and communicate information from the feedback to help future sprints.

The image is a composite of three screenshots:

- Tasks:** A screenshot of a task management interface. It has two dropdown menus at the top: "Task level" set to "Any" and "Task type" set to "Any". Below is a table titled "Priority tasks" with columns "Available" and "Task". The tasks listed are:

Available	Task
Yes	Manual testing (see task note)
Yes	Draft change notification
No	Create patch
Yes	Draft change notification
Yes	Expand test coverage
No	Improve issue summary
Yes	(advanced) Test installing Drush from a DB dump & then upgrade
Yes	Toolbar performance testing
Yes	Write tests
Yes	CMI testing
Yes	Draft change notification
Yes	Draft change notification
- Dev Env:** A photograph of a person's hands working on a laptop keyboard.
- Tools:** A screenshot of the Drupal Community page. The header says "Drupal™" and has links for "Drupal Homepage" and "Your Dashboard". The main content area is titled "New contributor tasks" with a "View" button. Below it is a list of bullet points:
 - Contributing makes you feel good
 - Your contributions are valued
 - Many of the tasks are fun to do
 - The more you contribute, the more you learn
 - Contributing makes you a more interesting developer

In this section (as it grows), you will hopefully find tasks that are quick (about an hour or less) and help you get more deeply involved.

[YesCT]

let's review.

Remember these three things:

Prepare a task list a few days before your sprint

Ask your participants to set up their dev environment in advance and have tasks on hand for people that don't have a dev env

Remember to use the contribution mentoring tools like the drupal office hours site and the contributor task documentation pages, they are so awesome!

Resources

- ▶ [#drupal-contribute](#)
- ▶ [@drupalmentoring](#)
- ▶ portland2013.drupal.org/node/2433
- ▶ drupal.org/core-mentoring/sprint-resources
- ▶ drupal.org/core-mentoring/portland



[YesCT]

follow @drupalmentoring for reminders about events this week.

Our slides are available on the portland drupalcon site. The session evaluation is there too. node 2 4 3 3

And, it also has a link to drupal.org/core-mentoring/sprint-resources

which is the handout of all the links and resources we have mentioned

We have extras of the post cards here with the mentoring schedule that were in your conference bag..
And that same schedule is online at drupal.org/core-mentoring/portland



[YesCT]

Picture yourself mentoring at a sprint. The chair in this photo is waiting for you.

Now: questions!

(Or if short on time)

We can address any questions at the BoF right after this which is in room B110.

[go to the thank you slide after ~2 mins]

Thanks

- ▶ Angie Byron
- ▶ Tim Plunkett
- ▶ Scott Reeves
- ▶ Kay VanValkenburgh
- ▶ Roger Goode
- ▶ Neil Kent
- ▶ Stephanie El-Hajj
- ▶ Megan Sanicki
- ▶ Holly Ross
- ▶ Pedro Lozano
- ▶ Kim Pepper
- ▶ Donna Benjamin
- ▶ drupalize.me
- ▶ The Nerdery
- ▶ Acquia
- ▶ Drupal Association
- ▶ And all of you!



[YesCT]

[go to the feedback slide after ~2 mins]



[YesCT]

Thank you all!

Please fill out the evaluation form for this session.

It's really important!

Mentor BoF right now in B110.