DRUPALCON LONDON
Web Services and Context
Core Initiative

Presented by Larry Garfield
@Crell

- Senior Architect, Palantir.net
- Web Services & Context Initiative Owner
- Architectural Gadfly
- Nerf Gunslinger
Web Services and Context Core Initiative (WSCCI)

http://www.flickr.com/photos/spaterson/4995021726/
"Drupal doesn't have multiple layouts. It has one layout it bends a lot."

--Earl Miles, c.2009
Five Fails

1) No unified context
2) One Layout to Rule them All
3) No plugin mechanism
4) HTML-only
5) Performance
The Butler Project

"Give Drupal a unified, powerful context system that will support smarter, context-sensitive, easily cacheable block-centric layouts and non-page responses using a robust unified plugin mechanism."
DrupalCon Chicago

Recast as

Web Services and Context Core Initiative
WSCCI Goals

The Web Services and Context Core Initiative (WSCCI) aims to transform Drupal from a first-class CMS to a first-class REST server with a first-class CMS on top of it.

To do that, we must give Drupal a unified, powerful context system that will support smarter, context-sensitive, easily cacheable block-centric layouts and non-page responses using a robust unified plugin mechanism.
1) Unified, powerful context system

2) Unified plugin mechanism

3) Any type of response

4) Smart block-centric layouts
<Insert lots of discussion here>
#0
HTTP
HTTP libraries

• Don't reinvent the wheel.
• Tried PECL HTTP... Nyet
• 3rd party research
  • Dave Hall (skwashd)
  • Dick Olsson (dixon_)
• Zend vs. Symfony2: FIGHT!
  • http://groups.drupal.org/node/167299
Status

Dries OKed this last night at 2 am.

Patch forthcoming...
#1

Context
Mediator Object

The mediator pattern provides a unified interface to a set of interfaces in a subsystem.

--Wikipedia
Work in progress

http://drupal.org/project/butler

... And core sandbox by the end of the week
class Foo {
    protected $context;

    public function __construct($context) {
      $this->context = $context;
    }

    public function behave() {
      $node = $this->context['node'];
      $var = $this->context['http:get:var'];
      // ...
    }
}
function behave() {
    $context = drupal_get_context();
    $language = $context['language'];
    $foo = $context['http:header:foo'];
}
$butler->registerHandler('http:get', 'CtxHandlerHttp', array('query' => 'GET'));

$butler->registerHandler('http:post', 'CtxHandlerHttp', array('query' => 'POST'));

$butler->registerHandler('http:header', 'CtxHandlerHeader');
$new_ctx = $butler->addLayer();

$new_ctx->registerHandler('og', 'AlternateHandler');
$new_ctx['og:nid'] = 5;

$new_ctx->lock();
$plugin = get_plugin('stuff', 'blah', $new_ctx);
Status

* Nearly ready to try implementing in core

* Come to the code sprint Friday and let's see how far we get!
#2 Plugins
We have too many

- One-off hacks
- Alt. include files
- Callback functions
- Pseudo-hooks
- Caching system with OO
- Views Plugins/Handlers
- Ctools Plugins
- Webform components
- Etc.
Research

- http://groups.drupal.org/node/137609
- Symfony2
- Zend Framework 2
- Kohana
- ctools.module

- No one is as good as we want to be...
Status

- ctools TNG
- All OOP
- Two skunkworks implementations
  - Crell
  - nclimdul
- After context is in core...
#3
REST Routing
More robust than hook_menu

- 2 layer routing
  - Method Path, domain, HTTP content-type
  - "Other stuff" - Implementation TBD
- Maps to response object, not dumb function
- Any HTTP return (404, 302, etc.)
- Any content type, natively (text/html, text/json, application/pdf)
- http://groups.drupal.org/node/148149
#4
Smarter block layout
Figure out the UI first…

http://groups.drupal.org/node/137469
Next steps...

Code sprint Friday!

Discuss here:
http://groups.drupal.org/wscci

Follow here:
http://drupal.org/node/1233232