CiviCRM for Drupal Developers
Coleman Watts

- CiviCRM user since 2010
- Found Civi through Drupal
- Implemented Civi for Woolman School
- Author of Webform Integration, and a few other CiviCRM modules and APIs
Matt Chapman

- Ninjitsu Web Development, Los Angeles
- CiviCRM User and Contributor since 2007 (CiviCRM 1.7)
- Found Drupal by way of CiviCRM
- Author of a dozen or so Drupal+Civi Integration modules
- Sponsor of CiviCRM events & co-teach training
Who are you?

- Drupal and/or CiviCRM Users?
- Implementer / Configurations?
- Developers?
  - Drupal Developers?
  - API Cooking?
- Managers?
- Drop outs?
- Anonymous?

VoicesOfSanDiego.org
Why CiviCRM is hard for Drupal Devs

- Different Theme layer (Smarty)
- Different Database Layer (PEAR:DB)
- Different Forms Library (Quickform)
- Different Javascript (Dojo -> own jQuery)
- Different permissions system (ACLs)
- Different Classification systems (no Taxonomy module)
- Different User Data Collection (Profiles)
- Different Menu System
- Different Mail system
- Different Wysiwyg Philosophy (ships own CKeditor)
- Different Version Numbering & Support. (CiviCRM 2.0 was the last version for Drupal 5. CiviCRM 2.1 was for Drupal 6.0. Civi 4.0 -> Drupal 7.0, CiviCRM 4.1 -> D6 & D7)
- Different Community (not hosted or supported on Drupal.org)
CiviCRM APIs

● Avoid the deep dark past of v1 and v2.
● APIv3 for CiviCRM 4 is awesome! (Even if they haven't synced the version numbers yet...)
● Many Improvements
  ○ More consistent parameters
  ○ 5 gateways: PHP, AJAX, REST, Smarty, Drush
  ○ Test coverage
  ○ Code generator! /civicrm/ajax/doc
    ■ Demo: get - list, search, single
    ■ getFields
    ■ json/XML
API Demo

**PHP**

```php
require_once 'api/api.php';
$result = civicrm_api('Contact', 'Get', array(
    'first_name' => 'Michael',
    'last_name' => 'McAndrew',
    'version' => 3
));
foreach ($result['values'] AS $contact) {
    // do something
}
```

**Javascript**

```javascript
$.crmAPI('Contact', 'get',
    {
        'first_name' : 'Michael',
        'last_name' : 'McAndrew',
        'version' : '3',
    },
    { success:function (data){
        $.each(data, function(key, value) {
            // do something
        });
    }

```}

**REST**

https://example.org/path/to/civi/codebase/civicrm/extern/rest.php?
q=civicrm/login&name=user&pass=password&key=yoursitekey&json=1

```json
{"is_error":0,"api_key":"per-settings.php","PHPSESSID":"4984783cb5ca0d51a622ff35fb32b590","key": "2e554f49c9fc5c47548da4b24da64681b77dca08"}
```

http://example.org/civicrm/ajax/rest?fnName=civicrm/contact/search
&json=1&key=YourSiteKey&PHPSESSID=4984783cb5ca0d51a622ff35fb32b590
CiviCRM Hooks

- Pre/Post hooks

```php
function example_civicrm_post($op, $objectName, $objectId, &$objectRef) {

  if ($objectName == 'Individual') {
    switch ($op) {
      case 'create':
        //We just created a new Individual Contact Record as $objectRef
        break;
      case 'edit':
        //We just edited the Individual Contact Record $objectRef
        break;
      case 'delete':
        //The $objectRef Individual Contact Record has been deleted
        break;
    } //end switch
  } //end if
} //end function
```
$op - operation being performed with CiviCRM object. Can have the following values: view, create, edit, delete.

$objName - can have the following values:
'Individual', 'Household', 'Organization'
'Address'
'Group', 'GroupContact'
'Relationship'
'Contribution'
'Profile' (while this is not really an object, people have expressed an interest to perform an action when a profile is created/edited)
'Membership'
'Event'
'Participant'
'UFMatch' (when an object is linked to a CMS user record)
'PledgePayment'
'ContributionRecur'
'Pledge'
'CustomGroup'
## More Hooks

### Form related hooks
- `hook_civicrm_validate`
- `hook_civicrm_buildForm`
- `hook_civicrm_postProcess`

### More Database related Hooks
- `hook_civicrm_custom`
- `hook_civicrm_copy`

### GUI related hooks
- `hook_civicrm_links`
- `hook_civicrm_dashboard`
- `hook_civicrm_xmlMenu`
- `hook_civicrm_navigationMenu`
- `hook_civicrm_pageRun`
- `hook_civicrm_customFieldOptions`
- `hook_civicrm_summary`
- `hook_civicrm_summaryActions`
- `hook_civicrm_searchTasks`
- `hook_civicrm_searchColumns`
- `hook_civicrm_buildAmount`
- `hook_civicrm_tabs`

### Even More Hooks
- `hook_civicrm_tokens`
- `hook_civicrm_alterMailingLabelParams`
- `hook_civicrm_alterPaymentProcessorParams`
- `hook_civicrm_tokenValues`
- `hook_civicrm_mailingGroups`
- `hook_civicrm_contactListQuery`
- `hook_civicrm_membershipTypeValues`
- `hook_civicrm_alterMailParams`
- `hook_civicrm_caseSummary`
- `hook_civicrm_config`
- `hook_civicrm_merge`
- `hook_civicrm_export`
- `hook_civicrm_emailProcessor`
- `hook_civicrm_import`
- `hook_civicrm_alterAPIPermissions`
- `hook_civicrm_dupeQuery`
- `hook_civicrm_buildStateProvinceForCountry`
The Easiest Way to Customize CiviCRM?
The Easiest Way to Customize CiviCRM?

- Don't!
- Write a Drupal module using the tools you already know and the API.
- Views Integration, CiviCRM Reference Field
- Better still, write a generalized Contrib module.
Case Study: Webform Integration
Groundhog Day

- Create custom module
- Use FAPI to build a form
- Use CiviCRM API to get/save data
- Test, Fix bugs
- Maintain module for the rest of your life
- And if you need another form...

See you tomorrow!
Open-Sourcing It

- Abstracted vs. single-purpose module
- Based on existing software
- Webform - one of Drupal's top 10 contrib modules
- Open source = community support
- Strengthens CiviCRM
How it works

- `hook_menu()`
- `drupal_set_message()`
- `hook_form_alter()`
- `drupal_add_js()`
- Webform submission hooks
- CiviCRM API
- CiviCRM database
- Drupal database
- `hook_civicrm_merge()`
What It Does

- Multiple contacts/types
- Relationships
- Custom data
- Multiple locations
- Tags & Groups
- Case & Activity
- Event Registration
- User messages
- Deduping, checksums
- Webform Features too

I'm outta here!
Tips & Ideas

- Dozens of add-ons for webform
  - Webform Validate
  - Webform Conditional
  - Webform Layout
  - Webform Autocomplete
  - Chosen

- Hidden Fields
  - Use field key to change a field to another type

- Shared addresses
  - Limit to one choice to make a nice checkbox
  - Exposed address fields will auto-hide for you

- IDs and Checksums
  - Create a hashed link or redirect with webform tokens
Taking it Further

- CiviCRM Hooks
  - pre and post
- Drupal Hooks
  - hook_form_alter
- Webform Hooks
  - presave, create, edit, delete
  - create your own type of form element
- Contribute it back!
  - Submit patches
  - Sponsor development
  - Release your own module
- Field Storage Engine: The Next Generation?

+ Drupal + CiviCRM + YOU!
Questions?

Matt Chapman
818-660-NINJA
support@NinjitsuWeb.com

Coleman Watts
colemanw@woolman.org
IRC: colemanw
drupal.org/webform_civicrm