Drupal Workshop, Part 4

Content Types, Fields & Views

TECH BRIEFING

Instructor's Name: Sharon L. Krossa

Instructor's Email: skrossa@stanford.edu

Instructor's URL: http://drupaltraining.stanford.edu

My URL: ______________________

STANFORD
UNIVERSITY
INFORMATION TECHNOLOGY SERVICES

techtraining.stanford.edu • 650.723.4391

This handout accompanies classroom instruction provided by IT Services Technology Training, Stanford University. While it is not intended as a stand-alone tutorial, it may be helpful in reviewing the topics covered in class.

STANFORD
UNIVERSITY

INFORMATION TECHNOLOGY SERVICES

Contents

These instructions are also available at:

`http://drupaltraining.stanford.edu/workshop4/handout`

This page is intentionally blank.

Introduction to Content Types, Fields (CCK) and Views

Today we're going to create a To Do List and Change Log, something every site should have. Making them will involve

- Creating & customizing content types
- Using the contributed module Views to create lists of information from multiple nodes
- Using the contributed module Content Construction Kit (CCK) to add custom fields

Drupal sites are basically just things and lists of things.

By default, Drupal comes configured with two different kinds of things, Page (which we renamed Basic Page) and Story, but you can add custom kinds of things, or custom "Content Types".

By default, Drupal only comes with one way to list things. You can promote a thing (a specific Basic Page or Story – which are generically known as "nodes") to the front page, where it will be listed in reverse chronological order with other things promoted to the front page.

Happily, there is a module called "Views" that lets you make lists of almost any things and display them almost anywhere in the site.

For those familiar with relational databases:
- A node (a specific Basic Page, Story, etc. content) is like a database record (a thing)
- A content Type (Basic Page, Story, etc.) is like a database table (a kind of thing)
- A view is like a database query result (a list of things)

There are at least three steps to creating a new content type:

- Create the new content type
- Set permissions for the new content type
- Set whether Drupal will display the author and creation date for the content type

When using certain contributed modules, there are also additional steps involving them. This site doesn't use any of those (yet).

I. Create a new content type: Change entry

1. Go to **Administer->Content management->Content types**
2. Click on **Add content type** tab
   - **Name** is the human readable name of the new content type. It can contain capital letters, spaces, punctuation, etc. We're going to call our new content type `Change entry`.
   - **Type** is the computer name for this content type. It can't contain capital letters, spaces, or any punctuation except underscores.
     So we'll enter `change_entry`.
   - **Description** is what will appear in the admin list of content types (the page we were just on) and on the Create content page `(node/add)`. So it should explain to users what this content type is for, so they'll know when to use it (as opposed to one of the other content types).

     `Change entry is used both to add tasks that need to be done for the site (changes to do) and to record changes that have been made to the site (changes done).`
3. Expand **Submission form settings**
4. Change the **Title** field label to `Change`
5. Change the **Body** field label to `Details`
6. Expand **Workflow settings**
7. Under **Default options**:
   - Check/enable **Published** (so, leave as is)

Note: With the site's current settings, if it were not off-line, this would mean the whole world could see these entries. In another workshop we'll cover access control and what settings to use to restrict access to Change entries to only site staff.

   - Uncheck/disable **Promoted to front page**
   - Uncheck/disable **Sticky at top of lists** (so, leave as is)
   - Check/enable **Create new revision**

Note: Whenever a new content type is created, **Create new revision** *should be turned* **ON**. *Without revisions, you can't simply "undo" content changes.*

   - Expand **Comment settings**
   - Leave Default comment setting set to **Read/Write**
     It can be useful to allow site staff to add comments or questions to tasks or completed changes.
   - Set Default display order to **Date — oldest first** (We want any comments to read like a conversation.)

- Leave **Comment subject field** set to **Enabled**
  Otherwise, Drupal will use the first part of the first sentence of the comment as the subject.
- Set **Preview comment** to **Optional**. Because change entries will be used by site staff, previewing posts shouldn't be necessary. If comments were going to be posted by, say, any SUNet user, we probably would want to require preview before posting, as a safeguard for naive posters unfamiliar with the system.
- Click the **Save content type** button

II. Set permissions and options for Change entry

1. Go to **Administer->User management->Permissions**
2. In the **node module** section, enable these permissions for the Zite Admin role:
   - **create change_entry content**
   - **edit any change_entry content**
   - **edit own change_entry content**

   Remember, leave all "delete" permissions disabled for all roles, to make site staff have to think really hard before deleting content. Once deleted, content is gone and can only be restored by restoring the entire database, which, in turn, means losing any changes made since the last backup.

   ---

3. Click the **Save permissions** button

IIa. Set whether Drupal will display the author and creation date for the content type

1. Go to **Administer->Site building->Themes**
2. Click on the **Configure** tab, which takes you to **Global settings**
3. In the **Display post information** section, make sure **Change entry** is enabled/checked
   For some content types you won't want to display the author and creation date, but for a To Do List/Change Log entry, that is relevant information.
4. Click on the **Save configuration** button.

IIb. Make a few sample Change entries.

For each one, either go to
Administer->Content Management->Create content->Change entry
or else click on the **Create content** link in the navigation menu on the left, then click the **Change entry** link.

Then fill out the form. Don't forget to click the **Save** button!

III. Install and configure the Views module

Zite Admins can go to **Content management->Content** to see a list of all the content on the site, but this isn't available to regular users, and also isn't all that user friendly. What we really need is to create a custom list of these things. To do that, we need the contributed module **Views**.

As a reminder, there are four steps to adding a contributed module to a Drupal site:

- **Install** the module
- **Enable** the module
- **Set permissions** for the module
- **Configure** the module

For the Views module, these four steps look like this:

Install the module

This has already been done by the Collaboration Tools Installer

Enable the module

1. Go to **Administer->Site building->Modules** `(admin/build/modules)`
2. In the **Views** section, enable (check) **Views** and **Views UI**.
3. Scroll down to the bottom of the page and click the **Save configuration** button

Set permissions for the module

1. Go to **Administer->User management->Permissions** `(admin/user/permissions)`
2. Scrolling through the list and/or using the browser's Find/Search, we find **Views** module at the bottom of the page.
3. Enable (check) **access all views** and **administer views** for the Zite Admin role.

   Note: Access all views means that that role can see any view on the whole site regardless of any access control restrictions on those views. Only Zite Admins should be given this permission.

4. Scroll down to the bottom of the page and click the **Save permissions** button

Configure the module

1. Go to **Administer->Site building->Views** `(admin/build/views)`
2. Click on the Tools tab
3. Most of these just affect what site staff see when creating views, but a few can affect performance. Remember these settings are here, but for now just enable/check **Ignore missing advanced help module**.
4. Click the Save configuration button

The Views module is now installed!

IV. Create a new View

1. To create a new view, click on the **Add** tab.
    - **View name** is the computer name for this view. It can contain only letters, numbers, and underscores. We'll call our view `change_list`.
    - The **View description** appears in the list of views that admins see. It doesn't need to be long, but should be descriptive. We'll use: `Display a To Do list and change log for site`.
    - The **View tag** is optional, but I've found it handy to tag views I create as belonging to the site, so I can easily tell which views came from some module or other and which are custom. So enter `sudwork` (for "Stanford University drupal workshop)
    - Views can display lists of information from or about nodes (that is, the actual content of the various content types), comments, users, and various other stuff. In this case, we want to list things from **nodes** (namely, the content of the Change entries).
2. Click the **Next** button.

Setting Filters

There are a lot of settings here, but you don't need to know what they're all for at this point. We're just going to start with the Filters section.Filters lets you choose which things to include in your list of things.

The first thing you want to do for any new view is make sure that only nodes a user has access to will be included.

1. Click on the **+** button next to **Filters**
2. Scroll down to **Node: Published or admin** (you can get there quicker by selecting Node in the Groups dropdown list)
3. Check/enable **Node: Published or admin**
4. Click the **Add** button.

    Note: You should make adding Node: Published or admin to the filters the very first thing you do every time you create a new view. It should be a habit indelibly burned into your brain!

5. Next, add a filter selecting only Change entries. Click on the Filters **+** button, check/enable **Node: Type**, and click **Add**.
6. On the **Defaults: Configure filter Node: Type** screen
    - For **Operator**, select **is one of**
    - Under **Node type**, enable/check **Change entry**
    - Click the **Update** button.

Note the error message:
Display "Defaults" uses fields but there are none defined for it or all are excluded.
So we need to define a field, which is done in the Fields section.

Setting Fields

1. Click the **+** button next to **Fields**.
2. Check/enable **Node: Title**, then click the **Add** button
3. On the Defaults: Configure field Node: Title screen:
    - Remove "Title" from the Label field (leave it empty)
    - Check/enable **Link this field to its node**
    - Click the **Update** button.
      See how under Live preview, there is now a list of our Change entry titles? Live preview lets us check that our view is behaving as expected even before we save it.

    Note the various indicators that we haven't saved this view yet. The things we added are highlighted, there is orange text saying "New View".

4. Click the **Save** button

Setting up a Page display

So, we have a View, but right now it isn't being shown anywhere except in the Live preview. To have the list show up as a web page on the site, we need to add a **Page display.**

1. **Page** is already selected in the dropdown menu under the Defaults tab.
2. Click the **Add display** button.
3. Try clicking the **Save** button. We get an error message:
   Display "Page" uses a path but the path is undefined.
4. To define a path for our page, click the **Page** tab below the **Defaults** tab.

   Note: every time you save, the display goes back to the Defaults tab. Make it a habit to always check whether you are in the right display tab before changing settings in Views!

5. In the **Page settings** section, click the link next to **Path**.
6. Type **`changes`** into the URL field
7. Click the **Update** button.
8. Click the **Save** button
   Now it saves without complaint.
9. Click the **View Page** link.

There's our view!

So now let's make various modifications to the view.

Modifying the View

In **Basic settings** section:

1. Change the name of the page display ("Page" isn't very helpful) to **`Changes page`**
2. Change the title to: **`Changes`**

In **Sort criteria** section:

3. Add sorting by **Node: Updated date**

V. Install and configure CCK (Content Construction Kit)

It would be nice to be able to distinguish **Changes To Do** from Changes that are **Done**. There are a number of ways this can be done, but for this exercise we're going to add a new field to the content type. For this we need to install the CCK module.

As a reminder, there are four steps to adding a contributed module to a Drupal site:

- **Install** the module
- **Enable** the module
- **Set permissions** for the module
- **Configure** the module

For the CCK module, these four steps look like this:

Install the module

This has already been done by the Collaboration Tools Installer.

Enable the module

1. Go to **Administer->Site building->Modules** `(admin/build/modules)`
2. In the CCK section, enable (check) these sub-modules:
   - **Content**
   - **Fieldgroup**
   - **Node Reference**
   - **Number**
   - **Option Widgets**
   - **Text**
   - **User Reference**
3. Scroll down to the very bottom of the page and click the **Save configuration** button.

Set permissions for the module

1. Go to **Administer->User management->Permissions** (`admin/user/permissions`)
2. Scrolling through the list and/or using the browser's Find/Search, we find **content module** at the bottom of the page. (CCK is often called Content rather than CCK in the Drupal admin interface)
3. The only permission to be set is one for using PHP, so we leave it **disabled/ unchecked** for all roles.

Since we didn't change anything, no need to click the **Save permissions** button

Configure the module

1. A look through the various Admin menus doesn't reveal anything likely, but to confirm that there are no settings, go to to **Administer->Site configuration->By module** (`admin/by-module`)
2. Find the **Content** section (again, CCK is often called Content rather than CCK in the Drupal admin interface)
3. As suspected, there is nothing to set except permissions, which we've already done.

The CCK module is now installed!

VI. Add fields to your Content Type

1. Go to **Administer->Content management->Content types->Edit Change entry**
2. Click on the **Manage fields** tab.
3. Scroll to the **New field** section of the **Add** section.
    - The **Label** is the human readable name of the new field, so it can have capital letters, spaces, etc. We'll label our new field **`Change type`**.
    - The **Field name** is the computer name of the field, so it must be only lowercase letters, numbers, and underscores. Type in **`change_type`**, to give it the Field name `field_change_type`.

Note: While the Label can be changed later, the Field name cannot be changed, so pick the Field name carefully. Keep in mind that a CCK field can be added to more than one Content type, so think about what, if any, other content types might end up using a field when naming it.

---

4. Select **Text** from the **Select a field type** drop down menu.
5. Select **Check boxes/radio buttons** from the **Select a widget/Form element** drop down menu.
6. Click the **Save** button

Settings for each Field

For each CCK field, there are both content type specific and global settings.

CHANGE ENTRY SETTINGS

The **Change entry settings** will only affect the **Change type** field as it is used in the **Change entry** content type. These settings can be different for this field if used in other content types.

In the **Help text** box, type:

```
Enter whether this is a change To Do or an already Done change.
```

GLOBAL SETTINGS

Global settings will apply not only in the **Change entry** content type but also every other content type in which it is used.

Note: So when planning content types and fields, and deciding whether to use the same field in two content types or create a separate field for each content type, keep in mind that whether a field is required and how many values it can have, etc., are global settings. (This means you can't use the same field in two content types if, say, you want the field to be required in one content type but optional in the other, or have only one value in one content type but allow multiple values in another, etc.)

For our **Change type** field:

1. Enable/check **Required**
2. Leave **Number of values** set to 1
3. Leave **Text processing** set to **Plain text**
4. In the **Allowed values** list, enter (one value per line)
   - `To Do`
   - `Done`
5. Click on the **Save field settings** button
6. **Change type** is now in the list of fields, but it is at the bottom, and we want it to come higher up on the form when people create Change entries. So grab the crossed arrows next to **Change type** and drag it up between **Menu settings** and **Details**.
7. Click the **Save** button.
8. Click the **Display field**s tab.
   This is where we configure how Change entry fields will display in nodes (thing pages) and in views (lists of things pages).

   For the Change type field, we want these settings:
   - Label = **Inline**
   - Teaser = **Default**
   - Full node = **Default**
9. Click the **Save** button

We already created three Change entries, so now we need to go back and edit each of those things to add their **Change type** information.

The easiest way to do this (with our current site configuration) is to go to the **Change list display page** we created (**changes**), then go and edit each Change entry in turn.

VII. Add the new fields to your View

It would be nice if the Changes list indicated which items were **To Do** and which were **Done**.

1. We can do this by editing the view to add **Content: Change type (field_change_type)** to the fields, with settings **Hide if empty** enabled and **Label** set to **None**.
2. Since we want this to apply not just to the page display but also to the defaults, click the **Update default display** button.
3. Next, click the **gear icon** next to **Row style: Fields** in the Basic settings section.
4. We want the Change type to display on the same line as the Change title, so under **Inline fields,** check/enable both **Node: Title** and **Content: Change type (field_change_type)**.
5. For the **Separator**, enter a hyphen with a space before and a space after. (**-**)
6. Click the **Update default display** button.
7. In the **Fields** section, click on the **up/down arrows** icon.
8. Grab the crossed arrows icon next to **Content: Change type (field_change_type)** Default and drag it to the top of the list, above **Node: Title**. (This changes the order the fields will be display in.)
9. Click the **Update** button
10. Click the **Save** button.