

Theming Omeka

Default Theme Files

Explain the default theme files in Omeka's core. These are the theme files that are used if a theme does not supply an overriding file, following the same directory path structure.

Located in Omeka/application/sites/views/scripts

- collections: theme files for displaying Omeka collections
- common: templates used across all theme files (like header.php and footer.php)
- css: CSS files
- error: error pages
- file: theme files for displaying files
- images: jquery-ui and other icons used when displaying images
- items: theme files for displaying Omeka items
- javascripts: jquery and tinymce files
- upgrade: pages used during Omeka's upgrade process
- users: pages used during registration and (public) login

Most important files are usually

- show.php -- used when displaying a single thing (collection, item, etc.)
- browse.php -- used when displaying several of a thing

Activities

- Copy the example-theme theme into your themes directory

From Scratch Theme

Explain how to use From Scratch to jump-start your theme development

- Contains files and folders for collections, items, header and footer, css, and more to get you up and theming quickly.

Activities

- Download From Scratch and look through style.css

Override the items/show page

Demonstrate the typical functions, CSS classes, and structures used in a show page

[Functions](#)

- `__()` : translates text. Best not to change the text, as translations will break!
- `head()` : data passed to the header.php file for display
- `get_current_item()` : makes sure there is an item for `item()` to work on
- `item()` : displays element set data
- `custom_show_item_metadata()` : in defaults--can't override! shows ALL the metadata

Changes coming in Omeka 2.0 (Fall 2012)

- `custom_show_item_metadata()` will go away. Unsure what, if anything, will replace it (May 20, 2012)

Activities

- Move the pagination links to the top of the page

- Highlight the item description by putting it prominently under the title
- Customize the metadata shown by getting rid of `custom_show_item_metadata` (some themes let you do this in their configuration settings)

Override the items/browse page

Demonstrate the typical functions, CSS classes, and structures used in a browse page

Copy items/browse into example-theme

- `loop_items()` : used in a `<?php while(); ?>` structure to loop through items
- `item_tags_as_string()` : echo out the tags
- `item_has_tags()` : checks if an item has tags using a `<?php if(): ?>` structure
- `link_to_item()` : creates a link to the current item
- `custom_nav_items()`

Omeka's loop record structure

Explain Omeka's looping of records (similar to WordPress')

Activities

- Add an additional metadata field (e.g., identifier) to the display for each item
- Add navigation links to Home by getting rid of the call to `custom_nav_items()`

Override the header/footer

Demonstrate the typical functions, CSS classes, and structures used in header and footer

- `settings()` : get settings set in the site configuration
- `queue_css()` : use before `display_css()` to queue CSS files to be added
- `queue_js()` : use before `display_js()` to queue javascript files to be added

Activities

- Copy 'common' folder into example-theme
- Add in new stylesheet
- Add in a new javascript file

Override the front page

Plugin hooks

Explain the plugin hook system, and why it is important for themers

[Hooks](#)

Functions beginning with `plugin_` provide mechanisms for plugins to add content to pages. Not including theme in your theme may cause plugins to lose functionality. Usually, it is safe to move them around though.

Hook `public_theme_header()` is fired by `plugin_theme_header()` in the theme. Similarly for all hooks.

Theme Configuration

Explain the config.ini file, and how to retrieve settings made there in your themes

Omeka has a robust mechanism for theme designers to easily create a configuration page for your themes. This happens via a `config.ini` file for the theme, and by calling

get_theme_option() appropriately in your theme files.

config.ini file and the theme config page

//screenshots of theme config page and the file

The theme config.ini page lets you give instructions to Omeka about how to put together the theme config page. For each option that you want to create, it needs:

- input type
- label
- description
- optional settings specific to the input type
- optional validation settings

When using get_theme_option(), logo in config.ini becomes get_theme_option('Logo'). background_header becomes get_theme_option('Background Header')

Activities

- Add a configuration option to change the color to use for item titles
- Add configuration option to highlight the description of an item (see items/show page activities)

Overriding plugin views

Explain how to look through plugin files and override their views

- Plugins have a 'views/public' (or 'views/shared') folder that contains the views (pages) they generate. You can override those the same way you override the core views. Copy the first folder under 'public' or 'shared' to your theme

Activities

- Install SimplePages and hack the public/show page to alter the title

Changes coming in Omeka 2.0 (Fall 2012)

- See [Updating Plugins For Omeka 2.0](#)