

AIMS website reengineering

Project status and goals - January 2012



Where we started

Stratification of several development actions

Lack of documentation

Even small features were difficult to implement/integrate

Complex information architecture generates complex content management

Difficult language management

Where we started: technical data

319 modules

47 menus

159 blocks

41 content types

22 taxonomy vocabularies

Table based theme

Business Logic inside PHP nodes

Several **vulnerabilities**

The approach

We decided to start from a plain vanilla Drupal install.

We then applied a mixed approach,
by using export-import techniques and
database-to-database migration.

Our goal is to offer almost the same features
while reducing the complexity of the system
thus achieving better performances.

Modules

We had 319 modules.

Some modules were deprecated due to security flaws, and unsupported, some were in early development state.

Actions:

- reduce the number of modules by analyzing and reengineering features**
- remove deprecated modules, replacing features with other modules**

We created a starting point of 155 modules.

All this modules are supported and the vast majority is in a mature development state.

Features supported by deprecated modules (eg. OG Forum) will be obtained with other modules combos.

Menus

We had 47 menus.

Due to the lack of a solid navigational system and interface several menus have been created, to support sections subnavigation and multilanguage interface.

Action: reduce the number of menus by

- creating a main navigation menu, including all section submenus**
- translating menus within Drupal interface, instead of duplicating**

We now have 1 primary navigation menu.

We're collapsing all other section menus and engineering other application specific menus. We're rendering menus with Menu Blocks, with complete control over tree navigation.

Content types

We have 41 content types.

Some are used to hold business logic that should be embedded in module functions.
Some share the same structure, and could be collapsed and managed with taxonomies.

Action: reduce the number of content types by

- collapsing similar content types**
- removing business logic related content types and move code to modules**
- categorizing using taxonomies**

Taxonomy vocabularies

We have 22 taxonomy vocabularies.

Most terms are shared between vocabularies, to define the same concept.

Action: reduce the number of vocabularies by

- collapsing identical terms in different vocabularies**
- applying the same vocabulary to different content types to allow querying and collating heterogeneous contents**

**This will imply revising existing views and building new views.
This approach will allow building panels that collate contents related to the same concept.**

Theme

Previous theme was based on HTML tables.

There was embedded business logic, embedded Javascript, and XSS vulnerabilities.
CSS was not engineered, and hard to extend due to heavy usage of !important clause.
Inconsistent typography, lack of grid foundation, lots of templates.
Inconsistent regions management, low navigation interface usability.

Action: create a new theme from scratch.

- Table-less layout**
- SASS based**
- Grid based**
- Consistent typography framework**
- Still adherent to FAO standards**
- Focus on usability**

Business logic inside PHP nodes

Many dynamic features are implemented with code in PHP nodes.

Some nodes process unsanitized URL parameters, opening vulnerability holes.

Action: port all PHP nodes in a custom module

- sanitize URL parameters
- optimize code
- use Drupal language functions (`t()`) instead of hard-coded arrays
- avoid code duplication by mean of shared libraries
- provide documentation and a single code repository

Business logic inside PHP nodes

```
/*
global $base_url;
global $language;

$page = "1";
if (!empty($_GET['Page']))
    $page = $_GET['Page'];
$yearNow = date("Y");
$year = $yearNow;
if (!empty($_GET['Year']))
    $year = $_GET['Year'];
$lang = $language->language;
if (strtolower($lang) == "zh-hans")
    $lang = "ZH";
$lang = strtoupper($lang);

if (!empty($_GET['Lang']))
    $lang = strtoupper($_GET['Lang']);

$currentUrl = $_SERVER['REQUEST_URI'];
if (strpos($currentUrl, '?', 0) > 0)
    $currentUrl = substr($currentUrl, 0, strpos($currentUrl, '?', 0));
else {
    if (strpos($currentUrl, '&', 0) > 0)
```

Business logic inside PHP nodes

```
function aims_support_menu() {
    $items = array();

    $items['publications'] = array(
        'title' => t('FAO Publications on Information Management'),
        'page callback' => 'aims_support_publications_page',
        'access arguments' => array('access content'),
        'type' => MENU_NORMAL_ITEM,
    );
    return $items;
}

function aims_support_publications_page() {
    global $language;

    //set arguments
    $page = (empty($_GET['Page'])) ? 1 : filter_xss($_GET['Page'], array());
    $year = (empty($_GET['Year'])) ? date("Y") : filter_xss($_GET['Year'], array());
    $lang = (empty($_GET['Lang'])) ? strtoupper($language->language) : strtoupper(filter_xss($_GET['Lang'], array()));

    drupal_set_title(t('FAO Publications on Information Management'));

    ...
}
```

Next steps

- Finalize taxonomies optimization**
- Build taxonomy panels for all section pages**
- Complete PHP nodes migration in module**
- Release first set of features + new theme**
- Implement OG Forums features with other modules**
- Revise content types**
- Release final version**