



Pam Fournier (Information Technologist)
Vitthal Kudal (Developer, expertise with multisite architecture)
David P. Shorthouse (Project Lead)
Lisa Walley (Developer, expertise with EduLifeDesks)

Philosophy

LifeDesks “Expert” support taxonomists’ desire to expose their work in online, publicly accessible environments and to participate in the Encyclopedia of Life as a byproduct of their activities. In essence, LifeDesks provide an effective solution to create new Encyclopedia of Life content partners without users requiring knowledge of XML transfer schema or of programming. They are also flexible, modular environments that can be extended with functions that may never be implemented on the Encyclopedia of Life because of issues with scale. Several requirements must be met to ensure that the platform is engaging and sustainable (adapted after Hine, 2008):

- Be meaningful (*i.e.* foster engagement with organisms)
- Attract funding
- Provide personal and institutional visibility
- Be scholarly (*e.g.* citation metrics)
- Be simple and task-oriented
- Federate workloads (*i.e.* share the work)

Basic Functions

LifeDesks were released May 2009 with three basic contributory functions: managing images, managing bibliography, and authoring species page text. In order for LifeDesks owners to best coordinate their media, several organizational functions were implemented: managing classification(s) and managing membership & permissions. LifeDesks owners may also adjust the basic look & feel of their sites including changing theme colours, logo, and/or banner image. They may also download their entire database, content, obtain the code and entirely delete their site if they so choose. Each LifeDesk is an independent database with its own users table, roles, and permissions. All LifeDesks may create the Encyclopedia of Life content partner XML document without requiring knowledge of XML. Similarly, all LifeDesks have the option to publish their primary working classification for deposition in the Global Names Architecture.

Relationship With Other Projects

The LifeDesks Team has a close working relationship with the Scratchpad Team (<http://www.scratchpads.eu>), who share a similar philosophy and who also

use the same Drupal Content Management Platform. We share code and approaches, though at the time of writing, Scratchpads are not set-up as content partners to the Encyclopedia of Life.

Drupal

Drupal is PHP-based, open-source software distributed under a GNU General Public License (GPL) (see <http://drupal.org>). It is a content management system that may be extended with modules, many of which have been uploaded by enthusiasts to the Drupal website. Whereas Drupal is an excellent system to help manage data objects (*i.e.* “nodes” in Drupal jargon), it lacks somewhat in its ability to have metadata associated with terms. This is particularly troublesome when using organism names as terms to tag content. However, we have done much in-house work to better coordinate metadata on terms and relationships among terms. This work is ongoing.

Hardware & Architecture

Table 1. Server specifications for the LifeDesks project

Server	Model	RAM	CPU	Hard Drive	Software
Ldapp1 production Application server	Dell Poweredge SC1435	12gb	Dual Dual Core 2 GHz	750gb	Centos 5.3 PHP 5.2.9 MySQL 5.0.77 Apache httpd-2.2.3
Ldapp2 production Application server and Load Balancer failover	Dell Poweredge 2950	16gb	Quad Core 2.66GHz	750gb	Centos 5.3 PHP 5.2.6 MySQL 5.0.74 Apache httpd-2.2.6 Nginx 0.6.32
Lddata1 production master database server	Dell Poweredge SC1435	16gb	Dual Quad Core 2.83GHz	500gb	Centos 5.3 MySQL 5.0.74 DRBD NFS 1.0.9
Lddata2 production data server failover	Dell Poweredge SC1435	16gb	Dual Quad Core 2.83GHz	500gb	Centos 5.3 MySQL 5.0.74 DRBD NFS 1.0.9
Ldload1 production load balancer	Dell Poweredge SC1435	12gb	Dual Dual Core 2 GHz	750gb	Centos 5.3 PHP 5.2.6 MySQL 5.0.67 Nginx 0.6.33
Lddemo/T1 failover development preview server and T1 failover	Dell Poweredge SC1435	12gb	Dual Dual Core 2 GHz	750gb	Centos 5.3 PHP 5.2.6 MySQL 5.0.74 Apache httpd-2.2.6 Nginx 0.6.32
Ldintegration development Integration server	Dell Poweredge SC1435	12gb	Dual Quad Core 2.83GHz	500gb	OpenVZ 3.0.23-1 MySQL 5.0.74 PHP 5.2.6 Apache httpd-2.2.6 Nginx 0.6.32 NFS 1.0.9

All traffic passes through a load balancer that shuttles visitors to one of two application servers. There is one MySQL master database and one file server. The former is quietly replicated to a MySQL slave that is not involved in serving data. Rather, it is used for fault tolerance in the event the master fails. The file server is mounted to both application servers via NFS and it too has a failover.

Apache and php.ini Configurations

In addition to the above settings, LifeDesks also require additional Apache server configurations to enable clean URLs and multisite hosting with wildcard subdomains (Figure 1). The Apache module, `mod_rewrite` will also need to be installed and enabled.

In order to make use of our Drupal `excel_reader` module, PHP must have the `zip.so` module installed and enabled in `php.ini` as well as `auto_detect_line_endings = On`. Additional `php.ini` configurations include: `max_execution_time=180`, `max_input_time=180` and `memory_limit=512M`.

```
<VirtualHost *:80>
  ServerName www.lifedesks.org
  ServerAlias *.lifedesks.org
  ServerAdmin myemailaddress@here.com
  DocumentRoot /Users/davidshorthouse/Sites/LifeDesks/drupal/trunk
  <Directory "/Users/davidshorthouse/Sites/LifeDesks/drupal/trunk">
    Options -Indexes
    AllowOverride None
    Order allow,deny
    Allow from all
    RewriteEngine on
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteCond %{REQUEST_FILENAME} !-d
    RewriteCond %{REQUEST_URI} !=/favicon.ico
    RewriteRule ^(.*)$ index.php?q=$1 [L,QSA]
  </Directory>
</VirtualHost>
```

Figure 1. Apache Virtual Host configurations to enable clean URLs and multisite hosting. The DocumentRoot and Directory locations should be adjusted to point to the directory where the Drupal instance will be served.

Multisite Hosting

Each LifeDesk has its own subdomain and the URL is resolved from the `lifedesks.org` domain via a wild-card DNS entry. Internally, the subdomain determines the name of the database and the file directory name, which are automatically created when a new site is created. A base, administrative Drupal installation coordinates creation, update, and deletion of sites and is accessible only via our VPN connection. Drupal has a bootstrap process that is very dependent on directory names (Figure 2), which must match the URL(s) being called. Connecting to the correct database and directory of files is coordinated with Drupal's settings.php files (Figure 3) within these directories.

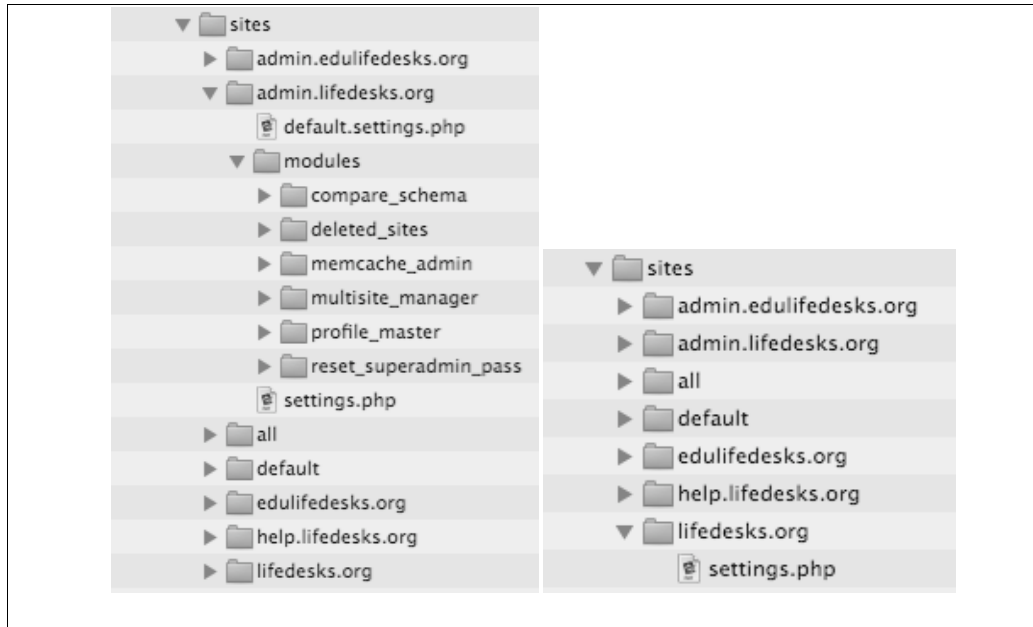


Figure 2. Directory structures illustrating how LifeDesks are served. The administrative, base installation coordinates the creation of sites (left, `admin.lifedesks.org`) and the individual LifeDesks sites are accessed via their subdomains (right, `lifedesks.org`). Modules in the `admin.lifedesks.org/modules` directory are not accessible to individual LifeDesks sites. Modules in the `sites/all/modules` directory are accessible to all hosted sites.

```
$db_prefix = "";
$base_url = "http://admin.lifedesks.org";
$db_url = "mysql://lifedesk:lifedesk@localhost/lifedesk_production";
$config['file_directory_path'] = "files/";
*****

$db_prefix = "";
$urlParts = explode('.', $_SERVER['HTTP_HOST']);
$subd = $urlParts[0];
$base_url = "http://" . $subd . ".lifedesks.org";
$db_url = 'mysql://lifedesk:lifedesk@localhost/' . $subd;
$config['file_directory_path'] = "files/" . $subd;
$config['theme_default'] = 'eol';
$config['maintenance_theme'] = "eol";
```

Figure 3. Relevant content of the `admin.lifedesks.org`'s `settings.php` (top) and that of other LifeDesks sites (bottom).

Code & Modules

Drupal core 6.15 is available for download at <http://drupal.org>. All the modules we use or wrote (Table 2, Appendix 1) are available for download at <http://www.lifedesks.org/modules>. These are updated with every code release, which occurs approximately once every two weeks.

The complex suite of classification modules uses a poorly used Drupal core mechanism to describe relationships between terms and to extend terms with

metadata. In order for this mechanism to meet our needs, we made a modification to the core taxonomy module and include this patch in our contributed classification module. This is the only modification we have made to Drupal core.

Table 2. Modules used in LifeDesks. * Indicate required modules and ★ indicate LifeDesks team modules

sites/admin.lifedesks.org Modules	sites/all Modules
compare_schema★	administerusersbyrole
db_switch★	apachesolr
deleted_sites★	biblio
drupal_sites★	captcha
lifedesk_stats★	catalog★
mass_contact	classification★
memcache_admin	classification_import★
multisite_manager★*	classification_export★
profile_master*	classification_settings★
reset_superadmin_pass	cmf
	compare_schema_required
	conceal_modules★
	config_perms
	creativecommons_lite
	excel_reader★
	front_page
	image
	jquery_ui
	jquery_update
	jstree★
	lifedesk_alter★
	lifedesk_announcements★
	lifedesk_biblio★
	lifedesk_foot★
	manage_site★
	members★
	memcache
	modify_account★
	modify_menu★
	ocmf★
	override_node_options
	path_redirect
	permissions_api
	piclens
	protect_critical_users
	recaptcha
	role_delegation
	smtip
	taxon_description★
	taxon_export★
	taxon_gallery★
	taxon_list★
	taxonpage★
	wysiwyg

Profiles

Drupal has a mechanism to create sites using what are known as profiles. These are simple PHP files (*.profile) with specifically named functions that install modules and populate the database. Custom profiles are placed in a folder named according to the profile it contains. For example, our “expert” profile (Appendix 2) is

/profiles/expert/expert.profile. In concert with the administrative multisite_manager, profile_master, and crud.inc in /includes, sites are created with the precise specifications prescribed in expert.profile.

Themes

Drupal also has the capacity to flexibly theme a site. Themes folders are placed in /sites/all/themes for cases where a theme is to be employed across all sites in a multisite hosted environment. We developed a simple “EOL” theme that makes use of the color module.

Internationalization

Drupal has an active community of developers who streamline and simplify internationalization. Drupal core itself has the capacity to permit on-the-fly selection of navigational language through *.pot and *.po files via its core locale module. Modules involved in navigation must also have these files in their translations directory. Background on Drupal internationalization may be obtained at <http://drupal.org/project/i18n>. Modules may be translated into other languages using the translation template extractor module: <http://drupal.org/project/potx>.

At the time of writing, none of our in-house modules have been translated. However, we have been diligently using Drupal conventions for snippets of navigational text requiring translation in both the PHP and JavaScript code such that the translation template extractor may be used to create the *.pot and *.po files for any language.

Deployment & Update Procedures

We have one svn repository, but organize it such that Drupal core has its own trunk, branches, and tags directories as does our code directory, which contains the profiles and modules directories. Deployment consists of exporting code from the current Drupal tag as well as the profiles and modules tag, merging the two directories (*i.e.* the /sites/all/modules and /profiles directories in Drupal core are replaced with our /modules and /profiles directories), then pushing the result to both application servers. This is typically done by first taking all sites offline as seen from the outside. We access all sites by first adjusting our VPN network and resolv.conf on our load balancer to point to an internal DNS server. In order to communicate to end-users that sites will be offline, we post a message to all sites long in advance of the event via the administrative installation.

Drupal has a command line utility called drush: <http://drupal.org/project/drush> that may be used to install and update modules. We have instead opted to craft our own UI-based update utility in a module installed in the base, administrative site. This module is called compare_schema and acts in concert with another module called db_switch to install new modules across all sites and to update any and all modules requiring an update. Because the administrative install is itself a Drupal site, modules MUST be placed in /sites/all and not

/sites/lifedesks.org/modules otherwise, the administrative site will not be able to access these modules when sites are created. We use an include file at /sites/admin.lifedesks.org/modules/profile_master/profile_master.admin.inc to coordinate the installation of new modules. We also have a copy of this include file in /sites/all/modules/conceal_modules such that individual LifeDesks owners cannot inadvertently uninstall mission critical modules.

Work in Progress

The LifeDesks team will soon be releasing EduLifeDesks, <http://www.edulifedesks.org>. This site will be a single database and will make heavy use of Drupal's organic groups and workflow. The intention of this new platform is for students to create species pages as part of curricular work. The anticipated release date is the end of February.

At the time of writing, we have been working on Apache Solr search across and within LifeDesk expert sites as well as an AJAX grid editor to complement the tree editor for taxonomic names. This new functionality will be deployed with version 2.25 of our code.

The classification suite of modules continues to receive the most attention by LifeDesks developers. While there is a functional mechanism to assign taxonomic ranks and to coordinate varied name relationships, this mechanism is not as flexible as it could be for end-users. We are currently working with the Scratchpad group to treat taxonomic names as Drupal nodes just as are images, bibliographic items, and other content types. This will result in a massive restructuring of the classification suite of modules. However, update hooks will afford a seamless migration path when this work is complete. Expected release of this restructuring is summer 2010.

References

Hine, Christine. 2008. Systematics as Cyberscience: Computers, Change, and Continuity in Science. MIT Press. Cambridge, Massachusetts. 307pp.

Appendix 1 Description of LifeDesks Team contributed modules

Modules used in Each LifeDesk

catalog

The catalog module is dependent on the classification suite of modules and is our first attempt at producing a flat list of names from a site's primary classification.

classification

This module is core to LifeDesks functionality. It has a block hook to permit navigation of taxa in a hierarchy, an elements hook to inject a form element used to multi-tag, and several dozen callback functions to coordinate the editing of taxonomic names in its editing interface.

classification_import

This module coordinates the import of names into biological classifications.

classification_export

This module permits the export of names in biological classifications for end-users and also packages a primary classification as a tarred and gzipped file containing Darwin Core Archive documents.

classification_settings

This module alters Drupal vocabularies form to permit more granular tagging options for content with taxonomic names.

compare_schema_required

This module contains a single callback function that flush the database caches for a site

conceal_modules

The profile_master.admin.inc file included in this module acts to remove disabled the viewing of modules in the modules listing page.

excel_reader

This module is a wrapper for the PHPExcel class,
<http://www.codeplex.com/PHPExcel>

jstree

The classification suite of modules is dependent on this jQuery-based extension, which is itself under active development. We use jsTree 0.9.8b, <http://www.jstree.com>, and it is bundled in this module. We recommend not upgrading to the current 0.9.9b version. jsTree 1.0 is under active development but will require jQuery 1.4.1 when released and a modest rewrite of all classification suite modules.

lifedesk_alter

This module bundles are necessary form_alter hooks and is weighted such that it executes in the bootstrap process after any and all modules have executed their form_alter hooks.

lifedesk_announcements

This module contains a single block that is populated with a message set from a variable. We set this variable from the administrative site in the advance of releasing new code.

lifedesk_biblio

This module alters the classification editing panel by inserting an input box to select a publication that contains the original taxonomic description.

lifedesk_foot

This module contains a single footer hook and injects Google Analytics code across all pages on all sites.

manage_site

This module permits the download of a LifeDesk database, all content, and also permits users to delete their own site.

members

A simple members list page.

modify_account

This module adds two columns to the Drupal core users table, givenname and surname and alters the profile form.

modify_menu

This module adjusts the arrangement, naming, and selection of menu items in the primary navigation.

ocmf

This module, short for “orphaned content management filter” acts in concert with `taxon_description` to allow users an opportunity to migrate their `taxon` page chapter content in the event another member of the site (with appropriate permissions) deleted a chapter heading.

taxon_description

This module creates two hidden vocabularies and populates them with Species Profile Model chapter headings. It also permits the renaming, moving, or deleting of chapter headings. Because we give users an opportunity to rename their chapter headings, this module also has the facility to map their new chapters to the Species Profile Model such that `taxon_export` (below) may continue to serve standard exports as required by the Encyclopedia of Life.

taxon_export

This module has many dependencies on other modules and its primary function is to produce the Encyclopedia of Life content partner XML document.

taxon_gallery

A simple image gallery.

taxon_list

A list of taxa for which a text exists on the site.

taxonpage

This module is similar to the `taxonomy/term` page, but is customized to produce `taxon` pages with a snippet of the hierarchy, images, text, biblio, a citation, a print & PDF version, among other function.

Modules Used by the Administration Site

compare_schema

This module is the administrative user interface to update the schema across, push messages, and/or flush the database caches across all sites.

db_switch

This module permits the opening and closing of database connections during site update procedures.

deleted_sites

This module installs a table in the administrative site that captures feedback from users should they elect to delete their site.

drupal_sites

The primary function of this module is a cron hook that gathers data across all sites for presentation on our <http://www.lifedesks.org> marketing pages. It also executes cron.php for all LifeDesks sites.

lifedesk_stats

This module creates a text file containing LifeDesks site statistics for consumption by the Encyclopedia of Life.

multisite_manager

This and the profile_master module below is core to the administrative site's functionality. It receives an http POST message from <http://www.lifedesks.org/create>, sends an email message to the user, then receives an additional call from the marketing pages to create a new site.

profile_master

This module allows creation and management of an installation profile. We also the array of modules in profile_master.admin.inc in concert with compare_schema and db_switch to install new modules on existing sites.

Appendix 2 The LifeDesks expert.profile file used in concert with the administrative multisite_manager and profile_master modules to create a new, preconfigured site

```
<?php
/*
  You must include the crud.inc file in your profile.
  The file is part of the install_profile_api module.
*/
include_once('includes/crud.inc');
require_once('profiles/default/default.profile');

/**
 * Details about the profile.
```

```

*/
function expert_profile_details() {
  return array (
    'name' => 'Expert',
    'description' => 'Installation profile for Expert LifeDesks.',
  );
}

/*
 * List of modules to install, in order of dependency.
 */
function expert_profile_modules() {
  return array (
    //Drupal core required modules are automatically installed
    //Drupal core optional modules
    'color',
    'comment',
    'dblog',
    'locale',
    'menu',
    'profile',
    'search',
    'taxonomy',
    'trigger',
    'upload',
    //LifeDesk required modules
    'administerusersbyrole',
    'apachesolr',
    'apachesolr_image',
    'apachesolr_search',
    'biblio',
    'biblio_pm',
    'captcha',
    'classification',
    'classification_biblio',
    'classification_export',
    'classification_import',
    'classification_settings',
    'cmf',
    'compare_schema_required',
    'conceal_modules',
    'config_perms',
    'cooliris_lifedesk_gallery',
    'creativecommons_lite',
    'excel_reader',
    'front_page',
    'image',
    'jquery_ui',
    'jquery_update',
    'jstree',
    'lifedesk_alter',
    'lifedesk_announcements',
    'lifedesk_biblio',
    'lifedesk_foot',
    'manage_site',
    'members',
    'modify_account',
    'modify_menu',
    'ocmf',
    'override_node_options',
    'path_redirect',
  );
}

```

```

    'permissions_api',
    'piclens',
    'protect_critical_users',
    'recaptcha',
    'smtp',
    'role_delegation',
    'taxon_description',
    'taxon_export',
    'taxon_gallery',
    'taxon_list',
    'taxonpage',
    'wysiwyg',
  );
}

/**
 * Tasks to perform during installation.
 */
function expert_profile_tasks(&$task, $url) {
  if($task == 'profile') {
    /**
     * include default profile tasks to install default content types and variables:
     * use with care and override as necessary.
     */
    default_profile_tasks($task, $url);

    /**
     * Install node types.
     */
    //no custom node types to add here (default page and story added from default profile).

    /**
     * Set variables.
     */
    variable_set('override_node_options_pocollapsed',0);
    variable_set('clean_url', '1');
    variable_set('css_js_query_string', 'F000000000000000000000');
    variable_set('date_default_timezone', '-18000');
    variable_set('file_directory_temp', '/tmp');
    variable_set('filter_html_1', 1);
    variable_set('install_task', 'done');
    variable_set('install_time', 1230151731);
    variable_set('user_picture_dimensions', '200x200');
    variable_set('javascript_parsed', array ());
    variable_set('menu_expanded', array ());
    variable_set('menu_masks', array (
      62,
      61,
      59,
      31,
      30,
      29,
      24,
      21,
      15,
      14,
      11,
      7,
      6,
      5,
      3,

```

```

2,
1,
));
variable_set('node_options_forum', array (
  'status',
));
variable_set('node_options_page', array (
  'status',
));

variable_set('theme_settings', array (
  'toggle_node_info_page' => false,
));

variable_set('user_email_verification', true);

//performance variables
variable_set('cache',1);
variable_set('page_compression',1);
variable_set('preprocess_css',1);
variable_set('preprocess_js',1);

//error reporting & logging
variable_set('site_404',"");
variable_set('site_403',"");
variable_set('error_level',0);
variable_set('dblog_row_limit',10000);

//file upload limits & user quota for file upload
variable_set('upload_uploadsize_default', '5'); //default maximum file size per upload is set to 5MB
variable_set('upload_usersize_default', '10240'); //default total file size per user is det to 10GB

//jquery update
variable_set('jquery_update_replace',true);
variable_set('jquery_update_compression_type', 'min');

//filters and allowed html
variable_set('filter_html_1','1');
variable_set('filter_default_format','1');
variable_set('filter_html_help_1',0);
variable_set('allowed_html_1','<a> <em> <strong> <cite> <code> <ul> <ol> <li> <dl> <dt> <dd> <p> <img> <sub>
<sup> <p> <br> <table> <thead> <tbody> <tr> <td>');

//comment variables for content types
$content_types = array('image','taxon_description');

foreach($content_types as $content_type) {
  variable_set('comment_'. $content_type, '2');
  variable_set('comment_default_mode_'. $content_type, '3');
  variable_set('comment_default_order_'. $content_type, '1');
  variable_set('comment_default_per_page_'. $content_type, '30');
  variable_set('comment_controls_'. $content_type, '3');
  variable_set('comment_anonymous_'. $content_type, 0);
  variable_set('comment_subject_field_'. $content_type, '1');
  variable_set('comment_preview_'. $content_type, '1');
  variable_set('comment_form_location_'. $content_type, '1');
}

//path redirect default settings
variable_set('path_redirect_redirect_warning',0);
variable_set('path_redirect_allow_bypass',0);

```

```

variable_set('path_redirect_auto_redirect',1);
variable_set('path_redirect_purge_inactive',0);
variable_set('path_redirect_default_status',301);

//frontpage variables
$welcome_anonymous = '<h3 class="h">' . t('Welcome') . '</h3>' . "\n";
$welcome_anonymous .= '<p>' . t('This is a new LifeDesk where participants will edit and maintain their
classification, upload images, maintain bibliographic resources, among other activities.') . '</p>';

variable_set('front_welcome_anonymous',array('body' => $welcome_anonymous, 'format' => 1));

$welcome_authenticated = '<h3 class="h">' . t('Welcome') . '</h3>' . "\n";
$welcome_authenticated .= '<p>' . t('You are free to edit this welcome message for your new LifeDesk. You have
two different welcome messages: one for anonymous visitors and a second for authenticated participants.') . '</p>' .
"\n";
$welcome_authenticated .= '<p>' . t('Names management and a hierarchical classification are core to this version of
LifeDesk. Once you import your working classification, the tree at top-right and on each of your taxon pages may be
used as a navigational structure. Recent images and bibliographic entries will also appear on anonymous and
authenticated versions of your frontpage. As you create content, tag it with names from your classification to enrich
your taxon pages.') . '</p>' . "\n";

variable_set('front_welcome_authenticated',array('body' => $welcome_authenticated, 'format' => 1));

/**
 * Set outgoing email messages
 */
$message_approval = t('A new user with username: !username, and email: !mailto requested an account on your
LifeDesk !site. Activation of !username's account is pending your approval.') . "\n\n" . t('To view and activate the
account, log in to your LifeDesk: !uri and edit !username's profile (!edit_uri), selecting 'active' under account status.')
. "\n\n" . t('For further information on activating new user accounts in your LifeDesk, please visit
http://help.lifedesks.org/members/new.');
```

```

variable_set('user_mail_register_pending_approval_admin_subject', t('A new user (!username) has requested an
account on your !site LifeDesk.');
```

```

variable_set('user_mail_register_pending_approval_admin_body',$message_approval);

/**
 * Set SMTP settings
 */
variable_set("smtp_on", "1");
variable_set("smtp_host", "X.X.X.X"); //obfuscated IP address here
variable_set("smtp_hostbackup", "X.X.X.X"); //obfuscated IP address here
variable_set("smtp_port", "25");
variable_set("smtp_protocol", "standard");
variable_set("smtp_username", "");
variable_set("smtp_password", "");
variable_set("smtp_from", "");
variable_set("smtp_fromname", "");
variable_set("smtp_debugging", 0);
variable_set("smtp_library", "sites/all/modules/smtp/smtp.module");

/**
 * Apache Solr search
 */
variable_set('search_cron_limit', '200');
variable_set('minimum_word_size', '3');
variable_set('apachesolr_host', 'localhost'); //change to Solr host for production env't
variable_set('apachesolr_port', '8983');
variable_set('apachesolr_path', '/solr');
variable_set('apachesolr_cron_limit', '100');
variable_set('apachesolr_rows', '10');
```

```

variable_set('apachesolr_failure', 'show_no_results');
variable_set('apachesolr_search_make_default', '1');
variable_set('apachesolr_search_spellcheck', 0);
variable_set('apachesolr_search_taxonomy_links', '1');
$apachesolr_search = array(
    'apachesolr_search' => array(),
);
variable_set('apachesolr_enabled_facets', $apachesolr_search);

/**
 * Set default roles and permissions.
 */
$permissions = array();
//anonymous user permissions
$permissions['anonymous user'] = array(
    'access content',
    'view revisions',
    'view uploaded files',
    'access user profiles',
    'access biblio content',
    'show export links',
    'show download links',
    'show filter tab',
    'show sort links',
    'view full text',
    'view user content list',
    'view original images',
    'view taxon description',
    'access comments',
    'inspect all votes',
    'vote on polls',
    'search content',
);
$permissions['authenticated user'] = array(
    'post comments',
    'post comments without approval',
    'cancel own vote',
    'change own username'
);
foreach($permissions['anonymous user'] as $perm) {
    $permissions['authenticated user'][] = $perm;
}
$permissions['contributor'] = array(
    'create page content',
    'create story content',
    'delete own page content',
    'delete own story content',
    'edit own page content',
    'edit own story content',
    'revert revisions',
    'upload files',
    'create biblio',
    'edit own biblio entries',
    'import from file',
    'show own download links',
    'access classification pages',
    'create images',
    'edit own images',
    'mass upload images',
    'edit captions',
    'filter and manage orphaned content',

```

```

'override image published option',
'override taxon_description published option',
'override page published option',
'override story published option',
'create taxon description',
'edit own taxon description',
'delete own taxon description',
'export template',
'import content',
'create poll content',
'delete own poll content',
'edit own poll content'
);
foreach($permissions['authenticated user'] as $perm) {
    $permissions['contributor'][] = $perm;
}
$permissions['editor'] = array(
    'post comments without approval',
    'delete revisions',
    'edit any page content',
    'edit any story content',
    'access administration pages',
    'administer taxonomy',
    'edit all biblio entries',
    'edit biblio authors',
    'create classification',
    'import classification',
    'edit classification',
    'delete classification',
    'export classification',
    'filter and manage site content',
    'edit images',
    'edit any taxon description',
    'edit any poll content'
);
foreach($permissions['contributor'] as $perm) {
    $permissions['editor'][] = $perm;
}
$permissions['administrator'] = array(
    'administer menu',
    'delete any page content',
    'delete any story content',
    'administer biblio',
    'settings for catalog',
    'settings for classification',
    'share classification',
    'administer modules',
    'administer site information',
    'administer themes',
    'administer users',
    'administer user profile fields',
    'create users',
    'delete users with role contributor',
    'delete users with role editor',
    'edit users with role contributor',
    'edit users with role editor',
    'display site building menu',
    'administer frontpage',
    'manage site',
    'perform backup',
    'perform export',

```



```

    'assign contributor role',
    'assign editor role',
    'delete any taxon description',
    'administer taxon description',
    'administer taxon pages',
    'export taxon content',
    'delete any poll content'
);
foreach($permissions['editor'] as $perm) {
    $permissions['administrator'][] = $perm;
}
$permissions['owner'] = array(
    'delete users with role administrator',
    'edit users with role administrator',
    'edit users with role owner',
    'delete site',
    'assign administrator role',
);
foreach($permissions['administrator'] as $perm) {
    $permissions['owner'][] = $perm;
}

foreach($permissions as $role => $perms) {
    $role_id = array();
    install_set_permissions($role_id[$role] = install_add_role($role), $perms);
}

/*****
*          URL ALIASES          *
*****/

/*****
*          BLOCKS              *
*****/

install_add_block('system', '0', 'eol', 1, 10, 'footer', 0, 0, 0, " ");
install_add_block('user', '0', 'eol', 1, 0, 'left', 0, 0, 0, " ");
install_add_block('user', '1', 'eol', 1, 0, 'left', 0, 0, 0, " ");
install_add_block('lifedesk_announcements', '0', 'eol', 1, -10, 'left', 0, 0, 0, " ", -1);

/*****
*          EXPORTING NODES      *
*****/

system_initialize_theme_blocks('eol');

/*****
*          INSTALL THEME        *
*****/

install_disable_theme('garland');
install_default_theme('eol');
db_query("UPDATE {system} SET status = 0 WHERE type = 'theme' and name = 'garland'");

/**
 * Add path redirect
 */
//add redirect -what about error handling here/during install?
update_sql("INSERT INTO path_redirect (path,redirect) VALUES ('user/1', 'members')");
/**
 * Assign owner role to site creator
 */
//get site owner uid

```

```

$owner_uid = variable_get('site_owner_uid',1);
//get owner role id
$owner_rid = db_result(db_query("SELECT rid FROM role WHERE name = '%s'",'owner'));
//update site owner to have owner role
if($owner_uid > 1 && $owner_rid > 0) {
  $ret = array();
  $ret[] = db_query("INSERT INTO users_roles (rid,uid) VALUES (%d, %d)", $owner_rid, $owner_uid);
  //TODO: error handling?
}

```

```

/*****
*           WYSIWYG Config           *
*****/

```

```

$wysiwyg_config_1 = array(
  'default' => 1,
  'user_choose' => 0,
  'show_toggle' => 0,
  'theme' => 'advanced',
  'language' => 'en',
  'buttons' => array(
    'default' => array(
      'bold' => 1,
      'italic' => 1,
      'undo' => 1,
      'redo' => 1,
      'link' => 1,
      'unlink' => 1,
      'sup' => 1,
      'sub' => 1,
      'charmap' => 1,
      'code' => 1,
      'cut' => 1,
      'copy' => 1,
      'paste' => 1,
    ),
    'paste' => array(
      'pasteword' => 1,
    ),
  ),
  'table' => array(
    'tablecontrols' => 1,
  ),
  'safari' => array(
    'safari' => 1,
  ),
  ),
  'toolbar_loc' => 'top',
  'toolbar_align' => 'left',
  'path_loc' => 'none',
  'resizing' => 1,
  'verify_html' => 1,
  'preformatted' => 0,
  'convert_fonts_to_spans' => 1,
  'remove_linebreaks' => 1,
  'apply_source_formatting' => 0,
  'paste_auto_cleanup_on_paste' => 1,
  'block_formats' => 'p,address,pre,h2,h3,h4,h5,h6,div',
  'css_setting' => 'theme',
  'css_path' => "",
  'css_classes' => "",
);

```

```
$wysiwyg_config_2 = array(
    'default' => 0,
    'user_choose' => 0,
    'show_toggle' => 0,
    'theme' => 'advanced',
    'language' => 'en',
    'buttons' => array(
    ),
    'toolbar_loc' => 'top',
    'toolbar_align' => 'left',
    'path_loc' => 'none',
    'resizing' => 1,
    'verify_html' => 1,
    'preformatted' => 0,
    'convert_fonts_to_spans' => 1,
    'remove_linebreaks' => 1,
    'apply_source_formatting' => 0,
    'paste_auto_cleanup_on_paste' => 1,
    'block_formats' => 'p,address,pre,h2,h3,h4,h5,h6,div',
    'css_setting' => 'theme',
    'css_path' => '',
    'css_classes' => '',
);

//filtered HTML
$ret[] = db_query("INSERT INTO wysiwyg (format,editor,settings) VALUES
(%d,'%s','%s')",1,'tinymce',serialize($wysiwyg_config_1));

//full HTML
$ret[] = db_query("INSERT INTO wysiwyg (format,editor,settings) VALUES
(%d,'%s','%s')",2,'tinymce',serialize($wysiwyg_config_2));

return;
}
}

?>
```