

```

<?php

$bfa_ata_version = "3.6.7";


// Load translation file above

load_theme_textdomain('atahualpa');


// To disable some default WP filters, remove the '#' character

#remove_filter('the_content', 'wptexturize');

#remove_filter('the_excerpt', 'wptexturize');

#remove_filter('comment_text', 'wptexturize');

#remove_filter('the_title', 'wptexturize');


// get default theme options

include_once (TEMPLATEPATH . '/functions/bfa_theme_options.php');

// Load options

include_once (TEMPLATEPATH . '/functions/bfa_get_options.php');

list($bfa_ata, $cols, $left_col, $left_col2, $right_col, $right_col2, $bfa_ata['h_blogtitle'],
$bfa_ata['h_posttitle']) = bfa_get_options();

// Sidebars:

if ( function_exists('register_sidebar') ) {

    register_sidebar(array(

        'name'=>'Left Sidebar',

        'before_widget' => '<div id="%1$s" class="widget %2$s">',

        'after_widget' => '</div>',

        'before_title' => '<div class="widget-title"><h3>',

        'after_title' => '</h3></div>',

    ));

    register_sidebar(array(

```

```

        'name'=>'Right Sidebar',

        'before_widget' => '<div id="%1$s" class="widget %2$s">',

        'after_widget' => '</div>',

        'before_title' => '<div class="widget-title"><h3>',

        'after_title' => '</h3></div>',

    ));

```

```

register_sidebar(array(

    'name'=>'Left Inner Sidebar',

    'before_widget' => '<div id="%1$s" class="widget %2$s">',

    'after_widget' => '</div>',

    'before_title' => '<div class="widget-title"><h3>',

    'after_title' => '</h3></div>',

));

```

```

register_sidebar(array(

    'name'=>'Right Inner Sidebar',

    'before_widget' => '<div id="%1$s" class="widget %2$s">',

    'after_widget' => '</div>',

    'before_title' => '<div class="widget-title"><h3>',

    'after_title' => '</h3></div>',

));

```

```

// Register additional extra widget areas:

```

```

# $bfa_ata_extra_widget_areas = get_option('bfa_widget_areas');

```

```

    if(isset($bfa_ata['bfa_widget_areas'])) $bfa_ata_extra_widget_areas =
    $bfa_ata['bfa_widget_areas'];

```

```

    else $bfa_ata_extra_widget_areas = "";

```

```

    if ($bfa_ata_extra_widget_areas != "") {

```

```

        foreach ($bfa_ata_extra_widget_areas as $widget_area) {

            register_sidebar(array(

                'name' => $widget_area['name'],

                'before_widget' => $widget_area['before_widget'],

                'after_widget' => $widget_area['after_widget'],

                'before_title' => $widget_area['before_title'],

                'after_title' => $widget_area['after_title']

            ));

        }

    }

}

```

```

#global $bfa_ata;

```

```

// Load functions

```

```

include_once (TEMPLATEPATH . '/functions/bfa_header_config.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_meta_tags.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_hor_cats.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_hor_pages.php');

```

```

// New WP3 menus:

```

```

include_once (TEMPLATEPATH . '/functions/bfa_new_wp3_menus.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_footer.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_recent_comments.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_popular_posts.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_popular_in_cat.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_subscribe.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_postinfo.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_rotating_header_images.php');

```

```

include_once (TEMPLATEPATH . '/functions/bfa_next_previous_links.php');

include_once (TEMPLATEPATH . '/functions/bfa_post_parts.php');

if (!function_exists('paged_comments'))

    include_once (TEMPLATEPATH . '/functions/bfa_custom_comments.php');


// Since 3.5.2: JSON for PHP 4 & 5.1:
if (!function_exists('json_decode')) {

    include_once (TEMPLATEPATH . '/functions/JSON.php');

    function json_encode($data) { $json = new Services_JSON(); return( $json->encode($data) ); }

    function json_decode($data) { $json = new Services_JSON(); return( $json->decode($data) ); }

}

function toArray($data) {

    if (is_object($data)) $data = get_object_vars($data);

    return is_array($data) ? array_map(__FUNCTION__, $data) : $data;

}


// old, proprietary bodyclasses() of Atahualpa. Usage: bodyclasses()

// include_once (TEMPLATEPATH . '/functions/bfa_bodyclasses.php');

// new, default Wordpress body_class(). usage: body_class()

// include only in WP 2.3 - WP 2.7 . From WP 2.8 on it is a core Wordpress function:

/*

if (!function_exists('body_class'))

    include_once (TEMPLATEPATH . '/functions/bfa_body_class.php');

*/


// For plugin "Sociable":

if (function_exists('sociable_html'))

```

```

include_once (TEMPLATEPATH . '/functions/bfa_sociable2.php');

// "Find in directory" function, needed for finding header images on WPMU
if (file_exists(ABSPATH."/wpmu-settings.php"))
    include_once (TEMPLATEPATH . '/functions/bfa_m_find_in_dir.php');

// CSS for admin area
include_once (TEMPLATEPATH . '/functions/bfa_css_admin_head.php');
// Add the CSS to the <head>...</head> of the theme option admin area
add_action('admin_head', 'bfa_add_stuff_admin_head');

include_once (TEMPLATEPATH . '/functions/bfa_ata_add_admin.php');
include_once (TEMPLATEPATH . '/functions/bfa_ata_admin.php');
add_action('admin_menu', 'bfa_ata_add_admin');

// Escape single & double quotes
function bfa_escape($string) {
    $string = str_replace('"', '&#34;', $string);
    $string = str_replace("'", '&#39;', $string);
    return $string;
}

```

```

function footer_output($footer_content) {

    global $bfa_ata;

    $footer_content .= '<br />Powered by <a href="http://wordpress.org/">WordPress</a> &amp; <a
href="http://forum.bytesforall.com/">Atahualpa</a>';

    return $footer_content;

}

```

```

// Move Featured Content Gallery down in script order in wp_head(), so that jQuery can finish before
mootools

```

```

// Since 3.6 this probably won't work because as per the new WP rules wp_head() must be right before
</head>

```

```

function remove_featured_gallery_scripts() {

    remove_action('wp_head', 'gallery_styles');

}

add_action('init','remove_featured_gallery_scripts', 1);

```

```

function addscripts_featured_gallery() {

    if(!function_exists('gallery_styles')) return;

    gallery_styles();

}

add_action('wp_head', 'addscripts_featured_gallery', 12);

```

```

/*

* Add custom header inserts through wp_head

*

* wp_head is supposed to be right before </head>, but html_inserts_header should be after/at the bottom
of wp_head

*

```

@ since 3.6.5

*/

```
function add_html_inserts_header() {
```

```
    global $bfa_ata;
```

```
    if( $bfa_ata['html_inserts_header'] != '' ) bfa_incl('html_inserts_header');
```

```
}
```

```
add_action('wp_head', 'add_html_inserts_header', 20);
```

```
// new comment template for WP 2.7+, legacy template for old WP 2.6 and older
```

```
// Since 3.6.: ToDo: Remove legacy.comments.php after a while. Older WP's won't work anyway
```

```
// with the new WP requirements to REPLACE older functions with newer ones introduced in 2.8 (i.e.  
get_the_author_meta)
```

```
if ( !function_exists('paged_comments') ) {
```

```
    include_once (TEMPLATEPATH . '/functions/bfa_custom_comments.php');
```

```
    function legacy_comments($file) {
```

```
        if( !function_exists('wp_list_comments') )
```

```
            $file = TEMPLATEPATH . '/legacy.comments.php';
```

```
        return $file;
```

```
    }
```

```
    add_filter('comments_template', 'legacy_comments');
```

```
}
```

```
// remove WP default inline CSS for ".recentcomments a" from header
```

```
function remove_wp_widget_recent_comments_style() {
```

```
    remove_filter('wp_head', 'wp_widget_recent_comments_style');
```

```
}
```

```
add_filter( 'wp_head', 'remove_wp_widget_recent_comments_style', 1 );
```

```
/* Remove plugin CSS & JS and include them in the theme's main CSS and JS files
```

```
This will be extended and improved in upcoming versions */
```

```
// remove WP Pagenavi CSS, will be included in css.php
```

```
if (function_exists('wp_pagenavi')) {
```

```
    remove_action('wp_head', 'pagenavi_css');
```

```
}
```

```
// If the plugin Share This is activated, disable its auto-output so we can control it
```

```
// through the Atahualpa Theme Options
```

```
if ( function_exists('akst_share_link') ) {
```

```
    @define('AKST_ADDTOCONTENT', false);
```

```
    @define('AKST_ADDTOFOOTER', false);
```

```
}
```

```
/* EXTERNAL OR INTERNAL CSS & JS, PLUS COMPRESSION & DEBUG */
```

```
// Register new query variables "bfa_ata_file" and "bfa_debug" with Wordpress
```

```
add_filter('query_vars', 'add_new_var_to_wp');
```

```
function add_new_var_to_wp($public_query_vars) {
```

```
    $public_query_vars[] = 'bfa_ata_file';
```

```
    $public_query_vars[] = 'bfa_debug';
```

```
    return $public_query_vars;
```

```
}
```

```
// if debug add/remove info
```

```

if ( function_exists('wp_generator') ) {

    remove_action('wp_head', 'wp_generator');

}

add_action('wp_head', 'bfa_debug');

function bfa_debug() {

    global $bfa_ata, $bfa_ata_version;

    $debug = get_query_var('bfa_debug');

    if ( $debug == 1 ) {

        echo '<meta name="theme" content="Atahualpa ' . $bfa_ata_version . '" />' . "\n";

        if ( function_exists('the_generator') ) {

            the_generator( apply_filters( 'wp_generator_type', 'html' ) );

        }

        echo '<meta name="robots" content="noindex, follow" />' . "\n";

    }

}

```

// redirect the template if new var "bfa_ata_file" or "bfa_debug" exists in URL

```

add_action('template_redirect', 'bfa_css_js_redirect');

add_action('wp_head', 'bfa_inline_css_js');

```

// since 3.4.3

```

function add_js_link() {

    global $bfa_ata, $homeURL;

    if ( $bfa_ata['javascript_external'] == "External" ) { ?>

        <script type="text/javascript" src="<?php echo $homeURL; ?>/?bfa_ata_file=js"></script>

        <?php }

    }

```

```
add_action('wp_head', 'add_js_link');
```

```
function bfa_css_js_redirect() {  
  
    global $bfa_ata;  
  
    $bfa_ata_query_var_file = get_query_var('bfa_ata_file');  
  
    if ( $bfa_ata_query_var_file == "css" OR $bfa_ata_query_var_file == "js" ) {  
  
        include_once (TEMPLATEPATH . '/' . $bfa_ata_query_var_file . '.php');  
  
        exit; // this stops WordPress entirely  
  
    }  
  
    // Since 3.4.7: Import/Export Settings  
  
    if ( $bfa_ata_query_var_file == "settings-download" ) {  
  
        if(isset($_FILES['userfile'])) $uploadedfile = $_FILES['userfile'];  
  
        include_once (TEMPLATEPATH . '/download.php');  
  
        exit; // this stops WordPress entirely  
  
    }  
  
    if ( $bfa_ata_query_var_file == "settings-upload" ) {  
  
        include_once (TEMPLATEPATH . '/upload.php');  
  
        exit; // this stops WordPress entirely  
  
    }  
  
}
```

```
function bfa_inline_css_js() {  
  
    global $bfa_ata;  
  
    $bfa_ata_preview = get_query_var('preview');  
  
    $bfa_ata_debug = get_query_var('bfa_debug');  
  
    if ( $bfa_ata_preview == 1 OR $bfa_ata['css_external'] == "Inline" OR  
  
        ( $bfa_ata_debug == 1 AND $bfa_ata['allow_debug'] == "Yes" ) ) {
```

```

        include_once (TEMPLATEPATH . '/css.php');

    }

    if ( $bfa_ata_preview == 1 OR $bfa_ata['javascript_external'] == "Inline" OR
    ( $bfa_ata_debug == 1 AND $bfa_ata['allow_debug'] == "Yes" ) ) {

        include_once (TEMPLATEPATH . '/js.php');

    }

}

```

```

function bfa_delete_bfa_ata4() {

    check_ajax_referer( "delete_bfa_ata4" );

    if (delete_option('bfa_ata4')) echo '<span style="color:green;font-weight:bold;">Successfully deleted
option \'bfa_ata4\' ...</span>';

    else echo '<span style="color:green;font-weight:bold;">Something went wrong...</span>';

    die();

}

// add_action ( 'wp_ajax_' + [name of "action" in jQuery.ajax, see functions/bfa_css_admin_head.php],
[name of function])

add_action( 'wp_ajax_bfa_delete_bfa_ata4', 'bfa_delete_bfa_ata4' );

```

// Custom Excerpts

```
function bfa_wp_trim_excerpt($text) { // Fakes an excerpt if needed
```

```

    global $bfa_ata, $post;

```

```

    if ( " <> $text ) {

```

```

// an manual excerpt exists, stick on the 'custom read more' and we're done

    $words = preg_split("/\s+/u", $text);

    $custom_read_more = str_replace('%permalink%', get_permalink(),
    $bfa_ata['custom_read_more']);

    if ( get_the_title() == '' ) {

        $custom_read_more = str_replace('%title%', 'Permalink', $custom_read_more);

    } else {

        $custom_read_more = str_replace('%title%', the_title("",FALSE),
    $custom_read_more);

    }

    array_push($words, $custom_read_more);

    $text = implode(' ', $words);

    return $text;

}

$text = get_the_content("");

$words = preg_split ("/\s+/u", $text);

$post_content = $post->post_content;

$post_content_length = count(preg_split("/\s+/u", $post_content));

// Build the excerpt from the post

$text = apply_filters('the_content', $text);

$text = str_replace(']]>', ']]>', $text);

$text = strip_tags($text, $bfa_ata['dont_strip_excerpts']);

$excerpt_length = $bfa_ata['excerpt_length'];

$words = preg_split("/\s+/u", $text, $excerpt_length + 1);

// this is to handle the case where the number of words

```

// in the post equals the excerpt length

```
        if ($post_content_length > $excerpt_length) {  
            array_pop($words);  
//            array_pop($words);  
            $custom_read_more = str_replace('%permalink%', get_permalink(),  
$bfa_ata['custom_read_more']);  
            $custom_read_more = str_replace('%title%', the_title("", "", FALSE), $custom_read_more);  
            array_push($words, $custom_read_more);  
        }  
        $text = implode(' ', $words);  
        return $text;  
  
        return $text;  
    }  
    remove_filter('get_the_excerpt', 'wp_trim_excerpt');  
    add_filter('get_the_excerpt', 'bfa_wp_trim_excerpt');
```

/* Custom widget areas.

Usage:

```
<?php bfa_widget_area([parameters]); ?>
```

Example:

```
<?php bfa_widget_area('name=My widget
area&cells=4&align=1&align_2=9&align_3=7&width_4=700&before_widget=<div id="%1$s" class="header-
widget %2$s">&after_widget=</div>'); ?>
```

Can be used anywhere in templates, and in theme option text areas that allow usage of PHP code.

Available parameters:

Mandatory:

name Name under which all cells of this widget area will be listed at Site Admin -> Appearance -> Widgets

A widget area with 3 cells and a name of "My widget area" creates 3 widget cells which appear as "My widget area 1", "My widget area 2" and "My widget area 3", with the CSS ID's "my_widget_area_1", "my_widget_area_2" and "my_widget_area_3".

Optional:

cells Amount of (table) cells. Each cell is a new widget area.
Default: 1

align Default alignment for all cells. Default: 2 (= center top). 1 = center middle, 2 = center top, 3 = right top, 4 = right middle, 5 = right bottom, 6 = center bottom, 7 = left bottom, 8 = left middle, 9 = left top.

align_1 Alignment for cell 1: align_2, align_3 ... Non-specified cells get the default value of "align", which, if not defined, is 2 (= center top).

width_1 Width of cell 1: width_1, width_2, width_3 ... Non-specified cells get a equal share of the remaining width of the whole table containing all the widget area cells.

before_widget HTML before each widget in any cell of this widget area. Default: <div id="%1\$s" class="widget %2\$s">

after_widget HTML after each widget ... Default: </div>

before_title HTML before the title of each widget in any cell of this widget area: Default:
<div class="widget-title"><h3>

after_title HTML after the title ... Default: </h3></div>

*/

```
function bfa_widget_area($args = '') {  
    global $bfa_ata;  
    $defaults = array(  
        'name' => '',  
        'cells' => 1,  
        'align' => 2,  
        'before_widget' => '<div id="%1$s" class="widget %2$s">',  
        'after_widget' => '</div>',  
        'before_title' => '<div class="widget-title"><h3>',  
        'after_title' => '</h3></div>',  
    );  
  
    $r = wp_parse_args( $args, $defaults );  
    extract( $r, EXTR_SKIP );  
  
    $area_id = strtolower(str_replace(" ", "_", $r['name']));  
  
    # $bfa_widget_areas = get_option('bfa_widget_areas');  
    $bfa_widget_areas = $bfa_ata['bfa_widget_areas'];  
  
    // If there are more than 1 cell, use a table, otherwise just a DIV:  
    if ( $r['cells'] > 1 ) {
```

```
echo '<table id="' . $area_id . '" class="bfa_widget_area" style="table-  
layout:fixed;width:100%" cellpadding="0" cellspacing="0" border="0">';
```

```
// If a width was set for any of the widget area cells:
```

```
# if ( strpos($args,'width_') !== FALSE ) {
```

```
// Since 3.6.7
```

```
$colgroup = 'no'; // If all table cells have the same width, this can be achieved by table-  
layout:fixed alone, without the colgroup element.
```

```
// Check if any of the cells have a set width
```

```
for ( $i = 1; $i <= $r['cells']; $i++ ) {
```

```
    if ( array_key_exists('width_' . $i, $r) AND !empty($r['width_' . $i]) ) {
```

```
        $colgroup = 'yes';
```

```
    }
```

```
}
```

```
if ($colgroup == 'yes') {
```

```
    echo "\n<colgroup>";
```

```
    for ( $i = 1; $i <= $r['cells']; $i++ ) {
```

```
        echo '<col';
```

```
        $current_width = "width_" . $i;
```

```
        if ( $r[$current_width] ) {
```

```
            echo ' style="width:' . $r[$current_width] . 'px";
```

```
        }
```

```
        echo ' />';
```

```
    }
```

```
    echo "</colgroup>";
```

```
}
```

```
echo "<tr>";
```

```
for ( $i = 1; $i <= $r['cells']; $i++ ) {
```

```
    $current_name = $r['name'] . ' ' . $i;
```

```
    $current_id = $area_id . ' ' . $i;
```

```
    $current_align = "align_" . $i;
```

```
    echo "\n" . ' <td id="' . $current_id . '"';
```

```
    if ( $r[$current_align] )
```

```
        $align_type = $r["$current_align"];
```

```
    else
```

```
        $align_type = $r['align'];
```

```
    echo bfa_table_cell_align($align_type) . ">";
```

```
// Register widget area
```

```
$this_widget_area = array(
```

```
    "name" => $current_name,
```

```
    "before_widget" => $r['before_widget'],
```

```
    "after_widget" => $r['after_widget'],
```

```
    "before_title" => $r['before_title'],
```

```
    "after_title" => $r['after_title']
```

```
);
```

```

        // Display widget area

        dynamic_sidebar("$current_name");


        echo "\n</td>";


        $bfa_widget_areas[$current_name] = $this_widget_area;
    }


    echo "\n</tr></table>";


} else {


    // If only 1 widget cell, use a DIV instead of a table
    echo '<div id="' . $area_id . '" class="bfa_widget_area">';


    // Add new widget area to existing ones
    $this_widget_area = array(
        "name" => $r['name'],
        "before_widget" => $r['before_widget'],
        "after_widget" => $r['after_widget'],
        "before_title" => $r['before_title'],
        "after_title" => $r['after_title']
    );


    // Display widget area
    dynamic_sidebar($r['name']);

```

```
echo '</div>';
```

```
$current_name = $r['name'];
```

```
$bfa_widget_areas[$current_name] = $this_widget_area;
```

```
}
```

```
# update_option("bfa_widget_areas", $bfa_widget_areas);
```

```
$bfa_ata['bfa_widget_areas'] = $bfa_widget_areas;
```

```
update_option('bfa_ata4', $bfa_ata);
```

```
}
```

```
function bfa_table_cell_align($align_type) {
```

```
    switch ($align_type) {
```

```
        case 1: $string = 'align="center" valign="middle"'; break;
```

```
        case 2: $string = 'align="center" valign="top"'; break;
```

```
        case 3: $string = 'align="right" valign="top"'; break;
```

```
        case 4: $string = 'align="right" valign="middle"'; break;
```

```
        case 5: $string = 'align="right" valign="bottom"'; break;
```

```
        case 6: $string = 'align="center" valign="bottom"'; break;
```

```
        case 7: $string = 'align="left" valign="bottom"'; break;
```

```
        case 8: $string = 'align="left" valign="middle"'; break;
```

```
        case 9: $string = 'align="left" valign="top"';
```

```
    }
```

```
        return $string;
    }
}
```

// Since 3.4.3: Delete Widget Areas

```
function bfa_ata_reset_widget_areas() {
    global $bfa_ata;

    check_ajax_referer( "reset_widget_areas" );

    $delete_areas = $_POST['delete_areas'];

    # $current_areas = get_option('bfa_widget_areas');

    $current_areas = $bfa_ata['bfa_widget_areas'];

    foreach ($delete_areas as $area_name) {
        unset($current_areas[$area_name]);
    }

    # update_option('bfa_widget_areas', $current_areas);

    $bfa_ata['bfa_widget_areas'] = $current_areas;

    update_option('bfa_ata4', $bfa_ata);

    echo 'Custom widget areas deleted...';

    die();
}

// add_action ( 'wp_ajax_' + [name of "action" in jQuery.ajax, see functions/bfa_css_admin_head.php],
[name of function])

add_action( 'wp_ajax_reset_bfa_ata_widget_areas', 'bfa_ata_reset_widget_areas' );
```

// Since 3.6.5: Import Settings

```
function bfa_ata_import_settings() {
    global $bfa_ata;
```

```

check_ajax_referer( "import_settings" );

// was encoded with encodeURIComponent in bfa_css_admin_head.php

// $import_options = rawurldecode($_POST['options']);

$import_options = stripslashes($_POST['ataoptions']);

// Since 3.5.2, use JSON

if ( json_decode($import_options) != NULL AND strpos($import_options, 'use_bfa_seo') !== FALSE ) {

    update_option('bfa_ata4', json_decode($import_options, TRUE));

    echo "<strong><span style='color:green>Successfully imported </span></strong><br />";

// Probably not a valid settings file:

} else {

    echo "<strong><span style='color:red>Sorry, but doesn't appear

        to be a valid Atahualpa Settings File.</span></strong>";

    #print_r($import_options);

}

die();

}

// add_action ( 'wp_ajax_' + [name of "action" in jQuery.ajax, see functions/bfa_css_admin_head.php],
[name of function])

add_action( 'wp_ajax_import_settings', 'bfa_ata_import_settings' );

/* CUSTOM BODY TITLE and meta title, meta keywords, meta description */

```

```

if(isset($bfa_ata['page_post_options'])) {

    if ($bfa_ata['page_post_options'] == "Yes") {

        /* Use the admin_menu action to define the custom boxes */

        if (is_admin())

            add_action('admin_menu', 'bfa_ata_add_custom_box');

        /* Use the save_post action to do something with the data entered */

        add_action('save_post', 'bfa_ata_save_postdata');

    }

}

```

```

/* Use the publish_post action to do something with the data entered */

```

```

#add_action('publish_post', 'bfa_ata_save_postdata');

```

```

#add_action('pre_post_update', 'bfa_ata_save_postdata');

```

```

/* Adds a custom section to the "advanced" Post and Page edit screens */

```

```

function bfa_ata_add_custom_box() {

```

```

    if( function_exists( 'add_meta_box' ) ) {

```

```

        add_meta_box( 'bfa_ata_sectionid', __( 'Atahualpa Post Options', 'atahualpa' ),

```

```

            'bfa_ata_inner_custom_box', 'post', 'normal', 'high' );

```

```

        add_meta_box( 'bfa_ata_sectionid', __( 'Atahualpa Page Options', 'atahualpa' ),

```

```

            'bfa_ata_inner_custom_box', 'page', 'normal', 'high' );

```

```

    } else {

```

```

        add_action('dbx_post_advanced', 'bfa_ata_old_custom_box' );

```

```

        add_action('dbx_page_advanced', 'bfa_ata_old_custom_box' );
    }
}

```

```

}

}

/* Prints the inner fields for the custom post/page section */

function bfa_ata_inner_custom_box() {

    global $post;

    // Use nonce for verification

    echo '<input type="hidden" name="bfa_ata_noncename" id="bfa_ata_noncename" value="'.
wp_create_nonce( plugin_basename(__FILE__) ). "' />';

    // The actual fields for data entry

    $thePostID = $post->ID;

    $post_id = get_post($thePostID);

#    $title = $post_id->post_title;

    $title = ($post_id->post_title !== 'Auto Draft') ? $post_id->post_title : "";

    $body_title = get_post_meta($post->ID, 'bfa_ata_body_title', true);

    if ( $body_title == "" ) {

        $body_title = $title;

    }

    $display_body_title = get_post_meta($post->ID, 'bfa_ata_display_body_title', true);

    $body_title_multi = get_post_meta($post->ID, 'bfa_ata_body_title_multi', true);

    if ( $body_title_multi == "" ) {

```

```

        $body_title_multi = $title;
    }

    $meta_title = get_post_meta($post->ID, 'bfa_ata_meta_title', true);

    $meta_keywords = get_post_meta($post->ID, 'bfa_ata_meta_keywords', true);

    $meta_description = get_post_meta($post->ID, 'bfa_ata_meta_description', true);


    echo '<table cellpadding="5" cellspacing="0" border="0" style="table-layout:fixed;width:100%">';

    echo '<tr><td style="text-align:right;padding:2px 5px 2px 2px"><input
id="bfa_ata_display_body_title" name="bfa_ata_display_body_title" type="checkbox" '.
($display_body_title == 'on' ? ' CHECKED' : '') . ' /></td><td>Check to <strong>NOT</strong> display the Body
Title on Single Post or Static Pages</td></tr>';

    echo '<tr><td style="text-align:right;padding:2px 5px 2px 2px"><label for="bfa_ata_body_title">' .
__("Body Title Single Pages", 'atahualpa') . '</label></td>';

    echo '<td><input type="text" name="bfa_ata_body_title" value="" . $body_title . "" size="70"
style="width:97%" /></td></tr>';

    echo '<tr><td style="text-align:right;padding:2px 5px 2px 2px"><label
for="bfa_ata_body_title_multi">' . __("Body Title Multi Post Pages", 'atahualpa') . '</label></td>';

    echo '<td><input type="text" name="bfa_ata_body_title_multi" value="" . $body_title_multi . ""
size="70" style="width:97%" /></td></tr>';


    echo '<colgroup><col style="width:200px"><col></colgroup>';

    echo '<tr><td style="text-align:right;padding:2px 5px 2px 2px"><label for="bfa_ata_meta_title">' .
__("Meta Title", 'atahualpa') . '</label></td>';

    echo '<td><input type="text" name="bfa_ata_meta_title" value="" .
$meta_title . "" size="70" style="width:97%" /></td></tr>';


    echo '<tr><td style="text-align:right;padding:2px 5px 2px 2px"><label
for="bfa_ata_meta_keywords">' . __("Meta Keywords", 'atahualpa') . '</label></td>';

    echo '<td><input type="text" name="bfa_ata_meta_keywords" value="" .
$meta_keywords . "" size="70" style="width:97%" /></td></tr>';

```

```
        echo '<tr><td style="text-align:right;vertical-align:top;padding:5px 5px 2px 2px"><label
for="bfa_ata_meta_description">' . __("Meta Description", 'atahualpa' ) . '</label></td>';
```

```
        echo '<td><textarea name="bfa_ata_meta_description" cols="70" rows="4"
style="width:97%">'. $meta_description.'</textarea></td></tr>';
```

```
    echo '</table>';
```

```
}
```

```
/* Prints the edit form for pre-WordPress 2.5 post/page */
```

```
function bfa_ata_old_custom_box() {
```

```
    echo '<div class="dbx-b-ox-wrapper">' . "\n";
```

```
    echo '<fieldset id="bfa_ata_fieldsetid" class="dbx-box">' . "\n";
```

```
    echo '<div class="dbx-h-andle-wrapper"><h3 class="dbx-handle">' .
```

```
        __('Body copy title', 'atahualpa' ) . "</h3></div>";
```

```
    echo '<div class="dbx-c-ontent-wrapper"><div class="dbx-content">';
```

```
// output editing form
```

```
bfa_ata_inner_custom_box();
```

```
// end wrapper
```

```
    echo "</div></div></fieldset></div>\n";
```

```
}
```

```

/* When the post is saved, save our custom data */

function bfa_ata_save_postdata( $post_id ) {

    /* verify this came from the our screen and with proper authorization,
    because save_post can be triggered at other times */

    if ( !wp_verify_nonce( $_POST['bfa_ata_noncename'], plugin_basename(__FILE__) ) ) {

        return $post_id;

    }

    if ( 'page' == $_POST['post_type'] ) {

        if ( !current_user_can( 'edit_page', $post_id ) )

            return $post_id;

    } else {

        if ( !current_user_can( 'edit_post', $post_id ) )

            return $post_id;

    }

    // Save the data

    $new_body_title = $_POST['bfa_ata_body_title'];

    $new_display_body_title = !isset($_POST["bfa_ata_display_body_title"]) ? NULL :
$_POST["bfa_ata_display_body_title"];

    $new_body_title_multi = $_POST['bfa_ata_body_title_multi'];

    $new_meta_title = $_POST['bfa_ata_meta_title'];

    $new_meta_keywords = $_POST['bfa_ata_meta_keywords'];

```

```
$new_meta_description = $_POST['bfa_ata_meta_description'];
```

```
update_post_meta($post_id, 'bfa_ata_body_title', $new_body_title);
```

```
update_post_meta($post_id, 'bfa_ata_display_body_title', $new_display_body_title);
```

```
update_post_meta($post_id, 'bfa_ata_body_title_multi', $new_body_title_multi);
```

```
update_post_meta($post_id, 'bfa_ata_meta_title', $new_meta_title);
```

```
update_post_meta($post_id, 'bfa_ata_meta_keywords', $new_meta_keywords);
```

```
update_post_meta($post_id, 'bfa_ata_meta_description', $new_meta_description);
```

```
}
```

```
// Since 3.4.3: Add Spam and Delete links to comments
```

```
/* Since 3.4.7 deactivated because it's causing issues see  
http://forum.bytesforall.com/showthread.php?t=6742
```

```
function delete_comment_link($id) {
```

```
    if (current_user_can('edit_post')) {
```

```
        echo ' | <a href="'.admin_url("comment.php?action=cdc&c=$id").">Delete</a> ';
```

```
        echo ' | <a href="'.admin_url("comment.php?action=cdc&dt=spam&c=$id").">Spam</a>';
```

```
    }
```

```
}
```

```
*/
```

```
// Add "in-cat-catname" to body_class of single post pages
```

```
/*
```

```
function add_cats_to_body_class($classes='') {
```

```
    if (is_single()) {
```

```

        global $post;

        $categories = get_the_category($post->ID);

        foreach ($categories as $category) {

            $classes[] = 'in-cat-' . $category->category_nicename;

        }

    }

    return $classes;

}

add_filter('body_class', 'add_cats_to_body_class');

*/

```

```

if ( function_exists( 'add_theme_support' ) ) { // Added in 2.9

```

```

    // Since 3.4.5: WP 2.9 thumbnails support:

    add_theme_support( 'post-thumbnails' );

    if ( $bfa_ata['post_thumbnail_crop'] == "Yes" )

        set_post_thumbnail_size( $bfa_ata['post_thumbnail_width'],
    $bfa_ata['post_thumbnail_height'], true );

    else set_post_thumbnail_size( $bfa_ata['post_thumbnail_width'],
    $bfa_ata['post_thumbnail_height'] );

    add_image_size( 'single-post-thumbnail', 400, 9999 ); // Permalink thumbnail size


    // Since 3.5.4:

    add_theme_support('automatic-feed-links');

}

```

// Since 3.4.7: Import/Export Settings

```
function bfa_import_settings_now() {  
    check_ajax_referer( "import_bfa_settings" );  
    $new_options = maybe_unserialize(bfa_file_get_contents($_FILES['userfile']['tmp_name']));  
    update_option('bfa_new_test', $new_options);  
    die();  
}  
  
// add_action ( 'wp_ajax_' + [name of "action" in jQuery.ajax, see functions/bfa_css_admin_head.php],  
[name of function])  
  
add_action( 'wp_ajax_import_bfa_settings_now', 'bfa_import_settings_now' );
```

// Since 3.5.2: New menu system in WP 3

```
if (function_exists('register_nav_menus')) {  
    add_action( 'init', 'register_new_menus' );  
    function register_new_menus() {  
        register_nav_menus(  
            array(  
                'menu1' => __( 'Menu 1' ),  
                'menu2' => __( 'Menu 2' )  
            )  
        );  
    }  
}
```

// Since 3.5.4: Add odd/even class to WP's post_class

```
add_filter ( 'post_class' , 'bfa_post_class' );
```

```
global $current_class;
```

```
$current_class = 'odd';
```

```
function bfa_post_class ( $classes ) {
```

```
    global $current_class;
```

```
    $classes[] = $current_class;
```

```
    $current_class = ($current_class == 'odd') ? 'even' : 'odd';
```

```
    return $classes;
```

```
}
```

```
// Since 3.6: Use stream wrapper instead of eval to include user code. Not used anymore since 3.6.5
```

```
class bfa_VariableStream {
```

```
    var $position;
```

```
    var $varname;
```

```
function stream_open($path, $mode, $options, &$opened_path)
```

```
{
```

```
    $url = parse_url($path);
```

```
    $this->varname = $url["host"];
```

```
    $this->position = 0;
```

```
    return true;
```

```
}
```

```
function stream_read($count)
```

```
{
```

```
$ret = substr($GLOBALS[$this->varname], $this->position, $count);  
$this->position += strlen($ret);  
return $ret;  
}
```

```
function stream_write($data)  
{  
    $left = substr($GLOBALS[$this->varname], 0, $this->position);  
    $right = substr($GLOBALS[$this->varname], $this->position + strlen($data));  
    $GLOBALS[$this->varname] = $left . $data . $right;  
    $this->position += strlen($data);  
    return strlen($data);  
}
```

```
function stream_tell()  
{  
    return $this->position;  
}
```

```
function stream_eof()  
{  
    return $this->position >= strlen($GLOBALS[$this->varname]);  
}
```

```
function stream_seek($offset, $whence)  
{  
    switch ($whence) {
```

case SEEK_SET:

if (\$offset < strlen(\$GLOBALS[\$this->varname]) && \$offset >= 0) {

 \$this->position = \$offset;

 return true;

} else {

 return false;

}

break;

case SEEK_CUR:

if (\$offset >= 0) {

 \$this->position += \$offset;

 return true;

} else {

 return false;

}

break;

case SEEK_END:

if (strlen(\$GLOBALS[\$this->varname]) + \$offset >= 0) {

 \$this->position = strlen(\$GLOBALS[\$this->varname]) + \$offset;

 return true;

} else {

 return false;

}

break;

```

        default:

            return false;

        }

    }

    function stream_stat()

    {

        return array('size' => strlen($GLOBALS[$this->varname]));

    }

    function url_stat()

    {

        return array();

    }

}

// Since 3.6: Register stream wrapper 'bfa'
stream_wrapper_register("bfa", "bfa_VariableStream")
    or die("Failed to register new stream protocol 'bfa'");

// Since 3.6: Make sure $GLOBALS has all values
foreach($bfa_ata as $key => $value) {

    $GLOBALS[$key] = $value;

}

// Since 3.6: New variables using newer WP functions

```

```
$templateURI = get_template_directory_uri();
```

```
$homeURL = home_url();
```

```
// Since 3.6: Include Javascripts here and with wp_enqueue instead of header.php
```

```
$isIE6 = (strpos($_SERVER['HTTP_USER_AGENT'], 'MSIE 6.') !== FALSE);
```

```
if ( !is_admin() ) {
```

```
    wp_enqueue_script('jquery');
```

```
    if ($bfa_ata['pngfix_selectors'] != "" AND $isIE6 = TRUE)
```

```
    {
```

```
        wp_register_script('ddroundies', $templateURI . '/js/DD_roundies.js', false, '0.0.2a' );
```

```
        wp_enqueue_script('ddroundies');
```

```
        add_action('wp_head', 'ddroundiesHead');
```

```
    }
```

```
    if (strpos($bfa_ata['configure_header'], '%image')!== FALSE AND  
$bfa_ata['header_image_javascript'] != "0"
```

```
    AND $bfa_ata['crossslide_fade'] != "0") {
```

```
        wp_register_script('crossslide', $templateURI . '/js/jquery.cross-slide.js', array('jquery'),  
'0.3.2' );
```

```
        wp_enqueue_script('crossslide');
```

```
    }
```

```
}
```

```
// Since 3.6.1: Add ddroundies script in head this way:
```

```
function ddroundiesHead() {
```

```

        global $bfa_ata;

        echo '

<!--[if IE 6]>

<script type="text/javascript">DD_roundies.addRule("'" . $bfa_ata['pngfix_selectors'] . "'");</script>

<![endif]-->

';

}

// Since 3.6: Content Width

if ( ! isset( $content_width ) )

    $content_width = 640;

// editor style, custom background & custom image header not activated yet. background & header work

// Atahualpa has most of this - confusing to have 2 places to set header images etc. ?

/*

add_editor_style();

add_custom_background();


define('HEADER_TEXTCOLOR', 'ffffff');

define('HEADER_IMAGE', '%s/images/header/header6.jpg');

define('HEADER_IMAGE_WIDTH', 775);

define('HEADER_IMAGE_HEIGHT', 200);


function header_style() {

    ?><style type="text/css">

        #header {

```

```

        background: url(<?php header_image(); ?>);

    }

</style><?php
}

function admin_header_style() {

    ?><style type="text/css">

        #heading {

            width: <?php echo HEADER_IMAGE_WIDTH; ?>px;

            height: <?php echo HEADER_IMAGE_HEIGHT; ?>px;

        }

    </style><?php
}

add_custom_image_header('header_style', 'admin_header_style');

register_default_headers( array(

    'header6' => array(

        'url' => '%s/images/header/header6.jpg',

        'thumbnail_url' => '%s/images/header/header6-thumb.jpg',

        'description' => __( 'Raspberry', 'twentyten' )

    ),

    'IMG_1479' => array(

        'url' => '%s/images/header/IMG_1479.jpg',

        'thumbnail_url' => '%s/images/header/IMG_1479-thumb.jpg',

        'description' => __( 'Apple Blossom', 'twentyten' )

    ),

    'IMG_1496.jpg' => array(

```

```

        'url' => '%s/images/header/IMG_1496.jpg',

        'thumbnail_url' => '%s/images/header/IMG_1496-thumb.jpg',

        'description' => __( 'Spring', 'twentyten' )

    )

    ));

*/

// Since 3.6.5: Process or don't process user included PHP code.
function bfa_incl($option) {

    global $bfa_ata;

    // 'extension_loaded' or 'ini_get' aren't always correct according to
http://stackoverflow.com/questions/3383916/how-to-check-whether-suhosin-is-installed

    /*

    ob_start();

    phpinfo();

    $phpinfo = ob_get_contents();

    ob_end_clean();

    if (strpos($phpinfo, "Suhosin") !== FALSE)

    $suhosin = "yes";

    */

    /*

    if($suhosin == "yes") echo $bfa_ata[$option]

    else include('bfa://' . $option);

    */

    $result = bfa_parse_widget_areas($bfa_ata[$option]);

```

```

        echo $result;
    }

/*

function bfa_parse_widget_areas($content) {

    // For custom widget areas

    //$widget_option_matches = array();

    if ( strpos($content, '<?php bfa_widget_area') !== FALSE ) {

        $widget_options = preg_match("/(.*)<?php
bfa_widget_area\('(.*?)'\)(.*)\?>(.*)/im",
        $content,$widget_option_matches);

        parse_str($widget_option_matches[2], $widget_option_array);

        ob_start();

        bfa_widget_area($widget_option_array);

        $widget_area = ob_get_contents();

        ob_end_clean();

        $content = preg_replace("/(.*)<?php bfa_widget_area(.*)\('(.*?)'\)(.*)\?>(.*)/im",
        "\${1}" .
        $widget_area . "\${5}", $content);
    }

```

```

        # echo $bfa_ata[$option];

        return $content; // . $widget_option_matches[2];
    }

    */

function bfa_parse_widget_areas($content) {

    if ( strpos($content,'<?php bfa_widget_area') !== FALSE ) {

        $content = preg_replace_callback("<\?php
bfa_widget_area(.*)\(((.*)'(.*)'(.*)\)(.*)\?>|", "bfa_parse_widget_areas_callback", $content);

    }

    return $content;
}

// Callback for preg_replace_callback
function bfa_parse_widget_areas_callback($matches) {

    parse_str($matches[3], $widget_options);

    ob_start();

    bfa_widget_area($widget_options);

    $widget_area = ob_get_contents();

    ob_end_clean();

```

```
return $widget_area;
```

```
}
```

```
?>
```