

Les règles de UP - version 2.8

- tous les fichiers d'une action dans le dossier de l'action.
L'ajout du dossier installe l'action, sa suppression la supprime.
- l'utilisateur dispose toujours d'une documentation.
Sans fichier up/xx-FR.ini, la doc est récupérée dans le script action.
- les shortcodes doivent pouvoir être saisis en mode wysiwyg
- Tous les noms (actions, fichiers et options) en anglais.
Traduction ou abréviation dans fichier dico.ini de l'action
- Utiliser `{up upactionlist | csv}` pour attribuer le nom des options
- Préfixer le nom des actions proches (ex: image_gallery, image_magnify)

Initialisation d'une action

```
// doit-on exécuter l'action ?
if ($this->filter_ok($options['filter']) != true) { return ''; }
```

Chargement ressources

```
// charge css ou js avec gestion dossier custom de l'action si path vide
$this->load_file($ficpath, $options=array(), $attributes=array());
$this->load_css_head($code_css, $id=null); // charge code CSS dans le head. gère #id
$this->load_custom_code_head($code); // charge code libre dans le head
$this->load_js_file_body($ficpath); // charge fichier JS après contenu article
```

Chemin et URL

```
// ajoute custom à $path si $file existe dans le dossier.
$file = $this->get_custom_path($file, $path=dossier_action);
$path = $this->get_asset_path($url);
// ajoute racine si besoin: images/foo.png -> /rootFolder/images/foo.png
$url = $this->get_url_relative($url, $urlencode=false);
// retourne URL sous forme absolue: images/foo.png -> https://site.fr/images/foo.png
$url = $this->get_url_absolute($url, $urlencode=false);
$this->on_server($url); // true si URL sur le serveur
```

HTML

```
// Charge flux HTML avec timeout pour éviter blocage
$str = $this->get_html_contents($url, $timeout=10, $url2='');
// récupère array avec les attributs d'une balise HTML
$attr = $this->get_attr_tag($tag, $force='id,class,style');
```

Nettoyage

```
// $tags: 0=a l'identique, 1=neutralisecode HTML ou liste tags autorisés ('a,img,b')
$content = clean_html($content, $tags=false, $forceEOL=false)
```

Les options

Pour mémoire, un passage obligé `ctrl_options($options_def, $js_options_def='')`

```
if (!empty($options['nom_option'])) { ... } // Tester si une option est vrai ou fausse

$this->load_infile($file, $sections=false, $alert=true) // array avec contenu fichier
```

Consolider

```
// convertir bcode en HTML
$options['title']=$this->get_bcode($options['title']);
$code=$this->get_code($input); // '[text&#226;];\]' en [text<{}]
// affecte $val si $option='' (true) ex: ('mp4', 'filename.mp4')
$this->set_option_user_if_true($option, $val);
// vérifie si argument dans liste sans tenir compte de la case, sinon retourne ler.
$arg = $this->ctrl_argument($arg, $authorized_list, $debug=true);
```

Images

```
// vérifier si unité existe, sinon affecte la première de la liste
$options['size']=$this->ctrl_unit($options['size'],'px,%rem');
// conversion px<->rem
$arr=convert_size($size, $unit_target='px') // $arr[0]=(int)$val, [1]=$unit_target
// Retourne UNC sans chemins, extensions, underscore, tirt (2+tirets=1), compteur
$str=link_humanize($unc, $capitalize=true);
```

Le contenu

```
// vérifie si contenu entre shortcode. Affiche message Joomla
if (!$this->ctrl_content_exists()) { return false; }
```

Récupérer les parties séparées par (===) dans un array

```
if ($this->ctrl_content_parts($this->content)) {
    $list = $this->get_content_parts($this->content);
}
```

Shortcodes internes

```
// retourne un tableau avec tous les shortcodes avec $keyword
$this->get_content_shortcode($content, $keyword='.')
```

CSV

```
// $cleanTags:liste tags non supprimés, $bcode:liste bbbcodes.
$arr=$this->get_content_csv($content, $cleanTags='', $bcode='').
```

Nettoyage avant analyse regex

```
$content = str_replace(PHP_EOL, '', $content);
```

Préparer attributs HTML

```
$attr['id']=$options['id'];
// ventile vers $attr['class'] et $attr['style']
$this->get_attr_style($attr, 'bg-red:color:#FFF', $options['class'], ...);
```

JS - jQuery

Consolider les paramètres

```
// les options saisies par l'utilisateur pour envoi sélectif au script JS
$js_options = $this->only_using_options($js_options_def, $options_user=null);
// conversion en chaîne json (mode1=normal, mode2=sans guillemets, mode3=crochets array)
$js_params = $this->json_arrtostr($js_options, $mode=1, $bracket=true);
// le contenu d'un fichier json dans un array
$arr=$this->get_jsontoarray($filename, $ficpath = '')
// Analyse une liste d'options non gérées par l'action pour un script JS
$arr=$this->params_decode($str, $sep_param=',', $sep_key=':', $quote='', $echap='\\');
```

```
$code = '$("#" . $options['id'] . ').xxx(';
$code .= $js_params;
$code .= ');';
$this->load_jquery_code($code,$in_head=false); // retourne code jquery encapsulé
```

```
$this->load_js_code($code, $in_head=true) // dans head ou retourne <script>code JS</script>
```

Construire le HTML en retour

```
$html[] = $this->set_attr_tag('ul', $attr);
$html[] = $this->set_attr_tag('li', array(), 'contenu'); // attr vide
$html[] = </ul>
return implode(PHP_EOL, $html);
```

Blocs imbriqués.

On construit à partir du milieu.

Un underscore devant le tag le rend optionnel si aucun attribut, on retourne le contenu.

```
$out = $this->set_attr_tag('div', $attr_niv3, 'niveau-3')
$out = $this->set_attr_tag('div', $attr_niv2, $out)
$out = $this->set_attr_tag('div', $attr_niv1, $out)
return $out
```

traduction

```
$str=$this->lang('en=yes;fr=oui');
// motclé dans up/language/xx-XX/xx-XX.ini ou up/actions/action/up/xx-XX.ini
$str=$this->trad_keyword($key);
// définir la langue affichée. $this->set_locale($options['date-locale']);
$this->set_locale($tag='');
```

Messages

```
// ajout message dans la file des messages de Joomla
$this->msg_error($msg);
$this->msg_info($msg; $title='');
// retourne texte très visible avec référence action si $infoUP
if (true) return $this->info_debug($txt, $infoUP=true);
```

Divers sur les chaines

```
$str=$this->preg_string('!alt="(.*)"!i', '<img alt="label">'); // retourne label
$array=$this->strtoarray('un:1,2;deux,$row=', $col=':'); // ['un']>1 [2]>deux'
$str=$this->replace($old,$new,$src,$nb=1); // remplace nb occurrences de $old par $new
```

Ajouter une chaîne à une autre avec gestion séparateur, préfixe et suffixe

```
$str = str_append($str, $add, $sep=' ', $prefix='', $suffix='');
add_str(&$str, $add, $sep=' ', $prefix='', $suffix='');
add_class(&$str, $newclass, $prefix='');
add_style(&$str, $property, $val);
```

Interne UP

```
$str=$this->get_action_pref($key, $default=null); // clé dans plugin
$arr=$this->up_actions_list(); // liste des actions
$str->$this->get_dico_synonym($key); // liste synonyme (sep virgule)
$str=$this->shortcode2code($str); // shortcode neutralisé pour affichage
$str=$this->up_action_infos($action,$lang=null); // info action pour doc
$arr=$this->up_action_options($action,$csv=false,$lang=null); // options pour doc
$str=$this->up_prefset_list($action, $full=true); // Liste des jeux d'options
$str=$this->up_help_text(); // contenu custom/help.txt
$this->set_demopage($webpage=''); // emplacement demo de l'action
```

Propriétés disponibles (\$this)

actionPath : chemin dossier action en cours | **upPath** : plugins/content/up | **name** : nom dossier action | **actionUserName** : nom action saisi dans shortcode | **options_user** : array options saisies dans shortcode | **content** : contenu HTML entre shortcodes | **article** : article ou module contenant les shortcodes | **actionprefs** : préférences plugin | **demopage** : URL vers page démo (usehelpsite & urlhelpsite) | **dico** : table correspondance des noms | **J4** : version Joomla >=4

Action privée

UPactionList affiche toutes les actions sauf celles commençant par un underscore.

Pour utiliser le nom sans l'underscore, utiliser dico.ini : nom_action=_nom_action

Utiliser la feuille de style UP

Pour fichier SCSS dans dossier d'une action.

```
@import "../assets/custom/variables"; // version webmaster
@import "../assets/scss/variables"; // UP par défaut
```

Mediaqueries: \$breakpoint-s, \$breakpoint-m **Couleurs**: voir feuille css pour noms FR et GB

Ombres: \$shadow[-max|-bottom|-large|-white] **bordures**: \$up-border-[0|1|2|3]

Espaces: \$up-space-[0|1|2|3]

Les couleurs sont déclarées comme variables CSS (--jaune, --yellow, ...)