

Tigra menus

Catégorie : Themes et templates

Publié par Romu le 29/11/2005

Tigra Menu dans un thème xoops L'objectif de mes recherches était de proposer dans mon thème un menu horizontal avec sous menus verticaux. J'ai ajouté au challenge le besoin d'intégrer le menu utilisateur dans le but de remplacer en un unique espace réduit au minimum les deux blocs généralement positionnés dans la colonne de gauche. En effet il m'a semblé important de proposer, lors de la réalisation de mon thème, le plus d'espace possible au contenu. Après de nombreux essais, j'ai trouvé une solution convenable : Tigra Menu Ce magnifique javascript, après quelques recherches pour son intégration complète dans un thème xoops, répond complètement mes besoins et permet de proposer un menu propre et simple à utiliser. Il en demeure que sa mise en place demande une certaine réflexion. Je prévois immédiatement, il n'utilise pas le menu standard au contenu automatique. Il faut donc adapter le contenu des scripts à chaque évolution dans la liste des options (modules). Ce menu est exploité dans le thème de [WDForge.org](http://www.wdforge.org). Bonne lecture, Romu aka R&B de [WDForge.org](http://www.wdforge.org)

[Tigra Menu dans un thème xoops](#) [Présentation de Tigra menu](#) [Adaptations pour Insertion dans un theme](#) [Menu_item.js](#) [Utilité et description](#) [Intégration au theme](#) [Menu_tpl.js](#) [Utilité et description](#) [Positionnement du menu](#) [Menu.css](#) [Utilité et description](#) [mise en forme](#) [Appel des scripts](#) [Compléments](#) [Messages privés](#) Présentation de Tigra menu Tigra Menu est un Javascript DHTML de qualité pour insérer dans vos pages un menu de belle facture. C'est un produit [SoftComplex](#) disponible à l'adresse suivante :

http://www.softcomplex.com/products/tigra_menu/ Parmi la gamme disponible, nous avons adapté à nos besoins la démonstration " Traditional Blue " qui propose un menu horizontal avec 2 niveau de sous menus actifs en survol et à orientation de navigation configurable. Tigra menu est un JavaScript. Par conséquent, il n'existe pas concrètement dans la page sous forme d'un bloc de balises HTML. Il est calculé dans le navigateur et est y est dessiné sur la page à une position fixe sur la page affichée. Il faut donc prévoir un espace dans la page dont la position est fixe. Cela suppose alors que le thème propose une telle zone dans sa construction. Se reporter au chapitre " positionnement du menu " pour les détails de cette spécificité. Tigra Menu est constitué à l'origine de 4 fichiers :

Menu.js le script de fonctionnement du menu

Menu_item.js script de description des contenu (options) du menu ; reporté dans theme.html pour notre exemple

Menu_tpl.js le script de paramétrage du menu (positions et styles des blocs)

Menu.css feuille de style des blocs du menu. Son contenu peut être reporté dans les feuilles de styles de votre thème (style.css, styleNN.css et styleMac.css). Tigra menu propose une succession de tableaux (variable JavaScript) qui vont constituer les différents niveaux du menu. Ainsi le menu principal est un tableau d'options et de tableaux constituant les sous menus composés eux même d'options et de tableaux etc. Dans notre exemple le menu principal est horizontal, les sous menu sont verticaux et sous l'option parente. Le niveau inférieur est aussi constitué de menu verticaux mais affichés à côté de l'option parente. Nous appellerons bloc une option du menu.



■ Adaptations pour Insertion dans un theme Menu_item.js Utilité et description Ce script contient la description des options du menu. Il crée un tableau MENU_ITEM de tableaux des options. Une option de menu est constituée selon deux structures distinctes § Option individuelle ['Libellé', 'Url destination'], § Option avec sous menu ['Libellé', null, null, options du sous menu], il est possible de faire en sorte que l'option qui crée le sous menu soit aussi source d'un lien en remplaçant le premier null par l'url comme dans l'exemple ci-après. ['Libellé', 'url destination', null, options du sous menu], Intégration au theme Le menuxoops est relativement fixe mais l'on s'aperçoit que l'on peut le rendre 'dynamique' via les variables smarty en reproduisant ainsi le menu utilisateur. Nous avons donc décidé de reporter cette partie de script dans le fichier **theme.php** de notre thème dans la partie HEAD de la page:

-->

```
/* --- menu items --- */
```

```
var MENU_ITEMS = [
```

```
['Accueil', '  
/,null, /* --- option cliquable --- */
```

```
['Module1...', '  
/modules/NomModule1/', null,
```

```
['Page1', '  
/modules/NomModule1/index.php?page=1'], ['Page2','
```

```
/modules/NomModule1/index.php?page=2'],
    ],
    ['Module2...!',
 /modules/NomModule2',null,
 /modules/NomModule2/index.php?page=1'],
 /modules/NomModule2/index.php?page=2'],
    ['Page1',
     ['Page2',
```

```
],
```

```
['AutreBarre', null, null, /* --- option non clicable --- */
```

```
['Module1...!', '
/modules/NomModule1/', null,
```

```
['Page1',
/modules/NomModule1/index.php?page=1'],
/modules/NomModule1/index.php?page=2'],
    ['Page2',
     ],
```

```
['Module2...!',
/modules/NomModule2',null,
/modules/NomModule2/index.php?page=1'],
/modules/NomModule2/index.php?page=2'],
    ['Page1',
     ['Page2',
```

```
],
```

```
],
```

```
if $xoops_isuser}> /* --- Menu utilisateur --- */
```

```
0)> /* --- utilisateur avec message => pointe directement messagerie --- */
```

```
['  
(  
)';  
/viewpmsg.php',null,
```

```
['  
,null,null,
```

```
/* --- Administration --- */
```

```
['Administration','  
/admin.php'],
```

```
['Votre compte','  
/user.php'],
```

```
0}>
```

```
['Messagerie (  
)';  
/viewpmsg.php'],
```

['Messagerie (vide)',
/viewpmsg.php'],

['DÃ©connexion',
/user.php?op=logout'],

],

/* --- Annonyme --- */

['Identification',null,null,

['Connecter',
/user.php'],

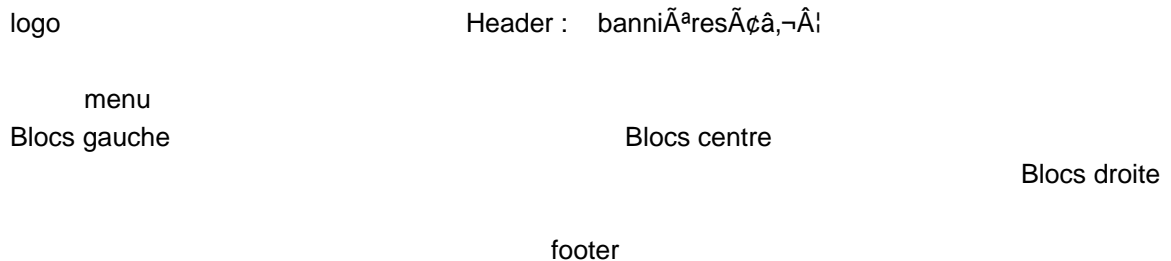
['S'enregistrer',
/register.php'],

],

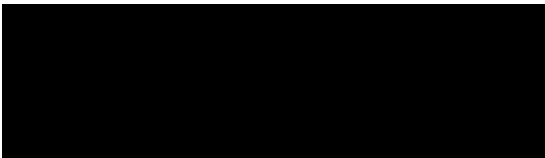
];



Menu_tpl.js Utilité et description C'est ce script qui permet de positionner le menu dans la page et de fixer le déroulement des options en indiquant leurs tailles et positions respectives. Positionnement du menu Rappel de la structure des thèmes xoops : L'organisation des blocs est organisée en 9 zones. Le header de votre theme.html est une bonne idée pour positionner le menu. C'est en tout cas la position que nous lui avons réservée. C'est aussi sans doute l'unique difficulté. En effet, comme précisé en introduction, le menu ne fait pas partie de la page (coté serveur) mais il est dessiné via menu.js dans le navigateur. Cela implique que sa position ne peut être que fixe (il faut donc rendre paramétrable le JavaScript en fonction de la position d'une cellule de table ou de balise div, ce qui ne fait pas partie de mes compétences à l'instant de l'écriture de ce guide). Il faut donc impérativement que votre header soit parfaitement immuable en terme de hauteur. Sinon, le menu va se dessiner à sa position et pourrait recouvrir des informations. Dans notre exemple nous avons procédé comme dans la démonstration avec une page dont le header est structuré comme suit :



La ligne logo et header doit toujours avoir la même hauteur : H (80 px) La colonne de gauche doit toujours avoir la même largeur fixe : X (170 px) Ainsi le menu pourra être dessiné aux coordonnées X,H Pour ce faire nous avons créé une table de deux cellules cet effet avec une hauteur fixes la hauteur des options du menu et positionnée après le header (en encadrement double sur l'illustration précédente). Le première cellule étant aussi fixe la largeur fixe de la colonne gauche de notre thème (X). Ces conjonctions permettent de placer le menu sans effet de bords.



Pour le détail des positions des blocs du menu voir le fichier menu_tpl.js qui fixe les paramètres pour chaque niveau dans la variable JavaScript MENU_POS :

```
var MENU_POS = [ {
  // NIVEAU 0 : MENU HORIZONTAL // Tailles des blocks 'height': 32, 'width': 120, //
  // positions des blocks de menu par rapport à l'origine : // Pour le niveau 0 : par rapport au coin
  // haut gauche de la page // Pour les niveaux suivants : par rapport au coin haut gauche du niveau
  // précédent // (prévoir un espace libre et la position fixe) 'block_top': 81, 'block_left':
  // 171, // positions relatives entre les éléments d'un même niveau // HORIZONTAL :
  // DECLALER D'UN BLOCK A DROITE 'top': 0, 'left': 121, // durée d'affichage/disparition en
  // milli secondes 'hide_delay': 200, 'expd_delay': 200, // Affectation des styles aux blocks :
  // Menu 0 ligne 0 'css': { 'outer': ['m0l0oout', 'm0l0oover'], 'inner': ['m0l0iout',
  // 'm0l0iover'] } },
```

Noter la facilité de positionnement des tableaux. 'block_top' et 'block_left' fixent la position du tableau par rapport à l'élément parent ou la fenêtre, 'top' et 'left' fixent les positions des blocks d'un même niveau (dans le même tableau/menu). Ainsi pour que le menu progresse vers la droite (horizontal) : donner 0 à 'top' et 'width'+1 à 'left'. Pour que le menu progresse vers le bas (vertical) : donner 0 à 'left' et 'height'+1 à 'top'.

Menu.css Utilité et description Ce script contient les définitions de style des options de menu. Pour des raisons pratiques nous avons intégré ces définitions dans la feuille de style du thème : **style.css** mise en forme. Pour ce qui est des styles, ils sont séparés en deux couples de définitions : **inner** (partie active de l'option) et **outer** (partie inactive de l'option). En effet si les cases sont plus grandes que le texte surligné (inner), il faut prévoir le comportement du reste de la 'cellule' (outer). Pour chacun de ces parties, il est prévu deux états : **out** (normal) et **over** (survol). Utiliser le jeu des couleurs pour mettre en surbrillance la cellule (inner et outer) survolé (out). L'effet de transparence des cellules normales des sous menus (en cas de recouvrement de texte) via la définition filter :alpha.

```
/* --- menu styles --- */
/*--- level 0 inner (texte menu horizontal) */
.m0l0iout { font-family: Verdana, Tahoma, Geneva, Arial, Helvetica, sans-serif;
font-size: 11px; text-decoration: none; color: #000000; padding: 4px; }
.m0l0iover { font: italic Verdana Tahoma Geneva Arial Helvetica sans-serif;
font-size: x-small; text-decoration: none; color: #ff6600; padding: 4px; }
/*--- level 0 outer (options menu horizontal) */
.m0l0oout { text-decoration:none; border : 1px solid #FFFFFF; background: #F4F4F4; }
.m0l0oover { text-decoration:none; border : 1px solid #FFFFFF; background: #99CCFF; }
/*--- level 1 inner (texte menus verticaux 1) */
.m0l1iout { font-family: Verdana, Tahoma, Geneva, Arial, Helvetica,
sans-serif; font-size: x-small; text-decoration: none; color: #000000; padding: 4px; }
.m0l1iover { font-family: Verdana, Tahoma, Geneva, Arial, Helvetica, sans-serif;
font-size: x-small; text-decoration: none; color: #FF6600; padding: 4px; }
/*--- level 1 outer (options menus verticaux 1) */
.m0l1oout { text-decoration:none; border : 1px solid white; background:
#F4F4F4; }
.m0l1oover { text-decoration:none; border : 1px solid white; background:
#99CCFF; }
/* level 2 inner (texte menus verticaux 2) */
.m0l2iover { font-family: Verdana, Tahoma, Geneva, Arial, Helvetica, sans-serif;
font-size: x-small; text-decoration: none; color: #FF6600; padding: 4px; }
/* level 2 outer (options menus verticaux 2) */
.m0l2oout { text-decoration:none; border : 1px solid white; background: #F4F4F4; }
.m0l2oover { text-decoration:none; border : 1px solid white; background: #99CCFF; }
```

Appel des scripts Vous l'aurez compris, nous avons ventilé la définition de la structure du menu et de ses styles dans les fichiers de notre thème. Néanmoins il nous faut encore faire appel aux scripts de lancement. Si SoftComplex indique qu'il faille lancer menu.js en toute fin de page (avant la balise /BODY), notre expérience nous montre qu'il est plus intéressant de positionner cet appel juste avant le code html de la balise TABLE ou DIV qui va délimiter la zone fixe du menu dans la page. En effet, si la suite de la page met un certain temps à se charger le menu sera déjà disponible :

```

<!--> var MENU_ITEMS = [ <!--> ];
```

À /menu.js"> /menu_tpl.js">
À! ■

Compléments Messages privés La smarty est issue d'un hack du fichier **header.php**. Elle contient simplement le nombre de messages personnels pour le membre connecté. Il faut alors ajouter ligne 90 de header.php le code suivant :
if (\$xoopsUser != "") { // rb \$pm_handler =& xoops_gethandler('privmessage'); \$criteria = new CriteriaCompo(new Criteria('read_msg', 0)); \$criteria->add(new Criteria('to_userid', \$xoopsUser-