

# Intégrations graphiques Publik - Bug #41734

## amélioration du rendu des champs type tableau en mobile

15 avril 2020 10:24 - Serghei Mihai

<b>Statut:</b>	Fermé	<b>Début:</b>	15 avril 2020
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Thomas Jund (congé, retour le 29/04)	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	Non
<b>Patch proposé:</b>	Oui		
<b>Description</b>			
Aujourd'hui il est impossible de voir complètement le texte saisi.			
<b>Demandes liées:</b>			
Lié à w.c.s. - Development #47510: utiliser un <thead> dans les tableaux long...			<b>Fermé</b> <b>09 octobre 2020</b>

### Révisions associées

#### Révision e536b5f2 - 16 octobre 2020 11:11 - Thomas Jund (congé, retour le 29/04)

scss: make table widgets responsive (#41734)

### Historique

#### #2 - 21 septembre 2020 09:55 - Thomas Jund (congé, retour le 29/04)

- Assigné à mis à Thomas Jund (congé, retour le 29/04)

#### #3 - 25 septembre 2020 17:04 - Thomas Jund (congé, retour le 29/04)

Widgets concernés par le problème d'affichage :

- Tableau
- Tableau de liste
- Tableau de longueur libre

#### #4 - 08 octobre 2020 14:31 - Thomas Jund (congé, retour le 29/04)

- Fichier desktop.png ajouté
- Fichier mobile.png ajouté
- Fichier very-small.png ajouté
- Fichier 0001-scss-make-table-widgets-responsive-41734.patch ajouté
- Fichier 0001-js-make-table-widgets-responsive-41734.patch ajouté
- Tracker changé de Support à Bug
- Statut changé de Nouveau à Solution proposée
- Patch proposé changé de Non à Oui

Proposition :

1. Vérifier si la tableau est plus large que son parent et dans ce cas le fixer à 100%
2. en mobile, positionner les cellules les unes en dessous des autres, récupérer les titres de colonnes et les placer en ::before
3. en very-small, positionner les td::before au dessus du champs

Nécessite du JS.

Une des difficultés fût de jongler entre les différents markup :

- les widgets *tableau* et *tableau de liste* ont les titres des colonnes en `thead tr th` , et les titres des lignes en `tbody tr th:first-child`
- les widgets *tableau de longueur libre* ont les titres des colonnes en `tbody tr:first-child th` et n'ont pas de titre de ligne
  - Ils ont également une balise .widget autour de chaque input

J'ai donc donné la possibilité de passer les sélecteurs des titres en paramètres du script

```
new Responsive_table_widget(table, {  
  col_headers_selector: 'thead th',
```

```
row_headers_selector: 'tbody th'
}))
```

J'ai placé le script dans `common.forms.js`, mais il pourrait se généraliser à combo également.

#### #5 - 08 octobre 2020 14:34 - Thomas Jund (congé, retour le 29/04)

Il faudra peut-être revoir le label du bouton "ajouter une ligne" puisque "ligne" il n'y a plus en mobile.

Ensuite JS ajoute quelques class et des data-attributs. Cette partie pourrait sauter si ces éléments étaient ajoutés depuis le template des widgets. À voir.

#### #6 - 08 octobre 2020 14:45 - Frédéric Péters

C'est quoi le résultat avec le js sur un site qui n'aurait pas les nouvelles css ? (ce qui serait le cas des sites au quotidien pas encore migrés)

Tu peux pointer les modifications dans le balisage qui seraient utiles ?

#### #7 - 09 octobre 2020 09:30 - Thomas Jund (congé, retour le 29/04)

C'est quoi le résultat avec le js sur un site qui n'aurait pas les nouvelles css ? (ce qui serait le cas des sites au quotidien pas encore migrés)

Rien du tout, le JS va ajouter des class, des data-attribut et un "width:100%" sur les tableaux (mais sans effet).

Bref, exactement comme si je n'avais rien fait.

Tu peux pointer les modifications dans le balisage qui seraient utiles ?

1. uniformiser le balisage des "tableau de longueur libre" avec les autres tableau (en plaçant les titres des colonnes dans un `thead`, ça simplifierait le JS et le CSS.
2. Ajouter des titres de lignes aux tableaux de longueur libre, ou à défaut un compteur, information qui serait vraiment utile en mobile. Le compteur peut s'ajouter en CSS only.
3. Ensuite, pour permettre d'alléger la partie JS :
  - Ajouter les data-attributs aux td directement dans les templates (mais là dessus j'hésite, c'est bien aussi que JS le fasse, cela permettrait de l'utiliser sur tout type de tableaux, à l'aide d'un `pk-class`).

#### #8 - 09 octobre 2020 09:43 - Frédéric Péters

Ajouter des titres de lignes

Qu'appelles-tu "titres de lignes" ?

Plus simplement sans doute, si tu peux faire un diff à partir du marqueur actuel pour me pointer les choses :

```
<div class="widget TableListRowsWidget widget-optional" data-field-id="18">
  <div class="title"><label class="field--label" id="form_label_f18" for="form_f18">tableau longueur libre</label></div>
  <div class="content" aria-labelledby="form_label_f18">
    <table>
      <tbody>
        <tr>
          <th>plop</th>
          <th>plip</th>
        </tr>
        <tr>
          <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element0$col0" id="form_f18$element0$col0"></div></div></td>
          <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element0$col1" id="form_f18$element0$col1"></div></div></td>
        </tr>
        <tr>
          <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element1$col0" id="form_f18$element1$col0"></div></div></td>
          <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element1$col1" id="form_f18$element1$col1"></div></div></td>
        </tr>
        ...
      </tbody>
    </table>
  <div class="widget SubmitWidget list-add widget-optional">
```

```

    <div class="content " aria-labelledby="form_label_f18$add_element">
      <button name="f18$add_element" value="Ajouter une ligne">Ajouter une ligne</button>
    </div>
  </div>
</div>
</div>

```

**#9 - 09 octobre 2020 11:16 - Thomas Jund (congé, retour le 29/04)**

```

<div class="widget TableListRowsWidget widget-optional" data-field-id="18">
  <div class="title"><label class="field--label" id="form_label_f18" for="form_f18">tableau longueur libre</label></div>
  <div class="content " aria-labelledby="form_label_f18">
    <table>
      <thead>
        <tr>
          <td></td>
          <th>plop col1</th>
          <th>plip col2</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <th>plop</th>
          <th>plip</th>
          <tr>
            <th>plop row1</th>
            <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element0$col0" id="form_f18$element0$col0"></div></div></td>
            <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element0$col1" id="form_f18$element0$col1"></div></div></td>
          </tr>
          <tr>
            <th>plop row2</th>
            <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element1$col0" id="form_f18$element1$col0"></div></div></td>
            <td><div class="StringWidget widget"><div class="content"><input type="text" name="f18$element1$col1" id="form_f18$element1$col1"></div></div></td>
          </tr>
          ...
        </tbody>
      </table>
      <div class="widget SubmitWidget list-add widget-optional">
        <div class="content " aria-labelledby="form_label_f18$add_element">
          <button name="f18$add_element" value="Ajouter une ligne">Ajouter une ligne</button>
        </div>
      </div>
    </div>
  </div>
</div>

```

**#10 - 09 octobre 2020 11:23 - Frédéric Péters**

Ok mais on n'a pas de libellés pour les lignes sur ce type de tableau, je ne vois pas le sens que ça aurait, de passer par exemple de :

Participants :

Prénom	Nom
Frédéric	Péters
Thomas	Jund

à :

Participants :

	Prénom	Nom
Informations sur un participant:	Frédéric	Péters
Informations sur un participant:	Thomas	Jund

Pour le <thead> ok.

**#11 - 09 octobre 2020 11:30 - Frédéric Péters**

- Lié à Development #47510: utiliser un <thead> dans les tableaux longueur libre ajouté

#### #12 - 09 octobre 2020 11:35 - Thomas Jund (congé, retour le 29/04)

Ok mais on n'a pas de libellés pour les lignes sur ce type de tableau

Oui, c'est pourquoi je proposais juste un compteur pour améliorer la compréhension en mobile

```
Participants :
|   | Prénom |   Nom   |
|---|-----|-----|
| 1 | Frédéric | Péters  |
| 2 | Thomas  | Jund    |
|<+>-----|
```

En, mobile

```
Participants :
| 1 |
| Prénom | Frédéric |
| Nom    | Péters   |
| 2 |
| Prénom | Thomas   |
| Nom    | Jund     |
|<+>-----|
```

à la place de

```
Participants :
| Prénom |
| Nom    |
| Prénom |
| Nom    |
|<+>-----|
```

Et avoir ainsi un feedback visuel permettant au user quand il clique sur "+" de voir "3"

#### #13 - 09 octobre 2020 11:48 - Frédéric Péters

Cet aspect ça me va bien si ce n'est pas dans le balisage. Aussi même en-dehors de cet aspect je ne trouve pas ça terrible, ça ne serait pas tranquillement compréhensible d'avoir une variation dans la couleur de fond pour identifier les "lignes" différentes ?

#### #14 - 09 octobre 2020 12:16 - Thomas Jund (congé, retour le 29/04)

ça ne serait pas tranquillement compréhensible d'avoir une variation dans la couleur de fond pour identifier les "lignes" différentes ?

Comme un zebra sur les tbody > tr. Techniquement possible (aussi trouver une solution qui garantie de s'adapter à tous les thèmes existants) et voir si on le limite qu'à l'affichage mobile.

Cet aspect ça me va bien si ce n'est pas dans le balisage

Je pensais à un th vide qu'on viendrait remplir avec un compteur css. Pour des raisons d'accessibilité cela me semble plus efficace qu'une indication colorée. (Mais à y regarder de plus prêt, pour une bonne accessibilité il faudrait surtout des attributs scope sur les th : `thead > tr > th[scope="col"]`).

Mon avis : zebra c'est mieux que rien. Compteur plus efficace en terme de compréhension.

#### #15 - 12 octobre 2020 17:24 - Thomas Jund (congé, retour le 29/04)

- Fichier 0001-scss-make-table-widgets-responsive-41734.patch ajouté

- Fichier 0001-js-make-table-widgets-responsive-41734.patch ajouté

- Fichier zebra.png ajouté

Suite à [#47510](#)

- Suppression des class CSS spécifiques pour identifier les headers des colonnes et des lignes.
- Suppression du paramètre de la fonction qui permet de personnaliser le sélecteur pour les headers de colonnes et de lignes.
- Ajout d'un zebra en mobile sur les tableaux pour aider à l'identification des lignes (voir capture zebra.png)

## #16 - 16 octobre 2020 10:42 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

## #17 - 16 octobre 2020 11:13 - Thomas Jund (congé, retour le 29/04)

- Statut changé de Solution validée à Résolu (à déployer)

### Patch JS wcs

```
commit 8407016c974961f067d980c9f904d34089c962c2 (HEAD -> master, origin/master, origin/HEAD)
```

```
Author: Thomas JUND <tjund@entrouvert.com>
```

```
Date: Thu Oct 8 14:13:01 2020 +0200
```

```
js: make table widgets reponsive (#41734)
```

### Patch scss publik-base-theme

```
commit e536b5f214f87720585c3737502911da8e73dd7d (HEAD -> master, origin/master, origin/HEAD)
```

```
Author: Thomas JUND <tjund@entrouvert.com>
```

```
Date: Thu Oct 8 14:15:57 2020 +0200
```

```
scss: make table widgets responsive (#41734)
```

## #18 - 16 octobre 2020 21:16 - Frédéric Péters

- Statut changé de Résolu (à déployer) à Solution déployée

### Fichiers

tableau.PNG	195 ko	15 avril 2020	Serghei Mihai
mobile.png	54,4 ko	08 octobre 2020	Thomas Jund (congé, retour le 29/04)
very-small.png	46,4 ko	08 octobre 2020	Thomas Jund (congé, retour le 29/04)
desktop.png	69,8 ko	08 octobre 2020	Thomas Jund (congé, retour le 29/04)
0001-scss-make-table-widgets-responsive-41734.patch	1,46 ko	08 octobre 2020	Thomas Jund (congé, retour le 29/04)
0001-js-make-table-widgets-reponsive-41734.patch	3,28 ko	08 octobre 2020	Thomas Jund (congé, retour le 29/04)
0001-scss-make-table-widgets-responsive-41734.patch	1,49 ko	12 octobre 2020	Thomas Jund (congé, retour le 29/04)
0001-js-make-table-widgets-reponsive-41734.patch	2,54 ko	12 octobre 2020	Thomas Jund (congé, retour le 29/04)
zebra.png	44 ko	12 octobre 2020	Thomas Jund (congé, retour le 29/04)