

Gadjo - Development #60396

La classe clickable-rows ne fonctionne pas avec les tableaux chargés dynamiquement

07 janvier 2022 10:21 - Benjamin Dauvergne

Statut:	Fermé	Début:	07 janvier 2022
Priorité:	Normal	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Ça concerne la cellule 'Demandes à traiter' qui est chargé en ajax (ticket #59872), dans ce cas il faut poser le gestionnaire d'évènement à la racine du document.			

Révisions associées

Révision 7546932d - 07 janvier 2022 10:22 - Benjamin Dauvergne

general: handle .clickable-rows at document level (#60396)

Historique

#2 - 07 janvier 2022 10:22 - Benjamin Dauvergne

- Fichier 0001-general-handle-.clickable-rows-at-document-level-603.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#3 - 07 janvier 2022 10:23 - Benjamin Dauvergne

J'ai testé le code sur toulouse-test c'est visible sur <https://agents-montoulouse.cutm-publik-preprod.nfrance.com/portail-police-municipale/>

#4 - 07 janvier 2022 10:26 - Serghei Mihai

Lors du chargement d'une cellule c'est l'évènement combo:cell-loaded qui est déclenché. Autant se baser dessus, je pense.

#5 - 07 janvier 2022 10:28 - Benjamin Dauvergne

Serghei Mihai a écrit :

Lors du chargement d'une cellule c'est l'évènement combo:cell-loaded qui est déclenché. Autant se baser dessus, je pense.

Autant pour générer du contenu ça me paraît obligatoire de passer par là, mais pour un onclick/onauxclick qu'est-ce que je gagne à faire ça ?

#6 - 07 janvier 2022 10:39 - Serghei Mihai

Ce que me paraît fragile est qu'au moment du chargement du document les .clickable-rows ne sont pas encore chargées et ne font pas partie de la DOM.

C'est peut-être un coup de chance que sur la recette de Toulouse ça fonctionne car la cellule se charge rapidement.

Mais je peux me tromper. Je laisse les autres connaisseurs de JS donner leur avis.

#7 - 07 janvier 2022 11:10 - Benjamin Dauvergne

Serghei Mihai a écrit :

Ce que me paraît fragile est qu'au moment du chargement du document les .clickable-rows ne sont pas encore chargées et ne font pas partie de la DOM.

Le fait de poser le handler sur document pallie justement à ce problème, les évènements traverse le DOM dans les deux sens de la racine (document) vers le noeud intérieur puis du noeud vers la racine, quand tu utilises la syntaxe \$(document).on('event', '<filtre>', ...) même si l'élément visé par filtre n'existe pas, quand il apparaîtra plus tard ça marchera.

C'est peut-être un coup de chance que sur la recette de Toulouse ça fonctionne car la cellule se charge rapidement.

Non justement c'est la correction ici qui fait que ça marche.

On passe par `combo:cell-loaded` quand il y a du contenu à générer/modifier suite au chargement d'une cellule par le code des cellules asynchrones, si au passage on ajoute des gestionnaires d'évènement c'est surtout par souci de garder le code un peu rassemblé mais ça n'est pas strictement nécessaire (mais on pourrait aussi utiliser un `MutationObserver`¹ et tout gérer sans trop d'"évènement" custom).

¹<https://developer.mozilla.org/en-US/docs/Web/API/MutationObserver>

#8 - 07 janvier 2022 11:15 - Serghei Mihai

- Statut changé de *Solution proposée* à *Solution validée*

Benjamin Dauvergne a écrit :

Le fait de poser le handler sur document pallie justement à ce problème, les évènements traverse le DOM dans les deux sens de la racine (document) vers le noeud intérieur puis du noeud vers la racine, quand tu utilises la syntaxe `$(document).on('event', '<filtre>', ...)` même si l'élément visé par filtre n'existe pas, quand il apparaîtra plus tard ça marchera.

Fire then.

#9 - 07 janvier 2022 11:23 - Thomas Jund (congé, retour le 29/04)

Le delegation d'évènement peut parfois bien plomber les performances. Et dans ce cas certainement impacter négativement (parfois c'est mieux, ça dépend).

L'évènement sera lancer chaque fois qu'on clic sur document, et par propagation (bubbling) de l'évènement, il va ensuite vérifier si le click est également effectué sur `.clickable-rows tr`.

Je suis ok avec Serghei, pourquoi ne pas simplement attaché l'évènement au tableau une fois qu'il est chargé ? et s'il y a beaucoup de tr au tableau, là il est préférable d'attacher d'évènement au tableau plutôt qu'à chaque row

#10 - 07 janvier 2022 11:29 - Benjamin Dauvergne

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

```
commit 7546932ddf4fec73d18c691831781484c5beedd5
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Fri Jan 7 10:22:17 2022 +0100
```

```
general: handle .clickable-rows at document level (#60396)
```

#11 - 07 janvier 2022 11:35 - Thomas Jund (congé, retour le 29/04)

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

(mais c'est juste pour info, si pas de prob de perf, faut pas s'embêter)

#12 - 07 janvier 2022 11:50 - Benjamin Dauvergne

J'en prends note, si quelqu'un constate des lenteurs quand on a un tableau 'clickable-rows' qu'il ouvre un ticket, je ferai le nécessaire.

#13 - 07 janvier 2022 12:01 - Benjamin Dauvergne

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

#14 - 07 janvier 2022 12:16 - Frédéric Péters

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

#15 - 13 mars 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-general-handle-.clickable-rows-at-document-level-603.patch 846 octets 07 janvier 2022

Benjamin Dauvergne