

Combo - Development #8454

Possibilité d'afficher une carte OSM

01 octobre 2015 09:37 - Victor Claudet

Statut:	Fermé	Début:	01 octobre 2015
Priorité:	Normal	Echéance:	
Assigné à:	Serghei Mihai	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			
Dans le cas du pilotage d'un momo en particulier, on a besoin d'afficher des cartographie, ne serait-ce qu'un plan de la ville. Dans Momo on inclue uniquement l'url vers la carte (ex: http://www.openstreetmap.org/export/embed.html?bbox=2.794818878173828%2C48.92373999482194%2C2.9863929748535156%2C48.991256905198185&layer=mapnik) et il se charge du reste.			
est-ce qu'on prévoit un objet spécifique dans combo, capable d'afficher la carte à partir de l'url dans un contexte navigateur web ? est-ce qu'on prévoit un objet capable d'afficher un iframe auquel on communique l'url ?			
Demandes liées:			
Lié à Publik - Development #15651: Éléments cartographiques souhaités pour vi...		Fermé	28 mars 2017
Lié à Combo - Development #16419: declencher un evenement quand le formulaire...		Fermé	20 mai 2017
Dupliqué par Combo - Development #15208: cellule permettant l'affichage des p...		Rejeté	02 mars 2017

Révisions associées

Révision 11da88d4 - 03 juin 2017 11:32 - Serghei Mihai

maps: add UI to manage map layers (#8454)

Révision 85ddc786 - 03 juin 2017 11:32 - Serghei Mihai

maps: add map cell (#8454)

Historique

#1 - 10 octobre 2016 11:33 - Frédéric Péters

Carte, paramétrée niveau de zoom / position initiale / fichier/url geojson avec les points à afficher.

#2 - 11 octobre 2016 10:34 - Thomas Noël

A voir si on prévoit de proposer plusieurs geojson (plusieurs layers), et si pour un layer donné on peut expliquer comment il s'affiche (le style, les popup). Cf <http://leafletjs.com/examples/geojson/>

(mais dans un premier temps, un affichage simple d'un geojson, c'est bien)

#3 - 29 novembre 2016 18:38 - Pierre Cros

Discussion EO camp :

- Quand on sauvegarde des filtres de recherche, on sélectionne les colonnes (les champs) qu'on souhaite exporter et c'est ça qui apparaîtra (en popup) quand on clique sur un point. (et donc dans avoir une popup sur chaque demande figurant sur la carte avec les infos minimales)
- Pour éviter d'afficher publiquement sur la carte des commentaires pourris, on pourrait choisir en configurant la cellule de n'afficher que les demandes dans tel ou tel statut (pour avoir des demandes déjà vue et validée par un agent).
- La cellule combo pourrait afficher en front les layers utilisés par la cellule et permettre à l'usager de les sélectionner/désélectionner.

#4 - 02 mars 2017 14:20 - Frédéric Péters

- Dupliqué par Development #15208: cellule permettant l'affichage des plusieurs demandes sur une carte ajouté

#6 - 24 mars 2017 13:05 - Frédéric Péters

- Sujet changé de Possibilité d'afficher une carte OSM (iframe ?) à Possibilité d'afficher une carte OSM

Mes idées comme ça :

- avoir une section de combo où gérer la cartographie (comme on a la gestion des paiements ou presque des ressources)
 - on peut y voir/gérer des layers, ceux-ci sont définis avec libellé, url du geojson, couleur/type de marqueur, options particulières (clustering / heatmap)
- avoir une cellule de type carte
 - paramétrable avec position initiale, niveau de zoom initial, niveaux de zoom possibles, (+ zone géographique possible ?)
 - une série de layers
 - une case à cocher pour déterminer si les layers peuvent être affichés/cachés par l'utilisateur

#7 - 28 mars 2017 13:49 - Frédéric Péters

- Lié à *Development #15651: Éléments cartographiques souhaités pour visualisation de demandes (type signalements) en FO* ajouté

#8 - 18 mai 2017 08:09 - Serghei Mihai

- Fichier *0001-maps-add-layers-8454.patch* ajouté

- Fichier *0002-maps-map-cell-8454.patch* ajouté

- Fichier *cell_manager.png* ajouté

- Fichier *layers.png* ajouté

- Fichier *cell_rendered.png* ajouté

- Statut changé de *Nouveau* à *En cours*

- Assigné à mis à *Serghei Mihai*

- Patch *proposed* changé de *Non* à *Oui*

Premier jet avec la gestion des couches dans le menu "kebab".

Pour la position initiale de la cellule je me suis inspiré de ce qu'a été fait dans wcs.

Il manque de tests et quelques glitches css à corriger.

#9 - 18 mai 2017 18:28 - Serghei Mihai

- Fichier *0001-maps-add-layers-8454.patch* ajouté

- Fichier *0002-maps-map-cell-8454.patch* ajouté

- Fichier *cellule_carte.png* ajouté

Glitches css et JS corrigés.

Il manque les tests, mais je suis preneur d'avis déjà sur le rendu et le code.

#10 - 19 mai 2017 11:25 - Thomas Noël

En vrac:

- *xstatic.pkg.font_awesome* dans le settings, il faut sans doute l'ajouter dans *setup.py* et *debian/control*
- *jquery.colourpicker.js* à passer en *xstatic* ; et après, presque envie de faire un patch séparé pour l'intégration du colorpicker
- Copyright (C) 2015
- *combo/apps/maps/views.py* est un peu vide

Dans les *MayLayer*, je vais prendre un peu d'avance, mais quid d'intégrer des geojson qui décrivent des polygones, des lignes, etc ?

Au niveau du chargement des geojson : il faut un contrôle d'accès identique à celui de la cellule. Et je pense que ça serait mieux d'avoir un chargement de chaque geojson indépendamment, plutôt qu'une agrégation pythonnée (enfin bon, on pourrait voir ça plus tard).

#11 - 19 mai 2017 13:50 - Serghei Mihai

- Fichier *0001-maps-add-layers-8454.patch* ajouté

- Fichier *0002-maps-map-cell-8454.patch* ajouté

Thomas Noël a écrit :

- *combo/apps/maps/views.py* est un peu vide

J'y pose une première vue dans le deuxième patch.

Dans les *MayLayer*, je vais prendre un peu d'avance, mais quid d'intégrer des geojson qui décrivent des polygones, des lignes, etc ?

Pour l'instant la seule source de json qu'on imagine utiliser (les vues geojson de bijoe) n'expose que des points, mais il faudrait en deuxième temps prendre en compte d'autres géométries, en effet. Ou encore la heatmap.

Au niveau du chargement des geojson : il faut un contrôle d'accès identique à celui de la cellule.

Tu veux dire l'accès aux usagers connectés/groupes, etc?

#12 - 19 mai 2017 14:19 - Thomas Noël

Serghei Mihai a écrit :

Thomas Noël a écrit :

- `combo/apps/maps/views.py` est un peu vide

J'y pose une première vue dans le deuxième patch.

Je voulais dire : fichier inutile dans ce premier patch (inutilisé).

Dans les `MayLayer`, je vais prendre un peu d'avance, mais quid d'intégrer des geojson qui décrivent des polygones, des lignes, etc ?

Pour l'instant la seule source de json qu'on imagine utiliser (les vues geojson de bijoe) n'expose que des points, mais il faudrait en deuxième temps prendre en compte d'autres géométries, en effet. Ou encore la heatmap.

On verra. Pour la heatmap c'est pas le geojson qui décide, c'est un plugin leaflet qui change l'affichage. Il y a donc deux choses à penser (pour plus tard, mais l'avoir en tête) : d'un côté les sources geojson qui ne sont pas forcément des points (ça peut aussi être des polygones, lignes, etc). Et de l'autre la façon d'afficher, avec les plugins leaflet (cluster, heatmap).

Bref, pour plus tard.

Au niveau du chargement des geojson : il faut un contrôle d'accès identique à celui de la cellule.

Tu veux dire l'accès aux usagers connectés/groupes, etc?

Yep. Reprendre les mêmes droits que pour l'affichage de la cellule (voir "`ajax_page_cell`"). Ca va peut-être nécessiter d'ajouter une référence à la page, comme pour `ajax_page_cell`, à voir.

#13 - 19 mai 2017 17:32 - Serghei Mihai

- Fichier `0001-maps-add-layers-8454.patch` ajouté

- Fichier `0002-maps-map-cell-8454.patch` ajouté

Et avec un contrôle à la vue fournissant le geojson en fonction de la visibilité de la page et de la cellule. Avec des tests.

#14 - 19 mai 2017 17:44 - Serghei Mihai

- Fichier `0002-maps-map-cell-8454.patch` supprimé

#15 - 19 mai 2017 17:45 - Serghei Mihai

- Fichier `0002-maps-map-cell-8454.patch` ajouté

#16 - 19 mai 2017 18:11 - Serghei Mihai

- Fichier `0002-maps-map-cell-8454.patch` ajouté

Thomas m'a rappelé sur Jabber qu'il faut pas avoir plusieurs éléments avec le même id sur une page, je me base donc sur le nom de classe pour lancer leaflet.

#17 - 19 mai 2017 18:27 - Serghei Mihai

- Fichier `0002-maps-map-cell-8454.patch` ajouté

Pareil pour l'id du input caché avec la position initiale.

#18 - 20 mai 2017 13:44 - Thomas Noël

Relecture rapide et partielle de samedi midi de 0002-maps-map-cell-8454.patch

combo/apps/maps/models.py

```
default='48.83369263315934;2.3233688436448574' : utiliser un settings COMBO_MAP_DEFAULT_POSITION =  
'48.83369263315934;2.3233688436448574'
```

Je sais pas si limiter les niveaux de zoom est pertinent... mais laisse ainsi, faisons comme dans wcs, personne n'a crié :)

map.js, map.css : plutôt combo.map.js et combo.map.css (pour rappeler qu'il s'agit de trucs locaux, on aurait dû le faire sur tipi ou usersearch)

(à un moment il faudra prévoir de laisser le choix d'un fond de carte, aussi, genre parmi une liste de settings ... plus tard)

combo/apps/maps/views.py

Y'a un import de "requests" inutile.

```
if not cell.page.public:  
    raise PermissionDenied()
```

→ Mmh... il faut aussi que ça marche pour des pages privées, hein. (sans doute cell.page.is_visible())

Globalement ça serait plus simple de mettre «l'intelligence» de la vue GeojsonView dans le modèle de la cellule map (la vue ne ferait, en gros, qu'un appel à MapCell::get_geojson)

combo/apps/maps/urls.py :

```
url(r'^maps/geojson/(?P<cell_id>\w+)/$', GeojsonView.as_view(),  
    name='cell-geojson'),
```

je préférerais /ajax/mapcell/geojson/(?P<cell_id>\w+)/ et nommé mapcell-geojson

combo/apps/maps/static/js/map.js

non relu

mapcell.html

pas besoin du load i18n

rien de méchant mais "initial" dans data-initial-zoom= alors que data-init-lat= et data-init-lng=

map_widget.html

ajouter le zoom aussi ? (en plus de lat/lng)

combo/manager/static/js/combo.manager.js

en faire un petit ticket/patch séparé en amont qui explique la raison, sinon on va oublier

combo/manager/templates/combo/manager_base.html

En vrai, on n'a pas besoin des js et css des maps même quand y'a pas de map sur la page... c'est pas possible que ça soit dynamique ? (je demande vraiment sans savoir)

Ah, dans 0001 :

```
urlpatterns = [  
    url(r'^manage/maps/', decorated_includes(manager_required,  
        include(maps_manager_urls))),  
    ..
```

ça devrait plutôt être, je pense, dans maps_manager_urls ? non ?

Et git+<http://git.entrouvert.org/debian/xstatic-jquery-colourpicker.git> dans les requirements, non : on doit pousser le paquet dans pypi → [HowDoWeDoPythonPackaging](#)

#19 - 20 mai 2017 14:42 - Serghei Mihai

- Lié à Development #16419: déclencher un événement quand le formulaire de paramétrage d'une cellule est configuré ajouté

#20 - 20 mai 2017 15:33 - Serghei Mihai

- Fichier 0001-maps-add-layers-8454.patch ajouté

- Fichier 0002-maps-map-cell-8454.patch ajouté

Thomas Noël a écrit :

En vrai, on n'a pas besoin des js et css des maps même quand y'a pas de map sur la page... c'est pas possible que ça soit dynamique ? (je demande vraiment sans savoir)

Pour l'instant nous n'avons rien pour un chargement conditionnels des statics en backoffice.

ça devrait plutôt être, je pense, dans maps_manager_urls ? non ?

J'ai suivi l'exemple de lingo et je préfère ne pas multiplier les fichiers.

Et git+<http://git.entrouvert.org/debian/xstatic-jquery-colourpicker.git> dans les requirements, non : on doit pousser le paquet dans pypi → HowDoWeDoPythonPackaging

Yep, je vais pousser.

#21 - 21 mai 2017 01:25 - Thomas Noël

Serghei Mihai a écrit :

Thomas Noël a écrit :

En vrai, on n'a pas besoin des js et css des maps même quand y'a pas de map sur la page... c'est pas possible que ça soit dynamique ? (je demande vraiment sans savoir)

Pour l'instant nous n'avons rien pour un chargement conditionnels des statics en backoffice.

Ok. Attention, css et js renommées combo.map.xx : à modifier dans manager_base.html

(je reste un peu perplexe parce que ckeditor a du js et des css chargées dynamiquement, mais ça doit être un autre bouzin)

ça devrait plutôt être, je pense, dans maps_manager_urls ? non ?

J'ai suivi l'exemple de lingo et je préfère ne pas multiplier les fichiers.

Oublie cette remarque, j'ai dit d'importe quoi.

Et git+<http://git.entrouvert.org/debian/xstatic-jquery-colourpicker.git> dans les requirements, non : on doit pousser le paquet dans pypi → HowDoWeDoPythonPackaging

Yep, je vais pousser.

<https://pypi.python.org/pypi/XStatic-jquery-colourpicker/> yeah, cool

Sinon, la migration a été refaite ? Elle comporte un "default=b'48.83369263315934;2.3233688436448574'" qui m'intrigue (et qui risque de râler une fois le paquet installer si le settings COMBO_MAP_DEFAULT_POSITION est différent ?)

#22 - 21 mai 2017 11:04 - Serghei Mihai

- Fichier 0002-maps-map-cell-8454.patch ajouté

Thomas Noël a écrit :

Ok. Attention, css et js renommées combo.map.xx : à modifier dans manager_base.html

Bien vu, merci.

(je reste un peu perplexe parce que ckeditor a du js et des css chargées dynamiquement, mais ça doit être un autre bouzin)

dans manager_base.html:

```
<script src="{% static "ckeditor/ckeditor/ckeditor.js" %}"></script>
<script type="text/javascript" src="{% static "ckeditor/ckeditor-init.js" %}"></script>
```

ou je rate peut-être un truc.

Sinon, la migration a été refaite ? Elle comporte un "default=b'48.83369263315934;2.3233688436448574'" qui m'intrigue (et qui risque de râler une fois le paquet installer si le settings COMBO_MAP_DEFAULT_POSITION est différent ?)

Oui, mais ça ne change rien au niveau de la migration. En revanche le settings est bien pris en compte dans le formulaire de paramétrage de la cellule, donc pour chaque tenant il peut être différent.

#23 - 22 mai 2017 08:09 - Serghei Mihai

- Fichier 0001-maps-add-layers-8454.patch ajouté

- Fichier 0002-maps-map-cell-8454.patch ajouté

Les deux patches à jour, notamment suite à la modification proposée dans [#16419](#).

#24 - 25 mai 2017 14:48 - Thomas Noël

Vite dit au bureau, dans 0002:

combo/apps/maps/migrations/0002_mapcell.py

- utiliser settings.COMBO_MAP_DEFAULT_POSITION au lieu de b'48.83369263315934;2.3233688436448574'

combo/apps/maps/static/js/combo.map.js :

- 'https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png' → à chercher dans un data-tile-urltemplate qui viendra de settings.COMBO_MAP_TILE_URLTEMPLATE
- idem pour l'attribution (sauf que je sais pas trop comment passer du HTML dans un data-truc)

(rien relu pour le reste, temps trop ensoleillé)

#25 - 26 mai 2017 11:22 - Serghei Mihai

- Fichier 0002-maps-map-cell-8454.patch ajouté

Avec les remarques prises en compte.

Pour l'attribution Django sort le HTML "safe" (remplace > par >, etc) et leaflet semble être assez intelligent pour le comprendre et faire le rendu proprement.

#26 - 26 mai 2017 14:00 - Frédéric Péters

Zapper la partie colourpicker, utiliser <input type="color"> est désormais ok dans Firefox et Chrome.

#27 - 26 mai 2017 14:08 - Frédéric Péters

Le COMBO_MAP_DEFAULT_POSITION, plutôt en faire un dictionnaire {'lat': ..., 'lon': ...}, ça sera plus clair.

Ne pas inclure "Cell" dans le nom de la cellule.

#28 - 26 mai 2017 14:12 - Frédéric Péters

Pas besoin de cet écran /manage/maps/ avec une seule option.

#29 - 26 mai 2017 14:13 - Serghei Mihai

Frédéric Péters a écrit :

Zapper la partie colourpicker, utiliser <input type="color"> est désormais ok dans Firefox et Chrome.

Et tant pis pour IE11 et compagnie?

#30 - 26 mai 2017 14:26 - Frédéric Péters

Pour le /manage/, clairement rien à taper d'IE.

#31 - 26 mai 2017 14:28 - Frédéric Péters

Pour que les appels soient faits avec les infos de l'utilisateur connecté (nécessaire pour taper dans le geojson de wcs) :

```

+++ b/combo/apps/maps/models.py
@@ -57,9 +57,10 @@ class MapLayer(models.Model):
     def __unicode__(self):
         return self.label

-     def get_geojson(self):
+     def get_geojson(self, request):
         response = requests.get(self.geojson_url,
                                 remote_service='auto',
+                                 user=request.user,
                                 headers={'accept': 'application/json'})

         if not response.ok:
             return []
@@ -117,7 +118,7 @@ class MapCell(CellBase):
     if not self.is_visible(request.user):
         raise PermissionDenied()
     for layer in self.layers.all():
-         geojson['features'] += layer.get_geojson()
+         geojson['features'] += layer.get_geojson(request)
     return geojson

```

#32 - 26 mai 2017 14:28 - Frédéric Péters

```

for feature in features:
    feature['display_fields'] = feature['properties'].copy()

```

L'indentation monstrueuse.

#33 - 26 mai 2017 14:31 - Frédéric Péters

Crash solide sur `feature['display_fields'] = feature['properties'].copy()` (string indices must be integers) quand confronté au geojson produit par w.c.s. :

```

{
  "type": "FeatureCollection",
  "features": [
    {
      "geometry": {
        "type": "Point",
        "coordinates": [
          2.321416,
          48.840442
        ]
      },
      "type": "Feature",
      ...
    }
  ]
}

```

#34 - 29 mai 2017 11:42 - Serghei Mihai

- Fichier 0001-maps-add-layers-8454.patch ajouté

- Fichier 0002-maps-map-cell-8454.patch ajouté

#35 - 29 mai 2017 11:59 - Frédéric Péters

Il y a par exemple dans 0002 :

```

-         feature['properties']['icon'] = self.icon
+         feature['properties']['icon'] = self.icon

```

Ça devrait plutôt être correct dès le 0001.

```

icon = models.CharField(_('Marker icon'), max_length=32, blank=True, null=True,
                        help_text=_('FontAwesome style name'))

```

Même si ça limiterait, ça serait plus pratique pour le début qu'on commence par taper un `choices=[]` avec quelques icônes qu'on saurait utiles. (je pense)

`maplayer_confirm_delete.html`, `maplayer_form.html`, etc. je sais que ce sont les noms automatiques mais je les trouve difficiles à lire, `map_layer_etc.` serait pour moi bien plus net.

```

url('^layers/(?P<pk>\w+)/edit$', LayerEditView.as_view(),
    name='maps-manager-layer-edit'),
url('^layers/(?P<pk>\w+)/edit/$', LayerEditView.as_view(),
    name='maps-manager-layer-edit'),

```

Doublon.

```
/* colour picker styles */
```

Plus utile. (et dans combo.manager.js pareil).

#36 - 29 mai 2017 14:33 - Serghei Mihai

- Fichier 0001-maps-add-layers-8454.patch ajouté

- Fichier 0002-maps-map-cell-8454.patch ajouté

Frédéric Péters a écrit :

Même si ça limiterait, ça serait plus pratique pour le début qu'on commence par taper un choices=[] avec quelques icônes qu'on saurait utiles. (je pense)

Mon idée ici est de partir pour l'instant d'un champ libre car nous n'avons aucune suggestion d'icône.

Une fois que la liste commencera à se définir (à travers les retours client) je verrais bien une liste à choix avec les noms des icônes qu'on pourrait plutôt alimenter avec celles de noun project.

#37 - 29 mai 2017 14:46 - Frédéric Péters

Mon idée ici est de partir pour l'instant d'un champ libre car nous n'avons aucune suggestion d'icône.

Si ce n'est que ça, pour commencer, je prendrais la série des "Transportation Icons" (moins fighter-jet, rocket, ship, space-shuttle et wheelchair-alt) + home + university + building + hospital + glass + beer.

Une fois que la liste commencera à se définir (à travers les retours client) je verrais bien une liste à choix avec les noms des icônes qu'on pourrait plutôt alimenter avec celles de noun project.

On partirait d'un truc qui ferme sur les icônes FontAwesome et demande à des clients de trouver et taper fa-bicycle. Et puis on basculerait sur des icônes techniquement toutes différentes ? Alors bien sûr intervient toujours mon aversion à balancer 30ko de CSS pour au final choisir deux marqueurs mais quand même, ça m'irait bien de commencer par un jeu fini d'icônes, et sur des retours clients on pourra en ajouter facilement.

#38 - 29 mai 2017 14:58 - Frédéric Péters

```
class Meta:
    verbose_name = _('Map Cell')
```

Ne pas inclure "Cell" dans le nom de la cellule.

#39 - 29 mai 2017 15:05 - Frédéric Péters

```
-         if (feature.display_fields) {
+         if (feature.properties.display_fields) {
            var popup = '';
-         $.each(feature.display_fields, function(key, value) {
+         $.each(feature.properties.display_fields, function(key, value) {
```

? J'ai d'abord imaginé une différence entre le geojson produit par bijoe et celui de w.c.s. mais non, de ce que je lis du code de bijoe les propriétés se trouvent aussi rangées sans .properties.

#40 - 29 mai 2017 15:09 - Frédéric Péters

```
"display_fields": [
  ["Num\u00e9ro", "144-3"],
  ["Date de cr\u00e9ation", "13/04/2017 17:56"],
  ...]
```

Le geojson produit par w.c.s. inclut libellé/valeur ainsi; le code actuel de ce patch donne alors comme résultat une popup avec :

- 0
 - Numéro,144-3
- 1
 - Date de création,13/04/2017 17:56

Plutôt que

- Numéro

- 144-3
- Date de création
 - 13/04/2017 17:56

(bien sûr le geojson côté w.c.s. peut évoluer mais le `display_fields` me semble déjà un truc qui lui est spécifique.

#41 - 29 mai 2017 16:35 - Serghei Mihai

- Fichier `0001-maps-add-layers-8454.patch` ajouté

- Fichier `0002-maps-map-cell-8454.patch` ajouté

Ok, sans les icones glass et beer pour l'instant. On les ajoutera plus tard.

Initialement j'avais pas pensé gérer les geojson en provenance de wcs, notamment avec le champ `display_fields`, mais autant de le prendre en compte de maintenant.

#42 - 29 mai 2017 16:40 - Frédéric Péters

Initialement j'avais pas pensé gérer les geojson en provenance de wcs, notamment avec le champ `display_fields`, mais autant de le prendre en compte de maintenant.

Mais pourquoi donc il y avait du code qui parlait de `display_fields` ?

#43 - 29 mai 2017 17:42 - Serghei Mihai

J'inventais ce champ, par analogie avec wcs, pour reprendre les champs dans propriétés d'origine (du geojson de bijoe) pour les afficher dans la popup avant d'y ajouter des attributs couleur du marker, icones, etc.

#44 - 29 mai 2017 17:48 - Frédéric Péters

```
'verbose_name': 'Map Cell',
```

Ne pas inclure "Cell" dans le nom de la cellule.

Il faut mettre à jour également la migration.

J'inventais ce champ, par analogie avec wcs, pour reprendre les champs dans propriétés d'origine (du geojson de bijoe) pour les afficher dans la popup avant d'y ajouter des attributs couleur du marker, icones, etc.

Si ça te dit de modifier w.c.s. pour correspondre à ce que tu souhaiterais ici, ça me va aussi très bien.

#45 - 29 mai 2017 18:02 - Serghei Mihai

- Fichier `0002-maps-map-cell-8454.patch` ajouté

C'est plutôt dans bijoe que je verrais le champ `display_fields` qui, comme dans w.c.s., contiendraient les variables de regroupement horizontal/vertical.

#46 - 29 mai 2017 21:42 - Frédéric Péters

C'est plutôt dans bijoe que je verrais le champ `display_fields` qui, comme dans w.c.s., contiendraient les variables de regroupement horizontal/vertical.

De manière générale, pour aider les relecutres successives, plutôt que juste poser un nouveau patch, ça aiderait d'explicitier ce qui change. Ici pour info la seule différence c'est :

```
- 'verbose_name': 'Map Cell',  
+ 'verbose_name': 'Map',
```

~~

```
marker_colour = models.CharField(_('Marker color'), max_length=7, default='#0000FF')  
icon_colour = models.CharField(_('Icon colour'), max_length=7, default='#FFFFFF')
```

Un coup color un coup colour.

```
<a href="{% url 'maps-manager-homepage' }">{ trans 'Maps Layers' %}</a>
```

Juste "Maps" qu'on traduirait ensuite par Cartes ?

--

```
css = {'all': ('xstatic/leaflet.css', 'xstatic/css/font-awesome.min.css',
```

Burn; déjà noté, ça m'horripile de consommer 30ko pour quelques icônes.

#47 - 30 mai 2017 08:03 - Frédéric Péters

```
geojson_url = models.URLField(_('Geojson URL'))
```

Trop court par défaut, par exemple

https://download.data.grandlyon.com/wfs/grandlyon?SERVICE=WFS&VERSION=2.0.0&outputformat=GEOJSON&maxfeatures=30&request=GetFeature&typename=ter_territoire.maison_du_rhone&SRNAME=urn:ogc:def:crs:EPSG::4171 n'entre pas dedans.

--

```
combo/apps/maps/templates/maps/mapcell.html
@@ -1,3 +1,3 @@
-<h2>{{ title }}</h2>
+{% if title %}<h2>{{ title }}</h2>{% endif %}
```

Pour ne pas avoir de barre vide quand on ne met pas de titre.

--

```
onEachFeature: function(feature, layer) {
    ...
```

J'ai l'impression que souvent en fait on ne voudra pas ce comportement; en première réaction, je dirais qu'il faut remplacer ça par la levée d'un événement à nous, genre `$(this).trigger('combo:map-feature-click', feature, layer)`. Et libre dans ce patch de ne même pas avoir de comportement par défaut.

Dans un patch ultérieur, l'événement serait attrapé et un appel ajax renverrait le rendu de `combo/map/feature_%(layer slug).html` (ça pourrait être un POST avec le contenu de feature, pour beaucoup de liberté dans le template), et c'est ça qui serait utilisé comme popup.

--

Et donc aussi, au modèle layer, ajouter un champ slug.

#48 - 31 mai 2017 10:19 - Serghei Mihai

- Fichier `0001-maps-add-layers-8454.patch` ajouté

- Fichier `0002-maps-map-cell-8454.patch` ajouté

Remarques prises en compte: longueur de l'url field augmentée à 1024 (j'espère que ça suffira même pour les urls de wcs avec plein de parametres), font-awesome.min.css supprimé aussi, tests mis à jour.

Ok pour l'idée de déclencher l'événement `combo:map-feature-click` pour afficher ensuite la popup avec l'info dans un template personnalisé.

#49 - 31 mai 2017 10:31 - Frédéric Péters

Copyright (C) 2015 Entr'ouvert

Il y a des jours où ça saute aux yeux.

#50 - 31 mai 2017 10:43 - Serghei Mihai

- Fichier `0001-maps-add-layers-8454.patch` ajouté

Je vais changer de lunettes. Merci.

#51 - 31 mai 2017 10:48 - Frédéric Péters

```
icons = [
```

Taper les constantes en majuscules; et taper ça après les import (monter from `combo.utils import requests` vers les autres). Taper la première lettre des noms des icônes en majuscule.

Et dans le second patch,

```
zoom_levels = [
```

Pareil.

```
- ('hospital-o', _('hospital')),  
+ ('hospital', _('hospital')),
```

Taper la valeur correcte dès le premier commit.

```
widgets = {'layers': forms.widgets.CheckboxSelectMultiple,  
          'default_position': MapWidget(attrs={'init_lat': lat,  
                                             'init_lng': lng,  
                                             'init_zoom': self.initial_zoom,  
                                             ...
```

Il y a un truc avec cette indentation default_position/MapWidget/attrs/ qui m'a arrêté à me demander pourquoi, pour une position par défaut, au-delà de la latitude et de la longitude on passait un niveau de zoom et tout un tas d'autres attributs; à me dire qu'un dictionnaire était mal fermé quelque part. Je me dis que simplement passer attrs={ sur sa propre ligne clarifiera.

```
ctx['initial_lat'], ctx['initial_lng'] = self.default_position.split(';');
```

Point-virgule perdu.

```
/* leaflet markers icons */  
i.icon {
```

C'est quand même très générique comme nom de classe; il y aurait moyen de mettre plutôt leaflet-icon, ou map-icon ? Et les classes dessous, de combiner, façon i.leaflet-icon.car ?

Dans map_cell.html :

```
<div class="combo-cell-map" data-init-zoom="{{ initial_zoom }}" data-min-zoom="{{ min_zoom }}" data-max-zoom="{{ max_zoom }}" data-init-lat="{{ initial_lat }}" data-init-lng="{{ initial_lng }}" data-geojson-url="{{ geojson_url }}" data-tile-urltemplate="{{ tile_urltemplate }}" data-map-attribution="{{ map_attribution }}">
```

vs dans map_widget.html :

```
<div class="combo-cell-map" data-init-zoom="{{ init_zoom }}" data-min-zoom="{{ min_zoom }}" data-max-zoom="{{ max_zoom }}" data-init-lat="{{ init_lat }}" data-init-lng="{{ init_lng }}" data-store-position="combo-map-{{ id }}" data-tile-urltemplate="{{ tile_urltemplate }}" data-map-attribution="{{ map_attribution }}">
```

D'un côté init_lat/init_lng, de l'autre initial_lat/initial_lng, uniformisation possible ?

#52 - 31 mai 2017 11:52 - Serghei Mihai

- Fichier 0001-maps-add-layers-8454.patch ajouté

- Fichier 0002-maps-map-cell-8454.patch ajouté

Ok pour les choix des icônes et les niveaux de zoom.

Pour être très explicite sur la classe "icône" du marker je choisis .leaflet-marker-icon.

Pour le widget carte pas besoin de niveaux de zoom, en effet.

J'ai déplacé les attributs dans une variable à part pour plus de clarté.

#53 - 01 juin 2017 16:29 - Frédéric Péters

C'est quand même très générique comme nom de classe; il y aurait moyen de mettre plutôt leaflet-icon, ou map-icon ? Et les classes dessous, de combiner, façon i.leaflet-icon.car ?

Je voulais dire, que, à la place de :

```
i.home::before {
```

écrire

```
i.leaflet-marker-icon.home::before {
```

```
font: normal normal normal 14px/1 FontAwesome;
```

Les autres dimensions des marqueurs sont en rem, pas en px.

```
div.leaflet-popup-content span.field-label {  
  text-transform: capitalize;
```

```
}
```

Je zapperais, ça me semble un hack sur de la mauvaise donnée.

```
class GeojsonView(View):
    def get(self, request, *args, **kwargs):
        try:
            cell = Map.objects.get(pk=kwargs['cell_id'])
        except Map.DoesNotExist:
            raise Http404()
        geojson = cell.get_geojson(request)
        content_type = 'application/json'
        return HttpResponse(json.dumps(geojson), content_type=content_type)
```

Il faudrait contrôler l'accès, vérifier que l'utilisateur a accès à la cellule.

#54 - 02 juin 2017 10:28 - Serghei Mihai

- Fichier 0001-maps-add-layers-8454.patch ajouté

- Fichier 0002-maps-map-cell-8454.patch ajouté

Ok pour la CSS.

Pour le contrôle d'accès, je le faisais dans la méthode `get_geojson` de la cellule, mais il vaut mieux le faire en amont, en effet.

#55 - 02 juin 2017 10:55 - Frédéric Péters

```
if 'display_fields' not in feature['properties']:
    display_fields = []
    for label, value in feature['properties'].iteritems():
        if value is not None:
            display_fields.append((label, value))
    feature['properties']['display_fields'] = display_fields
```

Je virerais cette partie.

```
default_position = models.CharField(_('Default position'), null=True, blank=True,
                                   default='%(lat)s;%(lon)s' % settings.COMBO_MAP_DEFAULT_POSITION,
```

Il faut éviter de mettre des valeurs des settings dans le paramétrage des champs, parce que là, si jamais on change le paramétrage, on va se taper sans fin des messages "attention faut créer une migration". Je ne mettrais pas de valeur par défaut à ce niveau, je choperais plutôt `settings.COMBO_MAP_DEFAULT_POSITION` au moment où c'est utile, façon :

```
default_position = self.default_position or settings.COMBO_MAP_DEFAULT_POSITION.
```

Et pour finir, les messages de commit,

- 0001 : maps: add UI to manage map layers
- 0002 : maps: add map cell

#56 - 02 juin 2017 14:06 - Serghei Mihai

- Fichier 0001-maps-add-UI-to-manage-map-layers-8454.patch ajouté

- Fichier 0002-maps-add-map-cell-8454.patch ajouté

Ok. `default_position` sans valeur par défaut mais en prenant en compte que la variable de settings est un dico.

#57 - 02 juin 2017 14:26 - Frédéric Péters

```
def get_cell_extra_context(self, context):
    ctx = super(Map, self).get_cell_extra_context(context)
    ctx['title'] = self.title
    ctx['init_lat'], ctx['init_lng'] = self.default_position.split(';')
```

Il faudrait ici aussi gérer la chose; ma proposition serait d'avoir une fonction genre ça et l'utiliser aux deux endroits :

```
def get_default_position(self):
    if self.default_position:
        return {'lat': self.default_position.split(';')[0], 'lng': self.default_position.split(';')[1]}
    return settings.COMBO_MAP_DEFAULT_POSITION
```

(et au passage faire que `settings.COMBO_MAP_DEFAULT_POSITION` utilise "lng" plutôt que "lon", ou l'inverse, mais uniforme).

#58 - 02 juin 2017 15:19 - Serghei Mihai

- Fichier 0002-maps-add-map-cell-8454.patch ajouté

Ok.

Patch (le dernier) à jour.

#59 - 02 juin 2017 16:17 - Frédéric Péters

```
self.default_position = '%(lat)s;%(lng)s' % default_position
lat, lng = self.default_position.split(';')
```

Ces deux lignes me semblent inutiles.

```
i.leaflet-marker-icon.home::before {
    font: normal normal normal 1rem/1 FontAwesome;
    content: "\f015"; /* home */
}
```

Je pense que mon commentaire a mal été interprété; il y avait deux éléments :

1. i.leaflet-marker-icon.home::before à la place de i.home.
2. ne pas utiliser px

Même si dans le commentaire les deux lignes se suivaient, sélecteur puis police, il ne s'agissait pas de les réunir dans le code.

#60 - 02 juin 2017 17:04 - Serghei Mihai

- Fichier 0002-maps-add-map-cell-8454.patch ajouté

Ok.

Je finissais self.default_position pour que dans le formulaire le champ caché correspondant à la position soit pre-rempli, mais je peux tout à fait utiliser "lat" et "lng" déjà passés dans le contexte pour remplir le "value".

#61 - 02 juin 2017 17:32 - Frédéric Péters

En l'absence de points, il y a une erreur js.

```
- map.fitBounds(geo_json.getBounds());
+ if (geo_json.getBounds().isValid()) {
+     map.fitBounds(geo_json.getBounds());
+ }
```

Et quand il y a plusieurs cartes sur une même page, que ce soit en frontoffice ou en backoffice : Error: Map container is already initialized.

#62 - 02 juin 2017 19:07 - Serghei Mihai

- Fichier 0002-maps-add-map-cell-8454.patch ajouté

Ok, vérification que les bounds sont bons et l'instanciation de la carte pour chaque cellule à part

#63 - 02 juin 2017 19:34 - Frédéric Péters

Quand j'édite la position par défaut, le comportement est étrange :

- la carte est à la position A
- je modifie la position, vers B
- je clique sur enregistrer
- la carte revient à la position A
- je modifie la position, vers C
- je clique sur enregistrer
- la carte va alors sur la position B
- etc.

Ce qui se passe c'est qu'à l'enregistrement, le formulaire contient un widget carte avec les coordonnées A, form.save() enregistre les nouvelles coordonnées, mais c'est le même objet form qui est utilisé pour afficher la cellule mise à jour, le même objet form, c'est-à-dire celui avec le widget carte contenant les anciennes coordonnées.

#64 - 02 juin 2017 19:41 - Frédéric Péters

Alternative, comme tu veux : virer ce paramétrage, avoir uniquement un paramétrage global de la position par défaut. (et dans un développement à venir, il y aurait possibilité dans hobo de définir celle-ci, et ça serait propagé vers combo et wcs).

#65 - 03 juin 2017 10:19 - Serghei Mihai

- Fichier 0002-maps-add-map-cell-8454.patch ajouté

En effet, partons sur la cellule sans position initiale dédiée.

#66 - 03 juin 2017 10:50 - Frédéric Péters

- Fichier 0001-map-js-changes.patch ajouté

Il y a des bizarreries dans le combo.map.js qui venait en partie de l'utilisation dans le /manage/, je me suis penché dessus pour nettoyer ça et au final, version remaniée un peu au-delà :

- retire l'utilisation du cellform-reloaded
- itère directement sur div.cell.map, plutôt que \$('div.combo-cell-map').parents('div.cell')
- retire la boucle dans le render_map, vu qu'il y a une seule carte par cellule (ça permet de gagner un niveau d'indentation)
- renomme le trigger de combo:map-feature-click en combo:map-feature-prepare, dans la mesure où il ne se déclenche pas au clic mais au chargement
- modifie le trigger pour avoir la cellule comme origine et pour passer feature/layer dans un unique paramètre (sinon le layer se perd)

#67 - 03 juin 2017 11:16 - Serghei Mihai

- Fichier 0002-maps-add-map-cell-8454.patch ajouté

Merci, je viens de le rajouter à mon dernier commit.

#68 - 03 juin 2017 11:25 - Frédéric Péters

Ok pour moi; si jamais tu veux, il y a quelques problèmes d'espace dans les fichiers (genre deux lignes blanches plutôt qu'une avant le is_enabled dans models.py, et des espaces avec le paramètre title dans les tests), mais c'est tout à fait mineur et ça m'irait que ça soit poussé ainsi.

#69 - 03 juin 2017 11:34 - Serghei Mihai

- Statut changé de En cours à Résolu (à déployer)

Corrigé et poussé.

```
commit 85ddc786e7158db5f3127504ea4f406447b7d0d7
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Mon May 15 19:01:38 2017 +0200
```

```
maps: add map cell (#8454)
```

```
commit 11da88d42c71264b041ab1b773163d26276717c7
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Fri May 12 11:53:07 2017 +0200
```

```
maps: add UI to manage map layers (#8454)
```

#70 - 13 août 2017 09:16 - Serghei Mihai

- Statut changé de Résolu (à déployer) à Fermé

Fichiers

0001-maps-add-layers-8454.patch	27,5 ko	18 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	17,7 ko	18 mai 2017	Serghei Mihai
cell_manager.png	997 ko	18 mai 2017	Serghei Mihai
layers.png	110 ko	18 mai 2017	Serghei Mihai
cell_rendered.png	837 ko	18 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	18,6 ko	18 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	27,5 ko	18 mai 2017	Serghei Mihai
cellule_carte.png	1,26 Mo	18 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	24,5 ko	19 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	19 ko	19 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	23,5 ko	19 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	23,8 ko	19 mai 2017	Serghei Mihai

0002-maps-map-cell-8454.patch	23,8 ko	19 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	22,8 ko	19 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	23,4 ko	20 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	23 ko	20 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	23 ko	21 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	23,4 ko	22 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	23,4 ko	22 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	24,2 ko	26 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	22,4 ko	29 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	25,1 ko	29 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	24,5 ko	29 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	20,8 ko	29 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	27,1 ko	29 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	21,6 ko	29 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	27,1 ko	29 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	21,4 ko	31 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	24,4 ko	31 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	21,3 ko	31 mai 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	23,8 ko	31 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	21,4 ko	31 mai 2017	Serghei Mihai
0001-maps-add-layers-8454.patch	21,4 ko	02 juin 2017	Serghei Mihai
0002-maps-map-cell-8454.patch	24,6 ko	02 juin 2017	Serghei Mihai
0001-maps-add-UI-to-manage-map-layers-8454.patch	20,8 ko	02 juin 2017	Serghei Mihai
0002-maps-add-map-cell-8454.patch	24,6 ko	02 juin 2017	Serghei Mihai
0002-maps-add-map-cell-8454.patch	25 ko	02 juin 2017	Serghei Mihai
0002-maps-add-map-cell-8454.patch	24,4 ko	02 juin 2017	Serghei Mihai
0002-maps-add-map-cell-8454.patch	24,7 ko	02 juin 2017	Serghei Mihai
0002-maps-add-map-cell-8454.patch	22,3 ko	03 juin 2017	Serghei Mihai
0001-map-js-changes.patch	5,78 ko	03 juin 2017	Frédéric Péters
0002-maps-add-map-cell-8454.patch	21,9 ko	03 juin 2017	Serghei Mihai