

w.c.s. - Bug #20364

geoloc : longitude à -357.7257085745805°, une autre à 2.2742914254195°

30 novembre 2017 11:11 - Thomas Noël

Statut:	Fermé	Début:	30 novembre 2017
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			
Vu sur un champ carte : 48.73105390371401;-357.7257085745805 et sur un autre champs 48.637356781598875;2.504296749830246			
Il faudrait redresser cela systématiquement dans les outils de geolocalisation, pour être toujours entre 0 et 360 (ou -180 et 180, je ne sais pas l'usage)			

Révisions associées

Révision c54766c4 - 04 décembre 2017 16:26 - Benjamin Dauvergne

normalize geographic coordinates into -90..90/-180..180 (#20364)

Révision d270f6ad - 04 décembre 2017 17:24 - Thomas Noël

do not try to normalize empty coordinates (#20364)

Historique

#1 - 30 novembre 2017 11:14 - Thomas Noël

- Fichier Capture d'écran de 2017-11-30 11-13-32.png ajouté

Je précise que la conséquence de ces valeurs est que leaflet affiche une carte du monde tellement large qu'on y voit deux fois la France : une à -360 et une à 0... et les points sur l'une ou l'autre France... Cf copie d'écran jointe.

#2 - 30 novembre 2017 18:33 - Benjamin Dauvergne

- Fichier 0001-normalize-geographic-coordinates-into-the-range-0.36.patch ajouté

- Patch proposed changé de Non à Oui

Je n'ai pas touché au JS je ne suis pas sûr que ce soit nécessaire pourvu qu'on normalise en sortie et au moment du stockage.

Si on m'indique comment tout ça se teste je veux bien en écrire.

#3 - 30 novembre 2017 18:50 - Frédéric Péters

- Statut changé de Nouveau à En cours

À mon sens faudrait d'abord regarder qui a retourné ces valeurs; si c'est via un connecteur dans passerelle (et le seul à faire du géocodage c'est base adresse), je serais pour y effectuer la normalisation. Si par contre c'est direct le nominatim d'osm, ok pour voir ça côté wcs.

#4 - 30 novembre 2017 22:53 - Thomas Noël

Comme je disais, c'est "sur un champ carte" (form_var_carte) que j'ai vu 48.73105390371401;-357.7257085745805

Au départ ça vient d'une URL indiquée par Mik sur le salon :

https://demarches-publik.entrouvert.com/backoffice/management/signaler-un-incident-de-voirie/map?order_by=-receipt_time&q=&filter=waiting&filter-status=on&id=on&time=on&last_update_time=on&user-label=on&2=on&8=on&30=on&11=on&36=on&status=on

Comme c'est un champ carte, je ne sais pas trop d'où viennent les données, mais potentiellement du système de geoloc du navigateur ?

#5 - 01 décembre 2017 11:35 - Benjamin Dauvergne

Les données viennent du endpoint geojson et donc du champ spécial geoloc, rempli à partir d'une action de workflow Geolocate

https://demarches-publik.entrouvert.com/backoffice/workflows/22/status/just_submitted/items/0/

qui elle prend sa donnée de [form_var_carte], on voit qu'on a bien la mauvaise donnée dans le champ lui même ici:

<https://demarches-publik.entrouvert.com/backoffice/management/signaler-un-incident-de-voirie/109/inspect>

et donc je viens d'essayer il suffit de scroller la carte vers la droite¹ pour faire tourner la terre plusieurs fois et on obtient des coordonnées en dehors de 0..360 (d'ailleurs je me dis que le plus standard c'est peut-être -180..180 en fait, mais bon il me semble que border dans n'importe quel intervalle de 360 degrés suffira).

¹ fait sur <https://demarches-publik.entrouvert.com/intervention/signaler-un-incident-de-voirie/> en utilisant l'inspecteur de firefox pour regarder ce qui se passe dans le champ caché, on peut donc aussi corriger le JS ; je doute que les APIs nous retournent des trucs tordus comme cela, ça vient vraiment plus certainement d'un usage du widget leaflet. On peut aussi regarder du côté de l'option worldCopyJump : <http://leafletjs.com/reference-1.2.0.html#map-worldcopyjump>

#6 - 01 décembre 2017 11:41 - Thomas Noël

je doute que les APIs nous retournent des trucs tordus comme cela

Je pense que c'est quand même possible, ça expliquerait pourquoi on a moitié de demandes en -357° et l'autre en +3° : c'est le téléphone ou le pécé qui envoie ces données et leaflet qui centre la carte dessus.

#7 - 01 décembre 2017 11:51 - Benjamin Dauvergne

Niveau JS on peut faire ça:

```
diff --git a/wcs/qommon/static/js/qommon.map.js b/wcs/qommon/static/js/qommon.map.js
index 8a6cb6e..3867ad3 100644
--- a/wcs/qommon/static/js/qommon.map.js
+++ b/wcs/qommon/static/js/qommon.map.js
@@ -53,6 +53,7 @@ $(window).on('load', function() {
     map.marker.addTo(map);
   }
   map.marker.setLatLng(coords);
+  coords = coords.wrap();
   hidden.val(coords.lat + ';' + coords.lng);
 });
 $map_widget.on('qommon:invalidate', function() {
```

Mais donc là par défaut ça wrap sur -180..180, la latitude mon code est faux c'est l'intervalle -90..90 qu'il faut utiliser.

#8 - 01 décembre 2017 16:16 - Thomas Noël

- Fichier 0001-normalize-geographic-coordinates-into-90.90-180.180-.patch ajouté

Code revu en reprenant la même méthode de wrap que celle de leaflet, histoire de. Je passe autant que faire ce peut en Decimal, parce qu'en float les tests explosent avec des .2499999 & co, logique.

#9 - 01 décembre 2017 16:18 - Thomas Noël

Note : j'ai pas touché la partie javascript, parce que je la comprends mal :/ Mais ça serait bien de faire de coord.wrap(), oui.

#10 - 04 décembre 2017 15:25 - Benjamin Dauvergne

Il faut normaliser la partie MapWidget sinon on laisse une porte ouverte (ok ça sera corrigé au moment de la copie de la donnée vers le champ de geoloc mais autant avoir des données correctes même dans les champs de formulaire, on sait jamais). C'est indépendant du fait de corriger aussi le JS.

#11 - 04 décembre 2017 15:26 - Benjamin Dauvergne

Je dis n'importe quoi ça y est, désolé.

#12 - 04 décembre 2017 15:27 - Benjamin Dauvergne

Ack.

#13 - 04 décembre 2017 16:26 - Thomas Noël

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

```
commit c54766c4e325823bcabccde78fb64f7d59453df6 (HEAD -> master, origin/master, origin/HEAD)
```

Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Fri Dec 1 16:14:55 2017 +0100

```
normalize geographic coordinates into -90..90/-180..180 (#20364)
```

#14 - 04 décembre 2017 16:53 - Frédéric Péters

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

Build cassé. <https://jenkins.entrouvert.org/job/wcs/2995/>

#15 - 04 décembre 2017 17:24 - Thomas Noël

- *Fichier 0001-do-not-try-to-normalize-empty-coordinates-20364.patch ajouté*

#16 - 04 décembre 2017 17:27 - Frédéric Péters

Ok.

#17 - 04 décembre 2017 17:27 - Thomas Noël

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

```
commit d270f6adc7bfa9ac017ca78cdfcfc0b88a9f4679 (HEAD -> master, origin/master, origin/HEAD)
```

Author: Thomas NOEL <tnoel@entrouvert.com>

Date: Mon Dec 4 17:24:12 2017 +0100

```
do not try to normalize empty coordinates (#20364)
```

#18 - 23 décembre 2018 14:44 - Frédéric Péters

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

Fichiers

Capture d'écran de 2017-11-30 11-13-32.png	73,9 ko	30 novembre 2017	Thomas Noël
0001-normalize-geographic-coordinates-into-the-range-0.36.patch	4,65 ko	30 novembre 2017	Benjamin Dauvergne
0001-normalize-geographic-coordinates-into-90.90-180.180-.patch	6,49 ko	01 décembre 2017	Thomas Noël
0001-do-not-try-to-normalize-empty-coordinates-20364.patch	1,04 ko	04 décembre 2017	Thomas Noël