

RFC 5870 : A Uniform Resource Identifier for Geographic Locations ('geo' URI)

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 7 juin 2010. Dernière mise à jour le 7 juillet 2010

Date de publication du RFC : Juin 2010

<http://www.bortzmeyer.org/5870.html>

Ce RFC normalise un nouveau **plan** pour les URI, `geo`, qui permet de représenter l'information concernant la latitude et la longitude sous la forme d'un URI, par exemple `geo:13.4125,103.8667`.

Le projet « plan geo <<http://geouri.org/>> » est un vieux projet de normalisation des URI de localisation géographique. En l'absence d'un tel plan, beaucoup de gens utilisaient des URI `http` comme ceux de Google Maps, ou bien des mécanismes spécifiques à un certain format (cf. section 1 du RFC). Ainsi, la cathédrale Saint-Étienne à Vienne, qui se notait souvent <http://maps.google.com/maps?f=q&source=s_q&hl=fr&sspn=0.355096,0.500565&ie=UTF8&ll=48.208223,16.372687&spn=0.00281,0.003911&t=h&z=18> sera désormais `geo:48.208333,16.372778`. Un tel URI est-il utilisable ? Oui, si on a l'extension Firefox appropriée (cf. <<http://geouri.org/2007/02/26/firefox-extension-ha>> mais la version actuelle est bien vieille et ne marche pas avec les Firefox récents). Comme expliqué par la section 5 du RFC, on pourrait imaginer d'autres applications. Un tel mécanisme va certainement aider tous ceux qui manipulent des informations géographiques sur l'Internet, par exemple le remarquable projet OpenStreetMap qui permettrait d'écrire le lien ci-dessus <<http://www.openstreetmap.org/?lat=48.208333&lon=16.372778&zoom=40>>.

D'autres exemples et scénarios d'usage figurent dans la section 6. Outre la création d'hyperliens classiques pointant vers un service de cartographie (section 6.2), on peut imaginer des code-barres 2D codant un URI `geo` (section 6.3) et permettant à un "smartphone" avec un GPS d'afficher l'itinéraire vers l'endroit indiqué.

Les sections 1 et 2 exposent les principes de base de ce plan, notamment le choix du système de coordonnées WGS 84, entre autre parce que c'est celui du GPS. La syntaxe des URI permettra l'indication d'autres systèmes plus tard mais, pour l'instant, WGS 84 est le système officiel. Dans WGS 84, un point est représenté par une latitude, une longitude et une altitude.

La section 3 est l'enregistrement officiel du plan auprès de l'IANA dans le registre des plans d'URI `<https://www.iana.org/assignments/uri-schemes.html>`, suivant les règles du RFC 4395¹. Elle inclut la syntaxe formelle des URI (section 3.3). De manière résumée, un URI `geo` comprend la chaîne de caractères `geo` : suivie de deux ou trois nombres, séparés par des virgules. Le premier est la latitude (en degrés décimaux), le second la longitude (idem) et le troisième (optionnel) l'altitude (en mètres). Ainsi, `geo:90.0,0.0,100` désigne un point à cent mètres au dessus du Pôle Nord (cf. section 3.4.2 pour les détails et, attention, WGS 84 mesure l'altitude par rapport au géoïde, pas par rapport au niveau de la mer, cf. section 3.4.5).

Un paramètre optionnel (et dont l'usage est actuellement déconseillé), `crs=`, permettra d'indiquer d'autres systèmes de coordonnées que WGS 84. Un autre paramètre optionnel, `u=` indique l'incertitude sur la mesure (section 3.4.3).

Une utilisation courante des URI est la **comparaison**. Ces deux URI sont-ils équivalents? La section 3.4.4 répond à cette question, moins évidente qu'il n'y paraît. La comparaison de deux URI `geo` ne signifie pas en effet qu'ils désignent le même point (ce qui serait trop compliqué à déterminer) mais est plus restrictive que cela : deux URI `geo` sont identiques si tous les nombres sont rigoureusement les mêmes. Donc, `geo:45.0,-10.0` est identique à `geo:45,-10` mais la fonction de comparaison n'essaie pas d'utiliser le paramètre `u=` (l'incertitude) pour considérer que deux points sont proches... (Voir aussi la section 6.4.)

À noter qu'il existe déjà des moyens de noter une position géographique dans un langage formel. Une des plus connues est GML et la section 7 indique comment convertir entre les URI `geo` et les éléments `<Point>` de GML. Ainsi, le code GML :

```
<gml:Point srsDimension="2" srsName="urn:ogc:def:crs:EPSG:6.6:4326">
  <gml:pos>49.40 -123.26</gml:pos>
</gml:Point>
```

deviendrait l'URI `geo:49.40,-123.26`. Et, si vous vous demandez où c'est, voyez `<http://www.openstreetmap.org/?lat=49.4&lon=-123.26&zoom=10>`.

Même si le RFC ne le cite pas, il y avait aussi le mécanisme "*ICBM addresses*" `<http://catb.org/jargon/html/I/ICBM-address.html>`, plus primitif (apparemment pas de norme écrite, pas d'indication de l'altitude, résolution plus faible...)

Deux nouveaux registres IANA créés (sections 4 et 8), pour stocker les paramètres possibles et les valeurs admises pour les paramètres : `<https://www.iana.org/assignments/geo-uri-parameters/geo-uri-parameters.xhtml>`.

Les mises en œuvre de ce RFC sont encore rares et une liste figure sur le site de référence `<http://geouri.org/applications/>`. Mais les progrès que ce plan d'URI sont cruciaux : Comme le note Nicolas Krebs, utiliser `geo` : permet de s'affranchir d'un fournisseur. Ce plan d'adresse est neutre quand au service cartographique utilisé. C'est l'utilisateur qui fait ce choix. `geo:48.208333,16.372778` est à `http://maps.google.com/maps?ll=48.208223,16.372778` ce que `news:hrctmq$ua8$1@news.le-studi` (RFC 5538) est à `http://groups.google.com/group/fr.usenet.divers/msg/cf1d3bd076ba6559`.

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc4395.txt`