

Passerelle - Development #17763

arcgis: endpoint "query" plus générique

23 juillet 2017 10:19 - Frédéric Péters

Statut:	Fermé	Début:	23 juillet 2017
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		

Description

Pour le moment il y a un endpoint /district qui est tout à fait hardcodé pour un cas précis à Nancy, il faudrait à mon sens faire quelque chose de plus générique, un connecteur qui permettrait :

- de renvoyer une liste ou du geojson (deux endpoints ?)
- d'avoir uniquement l'adresse deb base des services REST de l'arcgis dans la config
- de préciser dans l'url (ou lors de la config de requêtes comme on fait dans le csv) le "featureserver" arcgis et le "layer" sur lequel se fait l'interrogation
- de préciser dans l'url lat/lon d'un point, comme aujourd'hui

Révisions associées

Révision 3049f82a - 06 novembre 2018 11:42 - Thomas Noël

generalize ArcGIS connector (#17763)

Révision a00003f1 - 06 novembre 2018 11:42 - Thomas Noël

move ArcGIS from contrib to apps (#17763)

Historique

#1 - 07 juin 2018 00:11 - Thomas Noël

La doc pour faire une query "générique" sur un layer ArcGis : <https://developers.arcgis.com/rest/services-reference/query-map-service-layer.htm>

Le système ArcGis de Cannes : <http://geo.cannes.com:6080/arcgis/rest/>

Exemple de requête :

```
requests.get('http://geo.cannes.com:6080/arcgis/rest/services/geocannes/GEOCANNES_Adresse_reference/MapServer/0/query', params={'f': 'pjson', 'where': "adresse LIKE '%JAURES'", 'outFields': '*', 'inSR': '4326', 'outSR': '4326'})
```

Exemple d'objet retourné avec une géométrie de type point :

```
{u'displayFieldName': u'ident',
 u'features': [
     ...
     {u'attributes': {u'adresse': u'2 RUE JEAN JAURES',
                     u'adresse_complete': None,
                     u'canton': u'CANNES-2',
                     u'codepost': u'06400',
                     u'codinsee': u'06029',
                     u'coorx': 1024770.9,
                     u'coorx': 6280990.9,
                     u'idcanton': u'8',
                     u'ident': u'VOIE_288_2',
                     u'idquart': u'Q_1',
                     u'idsect': u'37',
                     u'nominus': u'Rue Jean Jaur\xe8s',
                     u'nomvoie': u'RUE JEAN JAURES',
                     u'numero': u'2',
                     u'numero_num': 2,
                     u'objectid': 10648,
```

```

        u'quartier': u'Centre-ville',
        u'secteur': u'Gare / Jean Jaur\xe8s',
        u'sectpost': u'Cannes',
        u'typo': u'Num\xe9ros de voies'},
    u'geometry': {u'x': 7.019463636560141,
                  u'y': 43.5535758463704}}},
    ...
],
'fieldAliases': [...],
'fields': [{u'alias': u'OBJECTID',
            u'name': u'objectid',
            u'type': u'esriFieldTypeOID'},
           {u'alias': u'ident',
            u'length': 30,
            u'name': u'ident',
            u'type': u'esriFieldTypeString'}],
    ...

```

Il faudra avoir un mapping/template configurable pour ajouter à chaque élément de réponse un "id" et un "text" ; par défaut ça serait objectid et le displayName indiqué dans le début de réponse.

#2 - 07 juin 2018 00:12 - Thomas Noël

- *Priorité changé de Bas à Normal*

#5 - 02 novembre 2018 18:58 - Thomas Noël

- Fichier 0002-move-ArcGIS-from-contrib-to-apps-17763.patch ajouté
- Fichier 0001-generalize-ArcGIS-connector-17763.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

En première étape un connecteur relativement "pass through". Patch difficile à relire car j'ai voulu garder le nom "arcgis" pour le connecteur, et donc j'ai ré-écrit par dessus l'existant, en veillant à ce que ça migre bien mais surtout en gardant le sale endpoint "district" spécifique à Nancy en place. Puis j'envoie la sauce dans contrib.apps et hop.

Ce connecteur est encore "brut" : il demande de connaître la notion de "folder", "service" et "layer" de l'ArcGIS en face, et il ne permet que des query sur les mapserver. Il ne sort que des datasources (liste de id/text). Les évolutions seront d'avoir un système de pré-configuration des requêtes (comme pour CSV) puis d'interrogation d'un featureserver voire de tuiles. A voir ensuite la possibilité de geojson, moins urgent (pas vraiment besoin de passerelle, arcgis fourni du geojson en natif).

La partie du code concernant le endpoint "district" n'est pas jolie, mais l'objectif est de s'en séparer, sans pour autant le faire dans l'urgence de la mise à jour.

#6 - 02 novembre 2018 19:09 - Frédéric Péters

Au cas où pour les tuiles j'avais <https://git.entrouvert.org/passerelle.git/log/?h=wip/arcgis-tile-poc> (je pense qu'il restait des erreurs de conversions de systèmes de coordonnées).

#7 - 06 novembre 2018 10:45 - Frédéric Péters

- *Statut changé de Solution proposée à Solution validée*

Mouais, ok pour partir sur cette base technique mais j'aimerais qu'on garde en tête un espoir d'API similaire entre arcgis et opengis. (côté connecteur opengis par exemple aujourd'hui on enregistre dans le connecteur la couche où faire la requête, l'endpoint en lui-même, équivalent au mapservice_query, prend juste lat/lon.

#8 - 06 novembre 2018 11:53 - Thomas Noël

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

Frédéric Péters a écrit :

Mouais, ok pour partir sur cette base technique mais j'aimerais qu'on garde en tête un espoir d'API similaire entre arcgis et opengis. (côté connecteur opengis par exemple aujourd'hui on enregistre dans le connecteur la couche où faire la requête, l'endpoint en lui-même, équivalent au mapservice_query, prend juste lat/lon.

Oui c'est vraiment le but ; dans mes patches en cours j'ai un système de query sur ArcGIS, à la csvdatasource, où on dit quel folder, quel service et quel layer et autres sont à taper, et il suffit de fournir lat/lon au endpoint concerné pour obtenir le ou les objets.

Quelque chose d'assez bête et méchant :

```
class MapServerQuery(models.Model):
    resource = models.ForeignKey('ArcGIS')
    slug = models.SlugField(_('Name (slug)'))
    label = models.CharField(_('Label'), max_length=100)
    description = models.TextField(_('Description'), blank=True)

    folder = models.CharField(_('Folder'), max_length=100)
    service = models.CharField(_('Service'), max_length=100)
    layer = models.CharField(_('Layer'), max_length=100, default='')

    where = models.TextField(_('Where statement'), blank=True)

    outfields = models.CharField(_('Out fields'), max_length=300)
    template = models.TextField(_('Template'), blank=True)
    id_template = models.CharField(_('id Template'), max_length=300)
    full = models.BooleanField(_('Full output'), default=False)

    default_args = jsonfield.JSONField(_('Default args (dict)'), null=True)
    ...
```

Bref, c'est poussé, merci.

```
commit a00003f1e97ea26f58301db59e6d616f0f130db2 (HEAD -> master, origin/master, origin/HEAD)
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Fri Nov 2 18:48:50 2018 +0100

    move ArcGIS from contrib to apps (#17763)

commit 3049f82af4c81257ed66f74c1f6a58cc05223e6f
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Fri Nov 2 16:50:32 2018 +0100

    generalize ArcGIS connector (#17763)
```

#9 - 03 décembre 2018 15:14 - Benjamin Dauvergne
- *Statut changé de Résolu (à déployer) à Fermé*

Fichiers

0002-move-ArcGIS-from-contrib-to-apps-17763.patch	5,37 ko	02 novembre 2018	Thomas Noël
0001-generalize-ArcGIS-connector-17763.patch	30,3 ko	02 novembre 2018	Thomas Noël