Utiliser OGDI DataLab

Solution Open Data Clé en main

Publication : Septembre 2014

Version: 1.0

Auteur : Philippe Beraud (Microsoft France)

Pour les dernières informations sur la solution communautaire Open Data Clé en main, consultez

aka.ms/OpenDataFrance

Sommaire

[Résumé 3](#_Toc398293374)

[Commentaires 3](#_Toc398293375)

[Tirer partir du protocole OData 4](#_Toc398293376)

[Requêter OGDI DataLab 5](#_Toc398293377)

[Paramètres des requêtes 6](#_Toc398293378)

[Paramètre $top 7](#_Toc398293379)

[Paramètre $skip 7](#_Toc398293380)

[Paramètre $orderby 8](#_Toc398293381)

[Paramètre $filter 8](#_Toc398293382)

[Format de données 9](#_Toc398293383)

[AtomPub 9](#_Toc398293384)

[JSON 10](#_Toc398293385)

[JSONP 10](#_Toc398293386)

[RDF 10](#_Toc398293387)

[Données géospatiales 10](#_Toc398293388)

[A propos de la pagination 11](#_Toc398293389)

[Télécharger un jeu de données depuis OGDI DataLab 12](#_Toc398293390)

[Télécharger un jeu de données au format CSV 12](#_Toc398293391)

[Télécharger un jeu de données au format DAISY 12](#_Toc398293392)

# Résumé

**OGDI** (Open Government Data Initiative) **DataLab** sous licence libre est une initiative menée par les équipes Microsoft en charge du [Secteur Public](http://www.microsoft.com/france/entreprises/secteur-public/solutions-pour-secteur-public.aspx)[[1]](#footnote-1). Il constitue l’un des accélérateurs de la solution Open Data Clé en main.

OGDI DataLab est une plateforme de publication des données ouvertes OGDI DataLab qui propose une gestion multi-catalogues de (jeux de) données ouvertes et un service de données RESTful [OData (Open Data Protocol)](http://www.odata.org/)[[2]](#footnote-2) pour la consommation des données ainsi mises à disposition.

Ce document décrit l’utilisation du service de données pour consulter les catalogues disponibles, leurs jeux de données, pour requêter ou télécharger un jeu de données, et ce au travers du protocole OData.

# Commentaires

Votre opinion nous intéresse. N’hésitez pas à nous transmettre vos commentaires via la boîte aux lettres Open Data France (<mailto:ogdifrance@live.fr>) afin que nous puissions améliorer OGDI DataLab, l’un des accélérateurs de la solution Open Data Clé en main disponible sur la forge GitHub.

Le [blog MSDN Open Data France](http://blogs.msdn.com/b/ogdifrance/)[[3]](#footnote-3) vous tient informé dans le même temps des dernières évolutions de la solution Open Data Clé en main, des différents accélérateurs proposés dans ce contexte, de l’actualité et des nouveautés relatives à ceux-ci.

# Tirer parti du protocole OData

La plateforme de publication de données OGDI DataLab expose un service de données Web RESTful conforme au protocole ouvert de donnée [OData](http://www.odata.org/) (Open Data Protocol).

Le protocole OData constitue une extension naturelle de technologies Web largement acceptées telles que http, service Web RESTful, XML, AtomPub, JSON, JSONP, etc. pour :

* Permettre aux sources de données, qu’il s’agisse de façon non exhaustive de bases de données relationnelles, de systèmes de fichiers, de systèmes de gestion de contenu, de sites Web traditionnels, de sites collaboratifs, de tables NoSQL, etc., d'être exposées sur le Web de façon simple, sécurisée et interopérable ;
* Et fournir ainsi un accès à l’information depuis un large éventail d’applications, de services et de magasins/stockages de données.

OData met à disposition un protocole et un langage de requête pour les données ouvertes. Ceci permet d’interroger, avec toute la sémantique associée, les différents jeux de données ainsi mis à disposition via potentiellement une seule et unique interface technique indépendante des jeux de données. OData résulte et tire bénéfice des expériences de mise en œuvre de serveurs et de clients AtomPub et exploite les identificateurs de ressource uniformes (URI) pour l'identification des ressources et une interface uniforme http permettant d'interagir avec ces ressources, tout comme le ferait un utilisateur avec n'importe quelle application sur le Web. Ainsi, les (jeux de) données et leur logique associée sont exposées comme un flux (feed) OData. Sur cette base, de simples requêtes http sur l’URL correspondante permettent de consommer les données. Il est bien entendu possible d’effectuer des opérations de filtrage, de tri ou d’agrégation en rajoutant certains paramètres à l’URL.

**Ceci évite de devoir définir et développer une API ou une interface propre à chaque jeu de données et type de données et favorise au passage les mashup de données et les applications composites.**

Une telle approche permet ainsi d’exposer les données ouvertes (open data) directement pour la programmation. Des fichiers peuvent être restitués dans différents formats si besoin comme nous l’abordons à la fin de ce document. Les spécifications du protocole ouvert de données OData (Open Data Protocol) sont un [standard de l’OASIS](https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=odata).

# Requêter OGDI DataLab

Comme nous venons de l’aborder, le service de données proposé par OGDI DataLab est un service Web implémenté à l'aide du protocole HTTP et des principes REST, pour consommer les différents catalogues de jeux de données publiés.

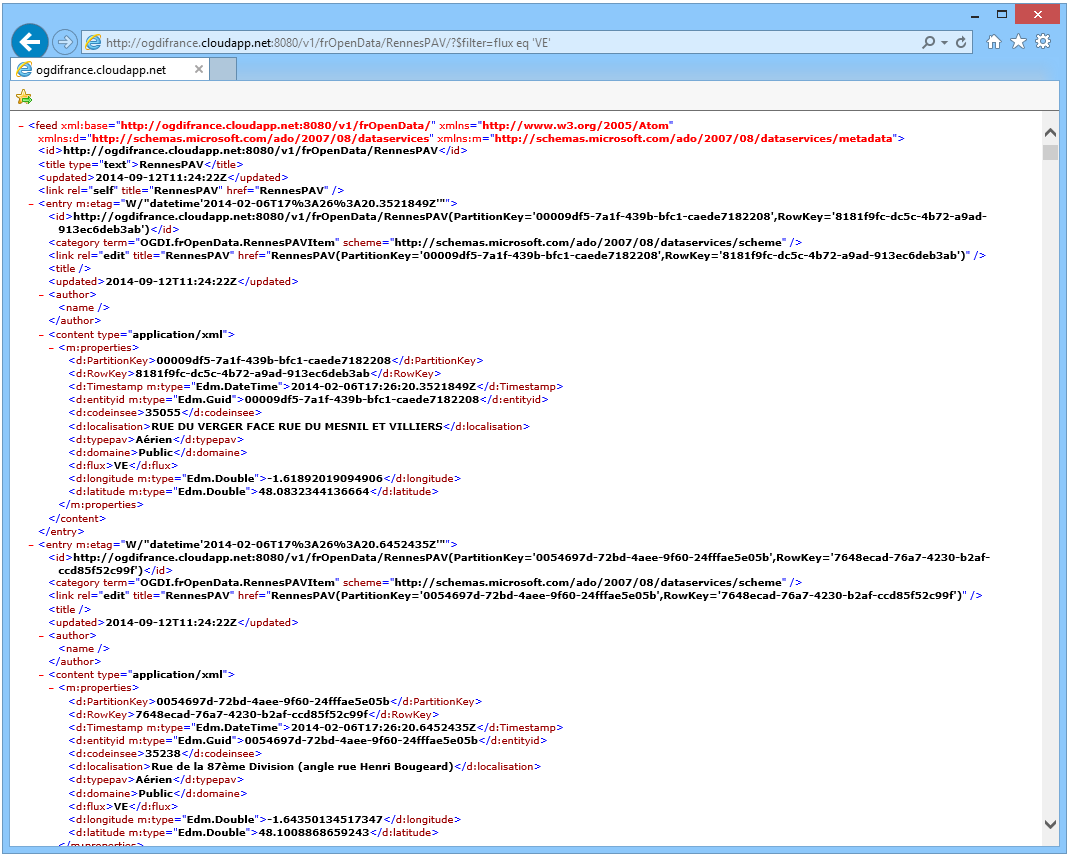
Le format de base des URLs pour invoquer le service de données d’une instance de la plateforme de publication est la suivante:

http://***projet***.cloudapp.net[:8080]/v1/***catalogue***/***jeudedonnees***?***requete***

où :

* ***projet*** est le nom du projet de déploiement pour la plateforme de publication OGDI DataLab dans Azure (par exemple *ogdifrance* pour la plateforme de démonstration mise à disposition par Microsoft France). Le port 8080 est utilisé à partir de la version 2013-06 (v6) d’OGDI DataLab).
* ***catalogue*** est le nom du catalogue/conteneur (par exemple, *frOpenData* pour les jeux de données du catalogue de démonstration par défaut) ;
* ***jeudedonnees*** est le nom du jeu de données (par exemple, *RennesPAV* pour le jeu des données localisant les points d'apport volontaire (PAV) des déchets ménagers mis à disposition par le [Service SIG Rennes Métropole](http://www.data.rennes-metropole.fr/les-donnees/catalogue/?tx_icsopendatastore_pi1%5buid%5d=47)) ;
* ***requete*** est l’ensemble des paramètres de la requête, exprimé en utilisant un sous ensemble de la [syntaxe de requête du protocole OData.](http://www.odata.org/documentation/)

Remarque OGDI DataLab ne supporte actuellement que les options de requête OData $top, $skip, $orderby, $filter et $top comme décrit dans la suite de ce document.



Comme tous les services conformes OData, le service de données expose un « sommaire ». Ainsi, la requête suivante permet de lister l’ensemble des catalogues exposés :

<http://ogdifrance.cloudapp.net:8080/v1/>

En complément, la requête suivante retourne les métadonnées ODGI DataLab associées à chaque catalogue :

<http://ogdifrance.cloudapp.net:8080/v1/AvailableEndpoints>

De même, comme tous les services conformes OData, le service de données expose un « sommaire » pour chaque catalogue exposé. Ainsi la requête suivante :

<http://ogdifrance.cloudapp.net:8080/v1/frOpenData>

retourne pour le catalogue *frOpenData* l’ensemble des jeux de données associés. En OData, le catalogue correspond à un service, l’ensemble des jeux de données à un espace de travail et un jeu de données à une collection.

Un catalogue expose par ailleurs ces métadonnées. Ainsi la requête suivante :

<http://ogdifrance.cloudapp.net:8080/v1/frOpenData/$metadata>

retourne le modèle d’entité associé pour chaque jeu de données ainsi exposé. En complément, les requêtes suivantes retournent les métadonnées de chaque jeu de données ainsi exposé en XML respectivement en JSON :

<http://ogdifrance.cloudapp.net:8080/v1/frOpenData/TableMetadata>

<http://ogdifrance.cloudapp.net:8080/v1/frOpenData/TableMetadata?$format=json>

Comme décrit au niveau de la syntaxe OData précédente, l’accès à un jeu de données spécifique au sein d’un catalogue s’effectue simplement à ajoutant le nom du jeu de données à la fin de la requête. Ainsi la requête suivante :

<http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV>

Retourne les 1000 premières lignes du jeu de données *RennesPAV*. Des requêtes exemples utilisant le jeu des données *RennesPAV* sont disponibles sur la page [visualisation des données](http://ogdifrance.cloudapp.net/Databrowser/frOpenData/RennesPAV) de l’instance de test mise à disposition par Microsoft France. La section suivante précise les principaux paramètres pris en charge au niveau des requêtes

A noter que si une propriété d'un jeu de données a une valeur nulle, elle sera inexistante dans le résultat retourné par le service de données. Par exemple, si dans un jeu de données vous avez la propriété ville et arrondissement et que votre requête ne retourne que des villes sans arrondissement, le champ arrondissement ne sera pas retourné. Votre application doit prendre en compte la possible absence d'un champ dans le résultat retourné.

## Paramètres des requêtes

Le service de données RESTful gère un ensemble de mots-clés OData qui peuvent être insérés dans l’URL de requête. Ces mots clés permettent de personnaliser les résultats retournés par le service de données afin d’offrir une meilleure flexibilité et de se rapprocher au maximum de la spécification OData en termes de couverture fonctionnelle au niveau de l’implémentation proposée en tant que producteur OData.

### Paramètre $top

Le paramètre $top est disponible dans toutes les versions d’OGDI DataLab. La version 6 d’OGDI DataLab permet de spécifier la valeur « all » pour le paramètre $top ; ce qui a pour effet de retourner l’ensemble des résultats d’un jeu de données (dataset) sans limiter du nombre de résultat.

Pour rappel, le service de données retourne par défaut les 1000 premiers résultats d’un jeu de données et le flux ainsi retourné contient à la fin des résultats de cette première requête http un élément de type link contenant l’URL pour obtenir les enregistrements suivants, et ainsi de suite jusqu’à ce que l’on ait parcouru tous les résultats. Pour ce faire, l’URL contient un [jeton de continuation](http://blogs.msdn.com/b/ogdifrance/archive/2011/12/20/powerpivot-pour-excel-pagination-et-projections.aspx) $skiptoken.

Le tableau ci-dessous présente quelques exemples d’utilisation du paramètre $top :

|  |  |
| --- | --- |
| Requête | Description |
| Pas de $top  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse) | Retourne un maximum de 1000 lignes |
| $top=2  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$top=2**](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$top=2) | Retourne un maximum de 2 lignes |
| $top=2390  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$top=2390**](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/Arbres?$top=2390) | Retourne un maximum de 2390 lignes |
| $top=all  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$top=all**](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/Communes?$top=all) | Retourne toutes les lignes |

Le service de données se conforme à la spécification OData pour le paramètre $top et ce comme décrit [ici](http://www.odata.org/documentation/odata-v2-documentation/uri-conventions/#43_Top_System_Query_Option_top).

### Paramètre $skip

Le paramètre $skip est disponible uniquement dans la version 6 d’OGDI DataLab.

Le paramètre $skip permet d’omettre les X premières lignes du résultat d’une requête (X étant la valeur du paramètre $skip).

Le tableau ci-dessous présente quelques exemples d’utilisation du paramètre $skip :

|  |  |
| --- | --- |
| Requête | Description |
| $skip=2  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$skip=2**](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$skip=2) | Les 2 premiers résultats seront omis |
| $skip=5&$top=10  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$skip=5&$top=10**](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$skip=5&$top=10) | Retourne 10 résultats à partir du 5ème |

Il convient de noter qu’il ne faut pas utiliser le mot clé $skip comme un moyen de pagination. Un lien prévu à cet effet est spécifié en bas de la réponse comme décrit précédemment.

Le service de données se conforme à la spécification OData pour le paramètre $skip et ce comme décrit [ici](http://www.odata.org/documentation/odata-v2-documentation/uri-conventions/#44_Skip_System_Query_Option_skip).

### Paramètre $orderby

Le paramètre $orderby est disponible uniquement dans la version 6 d’OGDI DataLab. Ce paramètre permet de demander au service de données de trier les résultats avant de les retourner. Il est aussi possible de spécifier l’ordre de tri, à savoir asc pour l’ordre croissant et desc pour l’ordre décroissant. Si aucun ordre n’est spécifié, c’est l’ordre croissant qui est utilisé par défaut.

Le tableau ci-dessous présente quelques exemples d’utilisation du paramètre $orderby :

|  |  |
| --- | --- |
| Requête | Description |
| $orderby=mairie  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$orderby=mairie**](http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$orderby=mairie) | Trier par mairie croissant |
| $orderby=insee desc  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?**$orderby=insee desc**](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$orderby=insee%20desc) | Trier par insee décroissant |

Le service de données se conforme à la spécification OData pour le paramètre $orderby et ce comme décrit [ici](http://www.odata.org/documentation/odata-v2-documentation/uri-conventions/#42_Orderby_System_Query_Option_orderby).

### Paramètre $filter

Le paramètre $filter est disponible dans toutes les versions d’OGDI DataLab. La version 6 permet par contre désormais d’utiliser des fonctions de chaîne de caractères, des fonctions de date et des fonctions de nombre dans l’URL de la requête.

Le tableau ci-dessous détaille l’ensemble des fonctions prises en charge :

|  |
| --- |
| Fonctions |
| Fonctions de chaînes de caractères |
| Bool substringof(string base, string find)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=substringof(mairie, "EAU") eq true](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=substringof(mairie,%20%22EAU%22)%20eq%20true) |
| Bool endswith(string base, string end)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=endswith(mairie, "ELLE") eq true](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=endswith(mairie,%20%22ELLE%22)%20eq%20true) |
| Bool startswith(string base, string start)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=startswith(mairie, "BEAU") eq true](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=startswith(mairie,%20%22BEAU%22)%20eq%20true) |
| Int length(string base)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=length(mairie) eq 5](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=length(mairie)%20eq%205) |
| Int indexof(string base, string find)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=indexof(mairie, 'Z') eq 4](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=indexof(mairie,%20'Z')%20eq%204) |
| String replace(string base, string find, string replace)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=replace(mairie, "ZELLE", "") eq "BEAU"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=replace(mairie,%20%22ZELLE%22,%20%22%22)%20eq%20%22BEAU%22) |
| String substring(string base, int pos)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=substring(mairie, 4) eq "ZELLE"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=substring(mairie,%204)%20eq%20%22ZELLE%22http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=substring(mairie,%204)%20eq%20%22ZELLE%22) |
| String substring(string base, int pos, int length)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=substring(mairie, 1, 3) eq "EAU"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=substring(mairie,%201,%203)%20eq%20%22EAU%22) |
| String tolower(string base)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=tolower(mairie) eq "beauzelle"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=tolower(mairie)%20eq%20%22beauzelle%22) |
| String toupper(string base)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=toupper(mairie) eq "BEAUZELLE"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=toupper(mairie)%20eq%20%22BEAUZELLE%22) |
| String trim(string base)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=trim(mairie) eq "BEAUZELLE"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=trim(mairie)%20eq%20%22BEAUZELLE%22) |
| String concat(string str1, string str2)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=concat(mairie, " CITY") eq "BEAUZELLE CITY"](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=concat(mairie,%20%22%20CITY%22)%20eq%20%22BEAUZELLE%20CITY%22) |
| Fonctions de date |
| Int day(DateTime date)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=day(date\_ajout) eq 28](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=day(date_ajout)%20eq%2028) |
| Int hour(DateTime date)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=hour(date\_ajout) eq 00](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=hour(date_ajout)%20eq%2000) |
| Int minute(DateTime date)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=minute(date\_ajout) eq 01](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=minute(date_ajout)%20eq%2001) |
| Int month(DateTime date)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=month(date\_ajout) eq 02](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=month(date_ajout)%20eq%2002) |
| Int second(DateTime date)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=second(date\_ajout) eq 09](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=second(date_ajout)%20eq%2009) |
| Int year(DateTime date)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=year(date\_ajout) eq 2012](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=year(date_ajout)%20eq%202012) |
| Fonctions sur les nombres |
| Double round(double nb)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=round(nombre\_test) eq 42](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=round(nombre_test)%20eq%2042) |
| Double floor(double nb)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=floor(nombre\_test) eq 42](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=floor(nombre_test)%20eq%2042) |
| Double ceiling(double nb)  [http://ogdifrance.cloudapp.net:8080/v1/catalogue/jeudedonnees?$filter=ceiling(nombre\_test) eq 43](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/MairiesGrandToulouse?$filter=ceiling(nombre_test)%20eq%2043) |

Le service de données se conforme à la spécification OData pour le paramètre $filter et ce comme décrit [ici](http://www.odata.org/documentation/odata-v2-documentation/uri-conventions/#45_Filter_System_Query_Option_filter).

## Format de données

### AtomPub

Par défaut, la plateforme de publication OGDI DataLab retourne les données dans le format du [protocole Odata (Open Data Protocol)](http://odata.org/documentation). Ce format étend le protocole largement utilisé [Atom Pub](http://tools.ietf.org/html/rfc5023) et peut être utilisé par une variété de plateformes incluant .NET, Java, Ruby, PHP et Python. Différents exemples de code sont proposés sur la page de visualisation des données [ici](http://ogdifrance.cloudapp.net/DataBrowser/frOpenData/RennesPAV#param=NOFILTER--DataView--Results) de la plateforme de démonstration mise à disposition par Microsoft France.

### JSON

Le service de données d’OGDI DataLab peut aussi retourner les données au format [JSON (JavaScript Object Notation)](http://fr.wikipedia.org/wiki/JSON) qui peut être plus facilement exploitable depuis JavaScript ou autre technologie. Pour retourner des données au format JSON, il suffit d'ajouter format=json à votre requête. Par exemple, pour retourner les points d'apport volontaire des déchets ménagers en verre pour la métropole de Rennes au format JSON :

[http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV/?$filter=flux eq 'VE' &format=json](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV/?$filter=flux%20eq%20'VE'%20&format=json)

### JSONP

Pour diminuer la vulnérabilité en terme de sécurité liée aux attaques de type scriptage intersite (cross-site scripting ou XSS en abrégé), généralement les navigateurs empêchent le JavaScript d'une page Web venant d'un domaine X de faire des requêtes http vers un autre domaine Y. Cela empêche tout code JavaScript hébergé sur un domaine de faire des appels directs au service de données OGDI DataLab, mais il existe plusieurs techniques qui peuvent être utilisées comme la technique fondée sur des IFrames.

OGDI DataLab fournit un support direct pour la technique [JSONP](http://fr.wikipedia.org/wiki/JSON#JSONP). En utilisant cette technique, le service de données OGDI DataLab appelle la fonction de rappel (callback) que vous aurez spécifiée, passant le résultat de votre requête au format JSON comme format d'entrée. Pour utiliser cette technique, il convient de construire votre requête avec les paramètres additionnels *format=json&callback=* où est le nom de votre fonction de rappel JavaScript définie dans la page émettant la requête.

Vous pouvez consulter l'exemple de code JavaScript sur la page Visualisation des données [ici](http://ogdifrance.cloudapp.net/Databrowser/frOpenData/RennesPAV) de la plateforme de démonstration mise à disposition par Microsoft France pour un exemple utilisant JSONP avec OGDI. Dans cet exemple, la fonction AdditionalDataLoaded() est la fonction de callback JSONP.

### RDF

Le service de données de la plateforme de publication OGDI DataLab peut aussi retourner les données au format [RDF (Resource Description Framework)](http://fr.wikipedia.org/wiki/Resource_Description_Framework). RDF est un modèle de graphe qui permet de décrire de façon formelle des ressources web et leurs métadonnées associées.

Développé par le W3C, RDF est le langage de base du Web sémantique. Pour retourner des données au format RDF, il suffit d’ajouter format=rdf à votre requête. Par exemple, pour retourner les points d’apport volontaire des déchets ménagers en verre pour la métropole de rennes au format RDF :

[http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV/?$filter=flux eq 'VE' &format=rdf](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV/?$filter=flux%20eq%20'VE'%20&format=rdf)

### Données géospatiales

La plupart des jeux de données ainsi proposés est susceptible d’inclure également des données géo spatiales, qui peuvent être alors retournées dans le format [KML (Keyhole Markup Language)](http://fr.wikipedia.org/wiki/Keyhole_Markup_Language), rendant le Kit de démarrage OGDI compatible avec des technologies de cartographie sur poste de travail ou sur le Web, comme [Géoportail](http://www.geoportail.fr/), [Google Earth](http://earth.google.com/), [Google Maps](http://maps.google.com/), ou encore [Yahoo! Maps](http://maps.yahoo.com/).

Pour retourner des données géospatiales au format KML, il suffit de rajouter *format=kml* à la requête. Par exemple, pour retourner les points d'apport volontaire des déchets ménagers en verre pour la métropole de Rennes au format KML :

[http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV/?$filter=flux eq 'VE' &format=kml](http://ogdifrance.cloudapp.net:8080/v1/frOpenData/RennesPAV/?$filter=flux%20eq%20'VE'%20&format=kml)

Remarque Si le jeu de données ne contient pas de données de géolocalisation, une requête de type KML au service de données OGDI DataLab retournera un résultat vide.

### A propos de la pagination

Le service de données OGDI DataLab et le service sous-jacent de tables Azure supportent la pagination pour les résultats d'un volume important. L’article MSDN [Délai d'expiration de requête et pagination](http://msdn.microsoft.com/fr-fr/library/dd135718.aspx) de la documentation Azure une description complète sur la façon dont OGDI DataLab et la plateforme Azure prennent en charge la pagination. Vous pouvez vous référer également au code exemple "Pagination C#/ASP.NET" la page Visualisation des données [ici](http://ogdifrance.cloudapp.net/Databrowser/frOpenData/RennesPAV) de la plateforme de démonstration mise à disposition par Microsoft France, qui illustre comment gérer la pagination en utilisant la bibliothèque cliente [WCF Data Services](http://msdn.microsoft.com/fr-fr/data/bb931106.aspx).

# Télécharger un jeu de données depuis OGDI DataLab

Au-delà du requêtage d’un jeu de données et de la consommation du résultat sous forme de flux, la plateforme de publication OGDI Datalab permet de télécharger un jeu de données ou un sous-ensemble de celui-ci sous forme de fichier.

## Télécharger un jeu de données au format CSV

Pour télécharger un jeu de données au format CSV, la syntaxe de l’URL est la suivante :

http://***projet***.cloudapp.net/DataBrowser/DownloadCsv?container=***catalogue***&entitySet=***jeudedonnees***&filter=***filtre***

où :

* ***projet*** est le nom du projet de déploiement de plateforme de publication OGDI DataLab (par exemple *ogdifrance* pour la plateforme de démonstration mise à disposition par Microsoft France).
* ***catalogue*** est le nom du catalogue de jeux de données (par exemple, *frOpenData*).
* ***jeudedonnees*** est le nom du jeu de données (par exemple, *RennesPAV* vis-à-vis du jeu des données localisant les points d'apport volontaire (PAV) des déchets ménagers mis à disposition par le [Service SIG Rennes Métropole](http://www.data.rennes-metropole.fr/les-donnees/catalogue/?tx_icsopendatastore_pi1%5buid%5d=47)).
* ***filtre*** est l’option de filtrage à appliquer avant de retourner le jeu de données. La valeur NOFILTER permet de retourner l’intégralité du jeu de données. L’ensemble des mots–clé décrits précédemment pour le paramètre $filter sont pris en charge.

A titre, d’illustration, la requête suivante :

<http://ogdifrance.cloudapp.net/DataBrowser/DownloadCsv?container=frOpenData&entitySet=RennesPAV&filter=NOFILTER>

retourne le jeu de données "*RennesPAV*" sous la forme du fichier *RennesPAV.csv*.

## Télécharger un jeu de données au format DAISY

Pour télécharger un jeu de données au format DAISY, la syntaxe de l’URL est la suivante :

http://***projet***.cloudapp.net/DataBrowser/Download?container=***catalogue***&entitySet=***jeudedonnees***&filter=***filtre***

où :

* ***projet*** est le nom du projet de déploiement de plateforme de publication OGDI DataLab (par exemple "*ogdifrance*" pour la plateforme de démonstration mise à disposition par Microsoft France).
* ***catalogue*** est le nom du catalogue de jeux de données (par exemple, *frOpenData*).
* ***jeudedonnees*** est le nom du jeu de données (par exemple, *RennesPAV* vis-à-vis du jeu des données localisant les points d'apport volontaire (PAV) des déchets ménagers mis à disposition par le [Service SIG Rennes Métropole](http://www.data.rennes-metropole.fr/les-donnees/catalogue/?tx_icsopendatastore_pi1%5buid%5d=47)).
* **filtre** est l’option de filtrage à appliquer avant de retourner le jeu de données. La valeur NOFILTER permet de retourner l’intégralité du jeu de données. L’ensemble des mots–clé décrits précédemment pour le paramètre $filter sont pris en charge.

A titre, d’illustration, la requête suivante :

<http://ogdifrance.cloudapp.net/DataBrowser/Download?container=frOpenData&entitySet=RennesPAV&filter=NOFILTER>

retourne le jeu de données "*RennesPAV*" sous la forme du fichier *RennesPAV.xml*.

1. Site portail Web Microsoft Secteur Public : http://www.microsoft.com/france/entreprises/secteur-public/solutions-pour-secteur-public.aspx [↑](#footnote-ref-1)
2. Protocole OData : http://www.odata.org [↑](#footnote-ref-2)
3. Blog MSDN Open Data France : http://aka.ms/OpenDataFrance/ [↑](#footnote-ref-3)