



Documentation utilisateurs API Carto module codes postaux

Version 1.0

SOMMAIRE

I Description générale du module codes postaux d'API Carto	2
I.1 Présentation d'API Carto	2
I.2 Présentation du module codes postaux d'API Carto	2
I.3 Fonctionnement général du module codes postaux d'API Carto	2
I.4 Mode d'interrogation des données disponibles	3
I.4.a Requête par code postal	3
I.4.b Format du résultat en sortie	3

GLOSSAIRE

API	« application programming interface », « interface de programmation applicative ».
API Carto	API développée initialement dans le cadre d'un PIA (projet d'investissement de d'avenir) dans le but de faciliter l'usage de la cartographie dans les formulaires administratifs
COG	Code Officiel de la Géographie édité par l'Institut National de la Statistique et des Études Économiques
FIMOCT	Fichier de structure de la Direction Générale des Finances Publiques

I Description générale du module codes postaux d'API Carto

I.1 Présentation d'API Carto

API Carto est une brique logicielle offrant des webservices de traitements et de calculs, facilement intégrables dans les interfaces avec les usagers des services publics et reposant sur un ensemble de données géographiques de référence détenues par différents organismes. L'API Carto est aujourd'hui composée de différents modules thématiques : RPG, Cadastre, Codes Postaux, AOC, et Urbanisme.

Les services proposés permettent de croiser des données entre elles, et donc de récupérer automatiquement certaines informations requises dans des formulaires administratifs (ex. : pour une demande de permis de construire, l'identifiant de la parcelle cadastrale peut souvent être directement obtenu à partir de l'adresse) ou d'extraire et télécharger des données.

API Carto a vocation à être utilisée par les utilisateurs finaux au travers de services clients.

Les grands principes techniques de l'API Carto sont les suivants :

- API Carto est une collection API Rest respectant la spécification OpenAPI,
- le format utilisé pour les données est JSON/GeoJSON,
- la projection utilisée est WGS84 (coordonnées en longitude, latitude),
- les API offrent des opérations génériques de filtrage simple:
 - par attribut (?nom_attribut=valeur),
 - par intersection géométrique (?geom=géométrie GeoJSON),
- l'enchaînement d'appels successifs aux différents API (pour répondre aux besoins métiers) est à réaliser côté client,
- les traitements géométriques métiers (calcul de surface, filtrage des résultats, etc.) sont réalisés côté client, à l'aide de bibliothèques de calcul géométrique.

Une documentation technique est disponible en ligne pour plus d'informations : <https://apicarto.ign.fr/api/doc/>.

I.2 Présentation du module codes postaux d'API Carto

Le module codes postaux d'API Carto permet d'obtenir la liste des communes correspondant à un code postal.

Le module utilise un fichier de données entretenu par Etalab à partir des données COG et du FIM OCT.

La documentation technique du module codes postaux de l'API Carto est disponible à cette adresse : <https://apicarto.ign.fr/api/doc/codes-postaux>.

I.3 Fonctionnement général du module codes postaux d'API Carto

Les requêtes ne peuvent se faire qu'en mode GET.

L'outil client doit envoyer un code postal vers le module.

Le module récupère la liste des communes correspondant au code postal saisi et la renvoie à l'outil client au format JSON.

I.4 Mode d'interrogation des données disponibles

I.4.a Requête par code postal

Il n'y a qu'un mode d'interrogation via l'envoi d'un code postal dans le paramètre. Un seul code postal peut être pris en paramètre. L'interrogation du module avec plusieurs codes postaux n'est pas permise. Si le besoin de l'utilisateur concerne plusieurs codes postaux, il faudra que l'outil client enchaîne plusieurs requêtes API Carto (une par code postal), puis traite les réponses successives de l'API pour présenter un résultat unifié à l'utilisateur final.

I.4.b Format du résultat en sortie

Le résultat de la requête est renvoyé par l'API au format JSON.

Il contient la liste des communes desservies par code postal saisi en paramètre avec leurs

- code postal
- code INSEE
- nom de commune
- libellé d'acheminement

Exemple de fichier résultat pour le code postal 11000 :

```
[
{
  "codePostal": "11000",
  "codeCommune": "11037",
  "nomCommune": "Berriac",
  "libelleAcheminement": "BERRIAC"
},
{
  "codePostal": "11000",
  "codeCommune": "11069",
  "nomCommune": "Carcassonne",
  "libelleAcheminement": "CARCASSONNE"
}
]
```