

Portail CFADock

Spécifications de l'API CFADock : v1.2

Objet :

Ce document a pour objectif de décrire l'API JSON du service CFADock.fr

Table des matières

I.	Principe général.....	3
	URL d'accès.....	3
	Swagger	3
	Limitations	3
II.	GetVersion.....	4
III.	Récupération des informations sur un OPCO.....	5
IV.	Recherche par lot	7
V.	Recherche par SIRET ou IDCC unitaire.....	8
VI.	Recherche par lot	10

I. Principe général

L'API CfaDock permet de récupérer l'OPCO auquel est rattaché une société via son Siret/Siren, ou de connaître l'OPCO associé à une convention collective donnée, de la même manière que le site CfaDock.fr lui-même.

L'API fournit un appel unitaire et un appel par lot qu'il s'agit de privilégier si plusieurs recherches doivent être faites simultanément pour des raisons de performance.

URL d'accès

L'URL d'accès au service est la même que le Servie CFADock suivi de « /api ».

GET : <https://www.cfadock.fr/api/>

Swagger

Open API est active afin de récupérer les méta-data de l'API. L'URL d'accès est la suivante :

<https://www.cfadock.fr/swagger/>

Limitations

L'accès au service n'est pas authentifié, il y a toutefois une limitation au nombre d'appels par minute autorisé. La règle est la suivante :

- Le nombre de requête possible pour une IP donné est de **30 / minutes**
- Si le nombre d'appel dépasse le seuil. Un message avec la chaîne « API calls quota exceeded! » est retourné.

Une autre limitation est définie au niveau de la requête par lot. Une même requête ne peut pas demander plus de **100** recherches en 1 seul appel.

II. GetVersion

Rôle : cette méthode permet de la version de l'application CFADock et la version de l'API.

Type de méthode : GET

Route : /api/version

Exemple :

- <https://www.cfadock.fr/api/version>

Réponse :

Le résultat retourné est un objet contenant la version actuelle de l'API et la version du service :

```
{
  "CfaDockVersion" : "v1.2.3"
  "ApiVersion" : "v1.0"
}
```

Propriétés :

- **CfaDockVersion** : Version de CfaDock.fr
- **ApiVersion** : Version de l'API

III. Récupération des informations sur un OPCO

Rôle : ces méthodes permettent de récupérer les informations retournées par le Site dans les fiches Opco. Il est possible de récupérer les informations globales de l'OPCO et de ses sur la base de son Siren, ou d'avoir un retour spécifiquement sur un établissement et le précisant via un Siret ou par un couple Siren/Code

Type de méthode : GET

Routes :

/api/opco/{siren_siret}

/api/opco/{siren}/{code}

Remarque : Si {code}=0 alors aucun établissement n'est retourné

Exemples sur l'OpCommerce:

- <https://www.cfadock.fr/api/opco/398522243>
- <https://www.cfadock.fr/api/opco/398522243/0>
- <https://www.cfadock.fr/api/opco/398522243/227>

Réponse :

Le résultat retourné est un objet au format JSON contenant les informations sur l'OPCO et ses établissements. Si le champ n'est pas défini dans CFADock, le champ est retourné avec la valeur « null »

```
{
  "color": "#80d33a",
  "description": "L'Opcommerce est le partenaire Alternance-Compétences des entreprises du commerce, soit 19 branches professionnelles.\r\n\r\nAgréé par l'Etat, il accompagne, en proximité, la performance des entreprises, contribue à l'évolution professionnelle des salariés et favorise l'insertion professionnelle des jeunes dans le secteur du commerce grâce aux formations en alternance (contrat d'apprentissage et contrat de professionnalisation).\r\nL'Opcommerce conduit l'ensemble de ses actions en synergie avec les acteurs emploi-formation présents dans les territoires.\r\n",
  "url": "https://www.lopcommerce.com",
  "name": "L'Opcommerce",
  "siren": 398522243,
  "sites": [
    {
      "address": "Z.I. des Granges Galand, 27 rue des Granges Galand ",
      "city": "SAINT-AVERTIN",
      "code": 227,
```

```

"contactEmail": "lopcommerce37@lopcommerce.com",
"contactFirstName": null,
"contactFunction": null,
"contactLastName": null,
"contactPhone": "02 47 71 01 01",
"description": null,
"url": null,
"name": "Délégation Régionale Centre Val de Loire",
"zipCode": "37550"
}
]
}

```

Propriétés :

- **Color** : Couleur dominante de la charte graphique de l'Opco.
- **Description** : Chaîne de caractère contenant la description de l'OPCO
- **URL** : URL du site institutionnel de l'OPCO
- **Name** : Nom de l'OPCO
- **Siren** : SIREN de l'OPCO
- **Sites** : Liste des établissements (Si demandé) Chaque objet de cette liste contient les propriétés suivantes :
 - **Name** : Nom de l'établissement
 - **Description** : Description spécifique à l'établissement
 - **URL** : url spécifique sur l'établissement
 - **Address** : Adresse postale de l'établissement
 - **City** : Ville où est localisé l'établissement
 - **ZipCode** : Code postal de l'établissement
 - **Code** : Code Etablissement du SIRET
 - **ContactEmail** : Email de contact
 - **ContactFirstName** : Prénom du contact
 - **ContactLastName** : Nom du contact
 - **ContactFunction** : Fonction du contact
 - **ContactPhone** : Numéro de téléphone de contact

IV. Recherche par lot

Rôle : c'est méthode permet de réaliser une recherche par lot en un seul appel. Cela permet d'éviter des requêtes en boucle (Voir limitations d'appels). Le principe est le même que l'appel unitaire à ceci près que le filtre est défini dans un objet du BODY de la requêtes.

Type de méthode : POST

Route : /api/opcos

Paramètres :

Objet JSON contenant un tableau de filtre :

```
[
  {
    "Siret" = "987654321",
    "IDCC" = 123
  },
  {
    "Siret" = "987654321",
    "IDCC" = 123
  }
]
```

Le tableau contient des objets dont les propriétés sont les suivantes :

- **siret** : [Optionnel] Siret de la société recherchée. La définition d'un Siren est possible par ce même paramètre. Format chaîne de caractère avec préfixes à 0 (Optionnel).
- **idcc** : [Optionnel] Numéro de convention collective recherché. Format numérique

V. Recherche par SIRET ou IDCC unitaire

Rôle : cette méthode permet de récupérer le SIREN de l'OPCO associé à une recherche par SIREN, SIRET ou IDCC.

Type de méthode : GET

Route : /api/opcos

Paramètres type query-string:

Au moins un des 2 paramètres suivant doit être défini :

- **siret** : [Optionnel] Siret de la société recherchée. La définition d'un Siren est possible par ce même paramètre. Format chaîne de caractère avec préfixes à 0 (Optionnel).
- **idcc** : [Optionnel] Numéro de convention collective recherché. Format numérique.

Exemples :

- <https://www.cfadock.fr/api/opcos?siret=123456789>
- <https://www.cfadock.fr/api/opcos?idcc=123>

Réponse :

Le résultat retourné est un objet au format JSON contenant à la fois les critères de recherche envoyés et l'identification de l'OPCO:

```
{
  "Filter" : {
    "Siret" = "987654321",
    "IDCC" = 123
  },
  "FilterNAFCode" : "68.20B",
  "FilterNAFType" : "NAFRev2",
  "Url" : "https://www.opco.com/infos"
  "ResultStatus" = "OK",
  "OpcoSiren" : "123456789",
  "OpcoName" : "Libellé de l'opco",
  "Idcc" : 123
}
```

Propriétés :

- **Filter** : Reprise du ou des paramètres définis dans la requête
 - **Siret** : Siret ou Siren de la société recherchée. Format chaîne de caractère avec préfixes à 0.
 - **Idcc** : Numéro de convention collective recherché. Format numérique.
- **ResultStatus** : Statut de la recherche. Une des valeurs suivantes :

Code	Signification
OK	Opco trouvé
NOT_FOUND	Aucun résultat associé au filtre

MULTIPLE_OPCO

Problème de cohérence des bases
OPCO : Plusieurs OPCO associé au filtre
donné

- **OpcoSiren** : Siren de l'Opco trouvé. Format chaîne de caractère avec préfixes à 0.
- **OpcoName** : Libellé de l'Opco trouvé. Format chaîne de caractères.
- **Idcc** : Convention collective de la société recherchée. Format numérique.
- **FilterNAFType** : Information sur la codification du champ FilterNAFCode.
- **FilterNAFCode** : Code NAF du Siren ou Siret défini dans Filter. Si la demande porte sur un Siren, alors le code NAF retourné est la valeur *ActivitePrincipaleUniteLegale* de la base SIREN. Si la demande porte sur un Siret, alors le code NAF retourné est la valeur *ActivitePrincipaleEtablissement* de l'Etablissement
- **URL** : URL d'information fournit par l'OPCO trouvée. Cette URL est potentiellement fonction du code NAF retourné

VI. Recherche par lot

Rôle : c'est méthode permet de réaliser une recherche par lot en un seul appel. Cela permet d'éviter des requêtes en boucle (Voir limitations d'appels). Le principe est le même que l'appel unitaire à ceci près que le filtre est défini dans un objet du BODY de la requêtes.

Type de méthode : POST

Route : /api/opcos

Paramètres :

Objet JSON contenant un tableau de filtre :

```
[
  {
    "Siret" = "987654321",
    "IDCC" = 123
  },
  {
    "Siret" = "987654321",
    "IDCC" = 123
  }
]
```

Le tableau contient des objets dont les propriétés sont les suivantes :

- **siret** : [Optionnel] Siret de la société recherchée. La définition d'un Siren est possible par ce même paramètre. Format chaîne de caractère avec préfixes à 0 (Optionnel).
- **idcc** : [Optionnel] Numéro de convention collective recherché. Format numérique

Exemples :

- <https://www.cfadock.fr/api/opcos>
 - POST BODY :

```
[
  {
    "Siret" = "987654321"
  },
  {
    "IDCC" = "987654321"
  }
]
```

Limitations:

- Le nombre d'objets définis dans le filtre est limité à **100**

Réponse :

Le résultat retourné est un tableau retournant des objets du même type que l'appel unitaire :

```

{
  "globalResult": "PARTIAL",
  "found": [
    {
      "filters": {
        "siret": "7080260",
        "idcc": 2216
      },
      "filterNAFType": "68.20B ",
      "filterNAFCode": "NAFRev2",
      "URL": "http://opco.com",
      "searchStatus": "OK",
      "opcoSiren": "7080260",
      "opcoName": "L'Opcommerce",
      "idcc": 2216
    },
    {
      "filters": {
        "siret": "7080039",
        "idcc": 1487
      },
      "filterNAFType": "68.20B ",
      "filterNAFCode": "NAFRev2",
      "URL": "http://opco.com",
      "searchStatus": "OK",
      "opcoSiren": "7080039",
      "opcoName": "L'Opcommerce",
      "idcc": 1487
    }
  ],
  "notFound": [
    {
      "filters": {
        "siret": "7080022",
        "idcc": 2216
      },
      "filterNAFType": null,
      "filterNAFCode": null,
      "URL": null,
      "searchStatus": "NOT_FOUND",
      "opcoSiren": null,
      "opcoName": null,
      "idcc": null
    }
  ]
}

```

Propriétés :

- **Filter** : Reprise du ou des paramètres définis dans la requête
 - **Siret** : Siret ou Siren de la société recherchée. Format chaîne de caractère avec préfixes à 0.
 - **Idcc** : Numéro de convention collective recherché. Format numérique
- **ResultStatus** : Statut de la recherche. Une des valeurs suivantes :

Code	Signification
OK	Tous les opcos ont été trouvés selon les filtres
NOT_FOUND	Aucun opco n'a été trouvé selon les filtres
PARTIAL	Certains opcos ont été trouvés
MULTIPLE_OPCO	Problème de cohérence des base OPCO : Plusieurs OPCO associé au filtre donné
MULTIPLE_SEARCH_LIMIT_REACHED	La limite des 100 objets dans le filtre est dépassée

- **OpcoSiren** : Siren de l'Opco trouvé. Format chaîne de caractère avec préfixes à 0.
- **OpcoName** : Libellé de l'Opco trouvé. Format chaîne de caractères.
- **Idcc** : Convention collective de la société recherchée. Format numérique.
- **FilterNAFType** : Information sur la codification du champ FilterNAFCode.
- **FilterNAFCode** : Code NAF du Siren ou Siret défini dans Filter. Si la demande porte sur un Siren, alors le code NAF retourné est la valeur *ActivitePrincipaleUniteLegale* de la base SIREN. Si la demande porte sur un Siret, alors le code NAF retourné est la valeur *ActivitePrincipaleEtablissement* de l'Etablissement
- **URL** : URL d'information fournit par l'OPCO trouvée. Cette URL est potentiellement fonction du code NAF retourné