

Infobel Presence Management API

Technical Specification Document



Infobel Presence Management API

Table of Contents

1. Document Control	3
1.1. Revision History	3
1.2. Approvals and Reviews	3
1.3. Change Requests	3
2. Contact person	4
3. Access Restriction	4
4. How to use the Infobel Presence Management API	4
4.1. Request a token	4
4.2. Push a record for insert	5
4.3. Push a record for update	6
4.4. Delete a record	6
4.5. Cancel or Roll-Back	7
5. Fields list description and possible values	7
5.1. Fields description	7
5.2. List of available fields	8
5.3. Requirements and specifications	9
6. JSON Response	10
7. Determine the claim status of a record	11
8. Search a record	12
8.1. Search UUID	12
8.2. Search Website	13
9. Merge duplicate record	14
10. Statistics of a record	15

1. Document Control

1.1. Revision History

Revision History			
Date	Version	Description	Author(s)
06.10.2015	1.0	Document creation	Dev. Team
07.30.2015	1.1	Document review and improvement	IT Team
05.19.2016	1.2	Adding API Field	Dev. Team
05.20.2016	1.2.1	Modification in section 5.3 and normalization flag	Dev.Team
05.24.2016	1.2.2	Reviews and improvements	Sales - Marketing
11.23.2016	1.3	Adding Search capability	Dev. Team
11.24.2016	1.3.1	Merge duplicate Statistics of a record Content of a record and claim status	Dev. Team
11.30.2016	1.3.2	Clarification regarding the CategoryLabelLanguageXX	Dev. Team

1.2. Approvals and Reviews

Date	Title
11.30.2016	<i>Chief Technology Officer</i> Marc Whaba
11.30.2016	<i>Senior Web Developer</i> Farid Elyazidi
11.30.2016	<i>System Administrator</i> Michaël Bonnarens

1.3. Change Requests

Our goal at KAPITOL S.A. is to define and use a Software Development Process that helps the organization and people involved, to perform better. As part of KAPITOL S.A. ongoing effort to ensure an improved Software Development Process that meets any needs, we would appreciate any feedback regarding the topic described herein.

Any change request to this document should be submitted to our ticketing system by providing the ID of this document or by email to one of the person in the approvals and reviews list.

2. Contact person

Marc Wahba	CTO	mwahba@infobel.com
Farid Elyazidi	DEV	felyazidi@infobel.com
Michaël Bonnarens	SYS.Admin	mbonnarens@infobel.com

3. Access Restriction

Kapitol Api is restricted based on IP access list, user/password authentication and a token.

First of all, we need you to send us your Public IP Address or IP Range to add it to our system in order to add a level of authentication.

Take contact with our System Administrator for those steps.

Host: api.infobel.com

4. How to use the Infobel Presence Management API

4.1. Request a token

Here is a sample using curl:

```
curl -H "Accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -X POST -d "grant_type=password&username=MyUser&password=MyPassword" "http://api.infobel.com/api/token" -v
```

The response will be your token, keep in mind that the token is only valid for 30 minutes.

Here is an example of a response of a token request, like you can see in the response the expiration is 1799 seconds. In **GREEN** the token

```
{"access_token": "txCwTgvNBcSjHuh00nBUO5-nSILdaokYYjGggI9apKglOPAi2X9NZFP3ZA95T7ORy2En9UISoZRVacDus2rham1xltprbANK51Y4h7fAs35eNZIbguQePMwnpDtbtkM6PccyS620NTRrugSw02nw76e30SkaNCvufUnilMkUVfscvi3biVRtef_sdXJ5HvStyRqwTA16KDECMz-hl_ZUnagVBBnBLijHta078qc45BYQ94h-miCPIKfkd-epx6MREnX-oLkNuVo78wvcmogrOz4_Y5nLyWagQHexDYxpBE8X_a-Xah9UT8T1eDFY_wTysNISgzabK_zrcqBt9_LUIGDcGutzp4PHKGcqhvjOpc", "token_type": "bearer", "expires_in": 1799}* Connection #0 to host api.infobel.com left intact
```

4.2. Push a record for insert

After you successfully receive a token, you can now submit a request to the API; you can find an example below:

HTTP POST using the following URL: <http://api.infobel.com/api/requests/insert>

Here is a sample using curl:

```
curl -H "Accept: application/json" -H "Content-Type: application/json" -H "Authorization: txCwTgvNBcSjHuh00nBUO5-nSILdaokYYjGggI9apKglOPAi2X9NZFP3ZA95T7ORy2En9UISoZRVacDus2rham1xltprbANK51Y4h7fAs35eNZlbgUqEPMwnpDtbtKM6PccyS620NTRrugSw02nw76e30SkaNCvufUnilMkUVfscvi3biVRtef_sdXJ5HvStyRqwTA16KDECMz-hl_ZUnagVBBnBLijHta078qc45BYQ94h-miCPIKfkd-epx6MREnX-oLkNuVo78wvcmogrOz4_Y5nLyWagQHexDYxpBE8X_a-Xah9UT8T1eDFY_wTysNISgzabK_zrcqBt9_LUIGDcGutzp4PHKGcqhvjOpc" -X POST -d '{"CountryCode":"DE","UniversalPublicationId":null,"KapitolRecord":{"CompanyName":"My company name","AddressStreet":"Company street name","AddressHouseNumber":"54","Address1":"Company street name 54","PostCode":"11178","City":"Berlin","Phone":"03012345678","Fax":null,"Mobile":null,"Email":null,"Website":null,"Logo":null,"Description":null,"CategoryLabel01":null,"CategoryLabel02":null,"CategoryLabel03":null,"CategoryLabel04":null,"CategoryLabel05":null,"CategoryLabel06":null,"CategoryLabel07":null,"CategoryLabel08":null,"CategoryLabel09":null,"CategoryLabel10":null,"AlternativePhone":null,"TollFreeNumber":null,"FacebookURL":null,"TwitterURL":null,"LinkedInURL":null,"YouTubeURL":null,"Path1":null,"Path2":null,"Path3":null,"Path4":null,"Path5":null,"Path6":null,"Path7":null,"Path8":null,"Path9":null,"Path10":null,"Keywords":null,"YearFounded":null,"OpeningHours":null,"OpeningHoursNotes":null,"OpenHoursMondayOpen":null,"OpenHoursMondayClose":null,"OpenHoursTuesdayOpen":null,"OpenHoursTuesdayClose":null,"OpenHoursWednesdayOpen":null,"OpenHoursWednesdayClose":null,"OpenHoursThursdayOpen":null,"OpenHoursThursdayClose":null,"OpenHoursFridayOpen":null,"OpenHoursFridayClose":null,"OpenHoursSaturdayOpen":null,"OpenHoursSaturdayClose":null,"PaymentTypes":null,"Products":null,"Services":null,"Brands":null,"Associations":null,"BulkUniqueId":null,"BusFlag":false,"TradeName":null,"Address2":null,"Province":null,"Country":null,"Language":null,"NationalID":null,"Longitude":null,"Latitude":null,"GeoLevel":null,"GeoConfidence":null,"InfobelCode01":"11421","InfobelCode02":null,"InfobelCode03":null,"InfobelCode04":null,"InfobelCode05":null,"InfobelCode06":null,"InfobelCode07":null,"InfobelCode08":null,"InfobelCode09":null,"InfobelCode10":null,"LocalCodeType":null,"LocalCode01":null,"LocalCode02":null,"LocalCode03":null,"LocalCode04":null,"LocalCode05":null,"LocalCode06":null,"LocalCode07":null,"LocalCode08":null,"LocalCode09":null,"LocalCode10":null,"LocalCode11":null,"LocalCode12":null,"LocalCode13":null,"LocalCode14":null,"LocalCode15":null,"InternationalCodeType":null,"InternationalCode01":null,"InternationalCode02":null,"InternationalCode03":null,"InternationalCode04":null,"InternationalCode05":null,"InternationalCode06":null},"Details":{"PackageType":1}}' "http://api.infobel.com/api/requests/insert" -v
```

Infobel Presence Management API

Note: Depending the shell you are using, it's possible that all double quote (") need to be doubled or must be escaped using a backslash (\) in your Json.

An alternative to avoid specific shell interpretation, you can put your request in a file and specify this file in the data.

For example:

```
curl -H "Accept: application/json" -H "Content-Type: application/json" -H "Authorization: Bearer txCwTgvNBcSjHuh00nBUO5-nSILdaokYYjGggl9apKglIOPAi2X9NZFP3ZA95T7ORy2En9UISoZRVacDus2rham1xltprbANK51Y4h7fAs35eNZlbgUQePMwnpDtbtKM6PccyS620NTRrugSw02nw76e30SkaNCvufUnilMkUVfscvi3biVRtef_sdXJ5HvStyRqwTA16KDECMz-hl_ZUnagVBBnBLijHta078qc45BYQ94h-miCPIKFkd-epx6MREnX-oLkNuVo78wvcmogrOz4_Y5nLyWagQHexDYxpBE8X_a-Xah9UT8T1eDFY_wTysNISgzabK_zrcqBt9_LUIGDcGutzp4PHKGcqhvjOpc" -X POST -d @request.txt "http://api.infobel.com/api/requests/insert" -v
```

4.3. Push a record for update

HTTP POST using the following URL: <http://api.infobel.com/api/requests/update>

The procedure is the same as an insert; the only difference is the "UniversalPublicationId" field. It must be filled in to identify the record to update.

4.4. Delete a record

HTTP POST using the following URL: <http://api.infobel.com/api/requests/delete>

Some parameters must be present in the Details part to remove a record or a duplicate:

Beside the Countrycode and the UNIVERSALPUBLICATIONID, either the DeleteReason or DeleteDuplicate must be filled in in the "Details" part.

For DeleteReason the value is a string (for instance, "business closed", "not existing anymore")

For DeleteDuplicate the value is the UUID of the duplicate (for instance "DeleteDuplicate": "DE999999999")

Sample:

```
{"CountryCode": "DE", "UniversalPublicationId": "DE888888888", "Details": {"DeleteDuplicate": "DE999999999"}}
```



The delete action removes a record from our database; it is not a way to roll back a change you made on a record! See the point **4.5. Cancel a request** for more information.

4.5. Cancel or Roll-Back

HTTP GET using the following URL: <http://api.infobel.com/api/requests/<requestId>/cancel>

Where **<requestId>** is the RequestID (integer), you get back in the response when you do an insert/update/Delete.


Every action that is done on a record is stored in a history table so you can cancel any step of your actions on a record with the appropriate RequestID.

e.g.: we have this situation for a record:

- Record: initial state
- Push1: Update1
- Push2: Update2
- Push3: Update3

If you want to cancel all your pushes and roll back to the initial state you need to send the **<requestId>** of Push1.

Any Step can be canceled directly as for example **<requestId>** of Push2 to roll back to Update1.

 A request can be cancelled for 7 days after its creation, only if no newer request was created for the same record.

5. Fields list description and possible values

5.1. Fields description

Field types: all fields are represented as strings, except for all the Flag fields:

BusFlag - PublicationFinancialInfoFlag - PublicationFinancialInfoBlockedFlag -
PublicationWebInfoBlockedFlag - PublicationWebInfoFlag - DisplayStandardizedAddressFlag -
DisplayStandardizedGeoInfoFlag

Field lengths: all fields have a max length of 500 - except those ones:

Video - Services - Products - PaymentTypes - Path1 ... Path10 - OpeningHoursNotes - OpeningHours
Logo - Keywords - Description - Brands - Baseline - Associations - Address2 - Address1

InfobelCode with custom label:

The combination of the values **<InfobelCodeXX,CategoryLabelLanguageXX>** must be unique.

5.2. List of available fields

AdditionalMobile01	DisplayStandardizedAddressFlag	OpenHoursFridayClose
AdditionalMobile02	DisplayStandardizedGeoInfoFlag	OpenHoursFridayOpen
AdditionalMobile03	Email	OpenHoursMondayClose
AdditionalMobileDescription01	FacebookURL	OpenHoursMondayOpen
AdditionalMobileDescription02	Fax	OpenHoursSaturdayClose
AdditionalMobileDescription03	GeoConfidence	OpenHoursSaturdayOpen
AdditionalPhone01	GeoLevel	OpenHoursSundayClose
AdditionalPhone02	InfobelCode01	OpenHoursSundayOpen
AdditionalPhone03	InfobelCode02	OpenHoursThursdayClose
AdditionalPhoneDescription01	InfobelCode03	OpenHoursThursdayOpen
AdditionalPhoneDescription02	InfobelCode04	OpenHoursTuesdayClose
AdditionalPhoneDescription03	InfobelCode05	OpenHoursTuesdayOpen
Address1	InfobelCode06	OpenHoursWednesdayClose
Address2	InfobelCode07	OpenHoursWednesdayOpen
AddressBoxNumber	InfobelCode08	OpeningHours
AddressHouseNumber	InfobelCode09	OpeningHoursNotes
AddressStreet	InfobelCode10	Path1
AlternativePhone	InternationalCode01	Path10
Associations	InternationalCode02	Path2
Baseline	InternationalCode03	Path3
Brands	InternationalCode04	Path4
BulkUniqueld	InternationalCode05	Path5
BusFlag	InternationalCode06	Path6
CategoryLabel01	InternationalCodeType	Path7
CategoryLabel02	Keywords	Path8
CategoryLabel03	Language	Path9
CategoryLabel04	Latitude	PaymentTypes
CategoryLabel05	LinkedInURL	Phone
CategoryLabel06	LocalCode01	PostCode
CategoryLabel07	LocalCode02	Products
CategoryLabel08	LocalCode03	Province
CategoryLabel09	LocalCode04	PublicationFinancialInfoBlockedFlag
CategoryLabel10	LocalCode05	PublicationFinancialInfoFlag
CategoryLabelLanguage01	LocalCode06	PublicationWebInfoBlockedFlag
CategoryLabelLanguage02	LocalCode07	PublicationWebInfoFlag
CategoryLabelLanguage03	LocalCode08	Services
CategoryLabelLanguage04	LocalCode09	SourceLatitude
CategoryLabelLanguage05	LocalCode10	SourceLongitude
CategoryLabelLanguage06	LocalCode11	TollFreeNumber
CategoryLabelLanguage07	LocalCode12	TradeName
CategoryLabelLanguage08	LocalCode13	TwitterURL
CategoryLabelLanguage09	LocalCode14	UserID
CategoryLabelLanguage10	LocalCode15	Video
City	LocalCodeType	Website
CompanyName	Logo	YearFounded
Country	Longitude	YouTubeURL
Description	Mobile	
DirectoryName	NationalID	

5.3. Requirements and specifications

For each requirement - at least one field must be provided:

Name: CompanyName - TradeName – DirectoryName
Contact: Mobile - Phone - Fax - Website – Email
Address: Address1 – AddressStreet
Location: PostCode – City
Category: InfobelCode01 ... InfobelCode10

Concerning the fields, you can fill in in Json format into the KapitoolApi

Enhanced descriptive fields:

Description, OpeningHours, OpeningHoursNotes, Products, Services, Brands, Keywords, Baseline

Those fields are string and they accept HTML format or simple text: the Infobel site detects which format is used and will display it accordingly.

Categories

CategoryLabel01, CategoryLabel02, CategoryLabel03, CategoryLabel04, CategoryLabel05,
CategoryLabel06, CategoryLabel07, CategoryLabel08, CategoryLabel09, CategoryLabel10

You can fill in the code with Infobel code (InfobelCodeXX) and the site will display the default Infobel category Label in the appropriate language available.

If for some reason, the Infobel category Label does not suit your needs, you can fill in the CategoryLabelXX along the CategoryLabelLanguageXX with you own custom label and the appropriate language, your label will be displayed instead of the default Infobel one only when the Infobel website is displayed in that language.

e.g.:

- "InfobelCode01":"016118" = site displays "Publishing Of Telephone Directories"
- "InfobelCode01":"016118" and "CategoryLabel01":"Online Telephone Directories" and "CategoryLabelLanguage01": "EN" = site displays the content of CategoryLabel01 when the website is in English, for other languages it will be the default label.



Pay Attention to the string format of InfobelCodeXX: Most Infobel codes starts with a zero! A numeric variable will truncate it and the category won't be displayed.

PaymentType

Here is the list of supported values for the field PaymentTypes

Mastercard, AmericanExpress, Cirrus, Maestro, VisaElectron, Paypal, Visa, Cash, Check

Infobel Presence Management API

PackageType

Here are the possible values:

- 0 = regular -> default record coming from the telecom provider
- 1 = basic -> enhanced record (free of charge)
- 200 = TopLocal-> enhanced record with advertising cover (**paid service**)
- 1000 = TopRegion -> enhanced record with advertising cover (**paid service**)

e.g.: "Details":{"PackageType":1}

If you need more info about the packages and what they contain, please visit this page:

<http://www.infobel.com/en/belgium/Packages/Pricing>

Note: For partners only, all fields listed in the link above are accessible, and if they are filled they will be displayed in the Infobel site.

6. JSON Response

When you push a record in the kapitolApi, a status (integer format) is returned in the response. This status can help you understand what is going on. You can find below a list of available status

- 1 = Pending
- 2 = Processed (This status means the record is successfully published on the website)
- 3 = Approved
- 4 = Rejected
- 5 = Error
- 6 = AutoApproved
- 7 = CancelPending
- 8 = Cancelled

For Insert, when you submit a request, you get in return an urlGenerated which is the link of the detail page of the record published on the Infobel site.

The urlGenerated is in the following format:

http://www.infobel.com/fr/belgium/infobel_kapitol/ucle/BE10005416-023792940/businessdetails.aspx

It contains the **name**, the **city**, the universalPublicationId and the **phone** number.

<http://www.infobel.com/fr/belgium/NAME/CITY/universalPublicationId-PHONE/businessdetails.aspx>

Record normalization:

The record sent via the Infobel Presence Management API is subject to a normalization concerning the street name and the city name fields.

If this is the case, the normalization procedure could change those fields with a more accurate and/or precise denomination.

This procedure is automatic and can be noticed with the fields with prefix "Std" (like for instance: "StdStreetName") which will be different from the standard input fields (like "AddressStreet" which is the field matching our sample).

The normalization can be avoided by setting the following flags to false:

- DisplayStandardizedAddressFlag : for the address/city fields
- DisplayStandardizedGeoInfoFlag : for the latitude/longitude fields

7. Determine the claim status of a record

This method will help you to determine if a record is already claimed or free.

Claim status:

- Taken = Claimed by a different account than yours
- Owned = Claimed by your account
- Available = Free, not claimed

HTTP GET using the following URL: <http://api.infobel.com/api/record/<uuid>>

Will report the claim status and details of the record

```
{
  "status": "Taken",
  "record": {
    "companyName": "Infobel / Kapitol",
    "tradeName": "",
    "directoryName": "",
    "address1": "Chaussée de Saint-Job 506",
    "address2": "",
    "addressStreet": "Chaussée de Saint-Job",
    "addressHouseNumber": "506",
    "addressBoxNumber": "",
    "postCode": "1180",
    "city": "Uccle",
    "country": "Belgium",
    "language": "FRE",
    "phone": "023792940",
    "fax": "023792988",
    "mobile": "",
    "website": "http://www.infobel.be",
    "email": "",
    "nationalID": "0453604761",
    ...
  }
}
```

Infobel Presence Management API

HTTP GET using the following URL: <http://api.infobel.com/api/record/claimable/<uuid>>

Will only report the claim status

```
{
  "status": "Taken"
}
```

8. Search a record

8.1. Search UUID

Give you the ability to make a match between a submitted record and an existing one.

POST using the following URL: <http://api.infobel.com/api/search/match>

You need to provide as much information as possible in order to have a potential match

Sample input:

```
"CountryCode": "be",
"KapitolRecord": {
  "Address1": "Chaussée de saint-job 506",
  "City": "Uccle",
  "CompanyName": "Infobel Kapitol"
  "Phone": "023792940",
  "PostCode": "1180"
}
```

If the provided record matched and existing one, it will return the existing record and its claim status.

Sample output:

```
"status": "Taken",
"record": {
  "address1": "Chaussée de Saint-Job 506",
  "address2": "",
  "addressHouseNumber": "506",
  "addressStreet": "Chaussée de Saint-Job",
  "city": "Uccle",
  "companyName": "Infobel Kapitol",
  "country": "Belgium",
  "email": "",
  "facebookURL": "",
  "fax": "023792988",
  "geoConfidence": "",
  "geoLevel": "3",
  "infobelCode01": "016118",
  "language": "FRE",
  "lastUpdate": "Sep 26 2016 11:55AM",
  "latitude": "50.7920000",
  "logo": "thumb_logo infobel 4bis fr.jpg",
  "postCode": "1180",
  "stdCityName": "Uccle",
  "stdStreetName": "Chaussée de Saint-Job",
  "stdStreetPostCode": "1180",
  "uniqueID": "014018041",
  "universalPublicationId": "BE100005416",
  ...
}
```

8.2. Search Website

Give you the ability to make search, as you would do on the Infobel website without the limitation present on the website.

POST using the following URL: <http://api.infobel.com/api/search/website>

Input fields are:

- **CountryCode:** You must input there the country code
- **Who:** You must input there a Name/Phone/Category Label or a NationalID
- **Where:** You must input there a City/PostCode
- **Page:** You can input there which result page you want (*Optional*) **Default: 1**
- **PageSize:** You can define how many records you want per page (*Optional*) **Default: 10**

Sample input:


```
"CountryCode": "be",
"Who": "023792940",
"Where": "brussels",
"Page": 1,
"PageSize": 10
```

Sample output:

```
"page": 1,
"total": 1,
"records": [
  {
    "universalPublicationId": "BE100005416",
    "busFlag": "",
    "companyName": "Infobel \\/ Kapitol",
    "tradeName": "",
    "directoryName": "",
    "namingMethod": "",
    "address1": "Chaussée de Saint-Job 506",
    "address2": "",
    "addressStreet": "Chaussée de Saint-Job",
    "addressHouseNumber": "506",
    "postCode": "1180",
    "city": "Uccle",
    ...
  },
  ...
],
"count": 1
```

9. Merge duplicate record

Gives you the ability to merge two records.

 In order to be able to do that, both records must have been claimed by your account.

HTTP GET using the following URL: <http://api.infobel.com/api/record/<uuid1>/merge/<uuid2>>

Indicate that record with uuid2 is a duplicate of record with uuid1, and that only the record with uuid1 must remain

Sample:

<http://api.infobel.com/api/record/BE100005416/merge/BE100005417>

The output will be in boolean and will indicate the merge operation result.

```
{
  "merged": true,
  "message": "Record with uuid BE100005417 will be merged to record with uuid
  BE100005416, Only record with uuid BE100005416 remains"
}
```



10. Statistics of a record

This method can be used in order to obtain the statistics of a record for the last 3 months.

Available statistics:

- *Potential Views*: Number of potential view a record can have if it was a payed record.
- *Total Views*: Number of times the record was displayed (Details + Search page)
- *Details Page Views*: Number of time the details page was displayed
- *Search Results Views*: Number of time this business appeared in a search result.
- *Website URL Click Count*: Number of click on the customer website link (Details + Search page)
- *Email Sent Count*: Number of time an email has been sent to this business

HTTP GET using the following URL: <http://api.infobel.com/api/record/stats/<uuid>>

```
"stats": [  
  {  
    "month": "2016-10",  
    "potentialViews": 33440,  
    "totalViews": 43739,  
    "detailsPageViews": 10299,  
    "searchResultsViews": 33440,  
    "websiteUrlClickCount": 315,  
    "emailSentCount": 0  
  },  
  {  
    "month": "2016-09",  
    "potentialViews": 18517,  
    "totalViews": 26919,  
    "detailsPageViews": 9576,  
    "searchResultsViews": 17343,  
    "websiteUrlClickCount": 205,  
    "emailSentCount": 80  
  },  
  {  
    "month": "2016-08",  
    "potentialViews": 7034,  
    "totalViews": 14533,  
    "detailsPageViews": 10011,  
    "searchResultsViews": 4522,  
    "websiteUrlClickCount": 206,  
    "emailSentCount": 92  
  }  
]
```

