



Data sharing through DjustConnect DATA CONSUMER

To enable secure data sharing, with consent of the farmer, some additional steps are needed to use data from a REST API via DjustConnect:

- Request Access to APIs in DjustConnect
- Client SSL certificate, this may be self-signed
- Additional HTTP header: DjustConnect-Subscription-Key

This document provides more information on the steps required and some additional functionalities offered by DjustConnect.

Contents

1	Configuring the application in the DjustConnect Portal	2
1.1	Partner details.....	2
1.2	My access (Access management).....	3
1.3	Farm access.....	4
1.4	Resource access.....	4
1.5	Farm/Resource access.....	5
1.6	Resource health	5
2	Integration with the DjustConnect partner API	5
2.1	ConsumerAccess	6
2.2	FarmStatus, RarStatus en DarStatus (optional).....	6
2.3	Consumer/resource-health (optional).....	6
2.4	FarmMapping (optional).....	6
3	Using data	7



1 Configuring the application in the DjustConnect Portal

1.1 Partner details

Here you can find the details of the selected partner.



Name
Djustconnect Consumer samples



Purpose
Consumer web-application in ASP.NET Core for the api's in the Djustconnect Provider Sample


Application Url
https://samples.djustconnect.be/consumerssharp

Naam data aanvrager

Url Application logo

Primary Key
.....  

Secondary Key
.....  

Certificate
Current certificate thumbprint: 323856CD8AB9E1890B9D654EA609B38849A54CA7 

Information to be provided for the *Partner details* tab, this information is configured by a DjustConnect administrator:

- Application name
- Purpose of the application
- Application URL: an url to the application (optional), this is used to redirect to your application from the data request overview of the farm

Information to fill in yourself for the *Partner detail* tab:

- Certificate: Public key of the SSL certificate (.cer file) that will be used to call APIs. This certificate can also be generated and subsequently downloaded.

On the *Partner details* tab, you will also find the *primary subscription key* that must be provided with each API call as an HTTP header with key: *DjustConnect-Subscription-Key*.

1.2 My access (Access management)

Below you can configure the resource and farm access you require.

Add Access Request

Here you can request data to different resources and farms for your application. For most data, you'll need approval from the provider owning the data resource and the farmer owning the data itself.

You can manage the resources you're requesting access to separately from the Farms you're requesting data from. This way, you can easily add a new farm or consumer to your data access, or, add a new resource if your application has new functionality. Your data access always must have an end date.

The end date for your access:

12/30/2030

Resource access (will be approved by the owning provider):

Get a count of cattle in- or outside for specified kbo - Provider CSharp × Get an overview of total count of cattle grouped per farm - Provider CSharp ×
Get Milk Invoices - Milk And Farms × ApiDjustconnectv2ByKboFieldactivitiesGet - AVR API ×
ApiDjustconnectv2ByKboByFieldActivityIdYieldimageGet - AVR API × ApiDjustconnectv2ByKboByFieldActivityIdYielddataGet - AVR API ×

Farm Data Access (will be approved by the owning farmer):

KBO

Search

Add farm numbers

Clear all

0073570013 × 0073570015 × 0073570017 × 0073570019 × 0073570021 ×
0262172489 × 0728406751 ×

Submit Request

In the *My access* tab, the access request can be configured. It consists of:

- End date (optional): applicable when the application itself has an end date;
- Access to data sources: choose here which APIs you wish to use;
- Farm data access: choose here which type of identification number (CBE number (*KBO*), production unit number (*PE nummer*), herd number (*Beslagnummer*), ...) you wish to use. In the test phase you can manually enter some ID numbers here, but in production we recommend using the Partner API provided for this purpose: *ConsumerAcces*.

1.3 Farm access

Below are the statuses of all the farms for which you have created access requests.

Nummer ↑ Filter Nummer	Status -- No. ▾
0073570013	Has user
0073570015	Has user
0073570017	Has user
0073570019	Has no user
0073570021	Has no user
0262172489	Has user
0728406751	Has user

Items per page: 10 1 - 7 of 7 << < > >>

In the *Farm access* tab, the status of the farms for which access requests have been sent can be consulted. This gives an indication for which farms a user is already registered.

1.4 Resource access

Below are all the statuses of all your resource access requests.

Resource ↑ Filter Resource	Api Filter Api	Provider Filter Provider	Resource Status -- None -- ▾
ApiDjustconnectv2ByKboByField ActivityIdYielddataGet	AVR API	AVR	Approved
ApiDjustconnectv2ByKboByField ActivityIdYieldimageGet	AVR API	AVR	Approved
ApiDjustconnectv2ByKboFieldacti vitiesGet	AVR API	AVR	Approved
Get a count of cattle in- or outside for specified kbo	Provider CSharp	Djustconnect Provider Samples	Approved
Get an overview of total count of c attle grouped per farm	Provider CSharp	Djustconnect Provider Samples	Approved
Get Milk Invoices	Milk And Farms	Milcobel	Approved

Items per page: 10 1 - 6 of 6 << < > >>

In the *Resource access* tab, you can check whether permission has been obtained from the various data suppliers for all requested API's.

1.5 Farm/Resource access

Farm 02621	Resource ↑ Filter Resource	Farm Status -- None -- ▾	Resource Status -- None -- ▾	DAR Status -- None -- ▾
0262172489	ApiDjustconnectv2ByKboByFieldActivityIdYielddataGet	Has user	Approved	No Data
0262172489	ApiDjustconnectv2ByKboByFieldActivityIdYieldimageGet	Has user	Approved	No Data
0262172489	ApiDjustconnectv2ByKboFieldactivitiesGet	Has user	Approved	No Data
0262172489	Get a count of cattle in- or outside for specified kbo	Has user	Approved	Approved
0262172489	Get an overview of total count of cattle grouped per farm	Has user	Approved	Rejected
0262172489	Get Milk Invoices	Has user	Approved	Approved
0262172489	Lijst van bodemanalyses	Has user	Approved	Pending
0262172489	Rapport bodemanalyse	Has user	Approved	Pending

Items per page: 10 1 - 8 of 8 < < > > |

In the *Farm/Resource access* tab, you can check whether a farm has given consent to retrieve and use its data from a data source.

1.6 Resource health

Below you find the current status of the resources you have access to.

Resource	Health
Get a count of cattle in- or outside for specified kbo	Healthy
Get an overview of total count of cattle grouped per farm	Healthy
Get Milk Invoices	Healthy
ApiDjustconnectv2ByKboFieldactivitiesGet	Healthy
ApiDjustconnectv2ByKboByFieldActivityIdYieldimageGet	Healthy
ApiDjustconnectv2ByKboByFieldActivityIdYielddataGet	Healthy

In the *Resource health* tab, you can check whether an API is reachable (healthy) or not (unhealthy).

2 Integration with the DjustConnect partner API

To facilitate integration with other systems, several API endpoints are provided in DjustConnect. The Swagger file of this REST API can be accessed at

<https://partnerapi.djustconnect.be/index.html>.

2.1 ConsumerAccess

The configuration of the access request in the *My access* tab can also be done via the *ConsumerAccess* endpoint.

If you want to configure the farms you want to request access for, we recommend first making a GET request on the *ConsumerAccess* endpoint, then adjusting the *farmIds* property of the JSON object based on the list of farms connected to your application and returning the full object with a POST request on the same endpoint. Alternatively, numbers can be added with the *ConsumerAccess/AddFarmId* and removed with the *ConsumerAccess/RemoveFarmId* methods.

The full access request can also be created using the *Resource* and *FarmIdType* endpoints.

2.2 FarmStatus, RarStatus en DarStatus (optional)

The information available on the *My access*, *Farm access* and *Farm/Resource access* tabs can also be checked via the *FarmStatus*, *RarStatus* and *DarStatus* API endpoints if you wish. The *DarStatus*, for instance, can be practical to check if a farmer has given consent and if not, to provide them with information on how to do so in your application.

2.3 Consumer/resource-health (optional)

Information from the *Resource health* tab can also be checked via the *Consumer/resource-health* endpoint.

2.4 FarmMapping (optional)

If one of the APIs you wish to use uses a farm identifier that is unknown in your own system, the translation for this can be requested via the *FarmMapping* endpoint. For example, a CBE number can be translated into a PE or herd number. Note that in some cases multiple values are returned.

3 Using data

Once the application and access request have been configured and approved in DjustConnect, the actual data can be retrieved from the data provider.

All technical information on the various APIs of the data provider can be found at <https://developer.djustconnect.be/>.

For example, looking at the API endpoint [SmartSoil samples](#), we see the following information:

- Url: `https://apim.djustconnect.be/smartsoil-samples/samples?kbo_number={kbo_number}&epsg_code={epsg_code}&wkt={wkt}&interval_start={interval_start}&interval_end={interval_end}`
- Parameter:
 - o *kbo_number* of type string
 - o *epsg_code* of type number
 - o *wkt* of type string
 - o *interval_start* of type string (yyyy-mm-dd)
 - o *interval_end* of type string (yyyy-mm-dd)
- Response:
 - o Samples object consisting of an array of SoilSample

To use this API, we need to make a HTTP GET request to this url with a CBE number by using the SSL certificate and an additional HTTP header with *DjustConnect-Subscription-Key* as the key and the Primary key as the value. If permission is granted, this API will then return a JSON object consisting of a Sample object with an array of SoilSample.

3.1 Creating a self-signed certificate

Steps for creating a self-signed certificate in a Windows environment where the data consumer is a web application running under IIS on the domain my.application.be:

- 1) In an admin powershell on the server:
`$cert = New-SelfSignedCertificate -certstorelocation cert:\localmachine\my -dnsname my.application.be`
- 2) In the Windows certificate store:
 - a. Export the certificate in the Windows certificate store once with the private key (pfx) and once without the private key (cer)
 - b. Right click the certificate -> Tasks -> Manage Private Keys -> Grant rights to the name of the IIS application pool user e.g. "IIS AppPool\my.application.be"
 - c. Copy the certificate to the "Trusted Root Certification Authorities"
- 3) Upload the exported cer file in portal.djustconnect.be in the *Partner details* tab
- 4) To use the same certificate during e.g. development, you can import the certificate into the "Local machine certificate store"

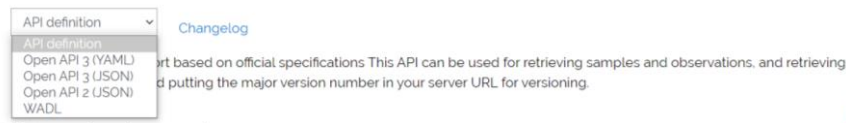
3.2 Fetching data

Example C# code to initialise an HttpClient object with the client SSL certificate and the *DjustConnect-Subscription-Key* header:

```
var store = new X509Store("My", StoreLocation.LocalMachine);
store.Open(OpenFlags.ReadOnly);
var certificates = store.Certificates.Find(X509FindType.FindByThumbprint, "HIER DE THUMBPRINT VAN UW SSL CERTIFICAAT", false);
if (certificates.Count == 0)
{
    throw new InvalidOperationException($"Certificate not found for CN=my.application.be in LocalMachine/My.");
}
var certificate = certificates[0];
var clientHandler = new HttpClientHandler();
clientHandler.ClientCertificates.Add(certificate);
clientHandler.ClientCertificateOptions = ClientCertificateOption.Manual;
var client = new HttpClient(clientHandler);
client.DefaultRequestHeaders.Add("DjustConnect-Subscription-Key", "HIER DE PRIMAIRE SLEUTEL VANUIT tabblad Partner details");
```

Once configured, the client object can be used to call the API endpoint. The code for this can be written manually or generated based on the specifications available for download at <https://developer.djustconnect.be>.

SmartSoil Samples



More information on generating client code can be found here:

- <https://swagger.io/tools/swagger-codegen/>
- <https://stackoverflow.com/questions/54094688/generate-net-client-from-swagger>
- <https://medium.com/@unchase/how-to-generate-c-or-typescript-client-code-for-openapi-swagger-specification-d882d59e3b77>