

# Data Stream Manager

## – User Manual –

# Content

Introduction to the Data Stream Manager .....	3
Opening the Data Stream Manager .....	3
PC and Browser Requirements.....	4
Welcome to the Data Stream Manager.....	6
Governance .....	7
User Administration .....	7
Privacy Settings .....	8
Audit Log .....	8
Connections .....	9
Create a Source Connection .....	9
Create a Destination Connection .....	18
Connections overview .....	28
Create a Template.....	30
Templates.....	32
Streams .....	33
Streams overview .....	33
Create and configure a new data stream.....	38
Mapping the stream variables.....	44
Confirm and publish the data stream.....	47
Datstreams.io support .....	48
Appendix 1: Data capturing methods .....	51
Parameters capturing methods .....	51
Event capturing method details.....	52
Appendix 2: Table configuration examples .....	53
Appendix 3: Default data models.....	56
Appendix 4: User roles and rights.....	61



# Introduction to the Data Stream Manager

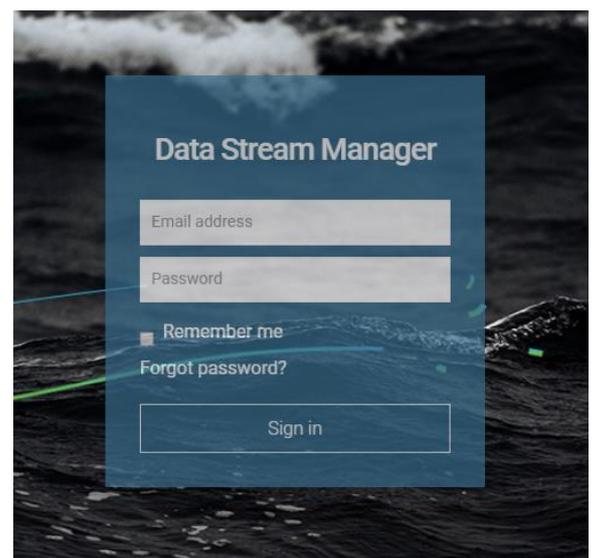
The 'Data Stream Manager' (DSM) empowers organizations of all sectors and sizes to quickly and easily get to grips with digital customer journeys and leverage digital data for a 360-degree customer view.

Collect, fuse and transform any type of data and distribute it in the right context to any or many end points. In short, once you have defined what data needs to be collected, you turn on the data stream and send the data in the file format and delivery method required. For example, this could mean you create CSV files on an FTP server, or stream data perfectly in real time or batch. The DSM is designed to put you in perfect control of the data you need and can be tailored to meet your specific requirements and use cases.

The Data Stream Manager is a single platform to collect data from any digital source and then stream this to any destination. It streams the data in-memory only, therefore it will not store data in any database or file itself in our infrastructure. The data that is sent to the end destination can be used by you for storage or immediate use in real-time processes. The Data Stream Manager platform is cloud based and can optionally be implemented on premise.

## Opening the Data Stream Manager

The Data Stream Manager can be accessed by visiting the dedicated URL on the Datastreams.io domain. The browser will show a screen asking you to provide login credentials. In case you do not have access to the dedicated URL or do not have login credentials please contact your administrator.



# PC and Browser Requirements

The Data Stream Manager is designed to be operated with Google Chrome version 23 and higher. This is because of the DSM 'Live configurator' feature, that is available specifically for website data collection. This configurator uses a Google Chrome extension that must be installed. To download and install this extension follow the below steps:

**STEP1:** Download the following plugin in a folder that won't change its location in the future. If location changes the plugin will not work anymore.

Download here: <https://www.datastreams.io/activate-chrome-plugin/>

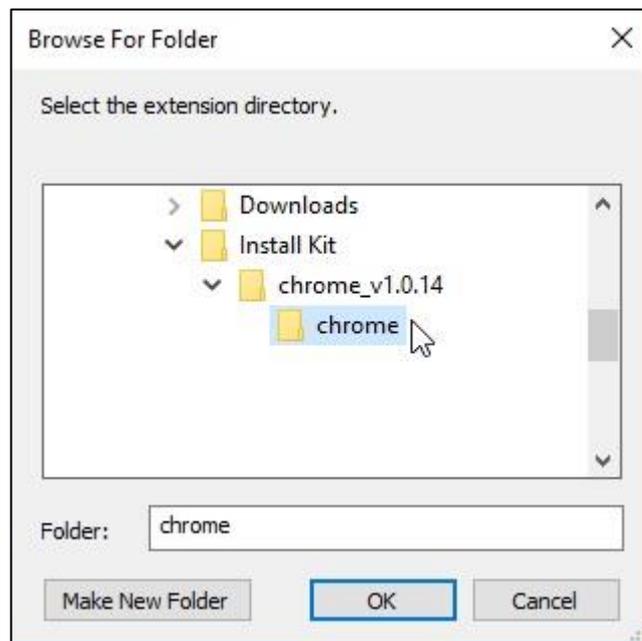
**STEP2:** Unpack the zip file.

**STEP3:** Type or copy to the Chrome browser the following URL: `chrome://extensions/`

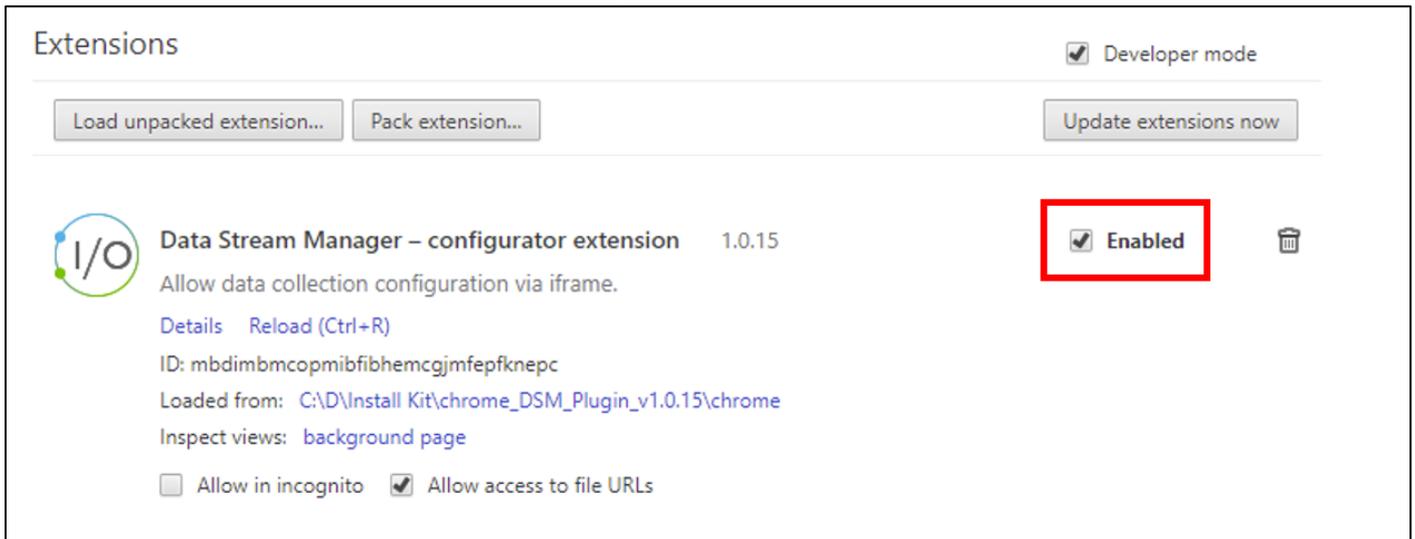
**STEP4:** Click on the Developer mode checkbox on top of the page:



**STEP5:** After having the developer mode active you can load the unpacked extension. Point the explorer window that opens towards the directory where you have the zip file unpacked.



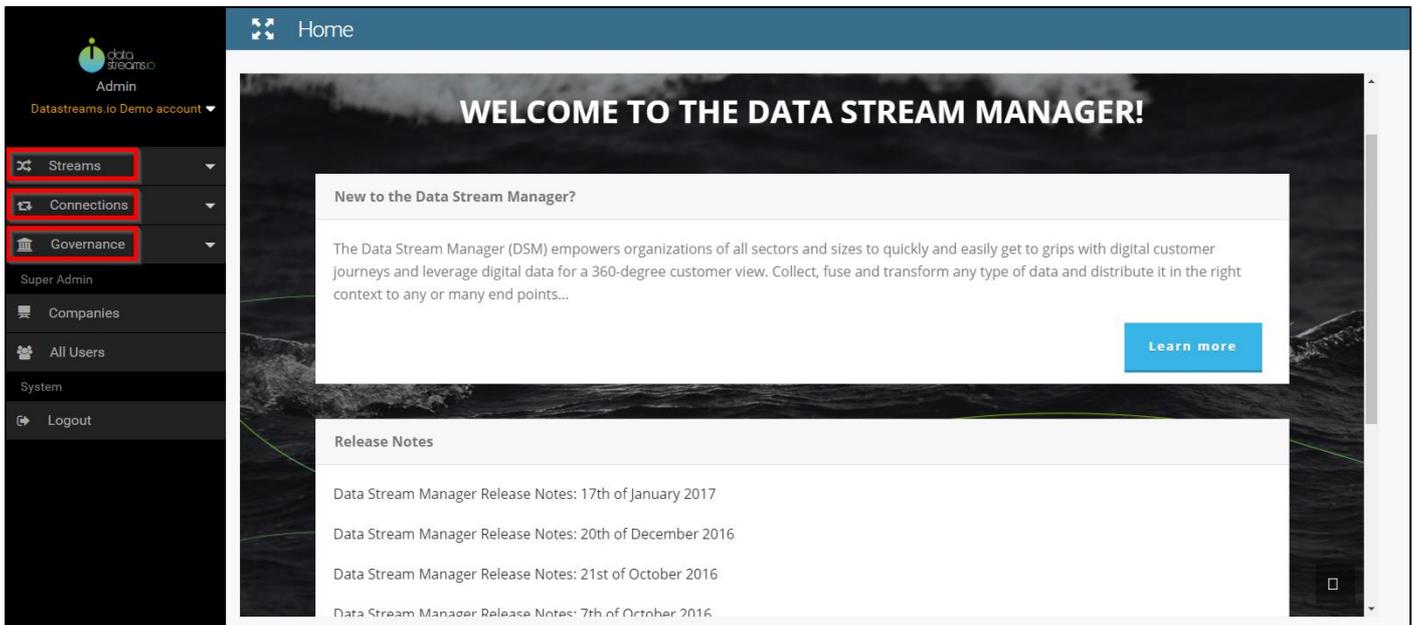
**STEP 6:** Make sure that the extension is enabled:



The screenshot shows the Chrome 'Extensions' page. At the top right, there is a checkbox for 'Developer mode' which is checked. Below this are three buttons: 'Load unpacked extension...', 'Pack extension...', and 'Update extensions now'. The main list of extensions contains one entry: 'Data Stream Manager – configurator extension' version 1.0.15. The extension's icon is a circle with 'I/O' inside. The description reads 'Allow data collection configuration via iframe.' Below the description are links for 'Details' and 'Reload (Ctrl+R)'. The extension ID is 'mbdimbcmcopmibfibhemcgjmfepfknepc'. The source is listed as 'Loaded from: C:\D\Install Kit\chrome\_DSM\_Plugin\_v1.0.15\chrome'. There are links for 'Inspect views: background page'. At the bottom of the extension card, there are two checkboxes: 'Allow in incognito' (unchecked) and 'Allow access to file URLs' (checked). The 'Enabled' status is highlighted with a red box.

Having the PC and Browser Requirements into place, you can now proceed to open the Data Stream Manager.

# Welcome to the Data Stream Manager



The main elements of the Data Stream Manager are:

## Streams

These are the actual streams of data that define when (events: page views, clicks, etc.) and what (parameters: page name, product name, etc.) you collect from a Source Connection and to which Destination connection you want to stream your data.

Each stream has a data model composed of parameters and events. While the **parameters** define what information should be collected in a stream, **events** define when they should be collected.

## Connections

These are the input and output connections that interact with a stream of data. There are two types of connections:

- Source: Defines the location you want to collect your real time streaming data from, e.g. website, social media, etc.
- Destination: Defines the location where you want to stream your data to, e.g. databases, FTP, etc.

## Governance

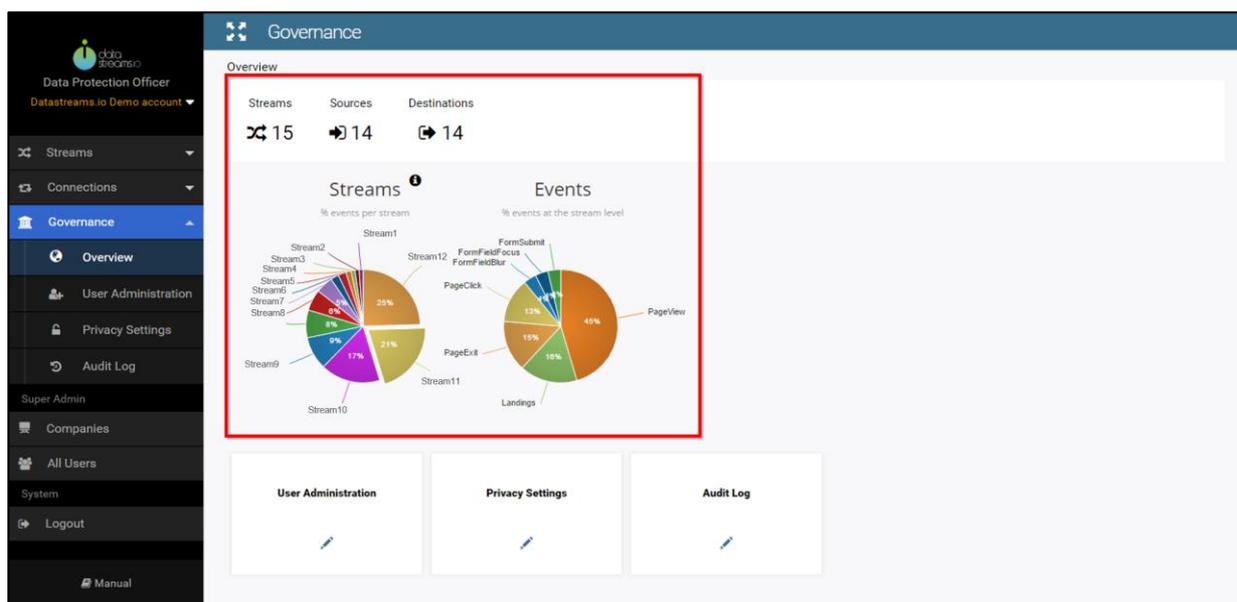
By using DSM, you are fully in control of all your data management requirements in order for you to offer your customers optimal data security and data transparency. This Governance section is only available to a Data Officer, where he/she can manage the user roles, define and set up the privacy levels for different data parameters and destination connections, and review the logs for user activities.

# Governance

This chapter is addressed the Data Officer in your company.

As a Data Officer, in this section you can manage the user roles, define and set up the privacy levels for different data parameters and destination connections, and review the logs for user activities. See the screenshot below. This chapter will focus on the three functionalities: User Administration, Privacy Settings, and Audit Log.

Additionally, included in the Governance Overview section, in the upper part of the screen, you will find a summary of the company streams, source/destination connections and triggered events.



## User Administration

Here you can see the list of users who have access to the Data Stream Manager and their roles. You can add a new user by clicking "Add New User" on the top right corner. You can also delete a user by clicking "Delete" at the end of the row.

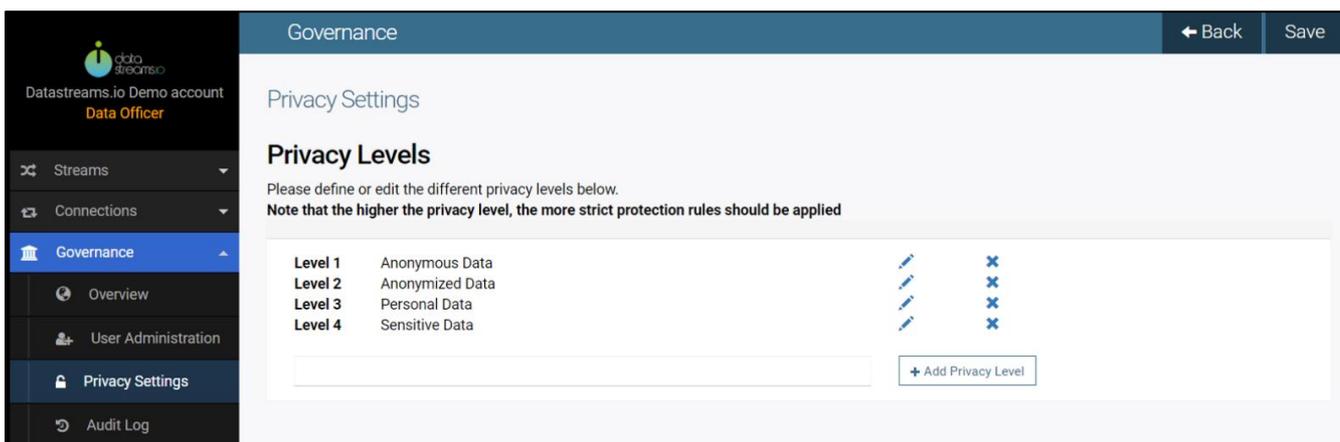
Users					+ Add new user				
Name	Mail domain	Roles	Last login	Toggle					
Admin	admin@datastreams.io	user, admin, streamer, viewer	23-12-2016 09:03	Viewer	Streamer	Developer	Data Officer	Admin	Delete
Data Officer	dataofficer@datastreams.io	user, viewer, streamer, admin, super_admin, developer	09-01-2017 08:53	Viewer	Streamer	Developer	Data Officer	Admin	Delete

# Privacy Settings

A unique feature of DSM is that you are in full control of all your data management requirements in order for you to offer your customers data security and data transparency. This can be done by setting up different privacy levels for the destination connections and the parameters of a stream, in order to make sure that data can only be sent to the destination whose privacy level is appropriate. The details of how to choose different privacy levels for different parameters and destination connections will be introduced in Connections and Streams section.

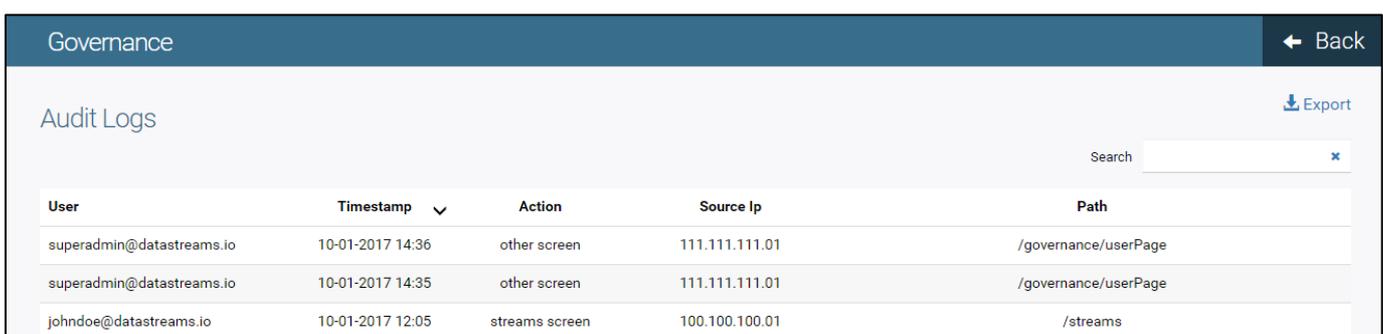
In the Privacy Settings, first of all, as a Data Officer, you need to define different privacy levels which will be used in the Connections and Streams sections. You need to define the privacy levels in such a way that the higher the level, the more sensitive the data is. See the example in the screenshot below.

You can add a new privacy level by typing its name on the text field and then clicking "Add Privacy Level" at the end of the row. You can also edit or delete a privacy level by clicking  or  behind it.



# Audit Log

In this section, you can see a list of user activities. You can export the list to an Excel file by clicking "Export" on the top right corner.

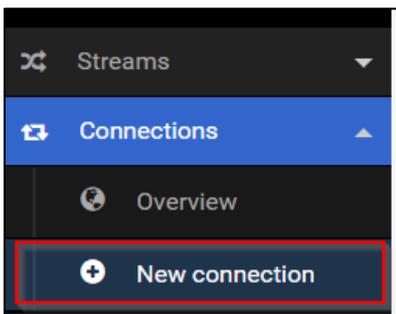


# Connections

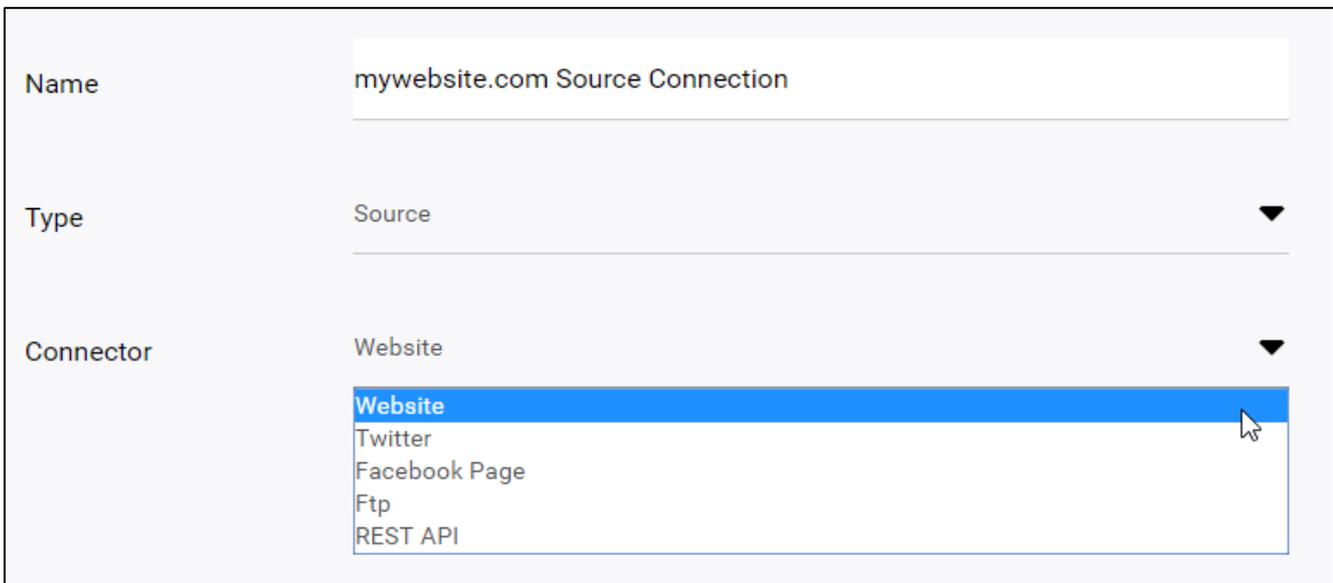
## Create a Source Connection

In order to be able to configure a Stream, you first need to define your Source and Destination Connections. This chapter will focus on these two types of connections.

To create a Source Connection, go to the left side and click the “New connection” button under “Connections” section.



Give your connection a name, then select Source as the type of Connection. Then choose the type of Connector you want to have as a data source.



The image shows a configuration form with three main sections: Name, Type, and Connector. The 'Name' field contains 'mywebsite.com Source Connection'. The 'Type' dropdown is set to 'Source'. The 'Connector' dropdown is set to 'Website', and a list of options is displayed below it, with 'Website' selected and highlighted in blue.

Name	mywebsite.com Source Connection
Type	Source
Connector	Website
	<ul style="list-style-type: none"> <li>Website</li> <li>Twitter</li> <li>Facebook Page</li> <li>Ftp</li> <li>REST API</li> </ul>

The following Connector options are available in this Data Stream Manager portal. Your interface may have more options available.

Website	Collecting data at any webpage load and webpage click. From any authorised domain. In Real-time.
Twitter	Collecting the past 24 hours of data from the Twitter reporting API. Automated to stream daily updates or an update when clicked. Maximum API volumes are 100 rows per request, 450 requests per 15 minutes. All per Twitter user account. <a href="#">More info on the API.</a>
Facebook page	Collecting the current data from a Facebook page Automated to stream daily updates or an update when clicked. Maximum API volumes are 4800 calls per 24 hours per engaged page user. <a href="#">More info on the API.</a>
FTP	Retrieves data from an FTP source, CSV format, and delivers it to any destination point.
REST API	Data can be provided to the DSM by using the data ingestion REST API. This means that any data source, sensor or device can provide data to a data stream if it can setup an HTTP connection (to the DSM).

Contact our support desk ([support@datastreams.io](mailto:support@datastreams.io)) for inquiries on any other data collection methods. Our core engine is highly integrated with the best open source, real-time intelligence technology available. We would be pleased to discuss your additional requirements, to tailor these to your specific needs. Note that some custom data collection methods options may be subject to mutual agreement before proceeding.

Click "Save & Proceed" on the right corner to go to the next step.

In the remaining of this section you can find the instructions on how to configure the Connector you just chose.

## Connect to a Website Source

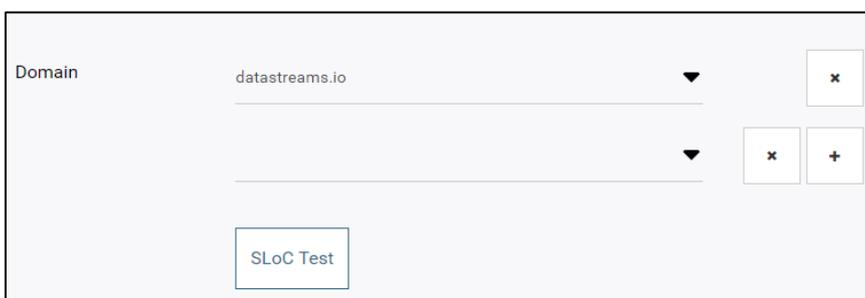
When using the DSM to measure a Website or Web based application, all functionalities are enabled through a single line of code (SLoC). This SLoC is a snippet of JavaScript and needs to be placed in the DOM of a website, or made available via any Tag Management Solution. The SLoC is generic and the same for any website or webpage to be measures.

On the right side of the screen, you can choose one or more domains to collect data from. Use the + button to add them and use the x button to delete.

Note: In case you do not find the domain that you are looking for, contact your company administrator for the DSM.

If you want to test if the SLoC (Single Line of Code) exists on your website, please click on “SLoC Test”.

Note: In case you are creating a connection for demo purposes, the SLoC does not need to be placed on the domain. The Chrome extension specially created for this application will take care of that for you to be able to configure your stream. However, this will be done locally and will collect data only for the interaction that you have with the website within the DSM application.



Next click on “Save & Proceed” and your Source Connection is set.

## Connect to a Facebook Source

To connect to a Facebook source, follow the same initial steps but choose “Facebook Page” as the source Connector. Then copy and paste the Facebook page name and click ‘Save & Proceed’.



## Connect to a Twitter Source

To connect to a Twitter source, you proceed in a similar way as in the previous type of connection above, but instead of the company’s Facebook page name, enter any Twitter author, hashtag or keyword. Then click ‘Save & Proceed’.

Edit your Twitter connection		← Back	Save & Proceed
Keywords	<input type="text" value="@datastreams_io OR #datastreams "/>	<p><b>How to create a Twitter connection</b></p> <p>Here you can specify the Twitter keywords, hashtags and accountnames to collect Twitter data from. When using multiple values separate them with a comma.</p>	

Note that many Twitter filtering and search functions can be used here. [Read more about the available options.](#)

## Connect to a FTP Source

To connect to a FTP source, complete the right credentials, decide the folder destination where you want to have your document placed after being processed and also the type of field separator to be used: i.e. “ ”, “;”, “-”.

Server	<input type="text" value="ftp.domain.tld/folder/location"/>
Username	<input type="text" value="Username"/>
Password	<input type="text" value="Password"/>
Processing complete folder	<input type="text" value="Directory"/>
Field separator	<input type="text" value=";"/>

Once you have completed the fields with the right information, click “Save & Proceed” to finish the configuration.

## Connect to a FTP Advanced Source

We take security seriously and therefore you can now connect to an FTP using a security key. Additionally, you have more flexibility to filter, exclude or further process the data you want to stream.

Please fill in on the right side the correct credentials and details required (\*) for this source connection.

On the Cron job field you need to introduce the information about when you would like the data processing to take place (i.e. certain time interval, continuously). Examples can be found on the left side of the screen. In the case of (s)FTP Host, you need to put the address of the folder where the files that you would like to process are. The folder address is based on the base/home folder you first encounter after logging in.

In order to connect to a (s)FTP, depending to the authentication requirements, you need to:

- fill in either the User Password or the Key
- fill in both User Password and the Key

If you would like to move your file from the current place to a different destination folder on the (s)FTP when the processing is finished, then fill in the folder path in this field.

You can use the File Filter option to filter files in the host folder. Furthermore, you can also exclude a column or for example, the header row, etc. For more examples check the left side of the screen.

Step 1: Instruction	Step 2: Proceed	← Back	Save & Proceed
<p><b>Edit your advanced Ftp Source Connection</b></p> <p>Please fill in on the right side the right credentials and details required (*) for this source connection.</p> <p><b>Cron job examples:</b></p> <ul style="list-style-type: none"> <li>@ */15 * * * * ? Every 15 minutes</li> <li>* */15 * * * * ? Every 15 seconds</li> <li>@ */5 * * * * ? Every 5 seconds</li> <li>@ 0 * * * * ? Every 24 hours, once at 00:00:00</li> </ul> <p><b>KeepAlive:</b> Use this option if you want to keep the processing flow continuously active. Otherwise the flow would be stopped by the system after a certain period, usually a half hour, of idle time.</p> <p><b>(s)FTP Host Examples:</b></p> <p>ftp://example.com/source/ This is an example of your server address together with the folder you are monitoring. Files in this folder will, depending on some of your further wishes in this connection, be processed. The folder address is based on the base/home folder you first encounter after logging in.</p> <p><b>(s)FTP Authentication:</b></p> <p>In order to connect to a (s)FTP, depending to the authentication requirements, you need to:</p> <ul style="list-style-type: none"> <li>• fill in either the User Password or the Key</li> <li>• fill in both User Password and the Key</li> </ul> <p><b>(s)FTP Processing Done Folder:</b></p> <p>If you would like to move your file from the current place to a different destination folder on the (s)FTP when the processing is finished, then fill in the folder path in this field.</p> <p><b>File Filter Examples:</b></p> <ul style="list-style-type: none"> <li>*.fileEnd One can filter files in the host folder based on the END of the file name.</li> <li>fileNameStart* One can filter files in the host folder based on the START of the file name.</li> <li>fileStart*fileEnd.csv.gz Or both.</li> </ul> <p><b>Exclude From Processing Example:</b></p> <p>columns[0] != "headerfirstColumn" This is an example of how you can exclude from processing the header row, based on the first column of that row. Use the Groovy language for further use cases.</p> <p><b>Column Split/Tokenize Examples:</b></p> <p>body.tokenize(',') This piece of Groovy code sets the columns data element with all the items in the row, split up, based on the character in quotes. So if a row consists of "a","b", the first column element will be "a".</p> <p>body.tokenize(',').collect { column -&gt; column.substring(1, column.length() - 1) } One can also use this piece of code to remove the extra elements from the columns. For example, if your row consists of "a","b", the first column element will be a. Or you can decide not to split the row at all, in which case you fill in body.</p>	<p>Cron Job* <input type="text" value="0 0 0 * * ?"/></p> <p>Type* <input type="text"/></p> <p>(s)FTP Host * <input type="text"/></p> <p>User Name * <input type="text"/></p> <p>User Password* <input type="text"/></p> <p>Key* <input type="text" value="Key"/></p> <p>(s)FTP Processing Done Folder <input type="text"/></p> <p>Checking For Files Delay <input type="text"/></p> <p>File Filter <input type="text"/></p> <p>Unzip* <input type="text" value="No"/></p> <p>Trim* <input type="text"/></p> <p>Exclude From Processing <input type="text"/></p> <p>Column Split/Tokenize* <input type="text"/></p>		

## Connect to a REST API Source

Any data source, sensor or device can provide data to a data stream if it can setup an HTTP connection (to the DSM). Choose your Connection domain and you are ready to go.

Connection	Select option > +
URL:	dynamic.dimml.io/restapi/669983075xSBW
cURL:	<pre>curl -H "Content-Type: application/json" -X POST --data '{"calltype":"restapi\'' - -referer http://restapi/669983075xSBW?dimml=tenadata.dev.0 http://dynamic.dimml.io/crea te</pre>
httpie:	<pre>http --json POST http://dynamic.dimml.io/create Referer:http://restapi/669983075xSBW? dimml=tenadata.dev.0 calltype=restapi</pre>

Note: To read more about this connection click [here](#).

## Connect to a DEQ Source

Listening to specific events and enriching the data object server side can now be done using the Digital Event Queue (DEQ) source and destination connections.

When using the DSM to measure a Website or Web based application, all functionalities are enabled through a single line of code (SLoC). This SLoC is a snippet of JavaScript and needs to be placed in the DOM of a website, or made available via any Tag Management Solution. The SLoC is generic and the same for any website or webpage to be measures.

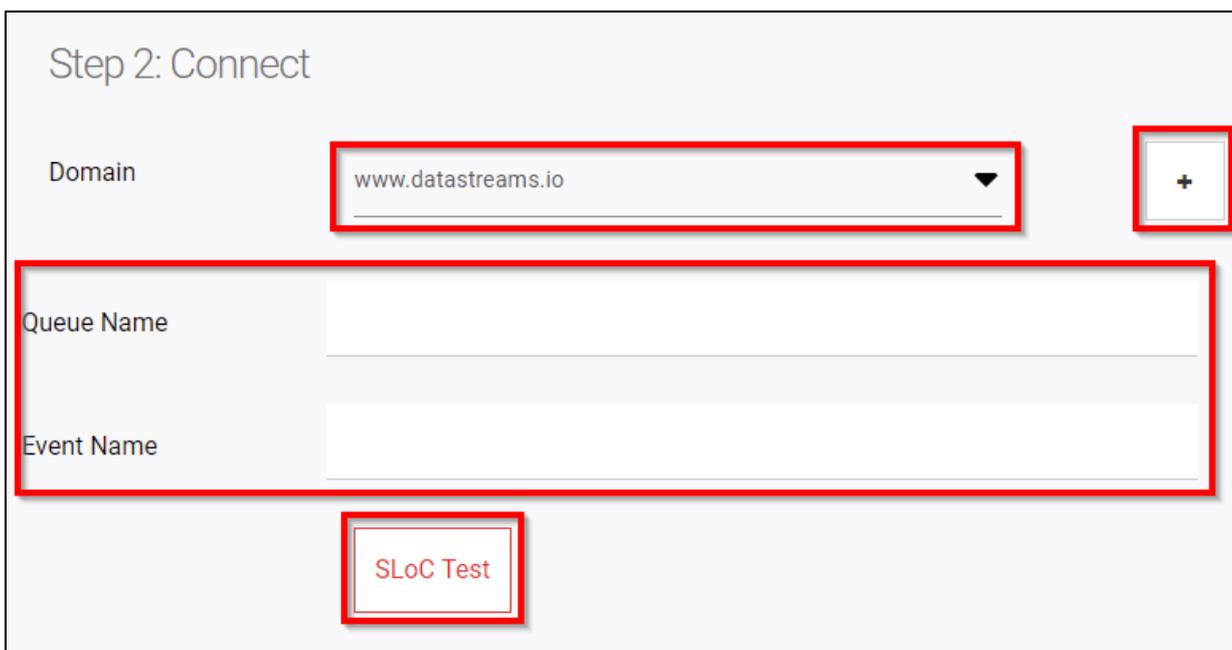
On the right side of the screen, you can choose one or more domains to collect data from. Use the + button to add them and use the x button to delete.

Note: In case you do not find the domain that you are looking for, contact your company administrator for the DSM.

If you want to test if the SLoC (Single Line of Code) exists on your website, please click on “SLoC Test”.

Note: In case you are creating a connection for demo purposes, the SLoC does not need to be placed on the domain. The Chrome extension specially created for this application will take care of that for you to be able to configure your stream. However, this will be done locally and will collect data only for the interaction that you have with the website within the DSM application.

Next, in order to create a DEQ source you need to feel in the Queue Name and Event Name:



Step 2: Connect

Domain

Queue Name

Event Name

Next click on “Save & Proceed” and your Source Connection is set.

## Connect to a Salesforce Source

To connect to Salesforce, provide the right credentials in the fields from the right side of the screen. You can use the examples in the left side of the screen for guidance.

**Note:** For objected and the data format, introduce the elements separated by commas.

**Edit your Salesforce Source**

On the right hand side you can fill in the details of the (Salesforce) API call.

An example url is: `https://test.salesforce.com`

Example data format code: `Name, first_name__c, last_name__c, credits__c`

Url	<input type="text"/>
Client Id	<input type="text"/>
Client Secret	<input type="text"/>
Refresh Token	<input type="text"/>
objectId	<input type="text"/>
Data Format	<input type="text"/>

## Connect to an Adobe Source

Do you want to reuse the data stored in Adobe Analytics and send it to different end points in a governance compliant way? Configure your source connection by introducing the right credentials and the report suite that you want to use as a source.

On the right hand side you can fill in the details of the (Adobe) API call.

An example api url is: `https://api.omniture.com/admin/1.4/rest/`

An example end point: `Company.GetReportSuites`

An example Data Object: `report_suites`

Api Url	<input "="" type="text" value="https://api.omniture.com/admin/1.4/rest/?method="/>
User Name	<input type="text"/>
Secret	<input type="text"/>
Endpoint	<input type="text" value="Company.GetReportSuites"/>
Data Object	<input type="text" value="report_suites"/>

## Connect to a Cobra Source

Have you been wondering how you can manage your HR data? With DSM, you can extract data from Cobra, in a privacy compliant way, do further processing if needed, and then reuse data based on your needs.

On the right hand side you can fill in the details of the (Cobra) API call.

An example authorization url is: `https://api.ctbps.nl/v3/oauth/token`

An example api url is: `https://api.ctbps.nl/v3/odata/`

The credentials string is the clientId with clientSecret, concatenated with a colon and then base64 encoded  
 so: `Base64Encode('clientId' + ':' + 'clientSecret')`.

Base64encoding can be done below.

Auth Url	<code>https://api.ctbps.nl/v3/oauth/token</code>
Api Uri	<code>https://api.ctbps.nl/v3/odata/</code>
Endpoint	
Credentials	

## Connect to a Timer Source

With each day, new technologies appear on the market that can enhance and ease your business processes. All these generate tremendous amount of data. Integrating everything can be challenging. With this in mind we have added a new connection in the Data Stream Manager. Connect now with over 200 sources and stream data in real-time or batch.

Use the left side of the screen for guidance to configure this source connection.

Cron job examples :

- `0 */15 * * * ?` Every 15 minutes
- `*/15 * * * * ?` Every 15 seconds
- `0/5 * * * * ?` Every 5 seconds
- `0 0 0 * * ?` Every 24 hours, once at 00:00:00

**keepalive** Normally the flow would be deactivated by the system after a certain period of time. To prevent this the keepalive mechanism from the example can be used to keep the flow active.

**once** This cron job code, will run the scheduled cron job once, when published. The development publishing will happen on refreshing the configure page or the configure confirm page. Pressing the preview button, will also publish, but also call the script through a rest api.

Dimml code part:

The following piece of custom code can be put in, or changed based your requirements. This will allow it to be automatically called and integrated into the normal flow of the stream. This part can also be left empty.

```
concept TimerConcept { flow (timer-in) => nop (timer-out)
}
```

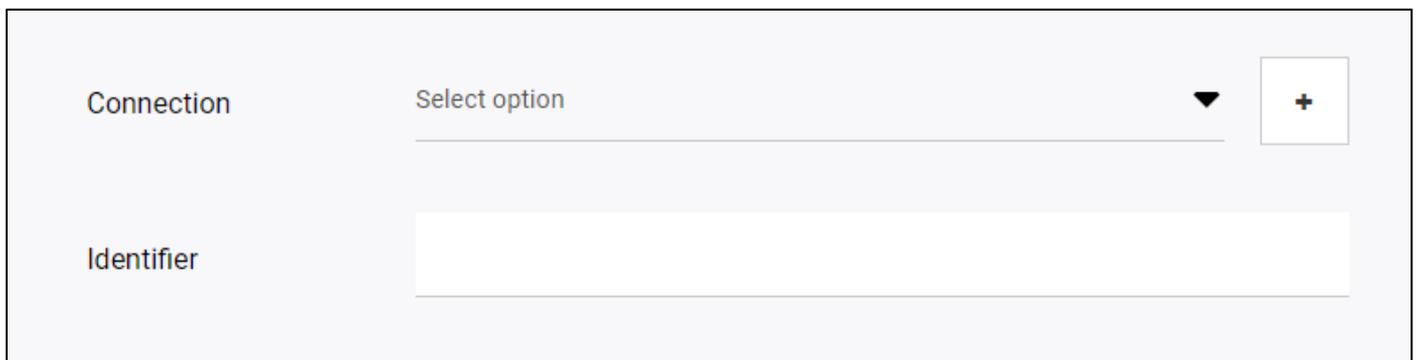
Connection	Select option
Cron	<input type="text"/>
dimml	<input type="text"/>

## Connect to a Native App as a Source

To connect to a Native App to start data collection and processing all you need to do is follow the below instructions:

- Give your connection a name, chose source as a Type and then Native App for connector, finally click on Save and Proceed to move to the next Step.
- Choose a connection domain. You associate a domain with a connection in order to make sure that you have control over the entire process of data collection. Unless a domain is enabled for going live for data collection, no one can use the implementation for production usage, data collection and/or processing is more secure in this way.
- Give an Identifier for your App. This will be used to identify your app and included in the configuration details for data collection and processing for Native Apps.

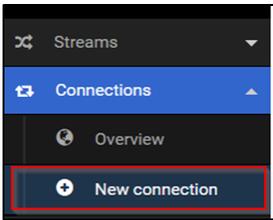
Click Save&Proceed and your connection is ready to be used to create a Stream.



Connection	Select option	▼	+
Identifier	<input type="text"/>		

# Create a Destination Connection

To create a Destination Connection, go to the left side and click the “New connection” button under “Connections” section.

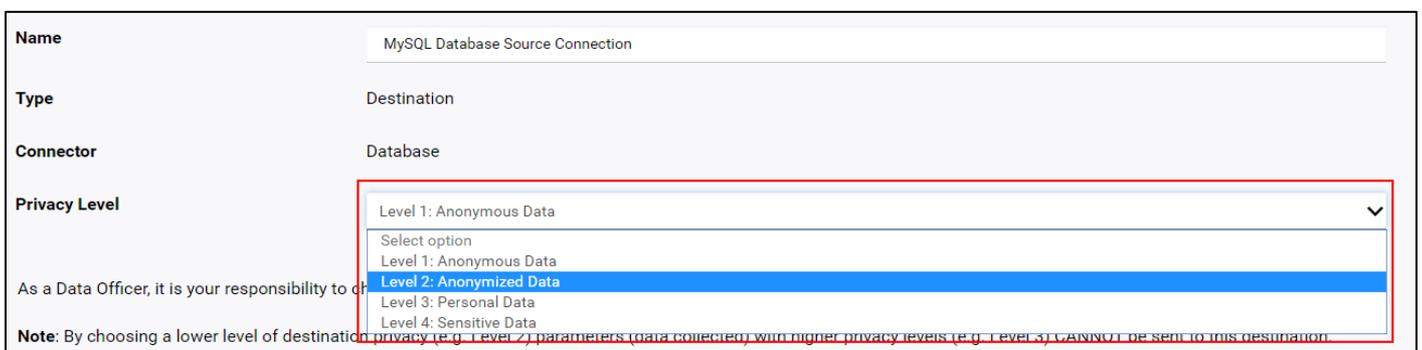


Give a name to your Destination Connection. After you choose the Type of connection as Destination, in the Connector field you can choose the destination where you want to send your data to. In the example below, FTP was chosen as a destination connection.



### For Data Officers:

As a Data Officer, you are responsible for setting up the privacy levels of the destination connections that are created by others. Therefore, when you edit the destination connection, you will see an extra option – “Privacy Level”, where you need to choose a privacy level for the destination connection from a list of privacy levels that you already defined in the “Governance→Privacy settings” section. See the screenshot below.



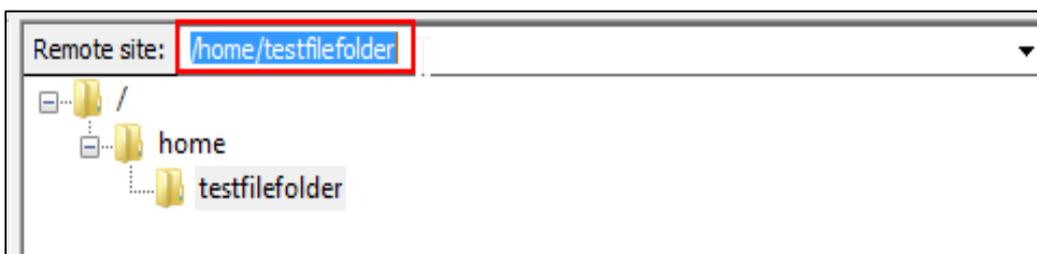
**Please be aware that by choosing a lower privacy level for a destination connection (e.g. Level 2), parameters (data collected) with higher privacy levels (e.g. level 3) CANNOT be sent to this destination.**

## Connect to an (S)FTP destination

When streaming to an FTP connection, set the structure of the file that will be created. Enter your FTP connection details. Set the correct subdirectory, filename, datestamp and timing attributes.

FTP type	FTP
FTP host	ftp.domain.tld
Username	Username
Password	Password
Directory	Directory
Filename prefix	File prefix
Filename date	<input type="checkbox"/>
Date format	Select option

In case of any doubt about what subdirectory on your FTP location where you should store the data to, check your FTP software. Typically, it shows the subdirectory you are connecting to in the tool, like this example in the tool FileZilla. Enter this directory structure straight into Datastreams but without the preceding /. In this example you would enter 'home/testfilefolder' as the directory in the FTP connection.



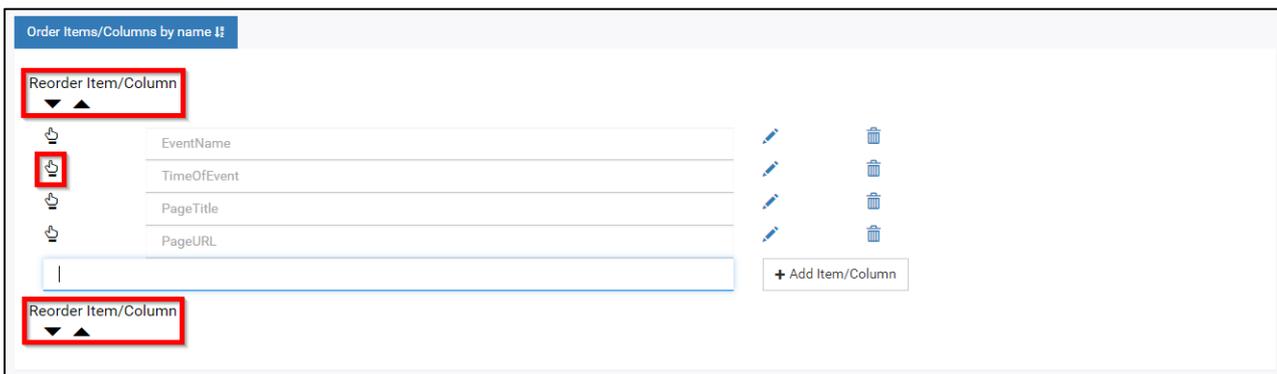
Once you introduced the correct credentials and specifications click "Save & Proceed". Next, you need to configure the column structure of your file. To do so, click on the 'Configure Ftp structure' button.

Name	my ftp destination
Type	Destination
Connector	Ftp
Details	ftp://ftp.domain/home/myFTP

Then the following screen will appear:



Adding items/columns can be done in the required order (position is kept based on the order of entry) or random. Once done, you can reorder if needed by simply clicking on the selection button and then on the up/down buttons.

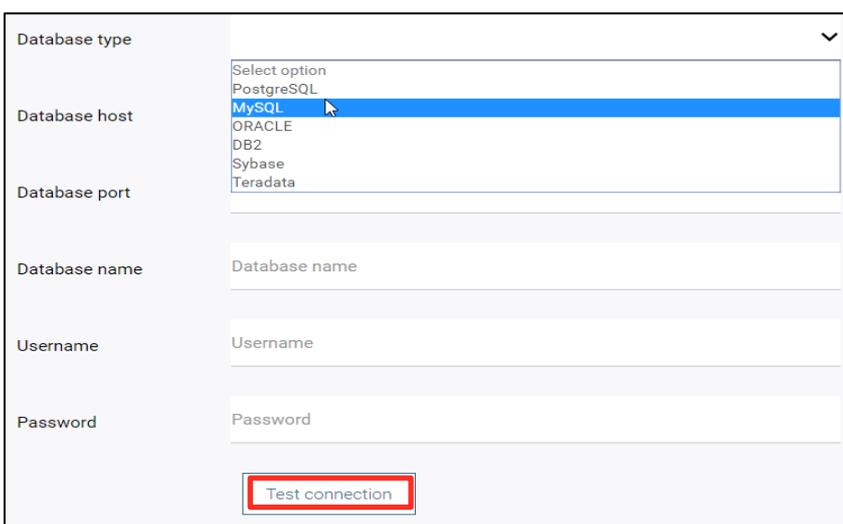


After finishing editing your configuration click "Save & Proceed".

## Connect to a Database destination

When choosing a Database Destination connection, several custom options may be available in your interface. In this example, the connection to a MySQL database will be addressed.

Set the Database credentials. If you want to test whether the database connection is correct, you can click "Test connection". Click on the "Save & Proceed" button to successfully create a connection.



On the following page you can configure the database by clicking “Configure Database Structure”.

Name	Database Test
Type	Destination
Connector	Database
Details	mysql://domain/my database

[Configure Database structure](#)

There are 3 approaches that can be taken afterwards:

1. Configure the table and the columns yourself, for your Database Destination Connection. **Note:** For the data to be streamed to your Database you will need to create a corresponding configuration within the Database itself
2. Add an already existent Database Template. **Note:** For the data to be streamed to your Database you will need to create a corresponding configuration within the Database itself
3. Click ‘Read Database structure’ and simply select the table which you want to stream the data into.

Use Template
Database Advanced Data Model Template
▼
Add
Replace

Read Database structure

Demo\_MySQL\_Database

datastreamsio\_facebookpage

datastreamsio\_website\_manageportal

datastreamsio\_website\_for\_demo

datastreamsio\_website

Name	Name			Update Behaviour
<input type="checkbox"/>	EventName			Empty
<input type="checkbox"/>	TimeOfEvent			Empty
<input type="checkbox"/>	WebsiteDomain			On Update Add
<input type="checkbox"/>	PageId			Key
<input type="checkbox"/>	PageSequence			Empty
<input type="checkbox"/>	TimeStep			Empty
<input type="checkbox"/>	PageIdPath			Empty

‘Update Behaviour’ is an optional feature. It is used to add up numeric values in the table upon each database write. This is done, in order to aggregate the data in a table.

The Primary Key of your database table will be checked to add this value with the previous value and store the result. If this feature is used, it is typical for a user to apply this to all the event metrics in the table.

This can be a useful option to help reduce data volumes for reporting purposes. Or it can also be beneficial when needing to deliver specific data to a marketing automation system.

When setting up tables in your database be sure to configure the table columns correctly. Consider if your data requires a primary key to be set, so that no duplicate data will be accidentally captured. It is also advised to set the column types of your table to *text* or *varchar* initially to be sure the table accepts all possible data coming in, allowing you to change it to a more restricted column type later on.

The Appendix 2 within this manual, includes several example cases for setting up a database table correctly.

## Connect to DOMO destination

When choosing DOMO as a Destination connection, you can begin your configuration by introducing the right credentials in order to connect.

<p><b>Edit your Domo Destination</b></p> <p>On the right hand side you can fill in the details of the (Domo) API call.</p> <p>An example authorization url is: <a href="https://api.domo.com/oauth">https://api.domo.com/oauth</a></p> <p>An example api url is: <a href="https://api.domo.com/v1">https://api.domo.com/v1</a></p> <p>The credentials string is the clientId with clientSecret, concatenated with a colon and then base64 encoded so: Base64Encode('clientId' + ':' + 'clientSecret').</p> <p>Base64encoding can be done below.</p> <input type="text"/> <input type="button" value="Base64 encode"/>	<table> <tr> <td>Auth Url</td> <td><input type="text" value="https://api.domo.com/oauth"/></td> </tr> <tr> <td>Api Url</td> <td><input type="text" value="https://api.domo.com/v1"/></td> </tr> <tr> <td>Domo Stream Id</td> <td><input type="text"/></td> </tr> <tr> <td>Credentials</td> <td><input type="text"/></td> </tr> </table>	Auth Url	<input type="text" value="https://api.domo.com/oauth"/>	Api Url	<input type="text" value="https://api.domo.com/v1"/>	Domo Stream Id	<input type="text"/>	Credentials	<input type="text"/>
Auth Url	<input type="text" value="https://api.domo.com/oauth"/>								
Api Url	<input type="text" value="https://api.domo.com/v1"/>								
Domo Stream Id	<input type="text"/>								
Credentials	<input type="text"/>								

On the left side of the screen you will find help instructions to fill in the correct credentials. Once done, click Save&Proceed. Afterwards, click on Configure DOMO structure button.

In the configuration page you can configure the structure of your destination table, the column names.

Use Template	<input type="text"/>	<input type="button" value="Add"/>	<input type="button" value="Replace"/>
<input type="button" value="Delete all Items/Columns"/>			
<input type="text"/>			<input type="button" value="+ Add Item/Column"/>



There are 2 approaches that can be taken for configuration:

1. Configure the table and the columns yourself, for your Salesforce Destination Connection. **Note:** For the data to be streamed to your Salesforce account you will need to create a corresponding configuration within your account.
2. Add or replace an already existent Salesforce Template. **Note:** For the data to be streamed to your Salesforce account, you will need to create a corresponding configuration within your account.

## Connect to Digital Event Queue (DEQ) destination

When choosing Salesforce as a Destination connection, you can begin your configuration by introducing the right credentials in order to connect.

**Edit your Deq Destination**

On the right hand side you can fill in the Technical properties such that DSM can stream the data real time.

Queue Name	<input type="text"/>
Event Name	<input type="text"/>

In the configuration page you can configure the structure of your destination, the items names.

Use Template

There are 2 approaches that can be taken for configuration:

1. Configure the items names yourself, for your DEQ Destination Connection.
2. Add or replace an already existent DEQ Template.

## Connect to a REST API destination

When streaming to a REST API connection it is recommended to test the connection manually first using curl. Only after manually sending data to the REST API successfully do we recommend to use this type of destination connection in a stream

After providing the name of the connection and clicking Save & Proceed, the details of the REST API connection need to be provided. One way of providing data to the endpoint is to provide them as query parameters to the URL. The URL property should contain the based URL of the REST API, so excluding query parameters. The query parameters to add (their names) can be added in the Dynamic Query Parameters field. The value of these parameters can be defined in the connect screen using the mapping functionality. Note that since not all REST APIs use query parameters, this field can be left empty.

The method of providing the data (GET or POST) is defined by the REST API endpoint owner, typically GET is used together with the query parameters as source and POST for a specific POST object containing the data. The object can be defined in the Body property by providing the code to generate the object. This provides the flexibility to create a data object in any way for instance a list in a specific order or a JSON formatted object. The fields (as defined in the stream) can be used in the code which can be executed in server side Javascript or Groovy. A list with a single element can be defined with the code for the Body that can be seen in the screen shot below.

**Edit your Rest Api Destination**

On the right hand side you can fill in the URL of the (REST) API call.

In the url field you can also fill in constant query items. By adding `?parameter=value`

The header `Content-Type: application/json` is added to the requests automatically. Adding it to the headers parameter will cause an error when the code is executed.

Example header: `Accept: application/json`

Groovy Body example with Dynamic Stream parameters: `' [{"Pagename":'+ PageTitle+' '+ ', ' + "'Pagenumber":'+ PageId+' '+ ', "Timestamp":'+ (new Date()).format("yyyy-MM-dd'T'HH:mm:ss")+', "Views":'+ metric_PageView+' } ]'`

Url	<code>https://api.mydomain.com/v1/ingestdata</code>
<p>Dynamic Query Parameters</p> <p>+ Add Item/Column</p>	
Method	Post
Headers	<code>Accept: application/json</code>
Body	<code>"["+Pagename+"]"</code>
Body Language	Groovy

The headers of the request can also be defined. Each header should be provided on a separate line. To complete the configuration click Save & Proceed.

## Connect to Vertica Database Destination Connection

When choosing a Database Destination connection, several custom options may be available in your interface. In this case, the connection to a Vertica database will be addressed.

Set the Database credentials. If you want to test whether the database connection is correct, you can click "Test connection". Click on the "Save & Proceed" button to successfully create a connection.

Database type: Vertica

Database host: [input field]

Database name: [dropdown menu]

Username: [input field]

Password: [password field]

[Test connection button]

On the following page you can configure the database by clicking “Configure Database Structure”.

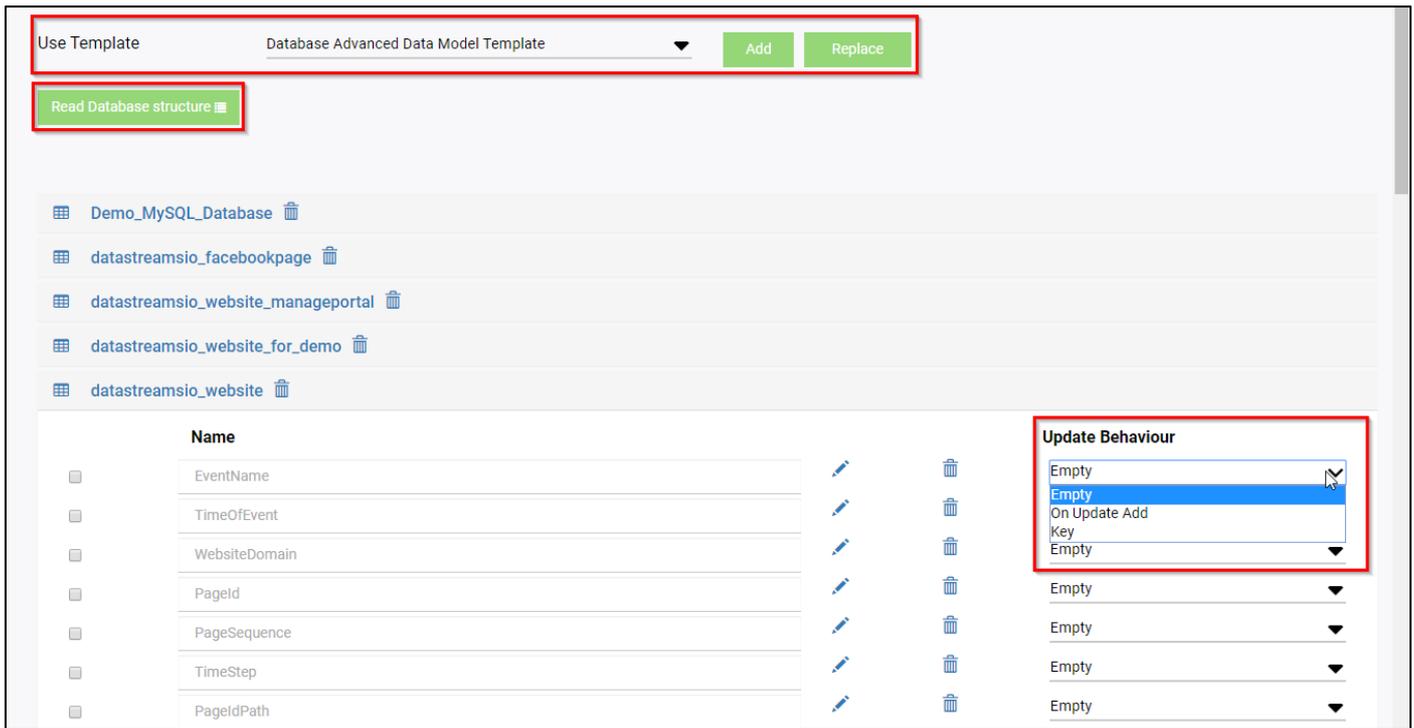
Congratulations, you successfully configured a connection.

Name	Vertica Destination Connection
Type	Destination
Connector	Database
Details	vertica://DomainVertica/Vertica Database

[Configure Database structure button]

There are 3 approaches that can be taken afterwards:

1. Configure the table and the columns yourself, for your Database Destination Connection. **Note:** For the data to be streamed to your Database you will need to create a corresponding configuration within the Database itself
2. Add an already existent Database Template. **Note:** For the data to be streamed to your Database you will need to create a corresponding configuration within the Database itself
3. Click 'Read Database structure' and simply select the table which you want to stream the data into.



Use Template Database Advanced Data Model Template Add Replace

Read Database structure

Demo\_MySQL\_Database

datastreamsio\_facebookpage

datastreamsio\_website\_manageportal

datastreamsio\_website\_for\_demo

datastreamsio\_website

Name	Update Behaviour
EventName	Empty
TimeOfEvent	Empty
WebsiteDomain	On Update Add
PageId	Empty
PageSequence	Empty
TimeStep	Empty
PageIdPath	Empty

‘Update Behaviour’ is an optional feature. It is used to add up numeric values in the table upon each database write. This is done, in order to aggregate the data in a table.

The Primary Key of your database table will be checked to add this value with the previous value and store the result. If this feature is used, it is typical for a user to apply this to all the event metrics in the table.

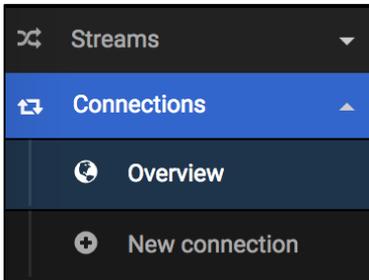
This can be a useful option to help reduce data volumes for reporting purposes. Or it can also be beneficial when needing to deliver specific data to a marketing automation system.

When setting up tables in your database be sure to configure the table columns correctly. Consider if your data requires a primary key to be set, so that no duplicate data will be accidentally captured. It is also advised to set the column types of your table to *text* or *varchar* initially to be sure the table accepts all possible data coming in, allowing you to change it to a more restricted column type later on.

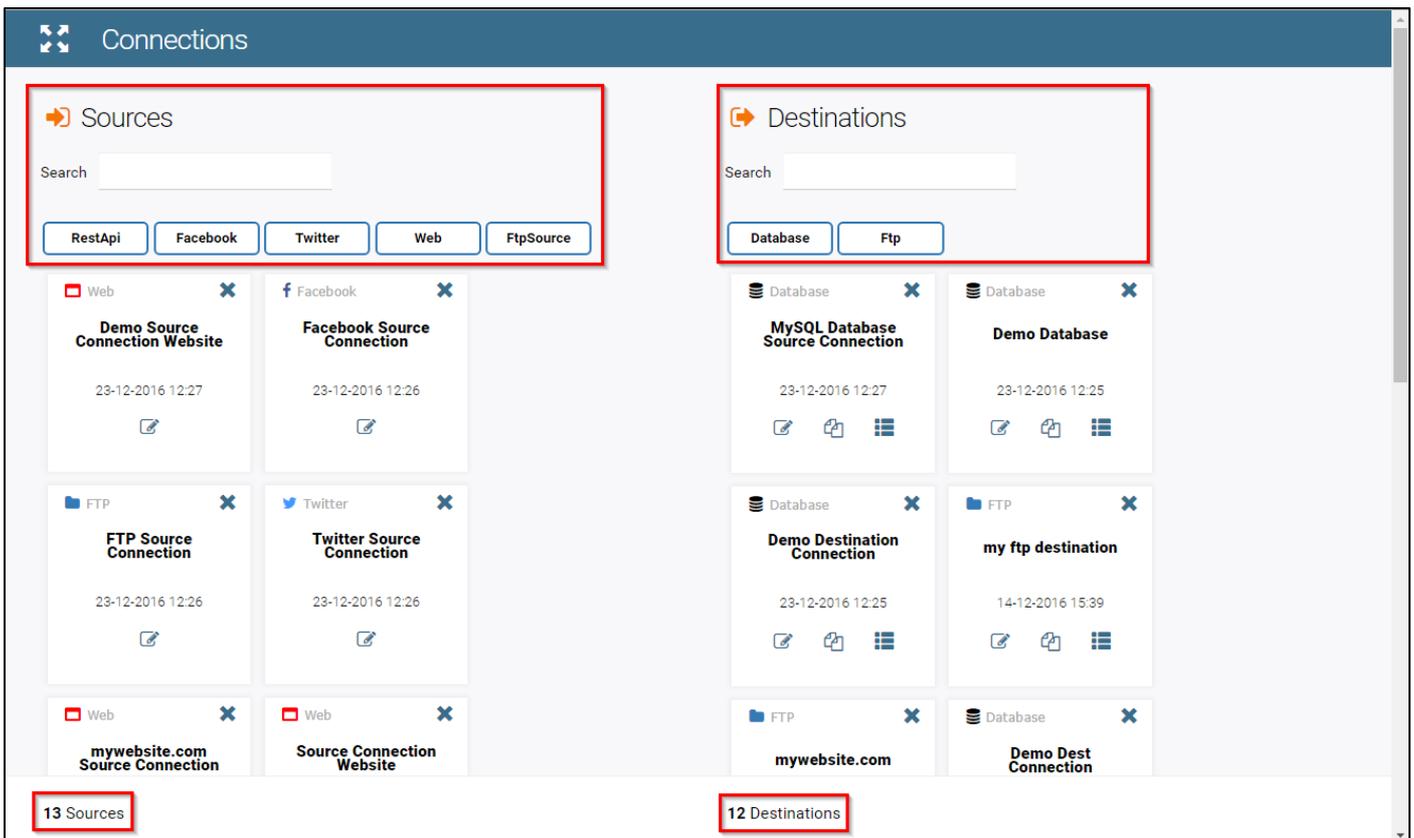
The Appendix 2 within this manual, includes several example cases for setting up a database table correctly.

# Connections overview

After you created source and destination connections, they will appear on the Connections overview page.



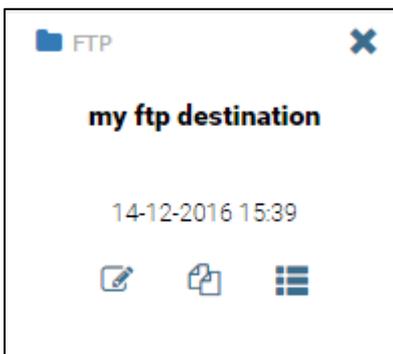
On this page you will see two columns: Sources and Destinations. Under each section, you can find a specific connection by typing its name in the searching text filed or using the filter. The filter is generated dynamically, which means that you can only see a “website” filter after you created a website source connection. Similarly, you can only see a "Ftp" filter after you created an Ftp destination connection. At the bottom of each section, you can see how many Sources/Destination Connections are created. See below.



Information of each connection is shown on a tile. For a source connection, its type is shown at the top left corner (e.g. web). You can find its name and the date of creation at the center. Clicking the pencil  allows you to start editing it. Clicking the delete icon  will delete the connection.

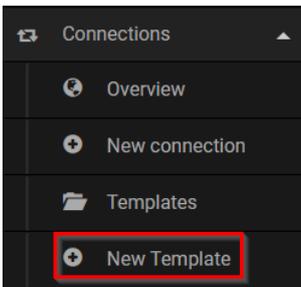


For a destination connection, there are two additional options. Clicking the duplication icon  will create a copy of the original connection. Clicking the configuration icon , you will be directed to the column configuration page, where you can edit the columns.



## Create a Template

In this Connections section, you will be able to create a Template for your Destination Connection. In this way, you can create a template for a specific type of connection and use it for all connections of that type (Ftp, Database, etc.) in the future. This will reduce configuration time and will make sure that you are in control of the data that you are streaming.



Once you click on new template the following screen will appear:



Give a name to your template and then choose one of the 3 types of templates:

### Database Template:



Once you have added the wanted tables and columns to your tables, you need to make sure that they also match in the database.

When you create a new Database Destination Connection, all you will need to do is to select and add your template in the Configuration step. Before you publish live your project you should make sure that the same tables and columns are added within your Database to which you connected to through the Destination Connection.

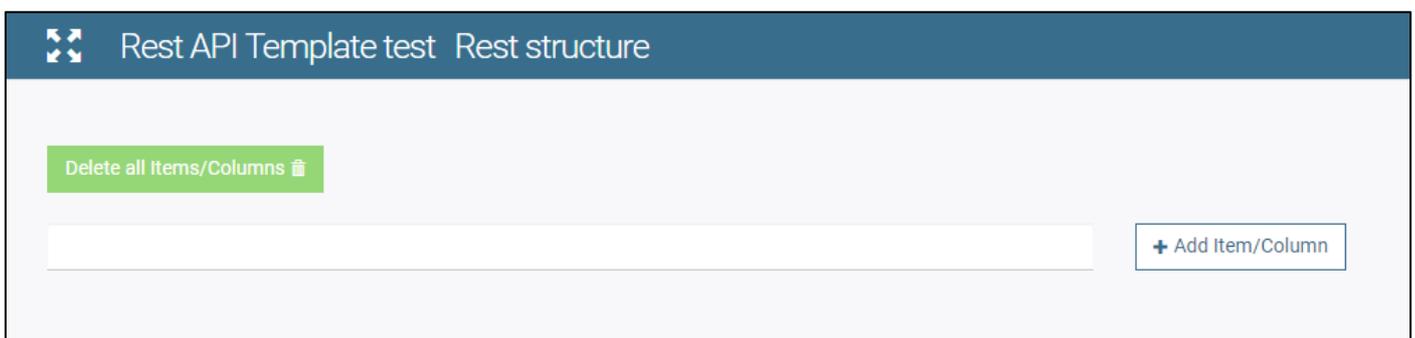
## FTP template



You can add your items/columns by clicking on the Add button. Once done, if required you can order your items in alphabetical order, or even delete everything should you want so.

When you create a new FTP Destination Connection, all you will need to do is to select and add your template in the Configuration step.

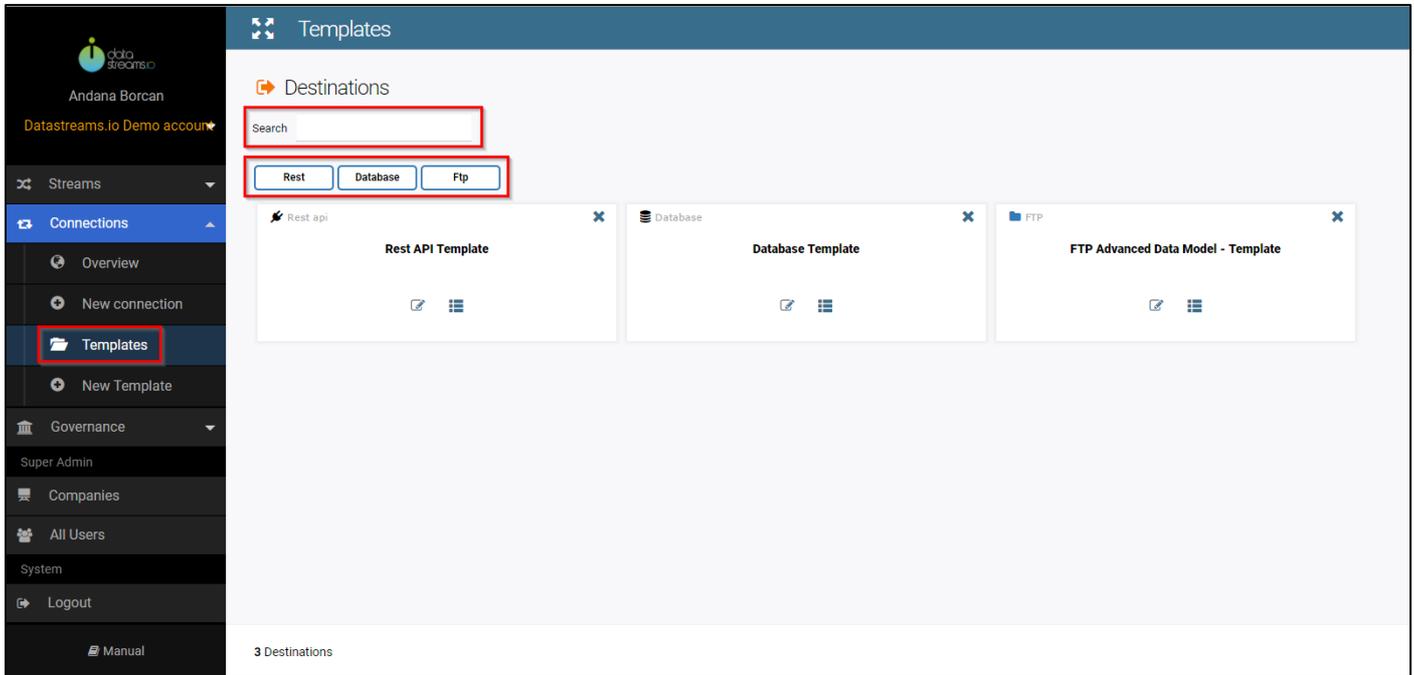
## Rest API template



Once you have added your items to your template, you can use this for your destination connection configuration. Before putting your project live, make sure that you have also added the corresponding parameters in the body of the Rest API connection configuration step.

# Templates

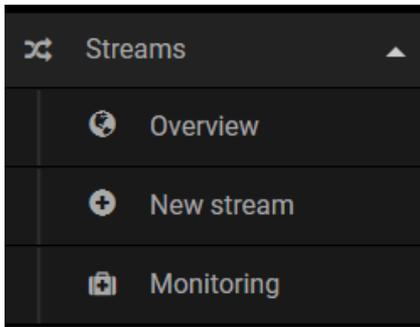
This is the overview of your already created templates. You can search or filter for your template based on the name or respectively on the type.



# Streams

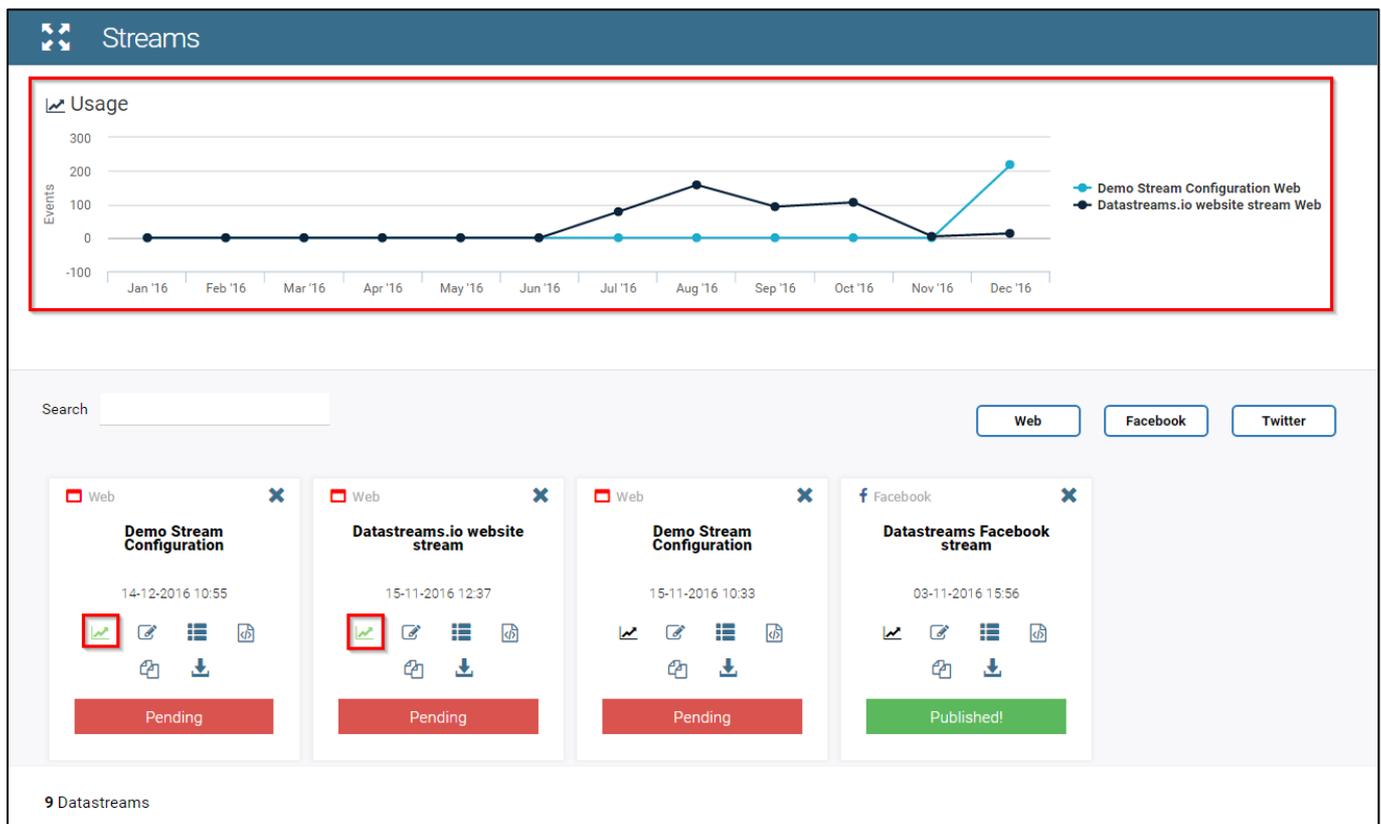
## Streams overview

Once you are logged in, the first screen you will see is the 'Stream Overview' page.



This screen shows all the streams available to you. This feature is available to the following user roles: i.e. viewer, streamer, data officer, admin or superadmin.

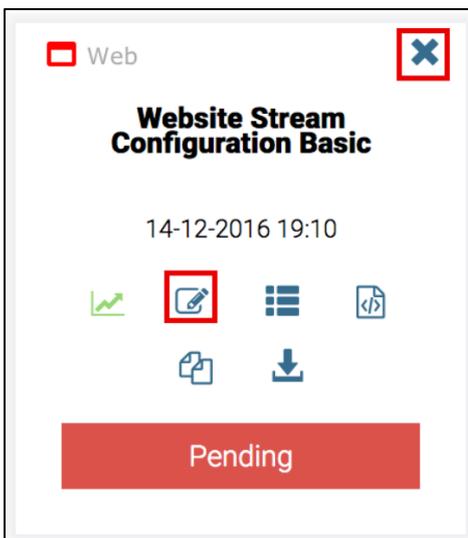
The *Stream Overview* page also contains usage overview of the stream per month. Clicking on the graph icon  on each stream, the trend line for the volume of events of a stream will show.



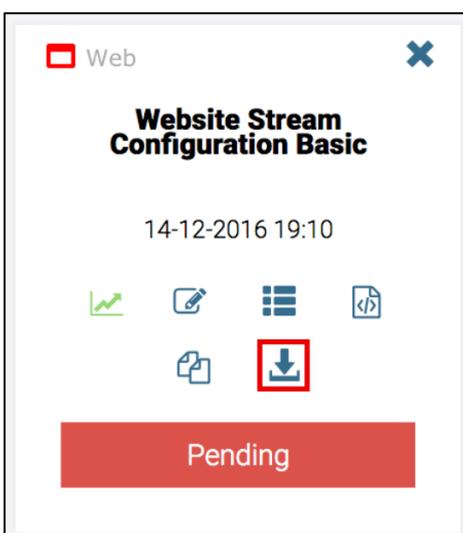
To find a stream, you can type the stream's name in the search text field. You can also use the filter to find the stream, see below. The filter is generated dynamically, which means that you can only see a "facebook" filter after you create a stream for facebook.



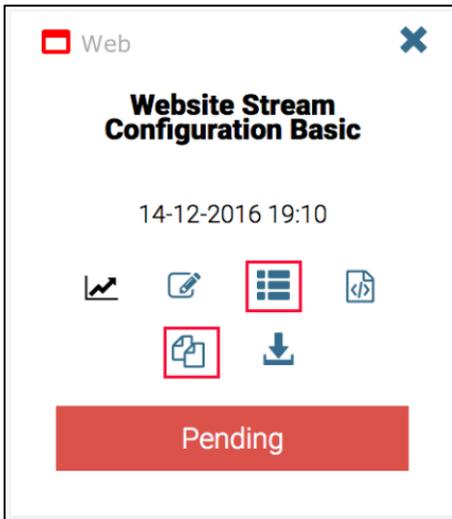
Each stream is shown as a file. The stream type is shown at the top left corner (e.g. web). Clicking the pencil  on each stream allows you to start editing it. Clicking the delete icon  on the right top of the stream file will delete the stream.



Clicking the download icon  allows you to download an Excel or pdf file specifying the data model of this stream. It includes details of the data capture method and description per variable so that the document serves as an explanatory document for stakeholders and users of the data stream.

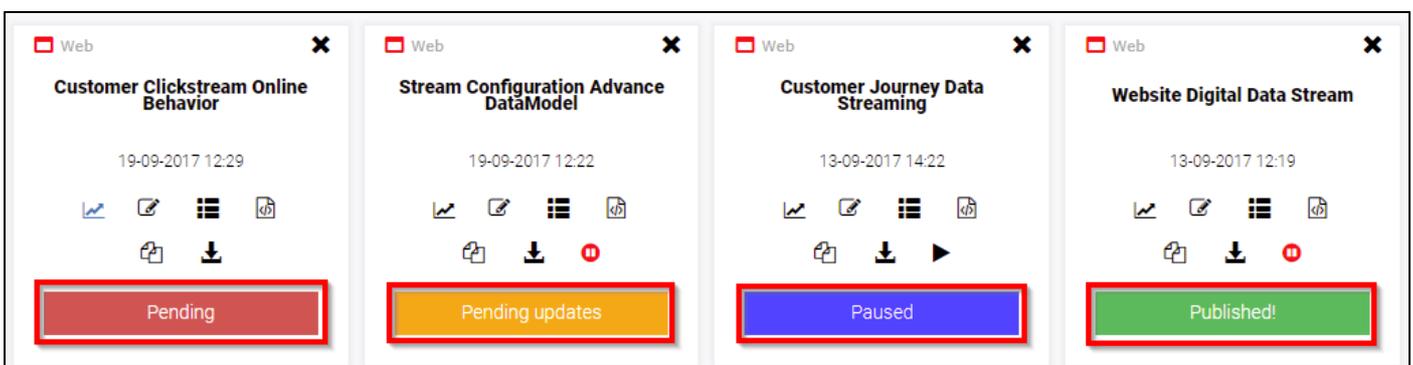


You can also duplicate a stream by clicking on the duplication icon . Clicking on the mapping icon  will lead you to a page where you need to map stream variables to the destination (see "Mapping the Stream Variables" section for more information).



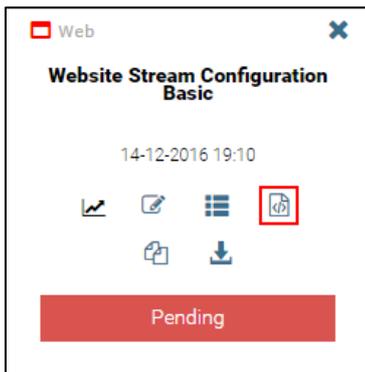
You are the person in control and can now pause your data streaming project at any moment. Ready to restart? No problem, just click on play and everything will be up and running again. For each of your streams/project you will have the following statuses:

- Pending: The stream has not yet been published; no data collection and processing takes place
- Pending Updates: The stream has an implementation already live; data collection and processing is taking place
- Paused: The stream was live and then paused; no data collection and processing takes place
- Published: The stream is live: data collection and processing is taking place

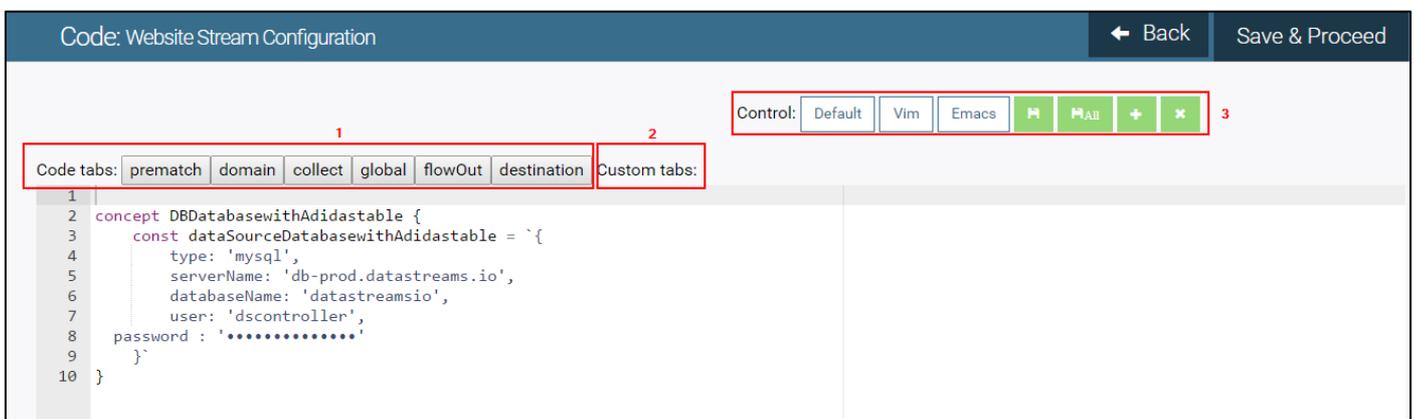


## For Developers:

Only users with a developer role can see the “developer tool” icon. See below.



Clicking on the icon will lead you to a page where you can customize your application.



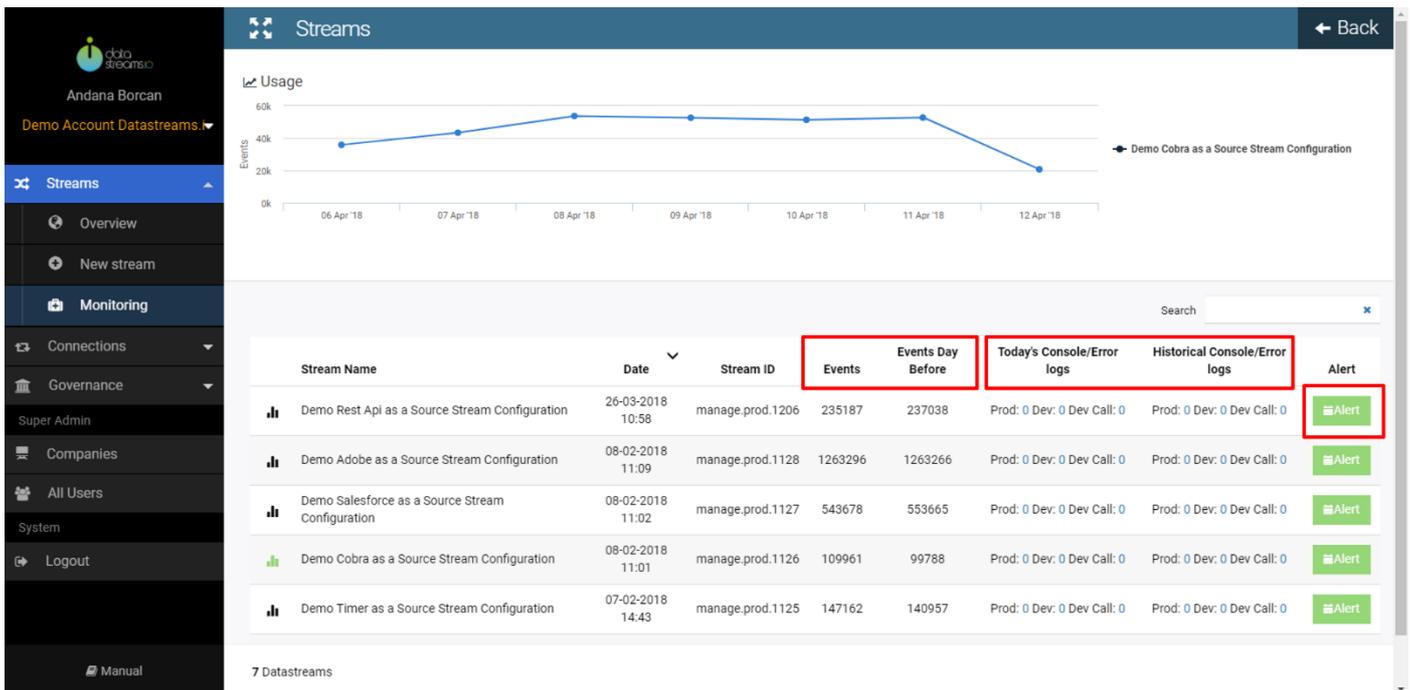
The important functionalities are explained below:

**Code tabs (1):** Those are the Dimml code that the Data Stream Generated in the back end. Editing these is not possible.

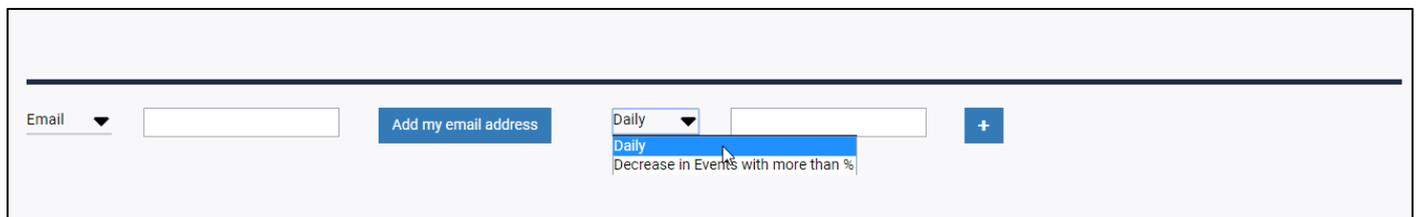
**Custom tabs (2):** Those are the customized code tabs added by the developer.

**Control (3):** A developer can use this menu to generate customized code. Clicking on one of the first three tabs “Default”, “Vim” and “Emacs”, you can switch to different editors to write your code. By clicking  you create a new custom tab. Clicking  or  you save your code individually or all at once. You can delete a custom tab by clicking .

# Monitoring



Once your stream goes live you can monitor the events processes and sent to your chosen destination. You can add daily alerts (see below) to be kept up to date and be in control of your data collection and processing. The graph can also be used to see the events for the past week.



The screenshot shows the alert configuration form. It includes an 'Email' dropdown menu, an input field for the email address, and a blue 'Add my email address' button. Below this, there is a 'Daily' dropdown menu, an input field for a percentage, and a blue '+' button. A dropdown menu is open under the 'Daily' selection, showing the option 'Daily Decrease in Events with more than %'.

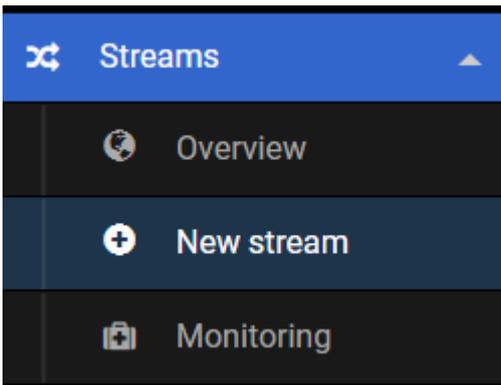
Alerts can also be set for the case when there is a decrease in events for a selected stream. Just type in the number that should represent the different (e.g. 20 for 20%)

## Create and configure a new data stream

With the Data Stream Manager, it is possible to setup new streams from any digital source. This section will focus on how to configure a data stream.

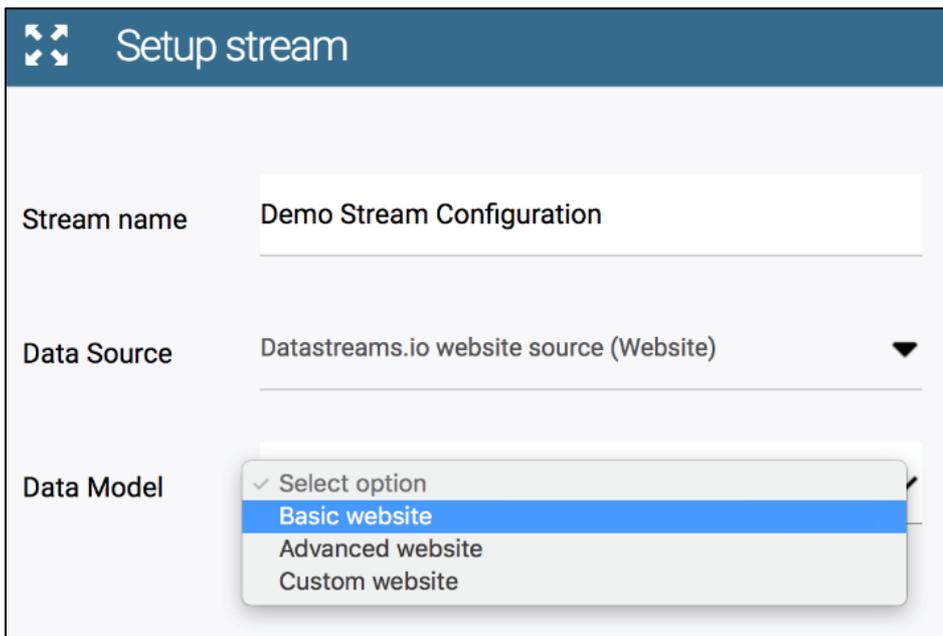
The following features are only available to the following user roles: streamer, data officer, admin, superadmin.

To start the process, click the 'New stream' button under Streams section on the right side of the screen.



### Create and configure Website as a source Stream

Then you will see the screen below.


 A screenshot of the 'Setup stream' configuration form. The form has a title bar 'Setup stream' with a maximize icon. It contains three main sections:
 

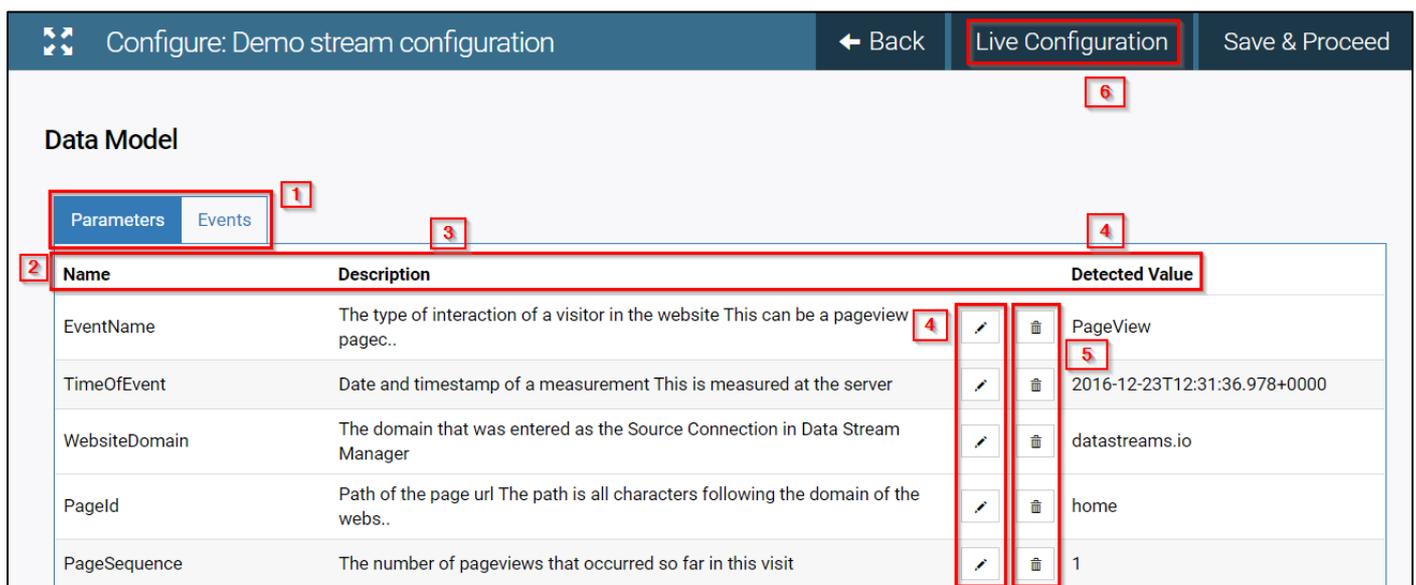
- Stream name:** A text input field containing 'Demo Stream Configuration'.
- Data Source:** A dropdown menu showing 'Datastreams.io website source (Website)' with a downward arrow.
- Data Model:** A dropdown menu with a list of options: 'Select option' (checked), 'Basic website' (highlighted in blue), 'Advanced website', and 'Custom website'.

Firstly, provide a name for your new stream. Secondly select the data source you wish to collect data from. Note that the Connector types of source connections are shown between brackets (e.g. website, Twitter, Facebook, other). Thirdly select the data model to start with the configuration for your website data stream.

By default, the following data models will be available. Your environment may have extra options implemented. For more information about the detail of each data model, please refer to Appendix 3.

Data model name	Contents
Basic website	Over 20 most used website measurements are preset. Use this for relatively simple data collection from a website
Advanced website	Over 30 most used website measurements are preset. Use this for the most extensive data collection from a website. All data that can be captured automatically will be available in the preset.
Custom website	No website variables are preset. Use this option to configure your own stream starting with an empty template.

After proceeding to the next page the preset data model will be displayed. The highlighted areas from the image below are important functions in this page:



Configure: Demo stream configuration   ← Back   **Live Configuration**   Save & Proceed

### Data Model

Parameters   Events

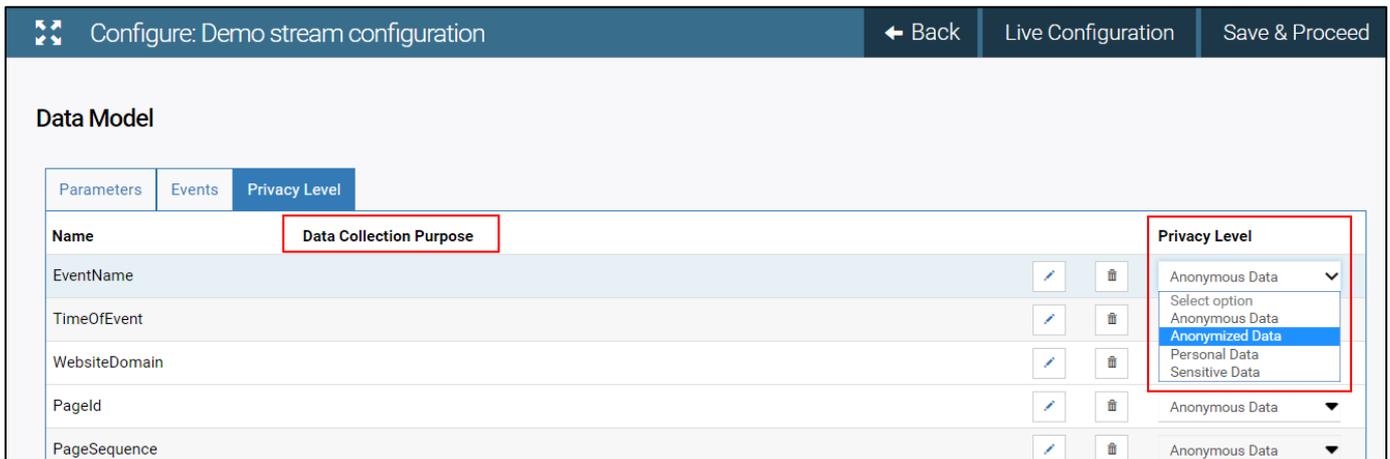
Name	Description	Detected Value
EventName	The type of interaction of a visitor in the website This can be a pageview pagec..	PageView
TimeOfEvent	Date and timestamp of a measurement This is measured at the server	2016-12-23T12:31:36.978+0000
WebsiteDomain	The domain that was entered as the Source Connection in Data Stream Manager	datastreams.io
PageId	Path of the page url The path is all characters following the domain of the webs..	home
PageSequence	The number of pageviews that occurred so far in this visit	1

- **Parameters & Events (1):** The **parameters** define what information should be collected in a stream, while **events** define when they should be collected.
- **Name (2):** The name of the parameter/ event in the stream
- **Description (3):** The rows of this column contain the description of the parameter/event
- **Detected value (3):** shows a snapshot of the value collected for this variable
- **Pencil icon (4):** Edit the description for this variable
- **Trashcan icon (5):** Delete this variable from the data model

- **Live configuration (6):** This opens the live configurator window to customize data collection for a website further.

For Data Officers:

As a data officer, besides “Parameters” and “Events” tabs, you will see an extra tab called “Privacy Level”, where you need to set the privacy level for each parameter of the stream. See the screenshot below.



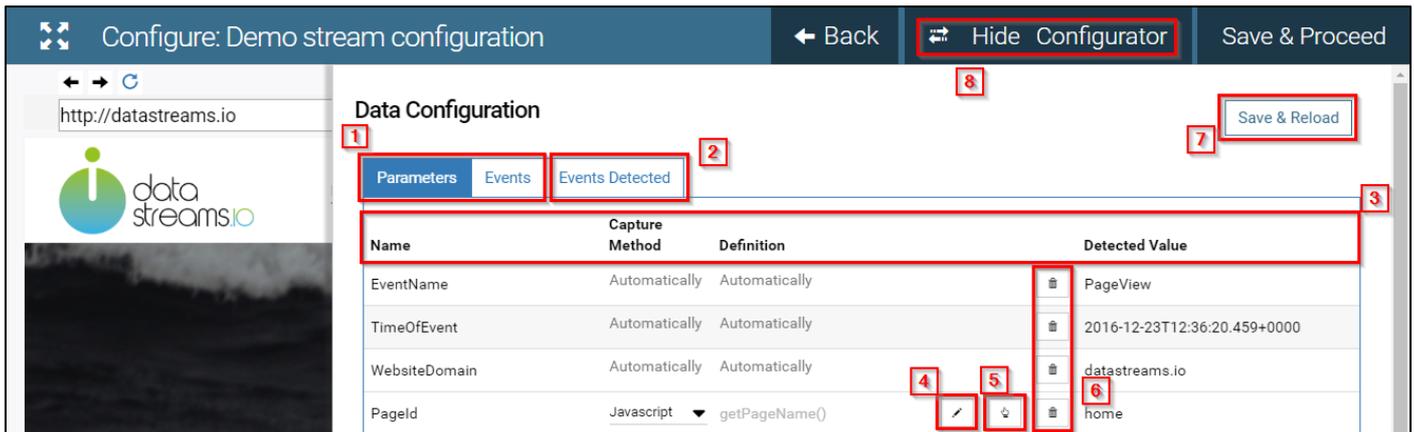
Under the “Data Collection Purpose” column, you need to describe the purpose for the company to collect this specific parameter and how it will be used.

Under the “Privacy Level” column, you need to choose the privacy level for each parameter. The list of available privacy levels are the ones that you already defined in the “Governance → Privacy settings” section. Note that by choosing a higher privacy level for some parameters, they will not be sent to the destination which has a lower privacy level. For example, if you set the privacy level as “Level 3: Personal Data” for the parameter “PageId”, and set all other parameters as “Level 1: Anonymous Data”, later on when you choose a destination connection with a privacy level of “Level 2: Anonymized Data”, the parameter “PageId” will be filtered out so it will not appear in the data mapping list (see “Mapping the stream variables” section for more information).

### Using the Live Configurator

The ‘Live Configurator’ is an interactive screen to click-and-select website elements to measure. Note that it only is available for ‘website’ as the data source. It also requires the Google Chrome plugin to function. Any data collection or website interaction that happens within this IFrame does not have any impact on the live website.

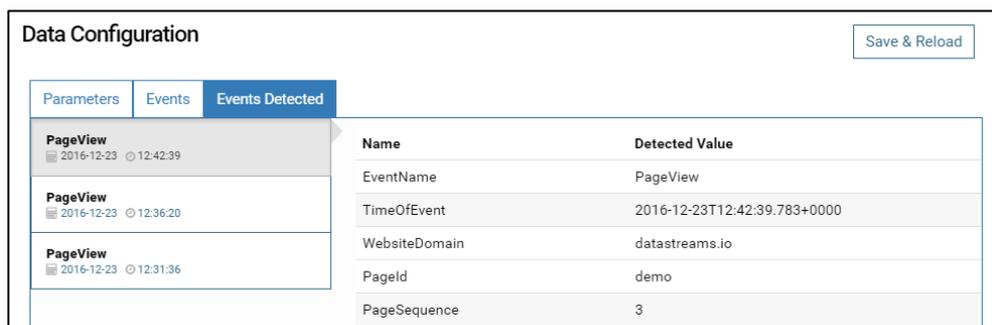
After clicking the Live Configurator button you will see the website for which you created a connection previously to collect the data from. When clicking the “Show Configurator”/”Hide Configurator” button the Configurator screen toggles in and out of the screen.



The screenshot shows the 'Data Configuration' interface for a demo stream. At the top, there are navigation buttons: 'Back', 'Hide Configurator' (highlighted with a red box and callout 8), and 'Save & Proceed'. Below this is a browser-like header with 'http://datastreams.io' and a 'Save & Reload' button (highlighted with a red box and callout 7). The main content area has three tabs: 'Parameters', 'Events', and 'Events Detected' (highlighted with a red box and callout 1). The 'Events Detected' tab is active, showing a table with columns: 'Name', 'Capture Method', 'Definition', and 'Detected Value'. The table contains four rows of data. The first row is 'PageView' with a detected value of 'home'. The second row is 'PageView' with a detected value of 'datastreams.io'. The third row is 'PageView' with a detected value of '2016-12-23T12:36:20.459+0000'. The fourth row is 'PageView' with a detected value of 'PageView'. There are icons for editing (pencil, callout 4), deleting (trashcan, callout 6), and a pointer (callout 5) for the 'PageView' row. A 'Save & Reload' button is also present in the top right of the configuration area (callout 3).

Please find below some explanations for the above highlighted elements:

- **Parameters & Events (1):** The **parameters** define what information should be collected in a stream, while **events** define when they should be collected.
- **Events Detected (2):** This shows the last 5 events that were measured from this data source. Clicking the event on the left will show you the values associated with it. See below.



This image shows a detailed view of the 'Events Detected' tab. On the left, there is a list of three 'PageView' events with timestamps: '2016-12-23 12:42:39', '2016-12-23 12:36:20', and '2016-12-23 12:31:36'. On the right, a table displays the detected values for these events:

Name	Detected Value
EventName	PageView
TimeOfEvent	2016-12-23T12:42:39.783+0000
WebsiteDomain	datastreams.io
PageId	demo
PageSequence	3

- **Name (3):** The name of the parameter/event in the stream.
- **Capture Method (3):** the data collection method used to collect the variable.
- **Definition (3):** the exact code or criterion used to collect the variable.
- **Detected value (3):** shows a snapshot of the value collected for this variable.
- **Pencil icon (4):** Edit the description for this variable.
- **Pointer Icon (5):** Allows you to select an element from the page and generates the code automatically based on the Capture Method used.
- **Trashcan icon (6):** Delete this variable from the data model.
- **Save & Reload (7):** Once you are done with editing the Parameters, Events and Privacy Level, you need to Save & Reload in order to save your configuration.
- **Hide/Show Configurator (8):** In order to see/hide the Configurator click on this button.

The configurator assists you with defining and collecting new variables from a webpage. To create a new variable, scroll to the bottom of the parameters or events tab. Then enter a name for the variable and

choose the data capturing method. Next select your data capturing method. Many data capturing methods are available to allow you to click-and-select from the website.

In order to capture data from a website page, click on the pointer icon- see highlighted area in the image below.



After clicking the Configurator automatically disappears and the website page is shown. You can now click and select whatever element from the website that you would like to collect information about. Once you have done that, the Configurator page will appear again and you will be able to see the results of your selection: the code generated automatically based on your selection. Optionally enter a description for the new variable. When done, click the save icon.



You will find various options for collecting parameters and generating events from a website source. See the Appendix 1 for more examples on how to use the advanced features. A high level overview is shown below:

#### Data capturing methods for Parameters:

- **Content of URL parameter:** Measure the contents of any query parameter. For instance: ?utm\_source ?cmp ?clid.
- **Form name:** Measures the contents of the form name html tag.
- **Content of link:** Measures the contents of the link clicked.

Extra options for users with the Developer user role assigned:

- **JavaScript:** Measures data with the use of JavaScript instructions (see Appendix)
- **Groovy:** Measures data with the use of Groovy instructions
- **DimML:** Measures data with the use of DimML instructions

#### Data capturing methods for Events:

- **Element is:** Measures when *this HTML* identifier was found as the clicked element. Select any HTML element from the website directly. The code to find the HTML identifier is generated automatically. This generates code to locate the item in the page.
- **Link name contains:** Measures when a link name contained this value.
- **URL contains:** Measures the number of times a page URL contained this value.
- **Form is:** Measures when a form has this HTML identifier near, or in, the form code.
- **Form name:** Measures when a form name contains this value.

Extra options for users with the Developer user role assigned:

- **JavaScript:** Measures data with the use of JavaScript instructions. (Appendix)

After completing your configuration, click 'Save & Proceed'.

## Create and configure Stream: Facebook & Twitter as sources

Choosing Facebook or Twitter as a source will add to your data model different parameters.

Facebook	All available Facebook page variables are preset.
Twitter	All available Twitter variables are preset.

To do a test and check the type of data collected based on the Data Model already created or other further parameters that you might have added for further processing, click on **Preview Data**.

This executes an API call to the data source to collect the data immediately. You should see data immediately appearing in your configured FTP or database destination connection.

**Twitter:** after clicking on Preview data, the data send to your destination connection will include information from the last 24 hours (with a limit of 100 messages).

**Note:** In case you do not see data coming in your destination please check if the keywords chosen in your source connection have been used in the past 24hours

**Facebook:** after clicking on Preview data, the data send to your destination connection will include information from the last month.

## Create and configure Stream: FTP as sources

When creating a stream using FTP as a source, there isn't an out of the box template. If you want to simply stream data from a csv file to a destination of your choice, use the below instructions:

- Make sure you chose the same separator in your source configuration as the one from your file
- For streaming first column's contents create a parameter, give it a name and in the definition field introduce fieldlist[0]. For the second column you would have to introduce fieldlist[1]. And so on and so forth.
- If you would like, for example, to concatenate the contents of 2 columns you would have to introduce in the definition field: fieldlist[0] + fieldlist[1]
- If you would like to add a space between the contents concatenated, for example, you should have to use the following definition: fieldlist[0] + " " + fieldlist[1]

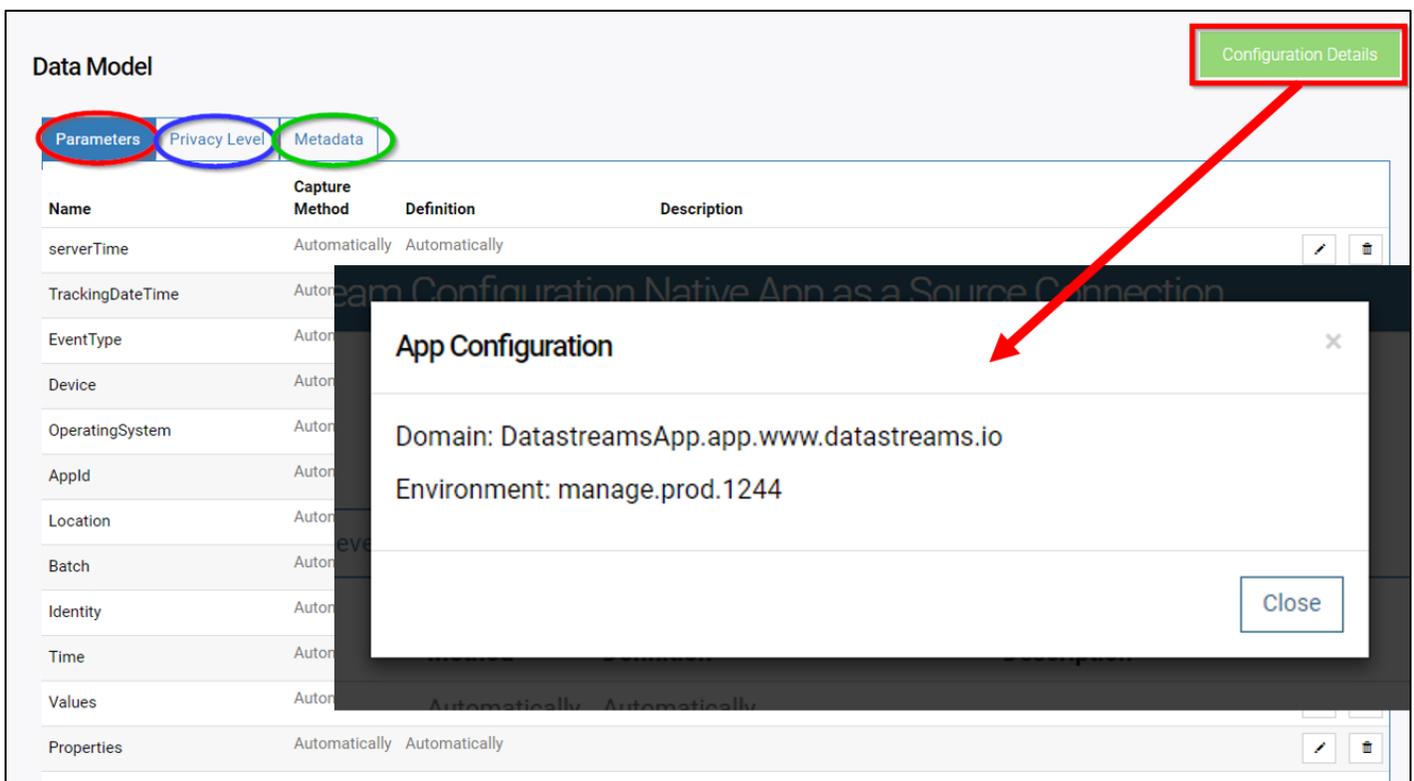
## Create and configure Stream: Native App as sources

For a Native App as a source Stream there is included a standard Data Model that can further be enriched with other parameters based on your requirements.

Furthermore, your data model is enriched with the parameters included in the Metadata Tab. Here you have data containing information about:

- storage period for the data collected,
- deletion data for that specific data,
- the consent given by the data subject, etc.

You can stream all data collected and processed in a GDPR compliant way using the Privacy Settings tab, to assign a Privacy level to each of the parameters collected



**Data Model**

Parameters Privacy Level Metadata

Name	Capture Method	Definition	Description
serverTime	Automatically	Automatically	
TrackingDateTime	Automatically		
EventType	Automatically		
Device	Automatically		
OperatingSystem	Automatically		
AppId	Automatically		
Location	Automatically		
Batch	Automatically		
Identity	Automatically		
Time	Automatically		
Values	Automatically		
Properties	Automatically	Automatically	

**Configuration Details**

**App Configuration**

Domain: DatastreamsApp.app.www.datastreams.io  
Environment: manage.prod.1244

Close

Furthermore, by clicking on the Configuration Details you will be provided with the information needed to configure your App for data collection and processing.

## Mapping the stream variables

When having completed the stream variable configuration click "Save & Proceed" to reach the next step in the Stream Configuration process: Mapping Parameters. This page allows you to map the stream variables to the destination connection correctly. First choose the destination connection from the dropdown menu on the left. Next, click the column icon to display the contents of the destination connection on the right hand side of the page.

**Destination Connection**

Database Destination Connection (Database) ▼

Select option

- Database table (Database)
- Database Destination Connection (Database)
- FTP Destination Connection (Ftp)

☰ +

The right hand side of the page will show the Filter and the table(s) in existence for your connection. The Filter allows you to choose based on which criterion you need to send the mapped parameters to your database. For example, you can choose to stream the mapped data only when a form is being submitted.

**Destination Connection**

Database Destination Connection (Database) >

☰
+
Read Database structure ☰

Filter for **Database Destination Connection (Database)** :

▼ Filter

Dimension	Value Equals
EventName <span style="float: right;">&gt;</span>	<input type="text" value="FormSubmit"/>

Mapped Dimensions for **Database Destination Connection (Database)** :

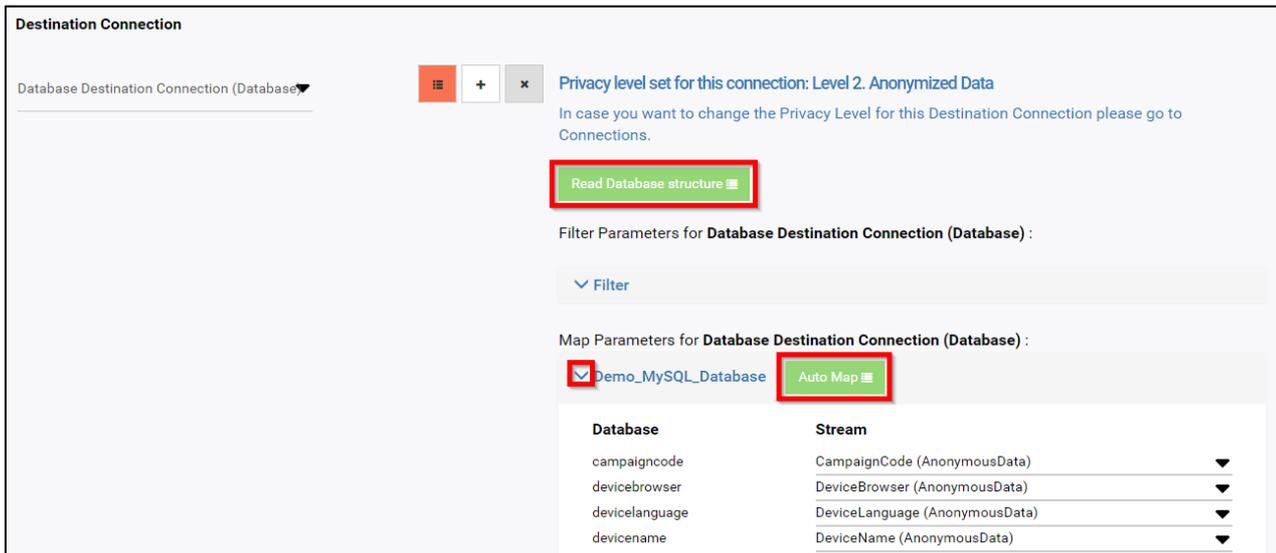
▼ Demo\_MySQL\_Database

Database	Stream
campaigncode	CampaignCode <span style="float: right;">&gt;</span>
devicebrowser	DeviceBrowser <span style="float: right;">&gt;</span>
devicelanguage	DeviceLanguage <span style="float: right;">&gt;</span>
devicename	DeviceName <span style="float: right;">&gt;</span>
deviceversion	DeviceVersion <span style="float: right;">&gt;</span>
encryptedid	EncryptedIp <span style="float: right;">&gt;</span>

You can choose to not use the filter if you want to stream your data for all events configured for your stream. In this step you also have the option to read the database structure of your Destination Connection.

If you have multiple tables, select the one(s) you want to send your data to and proceed with mapping using the dropdown to select the correct parameters from the stream configuration.

For the case when the columns in your database coincide with the naming of the parameters that you have in your stream, you can simply click on Auto Map and the mapping will be done for you automatically.



**Destination Connection**

Database Destination Connection (Database) ☰ + × **Privacy level set for this connection: Level 2. Anonymized Data**  
 In case you want to change the Privacy Level for this Destination Connection please go to Connections.

**Read Database structure** ☰

Filter Parameters for **Database Destination Connection (Database)** :

▼ Filter

Map Parameters for **Database Destination Connection (Database)** :

Demo\_MySQL\_Database **Auto Map** ☰

Database	Stream
campaigncode	CampaignCode (AnonymousData) <span>▼</span>
devicebrowser	DeviceBrowser (AnonymousData) <span>▼</span>
devicelanguage	DeviceLanguage (AnonymousData) <span>▼</span>
devicename	DeviceName (AnonymousData) <span>▼</span>

It is also possible to distribute the stream to multiple destination connections at once. To do so, click the plus icon to add an extra destination connection.



**Destination Connection**

Database Destination Connection (Database) ➤ ☰ +

When you have completed the mapping of the stream variables to the destination connection(s), click "Save & Proceed".

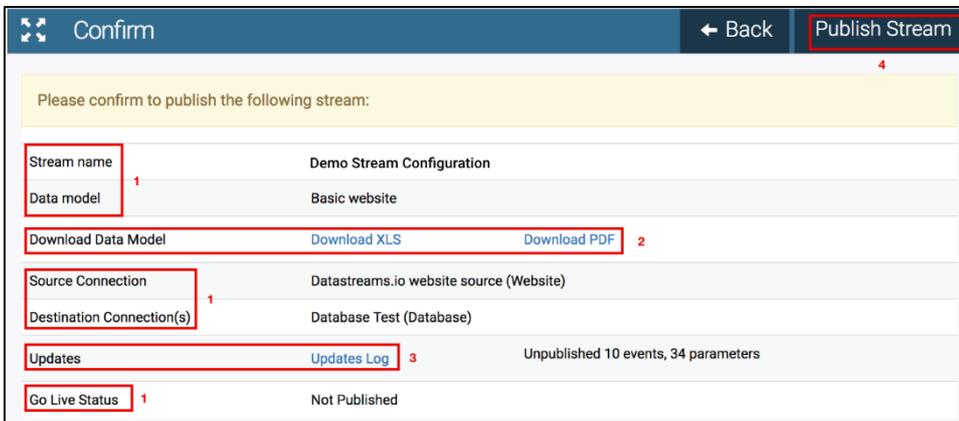
For Data Officers:

Based on the privacy level settings for the parameters and the destination connection, when mapping the stream variables, you can only see a list of parameters whose privacy levels are not higher than the destination connection's privacy level. Parameters with higher privacy levels are filtered out and will not appear in the list.

If a user with a streamer role mapped all parameters to a destination connection, and later on you (as a data officer) change the privacy level settings for the parameters and the destination connection, then the mapping field with the filtered out parameters will be automatically left empty.

## Confirm and publish the data stream

After completing the destination configuration, you will be directed to the stream configuration overview page as below.



Confirm		← Back	Publish Stream
Please confirm to publish the following stream:			
Stream name	Demo Stream Configuration		
Data model	Basic website		
Download Data Model	Download XLS	Download PDF	
Source Connection	Datastreams.io website source (Website)		
Destination Connection(s)	Database Test (Database)		
Updates	Updates Log	Unpublished 10 events, 34 parameters	
Go Live Status	Not Published		

**Stream information (1):** Here you can find information about the stream, such as stream name, data model, source/destination connection, and if it's already live.

**Download XLS & PDF (2):** You can also download the data model for the stream configuration.

**Updates Log (3):** By clicking on **Update Log** you can see all the changes that were done through the interface of the DSM. You will find two tabs in this section: Unpublished and Published. See below.



Updates log	Proceed
Unpublished	Published
Parameter <b>EventName</b> was added on 2016-12-16 10:17 by Super Admin	
Parameter <b>TimeOfEvent</b> was added on 2016-12-16 10:17 by Super Admin	

In Unpublished section you will find the information about the changes that were done but not yet published. This means that the changes made are not yet live. In the Published section you will be able to check all the changes made and that are live for your stream at the moment.

Very important to mention is that in terms of **governance** you can see here who made what changes and at what time. These can be useful for sharing any important context information about the stream within teams.

**Publish Stream (4):** You can publish your stream by clicking "Publish Stream".

Note: Only users with the Data Officer role assigned can place the new stream live. If you are not a Data Officer, clicking the confirm button will place an invite the data officer in your company to publish the stream. The creator of the stream will receive an Email confirmation once the data officer has published the stream to go live.

# Datastreams.io support

Your company has named users to be the point of contact between the Datastreams.io support team and any end users of the software. Be sure to contact a named user before reaching out to Datastreams.io support with any questions or issues.

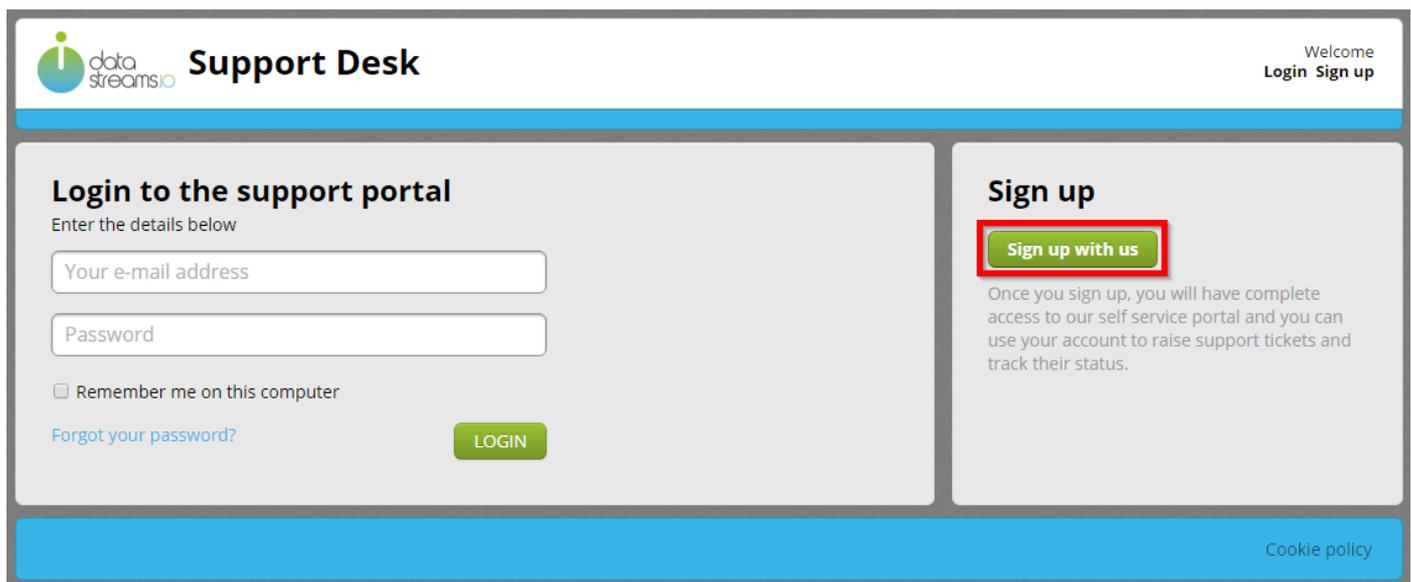
With this as a side note, Datastreams.io support is ready to support you further if a question or issue cannot be resolved directly. Of course we are always looking to continuously improve so we welcome all user input!

In this section, you will find the information about our product support centre. Included is the information on how you can contact us for support and what information you need to provide to open a service request.

Support requests can be created through our Support Desk: [Datastreams.io Support Desk](#)

## Create an account:

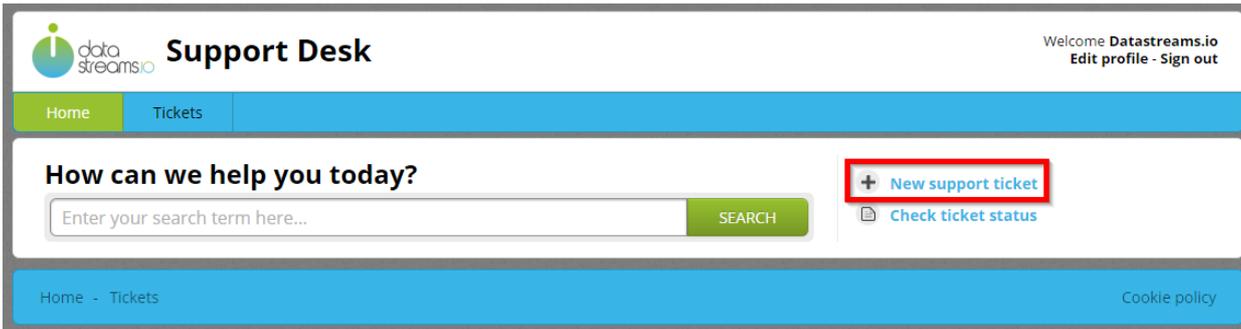
In order to send us a request you need to first register to our Support Desk: <http://support.datastreams.io>



The screenshot shows the 'Support Desk' interface. At the top left is the Datastreams.io logo and the text 'Support Desk'. At the top right, it says 'Welcome Login Sign up'. The main content area is divided into two columns. The left column is titled 'Login to the support portal' and contains the text 'Enter the details below'. It features two input fields: 'Your e-mail address' and 'Password'. Below these is a checkbox labeled 'Remember me on this computer' and a link 'Forgot your password?'. A green 'LOGIN' button is positioned at the bottom right of this section. The right column is titled 'Sign up' and features a green button labeled 'Sign up with us' which is highlighted with a red border. Below the button, it states: 'Once you sign up, you will have complete access to our self service portal and you can use your account to raise support tickets and track their status.' At the bottom right of the page, there is a link for 'Cookie policy'.

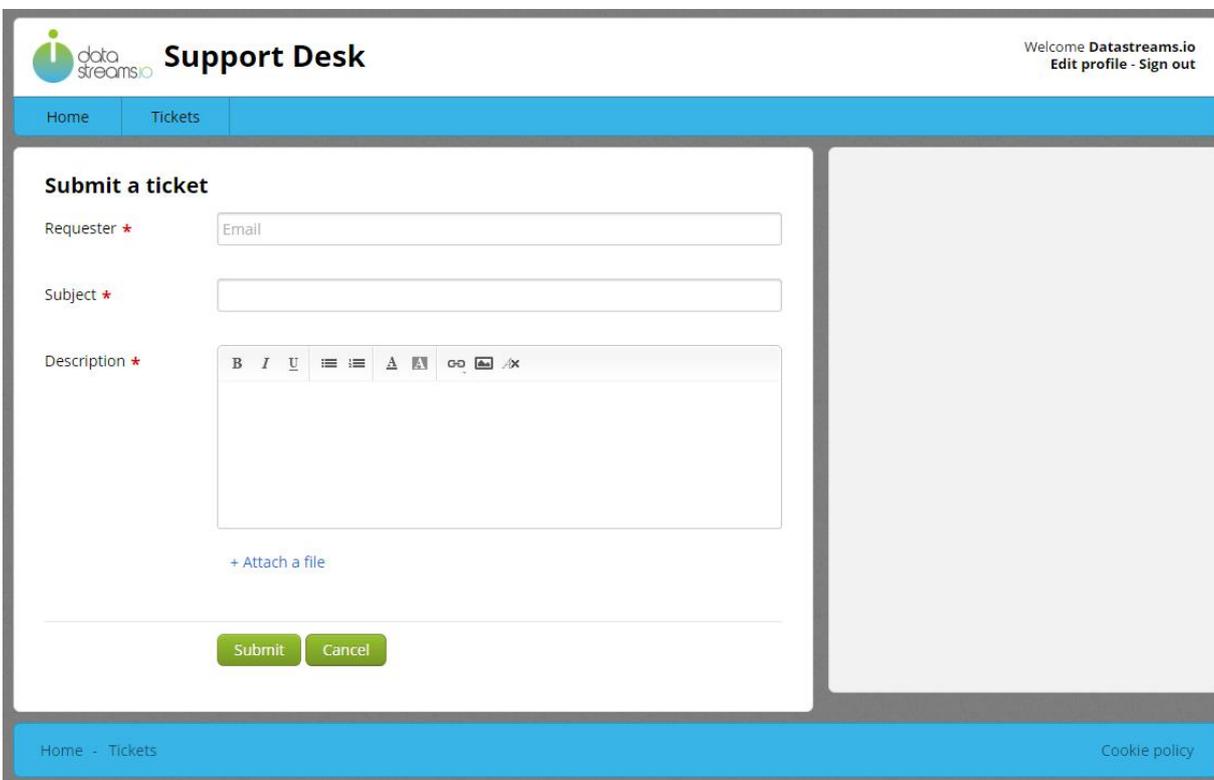
## Login into the Datastreams.io Support Desk

After you sign up, you will receive a confirmation email with a URL to activate your account and select a password. To login to the Datastreams.io Support Desk visit <http://support.datastreams.io> and login (see above image, left side) using the credentials established in the previous steps. The following screen will appear:



## Create a new Support Ticket

When logged into the Datastreams.io Support Desk you can add a support ticket by clicking on **New support ticket** – see image above, red circled button. A 'Submit a ticket' page will open:



Please fill out the fields as following:

Requester	The Datastreams.io Support Desk will automatically fill out your email address as configured in your profile.
Subject	Enter a short description of your request/issue.
Description	Describe your request/issue in more details.
Attach a file	You can add files to the ticket when you think the information will be useful for the resolution of the ticket.

When all fields are filled correctly you can click the submit button. Now your ticket has been created and one of our agents will handle your request.

### **Send in support ticket by email**

Support requests can also be created by E-mail. However, you need to create an account before you can send in support emails requests.

Your message will be converted to a new ticket and will be assigned to one of the Support Desk agents.

The support email address that should be used is: [support@datastreams.io](mailto:support@datastreams.io).

In case of an emergency please write 'High Priority:' at the beginning of the email title/subject line, followed by the rest of name you want to give to the email and include your phone number in the email itself.

You can reach our support desk by the default Email address. Note that your company may also have a dedicated Datastreams support address in use. Contact your company admin for details if this is the case.

**Email:** [support@datastreams.io](mailto:support@datastreams.io)

# Appendix 1: Data capturing methods

Additional technical info on the use of data capturing methods.

## Parameters capturing methods

### JAVASCRIPT, GROOVY AND DIMML:

The definitions will be executed (on global page level) whenever an interaction takes place. Based on the tag from which you would like to capture information, specific code is required to access the parameters in the tag. We refer to the documentation of the tags to find out which code is required to access a specific parameter of a tag. Also note that since any JavaScript is allowed, additional (not tag related) definitions are possible for the value of a parameter.

### CONTENT OF URL PARAMETER:

The Content of URL Parameters is a convenience JavaScript function. It can also be called in the console, with the function: `dimml.func.gup('UrlParameterName')`

If you have a URL, <http://www.domain.com/?dimml=LiveTestDimmlValue>, and have the value "dimml" in the definition field and [Content of URL Parameter] in the capture method, the value of this parameter will be LiveTestDimmlValue. Here either a double quote " (e.g. "value") or a single ' quote (e.g. 'value') needs to be around the value.

### FORM NAME:

The capture method Form Name, will get the value of the form item, with the name given in the definition. The name of the form item, is defined in the HTML of the page, and may not be the title of the form item that one can see on the front end.

The groovy or DimML code that runs in the DimML background for this is the following line:

```
`FormFieldValues.findAll(/DefinitionFilledInDataStreamsReplaceMe=([^\,]+)/){ it[1] }.toString()`@@groovy
```

### CONTENT OF LINK:

The Content of Link capture method, is also a convenience JavaScript function. It takes all links that have the given value in their address and combines them. If there are a number of links on the page, pointing to /news/we-make-a-profit, /blog/why-we-made-profit, /news/new-product, /blog/news/development-of-product, and we put in the definition of a blog, then the resulting value will be /blog/why-we-made-profit,/blog/news/development-of-product. If on the other hand we put news, the resulting value will be /news/we-make-a-profit, /news/new-product,/blog/news/development-of-product since the value of news was in those links. The value itself, unlike the previous one does not need any quotes around it.

The function itself is as following:

```
dimml.$('a[href*="DefinitionFilledInDataStreamsReplaceMe"]').map(function() {return  
dimml.$(this).attr("href");}).get().join(',')
```

The Groovy or DimML code that runs in the DimML background for this is the following line:

```
`FormFieldValues.findAll(/DefinitionFilledInDataStreamsReplaceMe=([^\,]+)/){ it[1] }.toString()`@@groovy
```

## GOOGLE DATA OBJECT

# Event capturing method details

## ELEMENT IS

This function takes the location from the nearest unique HTML id to determine what element to measure. You can use this option typically when you want to measure a specific action on a specific webpage. Or when you are sure all webpages where you want to measure this, have exactly the same page structure around the element you like to measure.

## URL CONTAINS

Any 'URL contains' event is triggered whenever a page with a specific string in the URL has loaded completely. By default, the page view event is measured, which is triggered whenever any page has fully loaded (DOM ready). Note that for some websites it may take a while to load completely. You can see if a page is still loading by looking at the top left icon in your browser. If the loading icon is shown, the page is still loading. If a fixed image is shown, the site has loaded (and data will be available in the overlay).

To define a 'URL contains' event, add a name for the event and then select the string 'URL contains' in the Type dropdown. In the definition put in front of the substring you want to use to trigger the event "/". Click the plus sign (this is required after every event definition). Click Reload to save the configuration.

## LINK NAME CONTAINS

This function uses the contents of a link name to measure. You can use this option typically when you want to measure every link click on any webpage with certain criteria.

## FORM IS

This function uses any form submit from the nearest unique HTML id to measure. You can use this option typically when you want to measure a specific form on a specific webpage. Or when you are sure all webpages where you want to measure, have exactly the same page structure around the form you like to measure.

## FORM NAME IS

This function uses the name of the form to measure. You can use this option typically when you want to measure form submits on specific forms that exist on differently structured webpages. It is a more robust option than 'from is' to measure a form across multiple webpages.

## Appendix 2: Table configuration examples

Additional information on the correct configuration of database tables.

### Case 1: Storing website visitor click stream data straight into a table

If the requirement is to store unprocessed, unaggregated data straight into a table, little table configuration is required.

Example of results in the table:

Timestep	Eventname	Pageurl	Pageview	Visit	Visitor id
20160425	pageview	/home	1	1	3746355274
20160425	pageview	/contact	1	0	4789273098

#### Setting the primary key

You do not need to set a primary key for your table. As each event must be recorded as is.

#### Setting the update behaviour

You do not need to set any update behaviour for your table. As each event must be recorded as is.

### Case 2: Storing website visitor summary data into a table

If the requirement is to fill a table with aggregated values, some configurations are crucial to ensure the quality of data.

Example of results in the table:

Timestep	Pageviews	Visits	Orders
20160425	84656	32945	430
20160426	56749	12435	345

#### Setting the primary key

Set the primary key to the parameter you wish to aggregate data to. This can be multiple parameters if required. In this example the primary key was set to 'Timestep'.

### Setting the update behaviour

Set the update behaviour to 'on update add' for each event in your data stream. So that all events in the table are added up correctly. In this example the update behaviour was set to 'on update add' for Pageviews, Visits and Orders.

### Case 3: Storing Twitter data straight into a table

If the requirement is to store unprocessed, unaggregated Twitter data straight into a table, a few configurations are advised. Duplicate entries may occur when multiple API calls within 24 hours are made to Twitter, since it will collect and store all Tweets in the past 24 hours every API call. Making the below settings will store all Tweets found only one time in the table, without any duplicates.

Example of results in the table:

Timeofevent	Tweetdate	Tweetid	Tweet	Keyword	Polarity
2016-08-04	Wed Aug 03	4789273098	1	Shoe	Neutral
2016-08-04	Wed Aug 03	4789273098	1	Sweater	Positive

### Setting the primary key

Set your primary key of the table only on the Tweetid column.

### Setting the update behaviour

You do not need to set any update behaviour for your table. Since you are not aggregating any events.

### Case 4: Storing Facebook page data straight into a table

If the requirement is to store unprocessed, unaggregated Facebook page data straight into a table, a few configurations are advised. Duplicate entries may occur when multiple API calls within 24 hours are made to Facebook, since it will collect and store all Facebook data in the past 24 hours with every API call. Making the below settings will store all Facebook page data only one time in the table, without any duplicates.

Example of results in the table:

Timeofevent	Post id	Postname	Postmessage	Likes	Talkingabout
-------------	---------	----------	-------------	-------	--------------

2016-08-04	4628023559	New update	Today the...	33	64
2016-08-04	4548873458	Anniversary	We are here...	24	54

### Setting the primary key

We suggest for you to set your primary key of the table on the date column, the post id column and the comment id column. In this way you can prevent duplicate rows on date, post or comments; which have been stored already for that specific day.

Note: The same post or comment will be collected on different days. This is correct behaviour. Setting the primary keys as previously specified will prevent duplicate entries.

### Setting the update behaviour

You do not need to set any update behaviour for your table. Since you are not aggregating any events.

# Appendix 3: Default data models

Additional information on the default data models and the standard variable descriptions.

## The Basic / Advanced Website data model:

Variable name	Description	Basic	Advanced
EventName	The type of interaction of a visitor on the website. This can be a pageview, pageclick or any other type of event.	√	√
TimeOfEvent	Date and timestamp of a measurement. This is measured at the server.	√	√
WebsiteDomain	The domain that was entered as the Source Connection in the Data Stream Manager	√	√
InternalSearchTerm	The term entered in the search bar of the website.	√	√
LinkName	The URL of the link that was clicked by a visitor. The LinkName variable is only set when a pageclick occurs.	√	√
PageId	Path of the page URL; characters following the domain of the website.	√	√
PageIdPath	The sequence of page IDs that occurred in a visit. This is a comma separated list.	√	√
PageTitle	The title of the page as is visible in Search engines and in the top bar of the browser when viewing the page. Everything between between the <title> tags is captured.	√	√
PageURL	The URL of the page as is visible in the address bar of the browser when viewing the page. The full URL is captured including the root domain.	√	√
SessionId	The unique id of a visit to the website.	√	√
TimeStep	The TimeOfEvent rounded to whole hours.	√	√
MarketingChannel	The marketing channel a visitor used to enter the website. Marketing channels are automatically defined based on the available information in the referring URL.	√	√
MarketingChannelDetails	The marketing source a visitor used to enter the website. Marketing sources are automatically identified based on the available information in the referring URL. It can be a specific referrer address, a campaign code or a search engine keyword.	√	√

ReferrerURL	The full URL of the source that referred a visitor to the website. The referring URL in the http request is captured.	√	√
ReferringDomain	The domain of the source that referred a visitor to the website. The referring URL in the http request is captured.	√	√
VisitorID	The numeric id of the cookie set in the browser of the visitor.	√	√
DeviceBrowser	The browser brand name.	√	√
DeviceLanguage	The language that is set in the browser.	√	√
DeviceType	The type of the device.	√	√
DeviceVersion	The version of the device operating system.	√	√
DeviceBrowserVersion	The version of the device operating system.		
EncryptedIp	The encrypted IP address of the visitor.	√	√
IP	The IP address of the visitor.		√
FormFieldValues	All values a visitor entered in the various fields of a form. This is a comma separated list.		√
FormFieldValuesEncrypted	All values a visitor entered in the various fields of a form. These are encrypted comma separated list.		√
FormName	The name of a form. The form name is automatically captured from the html code.		√
PreviousPageId	The PageId of the previous page viewed. Path of the page URL; characters following the domain of the website.		√
PreviousPageTitle	The title of the previous page viewed. The title of the page as visible in Search engines and in the top bar of the browser when viewing the page. Everything between the <title> tags is captured.		√
PreviousPageURL	The PageURL of the previous page viewed. The URL of the page as is visible in the address bar of the browser when viewing the page. The full URL is captured including the root domain.		√
Bounce	A Boolean value to detect if the user has visited only a single page. 1 stands for "Yes", 0 stands for "No".		

VisitDuration	The duration of a visit.		
NumberOfPageViews	Represents the number of page views.		
metric_Exits	A Boolean value to detect if the user has ended its visit. 1 stands for "Yes", 0 stands for "No".	√	√
metric_FormFieldBlur	A Boolean value to detect if the user has exited a form field with their pointer or mouse. 1 stands for "Yes", 0 stands for "No".		√
metric_FormFieldFocus	A Boolean value to detect if the user has entered a form field with their pointer or mouse. 1 stands for "Yes", 0 stands for "No".		√
metric_FormSubmit	A Boolean value to detect if a form has been submitted. 1 stands for "Yes", 0 stands for "No".		√
metric_InternalSearches	A Boolean value to detect if a search was done from the search bar of the website. 1 stands for "Yes", 0 stands for "No". An internal search is measured by capturing the internal search term(s).	√	√
metric_Landings	A Boolean value to detect if a visit started. 1 stands for "Yes", 0 stands for "No".	√	√
metric_PageClick	A Boolean value to detect if the user clicks a link or a button on the page. 1 stands for "Yes", 0 stands for "No".		√
metric_PageView	A Boolean value to detect if a page is loaded/viewed. 1 stands for "Yes", 0 stands for "No".	√	√

## The Twitter data model:

Variable name	Description
TimeOfEvent	The date the data was collected by Datastreams.
TweetFavoriteCount	The number of times a user has liked a Tweet.
TweetHashtag	All the hashtag(s) that were present in the Tweet.
TweetId	The unique identifier of a Tweet. For each single Tweet or ReTweet a new Tweet identifier is generated.
TwitterTweetPolarity	The sentiment of a Tweet. Possible values are positive, neutral, and negative. All Twitter sentiment is calculated using the Data Stream Managers' proprietary algorithm for sentiment scoring. Currently supporting the English and Dutch language.
TweetPolarityNegative	The number of Tweets with a negative sentiment.
TweetPolarityNeutral	The number of Tweets with neither a positive nor negative sentiment.
TweetPolarityPositive	The number of Tweets with a positive sentiment.
TweetRetweetedCount	The number of times a Tweet was Retweeted.
TweetRetweeted	Whether the Tweet was an original Tweet or a Retweet. Showing 0 for Tweets and 1 for Retweets.
TweetUser	The Tweet author name
TwitterTweetCreated	The date of the Tweet or Retweet.
TwitterTweetKeyword	The keyword, hashtag or username used to collect this Twitter data.
TwitterTweetMessage	The contents of a Tweet.

## The Facebook data model:

Name	Description
FacebookAbout	The general information on the owner of the page.
FacebookLikes	The number of likes this page has received.
FacebookName	The name of the page.
FacebookTalkingAboutCount	The number of likes, posts and comments of the page.
TimeOfEvent	The date the data was collected by the Data Stream Manager.
FacebookPostCreated	The date of a post published on the page.
FacebookPostId	The id of a post published on the page.
FacebookPostMessage	The contents of a post published on the page.
FacebookPostName	The name of a post published on the page.
FacebookPostShares	The number of shares of a post published on the page.
FacebookCommentCreated	The date of a comment published on the page.
FacebookCommentFromId	The id of a comment published on the page.
FacebookCommentFromName	The name of the person that published a comment on the page.
FacebookCommentId	The id of a comment published on the page.
FacebookCommentLikes	The number of likes for a comment on the page.
FacebookCommentMessage	The contents of a comment published on the page.

# Appendix 4: User roles and rights

Additional information on the different user roles and rights assigned to the roles.

	Viewer	Streamer	Developer	Data officer	Admin	Super admin
View stream	+	+	+	+	+	+
Create stream		+	+	+	+	+
View connection	+	+	+	+	+	+
Create connection		+	+	+	+	+
Add user			+	+	+	+
Toggle user roles viewer, streamer, admin				+	+	+
Toggle user role dataofficer					+	+
Toggle user role superadmin						+
Delete user				+	+	+
Approve publication of a stream				+		
View and select a company profile						+
Can enter Groovy, DimML & JavaScript			+			+