How to use dartfish.tv API in Power BI

Dartfish SA, 7 Mai 2020, MM + FR, v1.8

This document describes how to integrate dartfish.tv API in Power BI for allowing:

- to retrieve the (meta)data associated with a video directly and automatically from dartfish.tv.
- to retrieve the (meta)data associated to a full collection directly and automatically from dartfish.tv.
- to create a video data report automatically each time a new video is uploaded to dartfish.tv.

The current document is not a tutorial in any way for Power BI Tools (see "Prerequisites" section below).

Prerequisites

The user must:

- install Microsoft Power BI desktop in his/her computer
- have some knowledge of Power BI, enough to be able to create a data report with this tool.
- possess a dartfish.tv API authorization ID, that is, a key for a dartfish.tv channel where the dartfish.tv API use has been authorized.

There is no need for the user to buy a MS Power BI Pro account for publishing their reports on dartfish.tv

Workflow

Here follows the complete workflow for creating a data report for dartfish.tv using Power BI:

- the user creates a new Power BI blank report, or uses an existing one in Power BI desktop application.

- the user creates a new table inside the Power BI project by following the procedure described in the section "How to retrieve single video data with dartfish.tv API" or "How to retrieve collection data with dartfish.tv API". The newly created table uses dartfish.tv API for retrieving dartfish.tv data in a Power BI table, which is then used as an input for creating the data report.

- if the report was blank at the beginning, the user must now create the data report by using the data table created in the previous step. If the report was not blank, the user must "adapt" the table data and structure for interacting correctly with the existing data report.

- once the user feels that the data report is "OK", he can use it as a "template" for generating the data reports for the user's video or a full collection. This data report template can be uploaded on dartfish.tv and assigned to a specific collection.
  More info on https://support.dartfish.tv/support/solutions/articles/27000057760

- each time the user publishes a video in a collection, the video data report template provided by the user is used for generating the video data report automatically. The created data report is published to Microsoft Power BI server.
  Warning: The collection data reports are not updated automatically each time a video is published.

- whenever the user wants to modify a data report template, the new template must be uploaded to dartfish.tv. This new data report template will be used for all future published videos (for single video data report template) or at the next manual update of the collection report (for collection data report template).

How to retrieve single video data with dartfish.tv API

This section describes the steps for creating a table in a Power BI project in Power BI desktop application, which will retrieve a dartfish.tv video data using dartfish.tv API.

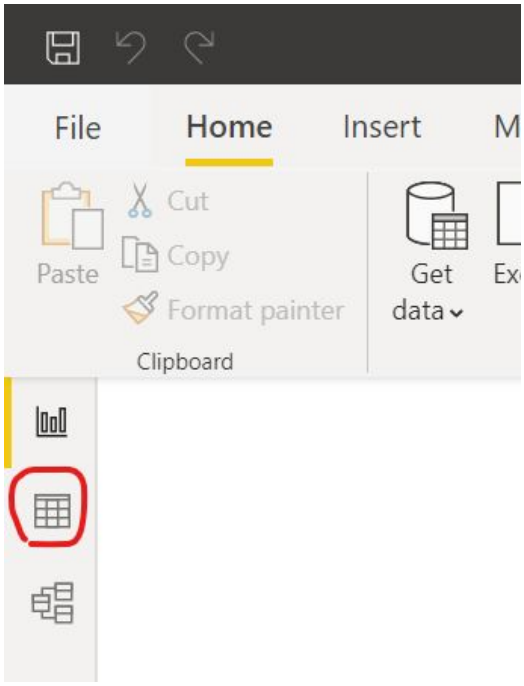
step 1 - get the collection id and match id of a currently published video belonging to the collection in which the user wants to put the data report. This video must contain the data which will be used by the data report. In other terms, the video must contain the metadata tags targeted by the data report (categories at least).

For this, open an Internet web browser, log in to dartfish.tv and play the video. Then, copy the URL displayed in the web browser and extract the collection id and match id from this URL.

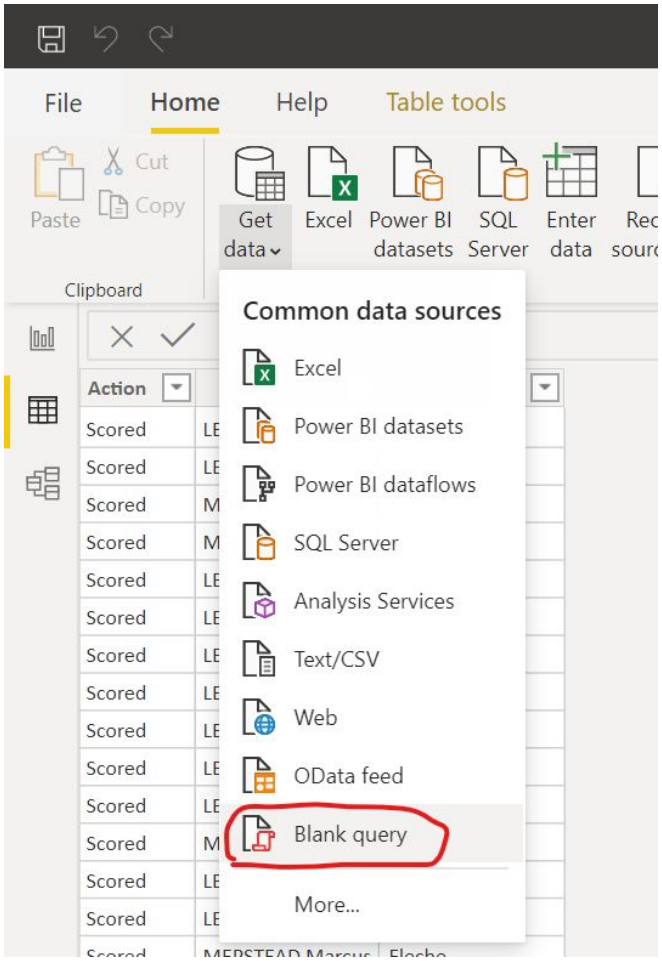
For example, if the URL is "https://www.dartfish.tv/Player?CR=p3c262173m4587016", the collection ID is **262173** and the match ID is **4587016**. These 2 ID's are necessary for the next steps.

All the remaining steps are performed in Power BI desktop application in which a new blank project has been created, or an existing project opened.

step 2 - (Power BI desktop) - select "Data" tab (on the left)

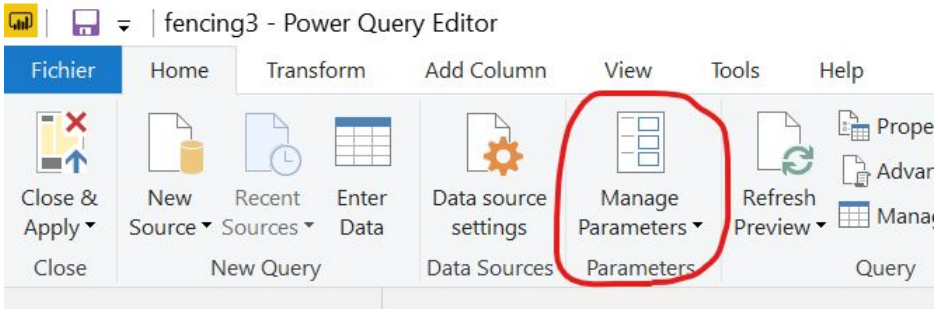


step 3 - (toolbar) "Get Data" -> "Blank Query"

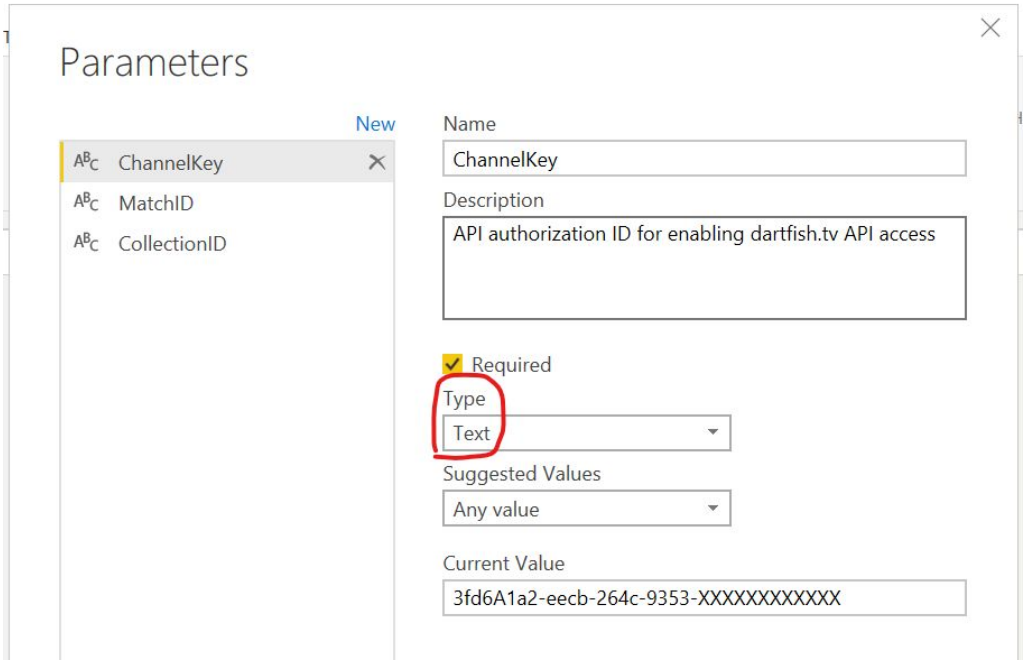


step 4 - this opens a new window.

Select the newly created query on the left pane, then (toolbar) "Manage Parameters"



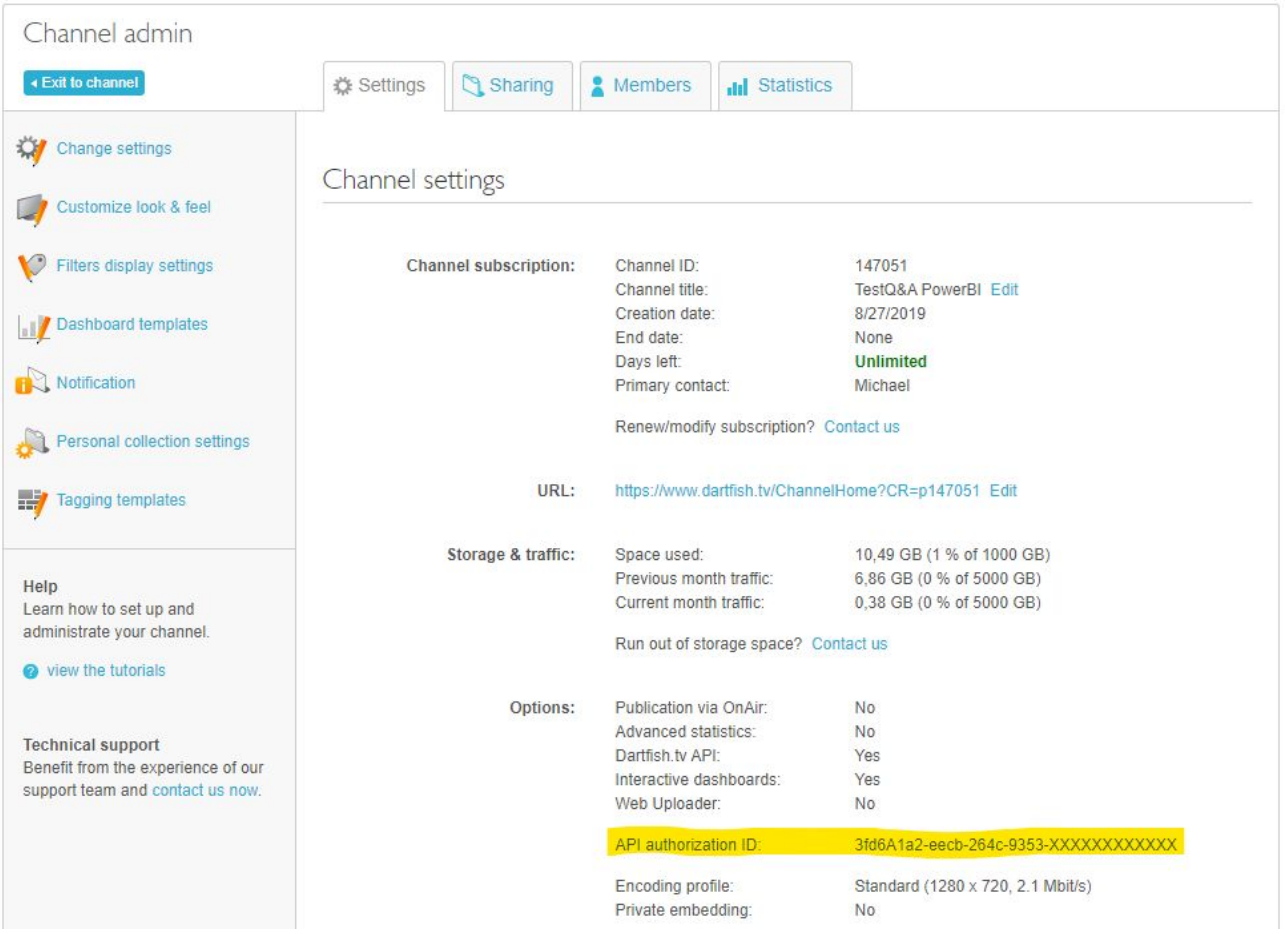
Add the **ChannelKey** parameter as described below by first clicking on "New" then filling in the different fields:



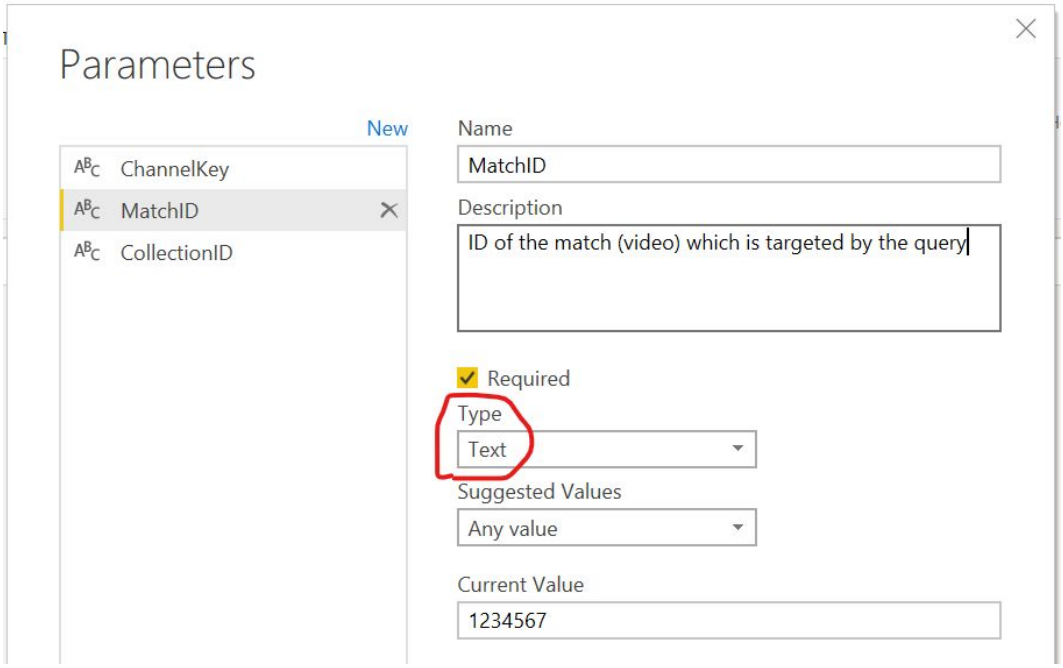
Very important: “Type” must be “Text”.

The "Current Value" must correspond to the dartfish.tv **API authorization ID** provided by Dartfish when authorizing the dartfish.tv API use on the channel.

It can be found on the homepage of the channel admin section:

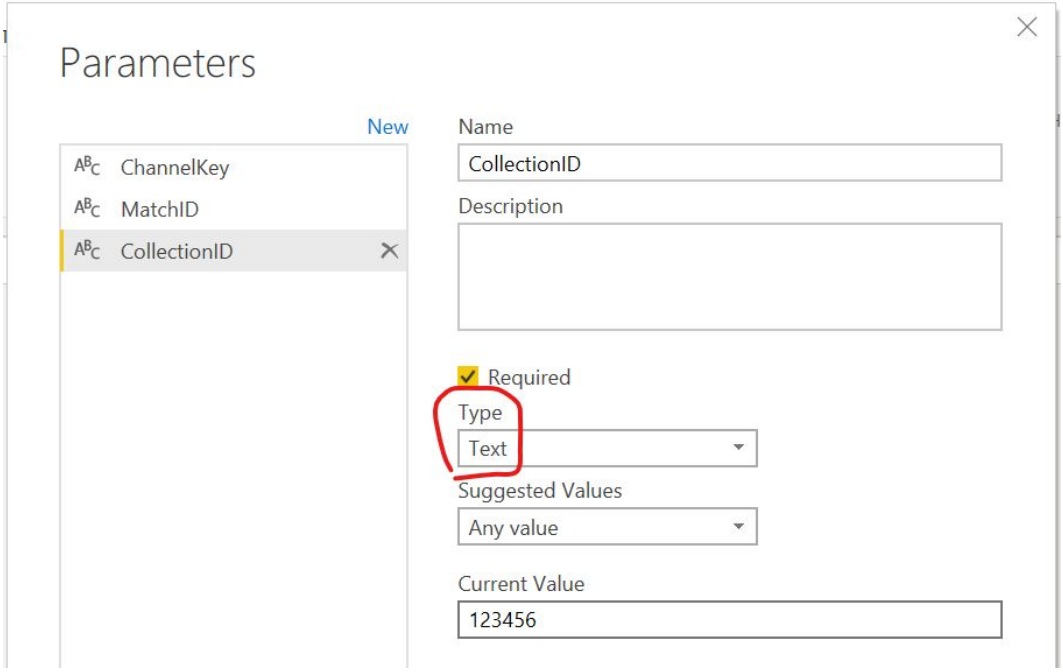


Add the **MatchId** parameter as described below:



Very important: “Type” must be “Text”.
The "current value" must correspond to the match ID (video ID) found in step 1.

And finally, add the **CollectionId** parameter as described below:

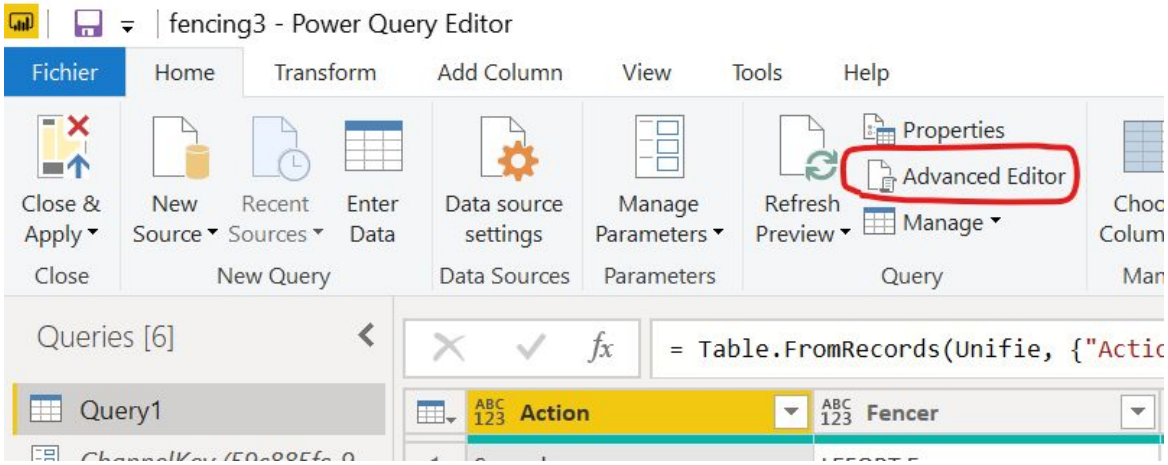


Very important: “Type” must be “Text”.
The "current value" must correspond to the collection ID as found in step 1.

The value supplied for each aforementioned parameter must correspond to your desired value and not the values shown in the figures here above (which are only examples).

step 5 - close the dialog with clicking OK.

step 6 - select the new query on the left pane and (toolbar) "Advanced Editor"



Replace all the existing program with the following piece of code:

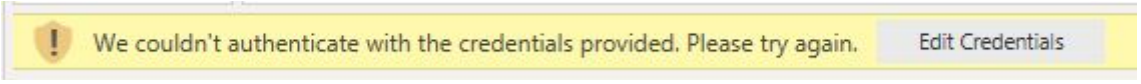
```
let
   ZeChannelKey = ChannelKey,
   ZeMatchID = MatchID,
   ZeCollectionID = CollectionID,
   // workaround with RelativePath for allowing data refresh on Power BI Services.
   Source = Json.Document(Web.Contents(
      "https://api.dartfish.tv",
      [
       Timeout=#duration(0,0,5,0),
       Headers=[Authorization="ChannelKey "&ZeChannelKey,#"Content-Type"="application/json"],
       RelativePath="/api/video/v1/"&ZeCollectionID&"/"&ZeMatchID&"/EventData",
       Content=Text.ToBinary("{""IncludeMatchKeywords"":false}")
      ]
   )),
   StripTopElem = Source[Events],
   Unifie = List.Transform(
      StripTopElem,
      each Record.TransformFields(_,{
            {"Team", Text.Trim},
            {"ActionType", Text.Trim},
            {"UniformNumber", Text.Trim}
         },
         MissingField.UseNull
      )
   ),
   Pivotage = Table.FromRecords(Unifie, {"Team", "ActionType", "UniformNumber"})
in
   Pivotage
```

This piece of code allows Power BI to retrieve the data with the dartfish.tv API in a table and process them with all the functionalities of Power BI.

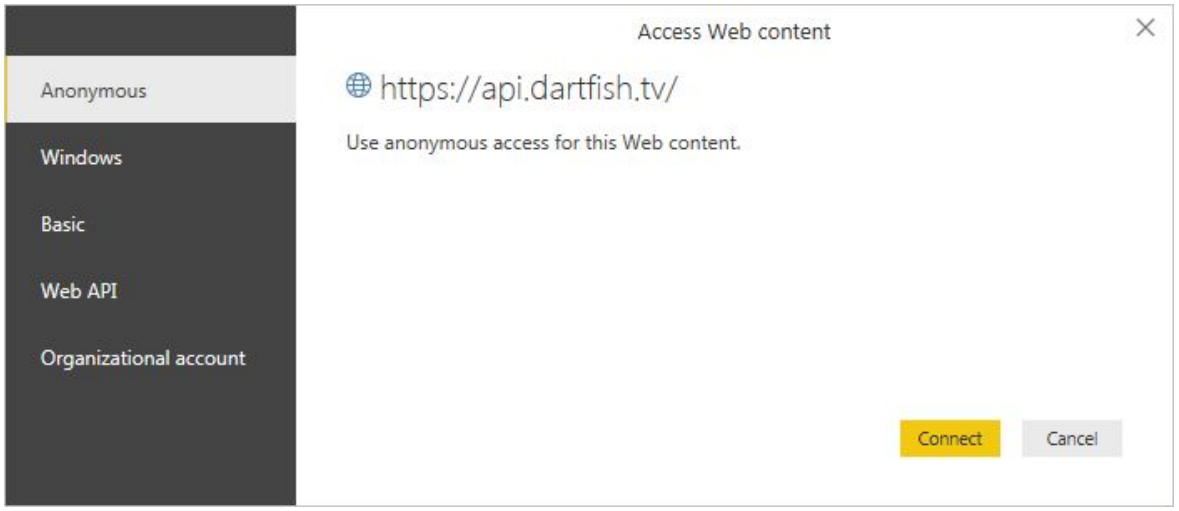
In this example, **Team**, **ActionType** and **UniformNumber** are assumed to be keyword categories. All the values are supposed to be some kind of text, so **Text.Trim** processing has been used. When the values are integers (for example), they can be transformed into genuine integers by substituting **Int64.From** for Text.Trim. In fact, any kind of processing is possible for transforming the keyword values.

step 7 - close the window by clicking the "Done" button.

At this point, you might get the following error message:



In this case, click on the **Edit Credentials** button. A new window opens:



Select **Anonymous** on the left and click the **Connect** button.

IMPORTANT:
By default and apart from the user defined keyword categories, the API always returns 3 predetermined "keywords" that are not localized, these keywords correspond to the 3 keywords defined in the Event Pane in the Dartfish Software Tagging Module. These keywords corresponds to the following definitions:

```
...{"Name", Text.Trim}, {"Position", Int64.From}, {"Duration", Int64.From}...
```

So, if we get back to the previous example where 3 categories where defined (**Team**, **ActionType** and **UniformNumber**), the full statement could also be:

```
...
   Unifie = List.Transform(
      StripTopElem,
      each Record.TransformFields(_,{
            {"Name", Text.Trim},
            {"Position", Int64.From},
            {"Duration", Int64.From},
            {"Team", Text.Trim},
            {"ActionType", Text.Trim},
            {"UniformNumber", Text.Trim}
         },
         MissingField.UseNull
      )
   ),
   Pivotage = Table.FromRecords(Unifie, {"Name", "Position", "Duration", "Team", "ActionType", "UniformNumber"})
in
   Pivotage
```

In this statement, the 3 predefined keywords are extracted (Name, Position, Duration) along with the 3 custom category keywords Team, ActionType and UniformNumber.

Hints

By modifying the **MatchID** parameter in Power BI and refreshing the Preview (by clicking on "Refresh Preview" in Power BI desktop application), it is possible to get the data from different videos. This allows checking the validity of the data report with data coming from different (published) videos.

How to retrieve collection data with dartfish.tv API

This section describes the steps for creating a table in a Power BI project in Power BI desktop application, which will retrieve the data for all the videos in a dartfish.tv collection using dartfish.tv API.

These steps are very similar to the steps to retrieve a single video data, so only the differences will be described in the next paragraphs.

step 1 - get the collection id of the collection in which the user wants to put the data report. This collection must contain at least one video containing the data which will be used by the data report. In other terms, the videos in the collection must contain the metadata tags targeted by the data report (categories at least).

For this, open an Internet web browser, log in to dartfish.tv and visit the collection. Then, copy the URL displayed in the web browser and extract the collection id from this URL.

For example, if the URL is "https://www.dartfish.tv/Videos?CR=p3c262173", the collection ID is **262173**. This ID is necessary for the next steps.

All the remaining steps are performed in Power BI desktop application in which a new blank project has been created, or an existing project opened, and are quasi identical to the steps to retrieve a single video data.

There are 2 differences:

1. At **step 4**, the MatchId parameter is not required
2. At **step 6**, the piece of code must be adapted: by modifying the "RelativePath" statement, it would be possible to invoke the "api/collection/v{version}/{collectionId}/EventData" function for getting the data for all the videos in the dartfish.tv collection. In this case, the "RelativePath" statement is modified as follows :

   ```
   RelativePath="/api/collection/v1/"&ZeCollectionID&"/EventData",
   ```

Important note

It is possible to modify and adapt the piece of code described in the previous sections for invoking other functions of the dartfish.tv API.

For more information, please check the online help of dartfish.tv API with the url: https://api.dartfish.tv/Help.

Current Limitations

The Dartfish automated data report publishing process is still under development. Here are some currently known limitations:

- the data report template for a single video will be used for all the videos published in the collection referenced by the collection id in the data report. There is currently no way to modify this parameter during the publishing process.

- whenever the user wants to modify a data report template, the new template must be uploaded to dartfish.tv. This new data report template will be used for all future published videos (for single video data report template) or at the next manual update of the collection report (for collection data report template). However, the data reports for all the documents already published on dartfish.tv are not updated automatically with this new data report template.

- only one-way filtering, from the data report to video player. The other way, from the video player to the data report, is currently not possible. This is a limitation of Power BI.

- manipulated data, that are different from the original tagged data, does not filter the video at all. Currently, there is no way for dartfish.tv to know that, and how, some data have been manipulated in Power BI from the original data and then there is no way to know how to use these manipulated data to filter the video.