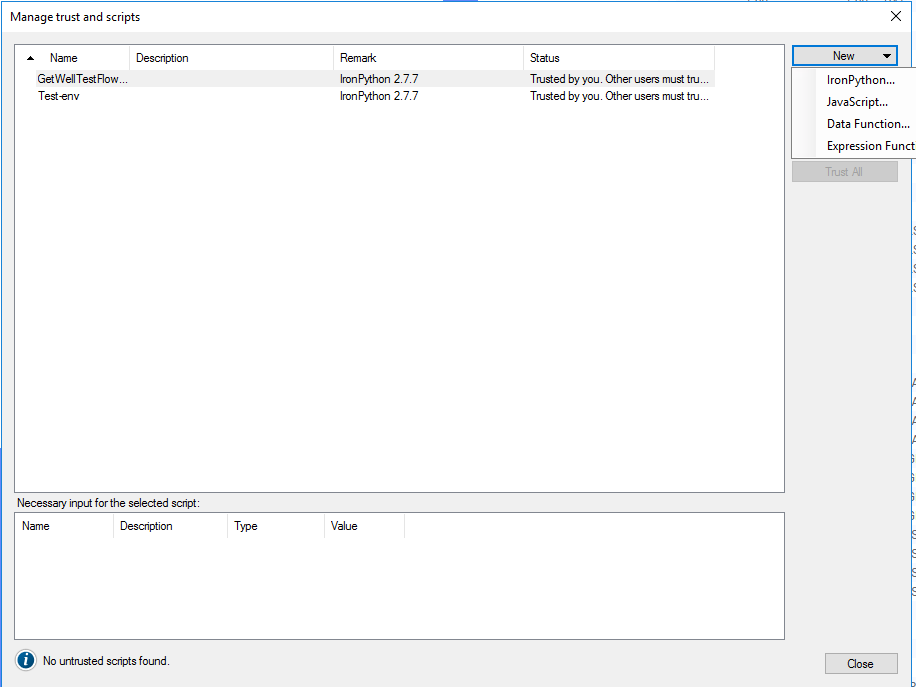
Spotfire connect REST API

1. How to use REST API in Spotfire?

We can do this using the Spotfire embedded scripts.

Start Spotfire client -> File-> Manage trust and scripts:



Useful links:

* **Using public data from a REST API in Spotfire**

[**https://community.tibco.com/feed-items/using-public-data-rest-api-spotfire**](https://community.tibco.com/feed-items/using-public-data-rest-api-spotfire)

* **IronPython Script to access data from Web Service using HttpWebRequest and parse returned JSON with JavaScriptSerializer**

[**https://community.tibco.com/wiki/ironpython-script-access-data-web-service-using-httpwebrequest-and-parse-returned-json**](https://community.tibco.com/wiki/ironpython-script-access-data-web-service-using-httpwebrequest-and-parse-returned-json)

* **Calling REST API using TIBCO Spotfire® and Advanced Data Services**

[**https://community.tibco.com/wiki/calling-rest-api-using-tibco-spotfire-and-advanced-data-services#toc-2**](https://community.tibco.com/wiki/calling-rest-api-using-tibco-spotfire-and-advanced-data-services#toc-2)

* **How to read data from a Web Service and save the data in TIBCO Spotfire® Using IronPython Scripting**

[**https://community.tibco.com/wiki/how-read-data-web-service-and-save-data-tibco-spotfirer-using-ironpython-scripting**](https://community.tibco.com/wiki/how-read-data-web-service-and-save-data-tibco-spotfirer-using-ironpython-scripting)

1. Demo code using DSPDM REST API

* **Step1 get an off-line token (python code)**
* def getOfflineToken():  
   *"""  
   get an off-line token* ***:return****: tokenType,OfflineToken  
   """* # token url  
   url = r'https://dssecurity.tiger3.dazlmkengdev01.landmarksoftware.cloud/auth/realms/DecisionSpace\_Integration\_Server/protocol/openid-connect/token'  
   data = dict()  
   data['grant\_type'] = 'password'  
   data['client\_id'] = 'dfs'  
   data['client\_secret'] = ''  
   data['username'] = 'buffalo'  
   data['password'] = '3AOti5DAcRfPwo8m'  
   data['scope'] = 'openid info offline\_access'  
   res = requests.post(url,data= data,verify=False)  
   jsonVal = json.loads(res.text)  
   offline\_token = None  
   token\_type = None  
   print(jsonVal)  
   if 'refresh\_token' in jsonVal:  
   offline\_token = jsonVal['refresh\_token']  
   if 'token\_type' in jsonVal:  
   token\_type = jsonVal['token\_type']  
   return token\_type.capitalize(), offline\_token  
    
    
  def getAccessTokenByOfflineToken(offlinetoken):  
   *"""  
   exchange an access token by offline token* ***:param*** *off-line token* ***:return****:tokenType,access token  
   """* url = r'https://dssecurity.tiger3.dazlmkengdev01.landmarksoftware.cloud/auth/realms/DecisionSpace\_Integration\_Server/protocol/openid-connect/token'  
   data = dict()  
   data['grant\_type'] = 'refresh\_token'  
   data['client\_id'] = 'dfs'  
   data['refresh\_token'] = offlinetoken  
   res = requests.post(url, data=data, verify=False)  
   jsonVal = json.loads(res.text)  
   access\_token = None  
   token\_type = None  
   print(jsonVal)  
   if 'access\_token' in jsonVal:  
   access\_token = jsonVal['access\_token']  
   if 'token\_type' in jsonVal:  
   token\_type = jsonVal['token\_type']  
   return token\_type.capitalize(), access\_token
* **Step2 Using Offline Token exchange Access Token (iron python code)**

# get access token by offline token

tokenWebRequest = HttpWebRequest.Create(token\_url)

tokenWebRequest.ContentType = 'application/x-www-form-urlencoded'

tokenWebRequest.Method = "POST";

api\_param\_str = 'grant\_type=refresh\_token&client\_id=dfs&refresh\_token={}'.format(offline\_token)

streamWriter = StreamWriter(tokenWebRequest.GetRequestStream())

streamWriter.Write(api\_param\_str)

streamWriter.Close()

streamReader = StreamReader(tokenWebRequest.GetResponse().GetResponseStream())

result = streamReader.ReadToEnd()

streamReader.Close()

jsonRes = json.loads(result)

# access token

access\_token = jsonRes['access\_token']

* **Step3 Using Access Token get data from PDM REST API (iron python code)**

pdm\_api\_param\_str = '''{"boName" : "WELL TEST FLOW",

"joinAlias" : "a",

"selectList" : ["WELL\_TEST\_FLOW\_ID", "WELL\_TEST\_ID", "UWI", "START\_TIME", "WH\_PRESS", "WATER\_CUT\_PERCENT", "GAS\_OIL\_RATIO"],

"orderBy": [{"order": "ASC", "boAttrName" : "WELL\_TEST\_FLOW\_ID"}],

"simpleJoins" : [{"joinType": "INNER",

"selectList" : ["WELL\_TYPE\_ID"],

"joinAlias" : "b",

"boName" : "WELL",

"joiningConditions" : [{"operator": "EQUALS",

"rightSide" : {"joinAlias": "b", "boAttrName" : "UWI"},

"leftSide" : {"joinAlias": "a", "boAttrName" : "UWI"}}]

}],

"language": "en", "readMetadata": false, "timezone": "GMT+08:00","readAllRecords":true }'''

pdmWebRequest = HttpWebRequest.Create(pdm\_url);

pdmWebRequest.ContentType = "application/json";

pdmWebRequest.Method = "POST";

pdmWebRequest.PreAuthenticate = True;

pdmWebRequest.Headers.Add("Authorization", "Bearer " + access\_token);

streamWriter = StreamWriter(pdmWebRequest.GetRequestStream())

streamWriter.Write(pdm\_api\_param\_str)

streamWriter.Close()

streamReader = StreamReader(pdmWebRequest.GetResponse().GetResponseStream())

result = streamReader.ReadToEnd()

streamReader.Close()

jsonRes = json.loads(result)

flowData = jsonRes['data']['WELL TEST FLOW']['list']

* **Step4 Convert API JSON Data to Spotfire Table (iron python code)**

colNames = ["WELL\_TEST\_FLOW\_ID", "WELL\_TEST\_ID", "UWI", "START\_TIME", "WH\_PRESS", "WATER\_CUT\_PERCENT", "GAS\_OIL\_RATIO","WELL\_TYPE\_ID"]

textData = "\t".join(colNames) + "\r\n"

for row in flowData:

vals = []

for col in colNames:

if col in row:

vals.append(row[col])

else:

vals.append(None)

textData += "\t".join(str(val) for val in vals) + "\r\n"

settings = TextDataReaderSettings()

settings.Separator = "\t"

settings.AddColumnNameRow(0)

settings.SetDataType(0, DataType.Real)

settings.SetDataType(1, DataType.Real)

settings.SetDataType(2, DataType.String)

settings.SetDataType(3, DataType.DateTime)

settings.SetDataType(4, DataType.Currency)

settings.SetDataType(5, DataType.Currency)

settings.SetDataType(6, DataType.Currency)

settings.SetDataType(7, DataType.Real)

stream = MemoryStream()

writer = StreamWriter(stream)

writer.Write(textData)

writer.Flush()

stream.Seek(0, SeekOrigin.Begin)

textDataSource = TextFileDataSource(stream, settings)

# add the data into a Data Table in Spotfire

if Document.Data.Tables.Contains("Flow Data"):

Document.Data.Tables["Flow Data"].ReplaceData(textDataSource)

else:

newTable = Document.Data.Tables.Add("Flow Data", textDataSource)

tableSettings = DataTableSaveSettings(newTable, False, False)

Document.Data.SaveSettings.DataTableSettings.Add(tableSettings)

1. Unit conversion

* **Please refer to a separate document called Unit Conversion**

1. The complete code of iron python in Spotfire

import json

import clr

clr.AddReference('System.Data')

clr.AddReference('System.Web.Extensions')

import System

from System import DateTime

from System.Data import DataSet, DataTable

from System.IO import StreamReader, StreamWriter, MemoryStream, SeekOrigin

from System.Net import HttpWebRequest

from Spotfire.Dxp.Data import DataType, DataTableSaveSettings

from Spotfire.Dxp.Data.Import import TextFileDataSource, TextDataReaderSettings

# token url,pdm url,offline token

# configure these variables

token\_url = r'https://dssecurity.tiger3.dazlmkengdev01.landmarksoftware.cloud/auth/realms/DecisionSpace\_Integration\_Server/protocol/openid-connect/token'

pdm\_url = r'https://tiger3.dazlmkengdev01.landmarksoftware.cloud/services/dev-dspdmservice-dfs/msp/secure/common'

offline\_token = 'eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICI4YTM5OTNhOC1kZGY5LTQ0OGUtOTAxNi0xZmE1NGNiMjAxY2UifQ..jnUqzeJf8ShS6O7fh5eAlAxObXjWQmeNHWZmC69Jn2s'

# get access token by offline token

tokenWebRequest = HttpWebRequest.Create(token\_url)

tokenWebRequest.ContentType = 'application/x-www-form-urlencoded'

tokenWebRequest.Method = "POST";

api\_param\_str = 'grant\_type=refresh\_token&client\_id=dfs&refresh\_token={}'.format(offline\_token)

streamWriter = StreamWriter(tokenWebRequest.GetRequestStream())

streamWriter.Write(api\_param\_str)

streamWriter.Close()

streamReader = StreamReader(tokenWebRequest.GetResponse().GetResponseStream())

result = streamReader.ReadToEnd()

streamReader.Close()

jsonRes = json.loads(result)

# access token

access\_token = jsonRes['access\_token']

#print(access\_token)

# pdm common api parameters

pdm\_api\_param\_str = '''

{

"boName" : "WELL TEST FLOW",

"joinAlias" : "a",

"selectList" : ["WELL\_TEST\_FLOW\_ID", "WELL\_TEST\_ID", "UWI", "START\_TIME", "WH\_PRESS", "WATER\_CUT\_PERCENT", "GAS\_OIL\_RATIO"],

"orderBy": [{"order": "ASC", "boAttrName" : "WELL\_TEST\_FLOW\_ID"}],

"simpleJoins" : [{"joinType": "INNER",

"selectList" : ["WELL\_TYPE\_ID"],

"joinAlias" : "b",

"boName" : "WELL",

"joiningConditions" : [{"operator": "EQUALS",

"rightSide" : {"joinAlias": "b", "boAttrName" : "UWI"},

"leftSide" : {"joinAlias": "a", "boAttrName" : "UWI"}}]

}],

"language": "en", "readMetadata": false, "timezone": "GMT+08:00","readAllRecords":true

}

'''

pdmWebRequest = HttpWebRequest.Create(pdm\_url);

pdmWebRequest.ContentType = "application/json";

pdmWebRequest.Method = "POST";

pdmWebRequest.PreAuthenticate = True;

pdmWebRequest.Headers.Add("Authorization", "Bearer " + access\_token);

streamWriter = StreamWriter(pdmWebRequest.GetRequestStream())

streamWriter.Write(pdm\_api\_param\_str)

streamWriter.Close()

streamReader = StreamReader(pdmWebRequest.GetResponse().GetResponseStream())

result = streamReader.ReadToEnd()

streamReader.Close()

jsonRes = json.loads(result)

flowData = jsonRes['data']['WELL TEST FLOW']['list']

#print(flowData)

colNames = ["WELL\_TEST\_FLOW\_ID", "WELL\_TEST\_ID", "UWI", "START\_TIME", "WH\_PRESS", "WATER\_CUT\_PERCENT", "GAS\_OIL\_RATIO","WELL\_TYPE\_ID"]

textData = "\t".join(colNames) + "\r\n"

for row in flowData:

vals = []

for col in colNames:

if col in row:

vals.append(row[col])

else:

vals.append(None)

textData += "\t".join(str(val) for val in vals) + "\r\n"

print(textData)

settings = TextDataReaderSettings()

settings.Separator = "\t"

settings.AddColumnNameRow(0)

settings.SetDataType(0, DataType.Real)

settings.SetDataType(1, DataType.Real)

settings.SetDataType(2, DataType.String)

settings.SetDataType(3, DataType.DateTime)

settings.SetDataType(4, DataType.Currency)

settings.SetDataType(5, DataType.Currency)

settings.SetDataType(6, DataType.Currency)

settings.SetDataType(7, DataType.Real)

stream = MemoryStream()

writer = StreamWriter(stream)

writer.Write(textData)

writer.Flush()

stream.Seek(0, SeekOrigin.Begin)

textDataSource = TextFileDataSource(stream, settings)

# add the data into a Data Table in Spotfire

if Document.Data.Tables.Contains("Flow Data"):

Document.Data.Tables["Flow Data"].ReplaceData(textDataSource)

else:

newTable = Document.Data.Tables.Add("Flow Data", textDataSource)

tableSettings = DataTableSaveSettings(newTable, False, False)

Document.Data.SaveSettings.DataTableSettings.Add(tableSettings)