**User manual**

**USER management Tool**

**Creation and expiration of T24 Users**

**And other fun stuff:**

**Password reset  
Execute OFS commands  
Update EB.LOOKUP table  
Extract Updates documentation  
Generate Data Dictionary  
Generate sample OFSML  
Maintain EB.USER.ROLES and USER.SMS.GROUPS**

**List INAU records and authorise**

**Translate online HelpText**

**DataDownload (fill TAABS-like excel)**

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Section(s) Amended |
| 2.0 | 23-06-2016 | Willem Gorter | Initial version tested in E2E |
| 3.0 | 23-06-2016 | Willem Gorter | added Password Reset worksheet |
| 4.0 | 23-06-2016 | Willem Gorter | Password is replaced in OFS log  Column L.ID.NUM is added to USER  Password screen shows environment/Funct.  Environment is shown with name |
| 4.2 | 26-06-2016 | Willem Gorter | Bugfix: range delete shifts environment list  Bugfix: delete range not correctly defined  Reserved account numbers worksheet added |
| 4.3 | 29-06-2016 | Willem Gorter | Added setting homepage for users with profile |
|  |  |  |  |
| 5.6 | 05-10-2016 | Willem Gorter | Added migration tool |
| 5.7 | 06-10-2016 | Willem Gorter | Added Updates documentation |
| 5.8 | 26-10-2016 | Willem Gorter | Added Datadictionary |
| 5.9 | 29-10-2016 | Willem Gorter | Added sample OFSML generation |
| 6.0 | 02-11-2016 | Willem Gorter | Added EB.LOOKUP update |
| 6.1 | 15-11-2016 | Willem Gorter | Amend DataDictionary to retrieve and update HelpText |
| 6.1.1 | 23-11-2016 | Willem Gorter | Add HelpText for Local applications |
| 6.1.2 | 27-11-2016 | Willem Gorter | Generate OFSML for Version with no fields |
| 6.1.3 | 6-12-2016 | Willem Gorter | Fix OFSML for fields with '\*' and ending in '-1.1'  EB.LOOKUP- works for BLM.KYC.ANSWER |
| 6.2 | 6-12-2016 | Willem Gorter | Migration: create OFS for routing table |
| 6.3 | 02-01-2017 | Willem Gorter | Updates documentation: Added GetList from SPF |
| 6.7 | 27-09-2017 | Willem Gorter | Added: Update EB.USER.ROLE and USER.SMS.GROUP's functionality |
| 7.0 | 27-11-2020 | Willem Gorter | Removing Leumi-specific stuff |
| 7.1 | 6-12-2020 | Willem Gorter | Added: creation and translation of helptext, including automatic translation with MS Translator |
| 7.2 | 10-12-2020 | Willem Gorter | Added: retrieval and authorisation of INAU records |
| 7.2.1 | 3-3-2021 | Willem Gorter | Finished translation of helptext |
| 7.3 | 4-3-2021 | Willem Gorter | Data Download |
| 7.3.10 | 11-3-2021 | Willem Gorter | Data Download for a list of applications in a new WorkBook |
| 7.3.20 | 22-3-2021 | Willem Gorter | Automatic translation with Google Translate |
| 7.4 | 26-3-2021 | Willem Gorter | Saving configuration 3.3 |

Document Sign Off

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Name | Company / Position |
|  |  |  |  |

Key Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Role | email | Phone |
| Willem Gorter | RSA | ewgorter@temenos.com | +45 24343891 |
|  |  |  |  |

Table of Contents

[1 Purpose 6](#_Toc67663890)

[2 Technical background 6](#_Toc67663891)

[2.1 TAFJ until R17 (this paragraph is now out of date) 6](#_Toc67663892)

[2.2 TAFJ as of R17 6](#_Toc67663893)

[2.2.1 TAFJ Application User 7](#_Toc67663894)

[3 Common features 8](#_Toc67663895)

[3.1 General configuration 8](#_Toc67663896)

[3.1.1 environment 8](#_Toc67663897)

[3.1.2 T24 User 8](#_Toc67663898)

[3.1.3 Version 9](#_Toc67663899)

[3.1.4 Function 9](#_Toc67663900)

[3.2 Execution of the OFS 9](#_Toc67663901)

[3.3 Saving and loading the configuration 9](#_Toc67663902)

[3.3.1 Saving the configuration 9](#_Toc67663903)

[3.3.2 Loading the configuration 10](#_Toc67663904)

[3.3.3 Delete private config 10](#_Toc67663905)

[4 Field List 10](#_Toc67663906)

[5 USERS 10](#_Toc67663907)

[5.1 Creating Users 10](#_Toc67663908)

[5.1.1 ID 11](#_Toc67663909)

[5.1.2 USER.NAME 11](#_Toc67663910)

[5.1.3 SIGN.ON.NAME 11](#_Toc67663911)

[5.1.4 LANGUAGE 11](#_Toc67663912)

[5.1.5 Profile 11](#_Toc67663913)

[5.1.6 SuperUser 11](#_Toc67663914)

[5.1.7 Expiration date 11](#_Toc67663915)

[5.1.8 ID num 11](#_Toc67663916)

[5.1.9 OFS 11](#_Toc67663917)

[5.1.10 RESULT 12](#_Toc67663918)

[5.2 Expiring Users 12](#_Toc67663919)

[6 Reset Passwords 12](#_Toc67663920)

[7 ExecuteOFS 13](#_Toc67663921)

[7.1 Construction of the OFS 13](#_Toc67663922)

[7.2 Executing the OFS 13](#_Toc67663923)

[8 EB.LOOKUP 13](#_Toc67663924)

[9 Updates documentation 14](#_Toc67663925)

[9.1 Get List of updates 14](#_Toc67663926)

[9.2 Get documentation of updates 14](#_Toc67663927)

[10 DataDictionary 15](#_Toc67663928)

[10.1 Entering tables manually 15](#_Toc67663929)

[10.2 Retrieving tables from FILE.CONTROL 15](#_Toc67663930)

[10.2.1 Selection of FileControl records 15](#_Toc67663931)

[10.2.2 Cleaning the list 16](#_Toc67663932)

[10.2.3 Configuring HelpText location 16](#_Toc67663933)

[10.3 Generating DataDictionary 16](#_Toc67663934)

[10.3.1 Index of the Datadictionary 16](#_Toc67663935)

[10.3.2 The DataDictionary sheet 17](#_Toc67663936)

[10.4 Updating HelpText 18](#_Toc67663937)

[11 Translating online helptext 19](#_Toc67663938)

[11.1 Setting up helptext creation/update and translation 20](#_Toc67663939)

[11.1.1 Set the paths of the helptext 20](#_Toc67663940)

[11.1.2 Activate the automatic translation 20](#_Toc67663941)

[11.2 Retrieving the helptext 22](#_Toc67663942)

[11.3 Retrieving Translated helptext 23](#_Toc67663943)

[11.4 Correcting or creating English helptext 23](#_Toc67663944)

[11.5 Translating helptext to local language 23](#_Toc67663945)

[11.6 Creating translatable helptext for multiple applications 24](#_Toc67663946)

[12 Sample OFSML generation 24](#_Toc67663947)

[13 Update EB.USER.ROLES and USER.SMS.GROUPS 25](#_Toc67663948)

[13.1 Retrieve EB.USER.ROLES 25](#_Toc67663949)

[13.2 Updating Roles 26](#_Toc67663950)

[13.3 Retrieve USG 26](#_Toc67663951)

[13.4 USER.SMS.GROUP worksheet 26](#_Toc67663952)

[13.5 Create USG's 27](#_Toc67663953)

[13.5.1 Execution line per line 27](#_Toc67663954)

[13.5.2 Fixing errors 27](#_Toc67663955)

[13.5.3 Special uses 27](#_Toc67663956)

[14 Retrieve Unauthorised records and authorise 28](#_Toc67663957)

[14.1 Creating a list of INAU records 28](#_Toc67663958)

[14.2 Authorising INAU records from the list 28](#_Toc67663959)

[15 Download T24 Data to Excel 29](#_Toc67663960)

[15.1 Create an excel with the structure of a T24 table 29](#_Toc67663961)

[15.2 Fill the excel with T24 Data 30](#_Toc67663962)

[15.3 Generating data for a list of applications 31](#_Toc67663963)

# Purpose

The purpose of this document is to provide guidelines for using the Excel workbook USER\_v3.xlsm (or later). The Excel workbook is a tool for creating or disabling T24 Users.

# Technical background

## TAFJ until R17 (this paragraph is now out of date)

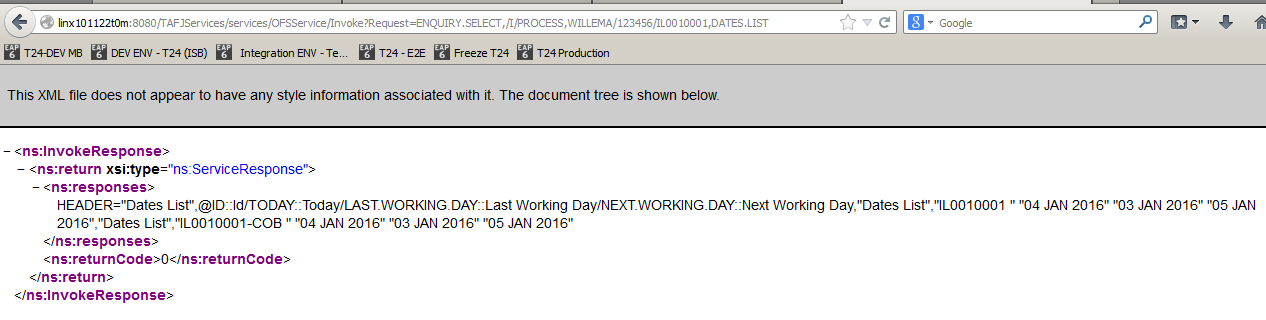
T24 browser (backoffice screens) uses OFS to communicate with T24 Server. In TAFJ, OFS can be run also directly as an HTTP-GET command from the address bar in a browser, if the address starts with:

<http://linx101122t0m:8080/TAFJServices/services/OFSService/Invoke?Request>=

(where the part starting with TAFJServices replaces the address of the Login screen starting with BrowserWeb). An example (it links to DEV server, from Leumi network you can Ctrl-click on it):

<http://linx101122t0m:8080/TAFJServices/services/OFSService/Invoke?Request=ENQUIRY.SELECT,/I/PROCESS,WILLEMA/123456/IL0010001,DATES.LIST>

will return, when pasted in Internet Explorer or Firefox:



The excel tool uses this mechanism to execute the actions. It will execute an HTTP-GET request, and read the response. The VBA code for this is:



To read the response, it will load the ResponseText property of the request, and parse it to find the OFS response. You can find the code in the Excel tool when editing macro's, in the routine LaunchOFSCommandTAFJ in module1.

## TAFJ as of R17

T24 browser (backoffice screens) uses OFS to communicate with T24 Server. In TAFJ, OFS can be run also directly as an HTTP-POST command, using Postman or any other API software, or through writing a program in any programming language, like Visual Basic for Applications, available in Excel.

The program should execute an HTTP request using (for example):

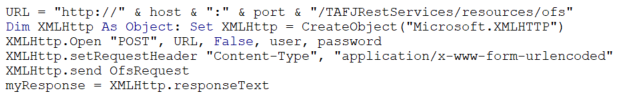
URL = <http://localhost:8089/TAFJRestServices/resources/ofs>  
Request = ofsRequest=ENQUIRY.SELECT,,INPUTT/123456,DATES.LIST  
Header = “Content-Type” = “application/x-www-form-urlencoded”

(where the URL part starting with TAFJServices replaces the address of the Login screen starting with BrowserWeb).

This will return, in the HTTP Response:

{"ofsRequest":"ENQUIRY.SELECT,,INPUTT/123456,DATES.LIST","ofsResponse":"HEADER=\"Dates List\",@ID::Id/TODAY::Today/LAST.WORKING.DAY::Last Working Day/NEXT.WORKING.DAY::Next Working Day,\"Dates List\",\"AU0010001 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"AU0010001-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"EU0010001 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"EU0010001-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010001 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010001-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010002 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010002-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010003 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010003-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010004 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010004-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010005 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"GB0010005-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"NL0010001 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"NL0010001-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0010001 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0010001-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0020100 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0020100-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0020101 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0020101-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0020102 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"SG0020102-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"US0010001 \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\",\"Dates List\",\"US0010001-COB \"\t\"17 APR 2020\"\t\"16 APR 2020\"\t\"20 APR 2020\""}

The excel tool uses this mechanism to execute the actions. It will execute an HTTP-POST request, and read the response. The VBA code for this is:



To read the response, it will load the ResponseText property of the request, and parse it to find the OFS response. You can find the code in the Excel tool when editing macro's, in the routine LaunchOFSCommandTAFJ in module1.

### TAFJ Application User

As you can see in the above example, when opening the HTTP connection, you need to pass a username and password. These are not the same as the T24 user/password that you have to provide inside the OFS command, and which is a user with which you can login to BrowserWeb, typically INPUTT/123456.

To access the Rest Services for TAFJ, you need a TAFJ application user, and that user needs to be created in the environment that you are targeting. For local modelbanks, there are scripts for managing the users in the jBoss installation (typically “add-user” script in jboss/bin). Please see the appendix, section 7.1, for a detailed explanation on how to create such a user, in short the process should be to get it working on a UTP installation in Windows:

1-     Go to the directory /jboss/bin

2-     Run the command “add-user.bat”

3-     Select “b”

4-     Type username

5-     Type password

6-     In groups type “t24user,TAFJAdmin” in exact case

7-     When asked as a slave type no

There should be no need to restart jBoss after adding the user. If, after adding the user, you get the following response:

<html><head><title>Error</title></head><body>Unauthorized</body></html>

Then the user was not created correctly, or the T24 installation is not correct (typically, using an installer version, not a UTP-version).

The username and password need to be provided in the “installation” worksheet (see chapter 4)

# Common features

Maintaining USER profiles will be done through IDM as of September. Until that time, users can be maintained in T24 through the standard screens. The tool can replace those standard screens and will have reduced functionality compared to the T24 screens, but they enable the user to execute multiple changes in one action, where the changes have been prepared beforehand.

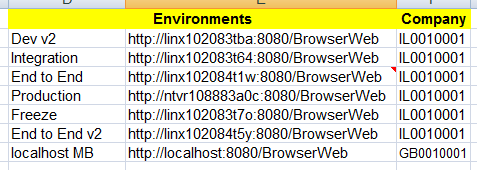
## General configuration

The first two lines of each worksheet contain the general configuration, which should be done once generally:



### environment

In cell B1, you should choose the environment, you can create your own in the parameter tab with a paste of the first part of the Browser address (Login Screen). The dropdown has some preconfigured environments (you can see the list in the Parameters worksheet)



The 3rd Column contains the Company of your user. It should always be the same for one project, but for the generic model bank for example, the company should be GB0010001.

### T24 User

You can specify the T24 user that you will be using. If you leave it blank, it will ask it when executing an activity, and store the user here. Please specify the Login Name of the user (like INPUTT)

### Version

You can specify here the version to use, this doesn't need to be changed except if an environment is missing a version.

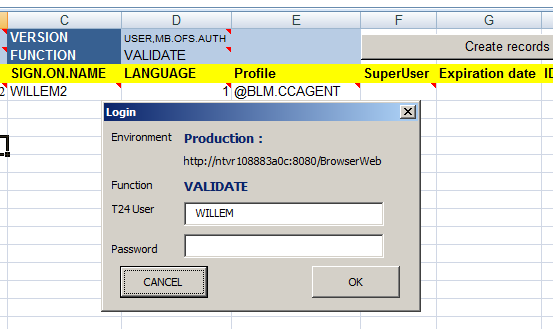
### Function

You can specify VALIDATE or PROCESS. These are OFS functions, where VALIDATE will execute the transaction, capture the result, and then rollback the transaction. It does not change anything in the target environment, and it can be used to test the correctness of the data before execution.

PROCESS will execute the changes in the target environment.

## Execution of the OFS

To execute the OFS, click on the button (Create Records or Reset PassWords).



This will pop up a Window where you need to enter the password corresponding to the user that is pre-configured (you can also change the User). The User provided here is the Login as used in the T24 login screen.

After entering the User and Password, click on OK to close the window. It will store the User (but not the password) in the Header. This will start the execution of the OFS.

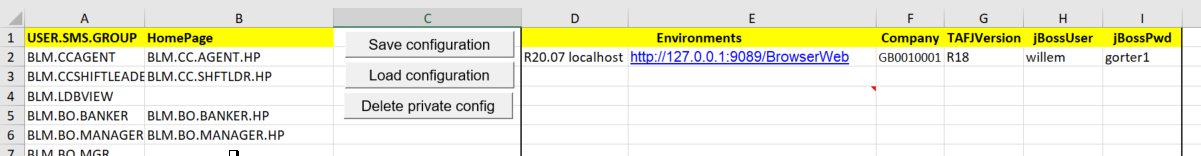
Check before clicking OK if environment and function are what you want.

## Saving and loading the configuration

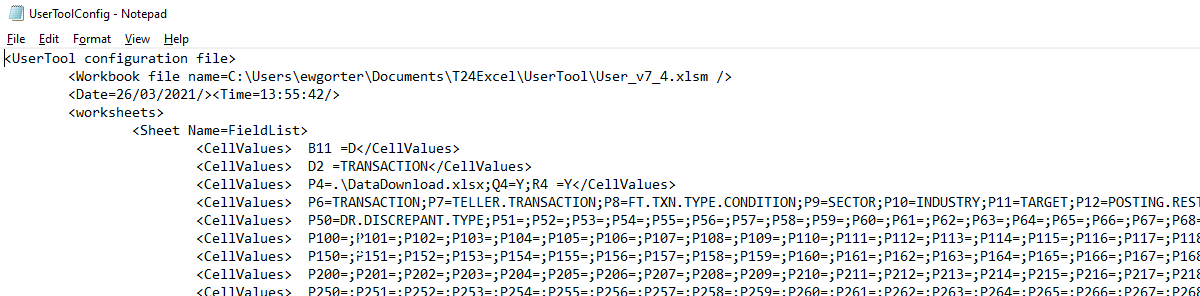
To enable users to switch to a new version of the excel, without loosing parameters proprietary to the user, it is now possible to save the configuration of this type of configurations, and to reload the same parameters afterwards. Typically, after making sure everything works, and after entering any new environment data or API keys, you should save your configuration.

### Saving the configuration

This is done from the “Parameters” worksheet, with the button “Save configuration”:



The values of certain cells will be saved to a text file “UserToolConfig.txt” in XML format in the same path as the Excel Workbook:



The list of cells is hardcoded in the excel (please refer to the VBA module “configuration”, function “GetConfiguration” where it can be changed)

In general, most cells that you can change in the “Parameters” worksheet are saved, as well as input sets in other worksheets (list of tables for which to download all data, list of tables for which to create a datadictionary, list of tables for which to translate helptext)

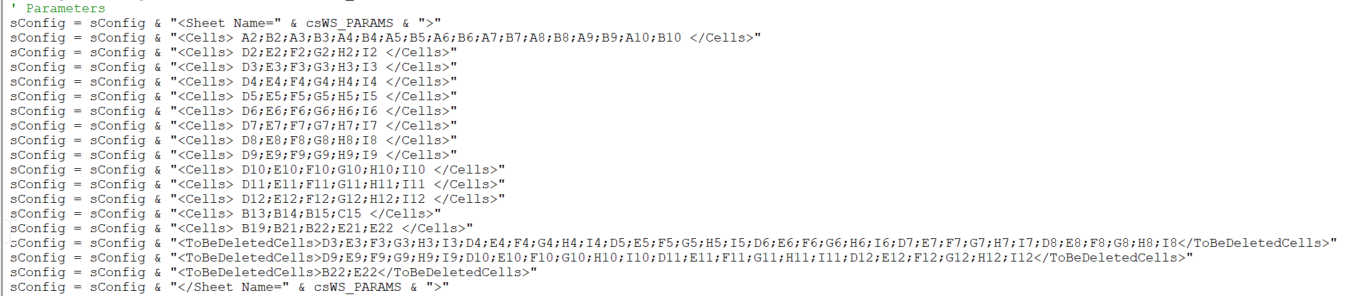
### Loading the configuration

To load a configuration, just press the “Load Configuration” button. There is no dialog, it will load the UserToolConfig.txt file present in the directory where the Excel is located. If you have multiple configurations, you need to manage, copy and rename yourself. It will overwrite anything that was located in the cells present in the configuration file, even those that were blank when the file was created (it will empty those cells).

### Delete private config

There is a button “Delete private config”. This button will delete API keys and all but the first in the configured environments (which should be set to local environment, and is therefore not secret). This will enable you to send the excel to someone else without compromising security. The receiver will be able to reload his own configuration, which should contain his own environments and API keys.

The cells that will be deleted can also be seen in the function “GetConfiguration”, in the tab <ToBeDeletedCells>:



# Field List

(this is moved to chapter 15 – download data from T24 to Excel)

# USERS

## Creating Users

To Create a user, fields need to be filled in in the list, from line 4, one line per user to be created.



### ID

This is the ID of the User record. Cannot be equal to either User Name, or the Login. Needs to start with a Letter (non-numeric), cannot contain spaces (example: WILLEMGORTER).

### USER.NAME

This is the Name of the User, as it is present on the Login Screen (Top Left corner). Can contain spaces (example: Willem Gorter).

### SIGN.ON.NAME

This is the Login that is used on the Login screen for T24 (example: WILLEM). Needs to be minimum 6 characters.

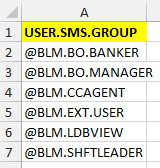
### LANGUAGE

The user language determines the layout of the T24 screens. Set to 1, it will choose the English language and the screens will be left-to-right. Set to 2, it will present the screens right-to-left, and where available, present Hebrew translations. The only valid values are 1 or 2.

### Profile

The profile is pre-defined and contains the access rights to applications, screens et. As well as the startup menu that will be presented upon logon. The profiles are all records present in the USER.SMS.GROUP (USG) table, and the record ID’s need to be pre-fixed with the @-character.

The list of delivered USG records is in Parameter sheet, which presents as the dropdown for this column. The second column in the Parameter sheet is the HomePage of the profile, which will be set as well through Browser Preferences. This changes the screen that appears when logging on.



To create a User with all rights (like INPUTTER), leave this field to blank.

### SuperUser

This determines if the user will have access to the commandline in the browser, which can launch any application not in the menu.

### Expiration date

If blank, the date will be set to 31/12/2049. You can set any other date greater than today.

### ID num

This column is present as of v4 of the tool. It enables to fill this column, which is a local reference field added for synchronisation later with IDM. It need not be filled, but it can, it is not used until interface with IDM is in place.

### OFS

This column is not an input column, execution will fill this column with the generated OFS corresponding to the line. For debugging purposes. After execution, should contain for example:

USER,MB.OFS.AUTH/I/VALIDATE,WILLEM/\*\*\*\*\*\*/IL0010001,u0586354408,USER.NAME::=%22%57%49%4C%4C%45%4D%47%4F%52%54%45%52%32%22,SIGN.ON.NAME::=%22%57%49%4C%4C%45%4D%32%22,LANGUAGE::=1,ATTRIBUTES:1:=%22%22,START.DATE.PROFILE::=%2220160623%22,END.DATE.PROFILE::=%2220491231%22,L.IS.ACTIVE::=0,CLASSIFICATION::=%22INT%22,COMPANY.CODE::=%22IL0010001%22,PASSWORD.VALIDITY::=%2220181201M0601%22,START.TIME:1:=0,END.TIME:1:=2400,TIME.OUT.MINUTES::=999,ATTEMPTS::=9,COMPANY.RESTR:1:=%22ALL%22,FUNCTION::=%22%22,APPLICATION::=%22@BLM.CCAGENT%22,SIGN.ON.OFF.LOG::=%22NO%22,SECURITY.MGMT.L::=%22NO%22,APPLICATION.LOG::=%22NO%22,FUNCTION.ID.LOG::=%22NO%22,INPUT.DAY.MONTH::=%22DDMM%22,DEPARTMENT.CODE::=%221%22,CLEAR.SCREEN::=%22Y%22,L.ID.NUM::=%2285974568%22

As of v4 of the tool, the password will be replaced by \*\*\*\*\*\* in the OFS column.

### RESULT

This column will hold the result of the execution of the OFS message. A typical result is:

u0586354408//1,USER.NAME:1:1=WILLEMGORTER2,SIGN.ON.NAME:1:1=WILLEM2,CLASSIFICATION:1:1=INT,LANGUAGE:1:1=1,COMPANY.CODE:1:1=IL0010001,DEPARTMENT.CODE:1:1=1,PASSWORD.VALIDITY:1:1=20181201M0601,START.DATE.PROFILE:1:1=20160623,END.DATE.PROFILE:1:1=20491231,START.TIME:1:1=0,END.TIME:1:1=2400,TIME.OUT.MINUTES:1:1=999,ATTEMPTS:1:1=9,COMPANY.RESTR:1:1=ALL,APPLICATION:1:1=@BLM.CCAGENT,SIGN.ON.OFF.LOG:1:1=NO,SECURITY.MGMT.L:1:1=NO,APPLICATION.LOG:1:1=NO,FUNCTION.ID.LOG:1:1=NO,INPUT.DAY.MONTH:1:1=DDMM,CLEAR.SCREEN:1:1=Y,L.ID.NUM:1:1=85974568,L.IS.ACTIVE:1:1=0,DATE.FORMAT:1:1=1,CO.CODE:1:1=IL00100010

typically, you can find the result after the first double slash //, it will be either 1 or -1 if the execution generated an error. Sometimes, even a -1 will be correct, for example if the record exists and nothing is changed (executed twice), the message after the -1 would be “LIVE RECORD NOT CHANGED”, it means nothing is written, but it is not really an error.

In case of success, OFS will return the record with all filled fields in OFS format, included fields not provided but for example defaulted by the application (like the last field, CO.CODE).

## Expiring Users



If you want to expire a user, you can add the User ID to the same list. ONLY fill two fields:

* ID
* Expiration date

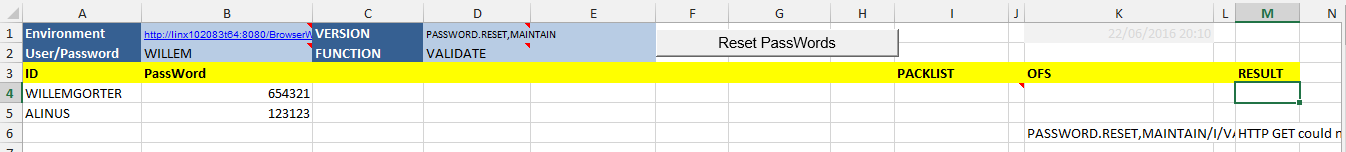
All other fields should be left blank. The expiration date should be set to today, the tool will also set the ACTIVE flag (L.IS.ACTIVE) to NO.

# Reset Passwords

Changing passwords can be done in two cases:

* When a user has typed 3 times a wrong password, the user is disabled and the password needs to be reset.
* When the password needs to be changed, typically for technical users

To change the password for a user, input the USER ID and the new password. The PASSWORD.RESET application can take a list of users to reset, so the whole list will only generate one single OFS message.



# ExecuteOFS

This sheet is used if you want to just execute a list of OFS commands, and save the result. It is an alternative to using "tRun tSS GCS" command, with the added advantage that the list can easily be executed without changes in multiple environments if the placeholder specified in the header for USER/PWD is used. Typically, you can put in the OFS column a string like:

EB.LOOKUP,MB.OFS.AUTH/I/PROCESS,<USER>/<PWD>,LEV.RISK.CODE\*61,DESCRIPTION:1:=%22Involved Party- corporation%2

And the <USER>/<PWD> will be replaced with the actual values from the execution

## Construction of the OFS

Alternatively, you can construct the first part of the OFS from the header fields, if you only have the field values.

To construct the above string, you would need to put LEV.RISK.CODE\*61 in the first column and DESCRIPTION:1:=%22Involved Party- corporation%2 in the second column. Clicking on the "Create OFS button would create the OFS, and it permits to vary the function (validate/process), the version to use, and it will automatically add the placeholder for USER/PWD.

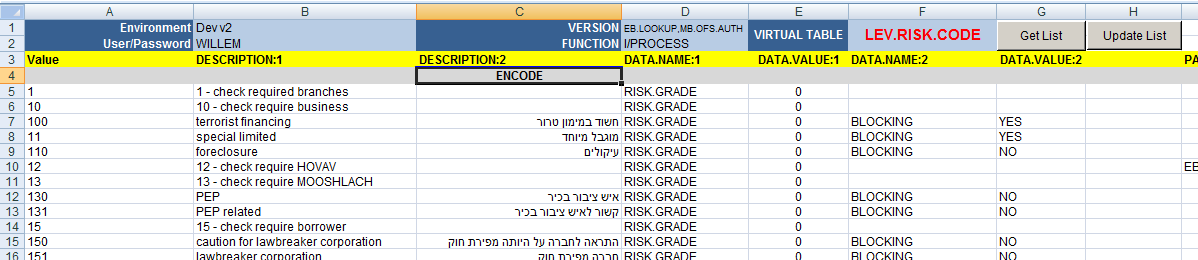
## Executing the OFS

Clicking on "Execute OFS" would execute the strings in the OFS column one by one and place the results in the result (G) column.

Attention, the execution is slower than using a terminal like tSS, as there is a lot of transformation and access control going on when going through TAFJEE services on jBoss. For huge lists, this is a good way to test the OFS, and then execute it on tSS.

# EB.LOOKUP

EB.LOOKUP contains the lookuplists for discrete values for applications in T24. The lists can be maintained in EB.LOOKUP with this tool.



You only need to specify (in cell F1) the Virtual Table (which is the part of the EB.LOOKUP record ID before the \*, as well as a field on the record.

After inputting the virtual table, clicking Get List will retrieve all records from T24.

The values can be changed in the excel, then select the lines to update (select more than one cell), and click Update List to push the values to T24.

Clicking Update List without selecting (more than one cell) will update the entire list.

Records can be added to the end of the list, these will also be created in T24.

For any fields that will contain non-Latin characters (Arabic, Hebrew), please make sure that line contains the string "ENCODE" so that the values are encoded before sending through http-get, else the OFS command will fail.

This excel sheet can also be used to retrieve and update generically any application that has a list of ID's with a common part and a \* as separator, and is tested on BLM.KYC.ANSWER. It will read the application for listing the records from the version, and prefix with % to use the default enquiry.

# Updates documentation

T24 Core fixes are mostly delivered as downloads from the TCSP Updates website, and installed with the updater program. That program will provide xml files the documentation of the content of the fixes, and that is uploaded to T24.UPDATES.RELEASE application. The list of installed updates is also updated to SPF.

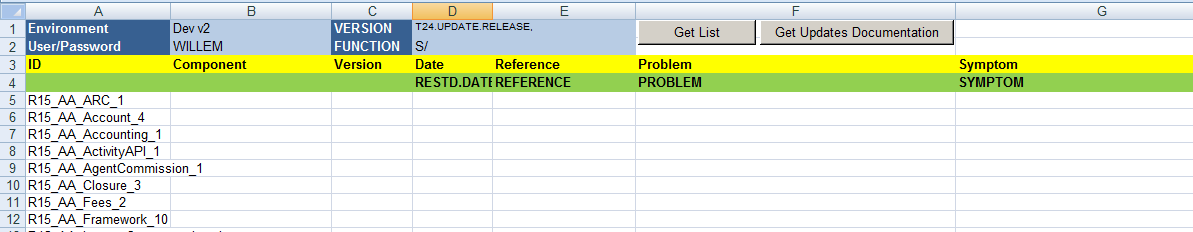
## Get List of updates

For the purpose of documentation of an update library that is deliverd, the list can be retrieved through a directory list from an unzip of the download for example.

The complete list of installed updates can also be retrieved from SPF, this could server for comparison of installed updates between environments. To retrieve the list of installed updates for an environment, click the "Get List" button. This will fill column A of the table only.

## Get documentation of updates

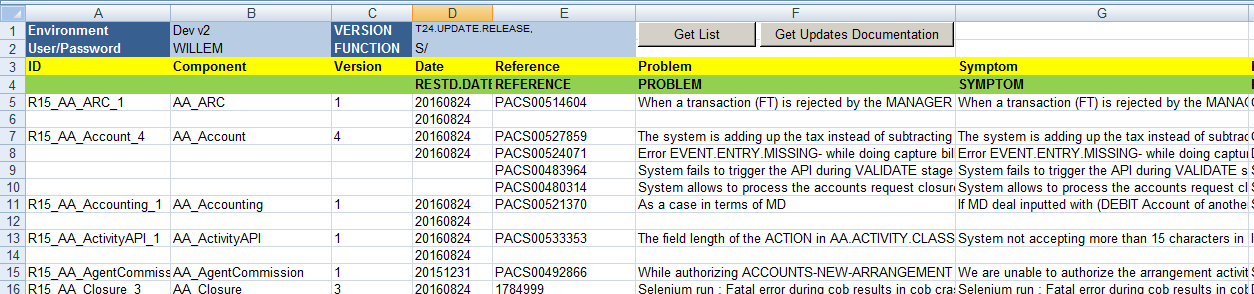
The tool will extract the documentation in a readable format from a list of installed updates:



Clicking on update list will retrieve the information for the whole list or for any selection of more than two fields.

**ATTENTION**: retrieving the documentation will insert lines, as there might be more than one fix for one module update.

After creating the download, select columns F through J, and click "wrap text"

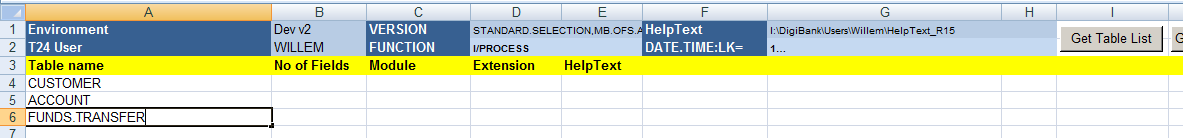


# DataDictionary

To generate a datadictionary for any T24 Table, this sheet can be used. Either the names of the tables for which to generate the structure are entered manually, or retrieved from the T24 table FILE.CONTROL, which has a record for all T24 data tables (Standard Selection would also have records for Nofile enquiries which are associated to virtual tables, and don't have a physical presence in the database)

## Entering tables manually

Enter the table names in column A starting line 4:



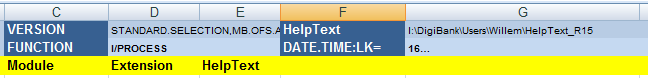
After entering the table names, click the "Generate Data Dictionary" button to generate the structure.

The tool will also retrieve the Module name from FILE.CONTROL if the column is empty, the module is needed to retrieve the helptext from the correct subdirectory.

## Retrieving tables from FILE.CONTROL

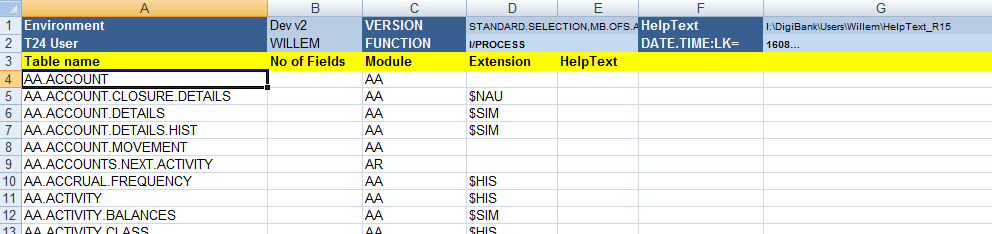
### Selection of FileControl records

To retrieve the list of files from File Control, you can input selection criteria. The most common one would be to retrieve the tables created during the project, and therefore all FILE.CONTROL records for which the DATE.TIME fields contains a date after the start of the project. For example, to retrieve all records created on or after 01-01-2016:



(when entering the three dots, take care that they remain three dots, Excel automatically transforms it to the "three dots" character, which would not generate a correct OFS syntax)

When clicking the "Get Table List" button, the list of File Control records retrieved is inserted in column A. The Module is also retrieved, which makes the generation of the Dictionary quicker:



### Cleaning the list

After retrieving the list, you might need to perform some cleanup: sometimes, with a T24 Updates installation, an updated version of FILE.CONTROL is delivered, and the DATE.TIME in that case will be recent, and will be included in the list. Just delete the rows with the table names you want to delete, please don’t leave any blanks – execution will stop at the first blank cell encountered in column A.

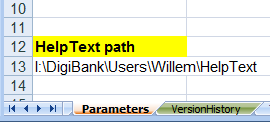
### Configuring HelpText location

In Cell G1, you will have to configure the (Local) location, if available, of the T24 HelpText. You can either take it from ModelBank, or copy it from the T24 Server to a local directory. It should be the path to a directory containing lots of folders with a 2-letter name (the module), like AA, EB, ST, etc.

This will enable the tool to extract the helptext from there to enrich the datadictionary with the explanations, and it will also enable you to update the helptext (preferably only on local tables...).

For that purpose, the tool will create the xml files in the same structure as T24 (in subdirectories per module), but in a specific, different location. You can copy the structure later into the T24 helptext path to update the helptext.

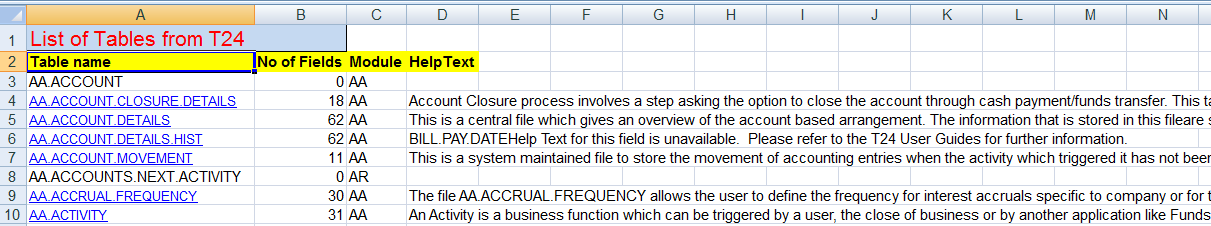
The path where the helptext files are created is defined in the parameters worksheet, in cell A13:



## Generating DataDictionary

When clicking the Generate Data Dictionary button, T24 will create a separate workbook named "T24\_DataDictionary.xlsx", saved in the default directory, containing one index sheet, and one sheet per table describing the structure. The workbook remains open after generation.

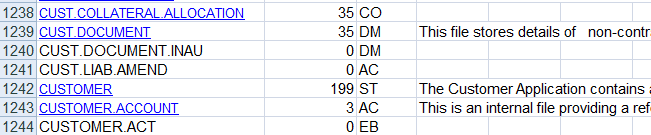
### Index of the Datadictionary



The first sheet of the new Workbook contains the index. It is a copy of the list of tables in Column A of the original "DataDictionary" worksheet. It contains an extra column which contains the number of fields, and the table name is a hyperlink which will navigate within the workbook to the relevant worksheet containing the structure of that table.

Column D holds the "Overview" part of the helptext for the application, if present

Some tables will not have a hyperlink, in that case the worksheet for that file will not be there, and this is because the file does not contain any fields. This can be the case for a concat or live file:

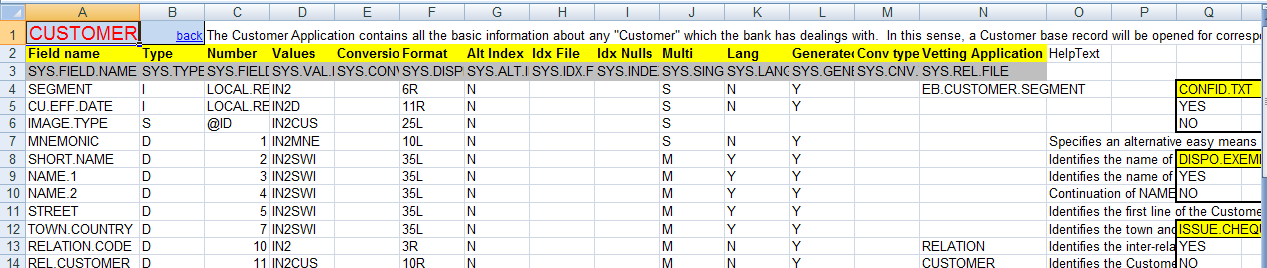


### The DataDictionary sheet

The data structure for one table is retrieved from the T24 table STANDARD.SELECTION. It contains the datatype, any vetting (if the field is a foreign key to another table) as well as the list of values if it is a discrete type. This includes values from EB.DICTIONARY if the field is vetted on that table.

For ease of Navigation, the "back" link in cell B1 will bring you back to the index (List of T24 Tables) worksheet. In Cell C1 you will find the "overview" part of the online helptext for the application if present.

The following columns that are retrieved contain important information (the rest can be discarded):



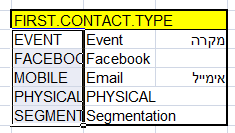
#### Field name

this is the name of the T24 field. In insight, the dots would be replaced with underscores.

#### Type

D stands for Data, which indicates that it is a datafield. I and J would be virtual fields. For some datatypes, the D is overwritten with either a LIST with a hyperlink to a table containing the discrete values possible for the field, or if the field cannot be changed by the user, with NOINPUT.

If a hyperlink in this column is clicked, it will bring the selected cell to the right of the list to the correct table of discrete values, either retrieved from the application or from EB.DICTIONARY:



#### Number

This is the fieldnumber, the sequence of the field as stored in the record in T24

#### Values

This contains the name of the T24 routine that validate the data. IN2 permits numeric values, IN2A permist alphanumeric values. Specific routines validate differently, IN2CUS would check if the customer exists. In case of discrete values, this formula contains the values as well as their descriptions in all languages. This values is used to generate the table (see ‎11.3.2.2)

#### Format

The format column defines how the field is displayed and can be inputted. Mostly it contains the number of characters and the alignment (12R would be right-aligned, maximum 12 characters or digits)

#### Multi

If this field is set to yes, then the field can contain multiple values divided by value-markers.

#### Lang

If this field is set to yes, the field can contain multiple languages. It is in fact a special case of a multivalued field, but for displaying, it is treated differently: only one value, that corresponding to the language of the user, is retrieved from the table.

#### Vetting application

If the field is a foreign key, the application for which the field is an ID is displayed in this column.

#### HelpText

This column contains the T24 online helptext for the field in question.

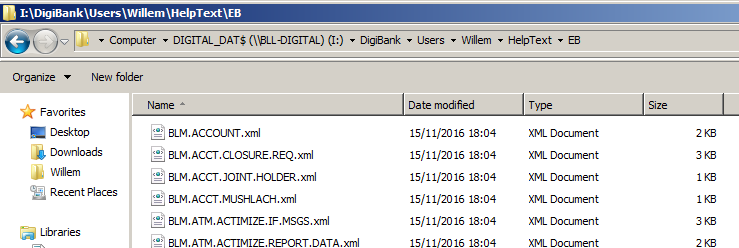
## Updating HelpText

The DataDictionary can be used to easily update the HelpText for T24. When the HelpText is updated, any future re-generation of the DataDictionary will also contain the new HelpText.

To update the HelpText, select two or more cells to define for which tables you want to generate helptext, and click if only one cell is selected, HelpText will be generated for all tables in the list.

It will check if there is an open WorkBook called "T24\_DataDictionary.xlsx", if not, it will try to open it from the default location.

For each table in the selection, it will check if it is present in the Data Dictionary, and if that is the case, it will take the text in C3 as text for the overview section of the helptext, and the text in the HelpText column as text for each field, and generate an xml file in the directory configured in Parameters sheet:

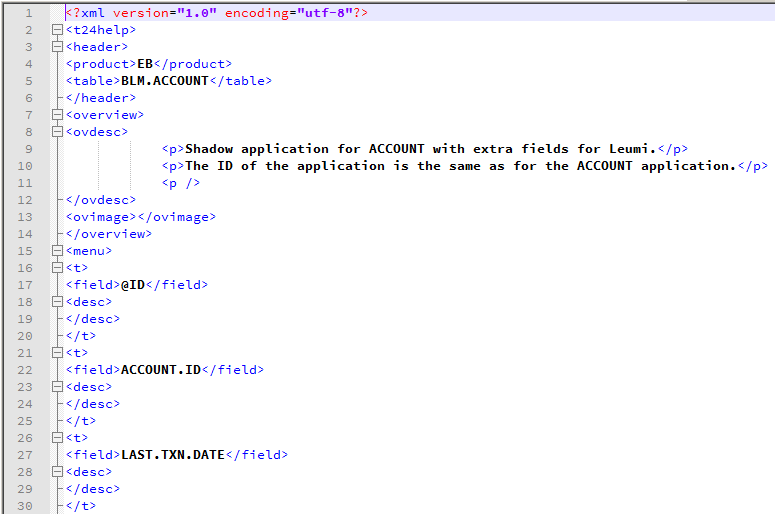


The xml file is in the format expected by T24 for its helptext files. To use the new helptext, copy "module" directories that contain the helpfiles, and merge them into T24's helptext location on the server or in your ModelBank. In a local ModelBank, the files are located in:

\Temenos\ModelBank-R15-TAFJ\T24\Programs\HelpText\MB

In other installations, you can find (and define) the location of the HelpText files in BrowserWeb.war, in the file HelpParameters.xml (in modelbank, you can find BrowserWeb.war in:  
\Temenos\ModelBank-R15-TAFJ\Infra\JBoss\server\default\lite

The structure of the xml files is as follows:



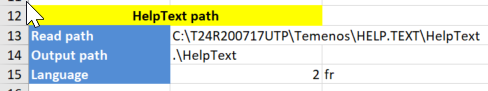
# Translating online helptext

Although the DataDictionary can be used to update helptext for applications, or create helptext for new local applications, it is not dedicated to helptext. The “Create Helptext” tab is that, and it also allows translation of the helptext in a local language, as well as translation using Microsoft Translator (part of Azure Cognitive Services).

## Setting up helptext creation/update and translation

To setup helptext translation, a few things need to be set up in the Parameters tab:

### Set the paths of the helptext

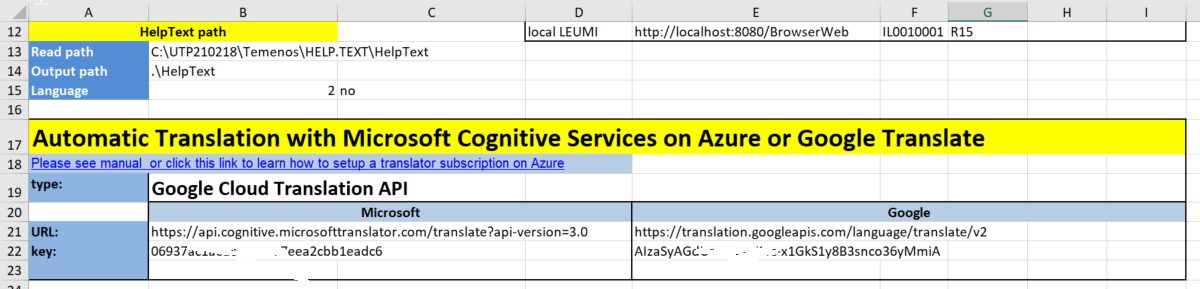


In the parameters tab in Cell A13, you should give the path to the helptext directory where your T24 installation reads from, or a copy. In A14 you should put the output path. It can be the same path, but if you don’t want to overwrite the original, or if you need to package the helptext files, it is easier to specify a different (local) path for outputting the helptext files. It will create the appropriate folder structure automatically in the output folder.

These two paths are sufficient information to update or create English helptext. If you need to translate, you also need to specify the language number (normally 2), and the language code in non-capital letters.

### Activate the automatic translation

To automatically translate English texts to the target language, the tool can call the Microsoft Translator Service hosted in the Azure cloud (Microsoft Cognitive Services) or Google Translate hosted in the Google Cloud. To use this, the excel tool needs a subscription key, which needs to be copied to the Parameters tab, cell B23 (for Microsoft) or E26 (for Google). Set the translation service you want to use in B19:



The URLs for the services should be the same for everyone, but might evolve in future.

To obtain a subscription key, you need to either create an Azure account or a Google Cloud Platform account:

#### Microsoft Azure and Cognitive Services

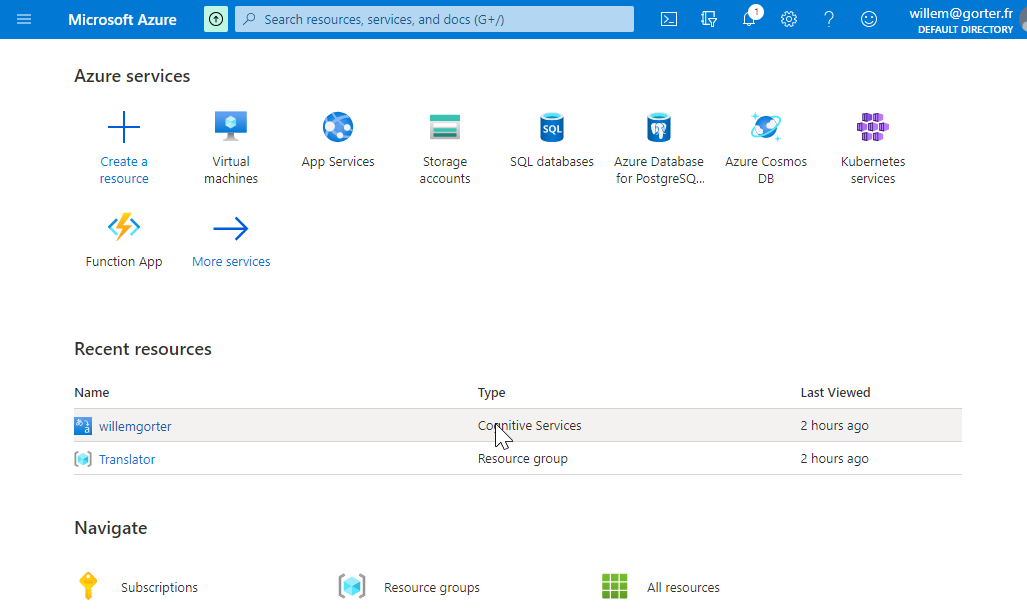
To get a Microsoft Azure Cognitive Services subscription key, you need to create a Microsoft Azure account, and create a Translator resource in the account. To do that, you can follow the instructions described here:

<https://docs.microsoft.com/en-us/azure/cognitive-services/translator/translator-text-how-to-signup>

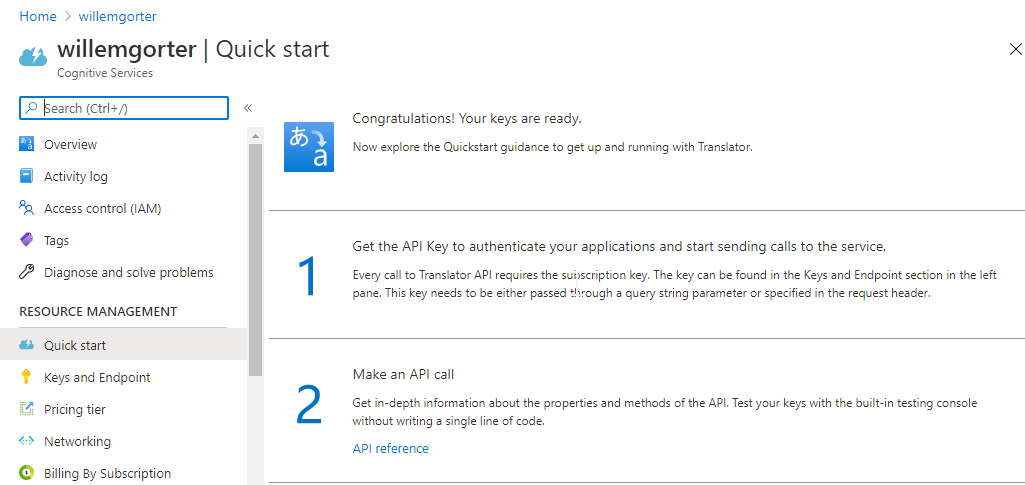
It takes about 5 minutes to sign up and activate the Translator service and obtain your keys, if everything goes well. You need a Microsoft Live account (or create one), as well as a phone (for dual factor authentication), and a credit card (or debit card), although nothing is debited.

The “free” pricing plan allows you to translate up to 2 million characters per month, and will not allow you to translate more (and debit your card) until you upgrade to “pay as you go”.

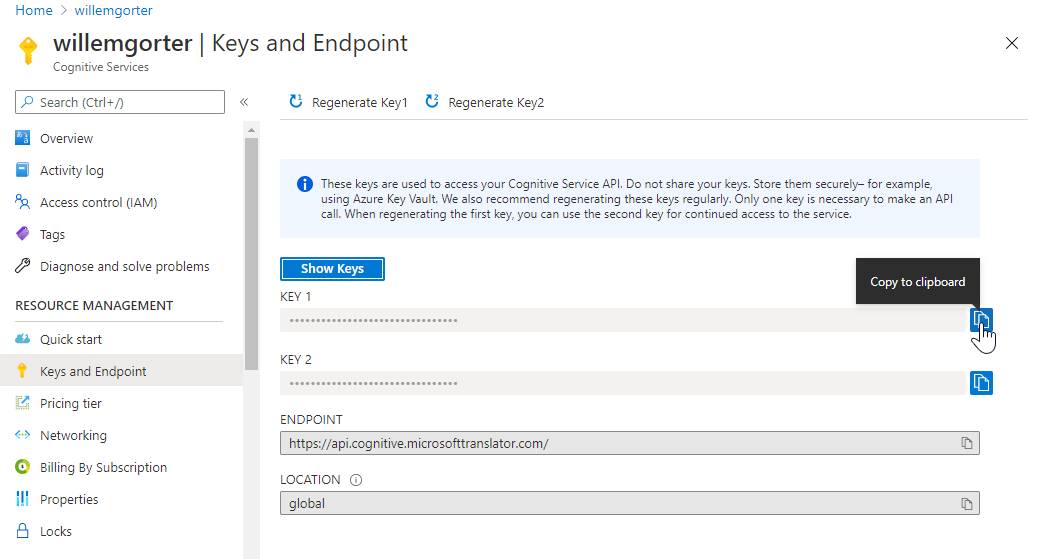
After activation of the Translator service, the Azure home screen looks like this:



Clicking on your resource (willemgorter for me), brings you to the resource menu:



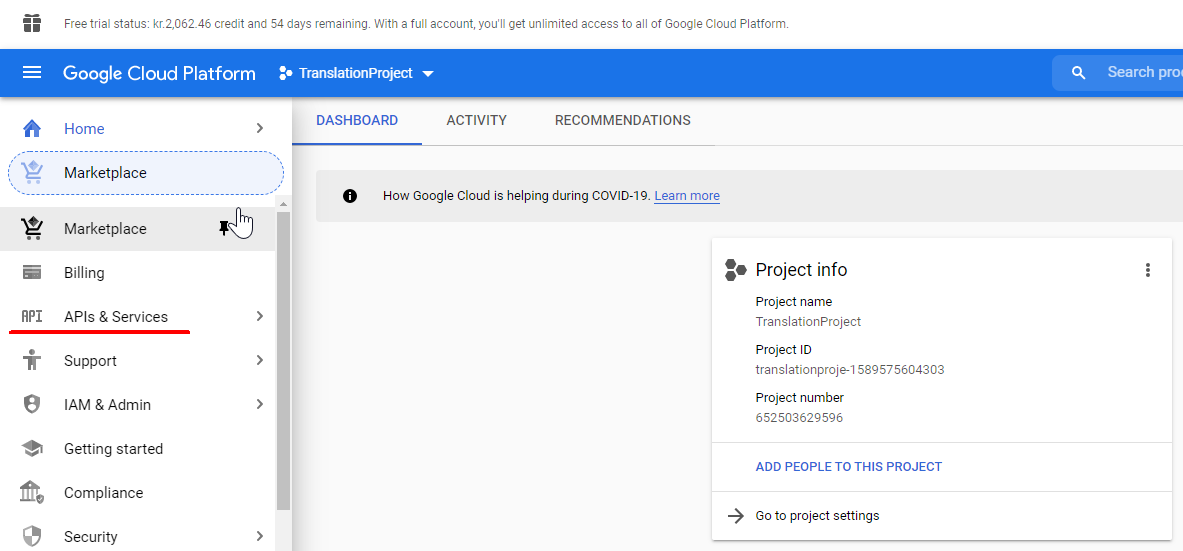
And when clicking on “Keys and Endpoint” in the left-hand menu, you can access and copy the key for the excel tool:

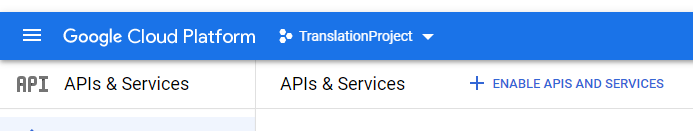


After copying the key to the Parameters tab, you can translate English texts retrieved from T24.

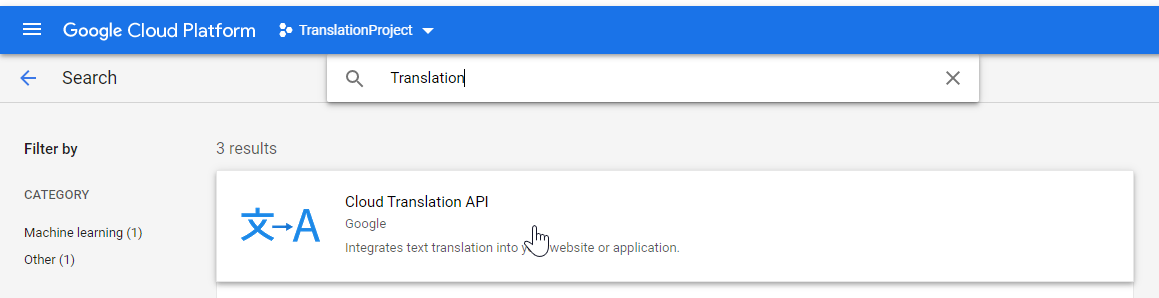
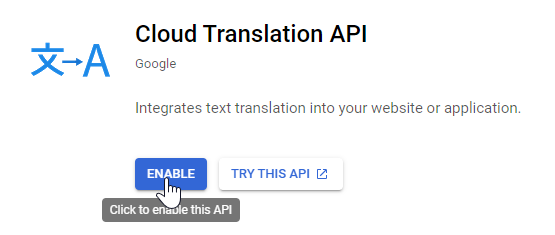
#### Google Cloud Platform and Cloud Translation API

To enable translation with Google, you need to create a Google Cloud Platform account. You need a google account to login to Google Cloud Platform. From the Platform Console screen, you need to create a project. After that, choose “API and Services” from the main menu, and click “+ enable APIs and Services”:

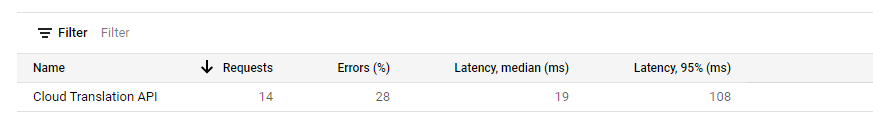




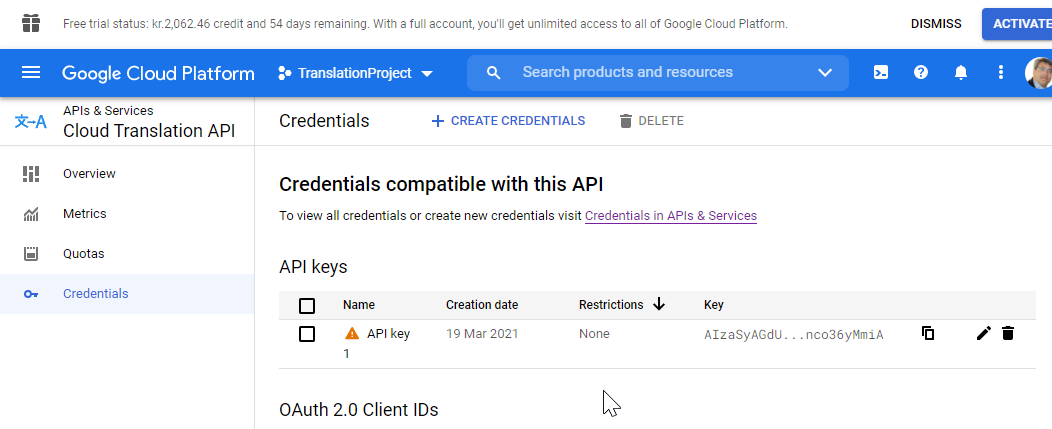
Search for “Translation” and choose to enable Cloud Translation API:

On your dashboard, you should have the API enabled:

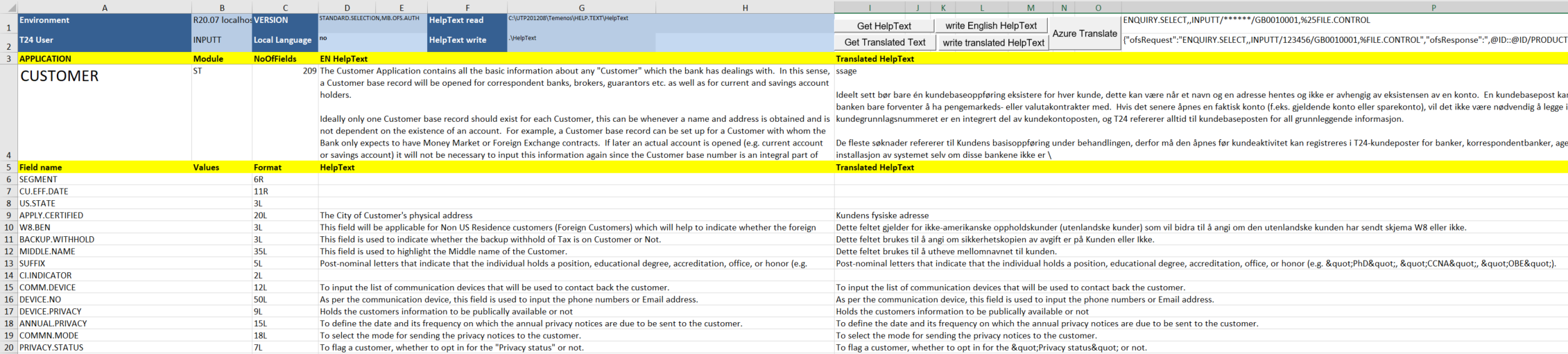


Click on the service, and choose “credentials” from the main menu. The API key will be displayed and you can copy it to the clipboard using the icon, and then paste it into the excel.:



## Retrieving the helptext

You should start with putting the name of the application for which you want to create or amend the helptext in Cell A4, and then press the button “Get HelpText”, see below for the result when inputting “CUSTOMER” in Cell A4:



In my case, the Norwegian translation was already present, and my language was set to Norwegian. In most cases on model bank, the right part will be blank. For local tables, or for tables for which model bank does not have helptext, only columns A, B and C will be filled (retrieved from Standard Selection).

## Retrieving Translated helptext

There is a separate button for retrieving Translated helptext. It will read (from the local, that is, the write path) the translated xml file (.\ST\CUSTOMER.no\_xml for Norwegian) and fill column I with the translation. It should not be needed, as it is automatically loaded when loading the English helptext if the file is present.

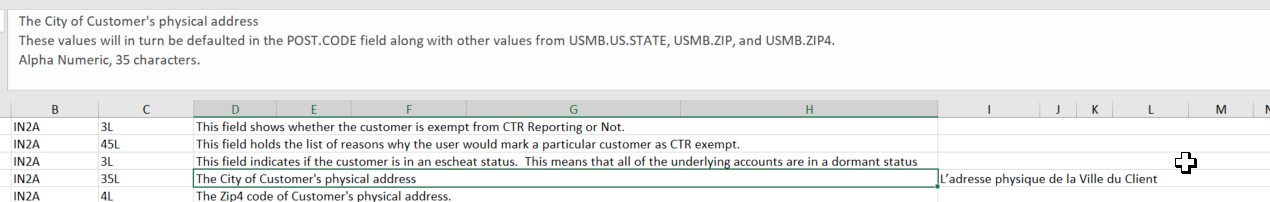
## Correcting or creating English helptext

If you want to create or update the English helptext, change or specify the text in column D. in line 3 you should specify the helptext for the application, in the lines 5 and higher the explanations for the corresponding fields. When finished, to create the helptext file, press “write English helptext” button, and retrieve the xml file in the specified local path, to copy it to the server where your T24 installation is reading from.

## Translating helptext to local language

You can translate the helptext in column I, either manually, or automatically. To translate manually, just type the text in column I. To translate automatically, select the cells in column D that you want to automatically translate, and press “Azure Translate” button. It will automatically put the translation in the corresponding cell in column I, but only if that cell is empty, so be careful to empty the cell if you want to translate. Be sure to check the automatic translation if you are able to, and correct manually if needed.

Check also if there are no hidden lines!

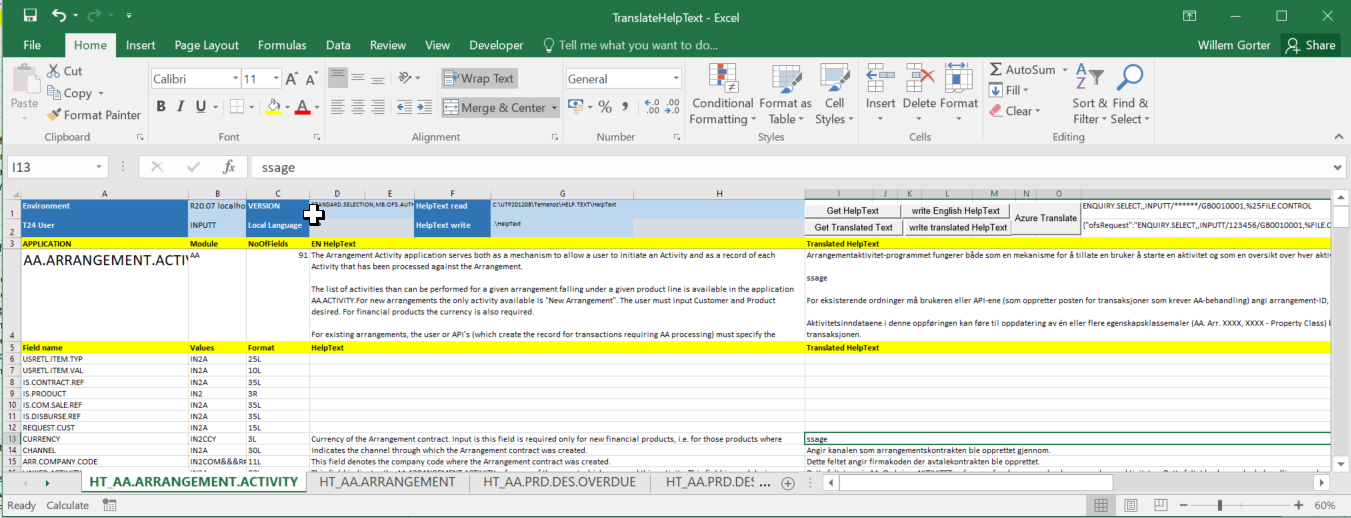


If the row hight is not correctly adjusted, it might be that you are only seeing the first line if the helptext contains soft returns! Either adjust rowheight, or if you are translating automatically, check the text in the input box after you extended it.

If you are finished, you can press the “write Translated helptext” button to create the helptext file for your local language, which will be in the same directory as the English helptext file in your local path. Copy it to the server in the same way as for English helptext.

## Creating translatable helptext for multiple applications

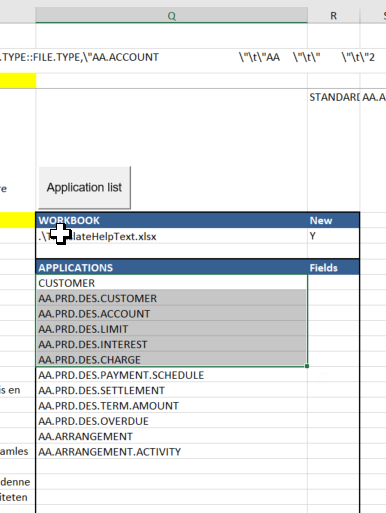
If you want to translate the helptext in Excel, you can create a workbook with a worksheet for each application that you want to translate:



To do that, from the “CreateHelpText” tab, fill in the list in column Q.

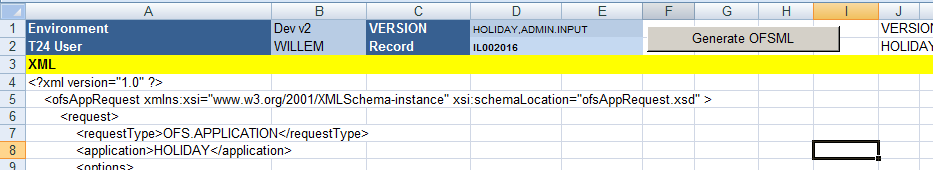
* The path and filename of the workbook to be created (make sure there is no existing file with that name in the folder) in cell Q6
* The list of applications starting from Q9

Select the applications to include in the workbook from the list, and click on the button “Application list” just above. Be aware that the system will create copies of the CreateHelpText worksheet in the workbook, renamed to “HT\_” plus the name of the application.

The buttons will not be working in the new workbook if the USER worksheet is not found in the original path (the macros are not copied to the new workbook!)

# Sample OFSML generation

Online incoming interfaces should trigger a T24 transaction using OFSML. To know what OFSML structure to use, this tab examines a Version, and generates a sample OFSML using data from any specified record:



2 variables need to be inputted in column D, line 1 and 2:

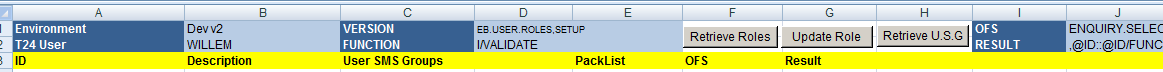
The Version that should be used by the interface  
the ID of a record that should be used to extract sample data

The sample OFSML will be generated in column A, and can be copied to a text or xml editor.

# Update EB.USER.ROLES and USER.SMS.GROUPS

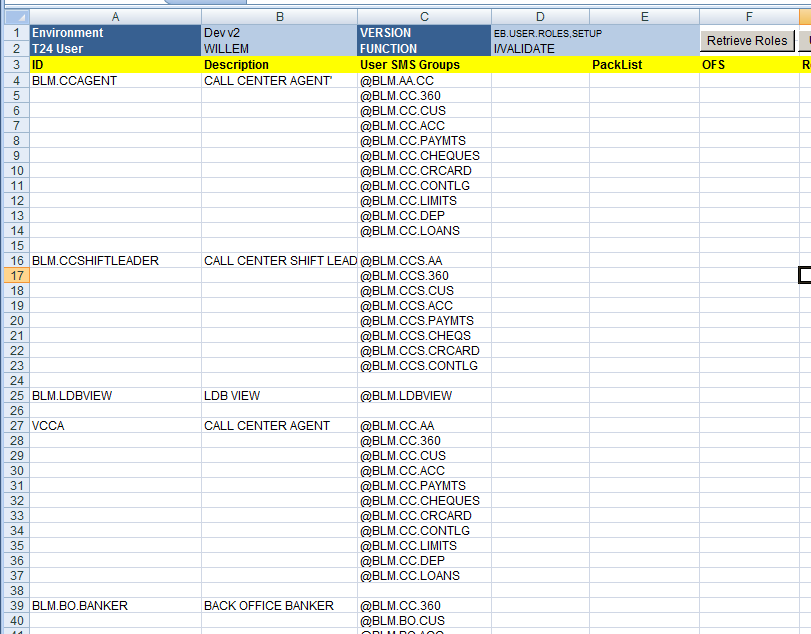
The security management in T24 is done through these two applications: a User is assigned one or more roles, a role is composes of multiple USER.SMS.GROUPS (USG's), and in a USG is defined exactly what function is allowed on what application, version or enquiry, with what datarestrictions.

To update these rights can be very tedious in T24, and the process is prone to mistakes. It is much easier to do the work in Excel and export this to T24. To do this, there is an excel sheet UserRoles:



## Retrieve EB.USER.ROLES

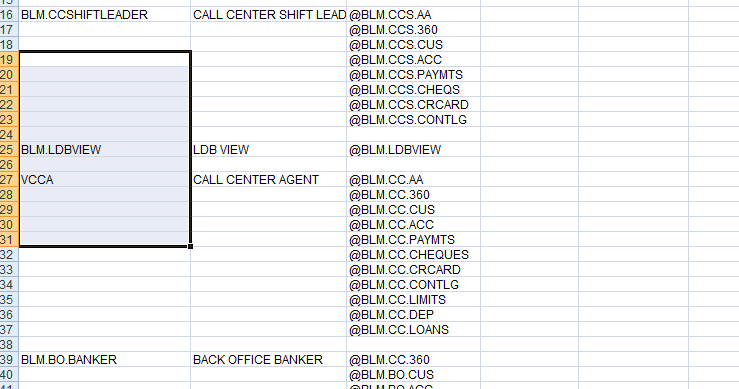
There is no specific setup necessary for this sheet, the first action is to retrieve from T24 the EB.USER.ROLE records that will list the USG's per role. Click on the "Retrieve Roles" button:

****

**The various Roles present in the environment are retrieved, and a blank line is present between the roles.**

## Updating Roles

Either the description or the list of USG's can be amended, and the tool will update T24 when using "Update Role" button. To ensure that the correct Role is updated, select more than one cell in column A, where the selection includes all the names of the Roles you want to update.



The above selection would update the two profiles BLM.LDBVIEW and VCCA, regardless of which USG's are included in the selection.

To update one Role, please select at least one cell above or below it too.

Click on the "Update Role" button to update T24.

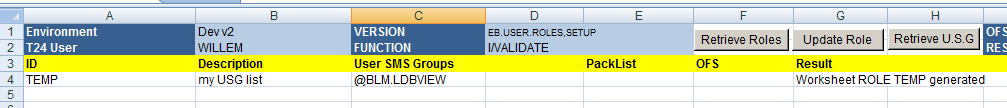
## Retrieve USG

Most of the work is generally done on the USER.SMS.GROUP's. After selecting one or more roles (see above paragraph), when clicking the "Retrieve USG" button, the tool will create A NEW WORKSHEET in the current workbook per selected Role.

The worksheets will be situated as the last sheets, and their name will be "USG\_" concatenated with the Role name:



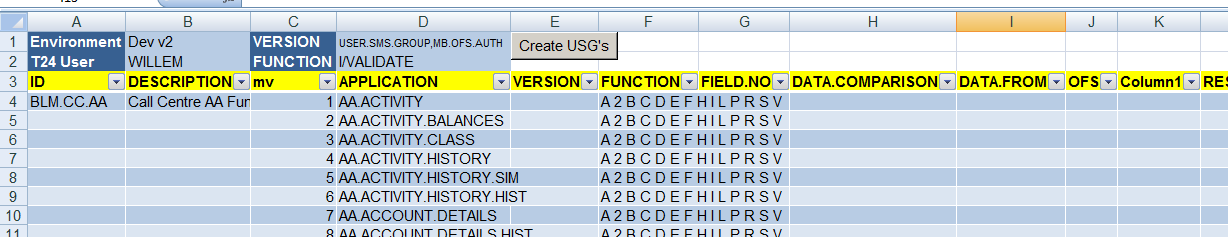
If you want only to edit one specific USG, you can even just create the input yourself, the role doesn't need to exist:



TEMP is used only for the name of the resulting worksheet, but is not checked. Be careful to add a '@ before the USG you want to edit.

## USER.SMS.GROUP worksheet

The created worksheet lists for the profile all USG's with for all USG's the applications that are allowed and per application the functions and the filter:



## Create USG's

When clicking on the "Create USG's" button, without a specific selection, the entire list of USG's is created in the target environment. This can be used easily to transfer profiles from one environment to another: retrieve from one environment, change environment (cell B2 in the USG\_ worksheet), and click Create USG's.

Alternatively, if your selection is one single non-blank cell in column A, it will only execute that one USG.

### Execution line per line

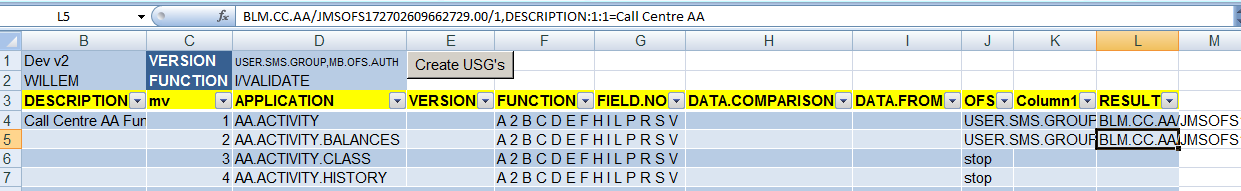
Mostly, USG's have huge lists of Applications, and an http-request is limited. Therefore the tool will update the USG multivalue by multivalue (or visually, Excel line by Excel line). This might take a certain time,you can follow the progress in the "OFS" column. The execution will stop as soon as it encounters an error.

### Fixing errors

When executing, T24 will skip all the lines where the "OFS" column is filled. Therefore, when you encounter an error in line 351 (multivalue 348), empty for that line only column J and L, fix the error (probably somewhere a spelling mistake, take a look at the result in column L before you delete it), and click the button again, it will continue where it left off.

### Special uses

When you only want to add or remove a function ("A" or "I") on an existing Application, and not change anything else, put everywhere in the whole sheet "STOP" (or anything else, just not blank) in column J, and make just for the line where you want to change something the J column blank. It will only execute the change for that single multivalue.

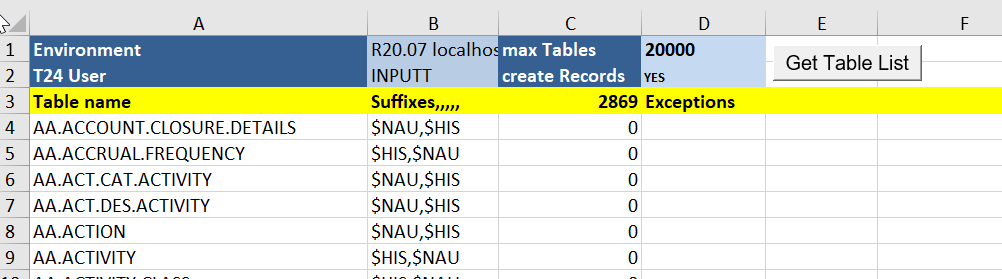


If you want to remove multi-values, this cannot be done with the tool. You can overwrite the values with already existing values keeping all the multivalues present. Alternatively, you can edit the USG with T24 browser and delete the multi-value sets there, then re-retrieve the Role in a new sheet.

# Retrieve Unauthorised records and authorise

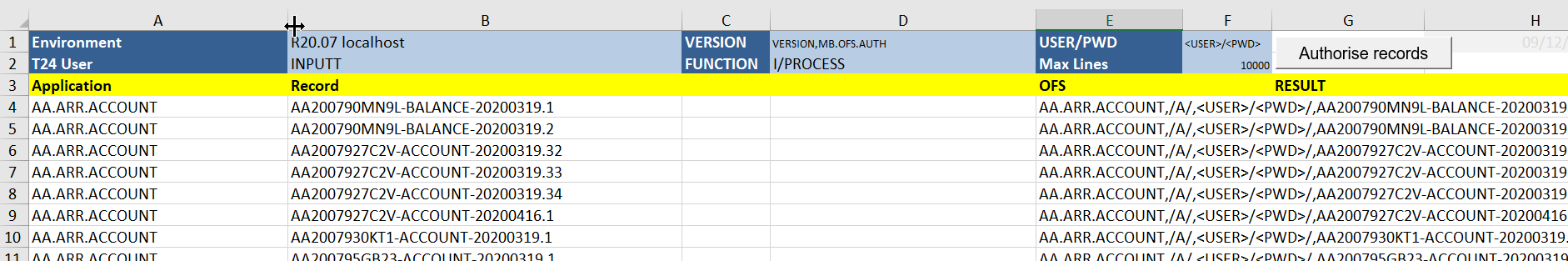
## Creating a list of INAU records

When installing updates or upgrades, Temenos might deliver datarecords, and these come installed in INAU record status (unauthorised records). To get the list of these unauthorised records, you can use the “INAU applications” tab:



When you press the “Get Table List” button, the tool will fetch the complete list of T24 tables that can have unauthorised records (live tables cannot, it needs to have a $NAU value in the suffix list in FileControl), and it will check for each table whether there are unauthorised records. As the core T24 model bank has 3400 tables that can have unauthorised records, it will run 3400 times an enquiry, and this might take some time. To test first, you can lower the “Max Tables” value, to get the complete list set it to 20.000 as shown above.

When finished, it will give the total number of unauthorised records in the header line – if that remains blank, the maximum number of unauthorised records is too low, it needs to be set in the record list tab “INAU records” in the header under Max Lines:



In this tab, when getting the list, it will list all the unauthorised records for all the tables of T24 which can be quite a lot, and it will generate the OFS command for authorising the record.

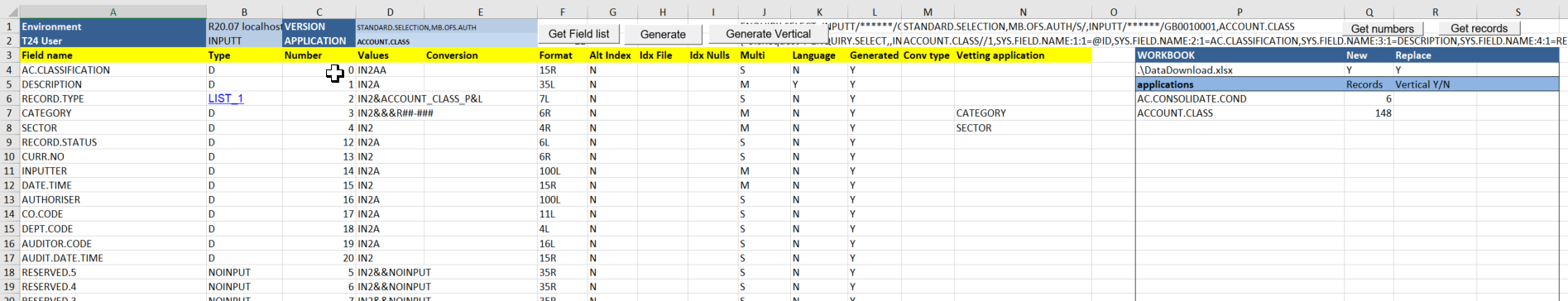
## Authorising INAU records from the list

To authorise the record (after examination and elimination of those that should not or cannot be authorised), you can authorise all records by pressing the “Authorise records” button, or you can make a selection: if you select more than one cell (either horizontally or vertically), it will authorise all the records for the lines that are in the selection (if you select cell B8 and C8, it will authorise the record on line 8 only, if you select B8 to B12 or D12, it will authorise the records from line 8 to 12)

For upgrade purposes, it is best to first retrieve all the records, save the excel list, perform the upgrade/update and re-execute the list retrieval, and then compare the two lists, so you know exactly which records came with the update/upgrade.

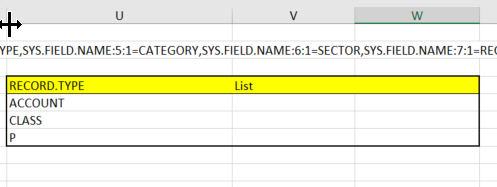
# Download T24 Data to Excel

This tool will allow you to create an excel with the same structure as a T24 table, with the fields either horizontally or vertically, and to fill that excel with one record (or more, but retrieved one by one), or with all records in the table. It will also allow you to create a new workbook with excel sheets filled with all records in the table for a list of tables. For all these functionalities, you use the “FieldList” tab:



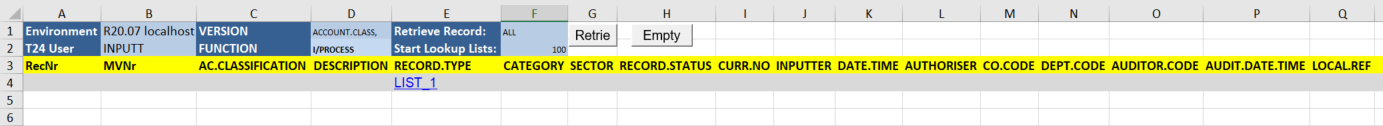
## Create an excel with the structure of a T24 table

In the FieldList tab, put the T24 application in cell D2, choose the correct environment from which to retrieve the data, and click on the “Get Field list” button. This will retrieve the Standard Selection record for the table, and decode the selection lists, which will be shown from column U:

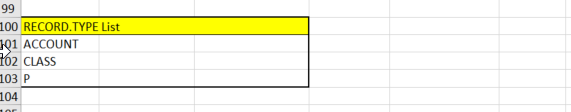


In Column B, you will have a hyperlink to the correct table for all fields that have a limited value list defined.

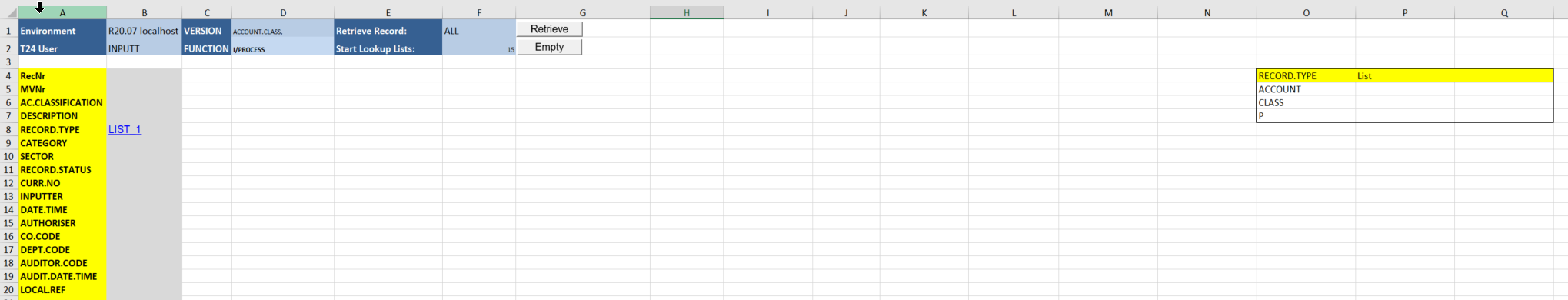
You now need to decide whether to create the excel with the fields horizontally (and the data in rows), or vertically, with the data in columns (this is useful for SPF for example, which has one data row “SYSTEM”, but lots of columns). For generating a horizontally oriented sheet, click on “Generate”, else click on “Generate Vertical”. The tool will generate a new excel sheet, named “Data\_” concatenated with the T24 application name (or “DataV\_” with the application, so you can have both at the same time to check which is the better layout):



../..

**

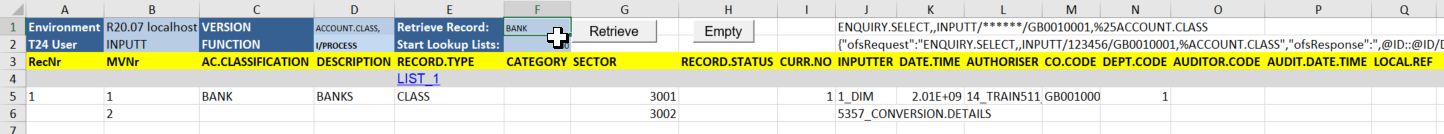
*Horizontal layout, the lookuplists start from line 100*



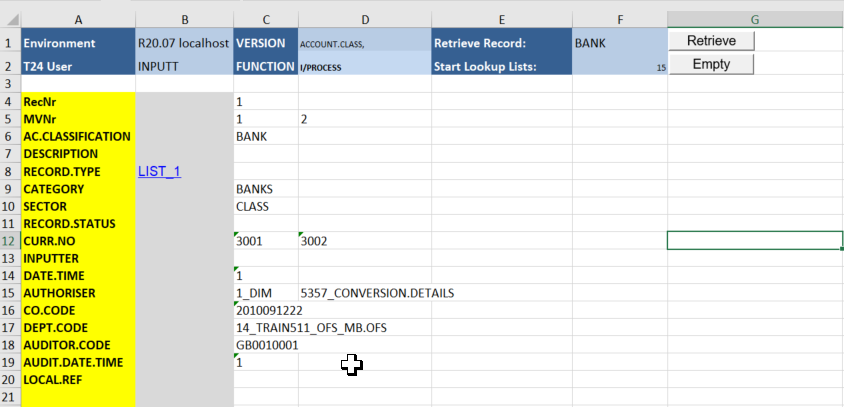
*Vertical layout, the lookuplists start from column O*

## Fill the excel with T24 Data

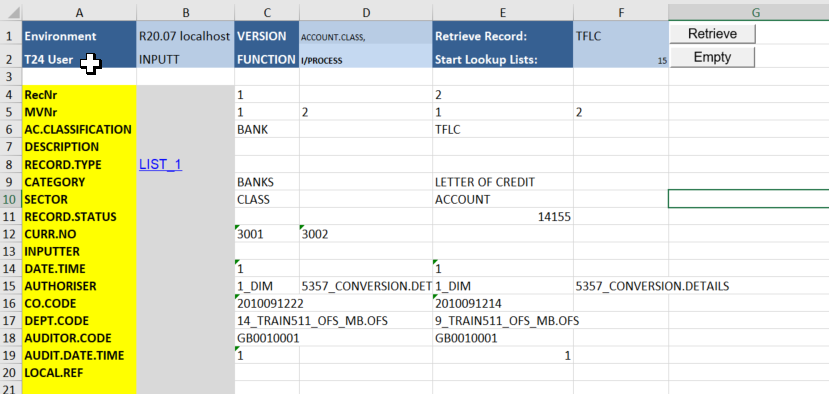
To retrieve data from one record from T24 to the excel, put the Record ID in cell F1, and click on “Retrieve”:



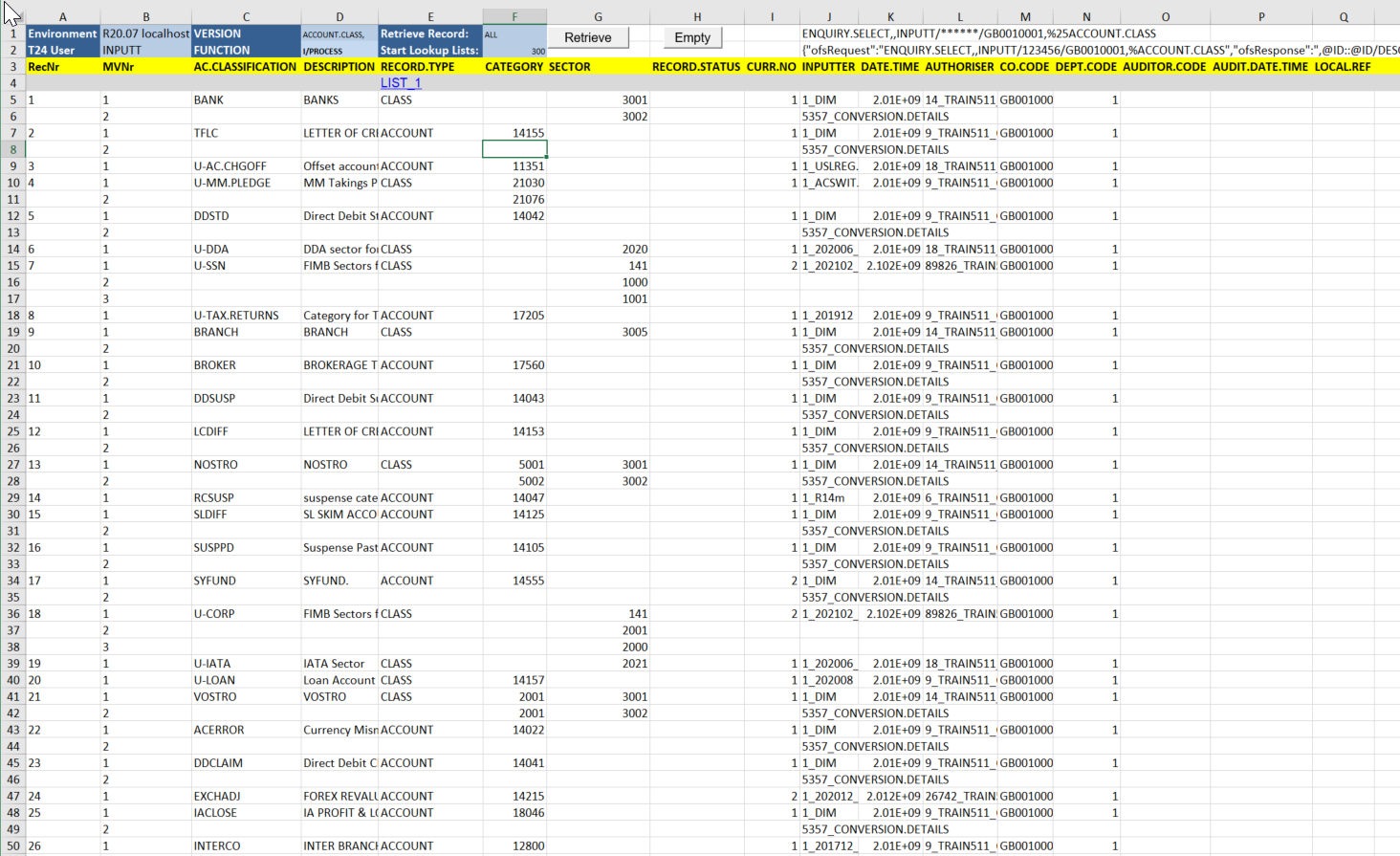
Or



Because the field “Sector” has to values, the record will use 2 lines or 2 columns, and column B indicates for each line the value number. If you specify another record, it will add it below or to the right of the record already present:



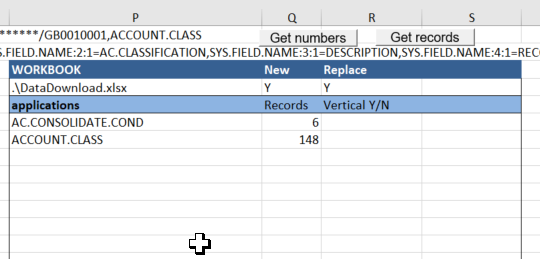
If you specify “ALL” in F1, Excel will retrieve all records from T24, be careful, that can take a long time. The Position of the lookup lists will get automatically pushed (in leaps of 100 lines, or 15 columns), you can find their new position in F2:



You can use the “Empty” button to empty your worksheet again and reposition the lookup lists.

## Generating data for a list of applications

Using the table starting column O:



You can automate data download for a list of T24 applications.

Create the list from P6 onwards. Before downloading the data, it is a good idea to check how many records T24 has (select all the applications you want to use and click “Get numbers”, T24 will put the record count for each application in column Q).

After examining the recordcount, you could decide to put some application vertically (if the recordcount is 1 or 2 for example), in that case, put a “Y” in column R against the application.

You can generate the worksheets in the same workbook, or you can choose to generate them in a new workbook, in that case put a “Y” in Q4, and the filename in P4. If you want to overwrite any existing workbook, put a “Y” in R4, but make sure that the old workbook is not open, because the system won’t be able to delete it.

To generate the data for a list of applications, select the applications from your list, and click “Get Records”. It will, for each application, generate the field list, then generate the worksheet, then fill the worksheet with all records, and then move the worksheet to the new workbook

