Qlik Connector for SAP - Tips and Tricks

Version 2.1

Contents

General

Installing the connectors

Download and double-click the installation file for 'Qlik Connector for SAP x64'. A folder structure is created under the path C:\Program Files\Common Files\QlikTech\Custom Data (QlikView) and C:\Program Files\Common Files\Qlik\Custom Data (Qlik Sense).

Connector dll and exe files are not from the installed release

To check the release version, look at properties/details for one of the connector dll files. If the release version is not correct, manually delete all sub folders of C:\Program Files\Common Files\QlikTech\Custom Data (QlikView) and/or C:\Program Files\Common Files\Qlik\Custom Data (Qlik Sense). Uninstall the 'Qlik Connector for SAP x64' from the 'Control panel' in Windows. Then double-click the connector installation file and perform the installation again.

Connector log files

The path to the log files is: C:\ProgramData\QlikTech\Custom Data\xxx\Log (for QlikView) or C:\ProgramData\Qlik\Custom Data\xxx\Log (for Qlik Sense).

Please check in the appropriate sub folder of 'Custom Data' dependent of which connector is used.

SAP services file

The file 'services' is located in the folder C:\Windows\System32\drivers\etc.

No SAP GUI installed

If there is no SAP GUI installed on the computer where the connector is to be installed, the SAP system gateway port has to be written manually in the file 'services'. An option could be to copy the 'services' file from a computer where the SAP GUI is installed.

Add the following information at the end of the file:

sapgw00 – sapgw99 3300/tcp - 3399/tcp #SAP System Gateway Port

sapdp00 – sapdp99 3200/tcp – 3299/tcp #SAP Dialog Service (dispatcher port)

Example:

Sapgw00 3300/tcp #SAP System Gateway Port

Sapdp00 3200/tcp #SAP Dialog Service Port

Sapgw01 3301/tcp #SAP System Gateway Port

Sapdp01 3201/tcp #SAP Dialog Service Port

Sapgw02 3302/tcp #SAP System Gateway Port

Sapdp02 3202/tcp #SAP Dialog Service Port

Passing through a Message server

If passing through a message server, an entry (for the Message server Port) may have to be added in the 'services' file. Add sapmsxxx 36nn/tcp, where xxx is the system ID and nn is the system number. If it is the last line in the file, add a new line break after the entry.

As an alternative to updating the 'services' file, you can enter the Message server Port number directly, instead of the System ID, when creating the connector connection. The Port number is of the form 36nn where nn is the system number.

Passing through a Firewall

If there is a firewall between the connector and the SAP system, port 33nn has to be open (where nn = system number of the SAP system).

SAP Transports

The transports must be installed in the correct order (first Data then Role transports). The Data transport is cross-client (import into the base client), whereas the Role transports are client-specific and must be imported into all clients where they will be used.

Import transports

Depending of SAP basis the following options might need to be selected in transaction **STMS**:

H6Q(1)/800 Import Transport Request

Transport Request	E66K900092	Package /QTQVC/QTDEV release 6.5.0 for basis …
Target System	H6Q	H6Q
Target Client	000	Targ.Client=Source Client

Date / **Execution** / **Options**

Import Options

- ☑ Leave Transport Request in Queue for Later Import
- ☐ Import Transport Request Again
- ☑ Overwrite Originals
- ☑ Overwrite Objects in Unconfirmed Repairs
- ☑ Ignore Non-Permitted Transport Type
- ☑ Ignore Non-Permitted Table Class
- ☑ Ignore Predecessor Relations
- ☑ Ignore Invalid Component Version

After import, check the transport log like below. It is OK to receive warnings at import but not errors.

Import Queue: System H6Q

Requests for H6Q: 0 / 31   02.06.2017 10:13:56

| Number	Request	RC	Owner	Short Text	St
1	E66K900035	⚠	HRG	Package /QTQVC/QTDEV release 5.80 SR3 for basis >= 7.20	
2	E66K900036	🟩	HRG	QTQVCACCESS Release 5.80 SR3 and above	
3	E66K900047	⚠	HRG	Package /QTQVC/QTDEV release 5.80 SR4 for basis >= 7.20	
4	E66K900050	⚠	HRG	Package /QTQVC/QTDEV release 6.00 IR for basis >= 7.20	
5	E6DK900712	🟩	HRG	RFC Server	
6	E6DK900716	⚠	HRG	Release 6.10	
7	E6DK900720	🟩	HRG	Get table info	
8	E6DK900722	🟩	HRG	Get table info	
9	E6DK900724	🟩	HRG	Get table info	
10	E6DK900726	🟩	HRG	Get table info	
11	E6DK900728	🟩	HRG	Get table info	
12	E6DK900730	🟩	HRG	Get table info	
13	E6DK900732	🟩	HRG	Get table info	
14	E66K900058	⚠	HRG	Package /QTQVC/QTDEV release 6.1 IR for basis >= 7.20	
15	E66K900059	🟩	HRG	QTQVCACCESS Release 6.1 IR and above	
16	E66K900063	⚠	HRG	Package /QTQVC/QTDEV release 6.1.1 for basis >= 7.20	
17	E66K900069	⚠	HRG	Package /QTQVC/QTDEV release 6.1.2 for basis >= 7.20	
18	E66K900066	🟩	HRG	QTQVCACCESS Release 6.1.2 and above	
19	E66K900070	⚠	HRG	Package /QTQVC/QTDEV release 6.1.2 for basis >= 7.20	
20	E66K900071	⚠	HRG	Package /QTQVC/QTDEV release 6.3.0 for basis >= 7.20	
21	E66K900072	⚠	HRG	Package /QTQVC/QTDEV release 6.3.2 for basis >= 7.20	
22	E66K900075	⚠	HRG	Package /QTQVC/QTDEV release 6.3.3 for basis >= 7.20	
23	E66K900076	🟩	HRG	QTQVCBWACCESS Release 6.3.3 and above	
24	E6DK900834	⚠	HRG	Move dictionary objects	
25	E6DK900827	⚠	HRG	Move test tables	
26	E6DK900835	🟩	HRG	Transport table entries	
27	E6DK900839	🟩	HRG	Move upload program	
28	E6DK900840	🟩	HRG	Update NULLVALUE	
29	E66K900080	⚠	HRG	Package /QTQVC/QTDEV release 6.4.0 for basis >= 7.20	
30	E66K900081	⚠	HRG	Package /QTQVC/QTDEV release 6.4.0 for basis >= 7.20	
31	E66K900092	⚠	HRG	Package /QTQVC/QTDEV release 6.5.0 for basis >= 7.20	

Put the cursor on the transport and press the 'Logs' button .

The log should look something like this (no errors should occur):

Overview of Transport Logs

Display restricted to target system H6Q

Log Overview for E66K900092 (HRG)

    E66K900092   Package /QTQVC/QTDEV release 6.5.0 for basis >= 7.20

       H6Q   H6Q

Selection for Import	02.06.2017 10:08:39	(0)	Successfully Completed
Import ABAP Dictionary Objects	02.06.2017 10:09:55	(4)	Ended with Warning
ABAP Dictionary Activation	02.06.2017 10:09:58	(4)	Ended with Warning
Import	02.06.2017 10:10:18	(4)	Ended with Warning
Check Versions	02.06.2017 10:10:20	(0)	Successfully Completed
Method Execution	02.06.2017 10:10:23	(0)	Successfully Completed
Generation of Programs and Screens	02.06.2017 10:10:27	(0)	Successfully Completed

Import steps not specific to transport request

Errors at import of transports

Error message like: ERROR: '/usr/sap/trans/cofiles/K900029.E6D' : can't open.

Make sure that the Read-only attribute of the 'cofiles' file is NOT selected like:
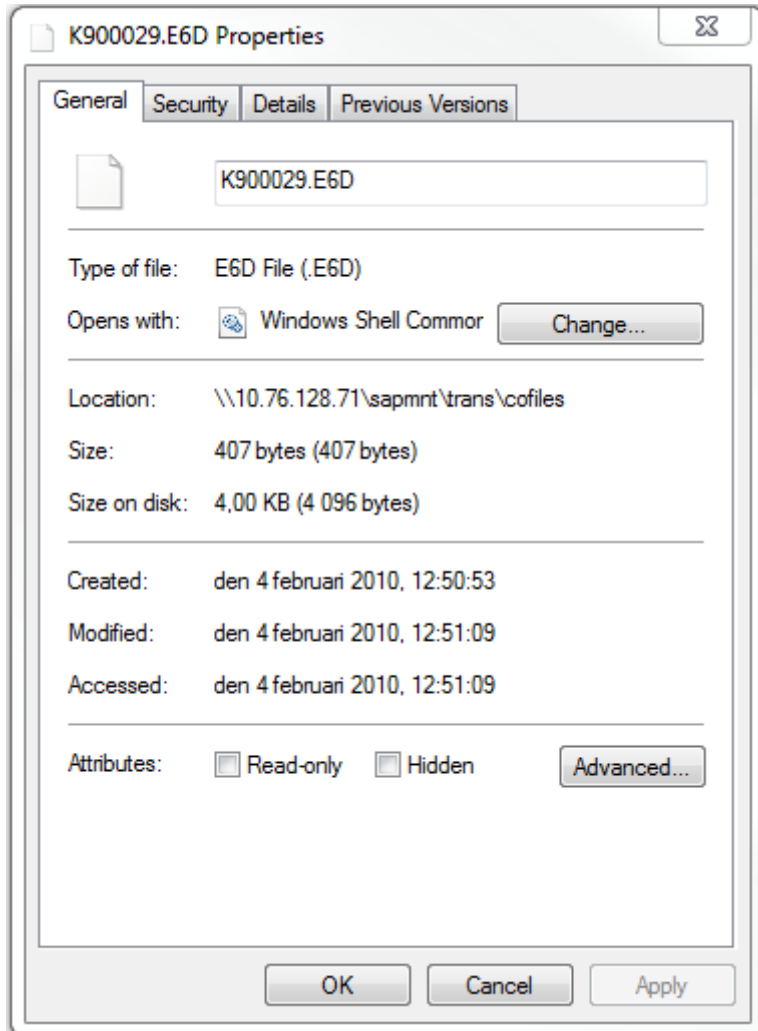
Using Logon Groups for Load Balancing

To achieve load balancing in the SAP system it is possible to use a Logon Group when creating the connection. Select Server Type 'Message Server', enter the Message Server Host, Client, System ID, Logon Group, Username and Password.
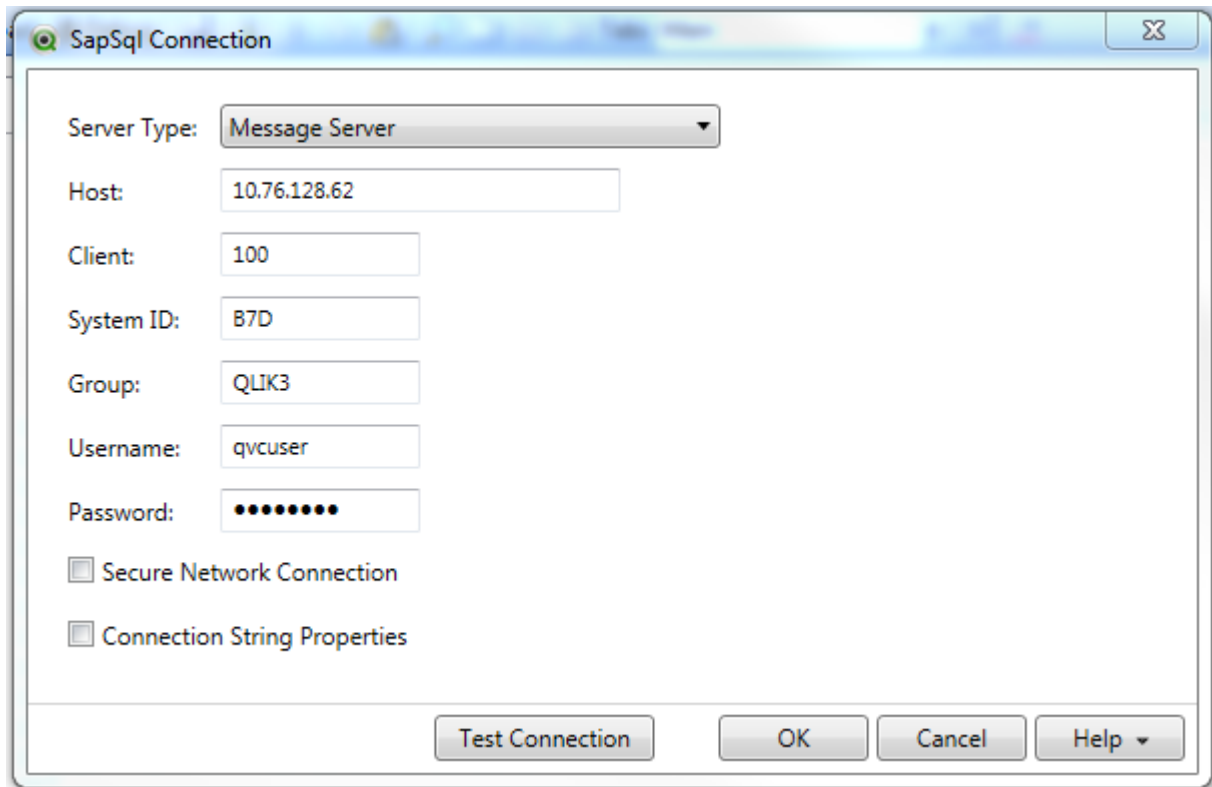
The appropriate server (depending on the current load in the system) from the Logon Group will then be used to execute each job in the script.

Please read the SAP documentation about how to set up Logon Groups.

The Qlik database table /QTQVC/STATUS

This table contains information about all jobs started for all connectors. Use transaction SE16 to view it.

The Qlik database table /QTQVC/CONTROL

This table contains information about all jobs started for the SQL connector. Use transaction SE16 to view it.

Table logging

In the technical settings of the Qlik database tables /QTQVC/CONTROL, /QTQVC/STATUS, /QTQVC/TABLECONT and /QTQVC/USERCONT the flag 'Log data changes' is on.
The logging flag defines whether changes to the data records of a table should be logged. If logging is

activated, every change (with UPDATE, DELETE) to an existing data record by a user or an application program is recorded in a log table in the database.

If table logging is not wanted, it is possible to contact Qlik Support and receive a patch containing an SAP transport. When importing the transport, table logging is switched off.

If the version of Qlik Connector for SAP is upgraded later, the patch transport needs to be re-imported.

Error when creating a connector Connection

An error message like:
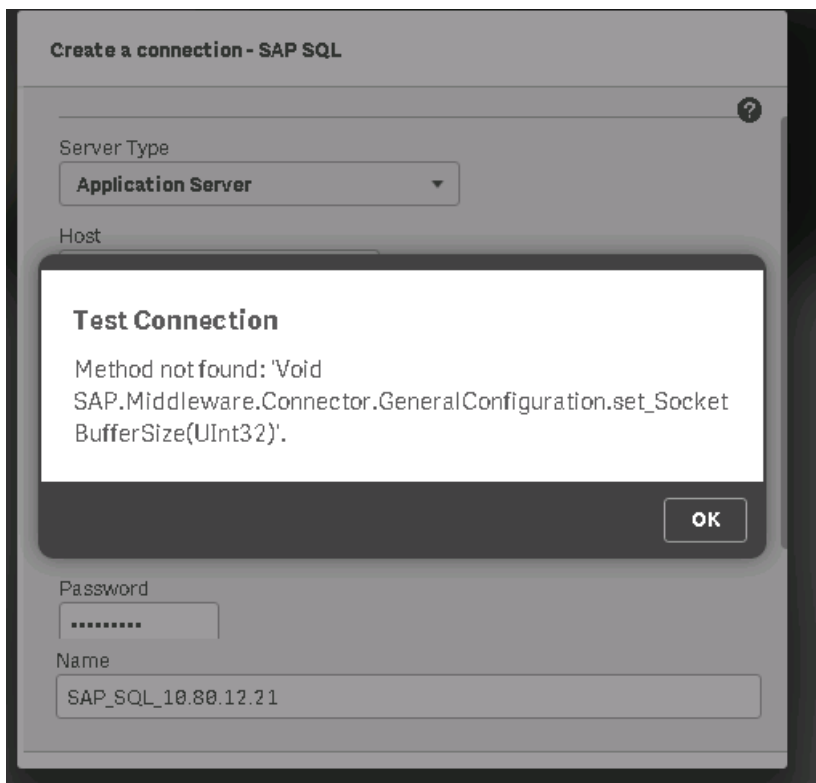
Cause: A '.NET' driver called 'sapnco.dll', different than the one Qlik provides, has been installed to GAC (Global Assembly Cache). Check: %windir%\Microsoft.NET\assembly.

Solution: Install the driver provided by Qlik into GAC. This may impact the application that installed the driver into GAC in the first place, but putting the original driver next to the binaries for the affected application should take care of that problem.

This error can for the moment occur when creating a connection for the SQL, Bex Query and InfoProvider connectors.

BAPI connector

BEx Query connector

FU_NOT_FOUND: Function Module /QTQVC/CHECK_RELEASE_BW

An error message like above indicates that the BEx Query connector transport is not imported correctly. Please see the Online help for more details about importing the BEx Query connector transports.

DSO connector

Extractor connector

Scripting

Best practice is to have one connection string per script. If a script contains many tabs it is enough to have one connection string in the first tab.

Qlik SAP Network Server

Do not start more than one Service 'Qlik SAP Network Server' pointing to the same SAP system. It will result in lost data packages or IDocs.

Error: WRONG DATA SELECTION reported from SAP , rqState5: DataSource &1 has to be replicated (time stamp, see long text)

Try this procedure:

1. Deactivate the Extractor in transaction /n/QTQVC/DEACTIVATE

2. Replicate in transaction RSA1

3. Activate the Extractor in transaction /n/QTQVC/ACTIVATE

Please note that reactivation of an extractor requires a reinitialization of the delta (if used).

Error: WRONG DATA SELECTION reported from SAP

Check that the correct authorization roles are assigned to the SAP user ID which is used in the connection string. In all types of SAP systems, the following Qlik roles should be assigned: QTQVCACCESS, QTQVCADMIN, QTQVCEXTRADM, QTQVCEXTRSETUP and QTQVCEXTRACTOR. In a BW system, an additional role should be assigned: QTQVCBWACCESS.

Troubleshooting of the Extractor connector

Some good guidelines for testing the Extractor connector:

1. Start with a clean system. No Extractor jobs should be running (check in transaction SM51). From the connector make sure to remove any active status records.

2. Restart the Extractor connector Service.

3. Execute one Extractor job at a time. When an error occurs, check the Extractor connector log (named like QvSAPExtr_20160830_122350.txt) and the Service log (named like SrvService_20160614_091022.txt or Server_20160516_111212.txt) and analyze the error. The Service log is located on the server where the Service is running (see section 'Connector log files' in this document.

4. Also check the corresponding SAP extraction job log. Prefix the job name from the Extractor connector log with 'BI' like BIREQU_QTQVCEXTR1_20160420125150. Check job log for this job name in transaction SM37.

5. Continue like above and try to resolve one error at a time.

Database error 1555 with FET access to table xxx

A message like above can be found in the system log (transaction SM21). It can be followed by a message like:
ORA-01555: snapshot too old: rollback segment number 24 with

Database used is Oracle.

A message like this can be found in the SAP extraction job log (transaction SM37):
The internal session was terminated with the runtime error DBIF_RSQL_SQL_ERROR (see ST22)

Solution:
Increase the Oracle redolog retention and PSAPUNDO tablespace.

Error when extracting big data volumes from an Oracle database

Error message in the Extractor Service log (for example: Server_20160325_103224.txt). The log is found on the server where the Service is running, under: C:\ProgramData\QlikTech\Custom Data\QvSAPConnector\Log.

Error message such as:
Error: WRONG DATA SELECTION reported from SAP , rqState5: &1 &2 &3 &4

Error message in the SAP Extraction job log (transaction SM37). To get the SAP job name, prefix the Job name from the Extractor connector log with 'BI', for example: BIREQU_QTQVCEXTR1_20160408072050.

Error message such as:

Beginn_Lock_DELETE:	*20.160.408.162.541,7965790*	*RSAR 051 S*
Before enq:	*20.160.408.162.541,7966500*	*RSAR 051 S*
After enq:	*20.160.408.162.541,7978420 Shared_buffer:*	*RSAR 051 S*
End_Lock_DELETE:	*20.160.408.162.541,7991240 Lines of Locks: 26*	*RSAR 051 S*

Sometimes there can be an error in the SAP system log (transaction SM21) such as:
Database error 1652 at FET

Explanation regarding Oracle error ORA-1652:

When extracting a large amount of data in a SAP system based on any relational database, data is being gathered together in a temporary table, until an SQL statement commit occurs.

In Oracle databases the table containing temporary data is stored in a tablespace named PSAPTEMP. All tablespaces in an Oracle database have a fixed size (until they are extended) measured in Gigabytes. If a tablespace is running full (no more available space to extend the table), Oracle will throw out database error ORA-1652 and roll back the changes.

This can be seen in the SAP SM21 log; "Database error 1652 at FET", at the exact time as the job fails.

There is a file limit size on data files in an Oracle tablespace. The maximum size of a data file depends on the block size of the database. It will eventually come to a point, where a data file cannot be extended any further, and a new data file has to be added to PSAPTEMP.

To resolve ORA-1652 errors with temporary tables, the tablespace PSAPTEMP needs to be extended either by adding another data file or extending the current one. For example increase the tablespace by 10 % and try again. If it is not enough, increase with another 10 % and so on.

More information can be found in the log files "alert<SID>.log" in
UNIX:/oracle/<SID>/saptrace/background/ or
Windows:<drive>:\oracle\<SID>\saptrace\background\.

RFC connection

ERROR registration of tp <program id> from host <external host> not allowed

A RFC connection fails when an external program / application tries to register a program in the gateway with the error message:

LOCATION SAP-Gateway on host <hostname> / sapgw<nr>
ERROR registration of tp <program id> from host <external host> not allowed

Meaning that the gateway is not allowing the program ID <program id> to be registered from the host <external host>

Cause

There are two possible reasons:

- Parameter "gw/acl_mode = 1" and not reginfo file has been set. This is the default value for NEW INSTALLATIONS for Netweaver 7.0x since December 2012 (including 7.20) and for Netweaver 7.3X since August 2012.

- There is a reginfo file that doesn't allow the registration of that external program

Resolution

Update the "reg_info" file from the SAP gateway that appears in the error message ("SAP-Gateway on host <hostname> / sapgw<nr>") creating a line that allows this registration.
The location of the "reg_info" file is specified by parameter "gw/reg_info" and should contain lines like:

#VERSION=2
 P TP=<program ID> HOST=<external host> CANCEL=internal,<external host> ACCESS=*
 ...
the following lines should be the LAST lines in the reginfo
#
P TP=* HOST=<LIST> CANCEL=<LIST> ACCESS=<LIST>
P TP=* HOST=local

Following recommendations from SAP Note 1408081 :Basic settings for reg_info and sec_info
and settings for ACCESS clause from SAP Note 1592493 : GW: Problems during reginfo configuration.

Then, reload the new settings via transaction SMGW. Sometimes it might be necessary to restart the SAP system to activate the changes.

And try to register the program again.

illegal parameter value (function=SAP_CMACCPTP /parameter=handle / value=548

For an error message like:

"illegal parameter value (function=SAP_CMACCPTP /parameter=handle / value=548".

Solution:

Add a line to the "reg_info" file (the location of the "reg_info" file is specified by parameter "gw/reg_info"):

P TP=QTQVCEXTR1 HOST=<Servername>,<FQDN Servername> ACCESS=<SAP Servername>,<FQSN SAP Servername>,internal CANCEL=internal

ERROR hostname 'xxx...' unknown

Enter the IP Address / Server name in the file 'hosts' under path: C:\Windows\System32\drivers\etc.

Restart the Qlik SAP Network Server service.

Many background ARFCXXXX jobs visible in SM37

The purpose of this wiki is to provide details as to why ARFC* jobs may be created on the system.

In this wiki we will look at reasons why these ARFC jobs are created, explain if they can be deactivated and if so, how failed calls can be reprocessed.

Reasons why the ARFC batch jobs are created

There are 2 cases when a batch job will be scheduled:

(1) There is a connection failure and the scheduler automatically creates a batch job for such connection failures.

(2) Application explicitly asks for batch job creation by calling the API 'START_OF_BACKGROUNDTASK' in the same flow that creates the queue entry.

Identifying the jobs

Background jobs with the name "ARFC:<XXXX>" can be seen in SM37:

```
Job overview from:  01.09.2010 at:     :  :
           to:  13.09.2010 at:     :  :
Selected      job names:  ARFC*
Selected user names:        *

☐Scheduled   ☑Released   ☑Ready    ☑Active   ☑Finished   ☑Canceled
☐Event controlled   Event ID:
☐ABAP program      Program name :
```

Job	Spool	Job Doc	Job CreatedB	Status	Start date	Start time	Duration(sec.)
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	15:16:04	2
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	15:31:14	2
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	15:46:25	2
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	16:01:35	5
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	16:16:45	3
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	16:31:56	2
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	16:47:07	3
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	17:02:18	4
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	17:17:28	2
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	17:32:38	3
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	17:47:49	5
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	18:03:00	2
ARFC:0A165C3F02544C8633FB307F			GRBIN	Finished	07.09.2010	18:18:10	3

The job logs shows that these jobs are running report RSARFCSE:

```
Job log overview for job:    ARFC:0A165C3F02544C8633FB307F / 15005500
```

Date	Time	Message text	Message class	Message no.
07.09.2010	15:16:04	Job started	00	516
07.09.2010	15:16:04	Step 001 started (program RSARFCSE, variant &0000000034016, user ID GRBIN)	00	550
07.09.2010	15:16:06	Job finished	00	517

The job RSARFCSE is responsible for asynchronous remote function calls that have not been successful.

For every unsuccessful ARFC found in table ARFCSSTATE a job called ARFC:<xxxxxxx> is created. In table ARFCSSTATE you can see the destination for the ARFC call.

The job will be created if there is a connection failure and the scheduler automatically creates a batch job for such connection failures. So if the tRFC call is in status CPICERR, then the entry will be retried as per settings in tRFC options. If you do not see CPICERR status and instead see SYSFAIL status, then there will not be any automatic retry.

Once you have determined the target system for the ARFC calls (as per table ARFCSSTATE), use transaction SM59 to check whether this is a valid RFC destination.

Possibility to disable the ARFC jobs from being created

It is possible to disable these background jobs if errors occur with the connection. To do this, go to the "tRFC options "for the relevant RFC destination reporting the connection error:
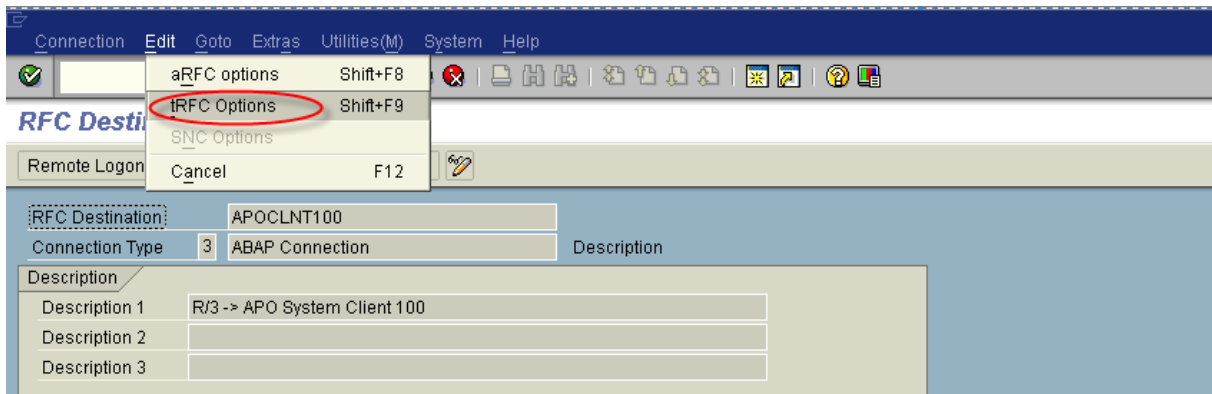
In the next screen choose the "Suppress background job if conn.error" option.
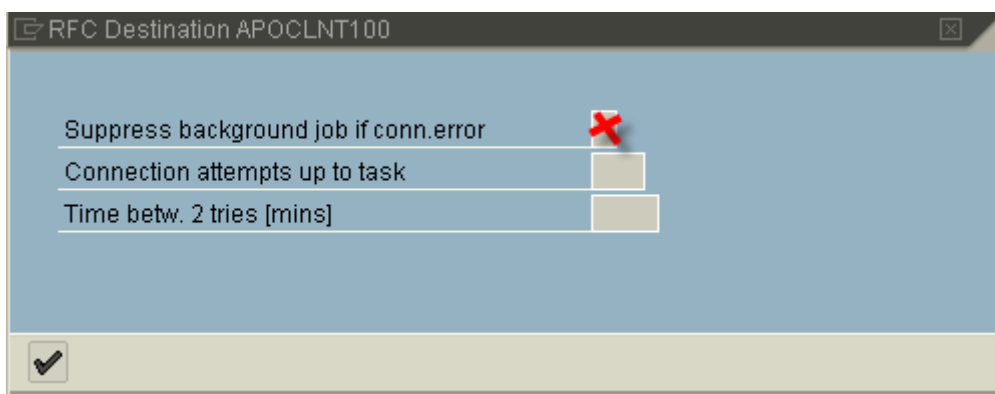
Please note that this option will only work in the case of option (1) above - if the application explicitly asks for the batch job then there is no use to suppress this background job.

The RFC's can then be reprocessed by scheduling report RSARFCEX. You can also delete the already scheduled jobs with report RSBTCDEL if there are too many.

Using the option "Suppress background job if conn. error" will stop the creation of these ARFC:<XXXX> background jobs. You can then reprocess the failed tRFC calls by scheduling report RSARFCEX at regular intervals.

As per option (2) above these batch jobs will also be created if the application explicitly asks for batch job creation by calling the API 'START_OF_BACKGROUNDTASK' in the same flow that creates the queue entry.

To see this, check the ARFCSSTATE table's ARFCRESERV field, here you would see the BATCHPLA flag set (ARFCRESERV is a continuous string, BATCHPLA flag corresponds to the offset as per ARFCRESERV structure (you can open ARFCRESERV in transaction SE11 to see this)).

If this the case, the issue must be checked with the corresponding application

Using the Extractor connector after a Client copy

Steps to perform after a Client copy:

1. Delete the Extractor environment by using the transaction /n /QTQVC/EXTRACTOR_ADM.

2. Use the transactions SM59, WE20 and WE21 to check that all settings for the old Extractor environment have been removed. More information about what settings are used for the Extractor environment can be found in the document 'Extractor Technical Design Specification' located on the Qlik download site.

3. Check in transaction SM51 which server should be used for the RFC transfer.

4. Create a new Extractor environment by using the transaction /n /QTQVC/EXTRACTOR_ADM. From the field 'Server Name for RFC', look up the server name from point 3 and save the environment.

5. Activate the Extractors that should be used, in transaction /QTQVC/ACTIVATE.

Error 'Connection does not exist'

Error message 'Connection does not exist' when trying to activate an Extractor in transaction /n/QTQVC/ACTIVATE. The reason is that the used Logical system environment has been damaged.

Create a new Logical system environment in transaction /n/QTQVC/EXTRACTOR_ADM, with a name not used before. Try to activate the Extractor with the new Logical system.

Error 'The given key was not present in the dictionary'

The error comes from starting the QvSAPExtractorConnector.exe directly (double clicking) – this is not supported.

The connector needs to be used through Qlik Sense. The QvSAPEXTRConnector.dll should be used in QlikView.

OLAP connector

Hierarchy Selection variables in the select dialog in QlikView

The OLAP connector has the limitation that it cannot handle Hierarchy Selection variables in the select dialog in QlikView. There are two possible solutions to this problem:

1. Recreate the BEx query in SAP Query Designer with static filters for the desired hierarchy to get around the variables issue.

2. Manually modify the variables section in the load script in QlikView.

When entering values for Hierarchy Selection variables in the select dialog of the OLAP connector, the selectable values are not correct. The value selection list shows all leaves of the InfoObject instead of the available hierarchies. The same behavior occurs when selecting values for a Hierarchy Node variable that is dependent on a Hierarchy Selection variable. The **variable names** of these will be entered correctly into the script but the variable values will not be correct.

If modifying the load script, the (SAP) technical name of the **hierarchy** and the **hierarchy node** must be known. These names can be retrieved by calling function modules in SAP, transaction SE37, or by calling the same function modules using the Qlik SAP BAPI connector.

Hierarchy: Call function module BAPI_MDPROVIDER_GET_HIERARCHYS with the input parameters CUBE_NAM = <InfoProvider>/<QueryName> and DIM_UNAM = [<InfoObjectName>].

```
Test for function group        RSOB
Function module                BAPI_MDPROVIDER_GET_HIERARCHYS
Uppercase/Lowercase            ☐

RFC target sys:
```

Import parameters	Value
CAT_NAM	
CUBE_NAM	0PCA_C01/ZPCA_REP1_HIER_TEST
DIM_UNAM	[0PROFIT_CTR]
HRY_NAM	
HRY_UNAM	

Tables	Value
HIERARCHIES	▦ 0 Entries

The result is displayed in the table HIERARCHIES, column HRY_UNAM. Copy the content of the column to the Hierarchy Selection Variable, in this example [0PROFIT_CTR   1000H1] (keep the brackets and the blank spaces).

Hierarchy node: Call function module BAPI_MDPROVIDER_GET_MEMBERS with the input parameters CUBE_NAM = <InfoProvider>/<QueryName> , DIM_UNAM = [<InfoObjectName>] and HRY_UNAM = [<InfoObjectName>   <HierarchyName>] (retrieved in the previous call to BAPI_MDPROVIDER_GET_HIERARCHYS).

```
Test for function group        RSOB
Function module                BAPI_MDPROVIDER_GET_MEMBERS
Uppercase/Lowercase            ☐

RFC target sys:
```

Import parameters	Value
CAT_NAM	
CUBE_NAM	0PCA_C01/ZPCA_REP1_HIER_TEST
DIM_UNAM	[0PROFIT_CTR]
HRY_UNAM	[0PROFIT_CTR 1000H1]
LVL_UNAM	
LVL_NUMBER	
MEM_NAM	
MEM_UNAM	
MEM_CAP	
MEM_TYPE	
TREEOP	
ALL_PROPERTIES	
START_ROW	000000
END_ROW	000000
SET_NAME	

Tables	Value
MEMBERS	▦ 0 Entries
OPTIONAL_PROPERTIES	▦ 0 Entries
SELECTED_PROPERTIES	▦ 0 Entries
RESTRICTIONS	▦ 0 Entries

The result is displayed in the table MEMBERS, column MEN_UNAM. Copy the content of the column to the Hierarchy Node variable, in this example
[0PROFIT_CTR   1000H1].[1000HE   0HIER_NODE] (keep the brackets and the blank spaces).

The final result should look like this:

```
Variables (
  [0H_PCTR] (I = [0PROFIT_CTR          1000H1]),
  [0N_PCTR] (I = [0PROFIT_CTR          1000H1].[1000HE          0HIER_NODE])),
From (0PCA_C01/ZPCA_REP1_HIER_TEST));
```

The structure of the field lengths should be like this:

```
                   30 characters
Variables (
  [0H_PCTR] (I = [0PROFIT_CTR          1000H1]),          32 characters
  [0N_PCTR] (I = [0PROFIT_CTR          1000H1].[1000HE          0HIER_NODE])),
From (0PCA_C01/ZPCA_REP1_HIER_TEST));
```

Report connector

Select Options

Some Select Option values have to be in SAP internal format, like material number (S_MATNR in this example). S_MATNR has to be entered with leading zeroes, like 000000000000000521 (18 characters). Script example:
OPTIONS (
 (SELNAME S_MATNR, KIND S, SIGN I, OPTION EQ, LOW 000000000000000521)
 (SELNAME S_MATNR, KIND S, SIGN I, OPTION EQ, LOW 000000000000000578)
 (SELNAME S_MATNR, KIND S, SIGN I, OPTION EQ, LOW 000000000000001268)
));

Date is normally entered like YYYYMMDD. Example: '20180930'. This is independent of the date setting in SAP for the user (which does not affect this input).

Auto Detect Column Width

At the configuration of a Report it might be possible (the check box in the screen below is enabled) to use the function 'Auto Detect Column Width'. In that case select the check box, press the OK button and the script will automatically be generated.

NOTE: For a small number of Reports the generated script will not work properly at Reload. In that case use manual configuration instead.

Test Report connector in SAP

Use transaction SE37:

Enter function module name /QTQVC/CALL_REPORT and Press the test button

Test Function Module: Initial Screen

⊕ ⊕Debugging   ⊠   Test data directory

Test for function group	/QTQVC/CALL_REPORTS
Function module	/QTQVC/CALL_REPORT
Uppercase/Lowercase	☐

RFC target sys:

Import parameters	Value
MODE	E
REPORT	SDORDE01
LINE_SIZE	1.023
DYNNR	1000
VARIANT	ZHRG
WINUSER	HRG
WINIPNR	1.2.3.4

Enter MODE 'E', your REPORT name and VARIANT name (WINUSER and WINIPNR are not necessary)

and press the 'Execute' button ⊕

If successful there will be entries in the tables at the bottom of the screen below. Otherwise there will be an error message in the top part of the screen.

Test Function Module: Result Screen

Test for function group   /QTQVC/CALL_REPORTS
Function module   /QTQVC/CALL_REPORT
Uppercase/Lowercase   ☐

Runtime: 21.874.148 Microseconds

RFC target sys:

Import parameters	Value
MODE	E
REPORT	SDORDE01
LINE_SIZE	1.023
DYNNR	1000
VARIANT	ZHRG
WINUSER	HRG
WINIPNR	1.2.3.4

Tables		Value
LINES		0 Entries
	Result:	1.719 Entries
LINE_BUFFER		0 Entries
	Result:	1.750 Entries
FMBS		0 Entries
	Result:	20 Entries
FMBX		0 Entries
	Result:	7 Entries
FSEL		0 Entries
	Result:	10 Entries
PAGES		0 Entries

SAP query connector

Error 'DATA_TO_MEMORY_NOT_POSSIBLE'

More authorization is needed. Compare SAP note 2212180.
An SAP query might check authorization objects that are not part of the Qlik roles. In that case, they can be added to the role QTQVCACCESS or added to an additional customer-developed role that is assigned to the Qlik SAP user.
A user with SAP_ALL authorization can be used to verify that it is an authorization problem.

SQL connector

How does the connector work

An SQL connector job is started from Qlik by sending an SQL SELECT statement to a function module in the SAP system. The function module performs various checks and handling of the statement before starting up a background (batch) job in the SAP system. The batch job performs the SELECT statement against the database. The result of the SELECT statement is divided into packages with a fixed number of rows in each. One package at a time is then stored in the shared memory buffer EIBUF (Export/Import buffer). The batch job then waits for Qlik to make a call and fetch the content of the buffer. When the content is fetched, the connector clears the used memory and the batch job stores the next package in the buffer. And like that the process continues until the last package is fetched by Qlik.

The memory buffer EIBUF is shared by all programs in the SAP system. The size and other settings of the memory buffer is decided and configured by the SAP BASIS team at each customer and nothing that the Qlik SQL connector can affect. Unfortunately, not all programs clear the memory they have used when finished, so the available space of the buffer is decreasing over time at most customers. The buffer can be cleared by restarting the SAP system or by special commands. However, this should only be performed by the SAP BASIS team.

The SQL connector uses a parameter named BufferPercentage with default value 10. The value can be changed in the connection string of the Qlik script. This value decides how many percent of the free space in the buffer EIBUF that the connector should use to store each package. The connector calculates the amount of memory available and how many rows that can be stored in each package. When the available space of the buffer has decreased too much, there will not be many rows in each package and the performance of the SQL connector gets bad.
In the latest releases of the SQL connector there are some improvements to avoid bad performance. If the free space of the buffer is less than 2 percent of the total buffer size, then the connector always uses 2 percent of the total buffer size by overwriting the oldest part of the buffer. It is also possible to use the parameter BufferPercentage with the value 999. This will force the connector to use 10 percent of the total buffer size, also by overwriting the oldest part of the buffer. Since the SQL connector clears the used memory this will also free 10 percent of the buffer memory when the connector job is finished.

Subselect

The sub select method is primarily developed for selecting data from a cluster table where the condition value(s) are fetched from another table. However, the sub select method can be used for two transparent tables as well.

Please note that the sub select method is not possible to use if the 'row based access control' concept of the Qlik SAP SQL connector is activated. The concept is activated if the database view /qtqvc/vuserc contains records.

The result of the select from the main table is temporarily stored in the ABAP program in an internal table (before it is transferred to Qlik). There is a size limitation of internal tables. SAP writes:
The maximum memory that can be occupied by an internal table (including its internal administration) is 2 gigabytes. A more realistic figure is up to 500 megabytes.
Take this into consideration for big data volumes. Try to avoid SELECT *. Only select the fields necessary.

When using the SUBSELECT method against big database tables, always try to limit the selection with a WHERE condition which is using KEY fields or INDEXED fields. Before using the SUBSELECT statement in a production system, monitor it in a test system to make sure it does not consume too much memory or affect the system performance.

The syntax of the sub select in the Qlik script is like:

SQL SUBSELECT BUKRS BELNR GJAHR BUZEI BUZID AUGDT FROM BSEG WHERE **BUKRS BELNR GJAHR** IN (SELECT **BUKRS BELNR GJAHR** FROM BKPF WHERE BLDAT GE '20070101');

Only one sub select is allowed (within the parenthesis).
The sub select can have condition(s).
One or many comparison fields can be passed.
If the SELECT of the sub table returns duplicates, they are removed before selecting from the main table. So there is no need (and it does not work) to use the DISTINCT addition to the sub select.
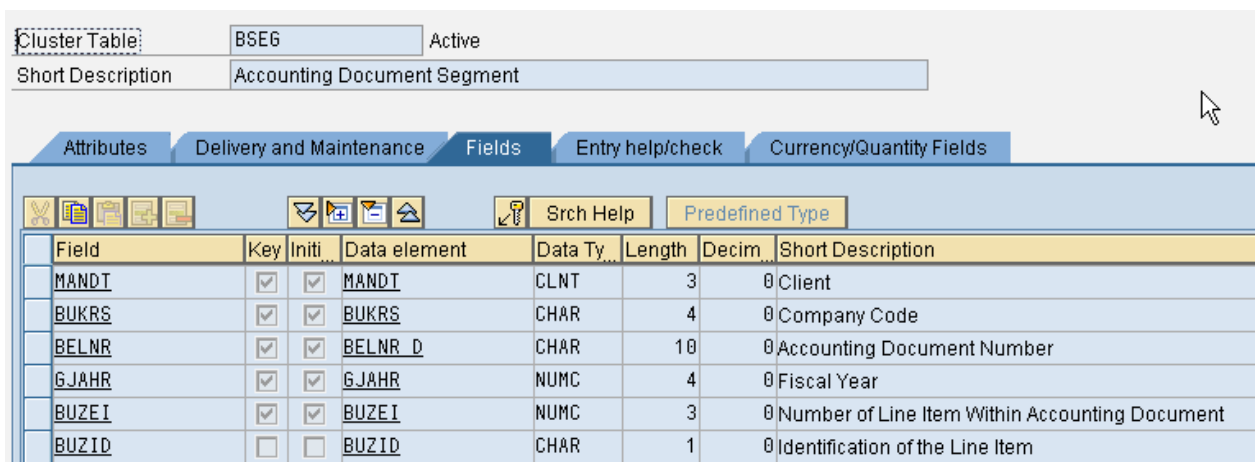
The comparison field or fields (in bold above) has to match between main and sub select. The field names of the comparison fields does not have to be the same in the main and sub select, but the corresponding fields (of main and sub select) must have the same Data Types.

To achieve a good performance it is important to provide as many of the key fields in the main table as possible. Try to select them from the sub table. Most important is to provide the first key field (after Client). In the table BSEG it is BUKRS:

Cluster Table	BSEG	Active
Short Description	Accounting Document Segment	

| Attributes | Delivery and Maintenance | Fields | Entry help/check | Currency/Quantity Fields |

Field	Key	Initi	Data element	Data Ty..	Length	Decim	Short Description
MANDT	☑	☑	MANDT	CLNT	3	0	Client
BUKRS	☑	☑	BUKRS	CHAR	4	0	Company Code
BELNR	☑	☑	BELNR_D	CHAR	10	0	Accounting Document Number
GJAHR	☑	☑	GJAHR	NUMC	4	0	Fiscal Year
BUZEI	☑	☑	BUZEI	NUMC	3	0	Number of Line Item Within Accounting Document
BUZID	☐	☐	BUZID	CHAR	1	0	Identification of the Line Item

If the sub select fails, the reason can often be found in the log for job /QTQVC/READ_DATA in transaction SM37.

Other examples of sub select statements:

SQL SUBSELECT KDATU KAWRT KBETR WAERS FROM KONV WHERE KNUMV IN (SELECT KNUMV FROM VBRK);

SQL SUBSELECT CHANGENR TABNAME TABKEY FNAME CHNGIND VALUE_NEW VALUE_OLD FROM CDPOS WHERE OBJECTCLAS OBJECTID CHANGENR IN (SELECT OBJECTCLAS OBJECTID CHANGENR FROM CDHDR WHERE CHANGENR BETWEEN '0000100000' AND '0000300000');

SQL SUBSELECT TABNAME FIELDNAME POSITION KEYFLAG ROLLNAME CHECKTABLE FROM DD03L WHERE AS4LOCAL = 'A' AND INTTYPE <> '' AND TABNAME IN (SELECT TABNAME FROM DD02L WHERE (TABCLASS = 'TRANSP' OR TABCLASS = 'POOL' OR TABCLASS = 'CLUSTER' OR (TABCLASS = 'VIEW' AND (VIEWCLASS = 'D' OR VIEWCLASS = 'P'))));

Wideselect

SAP database fields with a column width exceeding 800 characters cannot be extracted by the ordinary SQL connector. With the new Wideselect method there is no limitation of column width. It is possible to extract one wide field at a time together with a number of associated key fields. A key field can be any field in the database table with fixed column width. The total column width sum of the key fields cannot exceed 255 characters. The key fields are optional.
The 'wide' field must be of the datatype CHAR (character) or STRG (string).

At present there is no UI syntax support for the Wideselect method.

Please also note that the Wideselect method cannot be used if the row-based access control concept of the SAP SQL connector is activated.

Two Connection string parameters (advanced properties) are applicable for the key fields of the Wideselect:

ConvRoutine=0/1 (default = 0, on = 1): Indicates that output conversion routines are used. This is commonly used for fields like Material Number (MATNR).

Nulldate=0/1 (default/on = 1): If on, date type fields with the value '00000000' are returned as NULL to Qlik.
Nulldate=0, date type fields with the value '00000000' are returned as 12/30/1899 to Qlik.
Corresponding SAP transports send all data to the connector without using any of the conversion functions applied to the SAP system. Any data that is not valid for the datatype is interpreted in Qlik as NULL.
It is possible to display a NULL value as a selectable value by using the Qlik variable **NullDisplay**.

The advanced properties "RemoveAllBlanks" and "ReplaceNullValue" do not have any effect on fields loaded with Wideselect. To handle blank field values and/or null values coming from the SAP system,

it is recommended to use Qlik's built-in script functions in a preceding load statement, regardless if you use Wideselect or a regular Select statement.

Script syntax:

SQL WIDESELECT keyfield1 keyfield2 keyfieldn WIDE(widefield) FROM table WHERE <condition>

Example:

SQL WIDESELECT object joining_date WIDE(test_string) FROM zhrgtest WHERE connector = 'WIDE12000';

CDSSELECT

Another addition to the standard SAP OPEN SQL syntax in the SQL Connector is CDSSELECT. A system with SAP basis 740 SP08 or higher, is required to use the functionality.

It is possible to select CDS views with parameters.

A CDS view has got a name in the ABAP Dictionary. It also has got a DDL (Data Definition Language) Source name. For the CDSSELECT, the DDL Source name should be used.

The syntax of CDSSELECT in the script is as follows:

SQL CDSSELECT [*] [fieldname_1, fieldname_2, fieldname_n]
FROM 'CDS view name'(
parameter_1 = 'X',
parameter_2 = 'X',
parameter_n = 'X')
WHERE [condition];

- If there are parameters, the left parenthesis must come on the same row, immediately after the CDS view name (no white space). Following the left parenthesis, there should be at least one white space before the first parameter. After the last parameter there should be one white space before the right parenthesis.

- The fieldnames and parameters are comma separated.

Examples:

CDS view with parameters:

SQL CDSSELECT vbeln, erdat, erzet, netwr, waerk, telf1
FROM zcds_test_param2(
p_vbeln = '0000006521',
p_erdat = '20001018',
p_mandt = '800');

CDS view without parameters:

SQL CDSSELECT *
FROM zcds_test_param;

The CDSSELECT method cannot be used if the row-based access control concept of the SQL
Connector is activated

The available Connection String Properties are the same as for the ordinary SQL connector.

At present there is no UI syntax support for the CDSSELECT functionality.

Currencies not using 2 decimals

Currency values are stored in the SAP database in fields with the data type CURR (stored as datatype
DEC with 2 decimals). However not all currencies are using 2 decimals. First find the corresponding
currency key. For example, in the database table VBAP there is a currency field NETWR and the
corresponding currency key is found in the field WAERK.

Field	Key	Ini...	Data element	Data Type	Length	Decimal Places	Short Description
NETWR	☐	☐	NETWR_AP	CURR	15	2	Net value of the order item in document currency
WAERK	☐	☐	WAERK	CUKY	5	0	SD Document Currency

Check if the currency key is present in the database table TCURX. The number of decimals used for
the currency is stored in the field CURRDEC. If not present in the table, the currency has two decimals
and no conversion is necessary.

CURRKEY	CURRDEC
0	0
1	1
10	10
11	11
12	12
13	13
14	14
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
BEF	0
ESP	0
HUF	0
IDR	0
ITL	0
JPY	0
KRW	0
KUD	3
LUF	0
PKT	3
RUR	0
TJLP	5
TND	3
TWD	0
USDN	5

To get the correct value of VBAP-NETWR if the currency is found in table TCURX:

If number of decimals = 0, multiply the value of VBAP-NETWR with 100.

If number of decimals = 1, multiply the value of VBAP-NETWR with 10.

If number of decimals = 3, multiply the value of VBAP-NETWR with 0.1.

If number of decimals = 4, multiply the value of VBAP-NETWR with 0.01.

And so on.

Script template:

```
[DataFromSAP]:
Load
      Num(NETWR,'########.##') as [Value],  //Format the field so that it looks
like the default format with 2 decimals in SAP
      WAERK  as Currency;
SELECT NETWR WAERK FROM VBAP;

//Join the currency table onto the table where you want to process a currency
field.
Left Join Load
      CURRDEC as NoOfDecimals,
      CURRKEY as Currency;
SELECT CURRDEC CURRKEY FROM TCURX;

//In separate table, create shifter values based on the number of decimals.
```

```
ShifterTable:
Load
      *,
      If(NoOfDecimals=0,100,
      If(NoOfDecimals=1,10,
      If(NoOfDecimals=3,0.1,
      If(NoOfDecimals=4,0.01,
      If(NoOfDecimals=5,0.001,
      1))))) as Shifter
Resident DataFromSAP;


NewTable:
Load
      Currency,
      Value,
      Num(Value*Shifter) as NewValue //A simple multiplication of the shifter
value that was set based on the number of decimals
resident ShifterTable;

//After creating a new table and processing the data, these tables are no longer
needed and can be dropped
DROP Tables DataFromSAP, ShifterTable;
```

The table TCURX can be changed from the transaction OY04 or use transaction SM30 with table TCURX and press the Customizing button.

Troubleshooting of the SQL connector

Create and find Trace records

To activate the trace, add the Connection String Property **;trace=1;** (from release 6.1 it also works with **;trace=true;**) to the connection string in Qlik (example):

CUSTOM CONNECT TO
"Provider=QvSAPConnector.dll;ASHOST=10.77.1.18;SYSNR=06;CLIENT=800;KeepCasing=1;**trace=1**;log
=1;NullDate=1;XUserId=eMbOHUJOHbQB;XPassword=TYbYHUdNMLZSGXFGOF;";

Then reload the Qlik script.

Note that it is not recommended to have the Trace activated all the time since it has a negative impact on performance. Also, note that database records are created in the Qlik database table /QTQVC/TRACE. For a SELECT of 10 000 records, using a PacketSize of 600, about 100 trace records are created. Therefore, a Trace might not be suitable for SELECT statements which are resulting in very large amounts of records.
It is possible to delete old Trace records from the database by using the transaction **/n/qtqvc/delete** in the SAP system. From release 6.3.0 it is also possible to delete records in the background by using transaction **/n/qtqvc/delete_bgd**. This transaction is suitable to use when deleting large amounts of records.

To retrieve the Trace records, first find the Jobnumber in the SQL connector log:

Then go to the first screen of transaction SE16. Enter the Jobnumber in the field JOBNUMBER and press Execute button (Remember to enter a big value in the field 'Maximum No. of Hits' to avoid limitations in the number of displayed records). Enter a big value in the field 'Width of output list' to be able to view all columns.
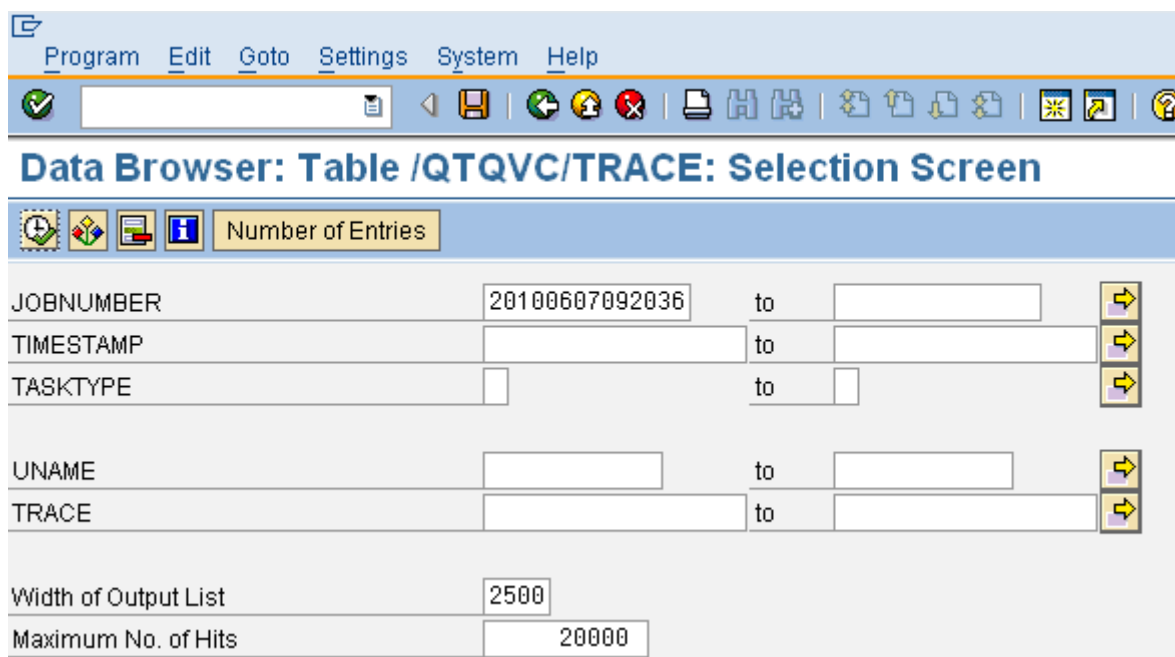
Press the 'Execute' button.

Point to the TIMESTAMP column.

Press the button ⬛ (Sort in ascending order).

The column TRACE now contains information about the job in chronological order. TASKTYPE shows which Function module or program that was involved in the step:
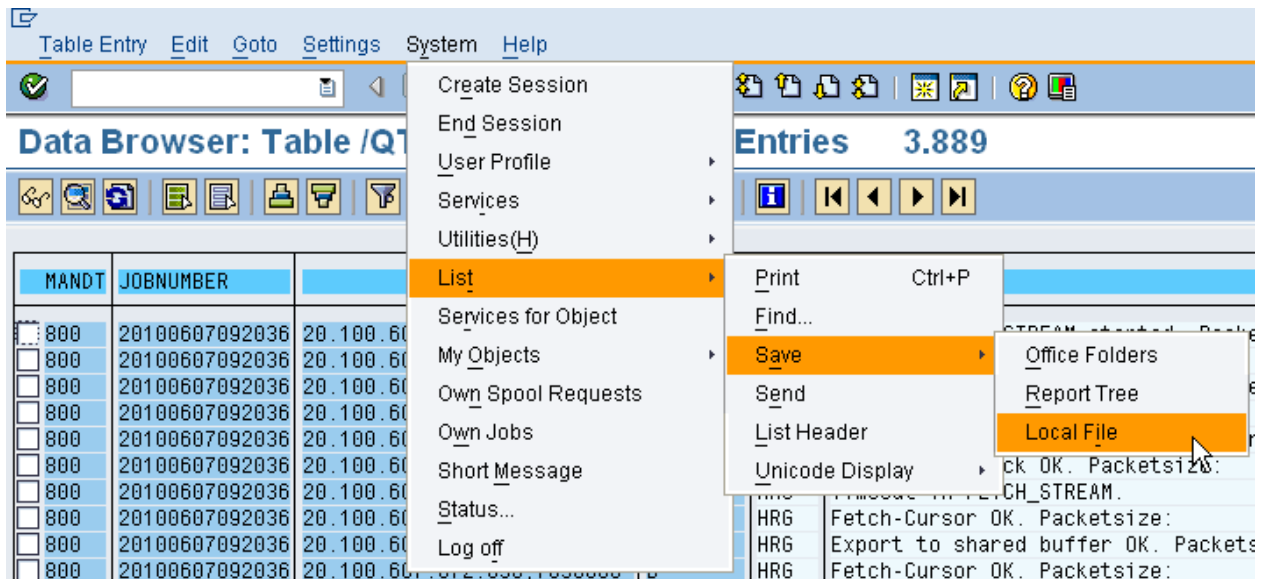
O = 'Open stream' (the function module which performs the first steps in each job).

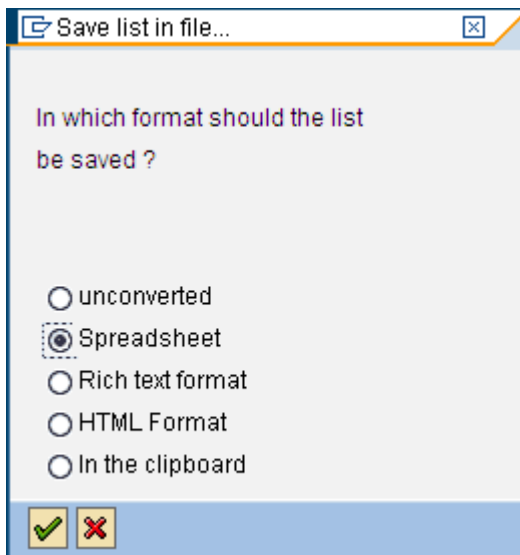B = 'Batch job' (the program that does the fetch of data from the database).

F = 'Fetch stream' (the function module used to get the data and send it to Qlik).

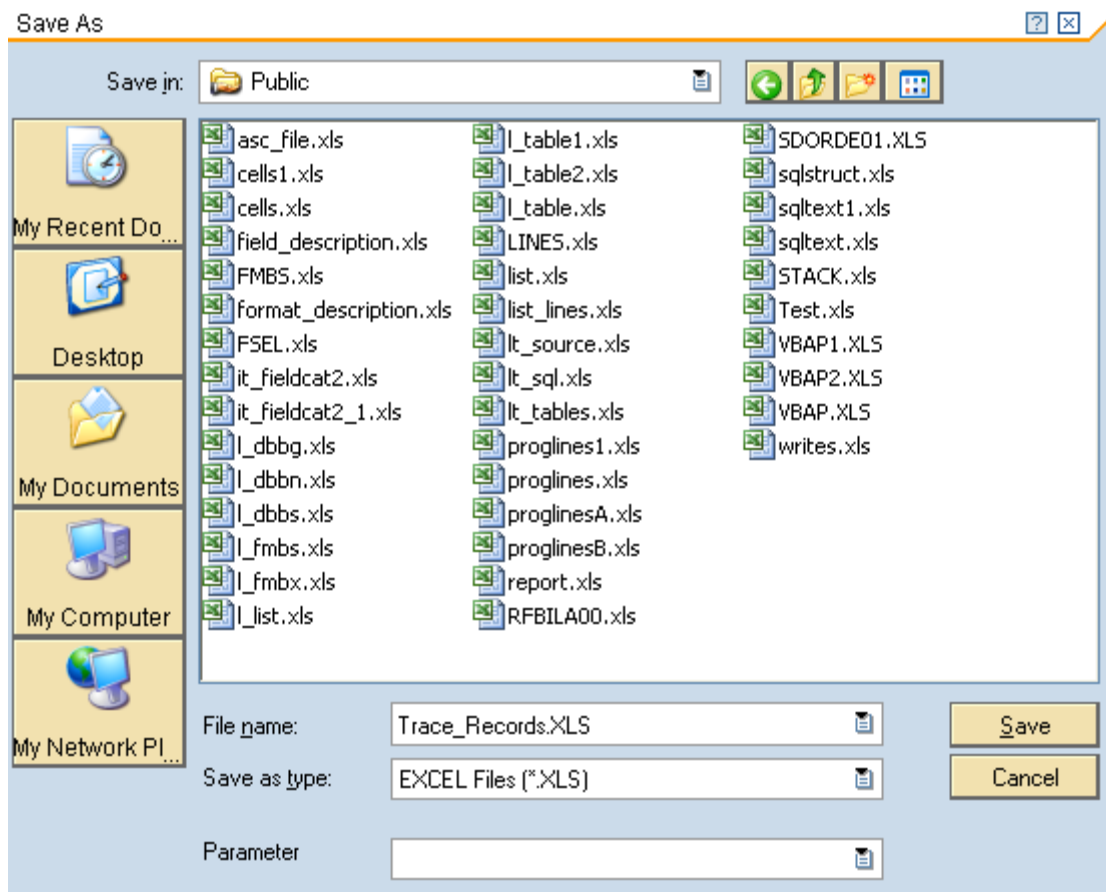Analyze the information in the Trace records.

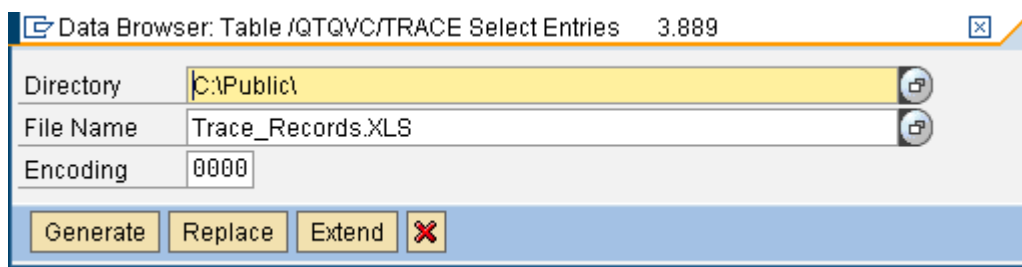The records can now be saved to an Excel file (System/List/Save/Local File):

Select 'Spreadsheet' and press Enter.

Save the file on your computer.

Press the 'Generate' button.

Connection String Properties

The Connection String Properties can be useful to modify the behavior of the connector. For example to get a preview of the resulting records of a SELECT statement, set the properties **FetchBuffers=1** and **PacketSize=100**. This will fetch 100 rows when the statement is reloaded.

Read more about the Connection String Properties for the SQL connector in the Online help under: **Using the SAP SQL connector**

Fetch aborted due to incorrect number of rows

Error message like: **Fetch completed with mismatching number of rows, expected 17876955 rows but read 17876867 rows!**

An error like above indicates that the parameters affecting the shared Export/Import buffer in the SAP system are not configured correctly.

Check the SAP note 702728 - Profile parameters for export/import buffer instances.

NOTE: when the parameters have been changed, restart the SAP system to activate the new values.

The following parameters are important:

rsdb/obj/buffersize

Syntax for correct size: PHYS_MEMSIZE * 1024 * 0,01.

For example, if the server has 35 Gb of memory, the correct value should be 358400.

rsdb/obj/max_objects

Syntax for correct size: rsdb/obj/buffersize / 4.

Correct value based on rsdb/obj/buffersize of 358400 = 89600.

rsdb/obj/large_object_size

Default: 8192. Because this default value has proven to be too small, it has been changed as follows in kernel patches:

- As of kernel Release 7.40 PL 68, 7.41 PL 35, and 7.42 PL 3, the default has been increased to 250000.

The parameter values can be checked in transaction RZ11 or in transaction ST02.