

**WINSHUTTLE™**

# TRANSACTION Advanced Guide

Version 10.3

## Legal

The Winshuttle TRANSACTION help file is for informational purposes only. WINSHUTTLE MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

The software contains proprietary information of Winshuttle, LLC.; it is provided under a license agreement containing restrictions on use and disclosure and is also protected by copyright law. Reverse engineering of the software is prohibited.

Due to continued product development this information may change without notice. The information and intellectual property contained herein is confidential between Winshuttle, LLC. and the client and remains the exclusive property of Winshuttle, LLC. Except as expressly provided in any written license agreement from Winshuttle, the furnishing of this document does not give you any license to patents, trademarks, copyrights, or other intellectual property rights covering subject matter contained in this document.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior written permission of Winshuttle, LLC.

If you find any problems in the documentation, please report them to us in writing. Winshuttle, LLC. does not warrant that this document is error-free.

© 2000-2012 Winshuttle, LLC. All rights reserved.

QUERY, TRANSACTION, SERVER, RUNNER, CENTRAL, Winshuttle Function Module, and Adaptive Query Throttling are trademarks of Winshuttle, LLC. in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

## Contact Information

For more information, please contact: 1-800-711-9798, or visit our Web site at [www.winshuttle.com](http://www.winshuttle.com)

# Contents

<b>TRANSACTION Advanced Guide</b>	<b>7</b>
New features in TRANSACTION version 10.3.....	9
<b>TRANSACTION Assistance</b>	<b>11</b>
Winshuttle Customer Support.....	11
Finding out which build you are using .....	11
Contacting Customer Support .....	12
Winshuttle Community .....	12
Using the Customer Center .....	12
<b>Setting up for use</b>	<b>13</b>
Auto update.....	13
Auto logon .....	13
Application defaults .....	14
SAP defaults.....	16
Error management and trace files .....	17
Client Connection Settings.....	18
Restore defaults .....	19
Enable GUI scripting.....	19
Enabling Scripting on the Server Side .....	19
Method 1.....	19
Method 2.....	19
Enabling Scripting in your SAP client GUI.....	20
SAP GUI 720 .....	21
Unicode implementation for TRANSACTION scripts.....	22
Production Server Warning (Prod. Svr. Warning) .....	24
Winshuttle Function Module for TRANSACTION .....	24
<b>Preparing to record and recording SAP transactions</b>	<b>27</b>
Using SSO and EP logon .....	27
Locating 64-bit SSO libraries on the computer .....	27
EP Logon .....	28
Production versus non-production servers .....	29
Recording modes.....	29
Properties .....	31
Using templates in TRANSACTION .....	32

## Winshuttle TRANSACTION Advanced User Guide

Recording long text .....	33
Backing up data and files in Winshuttle.....	33
Backing up the recording.....	33
Backing up SAP data.....	34
Preserving log files before uninstalling TRANSACTION.....	34
<b>Advanced mapping</b> .....	<b>35</b>
Editing in the Mapper.....	35
Cell-based mapping and column-based mapping.....	36
Filtering mapper setting .....	36
Advanced looping capability .....	37
Do...While Loop.....	38
Preparing your workbook for indented data.....	39
Loop data on multiple sheets .....	41
Loops and changing from cell-based mapping to column-based mapping.....	42
Using the Do...While Loop Feature with Access .....	42
Using Multiple and Nested Loops .....	44
Limiting input fields .....	45
Mapping Google spreadsheets .....	46
Setting conditions.....	46
Validating data .....	48
Validate by list.....	50
Compensating for missing SAP screens and fields .....	51
Undo and Undo All.....	53
Mapping long text .....	54
Editing a TRANSACTION script.....	54
Auto mapping .....	55
Auto mapping from the three-panel screen .....	55
Auto Mapping from the Mapper .....	56
Downloading from SAP .....	57
<b>Running scripts</b> .....	<b>59</b>
Running multiple or linked files.....	59
Scheduling runs.....	60
Field Definitions .....	60
Run options and troubleshooting errors .....	62
Advanced run settings.....	63
Writing Headers.....	65
Publishing macros in an Excel workbook.....	66

Reviewing transaction history .....	68
ROI chart .....	69
Uploading long text .....	70
Windows scripting.....	70
<b>Programs used with TRANSACTION</b>	<b>73</b>
<hr/>	
Winshuttle DIRECT .....	73
A1 Dataloader .....	74
Using TRANSACTION with Virsa Firefighter.....	77
Recording the transaction .....	77
Running the transaction.....	77
<b>Running TRANSACTION at the command line</b>	<b>79</b>
<hr/>	
TRANSACTION Script Command Reference .....	79
TRANSACTION and the command line .....	95
<b>Messages</b>	<b>97</b>
<hr/>	
<b>Troubleshooting</b>	<b>105</b>
<hr/>	
After mapping, can I go back to my original recording? .....	105
Use Transaction to extract from SAP.....	105
Commit code caution .....	105
Can I toggle between batch and non-batch modes while running a script?.....	105
Double-clicking in the Mapper is not making the field editable. What can I do? .....	105
<b>Index</b>	<b>107</b>
<hr/>	



# TRANSACTION Advanced Guide





## New features in TRANSACTION version 10.3


- ROI charts of the activities on your TRANSACTION deployment can be created from the Log Viewer screen.
- Runs can be faster and cleaner with the Run Excel in the background setting, which ensures that data files run in the background.
- Script metadata now includes the version of TRANSACTION that was used to create the script.
- Linked and multiple files that are in production can be updated.
- Autopost is available for password protected Excel files.









## TRANSACTION Assistance

You can click any of the buttons on the Help tab to obtain assistance. The Help tab selections appear on all TRANSACTION screens.

On the TRANSACTION Main Menu, the Help toolbar allows you to:

- Click Help button  at anytime to open this Help file to learn how TRANSACTION works.


The Help file allows you to:

- Click Print button  to print a selected topic or to print the selected heading and all subtopics.
- Click Release Notes button  to get more information about the latest product release including installation and set-up instructions.
- Click Check for Updates button  to check for recent product updates and related downloads such as for security and critical updates. You can also set up Transaction to automatically check for available software updates using the Auto Update option. See "Auto Update" for more information.
- Click Generate Support Information button  to generate an XML file of information that Winshuttle Support might ask you for to better assist you to use Transaction.
- Click Launch Connect Site button  to open the Winshuttle Connect site to update your Connect information, such as your password.
- Click About button  to view product version and licensing information. Click **Exit** to return to the previous screen.

## Winshuttle Customer Support

Winshuttle provides a full range of self-service options to assist you in using TRANSACTION effectively. You can also contact Winshuttle Customer Support directly.

## Finding out which build you are using

Click About button  to view product version and licensing information. Click **Exit** to return to the previous screen.

## Contacting Customer Support

Winshuttle offers fast and friendly support.

If you experience any difficulty in using TRANSACTION, you can contact us by e-mail at [support@winshuttle.com](mailto:support@winshuttle.com). Please describe the specific issue you encountered, the conditions under which it occurred, and attach any supporting information such as screen shots and other documentation.

### To contact Winshuttle directly

In the US, call the Winshuttle Support Hotline at 1-800-711-9798, and then choose option 2

In the UK, call Winshuttle at: +44 208-704-4170 Option 2

## Winshuttle Community

You can check the Winshuttle community site for questions or issues related to TRANSACTION. The Winshuttle community connects you to other TRANSACTION users by allowing you to search existing topics and questions or to enter your own questions. Go to [www.winshuttle.com/community](http://www.winshuttle.com/community).

## Using the Customer Center

You can access the online Winshuttle Customer Center at [www.winshuttle.com/Support-and-Training](http://www.winshuttle.com/Support-and-Training) at any time to:

- Access Winshuttle Knowledge Base articles.
- Add support cases and review the status of your open support cases.
- Get information about training and other events.

## Setting up for use

### Auto update

Use Auto Update to determine when TRANSACTION automatically checks for available software updates:

- On the Main menu, click Tools, and then click Options . In the list, click AutoUpdate.

#### To check for updates

- Click to select any one of the Auto Update settings to check for updates: At startup, After (the number of) days you specify, or Manually.

#### To manually check for available updates at any time

- Click Check for update now.

### Auto logon

Use ALF Management to create an Automated Logon File (ALF) to automate your logon to SAP. An ALF can be used to schedule a run or when you want to use the same logon information each time you run one of your scripts. When you save credentials, you are not asked to provide a user name and password for that system and are automatically logged in to the system.

- Click the Tools tab, and then click Options. Click ALF Management.

#### To create or modify an ALF

1. Choose one of the following responses:

Click Capture to create a new ALF file. When the SAP SERVER logon dialog box is displayed, type the logon information for SAP to capture the ALF information automatically. This is the easiest and error-free way to create an ALF. You can also mark the Use Logon Group checkbox and type the Logon Group and Message Server Host settings to save a logon group in the ALF.

-or-

Click Open to display an existing ALF file to make changes to the file. Click Open from CENTRAL to show an existing ALF on the server. Click Test to perform a test of your log on information. A message is displayed to confirm that your log on was successful.

2. Click Save to store the ALF when you are finished. Click Save to CENTRAL to save your ALF to the CENTRAL server.

NOTE: ALF files are encrypted, which makes it difficult to open and read your SAP password and other information. However, if the ALF file is accessed, it could be used to log on to SAP by any other person who has authorization to use Transaction. For this reason, we strongly recommend that you protect your ALF file, either by using Windows system encryption to prevent others from using it under their user name, or by storing the file on removable media that is password protected.

## Auto logon details

**SAP System Name, Applications Server Host, and System Number:** Capture or type the SAP system information for the ALF.

**Message Server Host:** Capture or type the message server host information.

**Client, Username, and Password:** Capture or type the credentials for the auto logon details.

**Language:** Capture or type the default language for the auto logon.

**Use SSO1:** Select the SSO1 option if you want to log on by using your Windows identity.

**Use Logon Group:** Mark the check box to enable the Logon Group and Message Server Host and type this information. Leave this check box blank to enable the application server host information to be used.

**Logon Group, Partner, and Enterprise Portal Address:** Capture or type this information.

**Auto Logon File:** The location of the auto logon file that is currently saved.

## Application defaults

Use Application Defaults to set the defaults for the TRANSACTION application:

- Select Tools from the Main Menu, and then click (Options button) to display the Options dialog box with the Application Defaults active.
- You can also click Application Defaults on the left of the Options dialog box at any time to display the Applications Defaults.

The Application Defaults settings contain:

### Path defaults

You can either click the (folder icon) on the right to browse to the default file location you want to select, or type any of the default file paths. These paths indicate where specified types of TRANSACTION files are stored, including the data source files (Excel or Access), script files (.TxR extension), Log files, and Auto Logon (ALF) files.

The defaults values are:

- Excel/Access files: C:\Documents and Settings\\MyDocuments\Winshuttle\Transaction\Data

- TRANSACTION scripts: C:\Documents and Settings\\MyDocuments\Winshuttle\Transaction\Scripts
- Log files: C:\Documents and Settings\\MyDocuments\Winshuttle\Transaction\Log
- Auto Logon (ALF) files: C:\Documents and Settings\\MyDocuments\Winshuttle\Transaction\ALF

## Microsoft Excel defaults

Enter the Excel worksheet default values:

- Sheet name: Default Excel sheet to use when running your script file (default is Sheet 1).
- Log column: Default Excel column where error messages are returned (default is E). See "Error Management" for more information.
- Use Excel Spreadsheet in the mapper preview: This setting allows you to toggle between the new Excel preview and the legacy preview. The Excel preview provides faster preview loading, a cleaner Excel-like interface, capability to map more than 256 columns, and a view of graphic and formatted data. This setting should be cleared only in consultation with Winshuttle support and ensure that you are not in the mapper when you enable the setting.

NOTE: Make sure the column you assign for the log results is empty. TRANSACTION overwrites any data in the specified Log column with the transaction log results.

## Data read preferences

Select how TRANSACTION interprets Excel data using either of the available formats: As stored (internally) or As displayed (the default). For example, the actual stored internal value in column C is 2.146, while this value is displayed as 2.15. If you select the **As stored setting**, it is read as 2.146. However, if you select the As displayed setting, it is read as 2.15. This is the value that is passed to SAP.

NOTE: You must make sure that the SAP field that is mapped to this column matches the format of the data in the Excel spreadsheet, as well as your Data read preferences. For example, when the SAP field is formatted to accept only two decimal places, and you are passing a number containing more than two decimal places, an error message is displayed.


## Usage defaults

Enter the Excel worksheet usage defaults:

- No. of rows to remain empty before TRANSACTION stops: Use this value to determine the number of consecutive empty rows allowed before the run stops automatically. For example, if the default is set to 5 (the default), TRANSACTION automatically stops running after it processes five consecutive empty rows.
- Trim data before uploading into your SAP System: Select this check box to remove the leading and trailing spaces from the data read before uploading into your SAP system.

## SAP defaults

Use **SAP Defaults** to set the defaults for interacting with SAP system.

- On the Main menu, click the **Tools tab**, click Options  and then click **SAP Defaults**.

**Logon defaults:** Logon defaults describe how SAP behaves when you logon without using auto logon.

**Default SAP Logon Language:** Type the default language that TRANSACTION uses when logging on to SAP. The default is EN for English.

**Remember last logon password:** Select this check box to save the last logon values that were entered and to display those values the next time you log on to SAP during a single TRANSACTION session.

**Sustained logon timeout (mins):** With a sustained logon, you are not required to logon between different runs of a specified file for the time period that is defined by the timeout setting. The possible timeout range is between 1 minute and 120 minutes.

**Translate messages while running in debug mode (longer run time during debug):** Select this check box to display descriptive messages while you run your transaction script in Step-by-Step (troubleshooting) mode. Because SAP does not provide descriptive messages while running in Step-by-Step mode, TRANSACTION extracts this information, which requires more processing time.

Clear the setting to view only the short messages, or when you are running a large number of rows of data in troubleshooting mode and require faster processing.

### **Execute GUI Scripting Transactions in SAP GUI session launched through Virsa**

**FireFighter:** Select this check box to enable the execution of GUI scripting transactions in an SAP GUI session that was launched through Virsa FireFighter. Clear this option if you don't want to attach to the already open SAP GUI sessions that were launched through Virsa FireFighter. See "Using TRANSACTION with Virsa FireFighter" in the TRANSACTION Advanced Guide for more information.

**SAP logon .ini folder:** Specify a shared location or a location other than C:\Windows for the .ini file. You can specify only the folder name, not the file name.


**Single Sign On (64-bit) configuration file:** Click the Search button to browse to your config file.

**Default mode for recording SAP transaction:** Select the default mode of recording SAP TRANSACTION.



## Error management and trace files

Use Error Management to determine how TRANSACTION processes errors that occur when a script file is run.

- On the Main menu, click **Tools**, and then click **Options** . In the list, click **Error Management**.

When an attempted upload produces errors, the rows with errors are not uploaded to SAP and the related error messages from SAP are returned to the log field or column that was specified.

### Default indicators for errors in messages

You can use these settings to set a background cell color or prefixed text to flag Excel log column cells that contain error messages. After you correct the data, you can go to the Run screen and select the Run on Errors option to rerun the script file for only those rows marked as erred. When the erred rows run successfully, they are marked as fixed.

**Background color of cell for errors:** Click this setting to select a background color for the errors listed in the log column specified in the Log Result Column field on the Run screen. The default color used is red. If you select this setting, the Prefix text for log column is left blank.

**Background color of cell for fixed errors:** Click this setting to select a background color for the errors that are fixed. The default color used for fixed errors is green.

**Use Prefix text for errors in the Log column:** Click this setting to mark errors by adding a prefix to the error messages stored in the Excel, Access, and Google Log columns or fields. The default prefix text is "Err:". If you select this setting, no background color is used in the cells.

**Use Prefix text for fixed errors in the Log column:** Click this setting to mark errors by adding a prefix to the fixed error messages stored in the Excel, Access, and Google Log column or fields. The default prefix text is "Fix:". If you select this setting, no background color is used in the cells.

**Warning Error Text:** Check this option and provide the warning text to appear in a separate column in case SAP returns an error in the log column.

### Trace

The trace tool is used by support to help to find the root cause of problems you might experience with the application. When this option is enabled, it traces all the actions you have taken in the software and writes them to a file at the log folder location set in the application options with a default value: C:\Documents and Settings\\My Documents\Winshuttle\TRANSACTION\Log.

If you experience a software problem, check the trace option and repeat the actions that resulted in an error. A text file is created in the default log folder that you can send to the support team for review.

When you are finished, clear the option. It is important to turn off Trace after you have finished tracing the problem so that you do not fill your file with information that is irrelevant to the next trace.

## Client Connection Settings

Depending on your firewall and email policies, your organization may select to route email through the internal email server or the Winshuttle default email server.

- Attachments can be added to email messages sent through an internal server.
- No attachments can be attached to messages that are routed through the Winshuttle default email server.

### To set up email connections

1. On the Tools tab, click Options.
2. Click Client Connection Settings.
3. Select Winshuttle default mail or Internal e-mail server.
4. For an internal account, select settings for the account that should receive the results:
  - Microsoft Exchange Server: Provide the name of the Microsoft Exchange Server.
  - Server port number: By default, the server port number is 25. Contact to your IT department to confirm the port number for your organization.
  - Attachment size limit (MB): Specify the maximum attachment size. The attachment size should not exceed the size allowed on the exchange server.
  - From Email address: Specify the Exchange server e-mail address to which the message should be sent.

Note: Application sends the e-mail message with attachment (with the configuration specified) only when the user is logged on to the system, with his or her domain authentication.

### Proxy settings for Connect

Proxy settings are used for Connect and are similar to Internet settings. Consult your system administrator before you change any proxy settings.

### To configure a proxy setting

1. Click Proxy Server Settings.
2. Do one of the following:
  - Click Use system proxy. Click OK.

-or-

- Click Custom proxy.
  1. Select the Authentication Required check box.
  2. Type in the username and password. Click OK.

## Restore defaults

If changes to settings cause incorrect updates or corrupted files, it may be useful to restore TRANSACTION defaults. You can restore all TRANSACTION options to their default settings by selecting the Restore Default options check box. For the changes to take effect, you must restart TRANSACTION.

### To restore default options

1. Click the Tools tab, and then click Options.
2. Click Restore Default.
3. Select Restore Defaults options.
4. Restart TRANSACTION.

## Enable GUI scripting

### Enabling Scripting on the Server Side

SAP GUI scripting is turned off by default on SAP systems. To enable scripting, an SAP administrator has to turn on scripting support on the Web Application Server. There are two methods.

#### Method 1

##### To turn on scripting support statically

In the first method, the administrator turns on support by setting the profile parameter `sapgui/user_scripting` to `TRUE` on the system to be tested and then restarts the application server. With this method, the administrator can determine which systems and groups of users have scripting support.

#### Method 2

In the second method, a user with the necessary permissions can turn on scripting support dynamically.

### To turn on scripting support dynamically

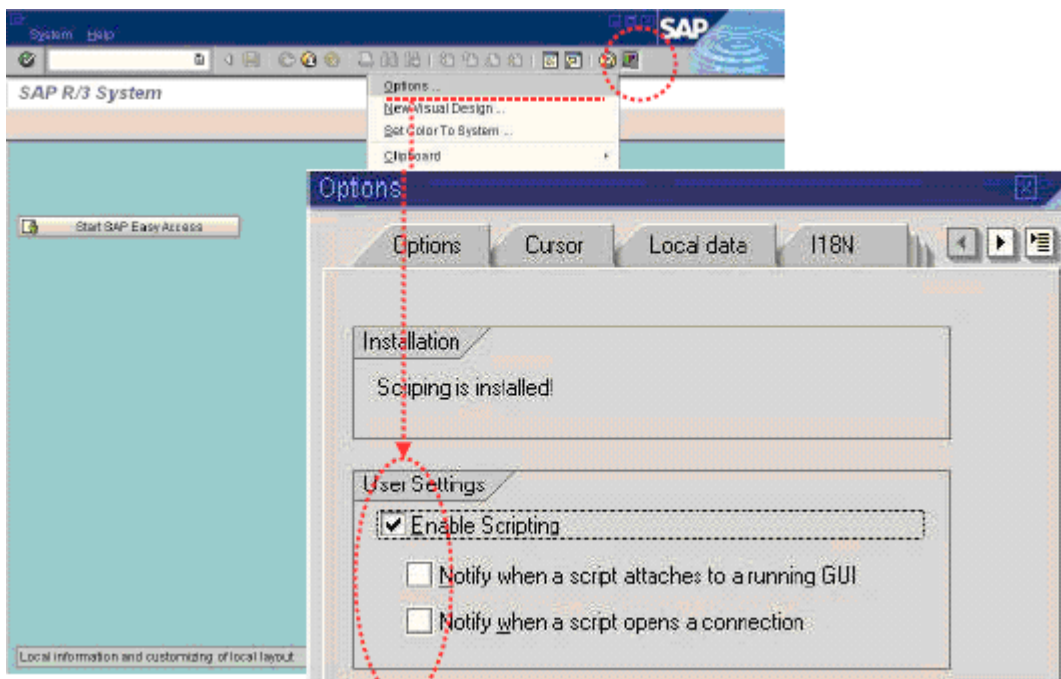
1. Log on to the target SAP system and run transaction RZ11.
2. Enter the parameter name `sapgui/user_scripting`.
3. Click Display, and then click Display Profile Parameter.
4. In the Attributes dialog box, click Change value.
5. In the New value field, select TRUE.
6. Click Save.

## Enabling Scripting in your SAP client GUI

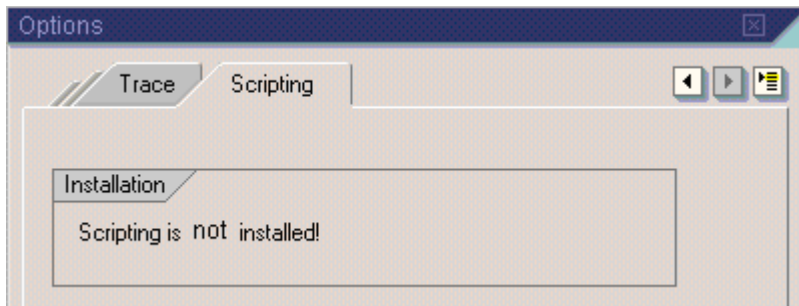
Both scripting support and the SAP GUI Logon pad are installed by default when the Windows SAP GUI client is installed. If they have not been installed, you need to reinstall the SAP GUI.

### To enable scripting on the GUI

1. Select the Options item, and then select Enable Scripting.



2. If the scripting component is not installed,



select the GUI scripting component.



## SAP GUI 720

To enable GUI scripting for SAP GUI 720, a few parameters on the server side must be set. These parameters do not necessarily exist on all SAP systems by default. If these variables are found in transaction RZ11, they must be set as follows:

sapgui/user\_scripting\_set\_readonly = false

sapgui/user\_scripting\_disable\_recording = false

sapgui/user\_scripting\_force\_notification = false

NOTE:

[http://help.sap.com/saphelp\\_nw70ehp2/helpdata/EN/47/954da7bee8700fe1000000a42189d/content.htm](http://help.sap.com/saphelp_nw70ehp2/helpdata/EN/47/954da7bee8700fe1000000a42189d/content.htm)

sapgui/user_scripting_disable_recording	If this parameter is set to TRUE, script playback is possible, but recording is not permitted.
sapgui/user_scripting_force_notification	If this parameter is set to TRUE, a notification is always displayed at the front end, regardless of the client options.  Notification refers to the message box that appears before record and run for attaching to GUI scripting session.
sapgui/user_scripting_set_readonly	If this parameter is set to TRUE, scripts may only act on read-only user interface elements.

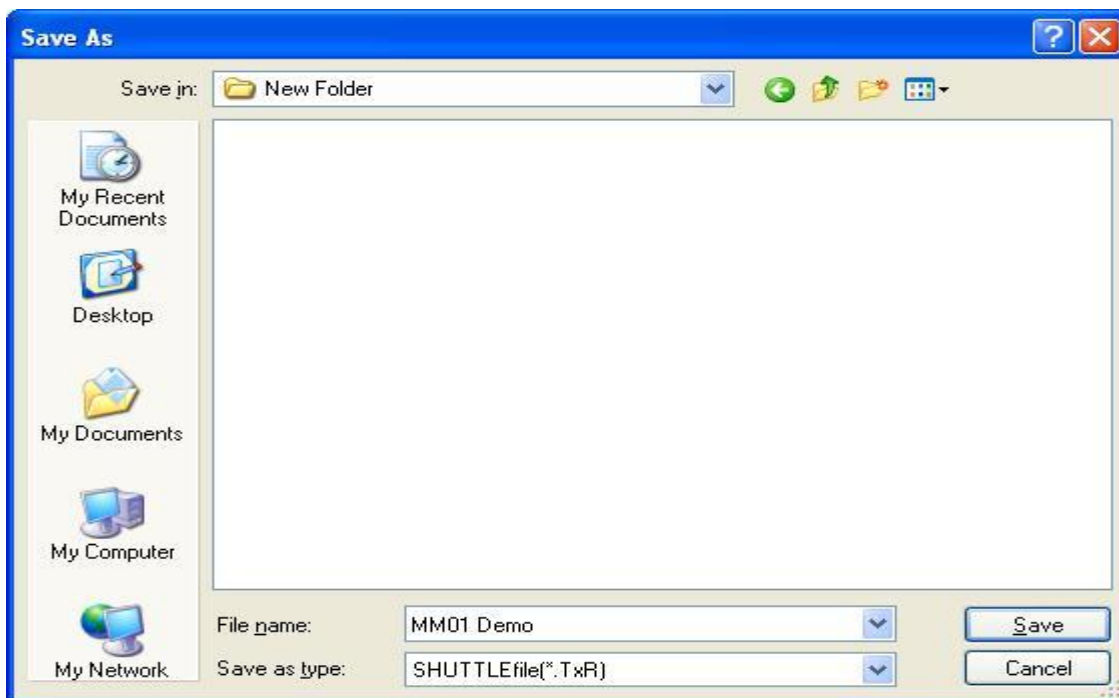
### To enable GUI scripting for SAP GUI 720

1. To open the SAP GUI options page, click on Control Panel, click SAP Front End, and then click SAP GUI Configuration.
2. Click Accessibility and Scripting.
3. Under User Settings, click Enable scripting.
4. Optional: You can select to be notified when a script is being attached to the SAP GUI as you record or run a TRANSACTION script.
5. Optional: You can select to be notified when a script opens an SAP connection.


### Unicode implementation for TRANSACTION scripts

TRANSACTION includes an adaptation for unicode so that scripts can be used with different language settings of Operating System/SAP System.

All TRANSACTION files recorded with 8.2 Update 1 and later versions of TRANSACTION are saved by default as TRANSACTION script (.TxR)



To ensure compatibility of Unicode Version TRANSACTION files with the earlier versions of TRANSACTION, new Save as Types have been introduced for TRANSACTION scripts in TRANSACTION:

If you try to open a file recorded using the earlier versions of TRANSACTION and click  (Save as), it will be automatically saved as type TRANSACTION script for that particular version. As in the example below, the TRANSACTION script was recorded with 8.2 version of TRANSACTION and hence is saved as type TRANSACTION script v 8.3 – v 8.2




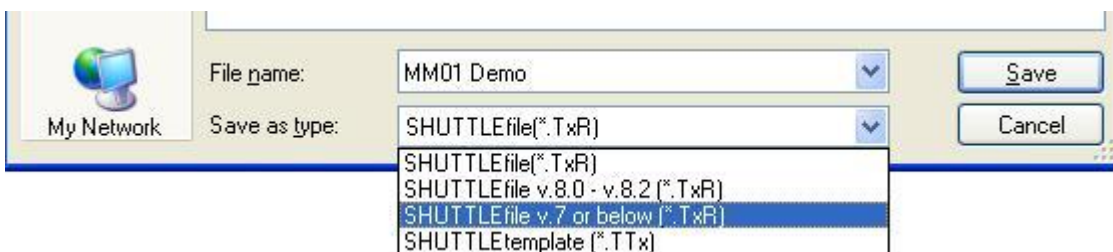
NOTE: This save type comes by default and the user doesn't need to specify the save type manually for this case.

IMPORTANT: Unicode guarantees that the TRANSACTION scripts will not be impacted by change in language setting. However, already corrupted files due to Language settings change can't be repaired by the Unicode Implementation.

If you want to use you current TRANSACTION script on an earlier version of TRANSACTION or want to distribute it to a user with an earlier version of TRANSACTION or RUNNER, choose to save it as that type of TRANSACTION script.

Example: To distribute a TRANSACTION script created with version 8.2 Update 1 and later to a user who is on version 7

1. Click  Save as.
2. Select SHUTTLEfile v.7 or below (\*.TxR) from the list.




The Transaction script is ready for use on version 7 or earlier of Transaction and Runner.

## Production Server Warning (Prod. Svr. Warning)

Use Prod. Svr. Warning to maintain a list of SAP Production servers:



- On the Main menu, click Tools, and then click Options button  to display the Options dialog box. Click Prod. Svr. Warning to display the settings shown below.

The Production Server Warning settings contain:

### Production Server Warning

- Production Server List: Use these setting to maintain a list of the SAP Production servers:
  - Click Add to identify a new Production server to the list and select the desired server. TRANSACTION can generate warnings for any of the servers listed.
  - Click Delete to remove a Production server from the list.
- Enable Production Server: Mark this check box to generate a warning every time you log on to a Production server. Clear this check box if you don't want to receive any warnings. If the Production Server warning flag is on, but you have not yet added any servers to the list, a warning is displayed each time you Record or Run a transaction. You can stop this warning by either turning the flag off or adding a server to your list. In this case, a warning is shown each time you log on to one of the systems included on the server list.

## Winshuttle Function Module for TRANSACTION

In some Winshuttle TRANSACTION environments, after the SAP BASIS 7.00 Support Package 24 was applied, certain transaction recording modes, particularly non-batch input recordings, no longer worked.

The primary cause of the problem relates to Support Package 24. However, related packages in other SAP BASIS versions also result in this problem.

### What caused this problem?

With the application of this support package changes were made by SAP developers to the function module CAT\_TCD\_CAL.

These changes were part of a security initiative undertaken by SAP to supposedly fix a legacy security issue in the SAP BASIS component. One of the changes was the introduction of a new authorization check.

This authorization check forces failures in the remote calling of transactions over RFC unless a given user has an S\_DEVELOP role assigned.

The assignment of S\_DEVELOP to regular business users is not an appropriate role assignment and in many instances this is not permitted for users in production environments.

Although the name of the authorization object implies a relationship to development tasks, it is also used to grant execution rights for specific development objects and SAP uses this in combination with object 'ECSC' (eCATT script) and activity '16' (execute) for testing automation.



Another change that was introduced was the check on customizing flag T000-CCIMAILDIS (Flag: eCATT allowed in client?).

Even if a user had authorization to run eCATT scripts - execution would fail if the flag was disabled.

### **What is the significance of eCATT in this context?**

The ability to run eCATT scripts is usually disabled on production systems and rightly so as testing of your solutions should not be done with automation tools in productive environments.

eCATT type functionality is used when Winshuttle calls SAP transactions remotely over RFC. The parallels are very limited however and are restricted to Winshuttle Transaction SAP transaction recordings. Winshuttle scripts, unlike eCATT scripts, never actually reside on the SAP system at all.

Winshuttle queries and Winshuttle Transaction scripts based on BAPIs and rFMs do not use eCATT type functionality and access technical objects in SAP directly.

### **How is Winshuttle Addressing this problem?**

To facilitate over 1200 Winshuttle customer installations being able to continue running existing Winshuttle scripts against their productive SAP systems a Winshuttle Z name-space Function Module effectively replicates the capabilities of the legacy SAP function module components that were changed by SAP.

This approach does not detract from the customer's system integrity or security model and nor does it represent any added vulnerability to a given SAP system beyond the state prior to the application of the SAP support package.

For more information, contact Winshuttle Support for the Winshuttle Function Module for TRANSACTION document.



# Preparing to record and recording SAP transactions

## Using SSO and EP logon

TRANSACTION If an SSO-enabled system is used for logon to SAP, the windows identity of the user is used for logging on to SAP. If you have installed 64-bit version of TRANSACTION, you might need to do additional settings for SSO to work. 64-bit SSO libraries for Windows should be available on the computer. Otherwise, the libraries can be downloaded from Microsoft.com.

## Locating 64-bit SSO libraries on the computer

32-bit SSO libraries are placed in folder %windir%\system32.

64-bit SSO libraries are placed in folder %windir%\syswow64.

The following libraries ensure that SSO works with 64-bit applications when TRANSACTION is in GUI scripting mode. If the libraries are not available on the computer, they can be downloaded from Microsoft.com

SSO Library on Windows 64-bit with SapGui 32-bit

SSO Library on Windows 64-bit with Winshuttle Transaction 64-bit

C:\windows\system32\gsskrb5.dll

gx64krb5.dll (Win64 bit- on AMD Athlon64/Opteron)

gi64krb5.dll (Win64 bit on Itanic)

C:\windows\system32\gssntlm.dll

gx64ntlm.dll (Win64 bit- on AMD Athlon64/Opteron)

gi64ntlm.dll (Win64 bit on Itanic)

## How to configure Winshuttle 64-bit product to work with SSO

1. Click Options, and then click SAP Defaults.
2. In the Single Sign On (64-bit) configuration box, click the Search button to browse to your config file. Click OK.
3. Make environment variable in "My Computer > Properties > Environment variables > SNC\_LIB". It should contain the path of 32 bit dll present in "C:\Windows\System32\<Name of the dll>" and save it.
4. Restart you system.

For more details on this, contact Winshuttle support for SSO configuration document.

## EP Logon

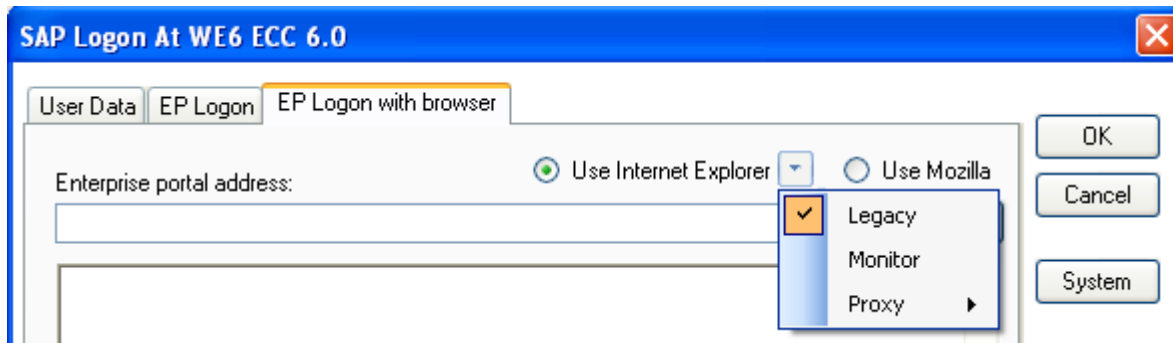
Enterprise portal (EP) logons can present ticket fetching problems. To solve these problems, Winshuttle offers the following four methods:

1. EP Logon using Microsoft Internet Explorer
2. EP Logon using Microsoft Internet Explorer with Monitor
3. EP Logon using Microsoft Internet Explorer with Proxy
4. EP Logon using the Mozilla package

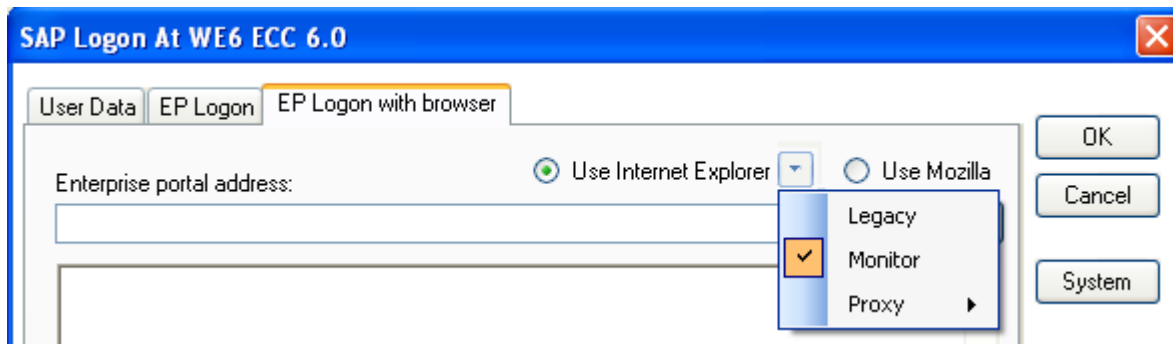
### Using EP logon

A chosen method should be used only if the previous method does not work. For example, Method 4 should be used only if methods 1, 2, and 3 were attempted in that order and did not work.

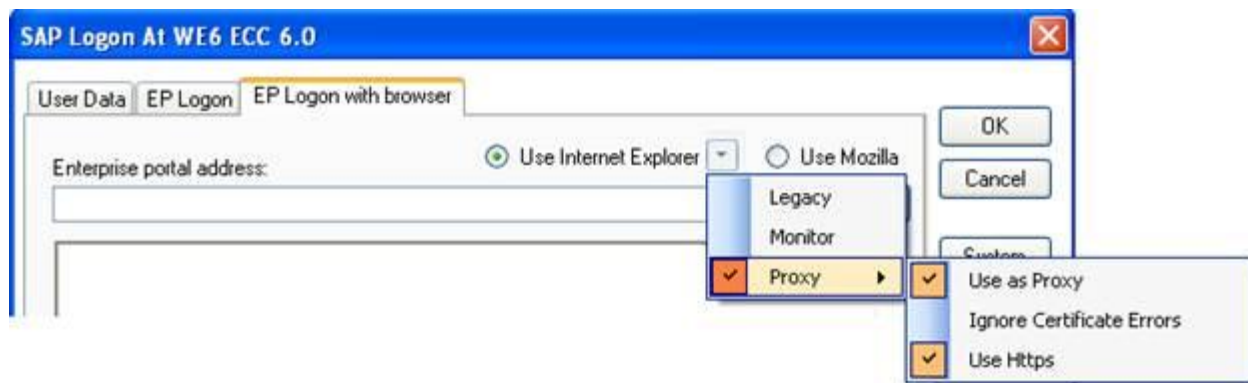
Method 1: Use the Internet Explorer method (Legacy) to logon to the SAP portal.



Method 2: Use the Monitor option with Internet Explorer to get the ticket from web browser events.



Method 3: The Proxy method still depends on reading web events, but with different settings. Depending on your setup, you may need to choose the different sub options available for this method.



Method 4: When you use the Mozilla option for the first time, you will be asked to download a configuration package and set the location of this package. On subsequent logons, the application automatically selects the package from this location and logs on to the appropriate EP Portal.

For more details, contact Winshuttle support.

## Production versus non-production servers

When creating a script, it is good practice to develop and test the script on a non-production server. Only after the script has been tested and approved should it be run on a production server.

## Recording modes

TRANSACTION automatically selects the recording mode most appropriate to your transaction, which is usually non-batch without SAP controls. We recommend that you run the transaction in the default mode, unless you are unable to accomplish the results you want because it does not present the SAP screens and fields you need.

For most t-codes, non-batch without SAP controls is appropriate the recording mode and it is the TRANSACTION default recording mode. However, for the following t-codes, other recording modes are more effective.

T-code	Recording mode
CSOI	Non-batch with SAP controls
CV01N	GUI scripting
ME22N	Batch
ME51N	Batch

ME52N	Batch
CA01	Batch
MIGO	Batch
PA40	Non-batch without controls
PA30	Non-batch without controls
SU01	Non-batch without controls
SU10	Non-batch without controls
PFCG	Non-batch without controls

### Batch Input Mode

Many of the transactions in SAP are optimized to run in Batch Input mode. This is primarily for performance reasons. Batch Input mode is more efficient for data processing. Some SAP transactions have screens and fields that are designed especially for batch input processing to facilitate automation.

Here are some of the differences you might notice in Batch Input mode:

- SAP screens might look slightly different from the ones in Non-Batch Input mode.
- Some of the fields, or the placement of the fields, might be different as compared to the screens in Non-Batch Input mode.
- Warning and error messages might be different from the ones you see in Non-Batch Input mode.
- We recommend that you run the transaction in Batch Input Mode, unless you are unable to accomplish the desired results due to differences in the screens or unless you want to enable Read from SAP or Extended logs.

### Non-Batch Input Mode

Non-Batch Input mode is recommended when recording SAP transactions either With SAP Controls such as ALV Grids. This mode is also used for features Without SAP Controls such as for downloading data or when using Extended Log.

NOTE: The setting you choose for recording your transaction is automatically saved and chosen during runtime.

Key considerations for using Non-Batch Input mode:

- You may need special authorization to run in Non-Batch Input mode.
- When you run a transaction in Non-Batch Input mode, it remembers your user settings. If these settings change between the time you recorded the script file and the time you run it, the script file might not work or may need some adjustment.
- It is possible to use long text recording with Non-Batch Input mode.

## GUI Scripting

The SAP GUI Scripting option allows you to record files that emulate the business user interaction with the SAP GUI. In simple terms, the feature works as a screen reader. Use GUI Scripting when Batch Input mode and Non-Batch Input mode do not record and upload the data correctly into SAP.

## Properties

**Title:** Type an appropriate title for the script file you are about to create. This title is shown on the home screen when you open this script file. It is displayed on the Selection bar on the Record, Map, and Run screens.

**Lock Script file:** Select this option to lock the script file. When you select this check box, the Password and Confirm Password fields are displayed. You are required to enter the password you want to use to protect your script file. After you password-protect the script file, you must type the password each time you open the file in order to map or run the recording. Leaving this check box blank allows the user to edit the script file mapping and run the script file without requiring a password.

- **SAP System:** Displays the name of the SAP system you selected on the SAP SHUTTLE Logon dialog box.
- **Created By:** Displays the name of the person who created this recording. This is picked up from the SAP Logon dialog box.
- **Date/Time:** Displays the date and time when this recording was created.
- **Purpose of creation:** Type an explanation that describes why you are creating this recording.
- **Comments:** Type any comments that provide additional information about this recording.
- **Retrieve field descriptions for read only or disabled fields:** A user can choose not to retrieve field descriptions for read-only or disabled fields if they are not required for a TRANSACTION script. Note that on a network bandwidth constrained environment, it is advisable to clear this setting to achieve better performance.

For information on starting a recording, see the *TRANSACTION Getting Started Guide*.

## Using templates in TRANSACTION

TRANSACTION provides Excel data sheet and TRANSACTION script templates for some of the key transactions, and for more complex processes, such as direct input into SAP through routings and bills of material. For each template, online collateral, such as videos and step-by-step guides, is available. Click the Collateral Link [Collateral Link](#) to view the documents.

NOTE: Templates help you to make the optimum choices for your recordings, but you should consider customizing them for your own use.

### To use a template

1. In the TRANSACTION home screen, click New.
2. On the Record pane, click Create recording from template.



Record

Enter the SAP Transaction code to begin a new recording.

Transaction Code:

Go

[▶ Create recording from template](#)

3. In the Templates window, browse through the available categories, and select either a .xlsx or a .txr template.

For .txr files, click Create. The .txr opens.

-or-

For .xlsx files, click Create. You are prompted to save the a copy to your local drive. You can then open the local file, or open the embedded script with its Excel file.

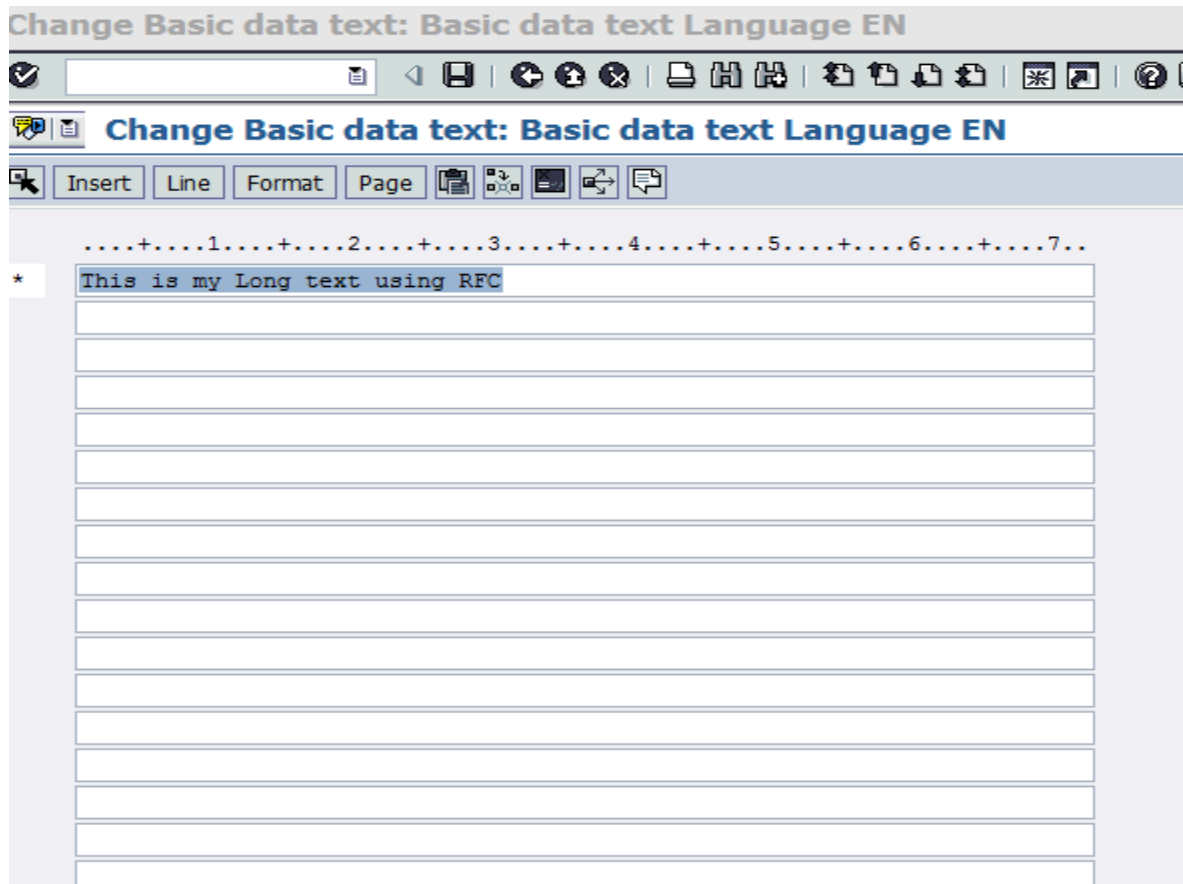
4. The Run screen appears. To update mapping for the .xlsx, click Update Mapping.

Save the file after you make your changes. It will automatically be saved as a TRANSACTION script file.



## Recording long text

**Batch Input Mode** To record long text in batch input mode, you should use a different sort of editor for Long Text. The screen is divided into several lines and each line has a counter and a limit on the number of characters allowed (usually 72 characters). In this mode, each line is already assigned a field name and that is what TRANSACTION captures and stores in the TRANSACTION script.



**Non-Batch Input Mode** To record long text editor in non-batch mode, the editor looks like Notepad. With the long text editor, SAP takes the text and, through internal processing, converts and assigns it to an SAP field. This is recorded as an additional field in the mapper.

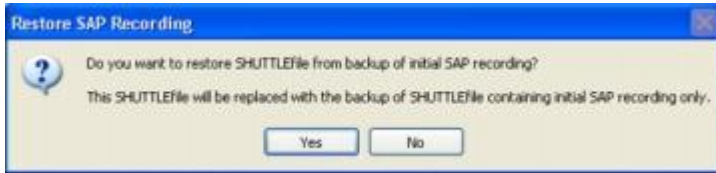
## Backing up data and files in Winshuttle

### Backing up the recording

.BAK, or backup files, are created when you save a new recording. Changes made in the mapper are not saved to the .BAK file.

## To restore original recording of SAP fields

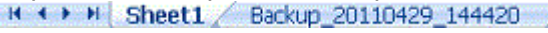
1. Click the Tools tab, and then click Restore SAP Recording  .



2. Click Yes to restore the backup of the TRANSACTION script containing the initial SAP recording only. This selection is enabled only if the original backup is present in the same folder. (The original backup is created when the TRANSACTION script is initially created.)

You can also click No to return to the Mapper screen.

## Backing up SAP data

Sometimes you may want to restore SAP data to a version previous to an update. To ensure that you have a backup copy of SAP, use the Backup SAP Data option. This option adds a new tab to your Excel workbook or Google spreadsheet  that includes the data as it was in SAP before the upload.

## To backup SAP data

Your script must be recorded in non-batch mode.

1. On the Run pane, in the Advanced Options slide-out select Backup SAP Data.
2. Click Test or Run as appropriate.

After the run, the tab appears in the Excel file. To restore data, specify the name of the tab in the Sheet name box of the Run pane, and run again.

## Preserving log files before uninstalling TRANSACTION

If your log files are not saved to a server, such as in a shared folder or through Winshuttle CENTRAL, when you uninstall TRANSACTION from a computer, the log files are not saved.

## Advanced mapping

### Editing in the Mapper

#### Data sources

In addition to data sources in Excel worksheets, Transaction also offers the capability to upload from Access tables, and Google spreadsheets. In TRANSACTION, you can upload data, but you can also download.

#### Mapping




For Excel, Access, and Google spreadsheets you can drag from the source preview to the SAP preview.

**Mapping the log column:** You can specify the column, cell, or field of your output where you want the log column to appear. Drag the log column field into the preview area to set it.


**Change Source Type:** Sets the data source for the script. The Mapper initially displays the data using Excel as the data source. Change the source type to Access or Google Spreadsheets.

**Mapping:** Indicates how individual fields are mapped. Individual fields might be uploaded to SAP, downloaded from SAP, or fixed in value throughout the run.

   Uploads/downloads data using Excel

   Uploads/downloads data using Access

   Uploads/downloads data using Google

 Fixed Value Upload. These are constant values, where the same value is used in every uploaded record. If the value is the same as entered into SAP while recording the Transaction, then no changes are required. For example, Company code might be a good candidate for a fixed value, whereas the Material Number or Employee ID is likely to change for every record uploaded.

**Value:** The data stored in this field is used while running the transaction. If data is coming from Excel, Access, or Google, specify the appropriate column that stores the data for the concerned field. Ensure that the correct value is entered in this field.

**Notes:** Use the Notes column to enter useful information for each row in the TRANSACTION script as needed.

**Cut, Copy, and Paste Rows:** Cut or Copy and Paste the row or range of rows selected. You can click anywhere in the row, on any cell, to see the shortcut menu. You can also copy rows that are part of a loop.

**Enable Rows:** Allow the selected row or range of rows to be mapped to the Data Source Preview.

**Disable Rows:** Disable the selected row or range of rows from mapping to the Data Source Preview.

### Recording Preview Fields


**Sno.:** Displays the Mapper row ID number.

**Field Description:** Displays the descriptive name of the SAP field. The description is always displayed in the language that you recorded your script in. You can change the descriptive name if needed. This field is for informational purposes only and is not sent to SAP.

**Field Name:** Displays the technical name of the SAP field for the recorded transaction. This field name is sent to SAP and must exactly match the SAP field name. You should not change the name of this field, unless you are certain that the changed name exists in the transaction.

For Access: Add as many columns as you need. Must add them manually. Add tables as required.

## Cell-based mapping and column-based mapping

With cell-based mapping, you can make absolute cell references when mapping. You can select this from the Column Mapping/Cell Mapping button . The default is column-based mapping.


When you use cell-based mapping, you can also display an exact preview of an Excel worksheet. With the Excel preview, the mapper is not limited to eight rows.

NOTE: TRANSACTION displays the results in the Excel worksheet you specify in the Application Defaults settings including the Log column. When converting recordings from one mode to the other, you must explicitly remap to the cell-based format. See also Auto Mapping.

## Filtering mapper setting

With filtering in the mapper, you can simplify the display of rows to make mapping, creating loops easier, and other mapper editing easier. Available are only those fields that were added during the recording.

## To filter rows in the mapper

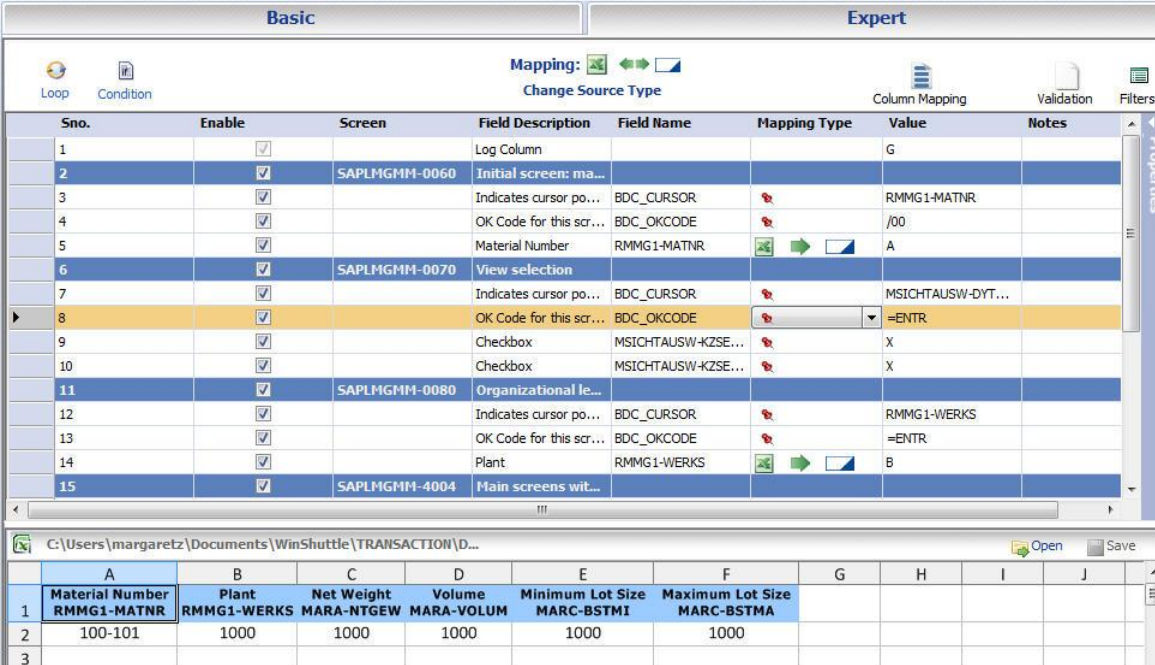
On the far right of the Expert tab, click the Filter button . Select from the following settings:

- Hide read from SAP fields: Select this setting to display or hide all rows identified as Read From SAP.
- Hide all disabled fields: (the Default) Select this setting to display or hide all disabled rows. Displaying the disabled rows allows those rows to be enabled.
- Hide mapped fields: Select this setting to display or hide all rows that have been mapped.
- Hide absolute value fields: Select this setting to display or hide all rows identified as (Fix Value Upload).
- Hide system fields (BDC fields): Select this setting to display or hide all SAP system fields (BDC\_OKCODE & BDC\_CURSOR rows).

## Advanced looping capability

### Handling multiline transactions

Many SAP transactions require multiple rows of data to be processed at the same time. Examples of these transactions include FB50, F-65, ME21, VA01, CS01, CA01, FB70, and others. For many of these transactions, SAP separates the data into a header section and one or more line item or detail sections. The header section usually contains data that remains constant for the entire transaction such as a date. Each line item section typically contains reiterative data, that is entered in a grid-like area of the SAP transaction such as journal entry line items.



The screenshot displays the SAP Transaction Mapper interface. The top bar has 'Basic' and 'Expert' tabs. The 'Expert' tab is active, showing a 'Mapping' section with icons for 'Change Source Type', 'Column Mapping', 'Validation', and 'Filters'. Below this is a table with columns: Sno., Enable, Screen, Field Description, Field Name, Mapping Type, Value, and Notes. Row 8 is highlighted in yellow. Below the mapping table is a data grid with columns A through J. The data grid shows a header row with 'Material Number', 'Plant', 'Net Weight', 'Volume', 'Minimum Lot Size', and 'Maximum Lot Size', followed by three data rows.

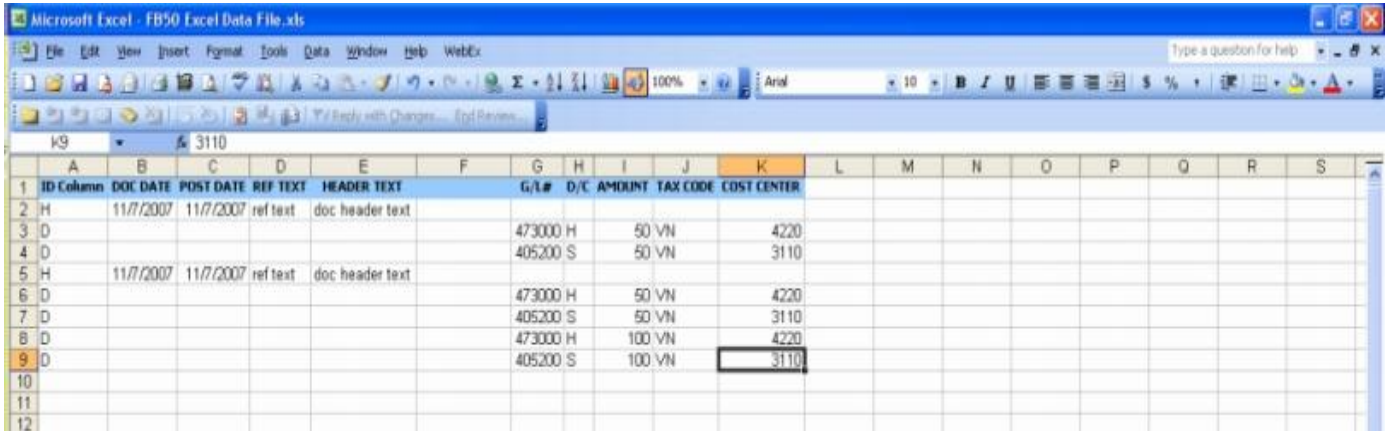
Sno.	Enable	Screen	Field Description	Field Name	Mapping Type	Value	Notes
1	<input checked="" type="checkbox"/>		Log Column			G	
2	<input checked="" type="checkbox"/>	SAPLMGMM-0060	Initial screen: ma...				
3	<input checked="" type="checkbox"/>		Indicates cursor po...	BDC_CURSOR		RMMG1-MATNR	
4	<input checked="" type="checkbox"/>		OK Code for this scr...	BDC_OKCODE		/00	
5	<input checked="" type="checkbox"/>		Material Number	RMMG1-MATNR		A	
6	<input checked="" type="checkbox"/>	SAPLMGMM-0070	View selection				
7	<input checked="" type="checkbox"/>		Indicates cursor po...	BDC_CURSOR		MSICHTAUSW-DYT...	
8	<input checked="" type="checkbox"/>		OK Code for this scr...	BDC_OKCODE		=ENTR	
9	<input checked="" type="checkbox"/>		Checkbox	MSICHTAUSW-KZSE...		X	
10	<input checked="" type="checkbox"/>		Checkbox	MSICHTAUSW-KZSE...		X	
11	<input checked="" type="checkbox"/>	SAPLMGMM-0080	Organizational le...				
12	<input checked="" type="checkbox"/>		Indicates cursor po...	BDC_CURSOR		RMMG1-WERKS	
13	<input checked="" type="checkbox"/>		OK Code for this scr...	BDC_OKCODE		=ENTR	
14	<input checked="" type="checkbox"/>		Plant	RMMG1-WERKS		B	
15	<input checked="" type="checkbox"/>	SAPLMGMM-4004	Main screens wit...				

	A	B	C	D	E	F	G	H	I	J
1	Material Number RMMG1-MATNR	Plant RMMG1-WERKS	Net Weight MARA-NTGEW	Volume MARA-VOLUM	Minimum Lot Size MARC-BSTMI	Maximum Lot Size MARC-BSTMA				
2	100-101	1000	1000	1000	1000	1000				
3										

## Do...While Loop

TRANSACTION handles complex multi-line transactions using the Do...While Loop feature. With Do...While loops you can map multiple rows to one transaction by defining the records in the Excel worksheet or Access database as Header section or Line Item sections.



In this example, the data for one record is divided among several rows. For multiline transactions, by default the first ID column of Excel is reserved for the Excel Column ID. The first ID Column stores the row type, which defaults to a value of either H (Header) or D (Detail/Line Item). When you run the transaction, TRANSACTION uploads all data from row type H to the Header section of the SAP transaction, and all data from row type D to the line items specified in the Mapper.

NOTE: We strongly recommend that you observe the following conditions when you use TRANSACTION to upload data using multiple loops.

1. If you are using the same identifier row for different loops, ensure that you use different identifier values. You can use the same identifier even if the loops refer to different identifier columns.
2. Regardless of whether you use the same or different identifier rows for different loops, if the loops are nested, the data for loops should be in the same order as the loops appear in the mapper. In addition, one row must contain data pertaining only to a single line item record set. If there are multiple loops at a single level, no indentation is required in the Excel file.

Sales Organization	Distribution Channel	Division	Sales Document Type	Partner Function	Customer Number 1	Line ID	Sales Order Line #	Material	QTY	Line ID2	
SALES_ORG	DISTR_CHAN	DIVISION	DOC_TYPE	PARTN_ROL	PARTN_NUMB						
1000	10	00	TA	E		H				H	Log TRANSACTION ID: 0 YES-000, mltst Mode: BAPI Template 8/24/2010 10:41:55 AM BAPI_SALESORDER_CREATEFROMDAT2- Training2.TstR
							10	PA-1000	5	D	SALES_HEADER_IN has been processed
							20	PA-1001	5	D	Item row D: See log below
							30	PA-1002	5	D	Item row D: See log below
				AG	1030	D1					Item row D: See log below
				WE	1030	D1					SALES_HEADER_IN has been processed

While recording the transaction, we recommend that you enter more than one line item in the transaction to create a visible pattern of repetition in the Mapper. Notice that the example shows more than one set of journal entry line items.

## Mapping nested loops

When you map the details column of nested loop, the details automatically appear in the second and third rows of the Preview. This functionality also applies when you use the Automap or Read from SAP features.

## Preparing your workbook for indented data

When you upload nested loops, the data in your Excel workbook must be laid out in a certain format. The header and its appropriate line items can be in the same column. Each loop can have a header.

### Loop data for column-based mapping

H - Header row

D - multiple line items for loop 1

D

D

D1- multiple line items for loop 2

D1

D1

D2- multiple line items for loop 3

D2

D2

A	B	C
H	H	H
D	D1	D2
D	D1	D2
D	D1	D2
D		D2
		D2
		D2
		D2
		D2

*Simple parallel loops*

<input checked="" type="checkbox"/>		<b>G,H,D</b>	
<input checked="" type="checkbox"/>			Material Description (Short Text)
<input checked="" type="checkbox"/>			
<input checked="" type="checkbox"/>			Indicates cursor position on the screen
<input checked="" type="checkbox"/>		<b>F,H,D1</b>	
<input checked="" type="checkbox"/>			Gross Weight
<input checked="" type="checkbox"/>			Weight Unit
<input checked="" type="checkbox"/>			Net Weight
<input checked="" type="checkbox"/>			

The Mapper layout

	A	B	C	D	E	F	G	
	Material Number RMMGI-MATNR	Material Description {Short Text} MAKT-MAKTX	Gross Weight MARA-BRGEV	Weight Unit MARA-GEVEI	Net Weight MARA-NTGEV	ID Column	ID Column	
1								
2	43					H	H	
3		Check Data111	150	kg	121	D1	D	
4		Check Data112	151	kg	122	D1	D	
5		Check Data113					D	
6		Check Data114	New format					D
7		Check Data115					D	

The corresponding layout for the Excel workbook

**Two parallel loops with one nested loop**

<input checked="" type="checkbox"/>		<b>I,H,D</b>	
<input checked="" type="checkbox"/>			Material Description
<input checked="" type="checkbox"/>			Base Unit of Measure
<input checked="" type="checkbox"/>			Material Group
<input checked="" type="checkbox"/>			Indicates cursor position on the screen
<input checked="" type="checkbox"/>		<b>J,H,D3</b>	
<input checked="" type="checkbox"/>			Gross weight
<input checked="" type="checkbox"/>			
<input checked="" type="checkbox"/>		<b>K,H,D2</b>	
<input checked="" type="checkbox"/>			Weight Unit
<input checked="" type="checkbox"/>			Net weight

The Mapper layout

	A	B	C	D	E	F	G	H	I	J	K
	Material Number RPHIG1-MATNR	Material Description MAKT-MAKTX	Base Unit of Measure MARA-MEINS	Material Group MARA-MATKL	Gross weight MARA-BRGEV	Weight Unit MARA-GEVEI	Net weight MARA-NTGEV	ID Column	ID Column	ID Column	
1											
2	100-200							H		H	
3		Test1	PC	001		KG	110	D		D2	
4					121	KG	111		D3	D2	
5					122	KG	112		D3	D2	
6		Test4	PC	001				D			
7					124				D3		
8		New Format			125				D3		

The corresponding layout for the Excel workbook



## Easily check loops

In simpler scripts you can easily see where loops are located. In the Basic tab, select a row, and then click the Advanced Options bar.

The Loop Column checkbox is selected for the rows mapped within the loop that contain data from records identified as Line Items (details). This setting instructs TRANSACTION to pull data from the Line Item (detail) contained in the Do...While Loop.

NOTE: The Loop Column check box must be checked for at least one of the fields. Otherwise, TRANSACTION script generates a syntax error.

## Copying and cutting rows with loops and conditions

You can copy or cut rows that contain loops and conditions. You must select all parts of the loop or condition to edit it. You cannot paste a construct into a log column.

## Loop data on multiple sheets

To create a clearer presentation of data when multiple loops are present in a script, the loops can reside on multiple data sheets. Multiple loops are available for Excel and Access data.

Parent column: provides the join key for all sheets

Parent sheet: The parent Excel worksheet that contains the join key

Current sheet: the Excel sheet containing the line items included in the loop

Current column: the column of the data sheet that contains the join key

You can assign the join key to any column. Therefore, consider carefully which columns already contain data.

## Invalid mapping

- You cannot map a loop to a column that is not part of that loop.
- You cannot validate when multiple sheets are present. If conditions validated as sheets are processed.
- You can use only column based mapping.
- No extended logs. Regular logs written only to header sheet.

## To map multiple loops

1. In the Preview pane of the Mapper screen, add as many new sheets as you need.
2. Click the first sheet. In the Mapper, select the items for the first loop. Click the Loop button.

The Loop Master Parent sheet is selected by default.

3. Define all parallel loops before you define any nested loops.

4. Map the rest of the line items.
5. To change the Master parent loop, click Delete all to delete all loops.
6. When a TXR is published, the published sheet must contain the same sheet names that are in the script.
7. If the data source is changed, the preview will revert to Untitled. Either add the sheets manually or open a sheet that contains the appropriate names.

Ensure that you save the Preview before closing the application or Preview sheets could be deleted.

### **New sheets and single sheet mapping**

- To add new sheets, you must use spreadsheet gear for the Excel sheet preview. Only alphanumeric sheet names are applicable.
- To switch to single sheet map, click the Clear Loop Data button, or delete the loops and then save the script file.

### **Loops and changing from cell-based mapping to column-based mapping**

Loops are retrained mapping can be changed from cell-based to column-based mapping. When the mapping type changes, loops become ID based. For example, B,H,D1 becomes B,D1.

If range-based loops are changed to column-based mapping, and then back to cell-based, all loops will be ID based.

### **Using the Do...While Loop Feature with Access**

TRANSACTION handles complex multi-line transactions using the Do...While Loop feature. The basic principles of using the Do...While Loop with Access as the data source include:

- All Header data must reside in one (primary) table selected in Data Source Preview (for Excel or Access).
- All Line Item data related to this Header must reside in a separate table within the same database. This Line Item table is selected while inserting the Do...While Loop in the Mapper.
- The two tables (Header and Line Item) must have a common field for linking the Line Item data record with its associated Header data record. This common field establishes the relationship between the two tables for the Do...While Loop in the Mapper and the values from these fields must match.

An example of the Header table and the Line Item table is shown below.

The screenshot shows two tables in Microsoft Access. The first table, 'HEADER\_TABLE', has the following data:

ID	COMMON_FIELD	BUDAT	BLDAT	XBLNR	BKTXT	LOG_COLUMN
1	100	11/5/2007	11/5/2007	REF	DOC HDR	

The second table, 'DETAIL\_TABLE', has the following data:

ID	COMMON_FIELD	HKONT	SHWKZ	WRBTR	KOSTL	MWSKZ
1	100	473000	S	50	3110	VN
2	100	405200	H	50	4220	VN

Both tables contain a field called `COMMON_FIELD`, as shown below. `COMMON_FIELD` can contain any values you choose, as long as the values in the common field Header Table records associated to the Line Item Table (details) records match. `COMMON_FIELD` is just an integer that matches in both tables.


The 'Do While Properties' dialog box is shown with the following settings:

- Start row: 8
- End row: 16
- Header Table: HEADER\_TABLE
- Line Item Table: DETAIL\_TABLE
- Header Table Field: COMMON\_FIELD
- Line Item Table Field: COMMON\_FIELD

When the items table is identified and a join is made, the Line Item table is made available in the Mapper for mapping purposes. Then you can map the Line Items (details) by dragging the Access database field and dropping it on the SAP field.

While recording the transaction, it is recommended to enter more than one line item in the transaction in order to create a visible pattern of repetition in the Mapper. Notice that the example shows more than one set of journal entry line items.

### To use the Do While...Loop feature to handle multiline transactions

1. Create a Do...While Loop around one set of data only.
2. Disable the duplicate rows of data by selecting the Disable Row(s) flag in the left-most column of the Expert Tab: Recording Preview.
3. After you insert the Do...While loop, the Mapper displays the fields contained within the loop by using two  (Loop icons) connected by a line.

You can select a row in the Do...While Loop and click the horizontal Properties bar on the far left to display the Properties slide-in. On the Properties slide-in, the Loop Column check box is marked for the rows mapped within the loop that contain data from records identified as Line Items (details). This setting instructs TRANSACTION to pull data from the Line Item (detail) contained in the Do...While Loop.

NOTE: The Loop Column check box must be selected for at least one of the fields. Otherwise, TRANSACTION script generates a syntax error.

## Using Multiple and Nested Loops

When your task requires more than one Do...While Loop to fulfill a business process, the Mapper offers two options for using multiple loops to handle these scenarios:

NOTE: Multiple and nested Do...While Loop are not supported when Access is used as the data source.

**Multiple Do...While Loop:** Multiple loops are useful in transactions that have more than one set of multi-line items (grids). Each of these sets could potentially have either their own set of Header and Line Items, or they could share a common Header section, but have different Line Items, as shown below.

**Nested Do...While Loop:** Nested loops are useful in transactions that require data to be uploaded into multi-line items, which contain embedded line items for each item in the topmost level/outer grid. In this case, there is an outer Do...While Loop that contains a number of Do...While Loop nested within it. For example, if you are uploading data in grid A, which contains grids B and C, with each having data related to each line of A, then use a nested Do...While Loop to accomplish the upload. TRANSACTION supports nested Do...While Loops up to any level.

In this example, there are two outer level loops, namely +Do-While A, H, D and +Do-While A, H, D1. Apart from these, the first loop has a nested loop, +Do-While A, H, D2.

The format of the Excel worksheet used for uploading data is:

- Rows with ID H represent the Header for all of the Do...While Loops.
- Rows marked with ID D represent the Line Item data for the first outer Do...While Loop.
- Rows marked D1 and D2 are the Line Items for the second outer loop and its nested Do...While Loop, respectively.
- In the first H, D, D1, D2 block above, there are two Line items for grid D and two Line Items for grid D1.
- There is another grid D2 for each Line Item of grid D1. The first entry of D1 has four rows of data for D2, and second entry of D1 has two rows of data for D2.

## Limiting input fields

To help limit errors during data input, you can restrict values available for certain fields in the Excel data file. These values are statically stored with the TRANSACTION script and do not change when other values are changed in the Excel or Text file during Run.

1. In the SAP recording pane, select the field you on which you want to restrict values.
2. Click the Properties slide-in. Select Validate by list.



3. Click the values button to create the validation list. Create your value list by using one of the following methods:
  - Import values from an Excel file
  - Import values from a CSV Text file
  - Type in the values manually

When you validate or run this script, the selected field values are validated against the list. If there are any errors, the corresponding row or record will not reach SAP for posting or for validation and an warning message is sent.

In addition, this list will be available for input instead of the SAP values list when you use the look up function from in the RUNNER add-in for Excel.

## Mapping Google spreadsheets

You can use any TRANSACTION script with Google spreadsheets.

### To map a Google spreadsheet

1. In the Mapper, change the source type to Google Spreadsheets and map the fields to columns.
2. In Google Docs, an additional column, the Error Warnings column, is available to differentiate any Google warnings from other messages. Log and Error Warnings columns can be mapped only as downloads.
3. To save the sheet, click Save. If you are not already signed in, the Google sign in dialog box appears. Enter your Google Documents credentials. Select Stay signed in to signed in during a current session.

Each file saved to Google must have a unique name. You cannot save an existing file in Google with the same name. This means that if you update mapping, you must rename the file.

4. The Google Spreadsheets dialog box appears. Click owned by me to view all your folders. If you have just signed on, folders must download. Although Google spreadsheets appear to have 256 columns available for mapping, some columns might not be present in your spreadsheet. If you have mapped to a column that is unavailable, you are notified which columns and cells are unavailable. You can choose to continue mapping and add these columns later or you can map one of the available columns.
5. You can also open an existing Google spreadsheet and map it to the TRANSACTION script.

## Setting conditions

Conditional Run statements can provide additional control over the way data is uploaded into SAP. Conditional Run perform at a field level to set conditions for executing the selected portions of the TRANSACTION script. Conditional Run statements can be inserted around a single row or a block of rows. You may add multiple Conditional Run blocks, as long as they do not interfere with any other existing Conditional Run blocks.

Every cell belonging to the specified column is checked against the specified constant value. When the Conditional Run is performed, TRANSACTION executes only the row for which the given condition holds true.


For example, when column H is compared to the word "Credit", then TRANSACTION perform the actions contained within the IF condition only when the condition is true. In this case, a value of H is uploaded to the D/C Column of the FB50 transaction.

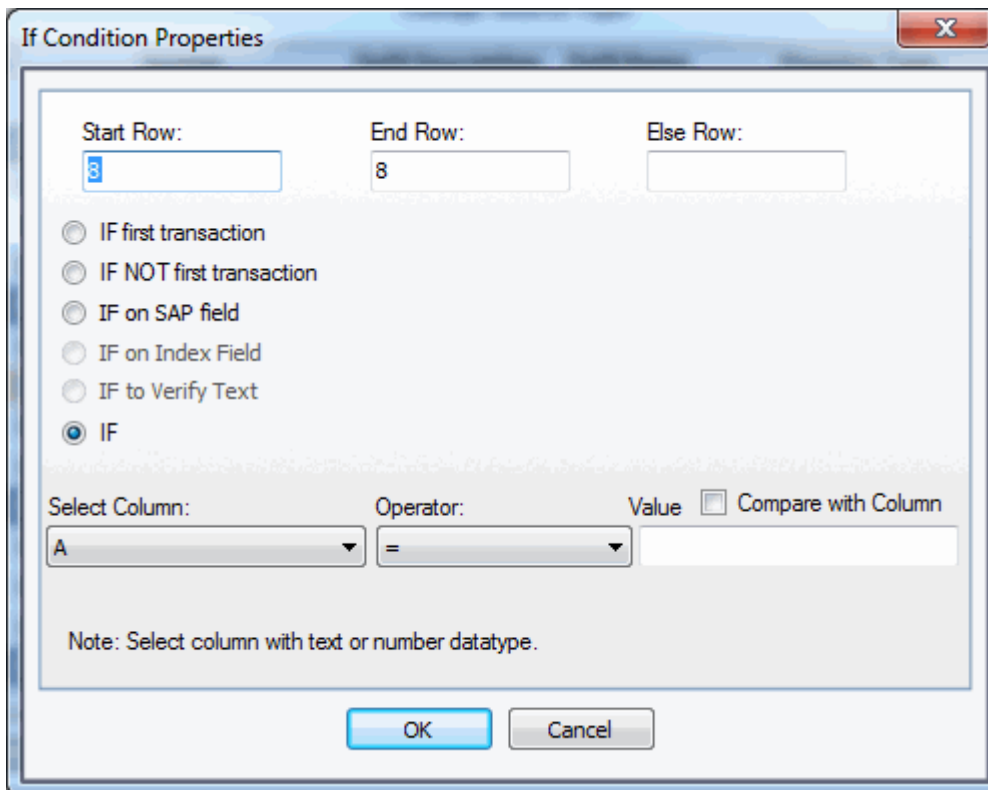
NOTE: The comparison of the constant value specified in the Conditional Run statement with the value in an Excel column depends upon the Data read preferences Microsoft Excel defaults. See "Application Defaults" for more information.

For example, suppose column D of your Excel worksheet contains a decimal value and TRANSACTION needs to execute only those rows that contain the value 3.141 (value of PI) in column D. In this case, column D is formatted to show numbers only up to one decimal place, but internally column D stores up to three decimal places. If the Data read preferences is set to As stored, the condition required will be "D=3.141". However, if this option is set to As displayed, the condition required will be "D=3.1".

A Conditional Run command in the Mapper is equivalent to using a +IF SHUTTLEScript command.

## Adding Conditional Run statements

- Click  (Conditional Run button) to set conditions for executing the selected portions of the TRANSACTION script, according to the conditions specified in the If Conditional Properties dialog box, as shown below.



The If Conditional Properties dialog box settings are:

**Start Row/End Row:** Type the Start Row and End Row you want to use, if other than the row or range of rows you initially selected on the Expert Tab: Recording Preview.

**Else Row:** Type the Else row where you want to store the If-Else row value.

**IF first transaction:** This setting is used to address SAP transactions in which dialog boxes, for example, occur only once on the first transaction record. Click this setting to execute the rows specified for the IF first transaction condition on the first record in the data source. For example, when the Classification View for a material is being changed, the Class Type dialog box pops up on the first record only, but not on any subsequent records.

**IF NOT first transaction:** Similar to the IF first transaction setting, this condition is used to address SAP transactions in which dialog boxes, for example, occur after the first transaction.

Click this setting to execute the rows specified for the IF NOT first transaction condition on all records after the first record in the data source.

**IF on SAP field:** Similar to IF condition. Condition applies only to SAP fields that are enabled. It can apply to disabled fields only if their corresponding screen is enabled in the mapper. However, IF on SAP cannot apply to SAP input fields. IF on SAP is not available in batch mode recording.

**IF on index field:** Use IF on index field with loops to search and update based on multiple search values. With IF on index field, you can find a row based on a value in a given column. To map, select only those fields that need to be updated. In the IF Condition Properties box, select If on index field. In Select Field, select the field you want to search on, select the operator, specify the value or the Excel column from which search values will be read. To replace the value in all matched instances of that field, click Find All. IF on index is available only for scripts recorded in GUI scripting mode.

**IF on verify text:** Available for imported A1 Dataloader scripts and for TRANSACTION scripts recorded with GUI scripting. Confirms that a value in Excel corresponds to a value in SAP. It does not change or remove data.


**IF:** Click the IF setting to base the condition on an Excel column or Access table/column you specify for the condition. When IF is selected, the Select Column, Operator, and Value fields are displayed near the bottom.

**Select Column, Operator, Value:** When the IF condition is selected, select the Excel column or Access table/column you want to use for the condition, select the Operator for the condition, and then type the Value (text or number) for this comparison. When you select a column, the screen field and condition then appear in a tooltip.

The allowed operators are:

=, >, <, >=, <=, <>, {BLANK}, {NOT BLANK}, {STARTS WITH}, {CONTAINS}, {ENDS\_WITH}, {BETWEEN}

**Compare with Column:** Select this check box to compare the Select Column (Excel column or Access table/column) to the column you select. If you mark the Compare with Column check box, the Value field displays a selection list of the columns available for this comparison.

When you complete the If Conditional Properties dialog box, the Mapper displays your changes using two  Conditional Run icons connected by a line.

## Adding notes

You can add notes in the Mapper Notes column to If condition rows. However, notes cannot be seen if the script is opened in an earlier version of the product.

## Validating data

When you create a script, you can ensure that anyone who runs that script first validates the data in the accompanying data sheet before posting it. With validation included in the script, you do



not need a separate validation sheet. Validation is off by default and must be turned on in the mapping screen.

Before you enable validation, ensure that the commit code appears only at the end of the script. If the script contains multiple commit codes, it is best practice not to enable validation.

Commit codes are used to confirm the uploading of data into SAP. If a transaction script that has validation turned on also contains more than one commit code, the script does not run. This prevents any partial upload of data.

Validation is available for all data sources.

The Run pane displays the Validation Log Column. You can also edit the column in the Run pane.

The screenshot shows a configuration window for running a script. The 'Select Data File' field contains the path 'My Documents\WinShuttle\TRANSACTION\Data\FB50.ACCDB'. The 'Table' dropdown is set to 'FB50\_Header' and the 'Log Column' dropdown is set to 'Header\_Log'. The 'Validation Log Column' dropdown is set to 'FB50\_Detail' and is highlighted with a red border. Below these are a 'Run Reason' text box, 'Run Options' set to 'Run Now', and a 'Launch GUI' checkbox. At the bottom are 'Validate' and 'Run' buttons.

## Validation on loops

For Excel, validation works for all line items and the validation log messages are written next to the individual line items. For Access tables, the innermost loop table is used for writing validation messages. The validation is performed on the individual line items along with their header values and does not check the integrity of the complete document.

## Validating for cell-based mapping

When you turn on validation for the script, you have the option to define a validation column. In the Mapper, the Validation log row appears below the Log row. Drag to map the column as you would any other column.


The Run pane displays the Validation Log Column. You can also edit the column in the Run pane.

## Validating linked scripts

Linked scripts, both Excel and Access, can be validated. Validation on linked scripts is possible for the following configurations:

- Column-based mapping and cell-based mapping in Excel forms
- Column-based mapping in Access forms
- Column-based files linked to cell-based files and vice versa
- Linked scripts with different types of mapping


### To turn validation on or off

- Click Validation , and then click Validation On or Validation Off.

### Validate in the Run pane

You can validate data to ensure that there are no errors before you upload it. When you validate, the data is not uploaded.

### To validate a data file

1. Click Validate .
2. Correct any errors or make changes, and then click Run to upload the data.

### Validate by list

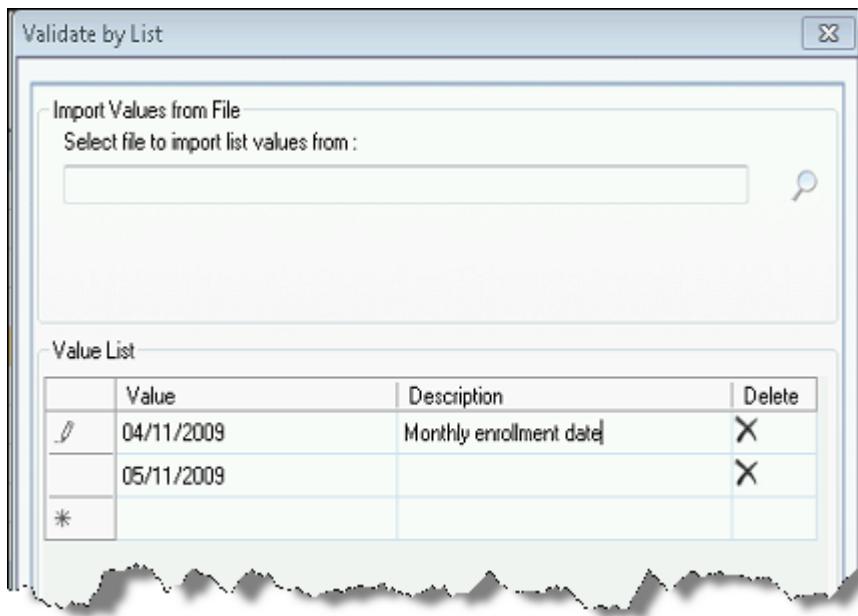
With the Validate by list feature, you can restrict values for a field.

### To validate by list

1. In the Mapper, select the field you want to validate or restrict values on.
2. Click the Properties pane and select the Validate by list check box.
3. Click the values button below the check box to create the validation list.
4. Create your value list by using one of these methods:
  - Import values from an Excel file
  - Import values from a CSV text file
  - Type in values manually

Note that these values are statically stored with the TRANSACTION script and do not change when the values in Excel or text file change during a run.

This list is available for input instead of the SAP values list when you use the look-up function from the add-in for Excel. In addition, when you validate or run this script, the selected field values will be validated against that list. If there are any errors, the corresponding row or record will not reach SAP for posting or validation and the user is notified of the error.



## Compensating for missing SAP screens and fields

Sometimes values required by an SAP screen are not available from your data sheet or the transaction script. If there are additional fields on the SAP screen which are not present in your script, you can enable them in the mapper area. TRANSACTION captures these additional fields as disabled fields which can be seen from the filter settings on the Expert tab. Uncheck hide all disabled fields, choose the fields that you need and map them with relevant column in the data sheet.

Sometimes, SAP encounters a screen that appears based on what data is used in previous steps. Skip screen if not found is available for recordings made in all modes.

### To enable a transaction to complete regardless of screens

1. Select the required field. Click the Expert tab.
2. On the right-hand side of the Mapper, click the Properties slide-in.



*The Properties slide-in that includes the Skip Screen If Not Found setting.*

**Description:** Displays the SAP field description shown in the Field Description column.

**Field:** Displays the SAP technical field name shown in the Field Name column.

**Source:** Displays the source of the data (either SAP or Excel).

**Destination:** Displays the destination of the data (either SAP or Excel).

**Skip Screen If Not Found:** Select this check box for screen fields to ensure that a transaction is completed whether the screen appears during the run or not. If a screen appears during a TRANSACTION, the appropriate data is uploaded. If a screen is not used, the Transaction continues. One script can be used for all runs of a TRANSACTION.

**Limitations:**

*Skip screen does not work for individual line items. If skip screen is inside a loop, either it applies to all line items or to none.*

Sometimes fields in your upload document are empty or you do not want to upload data from those fields. To ensure that the transaction continues despite those fields, you can set the script to skip those fields. To use the Skip Field indicator, you insert the indicator character in the Excel columns and rows or Access tables and columns that you do not want to update.

For example, suppose you want to update Material Group for all materials, with the exception of a selected few fields. Select the Skip Field indicator and add the indicator "\" to the indicator box. Modify the TRANSACTION script by inserting "\" in the Material Group column for each row that you do not want to update and save the changes. When you run this TRANSACTION script, the Material Group fields that contain "\" are not updated.

**To enable a transaction to complete regardless of fields**

1. When you have completed mapping, and before you prepare the script for approval or for running, open the script in the Run pane.
2. In the Advanced Run Settings slide-in, click the Skip Field Indicator. For more information, see Advanced Run Settings.

**Skip Transaction if Empty**

Sometimes you want to make some of the fields mandatory, and do not want to process the Transaction if these field value are not entered. You can use the Skip Transaction if Empty property.

For example, if you do not want to upload a record if it does not contain currency value, you can set the Currency field Skip Transaction if empty property. So, if in the data file, currency value is blank for a record, the whole record will not be posted to SAP.

## To set a mandatory field in TRANSACTION

1. Select the required field. Click the Expert tab.
2. On the right-hand side of the Mapper, click the Properties slide-in.
3. Click Skip Transaction if Not Found.
4. The skip field indicator must be set in the Advanced Run Options and the indicator must be added to the appropriate row in the Excel file.

## Undo and Undo All

### Undo mapping

After selecting a particular destination type user can undo any step performed in mapping. The number of levels of undo is unlimited.

### To undo mapping

- Press CTRL + Z or click Undo to see a list of actions. Click the action that you want to undo.

### Undo All

You can undo all your recent mapping changes by clicking the **Undo All** button that is available on the Edit tab. Note that Undo All undoes only those changes that could be undone with individual clicks. To make Undo All available again, you must make a change to the file. You are prompted to ensure that you want to undo all changes.

Undo updates the first three rows of the Preview.

Changes are permanently saved only when you click **Save** or **Save As**. When the file is saved, **Undo** and **Undo All** are no longer available.

### Update mapping

Mapping can be changed, even after the script has been saved. On the Mapping pane, click Update mapping.


### Undo on multisheet mapping

Undo action is applied to that sheet that is currently shown in Preview.

## Mapping long text

### To map the long text to the field you want

- For batch mode: depending on maximum number of lines that your long text can have, map the long text fields (representing each line in the editor) with respective Excel columns.

26	<input checked="" type="checkbox"/>	Text editor text line	RSTXT-TXLINE(03)		test
27	<input checked="" type="checkbox"/>	Text editor text line	RSTXT-TXLINE(04)		my own


-or-

- For non-batch mode: In the mapping pane, click Long Text.

All the long text strings that were recorded in the transaction appear. You can now map them to an Excel sheet or an Access table.

## Editing a TRANSACTION script

When the Mapper is open, you can use the TRANSACTION Editor to manually enter TRANSACTION Script commands to your TRANSACTION script:

- Select Edit from the Main Menu, and then click  Edit Transaction script recording button to display the TRANSACTION Script Editor as shown below.

```

MM02b.TxR - Winshuttle TRANSACTION Editor
Generated by Winshuttle TRANSACTION 10.3.111.2191
+HTTRANSACTION script for TCODE MM02
+MDSSWE6
+MDCBmargaretz
+MDDT1/9/2012 2:33 PM
+MDPC
+MDCM
+TV10.3.111.2191
&AvgRecordTime:2.89139261428571
#XS0
#DSMicrosoft Excel
#F1C:\Users\allfiles\Documents\Winshuttle\TRANSACTION\Data\MM0b2.xlsx
&GSpreadSheetID:
#S1Sheet 1
+SR2
+SA10
&MAPPINGPATTERN:0
&<SHEETTYPE><TypeVal>0</TypeVal><HeaderSheet>sheet 1</HeaderSheet></SHEETTYPE>
#L1A
#L30:
#L20:C
+RL700
+BI0
+RS1
+TXMM02
+TTChange Material
+SPSAPLMGMM
Line 13, Col 21

```

The TRANSACTION Script Editor contains the current TRANSACTION script you have open in the Mapper. You can use any of the TRANSACTION Script commands to manually edit the recording as needed. See *TRANSACTION Script Command Reference* for a complete description of the TRANSACTION Script Command set.

### Ctrl+F or the Find Button

Use Ctrl+F, the Find button, to search for text or commands in the Editor file.

Be sure to verify that any changes are correct. See the *TRANSACTION Script Command Reference* to ensure that the selected commands will implement the changes you want.

## Auto mapping


You can use auto mapping to automatically map all SAP data fields in the TRANSACTION script recording or to map selected SAP fields to Excel worksheet columns.

**IMPORTANT:** auto mapping is only available when Excel is selected as the data source.

## Auto mapping from the three-panel screen

Use this procedure to perform Auto Mapping from the Map screen after the TRANSACTION script recording is successfully completed:

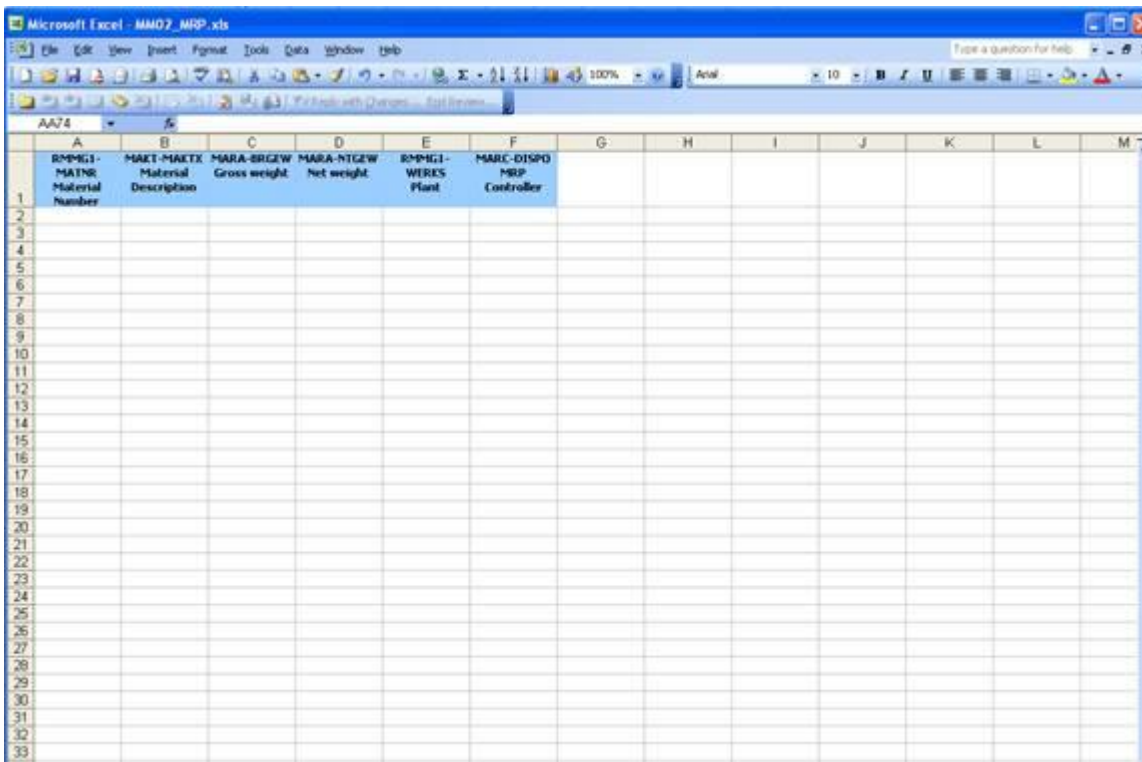
1. On the Map screen, click Auto Mapping as shown below.

2. On the Generate Auto Map Excel File dialog box, type the path for the Excel spreadsheet in which you want the preview to be generated, or click  (Folder button) to select the Excel spreadsheet from the TRANSACTION script folder as shown below. Specify the sheet name.




3. When selected, the preview is automatically mapped and the Run screen is displayed with the specified Excel spreadsheet listed as shown below.

When the TRANSACTION script recording is run, the auto mapped Excel spreadsheet is uploaded to SAP as shown below. This sample shows a run for creating the Material Master (MM02).



## Auto Mapping from the Mapper

- When the TRANSACTION script recording is completed successfully, click Create Mapping to display the Mapper.


In this case, you can click Auto Map button  on the Basic Tab: Settings to automatically map all SAP fields or selected fields to columns in the Excel worksheet in the Basic Tab: Recording Preview, including the rows containing selection actions.



## Auto Mapping for cell-based items

If your script includes cell-based items, you can use automapping.

### To auto map a cell-based transaction

1. In the Expert tab, specify ID columns and any loops the script requires.
2. In the Basic tab, click AutoMap . The Auto Map wizard appears.

Select the header items and any items that are not in loops. Specify the Excel column and row starting from B2. If the script does not contain header data, the preview will be empty. Click Next.

If the script contains loops, specify the starting column for the loop items. The log cell and the Validation column or cell maps to the next available column or cell.

## Downloading from SAP

TRANSACTION allows you to download or read data from SAP. This command allows you to assign columns for SAP fields which are being downloaded.

You can set up the TRANSACTION script for the download by using the Mapper's drag-and-drop feature or by using the options in the Mapping Type column.

### To use the drag and drop feature

- Drag the SAP field you want to Read From SAP to the appropriate data source field in the previewer.

### To use the options in the Value column

- Specify in the Value column the Excel column or the Access field to be used, according to the data source.

## Read from SAP

The Read from SAP button acts as a kind of Auto Map in reverse—it downloads data from SAP and automatically maps the downloads to the columns or cells of your data source.

### To automatically download from SAP and map

- On the Mapping pane, click Read from SAP.

The button marks only field and log rows for download.

Key Points:

This command is available only for SAP version 4.6 and later.

Downloading can be performed when the recording is created using Non-Batch Input Mode and GUI scripting. See "Entering the Recording Properties" for more information.

Assign columns with no data as receiver columns. TRANSACTION does not check if there are any values present in the assigned Excel receiver column. It overwrites all values in the receiver columns, even when no value from SAP is returned.

TRANSACTION does not format the Excel receiver column. The Excel column must be formatted to appropriate data types before running the TRANSACTION script. For example, if the column is not formatted as Text and a value of 004 is assigned to that column, Excel converts this value to 4. To retain the value of 004, format that column as Text before the run.

This command requires additional authorization in SAP.

TRANSACTION may run a little slower when this command is run. For this reason, use this command only when required. If the TRANSACTION script does not have any fields marked as Read From SAP, TRANSACTION automatically switches back to update only mode.


NOTE: This command works the same way for an Access database. Simply substitute <Access receiver field name> for <Excel receiver column ID>.

## Running scripts

### Running multiple or linked files

TRANSACTION supports recording of only one SAP transaction for each script file. If you want to use multiple transactions in a business process, you can record multiple scripts and link them together. The Link Script setting automatically launches and executes TRANSACTION scripts in sequence. You can even generate data from one script to be used in another script. You might use one script to extract data to another column in a datasheet, and then the second script processes the extracted data. Note: You cannot chain a script mapped to Google spreadsheets.

#### To link scripts

1. In the Run pane, click the Advanced Run Properties slide-in.
2. Select this check box to enable link scripting and then specify the next TRANSACTION script name to link multiple TRANSACTION scripts. You can click Search button  to find scripts as needed.

Carry Settings: In addition to specifying the TRANSACTION script name, you can also mark which execution settings are to be carried forward to the next TRANSACTION script by using any of these check boxes:

File: Use the same data file.

Sheet: Use the same sheet.

Start and End Row: Use the same Start and End rows. Both Start and End Row settings are carried forward together.

NOTE: If the path of the next TRANSACTION script is left blank, and only a file name is specified, then the linked TRANSACTION script path defaults to the path of the current TRANSACTION script.

#### Validating linked scripts

Linked scripts, both Excel and Access, can be validated. Validation on linked scripts is possible for the following configurations:

- Column-based mapping and cell-based mapping in Excel forms
- Column-based mapping in Access forms
- Linked scripts with different types of mapping

## Scheduling runs

The built-in Scheduler allows you to run your TRANSACTION files in an unattended mode at a user-defined pre-scheduled time. Any script can be set up to run automatically on a schedule by using the Run Later (Scheduling a Run) option on the Run screen. You can also set up a new run using Scheduler, as described below.

When you schedule an automated run, TRANSACTION performs the following process:


- Opens up at the scheduled time
- Makes a connection to SAP
- Performs the data transfer to and from SAP
- Closes the SAP connection
- Closes itself down, depending on the whether Close application after running on the Advanced Run Options pane is selected.

You can also use the Scheduler to edit any of the settings for the scheduled jobs listed or to delete a job run as needed.

### Selecting the Scheduler

- Click Schedule a run to create a new pre-scheduled job run.

When you select Schedule a run, the Windows Open dialog box is displayed. Select and open the script you want to schedule. Use the Schedule Upload Wizard to enter the settings for the run.

Select Tools from the Main Menu, and then click  Scheduler button to display the Scheduler screen as shown below.

## Field Definitions

The Scheduler screen settings contain:

### **TRANSACTION script**

Displays that name of each TRANSACTION script included in the scheduled run.

### **Data File**

Displays that name of each data file (Excel spreadsheet) included in the scheduled run.

### **Schedule**

Displays the pre-defined schedule for this job run including the time and day of the month, and the starting date for the schedule.

**Last Run Time**

Displays the last date and time when the scheduled job was run.

**Next Run Time**

Displays the next date and time when the job is scheduled to be run.


**E-Mail Notification**

Displays the recipient's name for the e-mail notification that is sent when the scheduled run is performed.


**Run on Console**

If the query is to be run from the command line, select Run on console.


You can edit any of the jobs in the scheduled run using these settings:

- Click  (Edit button) to change the line item.

When you select Edit, the Windows Open dialog box is displayed. Select and open the TRANSACTION script schedule you want to edit. Then use the Schedule Upload Wizard to enter the settings for the run.

- Click  (Delete button) to delete the line item. Then accept the confirmation message to remove the line item.

**Running a Recording (Access Data Source)**

The Run screen allows you to enter the run-time settings you want to use and then click  (Run button) to execute the recording as shown below. You can also click Validate to confirm that data in a file does not have errors, without uploading the data to the server. You do not have to create a separate validation file.

Validation is recommended when a TRANSACTION script is passed to a Runner user. The Runner user first validates the data, and then runs the file to upload the data.

Before you run a mass update and start modifying a large number of records, we recommend that you first do a test run with one or two records. Once you are satisfied with the test run, you may proceed with your mass update.

**NOTE:** TRANSACTION supports recording of only one transaction for each TRANSACTION script. If you want to use multiple transactions, you can record multiple scripts and link them together.

When the Scheduler is used, TRANSACTION automatically connects to SAP, runs the transaction on a predefined schedule you create, and closes the connection with SAP upon completion. You can send an e-mail notification indicating that the scheduled job was performed.

## To schedule a run

- On the Run pane, click Run Later, and then click Run .

Schedule Upload Wizard walks you through the scheduling process.

## Run options and troubleshooting errors

**Run Now:** Use this option to execute the TRANSACTION script immediately.

**Run Later (Scheduling a Run):** Use this setting to schedule an automated upload into SAP.

**Run on Errors:** After you correct any errors identified on the initial run, you can use Run on Errors to run the TRANSACTION script only for the rows marked as errors.

When the TRANSACTION script is run initially, the errors are marked using the background color or a text indicator (default text is "Err:") for the Log column cells that contain messages returned from SAP. See Error Management for more information.

**Run Step-by-Step (Stop on All Screens):** Select this option to execute the transaction by launching SAP GUI. When you select this option and click **Run**, a confirmation message is displayed.

This troubleshooting selection stops at every screen in the TRANSACTION script. A dialog box containing the OK code and the Enter button appears on each screen. You click Enter in the dialog box to move from screen to screen. In this mode, you can view all of the values being uploaded and pinpoint the exact location and reason for the error you are encountering.

**Run Step-by-Step (Stop on Errors only):** Select this setting to execute the transaction by launching SAP GUI. When you select this option and click Run, this troubleshooting mode stops at the point where the error is occurring in order to allow you to view what is occurring and to take corrective action to fix the problem.

**Run Step-by-Step on Errors:** Select this setting to run on the rows that returned an error and stop on the screen where the error occurs

### Key considerations for using the Step-By-Step Run Options:

You need appropriate authorization to run these troubleshooting modes. Most people do not have authorization to run the Step-by-Step modes in a Production environment. If you do not have the proper authorization, you can simulate the transaction manually using data from the Excel worksheet to determine why and where the error is occurring.

TRANSACTION allows you to troubleshoot all rows or just one row. If you want to troubleshoot only one row, insert the row value in the Start Row field and select the desired troubleshooting mode.

## Advanced run settings

You can click the horizontal Advanced Run Options bar on the far right of the Run screen to display the Advanced Run Options slide-in as shown below. This slide-in allows you to enter the run-time settings for TRANSACTION advanced run features.

### Auto run the TRANSACTION script when opened

Mark this check box to allow TRANSACTION to automatically run the selected TRANSACTION script upon opening. You are not required to press the Run button to execute the run. Leave this checkbox blank to start the TRANSACTION script run manually.

### Close application after running

Mark this check box to close TRANSACTION when the run is finished. Leave this checkbox blank to keep TRANSACTION open when the run ends.

### Extended log

Select this check box to show all of the messages encountered during the run of a transaction. These include warning messages, informational messages, and any other messages returned by SAP. Clear this check box blank to show only the last message returned by SAP. NOTE: The Extended log feature is not available for recordings made in batch input mode.

When you select Extended log, you can select one of these responses:

- Select As Text: Stores the Extended log in a separate column. When this setting is selected, the Extended log column is enabled.
- Column name: Displays the messages provided through the Extended log in this column.
- Select As Comment: Stores the messages as comments in the Log Result column that is specified in the Run screen. When this setting is selected, the Extended log column is stored as a comment in the Log Result column.

### Warning log column

Select this check box and specify the column for warning messages that appear in Google spreadsheets.

### Append time stamp to SAP log messages

Select this check box to append a time stamp of the run at the end of every log message returned from SAP. For all data sources, the time stamp appears in the current regional settings of the computer where the client is hosted. This option is available for all data sources, including BAPI scripts, and is available only for run and not for validate.

### Skip Field indicator

The Skip Field indicator tells TRANSACTION to ignore the update of certain fields in a record or records. Select this check box to turn on this setting. If you check this setting, you can also specify the character to use as the Skip Field indicator. (By default, "\" is the indicator.) To use

the Skip Field indicator, you insert the indicator character in the Excel columns and rows or Access tables and columns that you do not want to update.

For more information, see ***Compensating for missing screens and fields***.

### **Delay between transactions (ms)**

This setting delays the calls to SAP between transactions by n milliseconds. It can be used to slow the processing of the TRANSACTION script.

### **Link Script**

Select this check box to enable link scripting and then specify the next TRANSACTION script name to link multiple TRANSACTION scripts.

The Link Script setting is used to automatically launch and execute another TRANSACTION script in sequence when the current TRANSACTION script has finished running. You can use this setting to organize a chain of TRANSACTION scripts when a business process requires multiple transactions to run in a certain order.

Carry Settings: In addition to specifying the TRANSACTION script name, you can also mark which execution settings are to be carried forward to the next TRANSACTION script:

File: Use the same data file.

Sheet: Use the same sheet.

Start and End Row: Use the same Start and End rows. Both Start and End Row settings are carried forward together.

---

NOTE: If the path of the next TRANSACTION script is left blank, and only a file name is specified, the linked TRANSACTION script path defaults to the path of the current TRANSACTION script.

---

**Approval process for linked scripts:** You can send linked scripts for approval, but only one review process applies to all scripts.

If no data-review is required, you can run the data sheet for linked scripts from TRANSACTION.

**Validating linked scripts:** Linked scripts, both Excel and Access, can be validated. Validation on linked scripts is possible for the following configurations:

- Column-based mapping and cell-based mapping in Excel forms
- Column-based mapping in Access forms
- Column-based files linked to cell-based files and vice versa

If one of the following conditions is true for the files, you are prompted to provide the name of the log column before the second file runs:

- The second file is an Access file with loops
- The second file uses cell-based mapping with loops

You can run linked scripts from the RUNNER add-in for Excel.



## ALF file

Select this check box to use an Auto Logon File (ALF) to automate your SAP logon when you run the script. Specify this setting when you plan to schedule the script to run in the future or regularly so that SAP opens automatically.

This setting is especially useful when you want to schedule a run, or if you do not want to type the logon information each time you run a TRANSACTION script. See **Auto logon files** for more information.

## Backup SAP data

Select this check box to automatically create a backup copy of the current SAP data fields using another Excel worksheet or Access database table just before running the TRANSACTION script. This setting is useful when you want to have the ability to roll back the data.

NOTE: To use the Backup SAP Data setting, you must record the TRANSACTION script using the Non-Batch Input mode.

## Run Not Posted

Mark this check box to ensure that the rows in an Excel sheet, Access table, or Google spreadsheet were not run previously. TRANSACTION checks the TRANSACTION script for log information. (TRANSACTION stores all of the messages returned by SAP in your data file). If log information is present in the TRANSACTION script, the run stops.

## Excel Save and Close Options


Do Not Save: Excel is not saved automatically after the run has finished.

Save After N runs: Excel is automatically saved after N runs (default value 50). The number can be changed in the text box next to the option.

Save at End of Run: Excel is automatically saved at the end of each run.

Close Excel after run: Select this check box to close Excel after the run has finished

## Writing Headers

If a Google spreadsheet or Excel file is not available with a script, you may need to write headers to the sheet. You can write headers from the rich client and also from the add-in for Excel. In the rich client, the Write Headers button  appears on the Run pane.

### Write headers in cell-based mapping

It is possible to write headers for cell-based mapping in Excel and Google sheets.

In cell-based mapping, headers are written to the cell immediately to the left of the mapped cell. If the prospective header cell contains data, you can continue to write the data without the header or you can cancel the entire mapping.


- If a field is mapped to C4, the header is written to B4.
- If a header appears in column A, you can choose to continue to map the data but not write the header, or cancel the entire mapping.
- If an ID-based loop is mapped to column C and data appears in C1, you can choose to continue to map the data but not write the header, or cancel the entire mapping.
- If a range-based loop starts in column C4, the header appears in column C3. If the prospective header cell contains data, you can continue to map the data without the header or you can cancel the entire mapping. To write the header, the range must be specified.
- For Automap and Read from SAP, the header is written only if the header cell is empty.
- For multiple cells selection that is dragged to the mapper to map uploads, the header is written only if the header cell is empty.

## Publishing macros in an Excel workbook

In TRANSACTION, you can publish macros in Excel workbooks and customize aspects of the variables. You can also edit macros in that are already published to workbooks from within TRANSACTION without opening VBA. Macros created in TRANSACTION can be run in the TRANSACTION and RUNNER clients and in the RUNNER add-in for Excel.

- The Excel file that you publish to must be macro enabled.
- No VBA validation of the macro is possible. You must manually ensure the VBA syntax is correct.


### To publish a macro

1. Create the macro in or copy it to a .txt file.
2. In the Run pane, click Publish Macro .
3. In the Winshuttle Macro Publisher dialog box, click Import.

Browse to the location of the .txt file.


4. Click Save to publish the macro.

### To edit macros already published to workbooks


1. In the Run pane, click Publish Macro .
2. From the list in the Winshuttle Macro Publisher dialog box, select the macro you want to use. Click Edit.
3. The text editor opens to accept your changes. Click Save.

### To enable a macro as a button in a workbook

Note: Excel files that contain a space in their filename cannot contain buttons.

1. In the Run pane, click Publish Macro .
2. From the list, select the macro you want to use.
3. Click Make Button.
4. In the Caption box, name the button and in the Cell Address box, specify where you want the button to appear. Click OK.

### To run a macro

1. In the Run pane, click Publish Macro .
2. From the list, select the macro you want to use. Click Run.

### Macro templates

Macro templates are available for you to customize to your needs. To download templates, go to [wise.winshuttle.com](http://wise.winshuttle.com).

The form can be generated from only one method specified in the `<MN></MN>` tag. Variables must be coded in the following format:

```
<WSM_Declaration><VN>Variable Name</VN><LT>Label Text</LT><DV>Default Value</DV><DT>Data Type</DT><MN>Method Name</MN></WSM_Declaration>
```

### To add your own variables

1. Click Publish Macro button.
2. Click Import to import the macro or template.
3. Click Edit.
4. Add the variables. Click Save.

## Reviewing transaction history

The TRANSACTION Log Viewer provides you with a comprehensive log of all successful and unsuccessful transactions.

### To use the log viewer

- Click the **Tools** tab, and then click Log Viewer  button to display the Log Viewer.

The Log Viewer contains these settings:

**Total records uploaded through TRANSACTION.** Total number of records uploaded through TRANSACTION.

**Total time saved.** Displays the calculated total time saved in hours (hhmmss) and in working days (00.0) for a Full Time Equivalent (FTE).

### Viewing ROI details

Click any of the script names in the list to view the ROI details for that line item at the bottom of the screen.

### TRANSACTION scripts log list

The Log Viewer lists each of the TRANSACTION script log files stored including:

**TRANSACTION script:** Displays the name of the TRANSACTION script recording that was run.

**Mode:** Displays the mode of recording. If the file was created using DIRECT template, DIRECT template is set as the mode of recording.

**Data File:** Displays the data file name used for the run.

**Upload data and time:** Displays the date and time that the TRANSACTION script was run. Information is in the format date hh:mm:ss and am/pm.

**System:** Displays the SAP system used for the run.

**User:** Indicates the user name.

**Client:** Displays the Client used for the run.

**Run Reason:** Displays the Run Reason entered on the Run screen for this TRANSACTION script.

**Records:** Displays the number of records processed in this run.

**Errors:** Displays the number of errors identified in this run.

**Run Time:** Displays the total time taken for the TRANSACTION script to run in the format hh:mm:ss.

**Tcode:** Displays the TRANSACTION code or, for DIRECT files, Function Module, name with which the TRANSACTION script was created and run.

## ROI chart

You can create an ROI chart that depicts activities in your TRANSACTION environment.

### To create the ROI chart

- On the Log Viewer screen, click Create Chart.

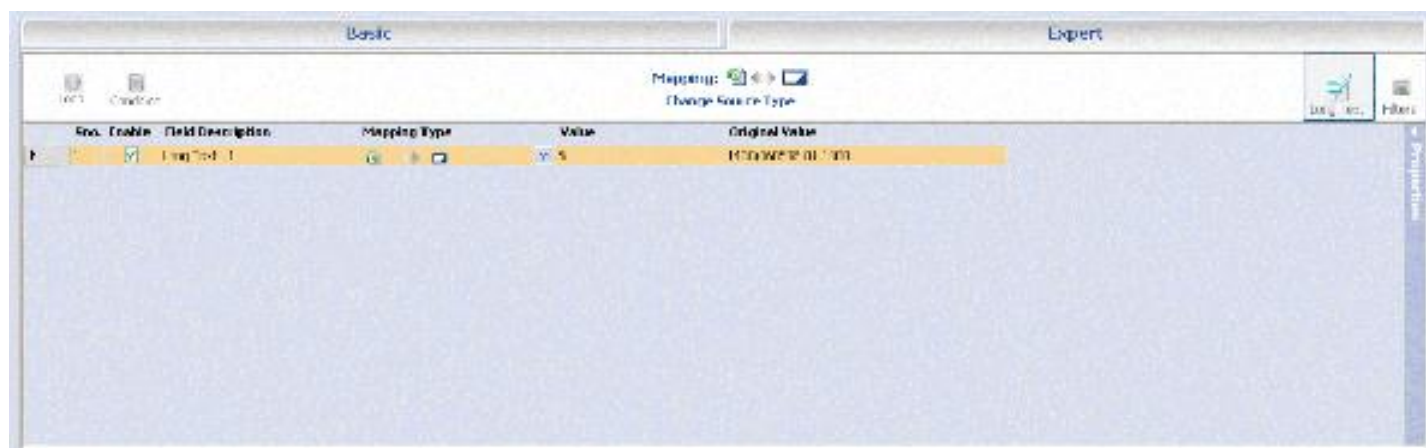
## Uploading long text

### To upload long text

- For batch mode recordings, break your test into several lines. Each line should not exceed 72 characters in length. Use a separate column for each line of long text to be included in the Excel worksheet. If you place all the long text in one column, the error "Text longer than field" appears.

-or-

- For non-batch mode recordings, you can directly upload the Excel cell values without breaking the text. Note the following:
  - You can only upload long text.
  - Long text fields cannot be used with If conditions or Do...While loops.



## Windows scripting

With Windows scripting, which is enabled through GUI scripting, TRANSACTION records the Windows file selector dialog box in transactions that allow files to be attached. The file location can be mapped to an Excel location. When you run the transaction, the file is selected from the specified location and the attachment is created.

To upload an attachment, the field must still be mapped to the Excel column you want in the mapper. The file path is `Windows_Dialog_file` with the default value as `untitled`. TRANSACTION sees only the dialog box; the name is added from the corresponding field in the Excel file.

A file name field is always recorded in the mapper, whether a file was chosen in the dialog box or not. If you don't want to upload a file, disable the dialog box fields. If you want to upload a file, map the field to an Excel column.

Limitations:

- Windows scripting works only with GUI scripting.

- The file is not attached if the button click that creates the attachment is not recorded by TRANSACTION and SAP.
- The dialog box is recorded even if you click Cancel. You must disable the Windows dialog box control that was recorded in the TRANSACTION script.
- Ensure that you close other instances of SAP.





## Programs used with TRANSACTION

### Winshuttle DIRECT

#### Recording scripts from Winshuttle DIRECT

##### To record a new script from a DIRECT template

1. Click New.
2. On the Record pane, click Select fields from a DIRECT template.

The DIRECT templates page appears. Select a template from the Winshuttle template repository, or browse to a locally-stored template. If not templates are listed, click Get/Refresh list to load available templates.

Enter info in any fields you require. You must add a title and you must add the time it takes for this business process to add one record by using the SAP GUI. The time setting is used by TRANSACTION for ROI calculation.

The SAP metadata details are filled from the DIRECT metadata details. Therefore, the creator for the TRANSACTION script is the same as the creator of the DIRECT template it is based on.

3. In the Import Direct file page, select from the fields as required.
4. Select individual fields as desired.

You can use Select all or Clear all for bulk selection or omission of fields in the currently loaded category and section. Fields marked "Mandatory" cannot be cleared.

5. Select groups of contiguous or noncontiguous fields by clicking CTRL and then clicking each row. Right-click the selected fields, and then click Select Row(s) or Deselect Rows(s).

Note: Fields marked "Mandatory" cannot be cleared.

6. Click Finish to save the file as a new TRANSACTION script. The Mapping pane appears. You are ready to map the template.

Note that loops are created automatically around table categories, but you must set properties.

#### All tab

The All tab displays all fields available in the imported DIRECT script for a selected category, regardless of its MR80 designation.

#### Mapping DIRECT scripts

The mapping for TRANSACTION scripts created using DIRECT templates is similar that of scripts that are created through the recording process. For more information about mapping, see *The TRANSACTION Getting Started Guide*.

Here are some ways in which DIRECT scripts differ from TRANSACTION scripts:

- The validation capability is not available for these scripts.
- IF on SAP field and Skip screen if not found conditions cannot be created over field uploads.
- Loops are automatically created around table categories.
- Loops can be edited only to change Excel properties: for cell-based mapping, the start row and end row; for column-based mapping, loop identifiers.
- Mapper properties, such as the start and end row for a loop, cannot be changed.
- New loops cannot be added to scripts, hence there is no possibility of nested loops.
- Among the Advanced Run options, Skip field indicator, and Back up SAP data are not available.
- Run the script as you would any other script.

## A1 Dataloader

TRANSACTION supports A1 Dataloader script files. Dataloader scripts can be imported into TRANSACTION. The Excel mapping and run properties are retained.


Dataloader scripts are stored in the folder \\mydocuments\A1\_DL\_Projects.

### Importing a Dataloader project

#### To import a dataloader project

1. On the Tools ribbon, click Import .
2. In the Import dialog box, enter the name of the file you want to import. A dataloader project has the extension .DLPrj. Click OK. The project is imported into TRANSACTION.
3. The Run panel displays the new .TxR file as an untitled file. Click Save As to name the file. You can now run the script.

#### To import an Easy Tester file

1. On the Tools ribbon, click Import  .
2. In the Import dialog box, enter the name of the file you want to import. An easy file project has the extension .ETF. Click OK.

The project is imported into TRANSACTION. The first script in the chain appears in the Run panel.

## Before you run new dataloader scripts in TRANSACTION

- Before you run a script that has been imported from Dataloader, remove the # prefix from Excel cells.
- Before you run a script that contains [+3] loop, go to the mapper and convert the loop to Index- based loop.
- Before you run a script the multiple loops, the representation of data in Excel should look the following examples:

Tab select	G/L Account	Debit/Credit Ind.	Amount	Button pres	select row	Tab select	G/L Account	Debit/Credit Ind.	Amount	Button pres	Button pres
Loop	50000	H	100	L-End	0	Loop	50000	S	100	L-End	
[+1]	50000	H	200	[-]		[+1]	50000	S	200	[-]	
[+1]	50000	H	300	[-]		[+1]	50000	S	300	[-]	
[+1]	50000	H	400	[-]		[+1]	50000	S	400	[-]	

Dataloader Excel files with multiple loops.

Tab select	G/L Account	Debit/Credit Ind.	Amount	Button pres	select row	Tab select	G/L Account	Debit/Credit Ind.	Amount	Button pres
Loop	50000	H	100	L-End	0	Loop	50000	S	100	L-End
[+1]	50000	H	200	[-]						
[+1]	50000	H	300	[-]						
[+1]	50000	H	400	[-]						
						[+1]	50000	S	200	[-]
						[+1]	50000	S	300	[-]
						[+1]	50000	S	400	[-]

In the example above, data from the first line item of the second loop is put in the Header row, not in the line items row.

## Mapping for imported Dataloader files

TRANSACTION recognizes the Dataloader column mapping and interprets it by mapping the field names to individual columns. If you want, after Dataloader files are imported, you can change the native Dataloader mapping in TRANSACTION.

## Looping

In mapped files, Dataloader uses symbols to indicate the loops: [+n] for the start column and [-] for the end column. Loop and End Loop commands are not included in the converted script, but the resulting Excel sheet indicates the presence of loops with these symbols at the start and end of each applicable row. Dataloader supports four types of loops: [+1], [+2], [+3], [+4]. However, in TRANSACTION, the specific type of loop is not indicated. Note: multisheet loops are not available for files imported from A1 Dataloader.

## [+3]

After conversion from A1, [+3] must first be converted to an index loop. For more information on index loops, see Expert Tab: Do...While Loop.

## Verify Text

In TRANSACTION, use the IF to verify text condition to test whether a value in an Excel column is the same as in SAP or not. If the values are the same, "Test passed" is added to the mapped column as a comment. If the values are different, "Test failed" is added to the mapped column as a comment. Text that is verified is not appended, changed, or deleted. NOTE: IF to verify text can be used only when GUI scripting is enabled.

You cannot compare text across screens. The text you are comparing must be on the same SAP screen.

### To set verify text

1. In the Mapping pane, click the Expert tab.
2. Click Condition.
3. Click IF to Verify Text.
4. Select the SAP field to compare. Click Compare with Column, and then specify the column in the Value box. Click OK.

## Get+Set Value

When an imported Dataloader script is displayed in Excel and if it has been recorded with GUI scripting, Get+Set Value downloads a value from SAP to one column, and then uploads the updated value from another column, all within the same sheet. After the AI file is imported, the Excel file, the marker %GSVal indicates the columns where the specified downloaded values will be added.

## Script chains

TRANSACTION imports chained Easy Test File scripts. When chained scripts are run, TRANSACTION uses only the SAP credentials that are attached to the first script.

If a script in a chain is disabled in Dataloader, it is not possible to load that script. To load it, you must go into Dataloader Manager and enable it.

## Dataloader commands not supported during conversion

The following commands are not supported during conversion:

- Auto logon
- Open Sap logon pad
- Run with recorded data
- Minimize SAP window

- Inverse rows of table controls before run

## Using TRANSACTION with Virsa Firefighter

When you want to use TRANSACTION with a session launched through Virsa Firefighter, under SAP defaults, select Execute Gui Scripting transactions in SAP GUI Session launched through Virsa Firefighter.

### Recording the transaction

For recording the transaction, specify the "transaction code" and "log on" to the desired SAP system. After successful logon to SAP, on recording properties dialog window select mode of recording as GUI Scripting

When you press the next button, a list of sessions launched through Virsa Firefighter is shown on the screen. Select the SAP GUI session that you want to attach with.

Record you transaction with the attached SAP GUI session. Map and save the TRANSACTION script

### Running the transaction

Open the TRANSACTION script and click the Run button. After successful SAP logon, List of sessions launched through Virsa Firefighter is shown on screen. Select a SAP GUI Session through which you want to upload the data.

TRANSACTION will upload the data using the session selected



## Running TRANSACTION at the command line

### TRANSACTION Script Command Reference

The TRANSACTION Script Commands allow you to edit the TRANSACTION scripts recording in the Mapper. See Editing a TRANSACTION script in the Mapper for more information.

You can also use the TRANSACTION script commands to pass these commands as parameters when you run a TRANSACTION scripts from the Command Line. See TRANSACTION and the command line for more information.

Each of the TRANSACTION Script Commands is described in this section.

#### **#AE**

Optional command.

When used, TRANSACTION closes after running the TRANSACTION scripts.

Example:

```
#AE
```

#### **#AR**

Optional command.

When used, TRANSACTION automatically runs your TRANSACTION scripts once it is opened. The Run button is not required to execute processing.

Example:

```
#AR
```

#### **#BD**

Optional command.

When used, TRANSACTION automatically create a backup copy of the current SAP data fields values.

Example:

```
#BD
```

### **#F1InputExcelFileName**

Required command.

When used, specifies the Excel spreadsheet path and name.

Example:

```
#F1C:\My Documents\Change MRP Controller.xls
```

### **#L1LogColumn**

Required command.

When used, TRANSACTION maintains the transaction log in the specified column of the selected Excel worksheet. Storing SAP data and returned messages in the same Excel worksheet allows you to view the processing results in one common place.

NOTE: Make sure the specified column on the worksheet is blank. Otherwise, TRANSACTION overwrites any data contained in the column with the returned SAP messages. The spreadsheet is saved automatically when TRANSACTION completes execution.

Example:

### **#L1G**

Maintains the transaction log in column G of the selected Excel worksheet.

### **#NTNext linked**

TRANSACTION scripts with complete path [,nCarryonSettings]

Optional command.

Use the #NT command to launch another (next) TRANSACTION scripts after the current one finishes running.

nCarryonSettings could be:



0	[Default Value] Do not carry on settings (Excel spreadsheet name, sheet name, start row, and end row) from current TRANSACTION scripts.
1	Carry over the settings from the current TRANSACTION scripts.

Examples:

```
#NTC:\My Documents\TRANSACTIONscript\ME01.TRANSACTIONscript
```

When Run is selected, TRANSACTION launches ME01.TRANSACTIONscript after the current TRANSACTION Scripts completes execution.

```
#NTC:\My Documents\TRANSACTIONscript\ME01.TRANSACTIONscript,1
```

The statement ignores the Excel filename, sheet name, start row, and end row values specified in the ME01.TRANSACTIONscript. Instead, TRANSACTION carries these values over from TRANSACTION scripts currently open.

Key Points:

Only one TRANSACTION scripts can be linked to the current TRANSACTION scripts. For example, if the current TRANSACTION scripts is MM01.TRANSACTIONscript, the ME01.TRANSACTIONscript can be linked to this file. However, when linking ME01.TRANSACTIONscript and ME11.TRANSACTIONscript, TRANSACTION gives you a syntax error that must be fixed before running the TRANSACTION Scripts.

Any number of TRANSACTION scripts can be chained. For example, MM01.TRANSACTIONscript can point to ME11.TRANSACTIONscript, which in turn can point to ME01.TRANSACTIONscript, and so on. TRANSACTION first runs MM01.TRANSACTIONscript, then ME11.TRANSACTIONscript, and ME01.TRANSACTIONscript.

The current TRANSACTION scripts cannot be linked to itself. There is no check to prevent indirect linking. For example, suppose MM01.TRANSACTIONscript links to ME01.TRANSACTIONscript, which links to ME11.TRANSACTIONscript, and ME11.TRANSACTIONscript links back to MM01.TRANSACTIONscript. This creates an infinite loop that causes TRANSACTION to continue until it is stopped manually.

A copy of the current TRANSACTION scripts can be linked to the original TRANSACTION Scripts. For example, suppose the links start with PA40.TRANSACTION script. The TRANSACTION Scripts can call PA40-1.TRANSACTION script, which in turn calls PA40-2.TRANSACTION script, and so on.

The linked TRANSACTION scripts starts only after all the records of the current TRANSACTION Scripts are processed.

The linked TRANSACTION scripts AutoLogon (ALF file) information is ignored. All linked TRANSACTION scripts use the SAP log on session of the current TRANSACTION scripts. In other words, only the original TRANSACTION Scripts can log on to SAP. All subsequent TRANSACTION scripts use the same SAP session.

All linked TRANSACTION scripts can use the same Excel spreadsheet or different spreadsheets. They can even use the same worksheet within one Excel spreadsheet or use a different

worksheet. Ideally, the same worksheet should be used in order to allow the log to be compared and reused after parsing it in Excel.

TRANSACTION displays last linked TRANSACTION scripts when it finishes running all linked TRANSACTION scripts. In other words, it does not reload the original TRANSACTION scripts that started the chain.

The log file indicates linked TRANSACTION scripts run activity.

If the path of the next TRANSACTION script is left blank and only a filename is specified, TRANSACTION looks for the next TRANSACTION Scripts in the current TRANSACTION scripts path.

### #S1SheetName

Optional command.

The default is Sheet1. If there's only one worksheet in the Excel spreadsheet, TRANSACTION selects it automatically. If there is more than one worksheet in the Excel spreadsheet, specify which worksheet (Sheet#) is to be used.

Example:

#S1Sheet2

Selects worksheet 2 in the specified Excel spreadsheet.

### #XS{ 0 | nnnnn | End } , [CLOSE]

Optional command.

This command controls how TRANSACTION saves and closes the Excel spreadsheet during and after each run.

#XS0	Indicates that TRANSACTION does not save the Excel spreadsheet at the end of a run. In this case, the Excel spreadsheet must be saved manually or any changes are lost, including the log messages returned by SAP. This subcommand is useful when there are multiple sessions of TRANSACTION running on the same Excel spreadsheet and only one of those Excel spreadsheets needs to be saved.
#XSnnnn n	Prompts TRANSACTION to save the Excel spreadsheet after every nnnnn transactions
#XSEnd	Saves the Excel spreadsheet at the end of a run. #XSEnd is the default value, unless otherwise specified.
CLOSE	If CLOSE is included with any of these subcommands, TRANSACTION closes Excel after each run. When the CLOSE subcommand is issued and the Excel session has files to be saved, Excel prompts you to save the files before closing.

Examples:

#XS100

Saves the Excel spreadsheet after every 100th transaction run.

#XS0

Does not save the Excel spreadsheet; it must be saved manually.

#XSEND

Saves the Excel spreadsheet at the end of a run.

#XSEND,CLOSE

Saves the Excel spreadsheet at the end of a run and then closes Excel. If Excel has any files to be saved, the user is prompted to save the files before closing.

#XS0,CLOSE

Does not save the Excel spreadsheet during the run. Once the run is finished, TRANSACTION attempts to close Excel. Excel prompts the user to save the file before closing.

### **\*Fv!FieldValue (read value from the spreadsheet column)**

Optional command.

This command is used to map data values stored in the spreadsheet to SAP. The \*FV command must always follow the +Fn command.

Example:

+FnRMMG1-MATNR

-FnMaterial Number

\*Fv!A

Maps the Material Number to column A of the Excel spreadsheet.

### **\*Fv?<Excel receiver column ID>**

Optional command.

TRANSACTION can download (read) data from SAP. This command allows you to assign a column for the SAP fields being downloaded.

Recommendation: Use the Mapper to assign Read From SAP values and avoid manual changes to the TRANSACTION scripts, unless required..

Example:

+FnEKKO-EKORG

\*Fv?K

Shows the SAP field EKKO-EKORG is read from SAP and added to column K in the Excel spreadsheet.

Key Points:

This command is available only for SAP release 4.6 and higher.

Assign columns with no data as receiver columns. TRANSACTION does not check if there are any values present in the assigned Excel receiver column. It overwrites all values in the receiver columns, even when no value from SAP is returned.

TRANSACTION does not format the Excel receiver column. The Excel column must be formatted to appropriate data types before running the TRANSACTION scripts. For example, if the column is not formatted as Text and a value of 004 is assigned to that column, Excel converts this value to 4. In order to retain the value of 004, format that column as Text before the run.

This command requires additional authorization in SAP.

TRANSACTION may run a little slower when this command is run. For this reason, use this command only when required. If the TRANSACTION scripts does not have any fields marked as Read From SAP, TRANSACTION automatically switches back to update only mode.

NOTE: This command works the same way for an Access database. Simply substitute <Access receiver field name> for <Excel receiver column ID>.

### **+ALautologin file.ALF**

Optional command.

When used, TRANSACTION uses the auto logon file (ALF) specified in this command to logon to the SAP system.

Example:

```
#ALC:\MY DOCUMENTS\TRANSACTION\TestSystem.ALF
```

When running this TRANSACTION Scripts, TRANSACTION uses the logon information stored in TestSystem.ALF file. As a result, it does not prompt for a logon.

### **+BI**

Optional command.

This command controls how a transaction is run.

Examples:

+BI1

Runs the transaction in Batch Input mode [Default setting].

+BIO

Runs the transaction in Non-Batch Input mode. This is similar to running the transaction in GUI.

See **Recording Modes** for more information.

### **+DBnnnnnn (Delay before SAP log on)**

Optional command.

This command delays the log on to SAP by nnnnnn milliseconds. Use this command to maintain the delay between various TRANSACTION sessions such as when the Scheduler cannot schedule sessions less than one minute apart, and sessions must be launched within that time frame. See **Scheduling runs** for more information.

This TRANSACTION script command can be passed from the command line as a parameter. If this is done, the command line parameters take precedence over the values specified in the TRANSACTION Scripts.

### **+DLnnnnnn (Delay between transaction run)**

Optional command.

This command delays the calls to SAP between transactions by nnnnnn milliseconds. This command can be used to slow down the processing of the TRANSACTION scripts.

### **+Do-While Loop**

Optional command.

This command provides a Do...While Loop mechanism for processing multiline transactions. TRANSACTION enters the Do...While Loop when the specified condition is met. Normally, this condition is determined dynamically. Fixed value <Check value> is compared against the value in the Excel column as indicated by <Excel identifier column id>.

Recommendation: Use the Mapper to manage your Do-While Loop and avoid manual changes.

Example 1:

## Winshuttle TRANSACTION Advanced User Guide

```
+Do While <A:Excel record type identifier column ID>,<H:Header identifier
string>,<L:Line Item identifier string>
...
+Fv!<Excel column id>
.....
@Fv!<Excel column id>
...
+Fv<Fixed Value>
...
*Fv!<Excel column id>
...
+Fv<Fixed Vale>
...
+LOOP
```

### Example 2:

This example shows that column A in Excel has item records for looping as identified with character D in that column. TRANSACTION loops through all of the rows in Excel spreadsheet, while the value of column A for that row is filled with D.

#### +DO WHILE A,H,L

```
'Start Do While loop from next line, Row Identifier is stored in Excel column A,
value of "H" indicates Header row while value of "L" indicates line item row
'-----
```

```
+SpSAPMF05A
+Sn1200
-SpMain Screen for Single-Screen Transaction: Outgoing Invoice
+FnBDC_OKCODE
?
-FnOK Code for this screen
+Fv=0005
```

```
+FnACGL_ITEM-HKONT(01)
-FnGeneral ledger account
@Fv!C
```

```

+FnACGL_ITEM-WRBTR(01)
-FnAmount in document currency
@Fv!F

+FnACGL_ITEM-MWSKZ(01)
-FnTax
@Fv!G

+FnACGL_ITEM-MARKSP(01)
+FvX

+LOOP
'End do while loop here..
'-----
+SpSAPMF05A
+Sn1200
-SpMain Screen for Single-Screen Transaction: Outgoing Invoice
+FnBDC_OKCODE
-FnOK Code for this screen
+Fv=BU
'-----

```

### Key Points:

There must be separate Header and Line Item (detail) rows in Excel. There can be only one Header record, followed by one or more line item records.

Header and Line Item records should be clearly marked in the identifier Excel column. TRANSACTION ignores the case, while validating the Excel identifier value for Header or Line Item rows. Also, the loop row cannot contain data for Header. However, Header information can be used in each Line Item.

TRANSACTION starts building the transaction from the row identified as a Header. If a value other than H or D is encountered, TRANSACTION skips all rows until it finds the next Header row.

Do not leave an empty row between Line Item records.

While using the Do...While Loop, avoid making unnecessary key fields. (Skip the transaction if empty or \*FV#<Column ID>). TRANSACTION uses the row type identifier column as the key column. If this column is left blank in any of the rows, TRANSACTION ignores the entire row and continue to the next record.

Key field (skip row, if value in Excel is empty for the field) has priority over Header/Line Item row types during the Do...While Loop. In other words, TRANSACTION first checks whether the row is selectable based on the key field settings. Then it performs validation based on the row type for Do...While Loop processing. If the row is skipped due to key field validation, it is not used as part of Do...While Loop processing.

**NOTE:** Do not mark any of the fields used inside the Do-While Loop as a key field. If any of the fields inside the Do...While Loop is marked as a key field, there can be unexpected results.

If you want to use incremental Excel rows, use identifier @ instead of + or \* while inside the Do...While loop. The chart below shows which row in Excel is used to retrieve the data.

TRANSACTION script element falling between Do- While Loop statement	Read from
*Fv!<Excel column ID> or *Fv#<Excel column ID> (Value from Excel)	Header row
@Fv!<Excel column ID> (Value from Excel)	Corresponding loop row

NOTE: This command works the same way for an Access database. Simply substitute <Access field name> for <Excel column ID>.

See also Handling Multiline Transactions.

### **+EL[!ExtendedLogColumn]**

Optional command.

This command allows users to retrieve extended log information for a transaction. Extended log information includes the complete message stack for a transaction that is run in SAP.

+EL	Retrieves the extended log and writes it as a comment in each log cell in Excel.
+EL!ExtendedLogColumn/+EL!Table.Field	Retrieves the extended log and writes it in the specified Excel column or table field in an Access database.

Example:

```
+EL
+EL!G
```

Writes the Extended Log in column G of the Excel spreadsheet.

```
+EL!MaterialMaster.ExtendedLog
```

Writes the Extended Log in the Extended Log field of the Material Master table in the Access database.

### **+EM**

Optional command.

When used, this command logs Extended Messages returned from SAP. Normally, TRANSACTION stores only the SAP message text. This text is placed in the assigned log column. However, the Message Type (E, I, S, and so on), Message No., and Message ID can be specified in the TRANSACTION Scripts.

The returned message includes these fields in the order shown below:

Message Type:Message ID - Message No - Message Text



Example:

```
"Selection criteria have been saved"
```

Message without +EM command (above).

```
"S:F4-431-Selection criteria have been saved"
```

Same message with +EM command, where S is Message Type, F4 is Message ID, 431 is Message Number, and the remainder is the actual Message Text.

## **&EndSession**

Optional command.

For ML81N and other transactions, the service entry sheets are created for the first purchasing document, the header item, but the subsequent purchase documents error out. The EndSession command resets the SAP session after each run.

```
&EndSession:<number>
```

Number specifies the number of transactions or rows after which the SAP session must reset.

```
+FnFieldName
```

Required command.

This command must follow the +Sn command. Each Screen Number (+Sn) may have more than one field to available.

Example:

```
+FnRMMG1-WERKS  
(WERKS = Plant)
```

## **+FvFieldValue (Fixed)**

Required command.

Use the +Fv command to specify a fixed value for a certain field. This command must follow the +Fn command.

Example:

```
+FnRMMG1-WERKS (WERKS = Plant)  
+Fv0001
```

Assigns a fixed value of 0001 to the Plant field for the current transaction.

## +HTHeader Text

Optional command.

This text is displayed as the title when you open your TRANSACTION scripts.

Example:

```
+ HtTRANSACTION for Tcode MM02
```

## +IF!Condition

Optional command.

This command controls the condition for uploading data into SAP. The syntax of the condition parameter is:

Excel column id =, >, <, <=, >=, <>, {Starts\_Width}, {Contains}, {Ends\_With} a Constant value

The left side of the operator above is the Excel column that is to be checked against the constant value on the right side. When the file containing an +IF condition is run, it executes only the Excel rows for which the specified condition is true.

Example:

```
+IF!AH=08/08/02004
```

Uploads only those Excel rows with 08/08/2004 in Excel column AH to SAP.

```
+IF!A{Starts_With} K
```

Processes only those Excel rows that contain data beginning with letter K in Excel column A.

## +LG

Optional command.

This command launches SAP GUI while running the TRANSACTION scripts. TRANSACTION requires SAP GUI to be launched for certain types of transactions, such as file input/output dialog boxes, report generation, uploading files (images, PDF....), many HR transactions, and so on.

The Launch GUI option is available on the Run screen to facilitate the processing of these types of transactions. Marking the Launch GUI checkbox activates the +LG command. Leaving the checkbox blank deactivates the +LG command.

Recommendation: Do not selecting this option, unless absolutely needed, as it may slow down processing.

NOTE: If the following error message is displayed: "Protocol error when communicating with SAP GUI", the Launch GUI flag can be enabled to resolve the issue.

**+MBn**

Optional command.

This command translates characters in Excel from Unicode to native SAP language format such as Shift-JIS. It is useful when entering data in Unicode into non-Unicode SAP systems. In this case, insert the +MB1 command before the +Tx command in the Editor. In Excel, insert the language code in the first cell of each column to be translated.

Example:

```
+MB 1000
```

For example, to upload Material descriptions in Japanese from column A in Excel, in the editor, specify +MB100 and, specify \*DB982, which is the code page for Japanese, in cell A1000.

**+MTMsgID-MsgNo**

Optional command.

The Message Transformation (MT) command customizes log messages stored in the Excel spreadsheet. This command can be used to change a message such as to simplify the message and make the log more meaningful. For example, change the "No batch input data for screen RFEPOS00 0100" to "Record not found".

Translating a certain message requires the proper Message ID and Message Number. Use the +EM command to obtain the Message Type, Message ID, and Message Number in sequence. The Message Type is not required for the MT command.

Example:

This example uses the "No batch input data for screen RFEPOS00 0100" error message. With the +EM command turned on, this expanded message is received:

```
"S:00-344-No batch input data for screen RFEPOS00 0110"
```

Where S is Message Type, 00 is Message ID, 344 is Message Number, and the rest is the actual text returned by SAP.

Then use the +MT command to change the text:

```
+MT00-344=Record Not Found
```

When TRANSACTION encounters Message 00-344, it replaces the SAP text with the text specified above in the +MT command.

**+NE[encrypted password]**

Optional command.

Use the NE (No Edit) command to prompt for the password each time a user attempts to change any mapping information and run the TRANSACTION scripts.

### **+RE**

Optional command.

Use the RE (Run on Errors) command to run the script even when errors appear.

### **+RP**

Optional command.

When used, TRANSACTION helps the user to ensure that the data file (Microsoft Excel or Microsoft Access) for the run has not been used previously.

### **+RS (Global Read SAP option)**

Optional command.

This command sends any transaction field read requests to SAP.

+R S1	Send the Read request to SAP.
+R S0	Do not send the Read request to SAP.

Key Points:

This option is available only for SAP release 4.6 or higher.

If you do not have any field mapped to Read From SAP, TRANSACTION disables this command. If any field is marked for Read From SAP later, this command is enabled, allowing it to be selected and deselected.

See "\*Fv?<Excel receiver column ID>" for more information.

### **+RT**

Optional command.

This command adds a text record to the log file containing the reason for a particular run.

Example:

+RTRequired fix to Material Group as requested by Steve.

### **+SAnnn: StopAfter Transaction nnn**

Optional command.

This command causes TRANSACTION to stop executing transaction calls after nnn transactions.

Example:

+SA100

TRANSACTION runs only 100 transactions.

## +SC

Optional command.

This command controls how a transaction behaves on COMMIT.

+SC1	[Default Value] Stop transaction on COMMIT.
+SC0	Continue transaction on COMMIT.

Recommendation: Run the transaction with the +SC1 (Stop transaction on COMMIT) command on, unless required otherwise.

## +SF

Optional command.

This command specifies a certain character as the Skip Field indicator. When TRANSACTION encounters the Skip Field indicator (character) in an Excel spreadsheet, it does not update that field in SAP.

Example:

Suppose an update is required for the Material Group (among other fields) for all materials, with the exception of a selected few. Use the +SF command in the TRANSACTION Scripts to specify the Skip Field indicator (character):

+SF\

Where \ is our Skip Field indicator.

In addition, modify the Excel spreadsheet by inserting \ in the Material Group column for all rows where an update is not required.

Key Points:

Comparison is not case sensitive. For instance, if +SFbaddata is specified in TRANSACTION scripts and Excel column contains value of BADdata or BADDATA, this field is skipped (the update does not occur).

The Skip Field indicator is global for the TRANSACTION Scripts. Once specified in the TRANSACTION scripts, it applies to all mapped fields. When the value from Excel is read for any field, it is first compared against the Skip Field indicator before it is sent to SAP.

See also Skip Field indicator.

### **+SnScreenNumber**

Required command.

This command specifies the Screen Number and it must follow the +Sp command. The command must always have a corresponding +Sp command.

Example:

+Sn0060

### **+SpScreenProgramName**

Required command.

A transaction is divided into several screen programs, which are called in a pre-defined sequence. Generally, more than one +Sp command is used.

Example:

+SpSAPLMGMM

### **+SRStartFromRow**

Optional command.

When used, the default value of 2 for the Start Row field is changed to the value stated in this command.

Example:

+SR7

First six lines of the selected sheet are ignored and processing begins from the seventh row.

### **+TxSAPR/3TransactionCode**

Required command.

This command specifies the transaction code for the TRANSACTION scripts. There can be only one transaction in the TRANSACTION scripts.

Example:

+TxMM02

## TRANSACTION and the command line

When you run a TRANSACTION script from the Command Line, you can pass any number of TRANSACTION Script commands as parameters. See the *TRANSACTION Script Command Reference* for a complete description of the TRANSACTION Script Command set.

When you enter TRANSACTION Script commands, separate each command using a comma as the delimiter. The order in which you include these commands is not significant. If the TRANSACTION Script command you specified does not exist in the TRANSACTION script you are using, the command is added at run-time.

Example:

```
"C:\Program Files\Winshuttle\Transaction\TxShuttle.exe"  
C:\MyDocuments\TRANSACTION\Sample.Txr,+DB5000,+SR4,+SA3
```

In this example, TRANSACTION runs, opens Sample.txr, reads the file, and then replaces or adds these SHUTTLEScript commands to the file:

+DB5000 adds a delay before logon for 5000 milliseconds (ignores +DB command in the TRANSACTION script if present).

+SR4 starts from row 4 (ignores the Start Row specified in the TRANSACTION script).

+SA3 stops after three transactions (ignores the End Row specified in the TRANSACTION script).

After modifying the TRANSACTION script with the new parameters, TRANSACTION runs the recording.

In addition:

- All TRANSACTION script commands take precedence over the same commands already contained in the TRANSACTION script. TRANSACTION overwrites the existing TRANSACTION script commands in memory with the commands passed from the command line.
- Commands passed as parameters are saved when you save your TRANSACTION script after the run is completed.
- To discard the parameter values, either refresh the file (select Edit, and then Refresh) or open the same file without saving the current one. Once you save the current file, the command line parameters entered overwrite the same commands stored in the TRANSACTION script.
- Avoid passing transaction-related TRANSACTION Script commands (avoid +Sp, +Sn, +Fn, +Fv, \*Fv and ?Fn) as parameters as this might lead to unpredictable results. (These commands usually appear more than once.)

**NOTE:** It is recommended that you first test these commands in a nonproduction system with a limited number of transactions before deploying the TRANSACTION Script.

The TRANSACTION Evaluation version might not support this feature.

Running a TRANSACTION script from the Command Line

You can run a TRANSACTION script from the command line using this syntax:

"TxShuttle.exe full path" <TRANSACTION script name with full path>,  
[TRANSACTIONScriptCommands - Optional]

NOTE: This command requires the "TxShuttle.exe full path" to be enclosed in double quotes. In addition, ensure that there is a single space between "Transaction.exe full path" and TRANSACTION script name with full path.

For example, suppose you want to execute SampleTxR in the default location at C:\MyDocuments\TRANSACTION\. In this case, the command line syntax is:

```
"C:\Program Files\Winshuttle\TRANSACTION\TxShuttle.exe"  
C:\MyDocuments\TRANSACTION\Sample.txr
```

This command launches TRANSACTION with the Sample.txr loaded and starts executing the TRANSACTION script. You can test this command line operation by running this command in the Windows Run dialog box.



## Messages

<b>Cursor field XXXXXX does not exist in the screen</b>	
Description	This message is displayed in Log column of the Excel worksheet after you run the script file.
Reason	This error occurs when TRANSACTION attempts to place the cursor in a field that does not exist on a given screen.
Response	<p>Use this procedure to resolve this problem:</p> <p>On the Mapper, uncheck the Hide system fields (BDC_CURSOR AND BDC_OKCODE) setting to allow the cursor fields to be displayed.</p> <p>Find the BDC_CURSOR field. It is set to the field that is causing the error. You can then either disable that BDC_CURSOR row or set it to another field that is always present.</p>

<b>ERROR_CNTL????? Exception condition "Error" raised</b>	
Description	This message is shown in the Excel Log column or Access Log field after you run the script file.
Reason	The reason for this error is not completely known, but may result from the script file using local system resources.
Response	Use the Launch GUI setting on the Run screen and re-run the script file. (Re-recording is not required).

<b>Evaluate button is disabled</b>	
Description	This message is displayed when you download the Trial version of TRANSACTION and attempt to start the program.
Reason	This error occurs when the Trial version of TRANSACTION is installed on the computer for the second time. Since the software has already been evaluated on that computer, it cannot be evaluated again.
Response	Contact the Winshuttle Support Team at <a href="mailto:support@winshuttle.com">support@winshuttle.com</a> to request an Evaluation extension key for the Trial version.

<b>Function Code cannot be found</b>	
Description	This message is displayed when the Function Code cannot be found.
Reason	<p>This error occurs for a number of reasons:</p> <ul style="list-style-type: none"> <li>• User is running an old script file on a new system.</li> <li>• The transaction in question has changed in SAP.</li> <li>• User is running an old script file on a new system.</li> </ul>
Response	<p>Choose one of these possible solutions:</p> <ul style="list-style-type: none"> <li>• Re-record the script file.</li> <li>• Record and run the script file in the same mode — Batch Input Mode or Non-Batch Input Mode.</li> <li>• If you encounter this error while trying to save a report to a local file, use the Mapper and uncheck the Hide system fields (BDC_CURSOR AND BDC_OKCODE) setting to allow the cursor fields to be displayed. See "Expert Tab: Filter Settings" for more information. Alternatively, you can try replacing =%PC with =&amp;PC or =PC.</li> </ul>

<b>Log error messages in my Excel worksheet are different from the ones I get in SAP</b>	
Description	When running the script file, the error messages logged in the Excel worksheet are different from the those provided by SAP.
Reason	This error occurs when the transaction is recorded and run in Batch Input Mode, which uses a set of messages that differ from the messages used during Non-Batch Input Mode in SAP.
Response	Record the transaction in Non-Batch Input Mode. See "Recording Modes" for more information.

<b>Log error messages in my Excel worksheet are different from the ones I get in SAP</b>	
Description	Log error messages in my Excel worksheet are different from the ones I get in SAP
Reason	This error occurs when the transaction is recorded and run in Batch Input Mode, which uses a set of messages that differ from the messages used during Non-Batch Input Mode in SAP.

Response	Record the transaction in Non-Batch Input Mode. See "Recording mode" for more information.
----------	--

<b>Long text fields are not getting captured</b>	
Description	When recording a transaction to upload long text, the Long Text field is not captured.
Reason	This problem occurs in some transactions where SAP opens a free text editor for any long text to be entered. Information entered in the editor is not captured by TRANSACTION.
Response	Use Batch Input Mode to capture the Long Text fields. See <b>Recording long text</b> in the "Advanced looping capability" for more information.

<b>MC86 - Entering Capacity information - cannot scroll down to the next page of the table</b>	
Description	This message is displayed when there is a SAP program error.
Reason	This is actually a SAP program error, which was corrected in SAP. See SAP Note 117420: "Symptom - The PgUp and PgDn keys don't work in the Planning Table. Additional Key Words - MC93, MC94, MC95, SOP, MC86. Solution".
Response	Change the source code as indicated in SAP Note 117420. Request this modification from your Basis staff. When this change is made, you can then use TRANSACTION to perform this transaction.

<b>ME21N - Cannot find a New Line or Insert Line button or a menu item for multi-line recording</b>	
Description	When recording the ME21N transaction, a method to increment the row cannot be found.
Reason	This error typically occurs because new line functionality is simply not available in the ME21N transaction. In order to record line items in multi-line transactions, there must be a way to capture the row increment. This is generally done by using the New Row button (or alternates).
Response	Filter by line number by requesting line 999 and then always use the first empty line available to enter line item data.

<b>Must have at least one field mapped to an Excel Upload</b>	
Description	This message is displayed when you download data from SAP using the Read From SAP setting.
Reason	This error occurs because you must upload at least one field from the Excel worksheet in order to download data from SAP.
Response	Map one field to be uploaded from the Excel worksheet into SAP to allow the data to be downloaded from SAP. If you want to only download the data from a given table, you can use Winshuttle tableSHUTTLE as an alternative.

<b>No batch input data for screen XXXXXX</b>	
Description	When you run a script file, this error is shown in the Excel Log column or Access Log field.
Reason	This error can occur in TRANSACTION for several reasons: <ul style="list-style-type: none"> <li>• Erroneous recording of the script file.</li> <li>• Encounters an unexpected field during the execution of the script file.</li> <li>• Encounters an unexpected screen that either does not have an OKCode or has an incorrect OKCode</li> </ul>
Response	Choose one of these suggested solutions: <ul style="list-style-type: none"> <li>• Use the Run Step-by-Step (Stop on Errors Only) Run option on the Run screen to determine which field or screen is causing the error.</li> <li>• Re-record the TRANSACTION script with the least possible redundant actions while recording.</li> <li>• Re-record the TRANSACTION script in Non-Batch Input Mode. See "Enter the Recording Properties" for more information.</li> </ul>

<b>No batch input data for screen XXXXXX</b>	
Description	When you run a script file, this error is shown in the Excel Log column or Access Log field.

Reason	For the CO and FI transactions, this error typically occurs because a SAP dialog box is used to enter certain fields only the first time the transaction is executed, and SAP retains this information for subsequent access to the transaction. For example, this issue occurs when a SAP dialog box is used to enter information such as the Controlling Area and Company Code. In this case, TRANSACTION performs correctly on the first entry and then returns this message for all the other entries.
Response	Use the Mapper and set the Start Row, End Row, and IF first transaction setting on the Conditional Run options on the script file to run these transactions. See "Expert Tab: Conditional Run" for more information.

<b>Program terminated by User</b>	
<b>Protocol error when communicating with SAP GUI</b>	
<b>TRANSACTION terminated by User</b>	
Description	When you run a script file, this message is displayed in the Log column of the Excel worksheet or Log field of the Access database. (The transaction might run successfully in troubleshooting mode, but not in the normal run mode.).
Reason	This error occurs when TRANSACTION requires SAP GUI to be launched for certain types of transactions such as file input/output dialog boxes, report generation, and uploading files (images, PDF).
Response	On the Run screen, mark the Launch GUI checkbox to run the transaction.

<b>Read From SAP feature is not working</b>	
Description	This error occurs when you attempt to use the script file to download data using Read From SAP.
Reason	This error occurs when a transaction is recorded in Batch Input Mode and does not perform the intended Read From SAP function.
Response	Record the transaction in Non-Batch Input Mode. See "Recording Modes" for more information.

<b>RFC subsystem not initialized</b>	
Description	This message is displayed when you select the SAP system for logon, get this message, and TRANSACTION aborts
Reason	When connecting to SAP, an unusual SAP logon screen is displayed. This error occurs when the SAPlogon.ini file has a different format as compared to the older version of SAP. (See SAP GUI 6.20 patch 36).
Response	Build a new SAPlogon.ini file with the new version of SAP GUI installed. If you cannot change SAPlogon.ini file, change the system by using the Server tab on the SAP Shuttle Logon dialog box. See "SAP SHUTTLE Logon" for more information.

<b>SAP data type not supported</b>	
Description	This message is displayed when you record or run a script file.
Reason	This error occurs when the SAP function control (wdtfuncs.ocx) version and SAP R/3 version have inconsistencies.
Response	Contact the Winshuttle Support Team at support@winshuttle.com for assistance. Follow the steps outlined by the Winshuttle Support Team and use the Winshuttle-supplied .ocx file.

<b>TRANSACTION behaves differently when run manually than when run through TRANSACTION</b>	
Description	This message is displayed when you run the transaction manually, move back to the first screen, and press Save to complete the transaction. When you use TRANSACTION to record the same transaction again, its behavior is different. The Back Arrow button does not follow the same screen sequence.
Reason	This error occurs when TRANSACTION allows users to record and run transactions in either Batch Input Mode or Non-Batch Input Mode. In Batch Input Mode, the actual screens and the flow of the screens might differ from those shown in Non-Batch Input Mode.
Response	Record the transaction in Non-Batch Input Mode. See Recording Modes for more information.

<b>A 'Fill in all required fields' error message appears in SAP GUI at the start of a recording</b>	
Description	This message is displayed in the SAP GUI when you select the recording properties in TRANSACTION and launch the SAP GUI.
Reason	This error usually happens when the SAP GUI window is not set to Maximum size and has just been changed before the logon.
Response	Open Appotions.xml file from App data folder. Change the values of DEFSIZE to "X" and NOBIEND to " "(Blank)




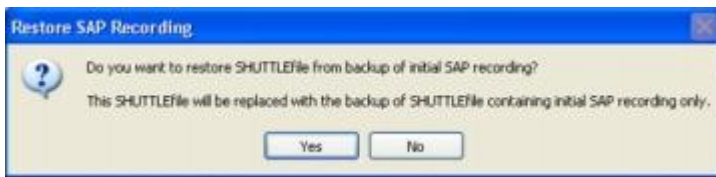


## Troubleshooting

### After mapping, can I go back to my original recording?

Yes, use Restore SAP Recording to roll back the changes to the Mapper to the original recording that was made before mapping.

1. Select Tools from the Main Menu, and then click  (Restore SAP Recording button) to display the confirmation message shown below.



2. Click Yes to restore the backup of the TRANSACTION script containing the initial SAP recording only. This selection is enabled only if the original backup is present in the same folder. (The original backup is created when the TRANSACTION script is initially created.)

You can also click No to return to the Mapper screen.

### Use Transaction to extract from SAP

On the Mapping screen, click Read from SAP. For more information, see the TRANSACTION Advanced Guide.

### Commit code caution

If a script includes more than one commit code, If on SAP field and Skip screen if not found may not work as expected. When you click the Back button to leave the mapper, you are prompted to confirm that you want to continue with the current mapping. You can easily confirm that there is only one commit code before you continue.

### Can I toggle between batch and non-batch modes while running a script?

No.

### Double-clicking in the Mapper is not making the field editable. What can I do?

Click the field and then press F2.



# Index

## A

A1 Dataloader	84
Advanced looping capability	43, 113
Advanced mapping	41
Advanced run settings	73
After mapping, can I go back to my original recording?	119
Application defaults	17
Auto logon	16, 75
Auto mapping	65
Auto update	15

## B

Backing up data and files in Winshuttle	39
---	----

## C

Can I toggle between batch and non-batch modes while running a script?	119
Client Connection Settings	21
Commit code caution	119
Compensating for missing SAP screens and fields	61, 74
Contacting Customer Support	14

## D

Double-clicking in the Mapper is not making the field editable. What can I do?	120
Downloading from SAP	67

## E

Editing a TRANSACTION script	64
Editing in the Mapper	41
Enable GUI scripting	23
Error management and trace files	20

## F

Filtering mapper setting	43
Finding out which build you are using	13

## L

Legal	121
Limiting input fields	53

## M

Mapping Google spreadsheets	54
Mapping long text	63
Messages	111

## N

New features in TRANSACTION version 10.311	
--	--

## P

Preparing to record and recording SAP transactions	31
Production Server Warning (Prod. Svr. Warning)	27
Production versus non-production servers	33
Programs used with TRANSACTION	83
Publishing macros in an Excel workbook	77

## R

Recording long text	37
Recording modes	33, 96
Restore defaults	22
Reviewing transaction history	79
Run options and troubleshooting errors	72
Running multiple or linked files	69
Running scripts	69
Running TRANSACTION at the command line	89

## S

SAP defaults	19
Scheduling runs	70, 96
Setting conditions	54
Setting up for use	15

## T

TRANSACTION Advanced Guide	9
TRANSACTION and the command line	108
TRANSACTION Assistance	13
TRANSACTION Script Command Reference	89
Troubleshooting	119

## U

Undo and Undo All	63
Unicode implementation for TRANSACTION scripts	26
Uploading long text	80
Use Transaction to extract from SAP	119
Using SSO and EP logon	31
Using templates in TRANSACTION	37
Using the Customer Center	14
Using TRANSACTION with Virsa Firefighter	88

## V

Validate by list	59
Validating data	58

## Winshuttle TRANSACTION Advanced User Guide

### W

Windows scripting	81
Winshuttle Community	14
Winshuttle Customer Support	13
Winshuttle DIRECT	83
Winshuttle Function Module for TRANSACTION	28
Writing Headers	76