

# External Data Transfer



**Release 4.6C**



## Copyright

© Copyright 2001 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft<sup>®</sup>, WINDOWS<sup>®</sup>, NT<sup>®</sup>, EXCEL<sup>®</sup>, Word<sup>®</sup>, PowerPoint<sup>®</sup> and SQL Server<sup>®</sup> are registered trademarks of Microsoft Corporation.

IBM<sup>®</sup>, DB2<sup>®</sup>, OS/2<sup>®</sup>, DB2/6000<sup>®</sup>, Parallel Sysplex<sup>®</sup>, MVS/ESA<sup>®</sup>, RS/6000<sup>®</sup>, AIX<sup>®</sup>, S/390<sup>®</sup>, AS/400<sup>®</sup>, OS/390<sup>®</sup>, and OS/400<sup>®</sup> are registered trademarks of IBM Corporation.

ORACLE<sup>®</sup> is a registered trademark of ORACLE Corporation.

INFORMIX<sup>®</sup>-OnLine for SAP and Informix<sup>®</sup> Dynamic Server<sup>™</sup> are registered trademarks of Informix Software Incorporated.

UNIX<sup>®</sup>, X/Open<sup>®</sup>, OSF/1<sup>®</sup>, and Motif<sup>®</sup> are registered trademarks of the Open Group.






HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C<sup>®</sup>, World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA<sup>®</sup> is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT<sup>®</sup> is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, ABAP, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

## Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

## Inhalt

<b>External Data Transfer .....</b>	<b>5</b>
<b>Performing an External Data Transfer .....</b>	<b>6</b>
<b>Parallel Runs.....</b>	<b>11</b>
<b>Starting External Data Transfer .....</b>	<b>12</b>
Menu Paths for Starting the External Data Transfer .....	14
<b>Start with File Splitting .....</b>	<b>15</b>
Menu Paths for Starting with File Splitting .....	17
<b>RFC Call - Example Program.....</b>	<b>18</b>
<b>Evaluating Logs.....</b>	<b>21</b>
Evaluating Logs.....	24
Menu Paths for Evaluating the Logs .....	25
Post Processing Incorrect Data Records.....	26
<b>Using the Test Data Editor .....</b>	<b>27</b>
<b>Menu Paths to the Test Data Editor.....</b>	<b>29</b>

## External Data Transfer

### Purpose

You use this component when you want to transfer data from an old/legacy system into the R/3 system using a direct input procedure.

This function is used in the following components:

- Treasury Loan Management (TR-LO)
- SAP Business Partner
- Executive Information System (EIS)
- Profitability Analysis and Market Segment Analysis (CO-PA)
- Bank Customer Accounts (BCA), Profitability Analysis and Risk Analysis for SAP Banking.
- Industry solution Insurance
- Contract Current Accounts (FI-CA)

### Features

The data is read and converted from a sequential input file, which you have created using data from your old/legacy system. Before being saved in the R/3 datapool, the data is put through several checks. In the case of an error(s), you can rectify the situation using the error log and additional functions.

To assist you with large masses of data, there is a function with which you can split the input file into several smaller files which can then be transferred at the same time.

In several components there are example programs which you can use as templates for your own program. You can then use an RFC to transfer data from a legacy system in "near time".

For testing purposes, you can create an input file using the *test data editor*.

### Constraints

Selecting data in the source system and creating a sequential input file is not done in the R/3 system.

You can only transfer data which have SAP interfaces in R/3. You can find further information under *Transfer Categories* in the IMG of the relevant component. Note that there are no transfer categories for some components (CO-PA, for example).

---

## Performing an External Data Transfer

# Performing an External Data Transfer

## Purpose

This component enables you to transfer data from a legacy or previous/feeder system to the R/3 system and process it in the R/3 system.

## Prerequisites

Before starting external data transfer you must have selected the required data from your source system. Additionally, you must have specified the relevant system settings in Customizing of the respective component.

## Process flow

Data is transferred from the source systems to the R/3 data pool in the following steps:

### 1. Acquisition of the source data:

With the help of data selection programs you create a sequential input file (for example, as Excel file that you have to save in CSV format or as (UNIX) text file) to make the required data available for transfer. The data record structure in this file corresponds to the sender structure you define in Customizing. You have the following options for this:

- You define the **sender structure on the basis of the data pool in your source system**. In this case, the conversion of the sender structure data to receiver structure data takes place by means of transfer rules defined **by you**.
- You define the **sender structure identically to the receiver structure of the R/3 system**. In this case, the conversion of the sender structure data to receiver structure data takes place by means of transfer rules defined **by SAP**. Then the data selection program must supply the data in accordance with the sender structure.

The file has to be available on the presentation or application server. Within Customizing, you need to have entered logical file names for the physical file paths in which the files are stored. (Refer to the Implementation Guide (IMG) of the respective component under *Platform-independent file name assignment*). This helps avoid any later adjustments in connection with the respective operating system.

The data requirements of the R/3 system depend on the transfer categories and other Customizing settings. (Refer to the IMG of the respective component under *Transfer categories*, and in *Treasury* and in the industry solution for banks *SAP Banking* also under *Display required/optional fields*). Ensure that you do not supply two data records (or logical units in the case of header/item structure) on the same subject (for example, transaction) in one input file. For example, if you enter a new loan and change data of this loan soon afterwards, you may not deliver the resultant data records for transfer in the same sequential file. If the same data records are processed in the same block, (see below), runtime errors occur or there is an error entry in the log.

### 2. External data transfer

First of all the system checks the parameters assigned at the start of external data transfer. This involves checking the name of the transfer program (corresponds to the sender structure) and the date details. Then the input file is divided into blocks that are processed one after the other. All blocks first pass through conversion. Once this is

Performing an External Data Transfer

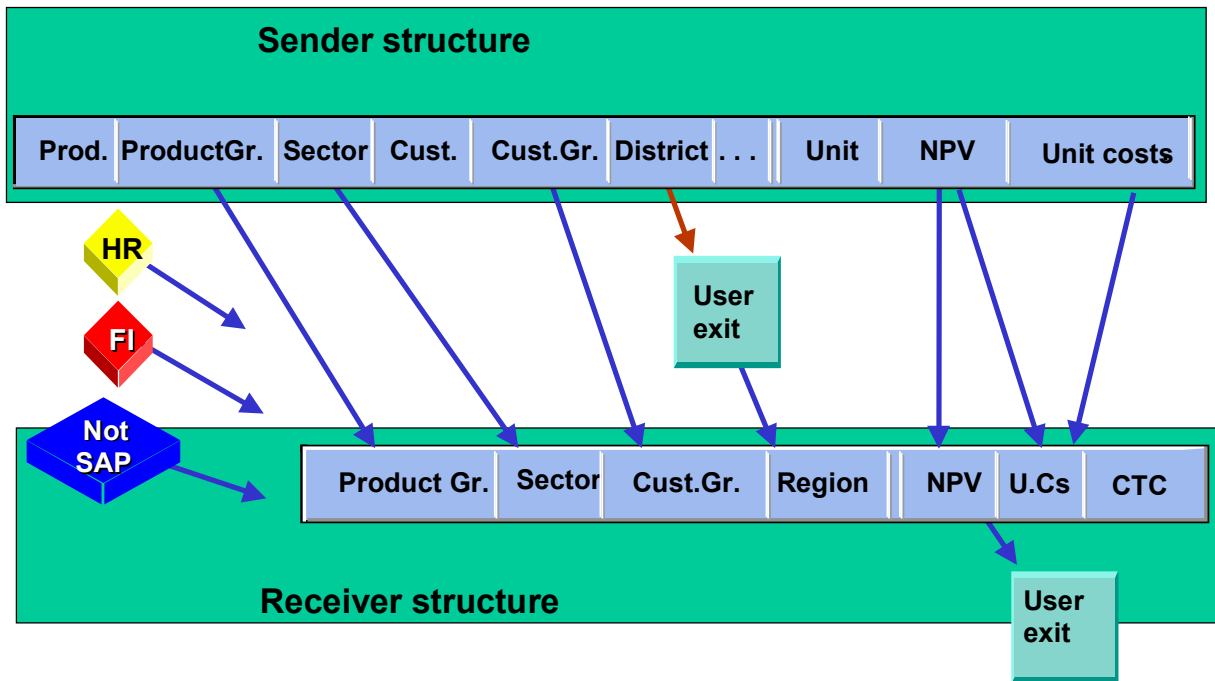
completed without termination or conversion errors, the blocks pass through the function part.

**a. Conversion part**

Every program run starts with the import of the acquired data to an internal table. Then you can trigger a user exit for sender records (see the IMG of the respective component under *Enhancements for external data transfer*). Then conversion of the sender records to receiver records takes place in accordance with the transfer rules and, depending on Customizing, other conversions such as currency conversion. At this point, in some components there can then be a user exit for receiver records.

If an error occurs during conversion, the transfer is terminated. In this case, no data records are written to the database. The error is noted in the conversion log.

An error-free conversion process results in an internal table containing all the receiver records. The table is made available to external data transfer so that the function part can be carried out.



**b. Function part**

In the function part, the system also reads the receiver records in blocks and processes these one after the other.

## Performing an External Data Transfer

First the system checks whether the delivered and converted data records are complete, logical and filled with permissible values. Incorrect data records are extracted and later forwarded to post processing. The system totally rejects a data record for a transfer category consisting of a header/item structure, even if only part of it is incorrect.

Correct data records are completed in an additional step. In this step, for some transfer categories the system enters certain empty fields (such as the transfer date and the time). Then the records are written in the data pool and the error handling routine is triggered.

The incorrect data records are written in an output file. You name this file when you start external data transfer. The program uses this name and adds other data to it. This helps to identify the output files at a later stage. The system also creates a detailed log for incorrect records.

With this, the program run is completed. If there is a program termination during the function part, you can perform a restart. For more information, see [Evaluation of Logs \[Seite 21\]](#): Restart.

You always have the option of repeating the entire external data transfer run. Records which have already been imported are recognized by the system and disregarded.

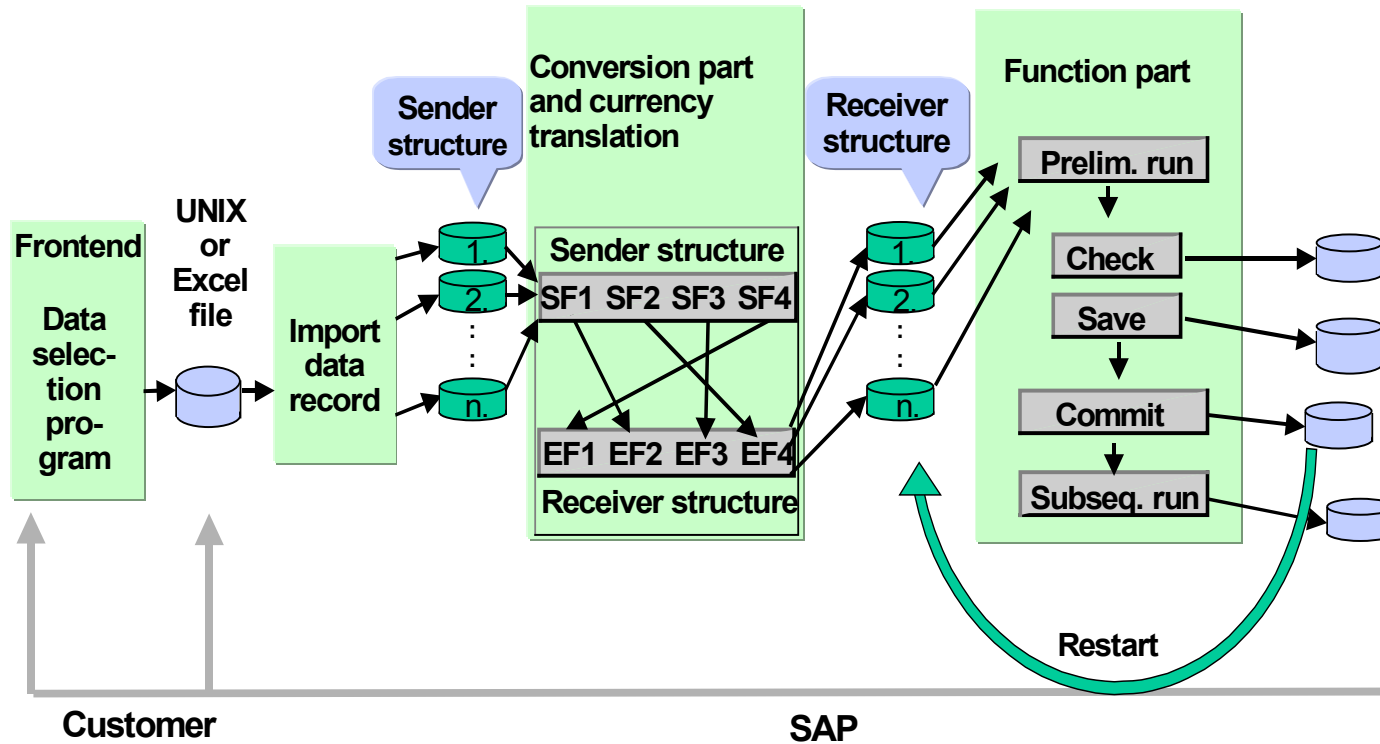
### 3. Logs and error processing

The system creates a log for every transfer - even an error-free one. This serves as basis for the correction of errors. Several functions are available for this. For more information, see [Evaluation of Logs \[Seite 21\]](#).

The following graphic explains the entire process flow of external data transfer from a technical perspective:



Performing an External Data Transfer



**Result**

The data from your legacy / previous systems is now in the R/3 data pool and can be processed in the R/3 system.

**Special features of a release upgrade in some components**

It is possible that a changed receiver structure is delivered with a new release. If a field is added, the system would not enter this field unless it is appropriately adjusted. It could be necessary for you to adjust the sender structure and the sequential input files or the transfer rules to incorporate the additional fields (see the IMG of the respective components under *Transfer categories*, in Treasury and in the industry solution for banks also in the *Required/optional field display* in Customizing).

If a field is removed, it is possible that an existing rule no longer refers to a target field. In this case, a transfer can terminate with an error message. For this reason, before other transfers, we recommend you first perform a test run and check the transfer rules. If you have entered complex rules for a transfer program, it is advisable to print these before a release upgrade.

---

Performing an External Data Transfer

## Parallel Runs

You can have the transfers for several transfer categories run in parallel. Since in cyclical processing every individual process sets up and then deletes locks, you must ensure that the total of all locks does not exceed the value of parameter `enqueue/table_size` (SAP\_PROFILE). Otherwise a lock overflow can occur, in which case the processes cannot be completed correctly. It is possible to determine the number of locks per job per transfer category in include MKCDDFF35 with the variables `con_kopf_*` or `con_commit_intervall_*`. (For transfer categories *Financial object* (001) and *Periods* (002), multiply this value by 3).

## Starting External Data Transfer

# Starting External Data Transfer

## Prerequisites

Before starting external data transfer you must have selected the required data from your source system. Additionally, you must have defined the relevant system settings in Customizing of the respective component.

Ensure that no data record is updated by the external data transfer and simultaneously changed in online mode.

Check the status of the **number assignment**. Once the last number of a number range has been assigned, the system proceeds with the first number of the number range. This can lead to run-time errors and result in an inconsistent data status.



Number assignment is also performed during a test run, meaning that these numbers can no longer be assigned again during an update run.

## Procedure

1. Choose the respective [Menu Path \[Seite 14\]](#). This brings you to the screen *External data transfer: Start transfer*.
2. Enter the name of the *transfer program*, which is identical to the name of the sender structure.
3. Enter the *physical* or *logical* name (including path) of the (sequential) *input file* from which the data is to be imported.

Assign a *physical* or *logical* name for the *output file* (including path), the length of which may not exceed 50 characters. We recommend you use the name of the input file, supplemented by the enhancement "err".

For information on logical file names, see [Platform-Independent File Name Assignment \[Extern\]](#).

4. Specify the *file system* in which the file to be imported is stored.  
If you are importing a UNIX file, choose *Application server*, for an Excel file from a diskette choose *Presentation server*.  
If you wish to transfer packed data (category *P*), you cannot import the file from the presentation server.  
Only files from the application server can be imported in background processing.
5. Select your required options by choosing *Edit*.
6. Decide to which of the listed *file categories* your data in the input file corresponds. When importing UNIX files, use file category *text file*, for Excel files, the *CSV format*.  
Note that in the case of TXT, CSV and MC format the system automatically converts category C fields (non-text fields) into capitals.
7. Choose *Number format* and select the indicator if the length details apply to your data in the input file. Do not use this indicator if you created the file with the test data editor.

## Starting External Data Transfer

Choose a type of decimal representation.

8. Specify the date format (only for TXT, CSV, MC format) of the data.
9. Choosing *Settings for repetitive fields* allows you to define data records that occur more than once.
10. Choosing *Other settings* allows you to choose EBCDIC as data format (as standard, the program assumes the data is in ASCII) and also to make other settings.
11. You need not make an entry in *Check*.
12. To change the block size or the log output, choose *Expert mode*.



Before performing external data transfer, we recommend you save your settings as variant. This will reduce the required effort if you have to repeat the transfer or if you wish to check whether any errors that possibly occur are the result of incorrect settings.

13. Choose *Execute*.

## Result

When external data transfer is carried out online (in the foreground), the system displays a message informing you that the data was successfully transferred or that the transfer process was terminated due to errors. Review the log in all cases, even if the data transfer is reported successful, since the detail log can contain warnings, for example.

When data transfer is carried out in the background, you can check the log to see whether the data was successfully transferred or whether the transfer process was terminated due to errors.

---

**Menu Paths for Starting the External Data Transfer**

## Menu Paths for Starting the External Data Transfer

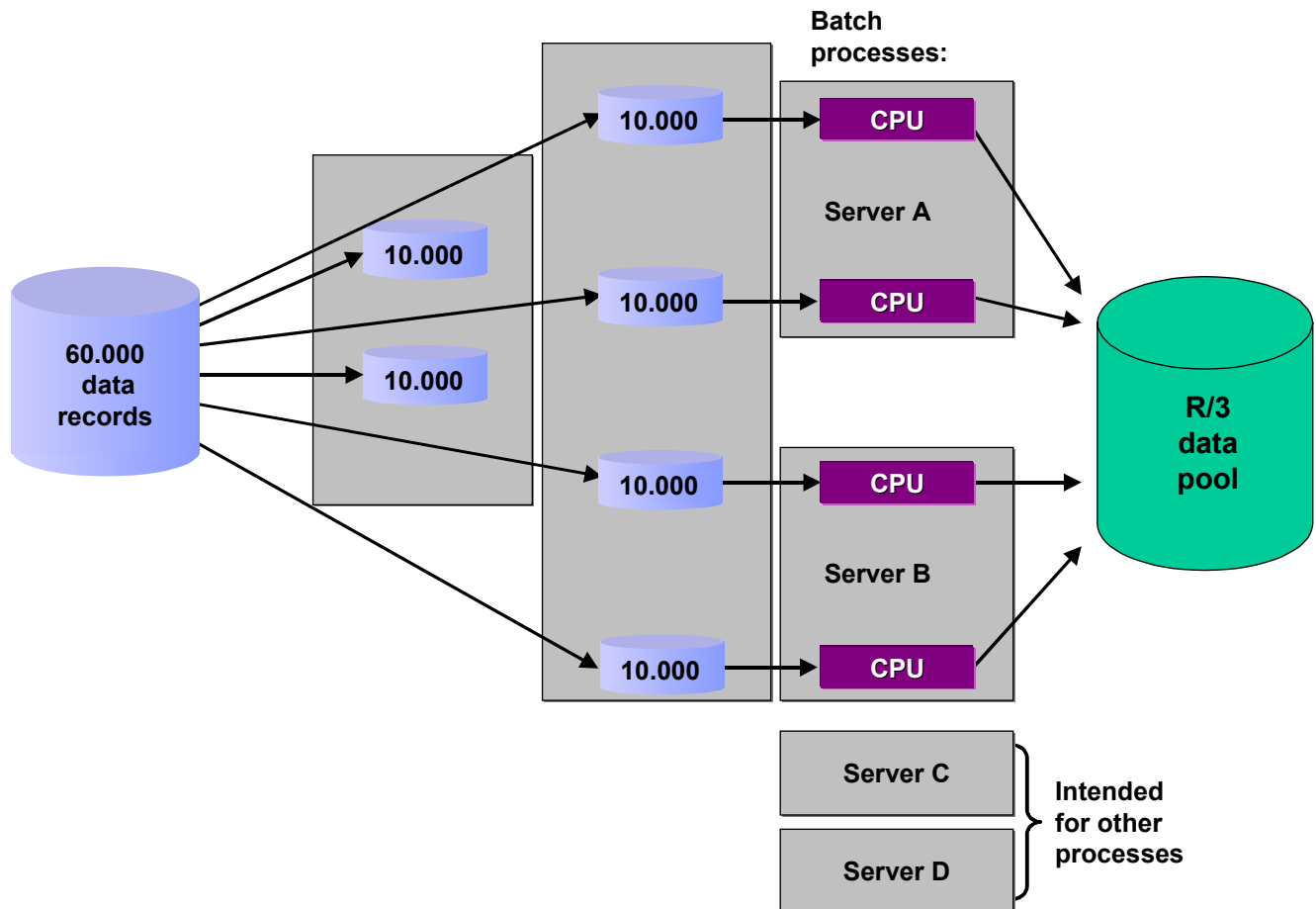
Choose the respective menu option for the component you use:

<b>Component</b>	<b>Menu path</b>
<i>Treasury → Treasury Management → Basic functions</i>	<i>Tools → External data transfer → Execute → Direct input → Start transfer</i>
<i>SAP Business Partner</i>	<i>Application → Direct Input → Start transfer</i>
<i>Executive Information System (EIS)</i>	<i>Transaction data → Transfers → Import file</i>
<i>Profitability Analysis</i>	<i>Tools → External data transfer → Execute</i>
<i>Industry solution SAP Banking</i>	<i>Tools → External data transfer → Start → General</i>

## Start with File Splitting

### Use

To enable processing a large number of data records during external data transfer of a transfer category in a short time, it can be advisable to split the input files into several small files. The program automatically creates the required number of jobs and distributes these evenly among the application servers, which you need to specify. You can then perform the transfer with several of these smaller files in parallel. You can have the jobs automatically released and then deleted by the program.



### Prerequisites

You have defined a *Transfer variant* for external data transfer. For more information, see [Starting External Data Transfer \[Seite 12\]](#).

If you use the function start **with** file split for transfer categories with header and item structures, the fields (which uniquely identify an object) have to be transferred from the sender structure with

## Start with File Splitting

the same names and contents. These field names are listed in the documentation for transfer categories in the implementation guide.

## Procedure

1. Choose the respective [Menu Path \[Seite 17\]](#). This brings you to the screen *External data transfer: Automatic file splitting*.
2. Enter the name of the *transfer program* or the *transfer category*.
3. Enter the *physical* or *logical* name and the *input file* path and the *transfer variant*.
4. In the *Number of jobs* field, enter how many files you want your input file split into.
5. In addition to the job names specified by the program during definition of the jobs, you can also specify a *job name enhancement*. This enables you to make a targeted search for jobs with the enhancement specified by you on the *Selection of jobs* screen (initial screen for the job overview).
6. Specify your *application servers* to which the jobs are assigned.
7. Now you can choose various *functions* that the system performs according to your entries.
  - a. *Split input file*: The input file is split into *<number of jobs>*.
  - b. *Create jobs*: The system defines the jobs. The names FDU\_0001<enhancement> to FDU\_nnnn<enhancement> appear. (nnnn = number of jobs)
  - c. *Release jobs*: This brings you to the screen *Job selection*. Here you can make additional restrictions or optional entries. Then choose *Execute* to release the jobs.
  - d. *Delete jobs*: The system deletes the jobs.
8. To activate the function(s), choose *Execute*.

## Result

The system executes the functions you chose one after the other.



## Menu Paths for Starting with File Splitting

Choose the respective menu option for the component you use:

Component	Menu path
<i>Treasury → Treasury Management → Basic functions</i>	<i>Tools → External data transfer → Execute → Direct input → Start with file splitting</i>
<i>SAP Business Partner</i>	<i>Application → Direct Input → Start with file split</i>
<i>Industry solution SAP Banking</i>	<i>Tools → External data transfer → Start → With file splitting</i>

## RFC Call - Example Program

## RFC Call - Example Program

### Use

An RFC (remote function call) is a function provided by another system which can call up and trigger a function module (an RFC function). You can find further information under [RFC interface \[Extern\]](#) and [RFC in SAP systems \[Extern\]](#).

If you create a new data record in your operative system and want to immediately transfer it to the R/3 system (neartime), you can run a data transfer using an RFC. You can use this example program as a template for an RFC you can write for your operative (source) system.

### Prerequisites

There is an established system connection between the two systems.

You have defined a *variant* for external data transfer in the R/3 system. You can find further information under [start external data transfer \[Seite 12\]](#).

### Features

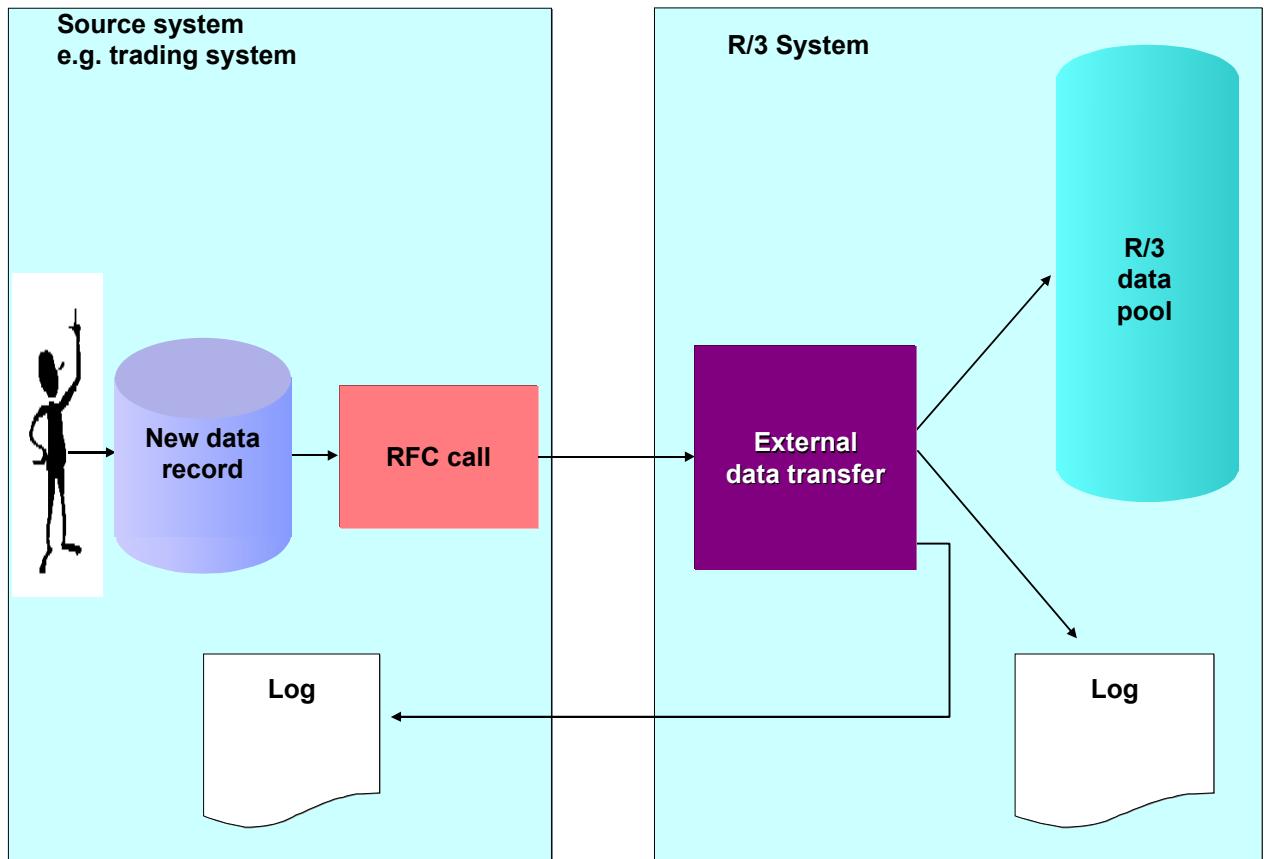
This program describes how you can start an external data transfer from a source system (in this example, an R/3 system named KI4) to another R/3 system (in the example, ALR). The data to be transferred is prepared in the source system in the format of the ABAP Dictionary structure used in the system ALR.

Two function modules have to be called up in the order given in the example:

First, function module FDU\_EXPORT\_GLOBAL\_VARIABLES is run. This will export different parameters to the memory of the R/3 system. These parameters can be set online in the source system at the start of the transfer. In the example program you can enter parameters using PARAMETER statements. You should only use the parameters explained in the example program.

After this, function module EIS\_DATA\_TRANSFER\_RKCT<nnn> is called up (where <nnn> the extension of the ABAP Dictionary structure RKCTnnn). You have to generate this module manually when defining the sender structure in Customizing by choosing *Environment* → *Generate RFC*. The function module has to be given the name of the transfer program, the transfer type CPIC, and a table containing the data to be transferred. The example program can take the messages written in the log for further use in an internal table in the function module.

## RFC Call - Example Program



When evaluating log entries, be aware that the messages in an EDT log can change over the course of several releases. Both the content and the name of a message can change. If messages are used in programs you developed, you should check them after changing releases.

## Activities

1. To be in a position to display the program coding, choose *Tools* → *ABAP workbench* → *ABAP editor*. The screen *ABAP Editor: Initial Screen* appears.
2. In the field *Program*, enter KCFDURFC as the name of the example program.
3. Under *Sub-objects*, you have to mark *Source code*.
4. Choose *Display*. You land on the screen *ABAP: Editor Display Program KCFDURFC*. Now you can look at the coding.

**RFC Call - Example Program**






## Evaluating Logs

### Use

After a data transfer has taken place or been terminated, you can use this function to find and rectify errors.

### Features

The system creates a list of all transfer runs in which you can see the statuses of the transfers. The status indicates whether a transfer was completed, or if there were errors. The following statuses are possible:

Status	Meaning
 Conversion error	The transfer was terminated in the conversion stage. No records were written to the database.
 Canceled	The conversion was successfully completed, but errors occurred in the subsequent processing. The transfer was therefore not completed. It is possible that no records were written to the database. You can find further information concerning confirmed data records in the detail log.
 Error	The transfer was completed, but there were business content errors. All of the data records except the ones containing errors were written to the database.
 Warning	The transfer was completed, but warnings were output to the log. All of the records were written to the database.
 OK	The transfer was completed without errors or warnings. All of the records were written to the database.

Two logs are written for each transfer which represent the two logical parts of the transfer:

#### 1. Conversion Stage:

The **conversion log** is created for every data transfer. The technical aspects of the transfer run are described step-by-step. Errors which prevented the program from running to completion are logged. In the conversion log, you can display all of the sender and receiver records, or just the ones containing errors.

#### 2. Function Stage:

The **detail log** is only created if the conversion is completed successfully. It contains information about the checks and the saving of the data for each record which contains errors. If, during a transfer involving header and item structures, only one item from many is incorrect, the whole block will be listed. This includes the header and all items.

You must always use this log in evaluating whether the transfer occurred successfully, as the messages on the screen and those in the spool directly after the transfer run do not always match those in the log.

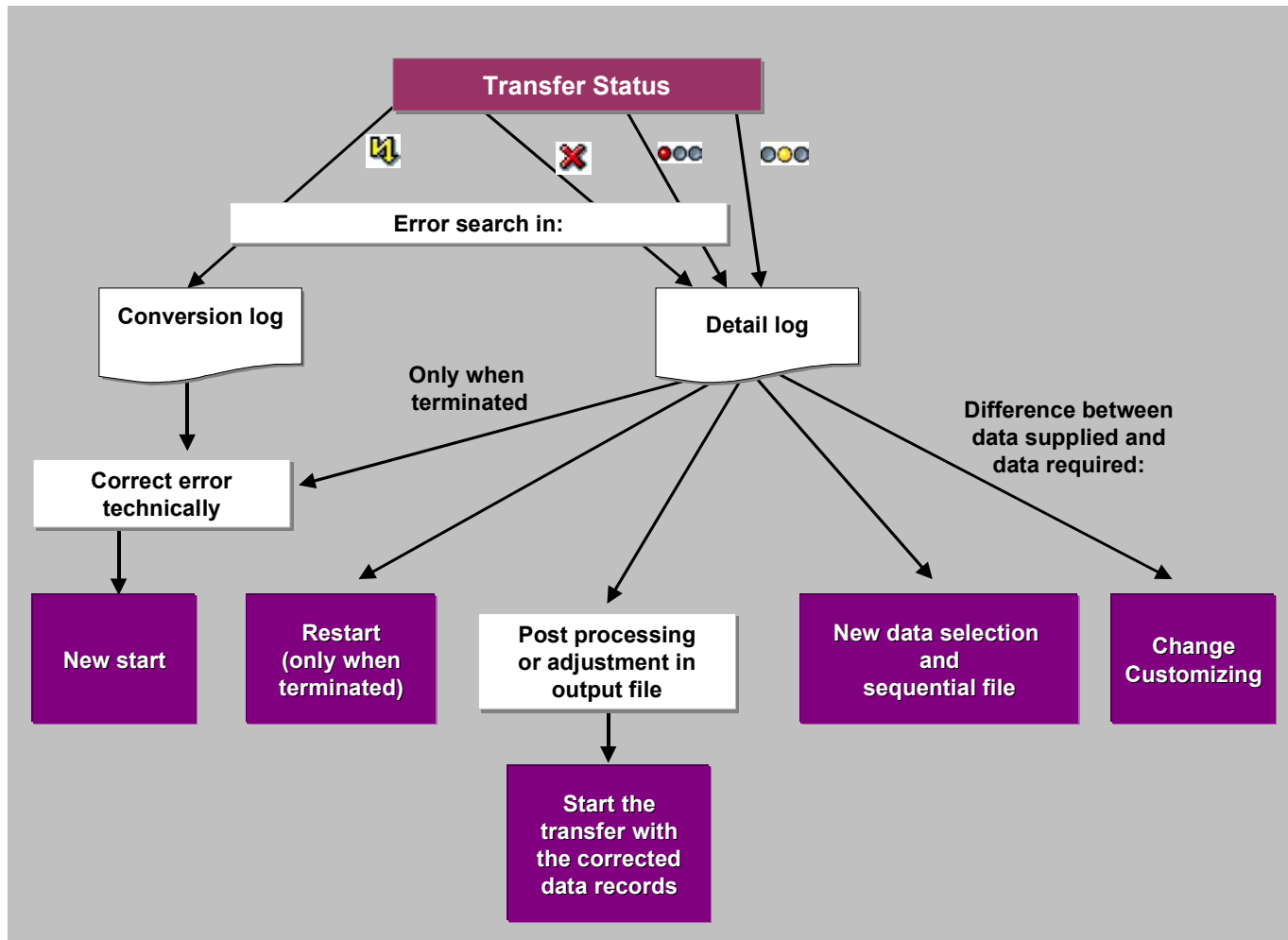
When evaluating log entries, note that the messages in an EDT log can change over the course of several releases. Both the content and the name of a message can change. If messages are used in programs you developed, you should check them after changing releases.

Evaluating Logs

**Error Correction**

Sometimes the scope of the imported data does not match the system’s requirements. In this case, you may have to make certain “conceptual adjustments” before you actually correct any errors. You may have to either:

- modify the system’s data requirements in Customizing (that is, modify the data requirements to more closely match the data in the operative system).
- modify the imported data by changing or enhancing your data selection programs (that is, modify the data to be imported to more closely match target system’s requirements).



**Postprocessing**

For transfers which end with status errors, it is possible for you to directly correct the data in each record with contains errors. Afterwards, you can transfer the corrected data records to the R/3 datapool.

**Output File**

**Evaluating Logs**

You can use the file to make corrections enmasse using a small self-authored program. This would be the best option, for example, if a particular field contains an incorrect value in all of the data records. After running such a program, you can restart the data transfer using the output file with the corrected records as the new input file. The name extensions added by R/3 are listed in the detail log. The extensions are such things as the external number, the client, the date, and an indicator whether the data records were saved in sender or receiver format.

**Restart**

Under certain circumstances, the transfer can terminate in the conversion or function stage. This could occur, for example, if there were an overflow of table spaces in the database, a runtime error, or a power outage.

- If this happens, and no data records actually reached the database (which means that there would be nothing listed in the detail log about confirmed data records), then you can simply repeat the transfer after correcting the error(s).
- If all of the data records have already been transferred, then the database condition is "consistent".
- What can cause problems, however, is when the transfer terminates during the function stage. If this happens, there is no way of immediately knowing how many records actually made it to the database. The program will thus note how many data records are confirmed (successfully committed to the database).

A restart makes it possible to repeat the transfer in such a way that the function stage will only be carried out for those data records which were not confirmed the first (previous) time(s). Subsequently, the database will be contain consistent records. Afterwards, if necessary, you can edit the data records which contain errors again, just as with every other data transfer.

## Evaluating Logs

## Evaluating Logs

### Use

With this function you can display conversion and detail logs, display sender and receiver records, post process and transfer incorrect sender records and perform a restart.

### Procedure







1. Choose the respective [Menu Path \[Seite 25\]](#).

This brings you to the screen *External data transfer: Evaluate logs*.

2. Under *Function*, choose whether you wish to have a list of the transfer runs (choose *Evaluate*) or whether you wish to *delete* logs.
3. In the field *Created by*, enter the user and in the field *Created on* the creation period. Choose whether you wish to have the *test* and *update runs* displayed, *all* runs or only transfer runs that were completed with a certain status. If you do not specify a user, the system displays or deletes all selected transfer runs during the specified period in the current client.
4. Choose *Execute*.

If you chose *Evaluate*, a list of all selected transfer runs appears.

The following functions are available:

Menu description	Other information
 <i>Conversion log</i>	The system displays all messages on the transfer run in a list. You can select the messages according to problem classes by double clicking on the corresponding node on the left of the screen.
 <i>Job log</i>	The system displays the job logs of background transfers.
 <i>Detail log</i>	This brings you to the detail log display. For transfers performed in an update run you can have an error message displayed for every incorrect data record by choosing <i>Expand all</i> .
 <i>Restart</i>	This brings you to the screen <i>External data transfer: Post processing logs</i> . Choose <i>Execute</i> .
 <i>Delete</i>	The system deletes the conversion and detail logs belonging to the selected transfer run.
 <i>Post processing</i>	<a href="#">Post Processing Incorrect Data Records [Seite 26]</a>



Menu Paths for Evaluating the Logs

## Menu Paths for Evaluating the Logs



Choose the respective menu option for the component you use:

<b>Component</b>	<b>Menu path</b>
<i>Profitability Analysis</i>	<i>Tools → External data transfer → Logs</i>
<i>Treasury → Loans</i>	<i>Environment → External data transfer → Logs → Transfer log</i>
<i>Treasury → Treasury Management → Basic functions</i>	<i>Tools → External data transfer → Logs → Transfer log</i>
<i>SAP Business Partner</i>	<i>Application → Direct Input → Logs</i>
<i>Executive Information System (EIS)</i>	<i>Transaction data → Transfers → Transfer logs</i>
<i>Industry solution SAP Banking</i>	<i>Tools → External data transfer → Logs</i>



## Post Processing Incorrect Data Records

# Post Processing Incorrect Data Records

## Procedure

1. In the list of transfer runs, select the transfer you wish to edit and choose  *Post processing*. This brings you to a different screen. The displayed list shows the incorrect data records. You can navigate within a data record in this sequential display by means of the arrow buttons in the application toolbar. The data displayed is then [Offset \[Extern\]](#) differently.
2. Select a data record and choose . This brings you to the screen *Sender records*.
3. Make the necessary corrections.

If you are correcting several records, you can use the arrow buttons in the application toolbar to navigate between the records.

Use the three left-hand columns *F* (front), *M* (middle) and *B* (back) in the table to reassign the lines. This makes correcting the data easier if the sender structure has a lot of fields and you do not need to correct the data in all of them. It is not possible to save these settings, meaning that the next time you call up this screen, the fields will again be displayed in the standard fashion. Navigating among the records has no effect on the settings.
4. You can write single or all data records in your own file by choosing *File* → *Physical*, entering a name for the file, selecting the data records you wish to write in the file and choosing *Write selected*.
5. To transfer the corrected data records, select them and choose .
6. If no errors have occurred during the test run, you can transfer the corrected data records by choosing . It is not possible to only transfer part of the incorrect data records after they have been corrected. It is however possible to delete data records from the list before an update run.

## Result

The corrected data records are now in the R/3 data pool and can be processed in the R/3 system.

## Using the Test Data Editor

### Use



With this function you can create sequential files and fill them with test data that you can use for the purpose of testing external data transfer.

### Prerequisites

You create the test data each for a certain transfer program or in a certain sender structure. You must already have defined these in Customizing.

### Procedure

1. Choose the respective [Menu Path \[Seite 29\]](#). A dialog box appears.
2. Enter the name of an existing *Transfer program* and choose *Continue*.  

This brings you to the screen *Editing sender records*. Initially, the displayed list contains only an empty data record.
3. Editing data records:
  - If you wish to create a new file with test data, choose *File* → *File name* → *Physical*. Enter a name - you can choose this at random. If you choose *Edit*, you then have the option of inserting other records, both new ones or copies. To create the test data, select one or more data records and then choose .
  - If you wish to change an existing file, choose *File* → *File name* → *Physical* or *Logical* and enter the name of the file with the test data you wish to change. Choose *Read file*. All data records are then displayed in the list. You can navigate within a data record in this sequential display by means of the arrow buttons in the application toolbar. The data displayed is then offset differently. For a more detailed display and to make changes, choose .
4. On the next screen, enter the test data in the *Contents* column.



If you are creating or changing several records, you can use the arrow buttons in the application toolbar to navigate among the records.

Use the three left-hand columns *F* (front), *M* (middle) and *B* (back) in the table to reassign the lines. This makes entering the data easier if the sender structure has a lot of fields and you do not wish or need to enter data in all of them. It is not possible to save these settings, meaning that the next time you call up this screen, the fields will again be displayed in the standard fashion. Navigating among the records has no effect on the settings.

5. To save your created or changed data, on the previous screen choose *Write selected*. All selected records are saved.

### Result

The files are then stored on your application server.

Using the Test Data Editor

## Menu Paths to the Test Data Editor

Choose the respective menu option for the component you use:

<b>Component</b>	<b>Menu path</b>
<i>Profitability Analysis</i>	<i>Tools → External data transfer → Test data editor</i>
<i>Treasury → Loans</i>	<i>Environment → External data transfer → Tools → Test data editor</i>
<i>Treasury → Treasury management → Basic functions</i>	<i>Tools → External data transfer → Tools → Test data editor</i>
<i>SAP Business Partner</i>	<i>Application → Direct Input → Test data editor</i>
<i>Industry solution SAP Banking</i>	<i>Tools → External data transfer → Test data editor</i>