SE16XXL

Table of Contents

Distributed by top *flow* GmbH – D-88348 Bad Saulgau.

The information contained in the following pages is provided without warranty of any kind. It is subject to change without prior notice. Any data serve information purposes only.
Neither the author nor the distributor are liable for any errors or omissions of the documentation.

Introduction

SE16XXL is a tool which is intended to make your life – at least your SAP database related user life – much easier. It is a logical extension of the standard Data Browser, a circumstance that makes the transition particularly easy for already versed users. The extended functionality of the tool eliminates the need of writing most ad hoc reports. Scripts, which may also be run in background, make it possible to produce interesting result lists. These can be optionally written out to server files. Less experienced users may take advantage of scripts created by others.
Etc. etc.

Prerequisites

A working knowledge of the Data Dictionary and of the most important database tables, as far as your field of activity is concerned, would do no harm. Neither would a certain familiarity with the standard Data Browser.
For a couple of useful hints please read How to find the “right” database tables.

Compatibility

SE16XXL runs without changes starting from SAP_BASIS version 700, with and without Unicode. In has been implemented under version 700.

Tutorials

A series of **examples** concerning the practical use of SE16XXL is available.
If you are interested, please make use of the following link: Tutorials.

Overview of the major features

In order to exemplify the features listed below, database tables like **MARA** (General Material Data) and **MARC** (Plant Data for Material) will be used.

Database selection

The following kinds of tables can be selected from the database:

Transparent tables
Pool tables
Cluster tables
Table pools
Table Clusters
Database views
Projection views

Select for all entries

Select entries of a second table using values taken from a list of previously selected entries – for example, starting from a list of MARA entries, select matching MARC entries using MATNR values as criteria.

Inner Join

As above, but the result list contains both MARA and MARC entries side by side. MARA entries without corresponding MARC entries are discarded. Up to twenty tables may be joined together in this manner.

Outer Join

As above, with the difference that MARA entries without corresponding MARC entries are retained. This renders this function especially useful for finding data combinations where something is “missing”.

Upload start file

Instead of selecting entries from a database table, a file on the frontend (for example a CSV file) can be uploaded. These data are handled in the same way as entries of a table. Further joins are possible.

Read SAPscript texts directly

Texts like the Sales Texts of materials, or Header/Item Texts of customer orders can be selected directly by making use of pseudo table **$STXL**.

Comparison of up to nine rows

Up to nine rows of the result list may be displayed in detail side by side to better locate differing values, which are highlighted.

Sort result list by up to nine criteria

This standard functionality is needed for the special hide functions listed below. The maximum of nine criteria is dictated by **ALV**.

Hide adjacent duplicates

From a sorted list adjacent duplicates (in regard to the sort criteria) can be hidden in order to leave only the “unique” rows visible.

Hide rows with duplicates

From a sorted list all rows in which the sort criteria occur more than once are hidden. Only rows with uniquely occurring combinations remain visible.

Hide rows without duplicates

Similar to the previous function. Only rows with non-unique criteria combinations remain visible.

Hide rows in which two columns contain the same value etc.

A very useful function to find special data combinations. In addition to the “equal” operator, “not equal”, “greater than” and so on can be used.

Hide rows according to specified criteria

This function behaves like setting a filter in ALV, but can be repeatedly applied and also combined with the other “hide” functions. In ALV in contrast, only **one** filter is active at any given time.

Formula fields

Special columns can be added to the ones selected from the database by defining them by means of a formula. Called “Formula fields”, these columns can be used as sort, filter or join criteria, or simply to display values derived from the other “normal” columns. The language used to draft formulas is a subset of ABAP.

Spreadsheet in place

The result list may be displayed on a spreadsheet embedded into the SAP screen, from which files can be created.

Write result list to server file

The result list can be written to a file on the SAP application server. In this manner it can be put at the disposal of other authorized users.

Detailed display of a row

Each row of the result list can be viewed in detail. This also applies to join lists consisting of up to 20 tables. Since the display is implemented as list, both the “print” and the “find” functions are available.

Save performed operations as a script

The operations performed step by step can be “saved” in form of an SE16XXL script, which may be performed afterwards as the need arises.
A Script Catalog is available, offering functions for listing, copying, renaming, deleting, editing, enhancing, downloading and uploading scripts.

Perform scripts in background

Scripts may also be performed in background, periodically if desired. The results are stored into a buffer that behaves, when imported, as if the list had been created directly. Optionally, the results can be stored into a server file.
A Background Overview is available to keep track of all scripts performed in background.

Special selection screen for scripts

For each created script, a “special selection screen” can be defined, specifically tailored to the peculiar needs of the script. When such a screen is used, script variants can be created, which, similar to report variants, may take advantage of dynamic date variables and TVARVC variables.

Individually tailored list headings

The column headings of the result list, not only of a script, may be redefined to better suit the particular meaning of the list. This redefinition can be carried out for different languages.

Special title lines and title templates

The result list, not only of a script, can be improved by adding up to ten title lines, possibly containing variables that are appropriately substituted.
Predefined title templates can be used for standard title assignments.

Short documentation for scripts

For each script, a short documentation in several languages can be composed.

Hotspots for calling transactions ("Jumps")

For certain columns of the result list, not only of scripts, so-called **jumps** can be defined: when the user clicks on such a column, a predefined transaction is called, possibly according to an additional criterion. For example, using the document category as criterion, a click on the VBELN column might call transaction VA03 for a customer order, VL03N for a delivery and VF03 for an invoice.

Permissions for tables and fields

Access to the database can be restricted in various ways, from authorization groups down to individual table fields and data elements. Special access roles can be defined, which can be associated with SAP authorization roles.

Authorizations for scripts

Script usage and maintenance can also be restricted. Global and user-specific scripts can be created.

Authorizations at record level

It is also possible to take advantage of standard SAP authorization objects, like for example V_VBAK_VKO, to restrict access to certain records of a given table according to specific fields, in this case table VBAK and field VKORG. Each authorization object can be activated separately.

Transaction /TFTO/SE16SCRIPT

A second transaction with reduced capabilities is available to call existing scripts. Advanced functions, such as joins, are not possible when making use of this transaction.

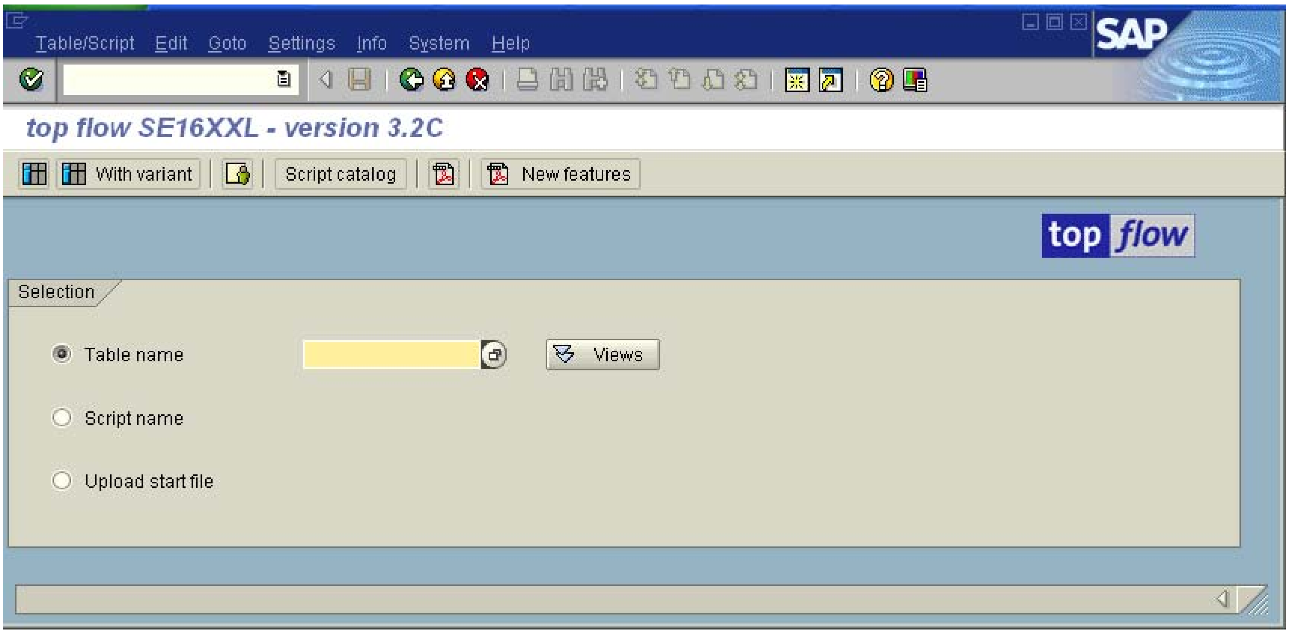
Three result list modes

The result list can be viewed in three different modes, **SE16 standard list**, **ALV list** and **ALV Grid display**. Each mode has its own peculiarities. At any moment the user can switch from one mode to another and back.

The first screen

Make use of transaction **/TFTO/SE16XXL** to call SE16XXL.
Transactions that start with a slash must be entered as **/n/…**, which is quite annoying. Therefore it's advisable to add the transaction to the favorites by using menu function ***Favorites → Insert transaction*** on the SAP Easy Access screen.

The **first screen** of SE16XXL is as follows:



top flow SE16XXL - version 3.2C

Selection

● Table name

○ Script name

○ Upload start file

Three main options are available:

Option	**Description**
Table name	Check this option to select a database table or view. Use the normal F4 help to choose a database table. Press on Views instead to search for views. For more information refer to F4 help for views.
Script name	Check this option in order to perform a script. A parameter for specifying the script will become visible.
Upload start file	Check this option to upload a frontend file.

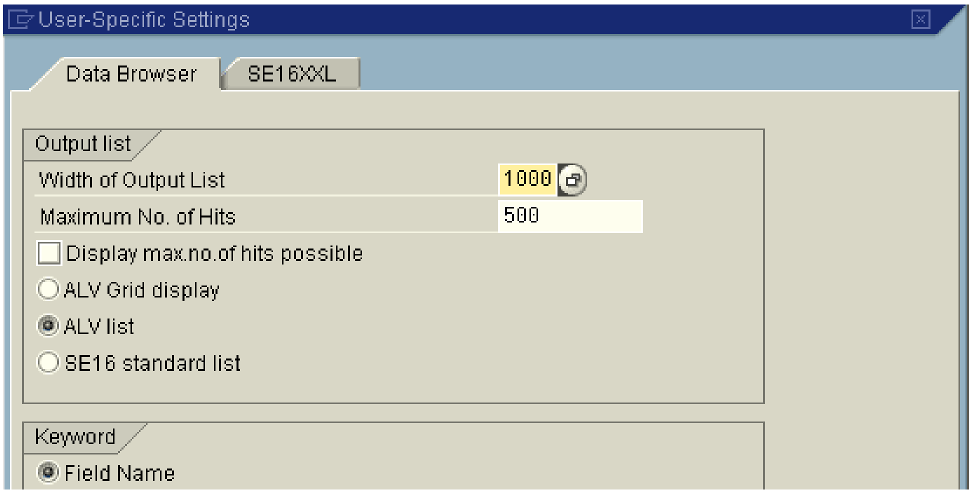
Functions on the application toolbar

Start – depending on the main option chosen, either entries of the specified table are selected, or the specified script is performed, or the file selector for uploading the frontend file shows up.

With variant

Start with variant – a dialog box pops up to let you choose an appropriate table variant or, in case of a script with special selection screen, a script variant.

User parameters – choose between SE16 standard list, ALV list and ALV Grid display. Additional settings, like “Field name” and “Field label” are also possible. These settings influence the layout of the selection screen and the result list.



User-Specific Settings

Data Browser | SE16XXL

Output list

Width of Output List	1000
Maximum No. of Hits	500

□ Display max.no.of hits possible

○ ALV Grid display

● ALV list

○ SE16 standard list

Keyword

● Field Name

Script catalog

Call the **Script Catalog**, which is the starting point for all operations concerning the management of scripts. The Script Catalog doesn’t have a transaction code of its own.

Display the present (or a related) documentation.

New features

See what’s new in the current version of SE16XXL.

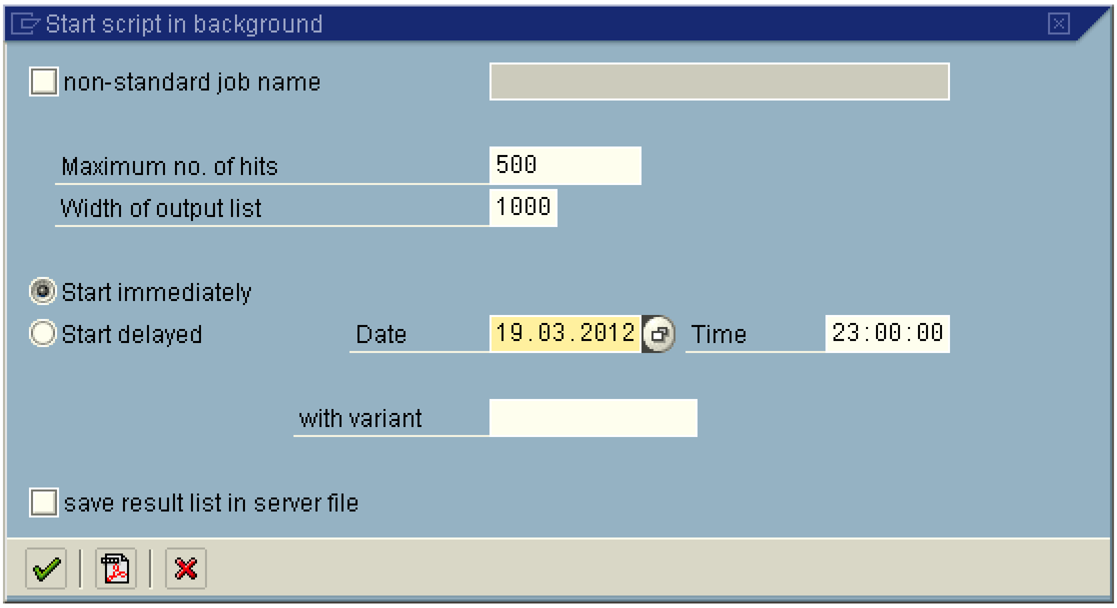
The remaining functions haven’t found their way to the application toolbar and are only available on the menu bar.

Available Menu Functions

Only the functions not present on the application toolbar are listed.

Table/Script → Execute in background

This function is only available for the main option "Script". The specified script is scheduled to be started in background.
A dialog box to this purpose shows up:



Start script in background

□ non-standard job name

Maximum no. of hits: 500

Width of output list: 1000

● Start immediately

○ Start delayed   Date 19.03.2012   Time 23:00:00

with variant

□ save result list in server file

For more information please refer to Performing scripts in background.

Goto → Results of background scripts

Make use of this function to select the result of a script previously performed in background. The results are specific for each user and can only be inspected by the initiator of the background script. Results of background scripts that write their lists to a server file are not visible here.

Goto → Overview of background jobs

Calls the Background Overview, a tool for managing SE16XXL background jobs and requests.

Goto → Table variants …

Calls a tool for downloading, uploading and deleting table variants.

Goto → Script variants …

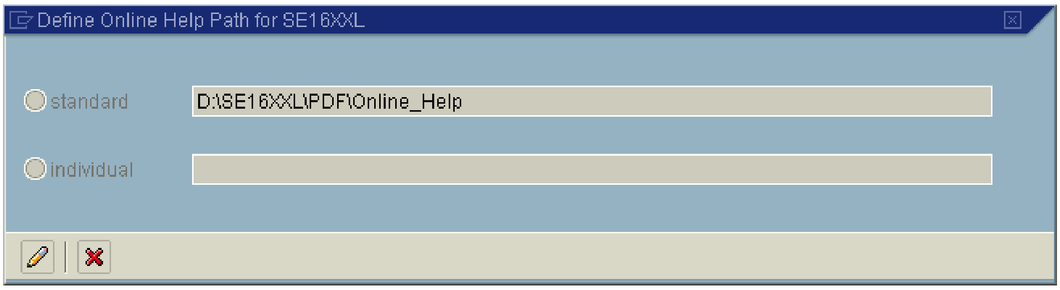
Calls a tool for downloading, uploading and deleting script variants.

Goto → Title template catalog

Calls the Title Template Catalog, a tool for managing Title Templates, which can be used to give a list a more appealing layout.

Settings → Path for documentation

Enables you to define a standard path (if you have administration rights) or an individual path for the PDF documentation files. An individual path is necessary if you cannot reach the standard path. The following dialog box shows up:

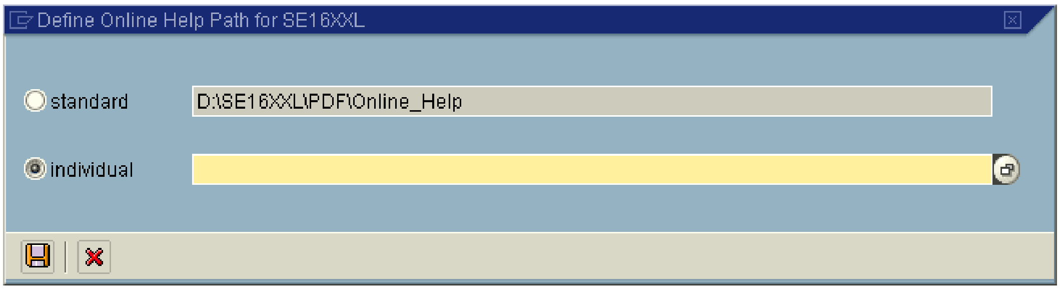


Define Online Help Path for SE16XXL

○ standard D:\SE16XXL\PDF\Online_Help

○ individual

Press the ✎ button to change the settings:



Define Online Help Path for SE16XXL

○ standard D:\SE16XXL\PDF\Online_Help

● individual

Make use of the F4 help to select the right directory. Press 💾 to save.

Settings → Standard title assignments

Here you can assign title templates to specific result list situations. For more information, refer to Standard Title Assignments.

Info → About …

Short information about the (present) program.

Info → Rewards …

Short information about a reward campaign of the company that distributes this program.

The Selection Screen

After pressing ▦ or ▦ With variant on the first screen, a selection screen is output to allow you to specify the selection criteria.

The procedure for the main option “**Upload start file**” is different. In this case, a file selector dialog window shows up. After you have chosen the file to be uploaded, a new screen for the column definition is output. The selection screen comes up after the characteristics of the file contents have been defined. All this is necessary because the structure of a start file is not known a priori as in case of a database table.

In case a script is being performed, which begins by uploading a start file, the file selector window also appears, but no column definition screen, since the structure of the file is already contained in the script itself.

Two kinds of selection screens are in use – a **standard selection screen** and a **special script selection screen**. The latter – special selection screen – is used only for those scripts for which such a screen has been explicitly defined. The standard selection screen is used in all other cases.

The select options of the selection screen vary from one table to another, because each table has different fields. On a standard selection screen the user can choose which criteria (i.e. fields) of the database table should be available on the screen. SE16XXL keeps track of these individual settings.

IMPORTANT: not all fields are suitable as selection criteria. For example, fields of type P with too many digits are not supported.

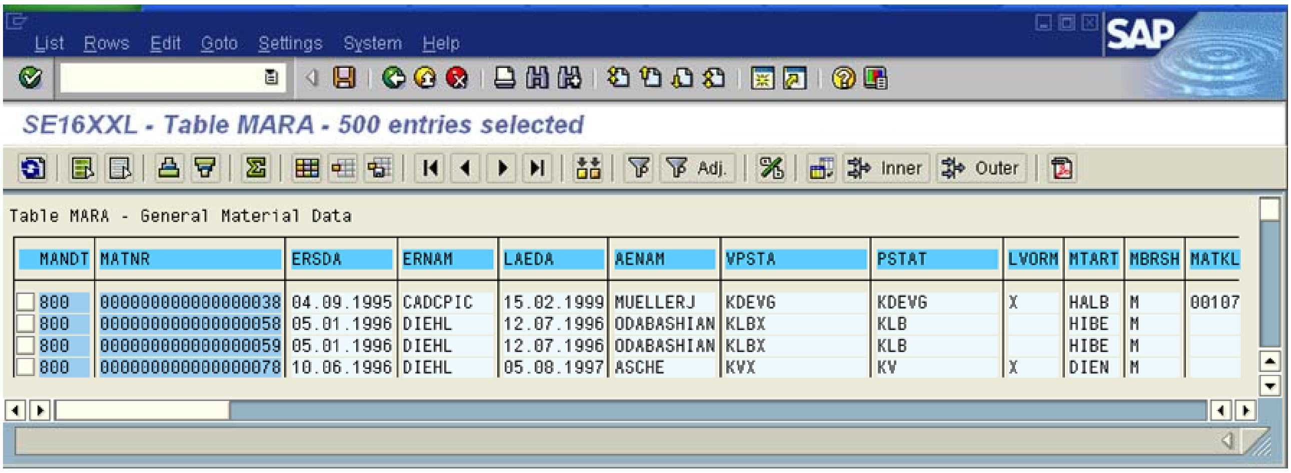
Both selection screens have most functions in common.

For more information regarding the standard selection screen, please refer to Standard Selection Screen.

For more information regarding the special selection screen, please refer to Special Selection Screen.

Result list

The table entries which have been selected are displayed in the result list (example):



SE16XXL - Table MARA - 500 entries selected

Table MARA - General Material Data

	MANDT	MATNR	ERSDA	ERNAM	LAEDA	AENAM	VPSTA	PSTAT	LVORM	MTART	MBRSH	MATKL
□	800	000000000000000038	04.09.1995	CADCPIC	15.02.1999	MUELLERJ	KDEVG	KDEVG	X	HALB	M	00107
□	800	000000000000000058	05.01.1996	DIEHL	12.07.1996	ODABASHIAN	KLBX	KLB		HIBE	M	
□	800	000000000000000059	05.01.1996	DIEHL	12.07.1996	ODABASHIAN	KLBX	KLB		HIBE	M	
□	800	000000000000000078	10.06.1996	DIEHL	05.08.1997	ASCHE	KVX	KV	X	DIEN	M	

Three modes are available, to be chosen by pressing the button on the first screen or on the selection screen, or using menu function ***Settings → User parameters*** :

- SE16 Standard List
- ALV List
- ALV Grid Display

The main features of the three modes are listed below:

SE16 Standard List

This is the simplest mode. The order in which the columns are displayed is that of the fields in the Data Dictionary and cannot be changed. The width of each column is also fixed. The maximum number of columns that can be output together is 256. The maximum width of the list is 1023. No fancy features like totals and subtotals are available. The main advantage is that the list is produced quite fast.

ALV List

This is the ABAP List Viewer – a standard SAP list function. Lots of fancy features are available, like totals, subtotals, groups of rows separated by lines, layouts etc. The maximum number of columns that can be output together is 90. The maximum width of the list is 1020. The main disadvantage of this mode is that, especially for large lists, the output can take quite a while to be produced.

ALV Grid Display

This is the **A**BAP **L**ist **V**iewer, but based on the **Grid Control**. Most features of ALV List are also available, but the output is not a normal list but a control.
The maximum number of columns that can be output together is 256. There is no restriction regarding the maximum width.
It seems very fast in regard to the output, but this is achieved by displaying only the first part of the list. The rest is output when the user scrolls the list downwards.

The main features are summarized in the table below:

Mode	Column Order	Output	Layouts	Max. Columns	Max Width	Fancy Features
SE16 Standard	Fixed	Fast	No	256	1023	No
ALV List	Variable	Slow	Yes	90	1020	Yes
ALV Grid	Variable	Fast	Yes	256	-	Yes

For examples regarding ALV please refer to Tutorial # 14.

IMPORTANT: as may be seen from the above table, ALV List can display at most 90 columns, whereas the other two modes can output up to 256 columns. In case of SE16 Standard List the real limit is usually the maximum width of 1023, which means that all columns that exceed this limit are not shown. When the mode is switched to ALV List, only the first 90 columns are kept for the output (if they don’t exceed the maximum width of 1020).

Most users tend to choose one mode, for example ALV Grid Display, and stick to it. But sometimes it makes sense to switch to another mode to take advantage of the peculiar features of that mode. Especially the dialog boxes used to choose the list columns differ widely from one mode to another.

In the following pages we will discuss the main functions of the result list. Most functions are available for all three modes and work in the same way. Possible exceptions will be pointed out. Standard functions that are not specific to SE16XXL, such as ***List* → *Send to* → *Mail recipient*,** will not be discussed.

Functions on the application toolbar

Refresh.

This function works differently depending on the situation:

1) If the list is the result of a main selection, i.e. a selection initiated from the selection screen, the selection is carried out once more. In case of a start file, the original records are displayed again.

2) If the list is the result of a “Select for all entries” operation, this last operation is performed once more. The original data used as starting point for the operation are not refreshed.

3) If the list is the result of an “Inner join” or “Outer join”, the join selection is carried out once again. The original data used as starting point for the join operation are not refreshed.

4) If the list is the result of a script, the script is performed once again.

5) If the list is the result of a script performed in background, the result is once again read from the corresponding buffer.

Even if the same data are displayed again, as in case of a start file, it may still make sense to use this function. For example if a part of the rows have been discarded using menu function ***Rows →Discard hidden rows*** after a filter was set.

This function is not available if a formula with global scope is active.

Display row in detail – equivalent to a **double click on a row**.

This function is only present (as button) in SE16 Standard mode.

Select all rows. It is normally not necessary to select rows in order to perform a given function. Only the function and the menu function ***Rows →Hide selected rows*** really need selected rows.

Deselect all rows.

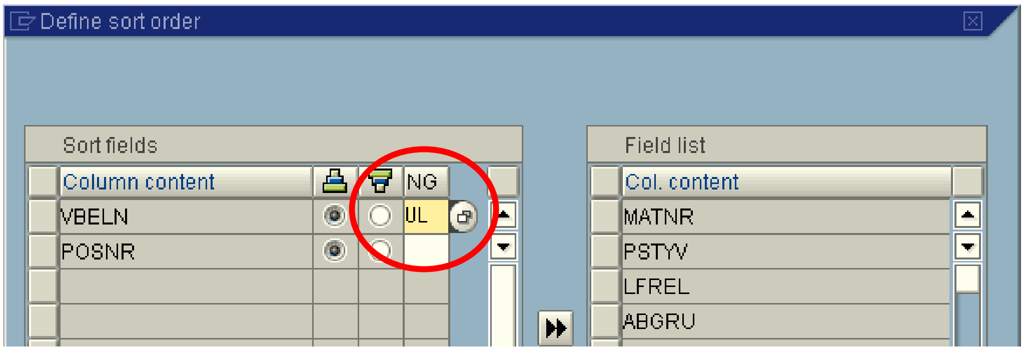
Sort in ascending or descending order.

If columns have been selected before pressing the button, the sort is carried out right away using the marked columns as criteria. Otherwise a dialog box for specifying the sort criteria is output.
A maximum of **nine** sort criteria may be defined. This number is hard coded into ALV and cannot be increased in any way. For compatibility reasons SE16 standard mode also allows a maximum of nine criteria.

In SE16 standard list mode it is possible to add a new sort criterion to the others already defined by setting the cursor on a column (not the column heading) and pressing one of the two buttons.

NOTE: the sort order of text fields obtained using ALV may differ slightly from the one obtained in SE16 Standard mode. When a script is being performed, the SE16 Standard mode sort is used internally.

In **ALV list** it is possible to separate groups of rows with a line by entering an '**UL**' (underline) into the appropriate field:



Define sort order

Sort fields

Column content			NG
VBELN	●	○	UL
POSNR	●	○	

Field list

Col. content
MATNR
PSTYV
LFREL
ABGRU

The resulting list looks like the following:

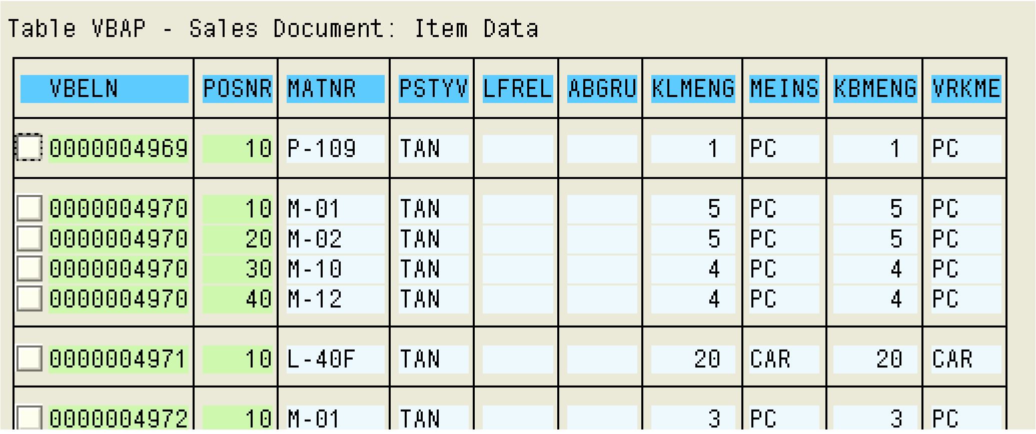


Table VBAP - Sales Document: Item Data

VBELN	POSNR	MATNR	PSTYV	LFREL	ABGRU	KLMENG	MEINS	KBMENG	VRKME
0000004969	10	P-109	TAN			1	PC	1	PC
0000004970	10	M-01	TAN			5	PC	5	PC
0000004970	20	M-02	TAN			5	PC	5	PC
0000004970	30	M-10	TAN			4	PC	4	PC
0000004970	40	M-12	TAN			4	PC	4	PC
0000004971	10	L-40F	TAN			20	CAR	20	CAR
0000004972	10	M-01	TAN			3	PC	3	PC

Compute totals of selected columns. Only columns containing numerical values (and NUMC) can be totaled.

This function is only available in ALV mode.

Compute subtotals – this button is only available if totals have been calculated. In order to obtain subtotals it is necessary to sort the list in some way.

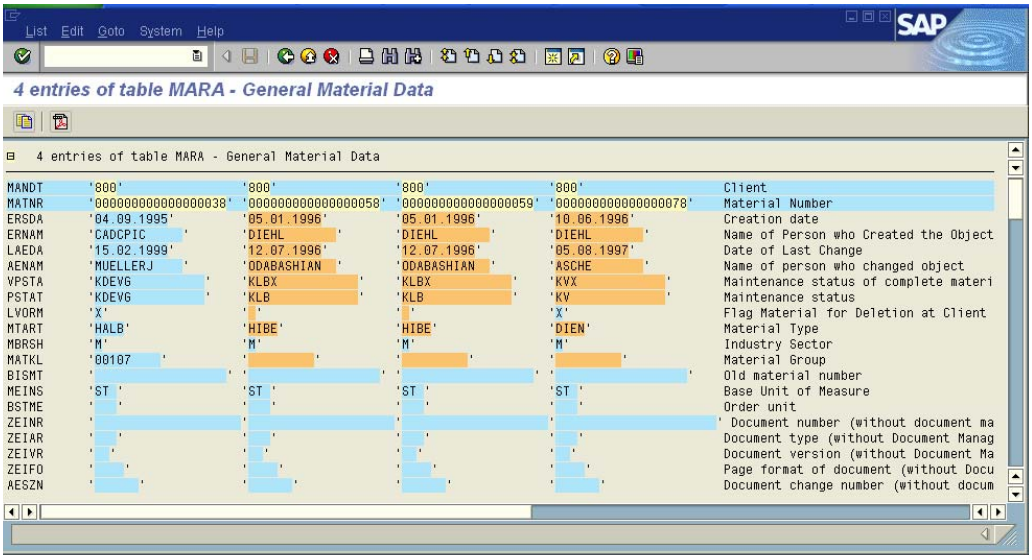
Choose columns for the output list.
Depending on the mode, different dialog boxes are used.
In ALV List and ALV Grid Display it is also possible to specify the order in which the columns appear on the list.

Select a layout – only available in ALV.

Save layout – only available in ALV.

Buttons for scrolling the list horizontally.
Not available in ALV Grid Display.

Compare up to nine rows with each other. The rows in question must be selected. The resulting list is similar to the following:



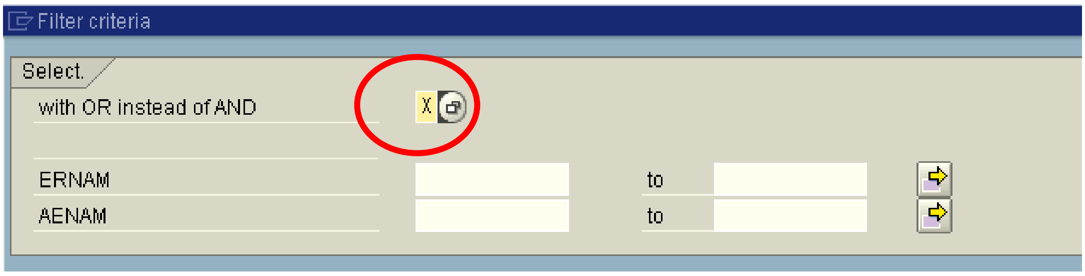
4 entries of table MARA - General Material Data

```
⊟   4 entries of table MARA - General Material Data

MANDT     '800'                 '800'                 '800'                 '800'                 Client
MATNR     '000000000000000038'  '000000000000000058'  '000000000000000059'  '000000000000000078'  Material Number
ERSDA     '04.09.1995'          '05.01.1996'          '05.01.1996'          '10.06.1996'          Creation date
ERNAM     'CADCPIC     '        'DIEHL       '        'DIEHL       '        'DIEHL       '        Name of Person who Created the Object
LAEDA     '15.02.1999'          '12.07.1996'          '12.07.1996'          '05.08.1997'          Date of Last Change
AENAM     'MUELLERJ    '        'ODABASHIAN  '        'ODABASHIAN  '        'ASCHE       '        Name of person who changed object
VPSTA     'KDEVG           '    'KLBX            '    'KLBX            '    'KVX             '    Maintenance status of complete materi
PSTAT     'KDEVG           '    'KLB             '    'KLB             '    'KV              '    Maintenance status
LVORM     'X'                   ' '                   ' '                   'X'                   Flag Material for Deletion at Client
MTART     'HALB'                'HIBE'                'HIBE'                'DIEN'                Material Type
MBRSH     'M'                   'M'                   'M'                   'M'                   Industry Sector
MATKL     '00107    '           '         '           '         '           '         '           Material Group
BISMT     '                  '  '                  '  '                  '  '                  '  Old material number
MEINS     'ST '                 'ST '                 'ST '                 'ST '                 Base Unit of Measure
BSTME     '   '                 '   '                 '   '                 '   '                 Order unit
ZEINR     '                    '                    '                    '                    ' Document number (without document ma
ZEIAR     '   '                 '   '                 '   '                 '   '                 Document type (without Document Manag
ZEIVR     '  '                  '  '                  '  '                  '  '                  Document version (without Document Ma
ZEIFO     '    '                '    '                '    '                '    '                Page format of document (without Docu
AESZN     '      '              '      '              '      '              '      '              Document change number (without docum
```

For more information on this topic please refer to Comparing Rows.

Set a filter.

This function is similar to the standard filter of ALV, but behaves differently. In ALV there is only one filter active at any given time. If the function is called a second time, the criteria defined on the first call will still be present on the dialog box. The previous filter is just updated.
The SE16XXL filter works differently (even in ALV): each filter is applied to the result of the preceding one. In this way it is possible to filter the list rows in stages, something that offers greater flexibility. Furthermore, if more than one criterion is specified, a special parameter "**with OR instead of AND**" becomes visible
(this parameter is not a checkbox due to technical limitations):



Filter criteria

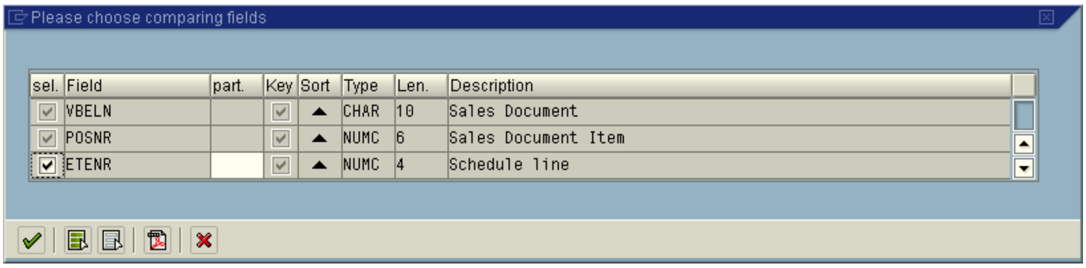
Select.

with OR instead of AND	X		
ERNAM		to	
AENAM		to	

Adj.

Hide adjacent duplicates.

This function requires the list to be sorted.
A dialog box with the sort criteria shows up:



Please choose comparing fields

sel.	Field	part.	Key	Sort	Type	Len.	Description
☑	VBELN		☑	▲	CHAR	10	Sales Document
☑	POSNR		☑	▲	NUMC	6	Sales Document Item
☑	ETENR		☑	▲	NUMC	4	Schedule line

Depending on which criteria should be used to identify the duplicates, it is possible to deselect in succession the last active criterion. Obviously, at least one criterion must be left selected.
As a result, only the **first row** of each group of "duplicates" will be left on the list. The others will be **hidden**. This function is a kind of filter.

For more information please refer to Choosing Comparing Fields.

Show again hidden rows. Is only available if hidden rows are present.

Any rows hidden by one of the following operations are displayed again:

1) Set filter
2) Hide adjacent duplicates
3) ***Rows → Hide selected rows***
4) ***Rows → Hide rows without duplicates***
5) ***Rows → Hide rows with duplicates***
6) ***Rows → Hide rows where field1 op field2***

Even if the rows were hidden in stages, the present function shows all hidden rows again, regardless of the way they were filtered.

Edit formula.

Calls the formula editor to define a new formula or change an existing one. A formula is a “small” amount of ABAP coding (using a subset of ABAP) that allows the user to add new fields (called formula fields) to the result list. The formula is called for each row of the result list, in the same order as the list is sorted.

Formulas usually have a **local scope**, i.e. they only “see” the row for which they have been called. But it is also possible to define formulas with a **global scope**, which, by means of STATICS declarations, remember the contents of the previously processed rows.

At any one time, only **one** formula can be active. Normally a formula remains active only as long as the current list is active. When a join is performed, it is possible to specify that the currently active formula be retained.

For more information please refer to SE16XXL Formulas.

	Select for all entries
Inner	Inner join
Outer	Outer Join

These three functions behave and perform similarly and will therefore be discussed together.

They are based on all rows of the current list (hidden rows are not considered) or, if some rows are selected, on the selected ones.

A new database table (or view) is selected. As selection criteria, values taken from chosen columns of the current result list are used.

In order to exemplify the procedure, we will start from a list of entries of table **KNVV** (Customer Master Sales Data). As new table we will make use of **KNVP** (Customer Master Partner Functions). These two tables have been chosen because they have many key fields.

Before performing the operation, it is advisable to mark the columns that are to be used as selection criteria. This is done by clicking on the corresponding column headings:

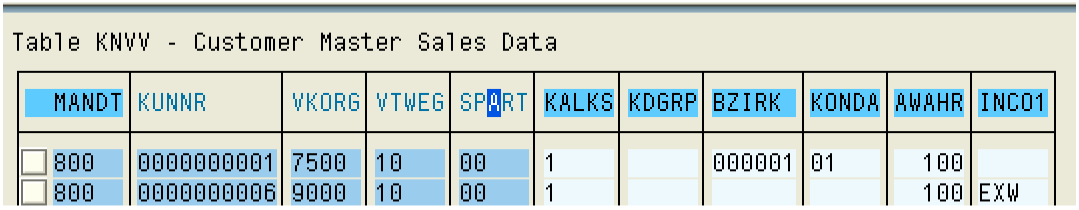
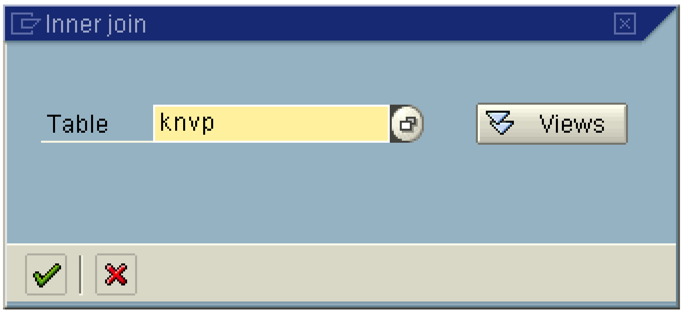


Table KNVV - Customer Master Sales Data

	MANDT	KUNNR	VKORG	VTWEG	SPART	KALKS	KDGRP	BZIRK	KONDA	AWAHR	INCO1
☐	800	0000000001	7500	10	00	1		000001	01	100	
☐	800	0000000006	9000	10	00	1				100	EXW

The headings change color to indicate that they have been marked.

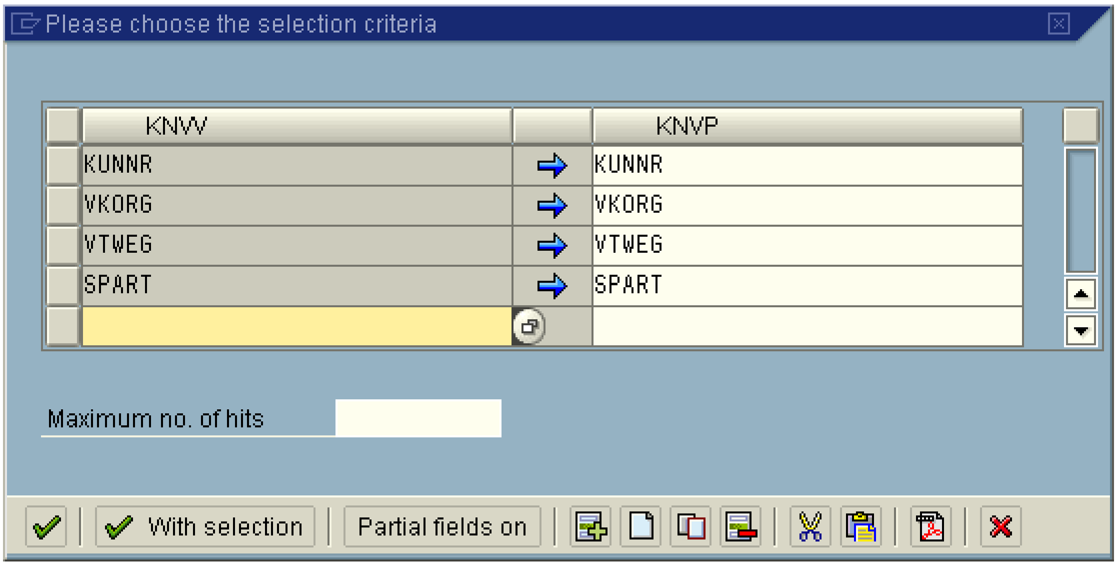
The first dialog box to show up after the function button has been pressed is for specifying the new table to be selected. We enter KNVP:



Inner join

Table: knvp

We acknowledge by pressing ✓ to proceed. The next dialog box is for specifying the join criteria. Since we have marked the relevant columns, the criteria are already present on the screen:



KNVV		KNVP
KUNNR	⇨	KUNNR
VKORG	⇨	VKORG
VTWEG	⇨	VTWEG
SPART	⇨	SPART

Maximum no. of hits

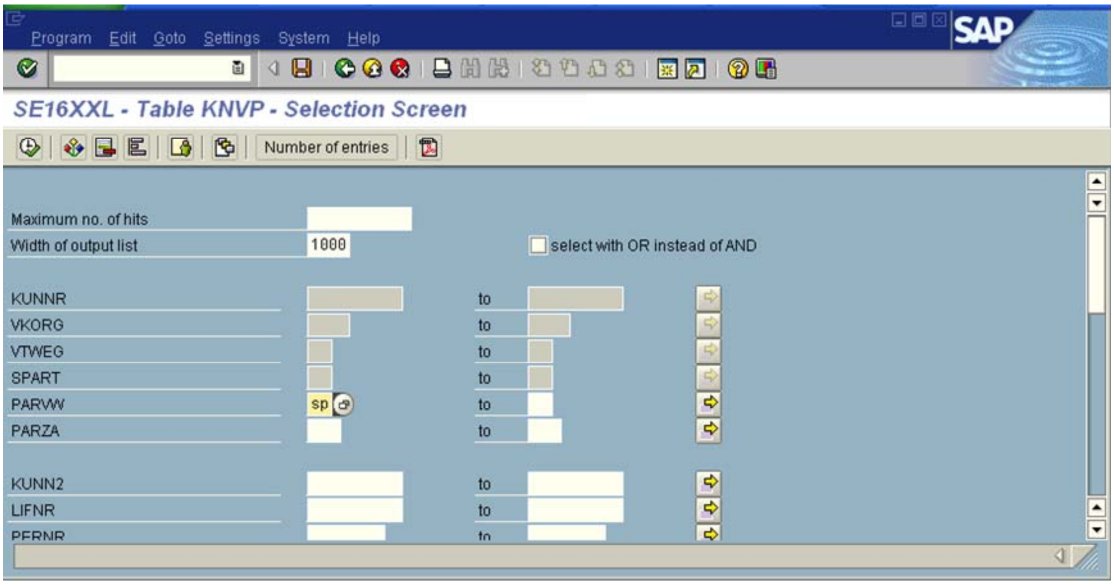
On the left side are the columns we have marked. On the right side one can find the corresponding fields of the new table, in this case KNVP. SE16XXL tries to match the criteria with fields of the new table by using a series of rules which usually produce a good result. But it may happen that no matching field is found, or not the one you expected. In such cases the “right” field has to be entered manually. For more information please refer to Join Criteria. Also take a look at the Tutorials.

At this point there are two ways to proceed:

1) By pressing ✓ to get the result list right away.

2) By pressing ✓ With selection to obtain an „intermediate“ selection screen for specifying additional criteria.

In our example we press the latter in order to show how such an intermediate selection screen might look like.



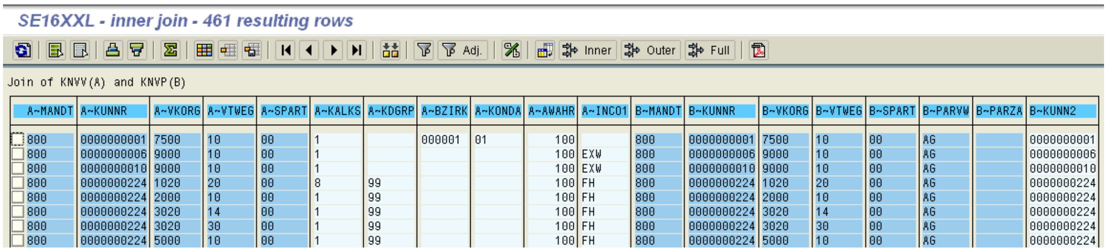
SE16XXL - Table KNVP - Selection Screen

Maximum no. of hits			
Width of output list	1000		□ select with OR instead of AND
KUNNR		to	
VKORG		to	
VTWEG		to	
SPART		to	
PARVW	sp	to	
PARZA		to	
KUNN2		to	
LIFNR		to	
PERNR		to	

Notice that the join criteria are protected.

As additional criterion we enter “**SP**” (Sold-to-party) into select option PARVW (Partner function).

After acknowledging by pressing on the toolbar, we finally obtain the result list, in our example the result of an „Inner join“:



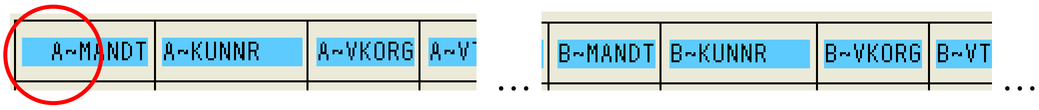
SE16XXL - inner join - 461 resulting rows

Join of KNVV(A) and KNVP(B)

A~MANDT	A~KUNNR	A~VKORG	A~VTWEG	A~SPART	A~KALKS	A~KDGRP	A~BZIRK	A~KONDA	A~AWAHR	A~INCO1	B~MANDT	B~KUNNR	B~VKORG	B~VTWEG	B~SPART	B~PARVW	B~PARZA	B~KUNN2
□ 800	0000000001	7500	10	00	1		000001	01	100		800	0000000001	7500	10	00	AG		0000000001
□ 800	0000000006	9000	10	00	1				100	EXW	800	0000000006	9000	10	00	AG		0000000006
□ 800	0000000010	9000	10	00	1				100	EXW	800	0000000010	9000	10	00	AG		0000000010
□ 800	0000000224	1020	20	00	8	99			100	FH	800	0000000224	1020	20	00	AG		0000000224
□ 800	0000000224	2000	10	00	1	99			100	FH	800	0000000224	2000	10	00	AG		0000000224
□ 800	0000000224	3020	14	00	1	99			100	FH	800	0000000224	3020	14	00	AG		0000000224
□ 800	0000000224	3020	30	00	1	99			100	FH	800	0000000224	3020	30	00	AG		0000000224
□ 800	0000000224	5000	10	00	1	99			100	FH	800	0000000224	5000	10	00	AG		0000000224

Each row now consists of an entry of table KNVV and one of table KNVP. The two entries have the same values for the fields used as join criteria.

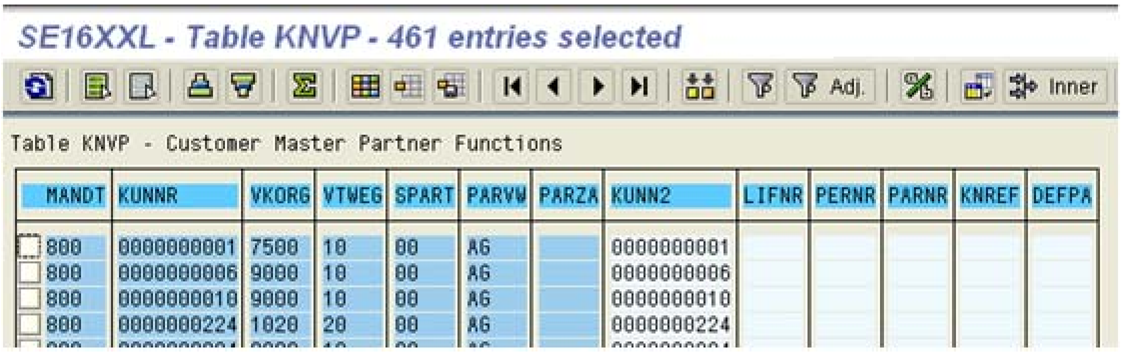
Notice that an alias has been added to the headings in order to keep the columns of the two tables apart:



At this point you may be asking yourself, “What’s the difference between ‘**Select for all entries**’, ‘**Inner join**’ and ‘**Outer join**’ ?”

We'll try to answer by means of our example.

With '**Select for all entries**' table KNVP is selected using values taken from the rows of KNVV, in our example KUNNR, VKORG, VTWEG and SPART. Each row of the original table furnishes a group of values which is used to select matching KNVP entries. The result is a list of KNVP entries:



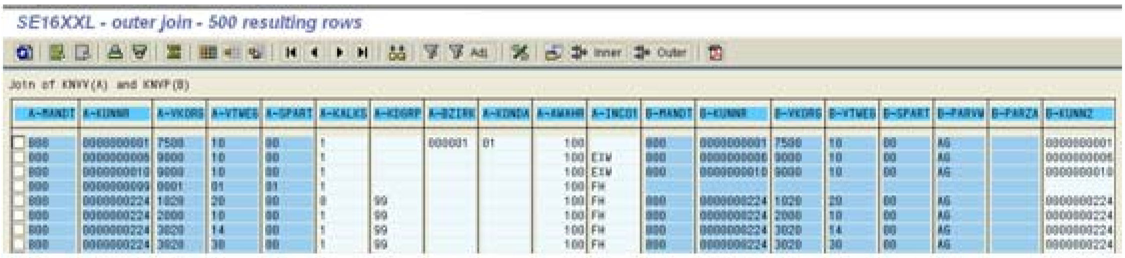
SE16XXL - Table KNVP - 461 entries selected

Table KNVP - Customer Master Partner Functions

MANDT	KUNNR	VKORG	VTWEG	SPART	PARVW	PARZA	KUNN2	LIFNR	PERNR	PARNR	KNREF	DEFPA
□ 800	0000000001	7500	10	00	AG		0000000001					
□ 800	0000000006	9000	10	00	AG		0000000006					
□ 800	0000000010	9000	10	00	AG		0000000010					
□ 800	0000000224	1020	20	00	AG		0000000224					

With "**Inner join**" the selection of table KNVP works the same as above described. But the result list is obtained by joining each original row (KNVV) with a matching entry of KNVP, thus giving rise to the join list displayed on the previous page. Rows of KNVV for which no matching entries of KNVP have been selected, are **<u>not present</u>** on the join list.

With "**Outer join**" the situation is similar to "Inner join". The only difference is that rows of KNVV with no matching KNVP are **<u>still present</u>** on the resulting join list. **This makes an outer join the ideal operation for finding "non-existing" combinations of data.** The join list might look as follows (notice the row with an **<u>empty</u>** KNVP entry):



SE16XXL - outer join - 500 resulting rows

Join of KNVV(A) and KNVP(B)

A-MANDT	A-KUNNR	A-VKORG	A-VTWEG	A-SPART	A-KALKS	A-KDGRP	A-BZIRK	A-KONDA	A-AWAHR	A-INCO1	B-MANDT	B-KUNNR	B-VKORG	B-VTWEG	B-SPART	B-PARVW	B-PARZA	B-KUNN2
□ 800	0000000001	7500	10	00	1		000001	01	100		800	0000000001	7500	10	00	AG		0000000001
□ 800	0000000006	9000	10	00	1				100	EXW	800	0000000006	9000	10	00	AG		0000000006
□ 800	0000000010	9000	10	00	1				100	EXW	800	0000000010	9000	10	00	AG		0000000010
□ 800	0000000099	0001	01	01	1				100	FH								
□ 800	0000000224	1020	20	00	0	99			100	FH	800	0000000224	1020	20	00	AG		0000000224
□ 800	0000000224	2000	10	00	1	99			100	FH	800	0000000224	2000	10	00	AG		0000000224
□ 800	0000000224	3020	14	00	1	99			100	FH	800	0000000224	3020	14	00	AG		0000000224
□ 800	0000000224	3020	30	00	1	99			100	FH	800	0000000224	3020	30	00	AG		0000000224

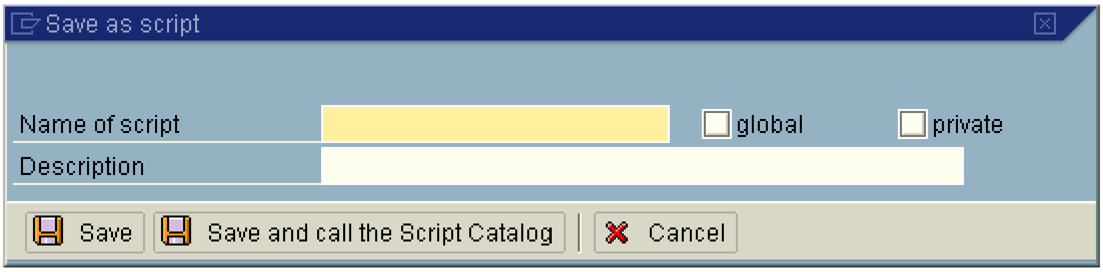
And what happens with the original list? It is pushed on a stack and becomes visible again when the current list is relinquished by pressing the button on the system function bar. The stack keeps a maximum of eight result lists. When this number is exceeded, the deepest list is discarded.

Functions on the system function bar

Save as script …

The operations performed up to this point are saved as an SE16XXL script. In this way they can be performed again at a later time.

The following dialog box shows up:



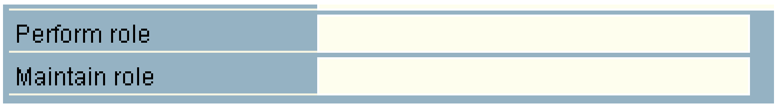
The name of a script may be up to 24 characters long. Global scripts, which are at the disposal of all other users, must begin with § or $. All other scripts are user-specific. A user-specific script may be marked as "private", in which case it is invisible for all other users. Otherwise user-specific scripts of other users can be inspected by means of the Script Catalog. For more information please refer to Script Catalog.

To save the script and proceed, press the Save button.

Press Save and call the Script Catalog instead to save the script and call the Script Catalog for further processing of the script.

Note: the Save and call the Script Catalog button is not present if you are already performing a script starting from the Script Catalog.

Depending on the current customizing settings, the above dialog box may differ slightly from the picture shown. In particular, parameters for specifying authorization roles for global scripts may be present:



Go back one level.

The current list is discarded and the “previous” list, if there is one, is popped from the stack and becomes once more the current one. Referring to our join example, leaving the join list would display the list of KNVV entries again.

Exit

The current list is relinquished and the first selection screen appears again.

Cancel

Works just like (Exit).

Print

The list is printed to spool. Only the first 255 characters of the result list can be printed. This is an SAP wide limitation.

Find

Standard find function. ALV Grid makes use of a somewhat different dialog box and behaves differently in regard to the hits.

Find next

Sets the cursor on the next hit. In older versions of ALV Grid this button does not work at all.

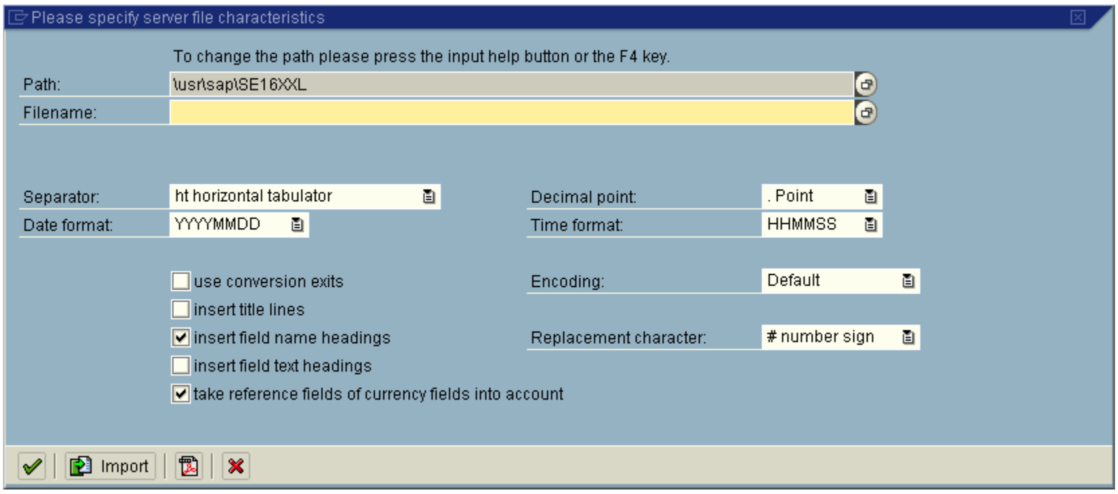
Scroll buttons.

These functions are not available with ALV Grid Display.

Menu functions

List → Save to server file

The current result list is saved as a file on the SAP application server. The following dialog box shows up for specifying the file characteristics:



Please specify server file characteristics

To change the path please press the input help button or the F4 key.

Path: \usr\sap\SE16XXL

Filename:

Separator: ht horizontal tabulator

Decimal point: . Point

Date format: YYYYMMDD

Time format: HHMMSS

☐ use conversion exits

Encoding: Default

☐ insert title lines

☑ insert field name headings

Replacement character: # number sign

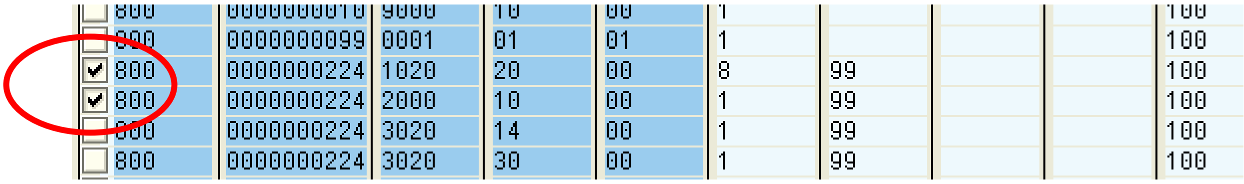
☐ insert field text headings

☑ take reference fields of currency fields into account

For more information please refer to Server Files.

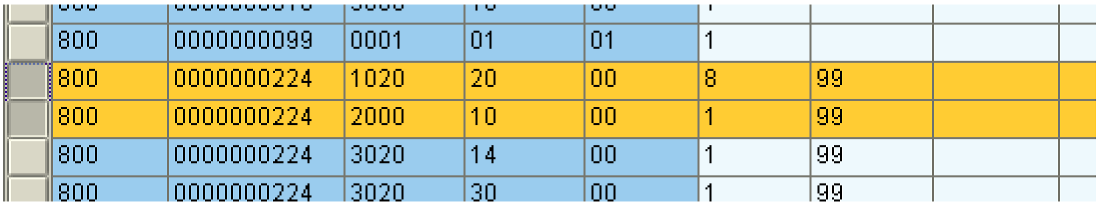
Rows → Hide selected rows

Just to make it clear: selected rows look like this:



☐	800	0000000010	9000	10	00	1				100
☐	800	0000000099	0001	01	01	1				100
☑	800	0000000224	1020	20	00	8	99			100
☑	800	0000000224	2000	10	00	1	99			100
☐	800	0000000224	3020	14	00	1	99			100
☐	800	0000000224	3020	30	00	1	99			100

Or, in ALV Grid, like this:



800	0000000099	0001	01	01	1			
800	0000000224	1020	20	00	8	99		
800	0000000224	2000	10	00	1	99		
800	0000000224	3020	14	00	1	99		
800	0000000224	3020	30	00	1	99		

These rows are then hidden, just as if they had been eliminated by a filter.

Note: this operation is ignored in a script, because which rows have been selected does not depend on any rule and cannot therefore be applied to other situations.

Rows → Hide rows without duplicates

This function works similarly to "Hide adjacent duplicates" (Adj.) .
But instead of keeping the first row of each group with identical sort criteria, all rows without duplicates are hidden. This means that only rows with duplicates remain visible on the result list.

If any rows were hidden, the button becomes available on the application toolbar.

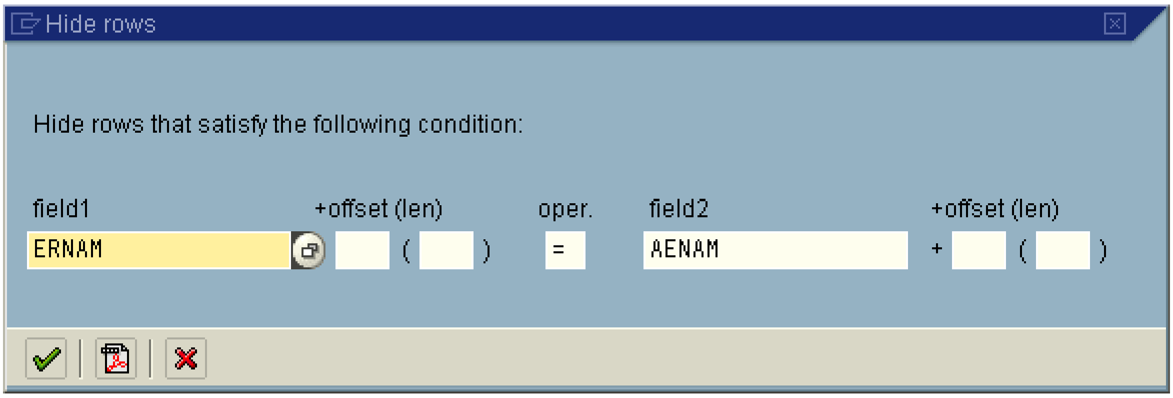
Rows → Hide rows with duplicates

This is the counterpart of the above. All rows with duplicates are hidden. Only rows without duplicates remain visible on the result list.

If any rows were hidden, the button becomes available on the application toolbar.

Rows → Hide rows where field1 op field2

This function compares the contents of two columns according to an operator op and hides the rows for which the condition is satisfied.
It is advisable to mark the two columns in advance. The following dialog box shows up for specifying the details of the operation:



Hide rows

Hide rows that satisfy the following condition:

field1	+offset (len)	oper.	field2	+offset (len)
ERNAM	( )	=	AENAM	+ ( )

The values of character-like columns can also be considered partially by means of an offset and a partial length.

For more information regarding this interesting function please refer to Comparing two fields of each row.

Rows → Show again hidden rows

This function is equivalent to the button on the toolbar.
But it is always available, even if no rows were hidden. This comes handy when creating a script, in case a particular situation does not lead to any hidden rows, but a more general situation would.

Rows → Discard hidden rows

By means of this function all currently hidden rows are discarded.
The button disappears from the application toolbar.

It is advisable to make use of this function when creating a script, so that the end users only get to see the result of any applied hide operations, and cannot make use of function “***Show again hidden rows***” to reveal rows that do not belong to the scope of the script.

The function also releases the memory allocated by the hidden rows, which may be considerable.

Edit → Formula → Display formula

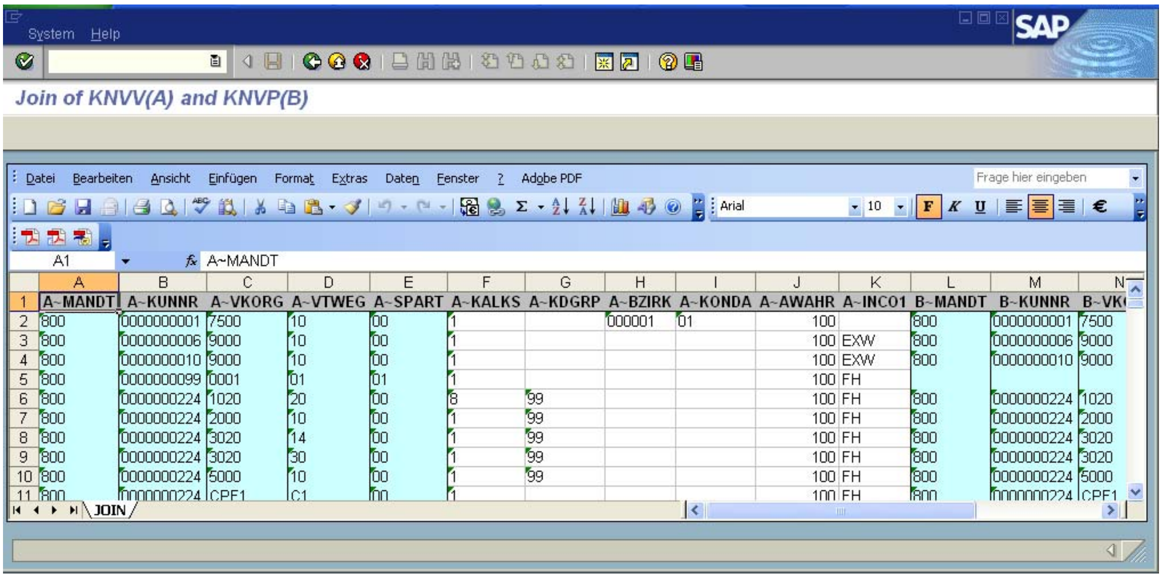
The currently active formula, if there is any, is displayed.
If only an inactive formula is present, i.e. the formula has been defined but not activated, it is also displayed. Once the formula is being displayed in the formula editor, it is possible to switch to “editing mode” (and back).

Edit → Formula → Delete formula

Any currently defined formula, either active or inactive, is deleted. If the formula was active, the formula fields (beginning with X~) disappear from the result list. If any of the formula fields are being used as sort criteria, it will not be possible to delete the formula.

Goto → Spreadsheet in place

The current result list is displayed in a spreadsheet embedded into an SAP window. The output might look as follows:



Join of KNVV(A) and KNVP(B)

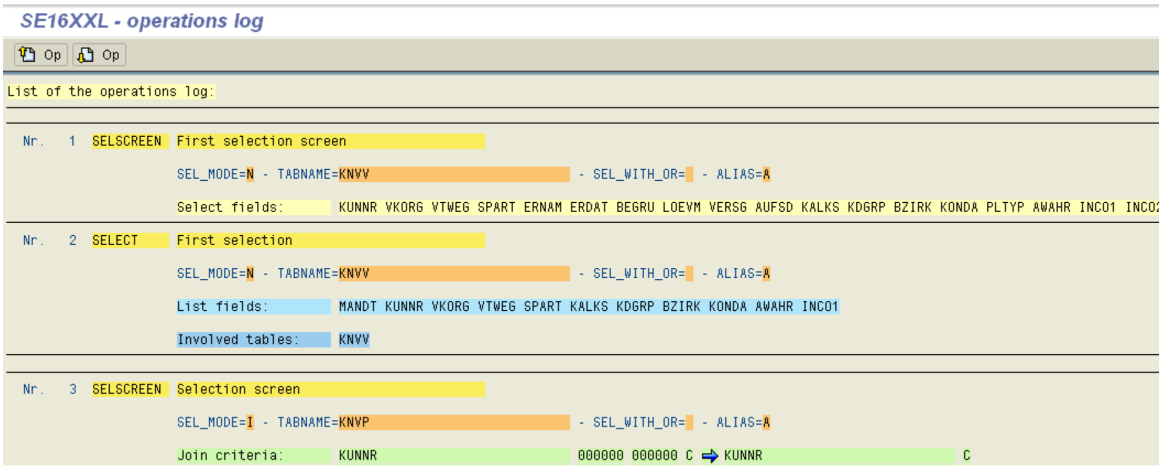
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	A~MANDT	A~KUNNR	A~VKORG	A~VTWEG	A~SPART	A~KALKS	A~KDGRP	A~BZIRK	A~KONDA	A~AWAHR	A~INCO1	B~MANDT	B~KUNNR	B~VK
2	800	0000000001	7500	10	00	1		000001	01	100		800	0000000001	7500
3	800	0000000006	9000	10	00	1				100	EXW	800	0000000006	9000
4	800	0000000010	9000	10	00	1				100	EXW	800	0000000010	9000
5	800	0000000099	0001	01	01	1				100	FH			
6	800	0000000224	1020	20	00	8	99			100	FH	800	0000000224	1020
7	800	0000000224	2000	10	00	1	99			100	FH	800	0000000224	2000
8	800	0000000224	3020	14	00	1	99			100	FH	800	0000000224	3020
9	800	0000000224	3020	30	00	1	99			100	FH	800	0000000224	3020
10	800	0000000224	5000	10	00	1	99			100	FH	800	0000000224	5000
11	800	0000000224	CPF1	C1	00	1				100	FH	800	0000000224	CPF1

Columns of type CHAR are formatted as text to avoid the spreadsheet program from interpreting any numerical values in the wrong way.

By taking advantage of the “Save as” functionality of the spreadsheet, it is possible to create files of different types on the frontend.

Goto → Show operations log

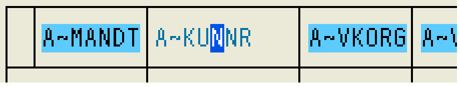
The operations carried out up to now (or the ones contained in a script being performed) are listed. A typical list might be as follows:



```
List of the operations log:

 Nr.   1  SELSCREEN  First selection screen

                     SEL_MODE=N - TABNAME=KNVV                   - SEL_WITH_OR=  - ALIAS=A

                     Select fields:     KUNNR VKORG VTWEG SPART ERNAM ERDAT BEGRU LOEVM VERSG AUFSD KALKS KDGRP BZIRK KONDA PLTYP AWAHR INCO1 INCO2
 Nr.   2  SELECT     First selection

                     SEL_MODE=N - TABNAME=KNVV                   - SEL_WITH_OR=  - ALIAS=A

                     List fields:       MANDT KUNNR VKORG VTWEG SPART KALKS KDGRP BZIRK KONDA AWAHR INCO1

                     Involved tables:   KNVV


 Nr.   3  SELSCREEN  Selection screen

                     SEL_MODE=I - TABNAME=KNVP                   - SEL_WITH_OR=  - ALIAS=A

                     Join criteria:     KUNNR                    000000 000000 C ⇨ KUNNR                    C
```

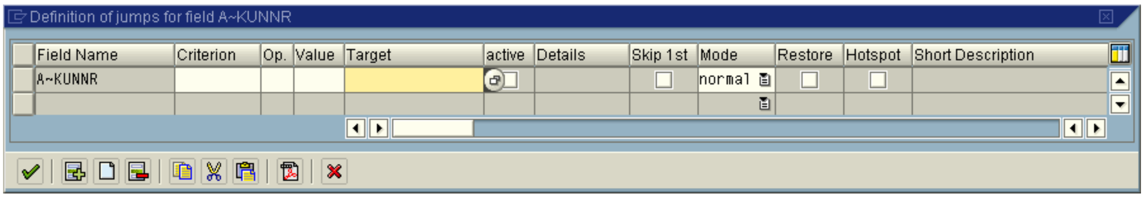
Extras → Define field jumps

This function can be used to define a "jump" for a particular column of the result list. A "jump" is a transaction which is called when the user double-clicks on a value of the column. Before calling the function, the involved column should be marked. For example:



A~MANDT	A~KUNNR	A~VKORG	A~\

The following dialog box shows up:



Definition of jumps for field A~KUNNR

Field Name	Criterion	Op.	Value	Target	active	Details	Skip 1st	Mode	Restore	Hotspot	Short Description
A~KUNNR					☐		☐	normal	☐	☐	

Jumps are usually defined for a script using the Script Catalog. The present function is used mostly to fine tune the definition carried out in the Script Catalog.

For further details about jumps please refer to Defining Jumps.

Extras → Define title lines

This is used to define special title lines (up to ten) for the result list.

For more information please refer to Defining Title Lines.

Extras → Define headings etc.

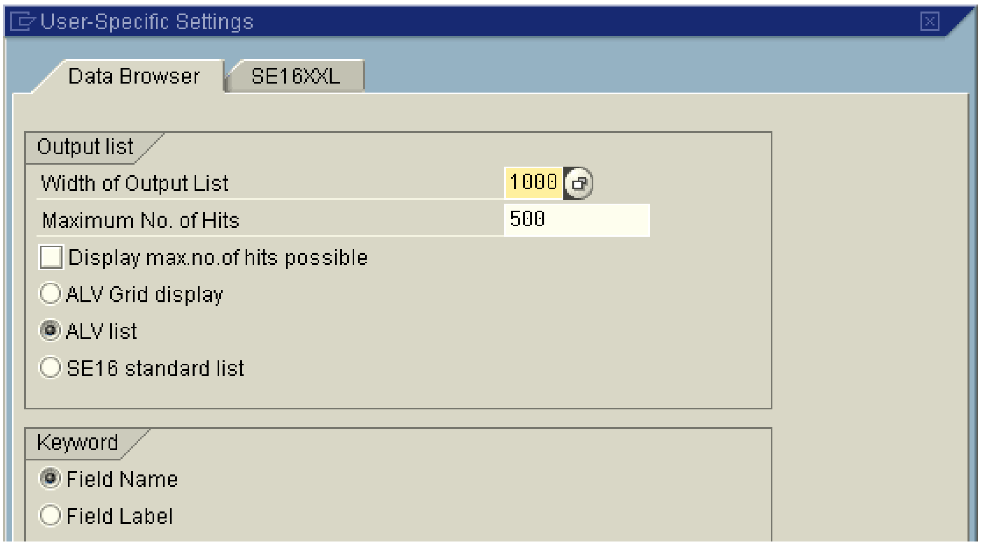
By means of this function it is possible to redefine the standard headings of the result list. This refers to the headings displayed when the user parameters are set to "Field Label". Field names cannot be redefined.
Additionally it is possible to restrict the columns which are available for display – i.e. to define which columns are "selectable".

Both features are usually defined for a script using the Script Catalog. The present function is used mostly to fine tune the definition carried out in the Script Catalog.

For more information please refer to Defining List Headings.

Settings → User parameters …

This is the same function as the button on the first screen of SE16XXL.
The usual dialog box shows up:



Output list

Width of Output List: 1000

Maximum No. of Hits: 500

□ Display max.no.of hits possible

○ ALV Grid display

● ALV list

○ SE16 standard list

Keyword

● Field Name

○ Field Label

Settings → Unfreeze key columns

This function is only available in SE16 standard mode for simple lists.
The columns containing key fields of the current table are unlocked and can then be scrolled horizontally like the remaining columns.
If the key columns occupy a great extent of the screen they are automatically displayed unlocked. The same is true for join lists.

Settings → Freeze key columns

The opposite of the above function. Not available for join lists.

Settings → Layout → Columns in DDIC order

This is a useful function to rearrange the order of the list columns in the same order as the fields in the Data Dictionary (DDIC). The function is only available in ALV List or ALV Grid Display mode.

Such a rearrangement may become necessary after the columns were shuffled around. To switch momentarily to SE16 Standard mode does not help, because SE16XXL memorizes the ALV order, and restores it once the user switches back to ALV mode.

Additional reading

Please make use of the following links for more information:

Reading SAPscript texts directly

How to find the “right” database tables

Script Catalog

Background Overview

Executing Scripts in Background

Saving a result list to a Server File

Uploading Start files

Defining Formulas

Special Selection Screens for Scripts

Script Variants

Editing Scripts

Frequently Asked Questions (FAQs)

Q: ***What is SE16XXL all about?***

A: SE16XXL is a tool – superficially similar to the standard Data Browser – for listing the contents of database tables according to complex criteria. Joins of tables together with an array of special filters make it possible to produce ad hoc results which normally require an ABAP report.
By means of scripts, users are able to perform the same tasks over and over again, either directly or in background.

Q: ***Is SE16XXL difficult to use?***

A: Owing to the similarity to the standard Data Browser, it is relatively easy for most users to switch to SE16XXL.

Q: ***What is the key aspect of SE16XXL?***

A: Actually there are two key aspects:

- the possibility of finding special data combinations covering several database tables – a task that presents itself each time under different circumstances and would otherwise require an ad hoc ABAP program.

- The capability of producing complex lists of the contents of several database tables – complete with customized headings and other special features. These complex lists are the result of scripts that can be performed either in dialog mode or, possibly periodically, in background.

Q: ***What exactly is an SE16XXL script?***

A: When using SE16XXL, the most important operations carried out, such as performing a join, setting a filter, defining a formula and so on, are stored into the so-called Operations Log. By means of the function "Save as script …" these operations can be stored away as a script for later use.

Once a script has been created, it can be enhanced in various ways, for example by defining a special selection screen tailored to its needs.
Please refer to Script Catalog for more information.

Q: ***Is it possible to edit an SE16XXL script?***

A: **Yes**. This function is available in the Script Catalog. For more information please refer to Script Editor.

Q: ***Can the contents of a database table be changed using SE16XXL?***

A: **No**. SE16XXL has no such functionality and will never have.

Q: ***Are any modifications of the standard SAP system necessary in order to run SE16XXL?***

A: **No**. The standard SAP system does not need to be modified. There is only one SAP note (number **1099527)** which it is advisable to implement. Please refer to the Technical Details documentation.

Q: ***Is SE16XXL compatible with Unicode?***

A: **Yes**. SE16XXL runs without changes starting from SAP_BASIS version 700 with and without Unicode.

Q: ***Does SE16XXL generate a program for every database table like the standard SAP data browser?***

A: **No**. SE16XXL uses the same program for all tables. In order to cope with the various tables, it generates dynamic coding.

Q: ***How can the use of SE16XXL be restricted?***

A: There are several levels of authorization and permission checks.

Permissions can be defined down to single database tables and their fields. Selected data elements can be globally forbidden.

Authorization checks can be defined for specific fields of a table, like the sales organization (VKORG) of table VBAK. Users will only be able to select the entries of “their” sales organization(s).

Authorization checks may also be defined for scripts.

Q: ***Is SE16XXL capable of reading files?***

A: Instead of selecting entries from a database table, it is possible to upload a file from the frontend. After the user has defined how the file contents are to be interpreted, the rest proceeds as for a normal database table.
Only frontend files with a simple structure, like CSV files, can be uploaded.
For more information please refer to Uploading a start file.

Q: ***Is it possible to redefine the standard column headings of a result list?***

A Yes. Such functionality is available in the Script Catalog for scripts and in a normal session for the current list. Please refer to Defining list headings.

Q: ***Is it possible to call transactions from a result list based on the displayed values?***

A: Yes. These are the so-called “Jumps” which can be defined for a script in the Script Catalog and also in a normal session for the current list.
For more information please refer to Defining jumps.

Q: ***Is it possible to perform “difficult” joins, i.e. joins where the fields used as criteria do not match completely?***

A: Yes. For example CHAR fields of different length can still be joined. The same is true for joining a CHAR field with a NUMC field. The fields of the original table can be taken partially by means of offset and length. Composite criteria can be used as well. If nothing else works, it is possible to create a special field customized to carry out the join by defining a formula.

Q: ***Can an SE16XXL script be performed in background?***

A: Yes. There are two main modes of doing this, “embedded script” mode, in which the script is copied into the background request, and is therefore immune against any subsequent changes of the script structure, and “referenced script” mode, in which only the name of the script is stored with the request. In the latter case, the current version of the script is fetched shortly before the script is performed. All this is relevant for periodically performed scripts. For more details please refer to Scripts in Background.

Q: ***Where are the results of background scripts stored?***

A: The results of a script run in background are stored in a special database table and are only at the disposal of the initiator of the run.
But it is possible to specify a file on the application server to be used as target. In this case all users authorized to access this file can get the results.
For more information please refer to Server Files.

Q: ***Is it true that SE16XXL can read SAPscript texts directly?***

A: Yes. To this purpose a pseudo database table called **$STXL** has been introduced, which behaves as if it were a real table. Internally function module READ_TEXT is called to expand the compressed data.
For more information please refer to Reading SAPscript texts directly.

Q: ***Is it possible to carry out "limited computations" in a result list?***

A: Yes. And not only computations. For this purpose "Formulas" have been introduced. A formula is a series of ABAP statements (not all ABAP statements are allowed) for defining additional fields of the result list and their contents. Thus a formula field may contain the sum of two other fields, or be related to other fields in other ways, acting for instance as a flag to denote particular data situations. For more information refer to Formulas.

Q: ***Is SE16XXL capable of selecting data on remote systems via RFC?***

A: No. This functionality has not been implemented yet. Maybe in a future version.