

SE16XXL – Neue Funktionen

- April 2025 Neue Version **4.1** mit folgenden Funktionen:
- Script-Versionen [Mehr ...](#)
 - Scripts mithilfe eines Transportauftrags löschen [Mehr ...](#)
 - Neue Operation “Eindeutige Werte zählen” [Mehr ...](#)
 - Zusätzliche Verbesserungen [Mehr ...](#)
- Oktober 2024 Neue Version **4.0** mit folgenden Funktionen:
- Neue Operation “Zeilen in Spalten umwandeln” [Mehr ...](#)
 - Verbesserungen der Hintergrund-E-Mail-Parameter [Mehr ...](#)
 - Script-Katalog – Wertehilfe für Script-Ersteller [Mehr ...](#)
 - Klonen von FLTP-Feldern zu anderen Typen [Mehr ...](#)
 - Verbesserungen für datumsbezogene Felder [Mehr ...](#)
 - GUID-Konversionen in Formeln [Mehr ...](#)
 - Neue Benutzer-Einstellung “Schlüsselspalten rollbar” [Mehr ...](#)
 - Liste von in Scripts definierten Sprüngen (TCodes) [Mehr ...](#)
 - Zusätzliche Verbesserungen [Mehr...](#)
- März 2024 Neue Version **3.6E** mit folgenden Funktionen:
- Namen von Formelfeldern bis zu 30 Stellen lang [Mehr ...](#)
 - Namen von V-Feldern bis zu 30 Stellen lang [Mehr ...](#)
 - Anmerkungen zu Script-Operationen hinzufügen [Mehr...](#)
 - Klonen von langen zeichenartigen Feldern [Mehr...](#)
 - Join-Layouts [Mehr...](#)
 - Zusätzliche Verbesserungen [Mehr...](#)

November 2023 Neue Version **3.6D** mit folgenden Funktionen:

- Werthilfe (F4-Hilfe) für Joins [Mehr ...](#)
- Werthilfe (F4-Hilfe) beim Ersetzen einer Tabelle [Mehr ...](#)
- Neue Option beim Einplanen eines Scripts [Mehr...](#)
- Hintergrund-Übersicht – einige neue Funktionen [Mehr...](#)
- Selektionswerte für die SFMI-Schnittstelle [Mehr...](#)

July 2023 Neue Version **3.6C** mit folgenden Funktionen:

- CDS-View-Entitäten unterstützt [Mehr ...](#)
- Neue dynamische Datumsvariablen für Script-Varianten [Mehr ...](#)
- Längere Spaltennamen für Frontend-Dateien [Mehr ...](#)
- Priorität für E-Mails in Hintergrund-Scripts [Mehr ...](#)
- Zusätzliche Verbesserungen [Mehr ...](#)

April 2023 Neue Version **3.6B** mit folgenden Funktionen:

- Scriptspezifische ALV-Layouts [Mehr ...](#)
- ätzliche Verbesserungen [Mehr ...](#)

Dezember 2022 Neue Version **3.6A** mit folgenden Funktionen:

- Neue Art von globalem Parameter – Zeitraum [Mehr ...](#)
- Zuordnung von globalen Parametern zu Select-Options [Mehr ...](#)
- Spezielle Selektionsmaske – dynamische Zuordnungen [Mehr ...](#)
- Spalten einfügen mit datumsbezogenen Feldern [Mehr ...](#)
- Summieren auf der DB mithilfe eines Jahres-V-Feld [Mehr ...](#)
- Einfärben einzelner Spalten der Ergebnisliste [Mehr ...](#)
- Gültigkeits-Prüfung – aktuelles Datum als Ref.-Datum [Mehr ...](#)
- E-Mails mit Server-Datei-Anhang – mehr Flexibilität [Mehr ...](#)
- Benutzer-Einstellungen – Default-Dateipfade [Mehr ...](#)

- Elementartyp UTCLONG unterstützt [Mehr ...](#)
- Zusätzliche Verbesserungen [Mehr ...](#)

Mai 2022 Neue Version **3.6** mit folgenden Funktionen:

- Neue Arten von Berechtigungsprüfungen [Mehr ...](#)
- Neue Pseudo-Tabellen \$APPLOGH und \$APPLOGS [Mehr ...](#)
- Neue Start-Optionen für Scripts im Hintergrund [Mehr ...](#)
- Standard-Scriptvarianten U_<benutzer> unterstützt [Mehr ...](#)
- Zusätzliche Verbesserungen [Mehr ...](#)

November 2021 Neue Version **3.5B** mit folgenden Funktionen:

- Summieren/Zählen direkt auf der Datenbank [Mehr ...](#)
- Standard-Selektionsmaske neu gestaltet [Mehr ...](#)
- Neue Script-Selektionsmaske mit Registerkarten [Mehr ...](#)
- Zusätzliche Verbesserungen [Mehr ...](#)

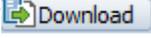
April 2021 Neue Version **3.5A** mit folgenden Funktionen:

- SAP® Data-Aging-Funktionalität unterstützt [Mehr ...](#)
- Spezielle Behandlung von invertierten Datumsfeldern [Mehr ...](#)
- Währungsumrechnungen in Formeln [Mehr ...](#)
- Upload von Frontend-Dateien vom Typ .XLS usw. [Mehr ...](#)
- Neue Option für Server-Datei als E-Mail-Anhang [Mehr ...](#)
- Anzeige der Struktur eines mehrstufigen Views [Mehr ...](#)

[Weitere historische Einträge ...](#)

Script-Versionen

Bisher gab es mehrere Möglichkeiten, eine Sicherungskopie eines SE16XXL Scripts zu erstellen:

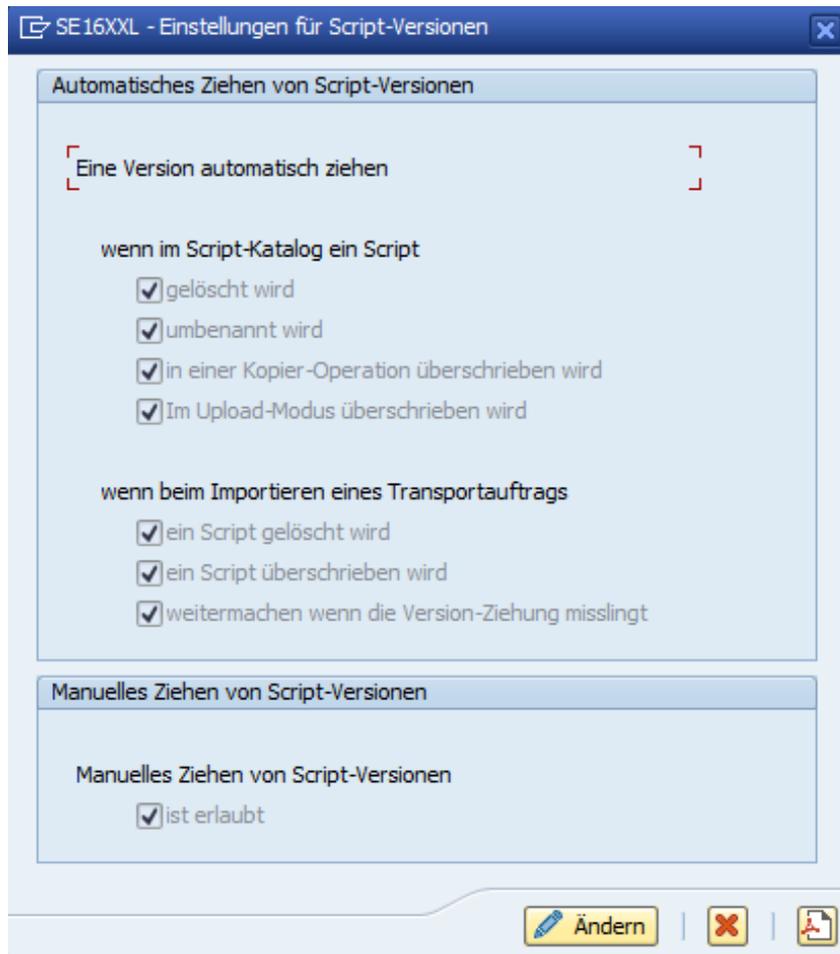
- 1) Im Script-Katalog mithilfe der Funktion  (Script kopieren);
- 2) Im Script-Katalog durch Herunterladen des Scripts ();
- 3) In der SE16XXL Ergebnisliste des Scripts durch sichern () unter einem anderen Namen;
- 4) Im Script-Editor über die Menüfunktion
Script → Sichern als ...
unter Verwendung eines anderen Namens;
- 5) Im Script-Katalog über die Menüfunktion
Ausgewählte Scripts → Transport → Einem Transportauftrag hinzufügen.

All diesen Methoden ist gemeinsam, dass sie **explizit** ausgeführt werden müssen, d.h. sie werden **nicht automatisch** vom Programm ausgeführt. Hinzu kommt dass einige der Komponenten eines Scripts, wie z.B. die inaktive Version, die Script-Varianten und die scriptspezifischen ALV-Layouts entweder optional sind oder gar nicht berücksichtigt werden. Falls jemand vergisst, eine Sicherungskopie eines Scripts zu erstellen, und es dann versehentlich löscht, gibt es keine Möglichkeit, es wiederherzustellen – es muss manuell neu aufgebaut werden. Selbst wenn es eine Sicherungskopie der oben genannten Art gibt, kann es nicht so einfach sein, die richtige zu finden – Scriptkopien können Namen haben, die nicht selbstverständlich sind, oder die Download-Dateien wurden in andere Verzeichnisse verschoben usw.

Aus all diesen Gründen sind ab SE16XXL Version 4.1 **Script-Versionen** eingeführt worden. Eine Script-Version ist eine vollständige Kopie eines Scripts (entweder global oder benutzerspezifisch), einschließlich der inaktiven Version (falls vorhanden), der Script-Varianten, der scriptspezifischen ALV-Layouts und der Default-ALV-Layouts. Für ein bestimmtes Script kann eine beliebige Anzahl von Versionen gezogen werden. Jede hat ihren eigenen Zeitstempel, so dass es möglich ist, sie nach Erstellungsdatum und Uhrzeit zu sortieren. Auf diese Weise ist es sehr einfach, die letzte Version eines bestimmten Scripts zu finden.

Eine Version eines Scripts kann entweder **automatisch** von SE16XXL oder **manuell** von einer interessierten Person gezogen werden. Die Systemadministration legt fest, für welche Situationen eine Script-Version automatisch erstellt werden soll. Sie kann auch die manuelle Ziehung von Script-Versionen erlauben.

Die möglichen **Einstellungen** sind in folgender Abbildung zu sehen:



Wie zu sehen ist, kann auch eine Script-Version gezogen werden, wenn eine **Löschung** mithilfe eines **Transportauftrags** durchgeführt wird. Dies ist eine **neue Funktion** der Version 4.1 (siehe nächstes Thema dieser Dokumentation).

Die Verwaltung der Script-Versionen ist eine Mischung aus dem Papierkorb von Windows® und der Versionsverwaltung eines ABAP® Programms. Es gibt einen Katalog der Script-Versionen, der eine Reihe von Funktionen für den Umgang mit den Versionen bietet:

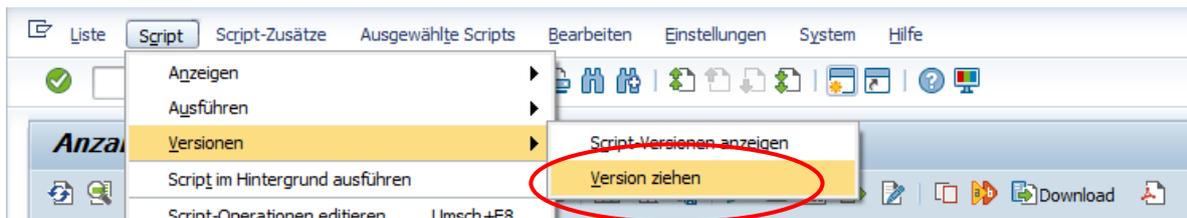
- Verschiedene Arten von Selektionskriterien;
- Kopieren einer Version in ein echtes Script;
- Eine Serie von Scripts aus älteren Versionen zurückholen;
- Kopieren ausgewählter Versionen in echte Scripts;
- Ziehung einer Version ausgewählter Scripts;
- Usw.

Weitere Informationen finden Sie im [Katalog der Script-Versionen](#).

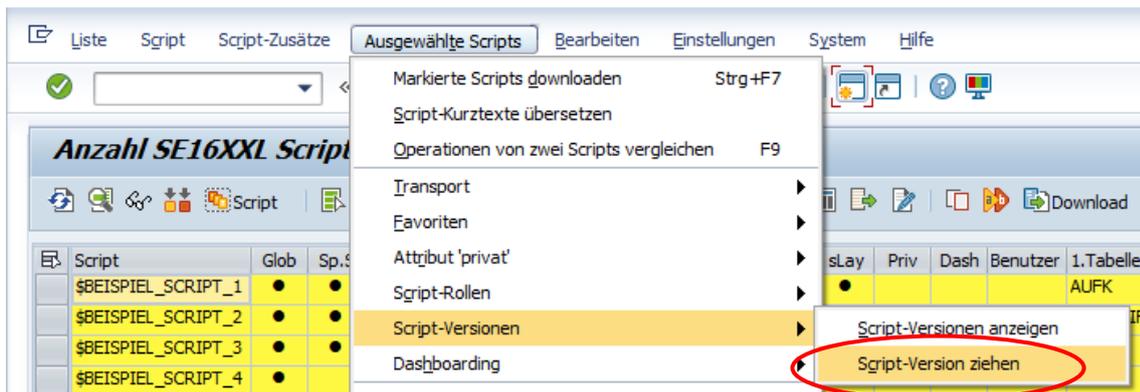
Manuelles Ziehen von Script-Versionen

Wie bereits erwähnt, werden Script-Versionen nicht nur **automatisch** erstellt wenn bestimmte Operationen durchgeführt werden (z. B. wenn ein Script im Script-Katalog gelöscht wird), sondern auch von einer Person **manuell** gezogen. Die verfügbaren Funktionen werden nun veranschaulicht.

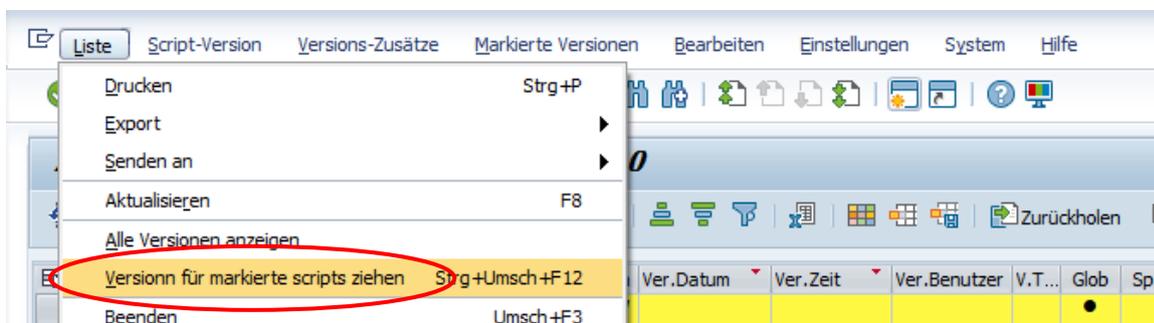
Im Script-Katalog für ein einzelnes Script:

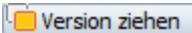


Im Script-Katalog für eine Serie von Scripts:



Im Katalog der Script-Versionen für eine Serie von Scripts:



Diese Funktion ist auch auf der Anwendungsleiste als  **Version ziehen** verfügbar.

[Zum Anfang](#)

Scripts mithilfe eines Transportauftrags löschen

Bisher konnten Scripts über einen Transportauftrag auf andere Systeme transportiert werden, es war jedoch nicht möglich, auf diese Weise Scripts auf anderen Systemen zu löschen. Scripts mussten immer lokal auf jedem einzelnen System gelöscht werden. Für große Unternehmen mit Dutzenden von produktiven Systemen kann dies ein ernsthaftes Problem darstellen. Aus diesem Grund ist es ab **Version 4.1** von SE16XXL möglich, **Script-Löschungen** zu einem Transportauftrag hinzuzufügen.

Beim Import dieses Auftrags werden die beteiligten Scripts mithilfe eines speziellen XPRA-Programms gelöscht. Vor dem Löschen eines jeden Scripts kann eine **Script-Version** erstellt werden, wenn die Systemadministration die entsprechenden **Einstellungen** vorgenommen hat (siehe das vorherige Thema dieser Dokumentation).

Die notwendigen Menüfunktionen stehen im Script-Katalog zur Verfügung. Eine Menüfunktion **fügt die markierten Scripts zu einem Transportauftrag hinzu**. Die andere bietet eine **Liste von Scripts auf einem entfernten System** an, um diejenigen auszuwählen, die zu einem Transportauftrag auf dem aktuellen System hinzugefügt werden sollen. Diese letzte Funktion ist hilfreich, wenn die zu löschenden Scripts sich nicht auf dem lokalen System befinden.

WICHTIG: Möglicherweise erwarten Sie eine Funktion wie *“ausgewählte Scripts löschen und in Löschungs-Transportauftrag eintragen”*. Nun, das ist **NICHT** der Fall. Die Funktion **“Löschen”** und die Funktion **“Einem Löschungs-Transportauftrag hinzufügen”** sind zwei getrennte Funktionen. Warum ist das so?

Nun, wenn das Hinzufügen eines Scripts zu einem Löschungs-Transportauftrag nur durch das Löschen des Scripts selbst möglich wäre, **wäre es schwieriger**, im Falle einer falschen Auswahl zu reagieren, da die beteiligten Scripts bereits gelöscht worden wären. Darüber hinaus ist es damit möglich, bestimmte Scripts **auf anderen Systemen**, jedoch **nicht auf dem aktuellen System** zu löschen. Aus diesen Gründen müssen die beteiligten Scripts in einem zweiten Schritt gelöscht werden, wenn dies wirklich gewünscht wird.

Weitere Informationen finden Sie unter [Löschen von Scripts mithilfe eines Transportauftrags](#).

[Zum Anfang](#)

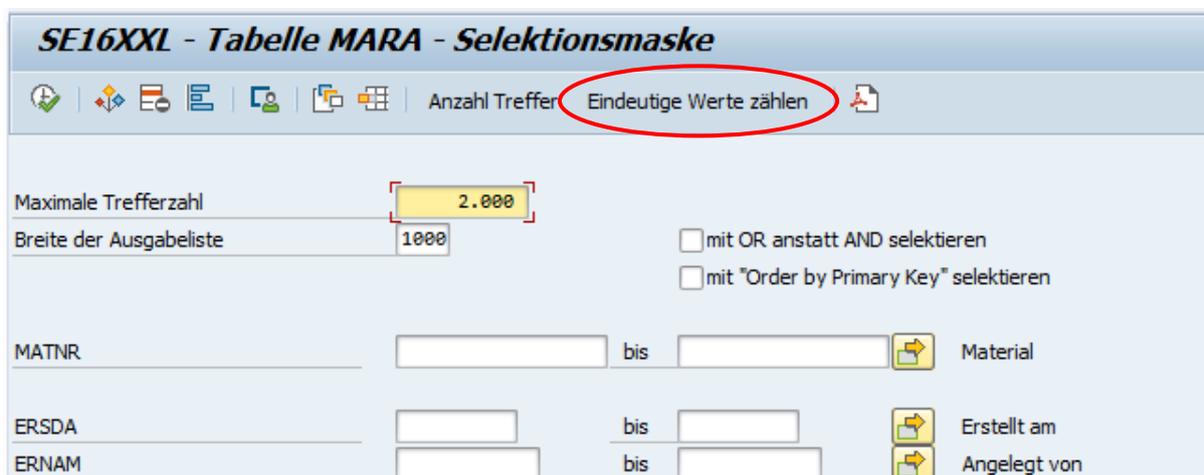
Neue Operation “Eindeutige Werte zählen”

Bisher war es nur möglich, die Anzahl der Einträge einer bestimmten Datenbanktabelle oder View gemäß den Selektionskriterien zu zählen. Ab Version **4.1** von SE16XXL ist es auch möglich, die **eindeutigen Werte** eines bestimmten Feldes einer Datenbanktabelle oder View zu zählen.

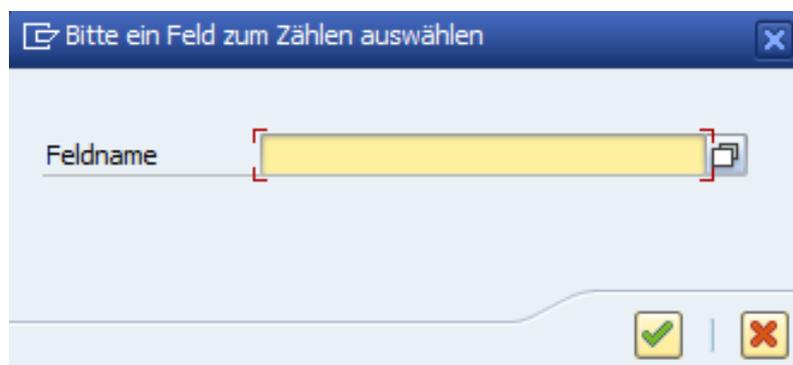
Diese Neuerung wird nun anhand einiger Beispiele veranschaulicht werden.

Beispiel #1 – mit Tabelle MARA (Allgemeine Materialdaten)

In SE16XXL wählen wir die Tabelle **MARA** und gelangen auf die Selektionsmaske:



Wie man sehen kann, gibt es jetzt eine neue Schaltfläche **Eindeutige Werte zählen** auf der Anwendungsleiste. Wenn wir von dieser Funktion Gebrauch machen, reagiert das Programm mit der Anzeige des folgenden Dialogfensters:



Wenn wir die **F4-Hilfe** nutzen, können wir ein beliebiges Feld der Tabelle auswählen, in diesem Fall MARA.

Wir wählen das Feld **MTART** (Materialart) und erhalten folgende Information:

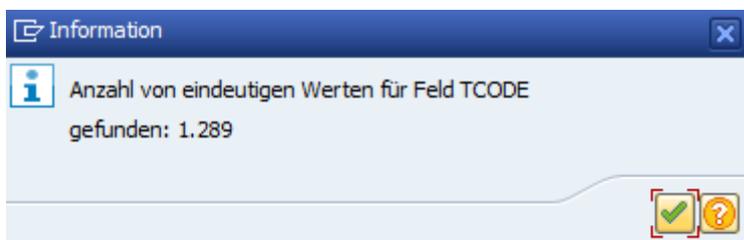


Diese Zählung ist sehr effizient, da sie, wie aus folgendem Coding hervorgeht, **direkt** auf der Datenbank durchgeführt wird:

```
CLEAR LL_MESS.  
TRY.  
    SELECT COUNT( DISTINCT MTART )  
          FROM MARA  
          INTO PP_COUNT  
          .  
CATCH CX_ROOT INTO LO_CXROOT.  
    PP_RC = 4.  
    LL_MESS = LO_CXROOT->GET_TEXT( ).  
ENDTRY.
```

Beispiel #2 – mit Tabelle CDHDR (Änderungsbelegkopf).

Diese Tabelle enthält in der Regel eine sehr große Anzahl von Einträgen. Die Zählung der eindeutigen Werte ist jedoch recht schnell, da sie auf der Datenbank durchgeführt wird. Wenn wir z.B. das Feld **TCODE** (Transaktion, in der eine Änderung durchgeführt wurde) wählen, das nicht einmal ein Sekundärindex der Tabelle ist, erhalten wir dennoch sofort folgende Antwort:



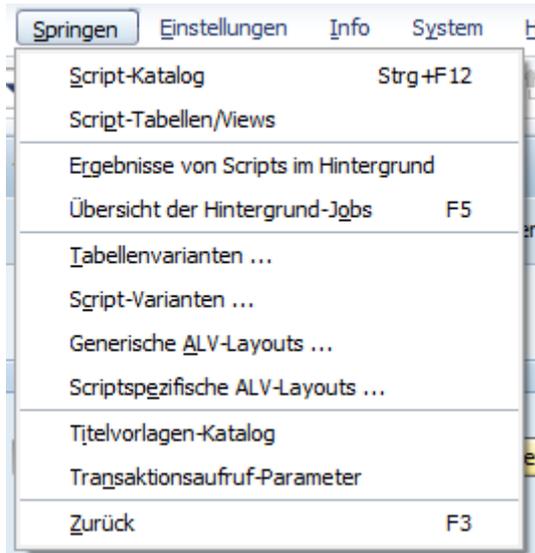
ANMERKUNG: Diese Funktion ist für Tabellen vom Typ POOL oder CLUSTER nicht verfügbar.

[Zum Anfang](#)

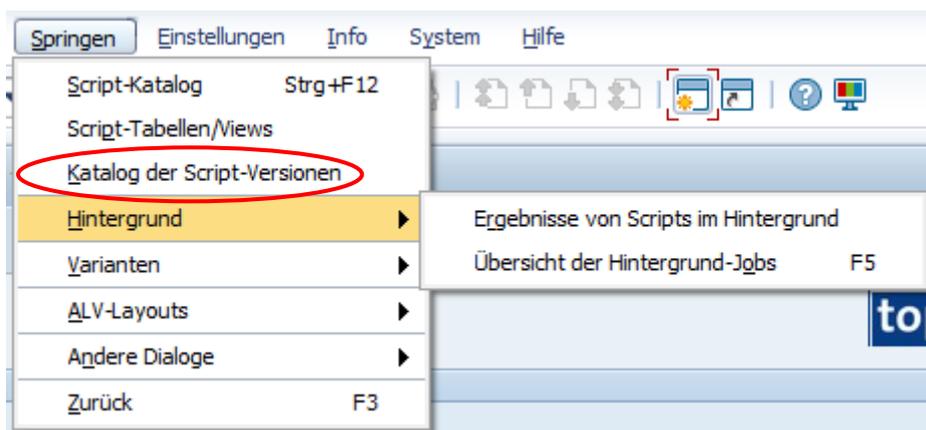
Zusätzliche Verbesserungen

Menü “Springen” auf der SE16XXL Startmaske neu organisiert

Das Menü “**Springen**” in der Menüleiste der Startmaske von SE16XXL war bereits an seine Grenzen gestoßen und konnte nicht weiter erweitert werden:



Um neue Funktionen hinzufügen zu können, sind ab Version **4.1** von SE16XXL **Untermenüs** eingeführt worden:

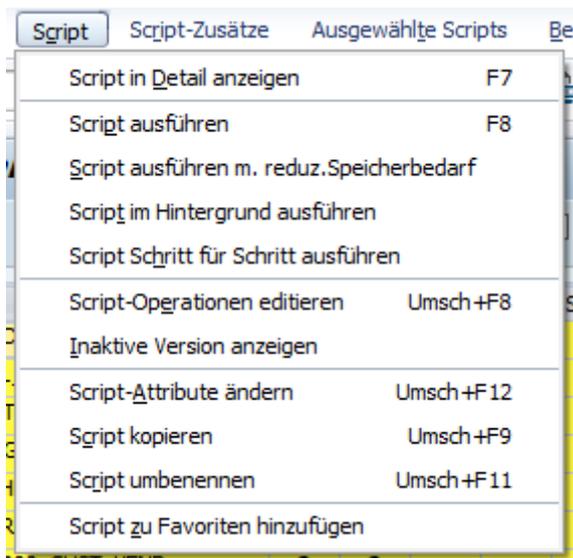


Wie man sehen kann, ist ein neuer Eintrag für “*Katalog der Script-Versionen*” hinzugefügt worden und die meisten der restlichen Funktionen sind in entsprechende Untermenüs verschoben worden.

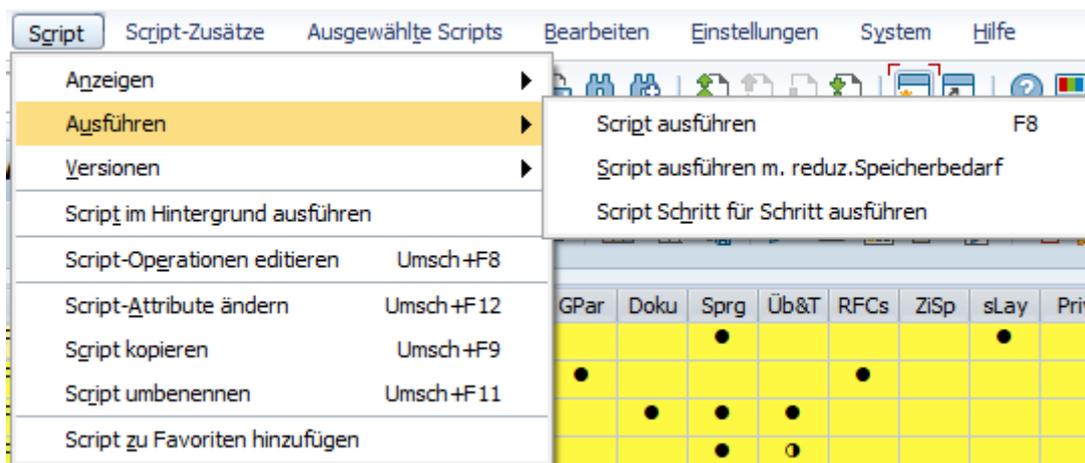
Mehrere Menüs der Hauptliste des Script-Katalogs neu organisiert

Die folgenden Menüs in der Hauptliste des Script-Katalogs sind ebenfalls neu organisiert worden, nachdem neue Funktionen hinzugefügt werden mussten:

Menü “Script” – bisherige Version

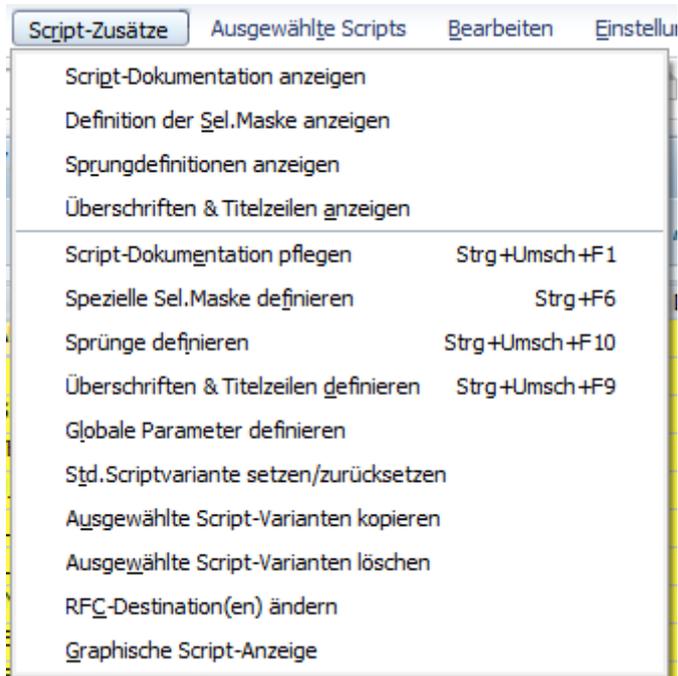


Menü “Script” – neue Version

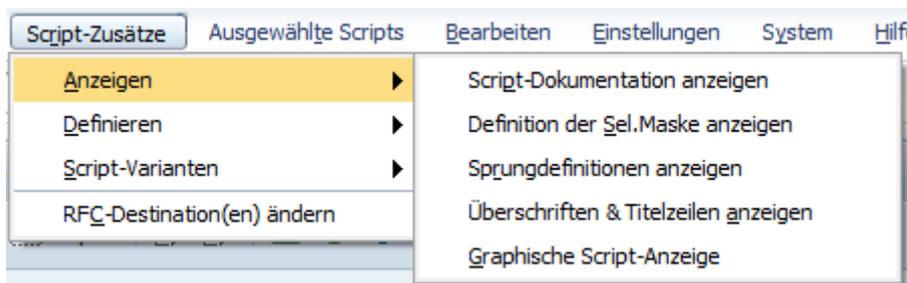


Wie man sehen kann, ist der untere Teil des Menüs unverändert geblieben. Im oberen Teil wurden Untermenüs hinzugefügt, mit Ausnahme von “*Script im Hintergrund ausführen*”, das keinem Untermenü sinnvoll zugeordnet werden konnte.

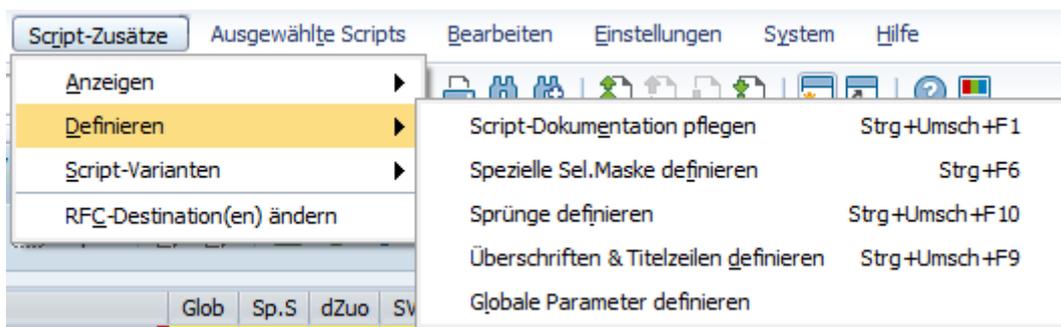
Menü “Script-Zusätze” – bisherige Version



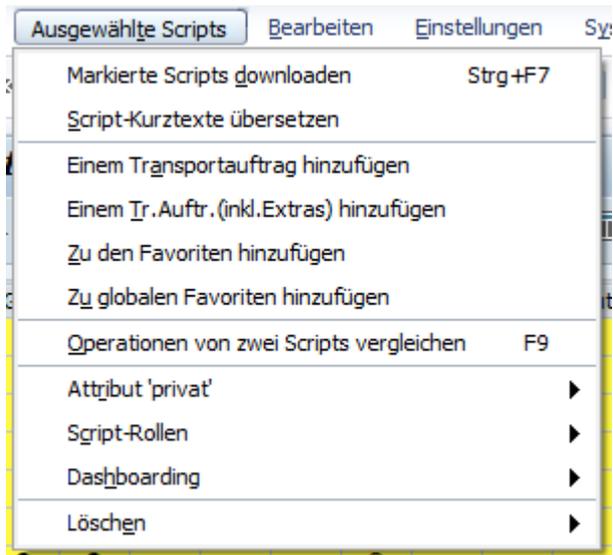
Menü “Script-Zusätze” – neue Version



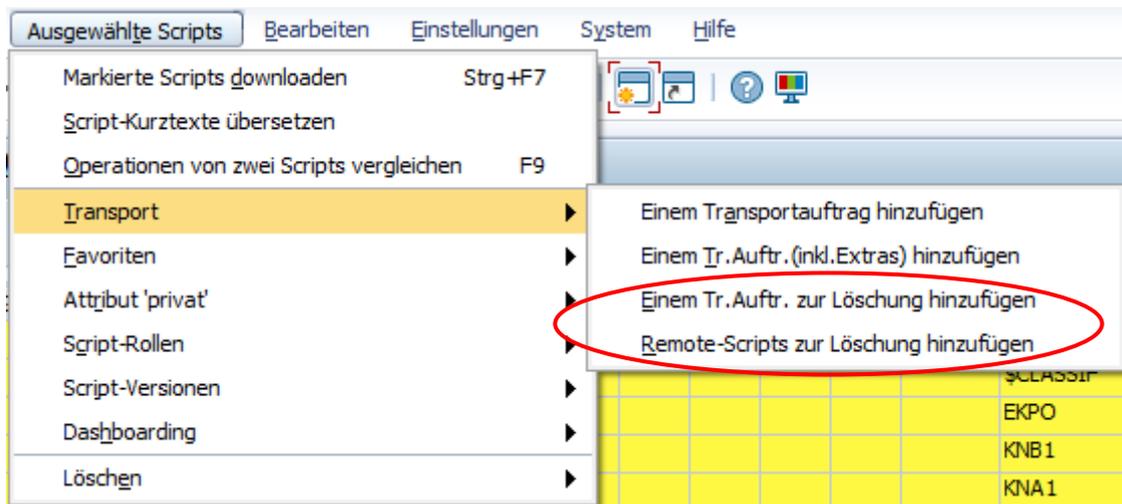
Und mit dem Untermenü “**Definieren**” expandiert:



Menü “Ausgewählte Scripts” – bisherige Version



Menü “Ausgewählte Scripts” – neue Version



Wie man sehen kann, sind zwei neue Transportfunktionen hinzugekommen. Darüber hinaus gibt es ein Untermenü für Script-Versionen.

Dateien vom Typ XLSX oder MHTML mit Ikonen

Angenommen, man hat eine SE16XXL Ergebnisliste wie die folgende:

Tabelle VBAP - Verkaufsbeleg: Positionsdaten										
MANDT	VBELN	POSNR	MATNR	NETWR	WAERK	KWMENG	KBMENG	VRKME	X~FLAG	
800	0000000012	000010	M100-100	300,00	EUR	2	2	ST		
800	0000000012	000020	100-100	200,00	EUR	2	2	ST		
800	0000000013	000010	M100-100	300,00	EUR	2	0	ST		
800	0000000013	000020	M03-001	500,00	EUR	500	0	ST		
800	0000000013	000030	M100-100	300,00	EUR	2	0	ST		

Bisher, wenn eine der folgenden Menüfunktionen

Liste → In Server-Datei sichern → in anderen Formaten (XLSX usw.)

Liste → Download zum Frontend → in anderen Formaten (XLSX usw.)

verwendet wurde, um eine Datei vom Typ XLSX oder MHTML zu erstellen, zeigte die Tabellenkalkulation die **internen Werte** der Ikonen anstelle der Ikonen selbst an:

	A	B	C	D	E	F	G	H	I	J	K
1	MANDT	VBELN	POSNR	MATNR	NETWR	WAERK	KWMEN	KBMENC	VRKME	X~FLAG	
2	800	0000000012	000010	M100-100	300,00	EUR	2	2	ST	@5B@	
3	800	0000000012	000020	100-100	200,00	EUR	2	2	ST	@5B@	
4	800	0000000013	000010	M100-100	300,00	EUR	2	0	ST	@5C@	
5	800	0000000013	000020	M03-001	500,00	EUR	500	0	ST	@5C@	
6	800	0000000013	000030	M100-100	300,00	EUR	2	0	ST	@5C@	

Das Gleiche passierte in einem **Script im Hintergrund**, wenn eine Server-Datei der oben genannten Typen als Ergebnis produziert wurde.

Mit Version **4.1** von SE16XXL wird dieses Problem behoben. In der Tabellenkalkulation werden die SAP-Ikonen jetzt **korrekt angezeigt**, wie in der folgenden Abbildung zu sehen ist:

	A	B	C	D	E	F	G	H	I	J	K
1	MANDT	VBELN	POSNR	MATNR	NETWR	WAERK	KWMEN	KBMENC	VRKME	X~FLAG	
2	800	0000000012	000010	M100-100	300,00	EUR	2	2	ST		
3	800	0000000012	000020	100-100	200,00	EUR	2	2	ST		
4	800	0000000013	000010	M100-100	300,00	EUR	2	0	ST		
5	800	0000000013	000020	M03-001	500,00	EUR	500	0	ST		
6	800	0000000013	000030	M100-100	300,00	EUR	2	0	ST		

E-Mails mit Anhang mit langem Dateinamen

Wenn ein Script im Hintergrund eingeplant wird, um eine Server-Datei zu erstellen, kann festgelegt werden, dass eine Reihe von E-Mails an ausgewählte Empfänger gesendet wird, wobei jede E-Mail die erzeugte Datei als Anhang enthält. Diese Funktion ist schon lange verfügbar. Bisher gab es jedoch ein Problem. Falls der Name der angehängten Datei länger als 50 Zeichen war, wurde er abgeschnitten. Dies führte zum Verlust des Dateityps, der für die korrekte Interpretation des Inhalts der Datei unerlässlich ist.

Dieses Problem ist darauf zurückzuführen, dass die Methode **ADD_ATTACHMENT** der Klasse **CL_DOCUMENT_BCS**, die für den Versand der E-Mails verwendet wird, in älteren Versionen von SAP nur den Parameter **I_ATTACHMENT_SUBJECT** zur Angabe des Namens der angehängten Datei anbietet. Dieser Parameter hat eine Länge von **50 Zeichen**. Daher werden längere Namen einfach abgeschnitten.

In **neueren Versionen** von SAP wurde dieser Methode ein zusätzlicher Parameter hinzugefügt. Der neue Parameter **I_ATTACHMENT_FILENAME** hat den Typ **STRING** und kann daher beliebige Namenslängen unterstützen. SE16XXL verwendet jetzt den neuen Parameter **wenn er auf dem System verfügbar ist**.

ANMERKUNG: Wenn Sie sich nicht sicher sind, wie sich das System verhält, haben Sie mehrere Möglichkeiten, dieses Problem zu lösen.

Am einfachsten ist es, Dateinamen zu verwenden, die kürzer als 50 Zeichen sind, um das Problem gänzlich zu vermeiden.

Ansonsten können Sie einen Blick auf die Parameter der Methode **ADD_ATTACHMENT** der Klasse **CL_DOCUMENT_BCS** werfen.

Wenn das alles nicht möglich ist, machen Sie einfach einen Test, indem Sie eine E-Mail mit dem gewünschten Dateianhang an sich selbst senden und prüfen Sie dann, ob der Name abgeschnitten wird.

WICHTIG: Diese Korrektur steht auch für die Version **4.0** von SE16XXL in Form eines **Support Packages** zur Verfügung.

Neuer Parameter

“Liste ohne Symbole ausgeben (für Tabellenkalkulation)”

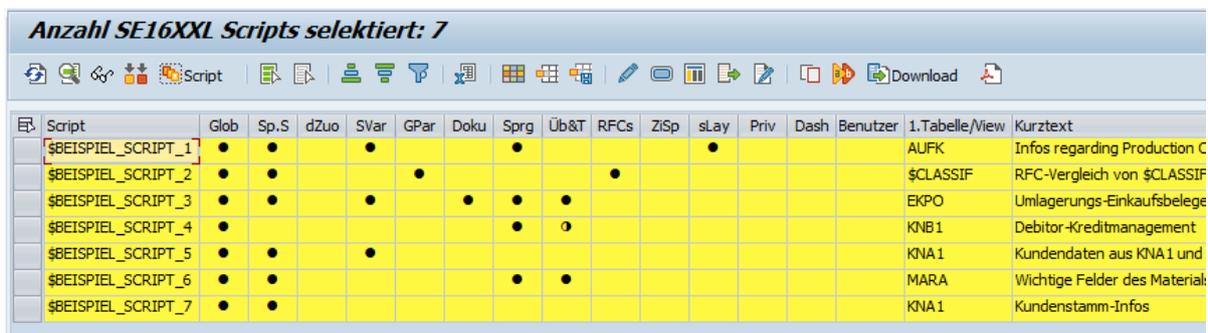
Dieser neue Parameter ist jetzt für den **Script-Katalog**, für den **Katalog der Script-Versionen** und für die **Liste von Script-Statistiken** verfügbar.

Zweck des neuen Parameters

Wir werden dieses Thema anhand des Script-Katalogs diskutieren. Die anderen Programme verhalten sich auf ähnliche Weise.

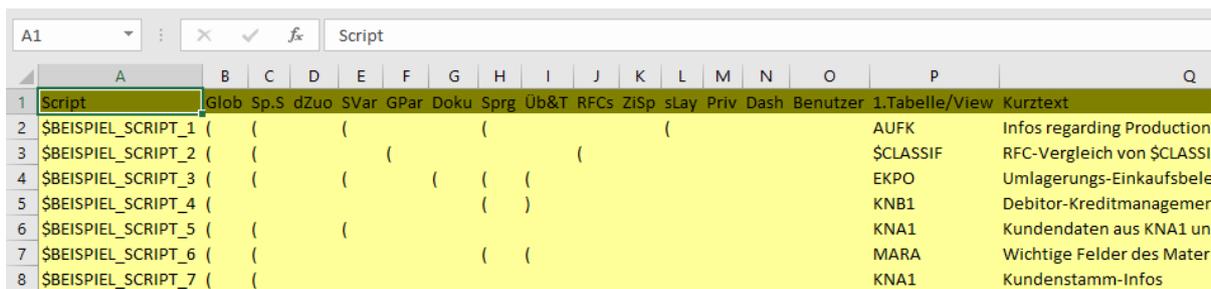
In der Ergebnisliste des Script-Katalogs sind die Hauptmerkmale der Scripts durch Symbole gekennzeichnet, wie in der folgenden Abbildung zu sehen ist:

Anzahl SE16XXL Scripts selektiert: 7



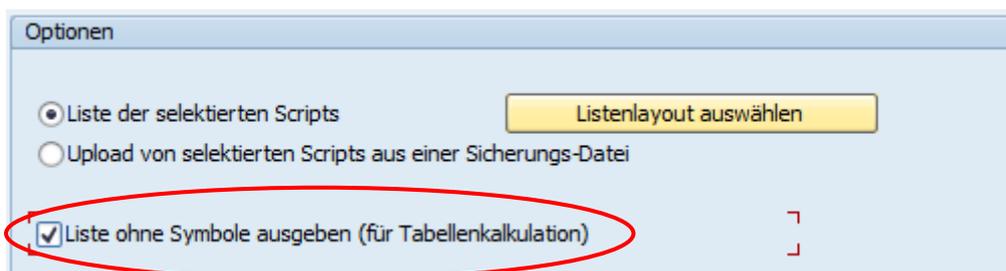
Script	Glob	Sp.S	dZuo	SVar	GPar	Doku	Sprg	Üb&T	RFCs	ZiSp	sLay	Priv	Dash	Benutzer	1.Tabelle/View	Kurztext
\$BEISPIEL_SCRIPT_1	●	●		●			●				●				AUFK	Infos regarding Production C
\$BEISPIEL_SCRIPT_2	●	●			●				●						\$CLASSIF	RFC-Vergleich von \$CLASSIF
\$BEISPIEL_SCRIPT_3	●	●		●		●	●	●							EKPO	Umlagerungs-Einkaufsbelege
\$BEISPIEL_SCRIPT_4	●	●					●	○							KNB1	Debitor-Kreditmanagement
\$BEISPIEL_SCRIPT_5	●	●		●											KNA1	Kundendaten aus KNA1 und
\$BEISPIEL_SCRIPT_6	●	●					●	●							MARA	Wichtige Felder des Material
\$BEISPIEL_SCRIPT_7	●	●													KNA1	Kundenstamm-Infos

Wenn diese Liste in eine Tabellenkalkulation exportiert wird, sieht das Ergebnis wie folgt aus:



Script	Glob	Sp.S	dZuo	SVar	GPar	Doku	Sprg	Üb&T	RFCs	ZiSp	sLay	Priv	Dash	Benutzer	1.Tabelle/View	Kurztext
\$BEISPIEL_SCRIPT_1	(((((AUFK	Infos regarding Production
\$BEISPIEL_SCRIPT_2	((((\$CLASSIF	RFC-Vergleich von \$CLASSIF
\$BEISPIEL_SCRIPT_3	((((((EKPO	Umlagerungs-Einkaufsbelege
\$BEISPIEL_SCRIPT_4	((()							KNB1	Debitor-Kreditmanagemer
\$BEISPIEL_SCRIPT_5	(((KNA1	Kundendaten aus KNA1 un
\$BEISPIEL_SCRIPT_6	((((MARA	Wichtige Felder des Mater
\$BEISPIEL_SCRIPT_7	((KNA1	Kundenstamm-Infos

Wie zu sehen ist, werden die Symbole (●) als Klammer (()) dargestellt. Aus diesem Grund bietet die Selektionsmaske des Script-Katalogs jetzt eine neue Option:



Optionen

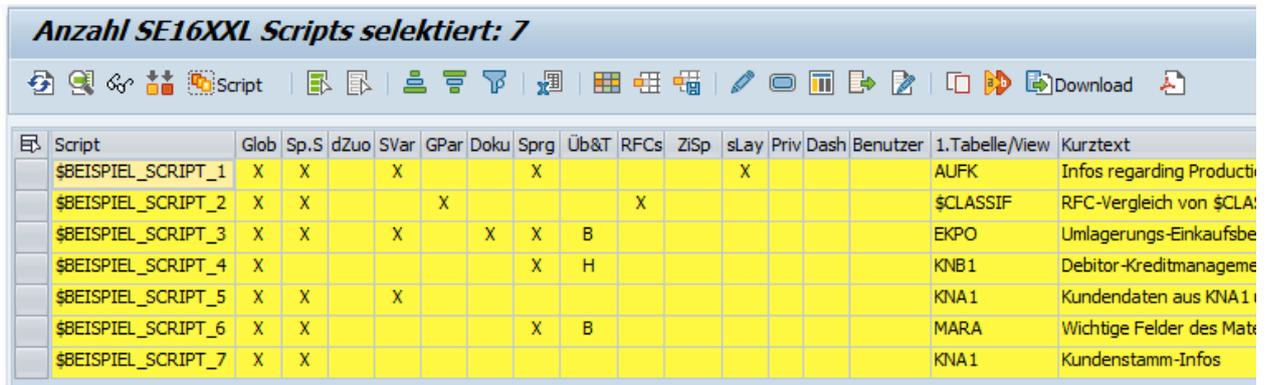
Liste der selektierten Scripts Listenlayout auswählen

Upload von selektierten Scripts aus einer Sicherungs-Datei

Liste ohne Symbole ausgeben (für Tabellenkalkulation)

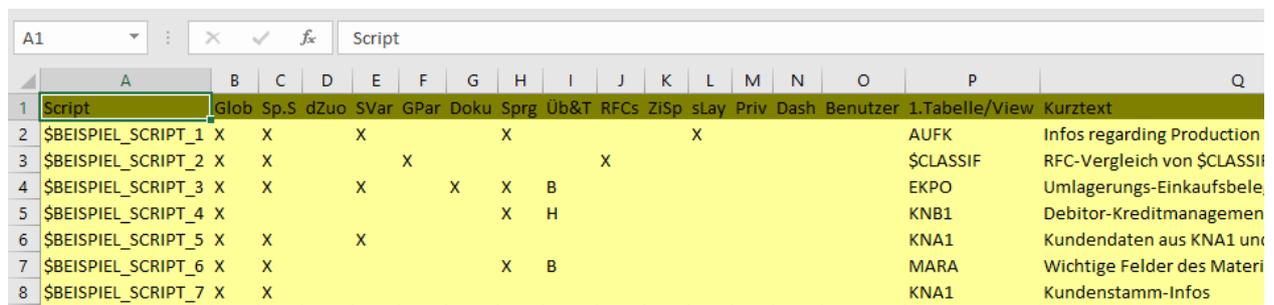
Wenn diese Option aktiviert wird, wird die Liste der Scripts ohne Symbole erstellt:

Anzahl SE16XXL Scripts selektiert: 7



Script	Glob	Sp.S	dZuo	SVar	GPar	Doku	Sprg	Üb&T	RFCs	ZiSp	sLay	Priv	Dash	Benutzer	1.Tabelle/View	Kurztext
\$BEISPIEL_SCRIPT_1	X	X		X			X				X				AUFK	Infos regarding Production
\$BEISPIEL_SCRIPT_2	X	X			X				X						\$CLASSIF	RFC-Vergleich von \$CLASSIF
\$BEISPIEL_SCRIPT_3	X	X		X		X	X	B							EKPO	Umlagerungs-Einkaufsbeleg
\$BEISPIEL_SCRIPT_4	X						X	H							KNB1	Debitor-Kreditmanagement
\$BEISPIEL_SCRIPT_5	X	X		X											KNA1	Kundendaten aus KNA1 und
\$BEISPIEL_SCRIPT_6	X	X					X	B							MARA	Wichtige Felder des Materials
\$BEISPIEL_SCRIPT_7	X	X													KNA1	Kundenstamm-Infos

Die Anzeige der Tabellenkalkulation sieht dementsprechend wie in der folgenden Abbildung aus:



Script	Glob	Sp.S	dZuo	SVar	GPar	Doku	Sprg	Üb&T	RFCs	ZiSp	sLay	Priv	Dash	Benutzer	1.Tabelle/View	Kurztext
\$BEISPIEL_SCRIPT_1	X	X		X			X				X				AUFK	Infos regarding Production
\$BEISPIEL_SCRIPT_2	X	X			X				X						\$CLASSIF	RFC-Vergleich von \$CLASSIF
\$BEISPIEL_SCRIPT_3	X	X		X		X	X	B							EKPO	Umlagerungs-Einkaufsbeleg
\$BEISPIEL_SCRIPT_4	X						X	H							KNB1	Debitor-Kreditmanagement
\$BEISPIEL_SCRIPT_5	X	X		X											KNA1	Kundendaten aus KNA1 und
\$BEISPIEL_SCRIPT_6	X	X					X	B							MARA	Wichtige Felder des Materials
\$BEISPIEL_SCRIPT_7	X	X													KNA1	Kundenstamm-Infos

ANMERKUNG: Diese Option ist nur für die normale Liste des Script-Katalogs verfügbar. In der Liste der hochgeladenen Scripts wird die Funktion **“Liste → Export → Tabellenkalkulation ...” nicht unterstützt**, da es sich bei dieser Liste um eine **hierarchisch-sequenzielle Liste** handelt.

[Zum Anfang](#)

Neue Operation “Zeilen in Spalten umwandeln”

Bisher bestand die Ergebnisliste von SE16XXL immer aus einer Reihe von Zeilen, wobei jede Zeile einige oder alle Felder einer bestimmten Datenbanktabelle oder View repräsentierte, oder aus einem Join von zwei oder mehr solcher Tabellen/Views. Daraus folgt, dass eine solche Liste sehr lang werden kann, was es schwierig macht, auf einen Blick eine Gesamtvorstellung der selektierten Daten zu erhalten.

Aus diesem Grund wurde ab Version 4.0 von SE16XXL eine neue Operation implementiert (“*Zeilen in Spalten umwandeln*”), die es ermöglicht, eine kompakte Ergebnisliste zu erstellen. Wie der Name schon sagt, wird eine Reihe von Zeilen in eine Zeile mit einer Reihe von Spalten umgewandelt, die von den Werten der ursprünglichen Zeilen abgeleitet sind. Die resultierende Liste enthält mehr Informationen pro Zeile als die ursprüngliche und eignet sich daher perfekt für die neueren Monitore, die in der Regel sehr breit sind.

Ein paar Beispiele werden eine Vorstellung davon vermitteln, welche Art von Ergebnislisten mithilfe dieser neuen Funktionalität erstellt werden können.

Beispiel 1 – Einträge der Pseudo-Tabelle \$CLASSIF

In diesem Beispiel werden die Merkmale eines Objekts als Spalten aufgelistet:

SE16XXL - Zeilen in Spalten - 90 Einträge

Zeilen in Spalten umgewandelt

Objekt	Klasse	Preisklasse	Hersteller	Bildschirmdurchmesser (Zoll)	Punkteabstand (Pitch)	Horizontalfrequenz (Khz)	Bildwiederholffrequenz
DPC2000	CL126	gehoben	PAQ	17 Zoll			
M-01		Standard	Sunny	14 Zoll	0,2800 mm	48	1024 x 768
M-02		gehoben	Sunny	15 Zoll	0,2800 mm	50	
M-03		gehoben	Sunny	17 Zoll	0,2800 mm	64	1024 x 768
M-04		exklusiv	Sunny	21 Zoll	0,2800 mm	85	1024 x 768
M-05		günstig	FLATSCREEN	14 Zoll	0,2800 mm	50	1024 x 768
M-06		günstig	FLATSCREEN	14 Zoll	0,2800 mm	60	1024 x 768
M-07		Standard	FLATSCREEN	15 Zoll	0,2800 mm	64	1024 x 768
M-08		Standard	FLATSCREEN	15 Zoll	0,2800 mm	64	1024 x 768
M-09		Standard	FLATSCREEN	15 Zoll	0,2800 mm	64	1024 x 768
M-10		gehoben	FLATSCREEN	17 Zoll	0,2800 mm	64	1024 x 768
M-11		gehoben	FLATSCREEN	17 Zoll	0,2600 mm	85	1024 x 768
M-12		Standard	MAG	15 Zoll	0,2800 mm	64	1024 x 768
M-13		gehoben	MAG	17 Zoll	0,2600 mm	64	1024 x 768
M-14		exklusiv	MAG	17 Zoll	0,2600 mm	68	1024 x 768
M-15		Standard	SEC	15 Zoll	0,2800 mm	64	1024 x 768
M-16		exklusiv	SFC	17 Zoll	0,2800 mm	65	1024 x 768

Beispiel 2 – Fakturierte Umsätze pro Auftraggeber und Jahr

In diesem Beispiel werden die Einträge der Tabelle S001 in kompakterer Form angezeigt:

Script \$R2C_S001_BEISPIEL - 360 Einträge

Fakturierter Umsatz pro Auftraggeber und Jahr

Kunde	Jahr	Währ.	Januar	Februar	März	April	Mai	Juni	Juli	August	September	Oktober	November	Dezember
0000001033	2017	EUR	86.764,18	46.816,44	44.812,69	46.466,21	88.238,24	13.264,95	7.586,55	11.206,50	66.292,06	209.625,08	179.166,90	171.334,42
0000001033	2018	EUR	59.136,52			164.581,79	141.494,92	85.484,43	90.508,89	85.949,19	292.122,52	54.296,13	117.551,63	140.647,70
0000001033	2019	EUR	438.731,39	234.946,80	168.412,38	82.972,96	73.992,62	366.796,19	196.443,46	60.490,94	354.241,43	205.755,60	226.453,72	83.222,98
0000001033	2020	EUR		117.556,23	72.552,32	187.269,84	210.635,90	231.498,65	209.728,87	223.071,90	89.043,40	325.079,20		201.270,00
0000001033	2021	EUR	70.473,50	242.620,80	104.009,12	204.151,81	529.693,31	429.487,21	379.646,91	345.038,51	269.901,71	317.472,01	343.938,91	86.700,51
0000001033	2022	EUR	410.575,00	356.949,11	56.086,11	244.671,11	175.813,51	358.296,31	338.658,71	33,91	413.152,31		79.540,31	176.834,91
0000001033	2023	EUR	260.329,71	192.175,31	399.839,41	33,91	295.220,70	425.845,61	374.807,41	200.198,91	115.860,91	44.437,71	269.486,71	33,91
0000001033	2024	EUR	315.192,11	345.270,91	179.400,01	112.141,81	170.259,61	232.492,31	33,91					
0000001034	2019	EUR		10.693,67	291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95
0000001034	2020	EUR		291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95	291,95
0000001034	2021	EUR	291,95	291,95	291,95	291,95	291,95	291,95	291,95					
0000001171	2017	EUR	106.844,59	97.285,05	132.636,33	87.226,40	152.635,45	87.226,40	87.226,40	87.310,51	91.407,06	91.776,89	91.776,89	91.776,89
0000001171	2018	EUR	91.776,89	91.776,89	91.776,89	91.776,89	91.776,89	89.527,21	94.026,58	4.517,51	4.243,72			306,78
0000001171	2019	EUR		10.401,72		25.506,59								
0000001172	2017	EUR	16.966,71	12.645,27	59.271,39	59.976,38	84.192,40	15.519,51	9.539,17	9.366,83	115.563,94	113.564,47	57.761,66	121.155,30
0000001172	2018	EUR	100.045,51	4.601,62	4.601,62	73.976,26	28.863,95	4.601,62	69.988,04	8.037,50	170.294,94	97.145,96	34.962,13	159.801,00
0000001172	2019	EUR	165.490,55	296.826,99	123.516,07	119.378,57	120.966,84	144.002,74	111.082,96	36.447,95	197.842,95	171.109,24	137.267,64	37.183,59
0000001172	2020	EUR	107.951,28	113.181,72	175.763,38	105.024,50	146.086,71	157.151,88	112.369,72	130.641,71	225.434,81	161.512,11	27.389,61	120.617,51
0000001172	2021	EUR	108.815,01	60.449,21	189.075,41	120.083,81	261.020,41	217.115,71	303.393,11	139.526,41	232.631,61	62.277,01	165.195,11	174.583,01

Beispiel 3 – Tabellen A000 – A999 mit ihren Schlüsselfeldern

In diesem Beispiel werden alle **Annn**-Tabellen mit ihren Schlüsselfeldern angezeigt:

Script \$R2C_ANNN_BEISPIEL - 529 Einträge

Zeilen in Spalten umgewandelt

Tabelle	Schl.001	Schl.002	Schl.003	Schl.004	Schl.005	Schl.006	Schl.007	Schl.008	Schl.009	Schl.010	Schl.011	Schl.012	Schl.013	Schl.014
A000	MANDT	KAPPL	KSCHL	VAKEY	DATBI									
A001	MANDT	KAPPL	KSCHL	LIFNR	MATNR	DATBI								
A002	MANDT	KAPPL	KSCHL	ALAND	TAXK1	TAXM1	DATBI							
A003	MANDT	KAPPL	KSCHL	ALAND	MWSKZ									
A004	MANDT	KAPPL	KSCHL	VKORG	VTWEG	MATNR	DATBI							
A005	MANDT	KAPPL	KSCHL	VKORG	VTWEG	KUNNR	MATNR	DATBI						
A006	MANDT	KAPPL	KSCHL	VKORG	VTWEG	PLTYP	WAERK	MATNR	DATBI					
A007	MANDT	KAPPL	KSCHL	VKORG	VTWEG	SPART	KUNNR	DATBI						
A008	MANDT	KAPPL	KSCHL	WERKS	WERKV	DATBI								
A009	MANDT	KAPPL	KSCHL	WERKV	EANNR	DATBI								
A010	MANDT	KAPPL	KSCHL	WERKV	MATNR	DATBI								
A011	MANDT	KAPPL	KSCHL	ALAND	LLAND	TAXK1	TAXM1	DATBI						
A012	MANDT	KAPPL	KSCHL	WERKV	WAGRP	DATBI								
A013	MANDT	KAPPL	KSCHL	VKORG	ALIFA	DATBI								

Weitere Informationen zu dieser interessanten Operation finden Sie in der folgenden Dokumentation: [Zeilen in Spalten umwandeln](#).

[Zum Anfang](#)

Verbesserungen der Hintergrund-E-Mail-Parameter

Wenn die Ausführung eines Scripts im Hintergrund eingeplant wird, kann angegeben werden, dass die Ergebnisliste in einer Server-Datei gespeichert werden soll. Dabei kann auch festgelegt werden, dass eine Kopie der Server-Datei als **E-Mail-Anhang** an einen oder mehrere Empfänger gesendet wird. Bisher boten die Parameter zur Definition der E-Mail nicht viel Spielraum für Kreativität. Beginnend mit **Version 4.0** von SE16XXL wurden einige Verbesserungen implementiert:

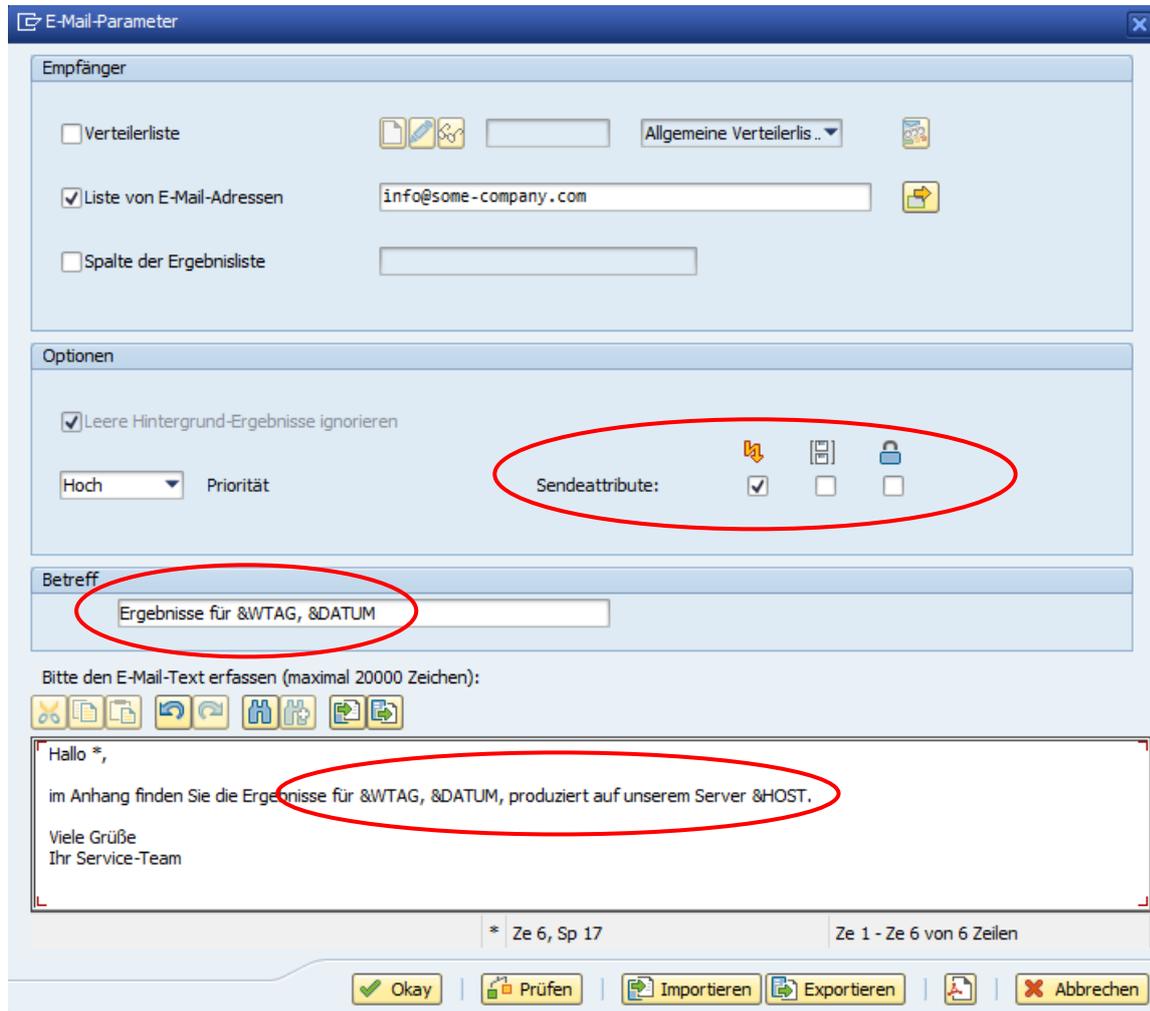
- Es ist nun möglich, die **Sendeattribute** (Expressmail, Als Kopie senden usw.) anzugeben.
- Sowohl der Betreff als auch der Text können **Ersetzungsvariablen** wie &DATUM enthalten.
- Der Text kann bis zu **20000 Zeichen** lang sein (das alte Limit lag bei 2000).

Ein Beispiel soll nun diese Verbesserungen verdeutlichen.

Wir planen die Ausführung des Scripts \$BEISPIEL_SCRIPT im Hintergrund ein. Die erzeugte Server-Datei soll an einen E-Mail-Empfänger gesendet werden. Der relevante Teil des Dialogfensters ist wie folgt:

The screenshot shows the configuration options for a background job. The 'Start verzögert' option is selected. The date is set to 03.09.2024 and the time to 06:15:00. The 'Jobausführung kalenderbezogen' section is expanded, showing 'Job immer ausführen' checked. The 'periodisch alle' checkbox is checked, with '1' entered in the 'Tag(e)' field. Other options include 'mit reduziertem Speicherbedarf' checked, 'Ergebnisliste in Server-Datei speichern' checked, and 'keine leeren Server-Dateien erzeugen' checked. The 'E-Mail-Parameter' button is highlighted with a red circle.

Wenn wir  betätigen, erhalten wir folgendes Definitionsfenster:



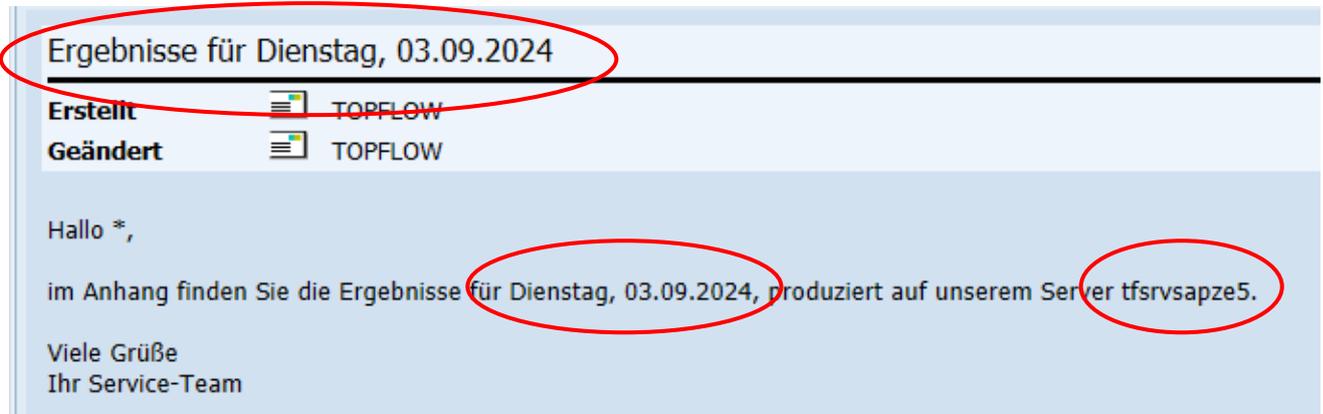
Wir haben die notwendigen Parameter bereits ausgefüllt.

ANMERKUNG: Um eine Liste der verfügbaren Ersetzungsvariablen zu erhalten, setzen Sie den Cursor auf den E-Mail-Betreff oder -Text und drücken Sie **F4**.

Der Job-Log des Hintergrund-Jobs sieht dann wie folgt aus:

Datum	Uhrzeit	Nachrichtentext
03.09.2024	06:15:55	Job wurde gestartet
03.09.2024	06:15:55	Step 001 gestartet (Programm /TFTO/TX_BATCH_SCRIPT_PER_X, Variante &0000000000505, Benutzername TOPFLOW)
03.09.2024	06:15:55	Ziel-Server-Dateipfad ist \usr\sap\ZES\tmp
03.09.2024	06:15:55	Globales Script \$BEISPIEL_SCRIPT wird ausgeführt
03.09.2024	06:15:55	Option "mit reduziertem Speicherbedarf" ist aktiv
03.09.2024	06:15:55	Datei \usr\sap\ZES\tmp\ERGEBNISSE_20240903_061555.txt geschrieben
03.09.2024	06:15:55	*****
03.09.2024	06:15:55	Folgende E-Mail-Empfänger sind direkt angegeben worden
03.09.2024	06:15:55	*****
03.09.2024	06:15:55	E-Mails an 1 Empfängern versendet
03.09.2024	06:15:55	E-Mail mit Anhang gesendet an: info@some-company.com
03.09.2024	06:15:55	Job wurde beendet

Der Inhalt der E-Mail lautet dann wie folgt:



Wie man sieht, wurden die Ersetzungsvariablen durch die echten Werte zum Zeitpunkt der Ausführung des Jobs ersetzt.

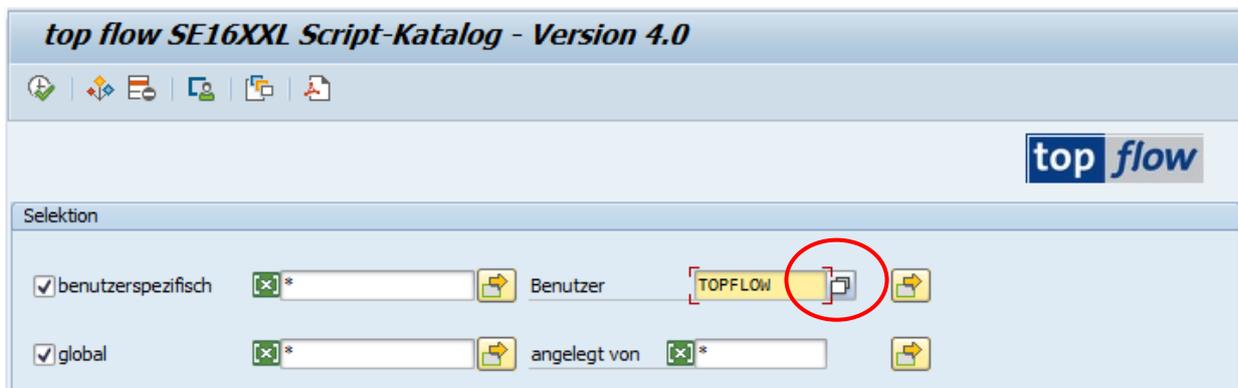
ANMERKUNG: In diesem Beispiel haben wir einen kurzen E-Mail-Text angegeben. Sie können jedoch einen sehr langen Text eingeben, bis zu **20000** Zeichen. Dieses Limit sollte für die meisten Situationen ausreichend sein.

[Zum Anfang](#)

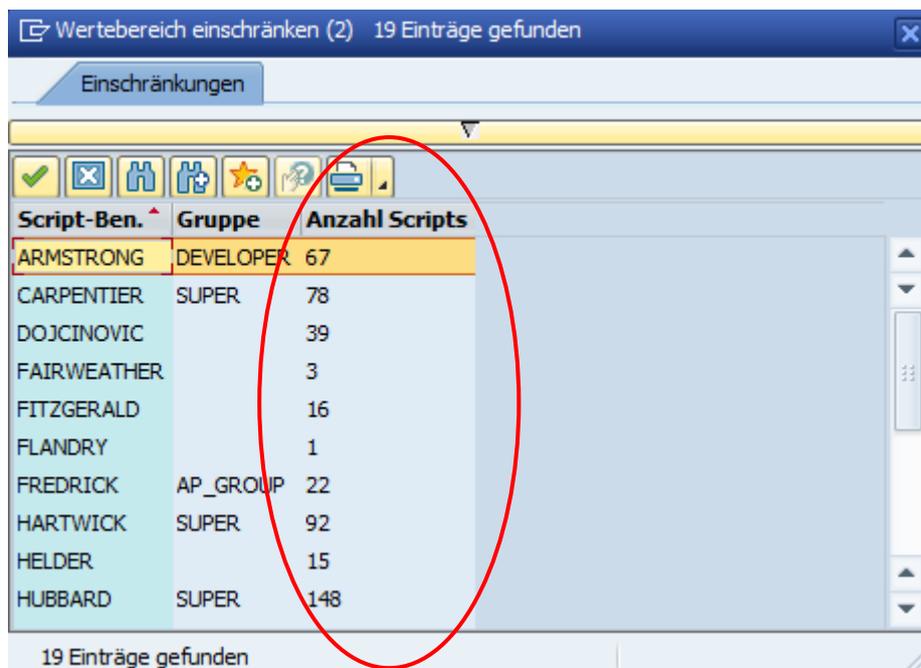
Script-Katalog – Wertehilfe für Script-Ersteller

Die Selektionsmaske des **Script-Katalogs** bietet nun eine **F4-Hilfe** für die **Benutzer** von benutzerspezifischen Scripts und für die **Ersteller** von globalen Scripts. Auf diese Weise ist es einfacher, die gewünschten Scripts zu selektieren. Das Programm untersucht die vorhandenen Scripts und schlägt nur die Logon-Kennungen vor, die sie angelegt haben. Auf diese Weise wird die Anzahl der vorgeschlagenen Kennungen auf ein Minimum reduziert und irrelevante Namen werden nicht angezeigt.

Machen wir ein Beispiel. Wir klicken auf die Schaltfläche für die Wertehilfe () für die Benutzer der benutzerspezifischen Scripts, wie unten dargestellt:

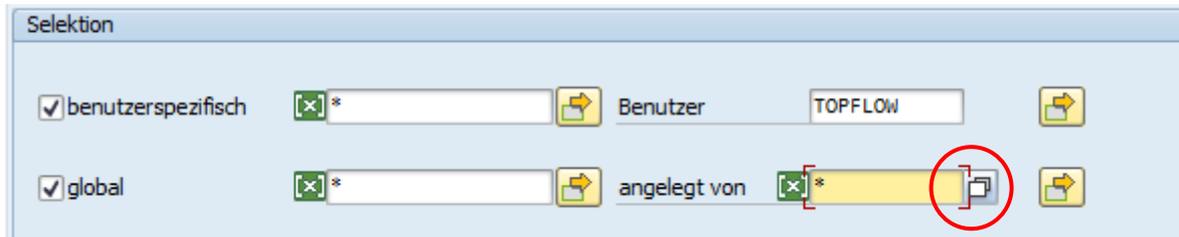


Wir bekommen folgende Vorschläge:



Jede Zeile zeigt auch die Anzahl der Scripts an, die von jeder Kennung erstellt wurden.

Nun wiederholen wir die Prozedur für die Ersteller von globalen Scripts:

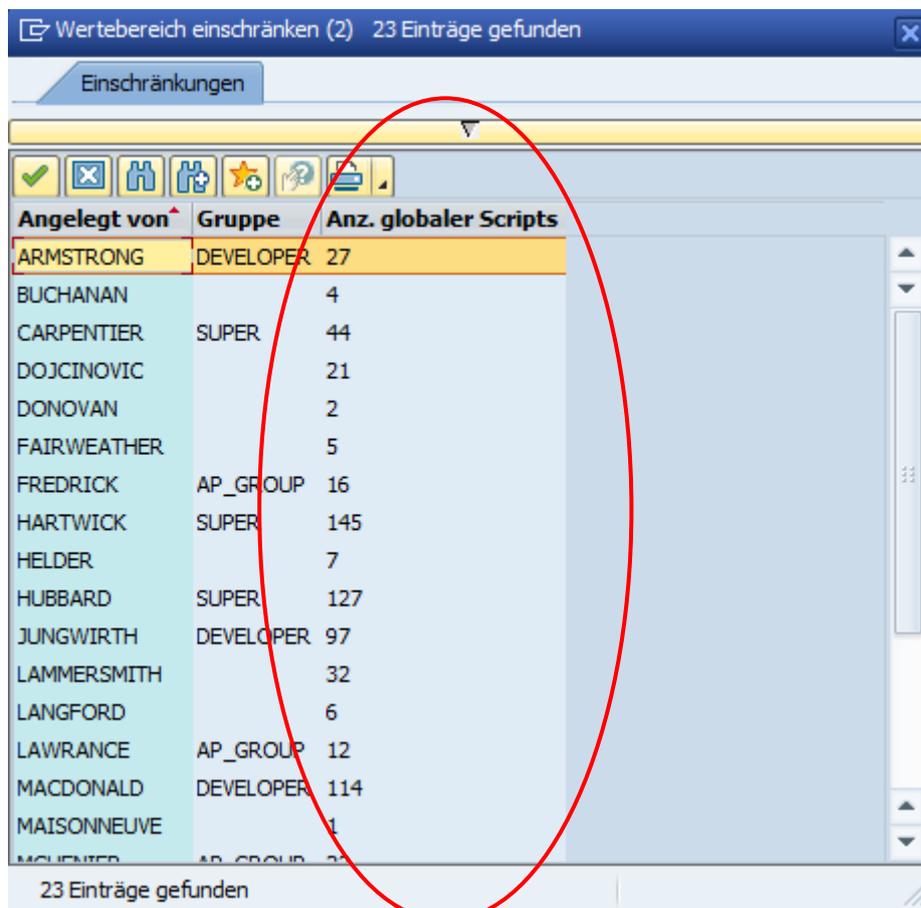


Selektion

benutzerspezifisch Benutzer TOPFLOW

global angelegt von

Diesmal erhalten wir:



Wertebereich einschränken (2) 23 Einträge gefunden

Einschränkungen

Angelegt von	Gruppe	Anz. globaler Scripts
ARMSTRONG	DEVELOPER	27
BUCHANAN		4
CARPENTIER	SUPER	44
DOJCINOVIC		21
DONOVAN		2
FAIRWEATHER		5
FREDRICK	AP_GROUP	16
HARTWICK	SUPER	145
HELDER		7
HUBBARD	SUPER	127
JUNGWIRTH	DEVELOPER	97
LAMMERSMITH		32
LANGFORD		6
LAWRANCE	AP_GROUP	12
MACDONALD	DEVELOPER	114
MAISONNEUVE		1
MCHENTER	AP_GROUP	2

23 Einträge gefunden

In diesem Fall wird die Anzahl der erstellten globalen Scripts angezeigt.

[Zum Anfang](#)

Klonen von FLTP-Feldern zu anderen Typen

Bisher haben die Menüfunktionen

Extras → *Spalten klonen* → *Normales Klonen* und
Extras → *Spalten klonen* → *Spezial-Klonen*

auch Felder vom Typ FLTP (Gleitpunktzahl) akzeptiert. Die resultierenden V-Felder hatten jedoch **den gleichen Typ** wie die ursprünglichen Felder, nämlich ebenfalls FLTP. Das Problem ist nun, dass die Werte der FLTP-Felder in **wissenschaftlicher Notation** dargestellt werden, d.h. als Mantisse und Exponent. Solche Werte sind **schwer zu interpretieren**, insbesondere in Bezug auf andere numerische Werte, die normal angezeigt werden. Aus diesem Grund, beginnend mit der Version 4.0 von SE16XXL, bieten die oben aufgeführten Menüfunktionen nun die Möglichkeit, den Typ der resultierenden V-Felder von FLTP auf **CURR, QUAN** oder **DEC** zu ändern, abhängig von der Währung oder Einheit des ursprünglichen Feldes. Zusätzlich kann auch die Anzahl der **Nachkommastellen** festgelegt werden. Damit können Tabelleneinträge mit FLTP-Werten in benutzerfreundlichere Felder umgewandelt werden.

Ein Beispiel wird die neue Funktionalität verdeutlichen.

Angenommen, Sie haben eine Liste von Einträgen der Tabelle VBFA (Vertriebsbelegfluß):

SE16XXL - Tabelle VBFA - 2000 Einträge selektiert

Tabelle VBFA - Vertriebsbelegfluß

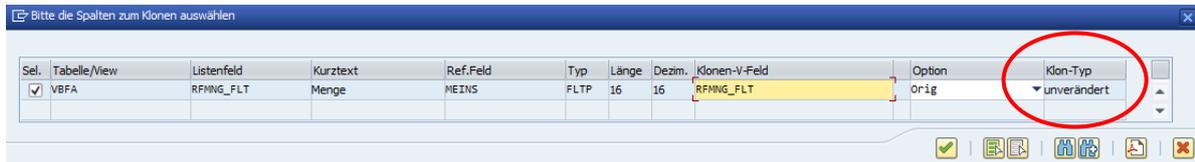
VBELV	POSNV	VBELN	POSNN	VBTYP_N	RFMNG	MEINS	BWART	RFMNG_FLT
0000001536	000010	0049003181	000001	R	4	ST	601	4.0000000000000000E+00
0000001536	000010	0080001206	000010	J	4	ST	601	4.0000000000000000E+00
0000001536	000010	0090001412	000010	M	4	ST	601	4.0000000000000000E+00
0000002465	000010	0049004240	000001	R	4	ST	601	4.0000000000000000E+00
0000002465	000010	0080002128	000010	J	4	ST	601	4.0000000000000000E+00
0000002465	000010	0090002332	000010	M	4	ST	601	4.0000000000000000E+00
0000002871	000010	0049004721	000001	R	1,667	ST	601	1.6670000000000000E+00
0000002871	000010	0080002536	000010	J	1,667	ST	601	1.6670000000000000E+00
0000002871	000010	0090002739	000010	M	1,667	ST	601	1.6670000000000000E+00
0000003313	000010	0049005257	000001	R	15	ST	601	1.5000000000000000E+01
0000003313	000010	0080002969	000010	J	15	ST	601	1.5000000000000000E+01
0000003313	000010	0090003273	000010	M	15	ST	601	1.5000000000000000E+01
0000003314	000010	0049005258	000001	R	5	ST	601	5.0000000000000000E+00

Diese Liste enthält ein FLTP-Feld (**RFMNG_FLT**), das wir nun klonen werden. Dabei werden wir den Typ von **FLTP** in **QUAN** ändern.

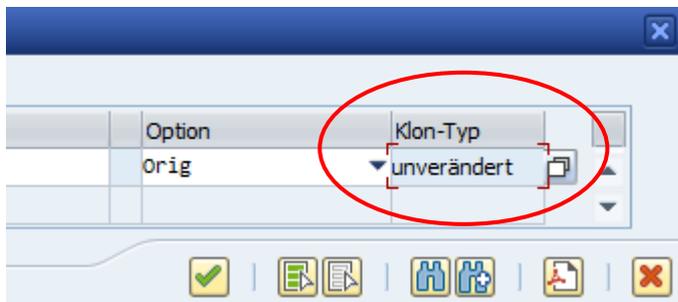
Wir markieren die FRMNG_FLT-Spalte und verwenden dann die Menüfunktion

Extras → Spalten klonen → Normales Klonen.

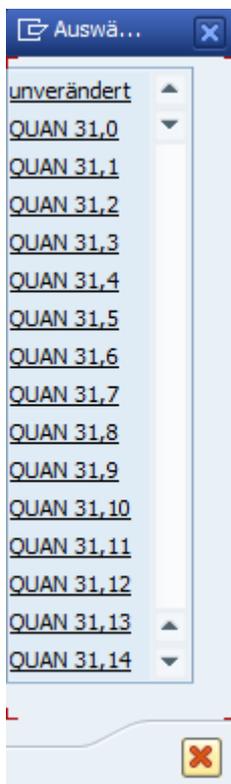
Es erscheint folgendes Dialogfenster:



Um den **Klon-Typ** des V-Feldes zu ändern, verwenden wir die Werthilfe:



Folgende Vorschläge werden angeboten:



Nachdem das andere referenzierte Mengenfeld (**RFMNG**) drei Nachkommastellen hat, wählen wir für unser Klon-V-Feld den Typ **“QUAN 31,3”**.

ANMERKUNG: Alle vorgeschlagenen Typen haben **31 Ziffern**. Sie unterscheiden sich nur in der Anzahl der Nachkommastellen. Es wurden 31 Ziffern ausgewählt, weil FLTP-Felder sehr große Zahlen enthalten können.

Die resultierende Liste sieht nun wie folgt aus:

SE16XXL - Tabelle VBFA - 2000 Einträge selektiert

Tabelle VBFA - Vertriebsbelegfluß

VBELV	POSNV	VBELN	POSNN	VBTYP_N	RFMNG	MEINS	BWART	RFMNG_FLT	V~RFMNG_FLT
0000001536	000010	0049003181	000001	R	4	ST	601	4.0000000000000000E+00	4
0000001536	000010	0080001206	000010	J	4	ST	601	4.0000000000000000E+00	4
0000001536	000010	0090001412	000010	M	4	ST		4.0000000000000000E+00	4
0000002465	000010	0049004240	000001	R	4	ST	601	4.0000000000000000E+00	4
0000002465	000010	0080002128	000010	J	4	ST	601	4.0000000000000000E+00	4
0000002465	000010	0090002332	000010	M	4	ST		4.0000000000000000E+00	4
0000002871	000010	0049004721	000001	R	1,667	ST	601	1.6670000000000000E+00	1,667
0000002871	000010	0080002536	000010	J	1,667	ST	601	1.6670000000000000E+00	1,667
0000002871	000010	0090002739	000010	M	1,667	ST		1.6670000000000000E+00	1,667
0000003313	000010	0049005257	000001	R	15	ST	601	1.5000000000000000E+01	15
0000003313	000010	0080002969	000010	J	15	ST	601	1.5000000000000000E+01	15
0000003313	000010	0090003273	000010	M	15	ST		1.5000000000000000E+01	15
0000003314	000010	0049005258	000001	R	5	ST	601	5.0000000000000000E+00	5
0000003314	000010	0080002970	000010	J	5	ST	601	5.0000000000000000E+00	5

Wie man sehen kann, ist es nun viel einfacher, die beiden Mengenfelder **RFMNG** und **V~RFMNG_FLT** miteinander zu vergleichen, da sie auf die gleiche Weise dargestellt werden.

ANMERKUNG: In diesem Beispiel wurden Klon-Typen vom Typ **QUAN** vorgeschlagen, denn das Referenzfeld des Originalfeldes **RFMNG_FLT** ist **MEINS**, das vom Typ **UNIT** ist. In anderen Situationen, wenn der Referenztyp z. B. **CUKY** ist, wird der Klon-Typ **CURR** vorgeschlagen. Für den Fall, dass das Originalfeld überhaupt kein Referenzfeld hat, ist der vorgeschlagene Klon-Typ **DEC**.

[Zum Anfang](#)

Verbesserungen für datumsbezogene Felder

Die Analysetabellen **S000 – S999** enthalten häufig die Schlüsselfelder **SPMON** (Monat) und **SPWOC** (Woche). Bisher wurden diese Felder von der Menüfunktion

Extras → Spalten einfügen → mit datumsbezogenen Feldern

ignoriert. Ab Version 4.0 von SE16XXL werden diese Felder berücksichtigt.

Darüber hinaus steht nun ein neues abgeleitetes V-Feld zur Verfügung, nämlich **“Woche (WW)”**. Dieses abgeleitete Feld ist vom Typ **NUMC(2)** und repräsentiert die Woche des Jahres von **01** bis **52** (oder möglicherweise **53**).

Nun ein paar Beispiele, um zu veranschaulichen, was besprochen wurde.

Beispiel #1 - Liste der Tabelle S001 mit SPMON

Nehmen wir eine Liste mit Einträgen der Tabelle **S001** (VIS: Kundenstatistik):

SE16XXL - Tabelle S001 - 1529 Einträge selektiert

Tabelle S001 - VIS: Kundenstatistik

MANDT	SSOUR	VRSIO	SPMON	SPTAG	SPWOC	SPBUP	KUNNR	VKORG	VTWEG	SPART	MATNR	STWAE	BASME	AENETWR	UMNETWR
800		000	202401		000000	000000	0000001032	1000	10	00	P-102	EUR	ST	72.890,40	0,00
800		000			000000	000000		1000	10	01	P-102	EUR	ST	0,00	0,00
800		000			000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE	0,00	255,70
800		000	202402		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE	0,00	255,70
800		000	202403		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE	0,00	255,70
800		000	202404		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE	0,00	255,70
800		000	202405		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE	0,00	255,70
800		000	202406		000000	000000		1000	10	00	P-101	EUR	ST	104.488,80	104.488,80
800		000			000000	000000		1000	10	00	P-103	EUR	ST	96.378,60	96.378,60
800		000			000000	000000		1000	10	00	P-104	EUR	ST	85.385,00	85.385,00
800		000			000000	000000		1000	10	00	P-402	EUR	ST	207.051,00	207.051,00
800		000			000000	000000		1000	10	01	P-101	EUR	ST	0,00	0,00

Die Liste ist nach **KUNNR** und **SPMON** sortiert, um unterschiedliche **SPMON**-Werte zu zeigen. Nun markieren wir die **SPMON**-Spalte und nutzen die Menüfunktion *Extras → Spalten einfügen → mit datumsbezogenen Feldern*. Es erscheint folgendes Dialogfenster:

Bitte die Spalten zum Hinzufügen auswählen

Sel.	Tabelle/View	Listenfeld	Kurztext	Typ	Abgeleiteter Typ	Abgeleitetes V-Feld
<input checked="" type="checkbox"/>	S001	SPMON	Monat	Jahr + Monat (JJJJMM)	→ Jahr (JJJJ)	SPMON_YYYY
<input checked="" type="checkbox"/>	S001	SPMON	Monat	Jahr + Monat (JJJJMM)	→ Monat (MM)	SPMON_MON

Wenn wir mit  bestätigen, erhalten wir folgende Ergebnisliste:

SE16XXL - Tabelle S001 - 1529 Einträge selektiert

Tabelle S001 - VIS: Kundenstatistik

MANDT	SSOUR	VRSIO	SPMON	V~SPMON_YYYY	V~SPMON_MON	SPTAG	SPWOC	SPBUP	KUNNR	VKORG	VTWEG	SPART	MATNR	STWAE	BASME
800	000	000	202401	2024	01		000000	000000	0000001032	1000	10	00	P-102	EUR	ST
800	000	000		2024	01		000000	000000		1000	10	01	P-102	EUR	ST
800	000	000		2024	01		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE
800	000	000	202402	2024	02		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE
800	000	000	202403	2024	03		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE
800	000	000	202404	2024	04		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE
800	000	000	202405	2024	05		000000	000000		1000	14	08	PC_SERVICE_CONF	EUR	LE
800	000	000	202406	2024	06		000000	000000		1000	10	00	P-101	EUR	ST
800	000	000		2024	06		000000	000000		1000	10	00	P-103	EUR	ST
800	000	000		2024	06		000000	000000		1000	10	00	P-104	EUR	ST

Die beiden Spalten V~SPMON_YYYY und V~SPMON_MON können dann z.B. für eine Operation “Zeilen in Spalten umwandeln” verwendet werden.

Beispiel #2 - Liste der Tabelle S002 mit SPWOC

Wir beginnen mit einer Liste der Tabelle S002 (VIS: Verkaufsbürostatistik):

SE16XXL - Tabelle S002 - 568 Einträge selektiert

Tabelle S002 - VIS: Verkaufsbürostatistik

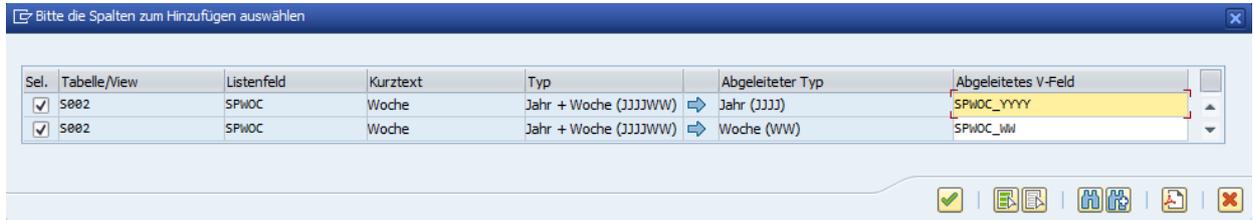
MANDT	SSOUR	VRSIO	SPMON	SPTAG	SPWOC	SPBUP	VKORG	VKGRP	VKBUR	VTWEG	SPART	STWAE	AENETWR	RENETWR	OAUWE	UMN
800	000	000	000000		202408	000000	1000	100	1000	10	00	EUR	85.385,00	0,00	85.385,00	
800	000	000	000000			000000	1000			10	01	EUR	0,00	0,00	85.385,00-	
800	000	000	000000		202409	000000	1000			10	00	EUR	0,00	0,00	0,00	85.3
800	000	000	000000		202417	000000	1000			10	00	EUR	503.692,60	0,00	503.692,60	
800	000	000	000000			000000	1000			10	01	EUR	0,00	0,00	503.692,60-	
800	000	000	000000		202418	000000	1000			10	00	EUR	0,00	0,00	0,00	503.6
800	000	000	000000		202421	000000	1000			10	00	EUR	503.692,60	0,00	503.692,60	
800	000	000	000000			000000	1000			10	01	EUR	0,00	0,00	503.692,60-	
800	000	000	000000		202422	000000	1000			10	00	EUR	0,00	0,00	0,00	503.6
800	000	000	000000		202429	000000	1000			10	00	EUR	442.372,90	0,00	0,00	
800	000	000	000000		202434	000000	1000			10	00	EUR	390.055,60	0,00	390.055,60	
800	000	000	000000			000000	1000			10	01	EUR	0,00	0,00	390.055,60-	
800	000	000	000000		202435	000000	1000			10	00	EUR	0,00	0,00	0,00	390.0
800	000	000	000000		202402	000000	1000	101		10	00	EUR	132.965,15	0,00	132.965,15	
800	000	000	000000			000000	1000			10	07	EUR	0,00	0,00	132.965,15-	
800	000	000	000000			000000	1000			12	04	EUR	26,38	0,00	0,00	

Die Liste wurde nach VKGRP, VKBUR und SPWOC sortiert.

Nun markieren wir die SPWOC-Spalte und verwenden erneut die Menüfunktion

Extras → Spalten einfügen → mit datumsbezogenen Feldern.

Dieses Mal bietet das Dialogfenster das **Jahr** und die **Woche** an:



Die resultierende Liste sieht dann wie folgt aus:

SE16XXL - Tabelle S002 - 568 Einträge selektiert

Tabelle S002 - VIS: Verkaufsbürostatistik

MANDT	SSOUR	VRSIO	SPMON	SPTAG	SPWOC	V~SPWOC_YYYY	V~SPWOC_WW	VBUP	VKORG	VKGRP	VKBUR	VTWEG	SPART	STWAE	AENETWR	RENETWR	O
800	000	000000			202408	2024	08	000000	1000	100	1000	10	00	EUR	85.385,00	0,00	85.38
800	000	000000			202409	2024	09	000000	1000			10	00	EUR	0,00	0,00	85.38
800	000	000000			202417	2024	17	000000	1000			10	00	EUR	503.692,60	0,00	503.69
800	000	000000			202418	2024	18	000000	1000			10	01	EUR	0,00	0,00	503.69
800	000	000000			202421	2024	21	000000	1000			10	00	EUR	503.692,60	0,00	503.69
800	000	000000			202422	2024	22	000000	1000			10	01	EUR	0,00	0,00	503.69
800	000	000000			202429	2024	29	000000	1000			10	00	EUR	442.372,90	0,00	
800	000	000000			202434	2024	34	000000	1000			10	00	EUR	390.055,60	0,00	390.05
800	000	000000			202435	2024	35	000000	1000			10	01	EUR	0,00	0,00	390.05
800	000	000000			202402	2024	02	000000	1000	101		10	00	EUR	132.965,15	0,00	132.96
800	000	000000			202402	2024	02	000000	1000			10	07	EUR	0,00	0,00	132.96

Diese neue Liste kann dann verwendet werden, um eine umgewandelte Ergebnisliste wie folgt zu erstellen:

SE16XXL - Zeilen in Spalten - 18 Einträge

Zeilen in Spalten umgewandelt

Verkäufers	Verk.Büro	Währung	Woche 01	Woche 02	Woche 03	Woche 04	Woche 05	Woche 06	Woche 07	Woche 08	Woche 09	W
100	1000	EUR									85.385,00	
101				26,38		205.714,10		26,38	109.270,60	206.095,70	21.810,60	
103						1.058.460,00				586.400,00		
110	1010		281,21			209.340,30	281,21		210.704,10	435.125,10	26.157,41	1.
111				56.269,66				56.269,66				56.2
130	1030		499.561,70	65,37		405.782,30	255,70	65,37	98.681,65	456.304,20	484.311,00	
131				49.500,00		385.000,00		49.500,00		930.000,00		49.5
240	2400				383.550,00					417.300,00		
250	2500				501.100,00					550.800,00		
301	3000	USD			38.850,00		225.622,00			101.281,00		
302					69.038,00		514.534,60		154.064,85	69.038,00	514.534,60	
303					45.500,00					45.500,00		
321	3020				209.014,40		43.809,70		138.842,20	113.981,90		

[Zum Anfang](#)

GUID-Konversionen in Formeln

In SAP ist eine **GUID** (Globally Unique Identifier) ein vom System generierter eindeutiger Schlüssel der in verschiedenen Datenbanktabellen verwendet wird. Das Problem bei GUIDs ist, dass sie **drei** verschiedene Formate annehmen können, nämlich **X16** (16 Byte hexadezimal), **C22** (22 Zeichen) und **C32** (32 Zeichen). Jedes Format kann in eines der anderen zwei Formate umgewandelt werden. Das Problem tritt auf, wenn eine bestimmte GUID (z. B. als **C32** gespeichert) in einer Tabelle auf dieselbe GUID (als **C22** gespeichert) in einer anderen Tabelle verweist. Dadurch ist es unmöglich, einen Join zwischen den beiden Tabellen durchzuführen. Aus diesem Grund ist es ab Version 4.0 von SE16XXL möglich, in einer Formel **eine GUID** von einem Format in ein anderes zu **konvertieren**. Wir werden nun diese Funktionalität anhand eines Beispiels verdeutlichen.

Nehmen wir eine Liste mit Einträgen der Tabelle **/SCWM/AQUA** (Verfügbare Menge):

SE16XXL - Tabelle /SCWM/AQUA - 280 Einträge selektiert

Tabelle /SCWM/AQUA - Verfügbare Menge

MANDT	GUID_PARENT	GUID_STOCK	LGNUM	LGTyp	LGPLA	MATID	QMAN	UNIT
800	00505691C1DC1ED9979349A7B66D20FB	00505691C1DC1E0EB180DA3A14BA8120	9000	0022	021.01.01.01	00505691C1DC1EDEAFD27F8DAD85C120	2.000	ST
800	00505691C1DC1ED9979349A7B66D40FB	00505691C1DC1E0EB180DA3A14BA8120	9000	0022	021.01.01.02	00505691C1DC1EDEAFD27F8DAD85C120	5.000	ST
800	00505691C1DC1ED9979349A7B67160FB	00505691C1DC1E0EB180DA3A14BA8120	9000	0022	021.03.04.02	00505691C1DC1EDEAFD27F8DAD85C120	3.500	ST
800	00505691C1DC1ED997935684427760FB	00505691C1DC1ED99CAD7E9591B980FB	9000	0041	041.08	00505691C1DC1ED993E7B7053BA6C0FB	1.728	ST
800	00505691C1DC1ED9979357434A8820FB	00505691C1DC1ED99CAD7E9591BA00FB	9000	0041	041.12	00505691C1DC1ED993E7C941D69D80FB	3.840	ST
800	00505691C1DC1ED997935802529FE0FB	00505691C1DC1ED99CAD7E9591BA80FB	9000	0042	042.08	00505691C1DC1ED993E7D50D0B1000FB	48	ST
800	00505691C1DC1ED99793596D0DEB40FB	00505691C1DC1ED99CAD7E9591BA80FB	9000	0052	052.08	00505691C1DC1ED993E7D50D0B1000FB	24	ST
800	00505691C1DC1ED99C9570098824C0FB	00505691C1DC1ED99C9570098824E0FB	9000	0011	011.02.29.05	00505691C1DC1ED992FE7894E04500FB	48	ST
800	00505691C1DC1ED99C957009882520FB	00505691C1DC1ED99C9570098824E0FB	9000	0011	011.02.29.06	00505691C1DC1ED992FE7894E04500FB	48	ST
800	00505691C1DC1ED99C957009882540FB	00505691C1DC1ED99C9570098824E0FB	9000	0011	011.02.30.06	00505691C1DC1ED992FE7894E04500FB	48	ST
800	00505691C1DC1ED99C957009882560FB	00505691C1DC1ED99C957009882580FB	9000	0011	011.02.29.04	00505691C1DC1ED992FE7894E04500FB	192	ST

Diese Liste enthält unter anderem die **X16-GUID** des Produkts (**MATID**). An dieser Stelle möchten wir die Einträge dieser Liste mit den entsprechenden Einträgen der Tabelle **/SAPAPO/MATKEY** joinen. Der Schlüssel dieser Tabelle heißt ebenfalls **MATID**, hat jedoch ein anderes Format, nämlich eine **C22-GUID**. Bisher wäre ein solcher Join unmöglich gewesen. Dank der neuen Funktionen sind wir nun jedoch in der Lage, diese Operation durchzuführen. Wir beginnen mit der Definition einer Formel (🔗):

Formel editieren - inaktiv

* Bitte Formelanweisungen eingeben:

```
FF_DEF MATID_C22 TYPE SYSUID-C22.
MATID_C22 = FF_GUID_X16_TO_C22( ROW-MATID ).
```

top flow

ROW-Feld Weiter

Drag & Drop

- Beispiele
- Formelfelder definieren
- verfügbare Anweisungen
- Spezielle Anweisungen und Funktionen

Nach Aktivierung der Formel (📝) erhalten wir:

GUID_STOCK	LGNUM	LGTYP	LGPLA	MATID	QUAN	UNZT	X~MATID_C22
A3A14BA8120	9000	0022	021.01.01.01	00505691C1DC1EDEAFD27F8DAD85C120	2.000	ST	051MaS7S7jwlqd{DhON180
A3A14BA8120	9000	0022	021.01.01.02	00505691C1DC1EDEAFD27F8DAD85C120	5.000	ST	051MaS7S7jwlqd{DhON180
A3A14BA8120	9000	0022	021.03.04.02	00505691C1DC1EDEAFD27F8DAD85C120	3.500	ST	051MaS7S7jwlqd{DhON180
7E9591B980FB	9000	0041	041.08	00505691C1DC1ED993E7B7053BA6C0FB	1.728	ST	051MaS7S7jcJvxS5EwR0{m
7E9591BA00FB	9000	0041	041.12	00505691C1DC1ED993E7C941D69D80FB	3.840	ST	051MaS7S7jcJvyb1rfs0{m

Dann wandeln wir das Formelfeld in ein V-Feld um, mithilfe der Menüfunktion
Edit → Formula → Fix formula fields.

Dadurch können wir dieses Feld in die nächste Ergebnisliste mitnehmen. Und nun können wir unter Verwendung dieser konvertierten GUID (**V~MATID_C22**) einen Outer-Join (🔗 Outer) mit der Tabelle **/SAPAPO/MATKEY** (Product) durchführen:

Die Selektionskriterien lauten wie folgt:

Das Ergebnis des Joins ist dann wie folgt (wir haben die Liste nach rechts gescrollt, um die /SAPAPO/MATKEY-Felder anzuzeigen):

SE16XXL - Outer-Join - 280 Ergebnis-Sätze

Join von /SCWM/AQUA(A) und /SAPAPO/MATKEY(B)

Ein	A~MATID	A~QUAN	A~UNIT	V~MATID_C22	B~MANDT	B~MATID	B~MATNR
	00505691C1DC1E0EAFD27F8DAD85C120	2.000	ST	051MaS7S7jwlqd{DhON180	800	051MaS7S7jwlqd{DhON180	1010-EWM-507
	00505691C1DC1E0EAFD27F8DAD85C120	5.000	ST	051MaS7S7jwlqd{DhON180	800	051MaS7S7jwlqd{DhON180	1010-EWM-507
	00505691C1DC1E0EAFD27F8DAD85C120	3.500	ST	051MaS7S7jwlqd{DhON180	800	051MaS7S7jwlqd{DhON180	1010-EWM-507
	00505691C1DC1ED993E7B7053BA6C0FB	1.728	ST	051MaS7S7jcJvxSSEwR0{m	800	051MaS7S7jcJvxSSEwR0{m	1010-EWM-40
	00505691C1DC1ED993E7C941D69D80FB	3.840	ST	051MaS7S7jcJvyb1rfs0{m	800	051MaS7S7jcJvyb1rfs0{m	1010-EWM-41
	00505691C1DC1ED993E7D50D0B1000FB	48	ST	051MaS7S7jcJvzKD2n00{m	800	051MaS7S7jcJvzKD2n00{m	1010-EWM-42
	00505691C1DC1ED993E7D50D0B1000FB	24	ST	051MaS7S7jcJvzKD2n00{m	800	051MaS7S7jcJvzKD2n00{m	1010-EWM-42
	00505691C1DC1ED992FE7894E04500FB	48	ST	051MaS7S7jci}dYKu4K0{m	800	051MaS7S7jci}dYKu4K0{m	1010-EWM-01
	00505691C1DC1ED992FE7894E04500FB	48	ST	051MaS7S7jci}dYKu4K0{m	800	051MaS7S7jci}dYKu4K0{m	1010-EWM-01
	00505691C1DC1ED992FE7894E04500FB	48	ST	051MaS7S7jci}dYKu4K0{m	800	051MaS7S7jci}dYKu4K0{m	1010-EWM-01
	00505691C1DC1ED992FFC0C58397A0FB	192	ST	051MaS7S7ic13i35Ww1JW{m	800	051MaS7S7ic13i35Ww1JW{m	1010-EWM-02

Bevor wir dieses Thema abschließen, noch ein paar Worte zu den Konvertierungsroutinen. Im Formelfenster befinden sich diese Funktionen auf der rechten Seite:

verfügbare Anweisungen	
Spezielle Anweisungen und Funktionen	
• FF_DATE_GET_DAY_OF_WEEK()	Wochentag eines Datums erhalten (Montag = 1)
• FF_DATE_GET_WEEK()	Woche eines Datums erhalten
...	
• FF_CONV_TO_FOREIGN_CURRENCY()	Umrechnung eines Betrags in Fremdwährung
• FF_GUID_X16_TO_C22()	GUID X16 in GUID C22 umwandeln
• FF_GUID_X16_TO_C32()	GUID X16 in GUID C32 umwandeln
• FF_GUID_C22_TO_X16()	GUID C22 in GUID X16 umwandeln
• FF_GUID_C22_TO_C32()	GUID C22 in GUID C32 umwandeln
• FF_GUID_C32_TO_X16()	GUID C32 in GUID X16 umwandeln
• FF_GUID_C32_TO_C22()	GUID C32 in GUID C22 umwandeln
ROW-Felder	

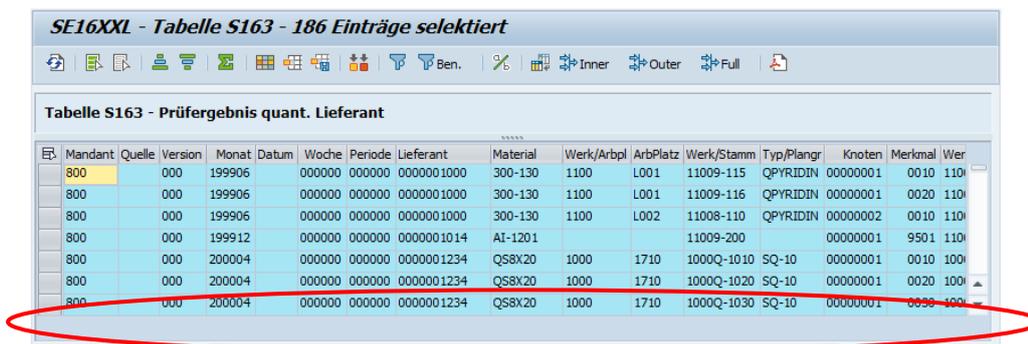
Mit einem Doppelklick auf eine Funktion erhalten Sie eine kurze Dokumentation:

```
GUID-Umwandlungs-Funktionen
*
* Eine GUID (Globally Unique Identifier) ist ein vom System
* generierter eindeutiger Schlüssel der in verschiedenen Datenbank-
* Tabellen verwendet wird. Eine GUID kann drei verschiedene Formate
* annehmen: X16 (16 Byte hexadezimal), C22 (22 Zeichen) und C32
* (32 Zeichen). Jedes Format kann in eines der anderen zwei Formate
* umgewandelt werden.
* Die drei Formate basieren auf folgenden Domänen:
```

[Zum Anfang](#)

Neue Benutzer-Einstellung “Schlüsselspalten rollbar”

ALV bietet unter bestimmten Umständen **keine horizontalen Scrollbalken** an. Dies ist der Fall, wenn eine Datenbanktabelle über zahlreiche Schlüsselfelder verfügt, die den gesamten sichtbaren Teil des Bildschirmfensters einnehmen. Dies geschieht auch bei CDS-Views mit vielen Schlüsselfeldern. Bei Views, die nur aus Schlüsselfeldern bestehen, tritt das Problem nicht auf, d.h. es muss mindestens ein normales Feld da sein. Ein Beispiel für eine solche Situation ist in folgender Abbildung zu sehen:



SE16XXL - Tabelle S163 - 186 Einträge selektiert

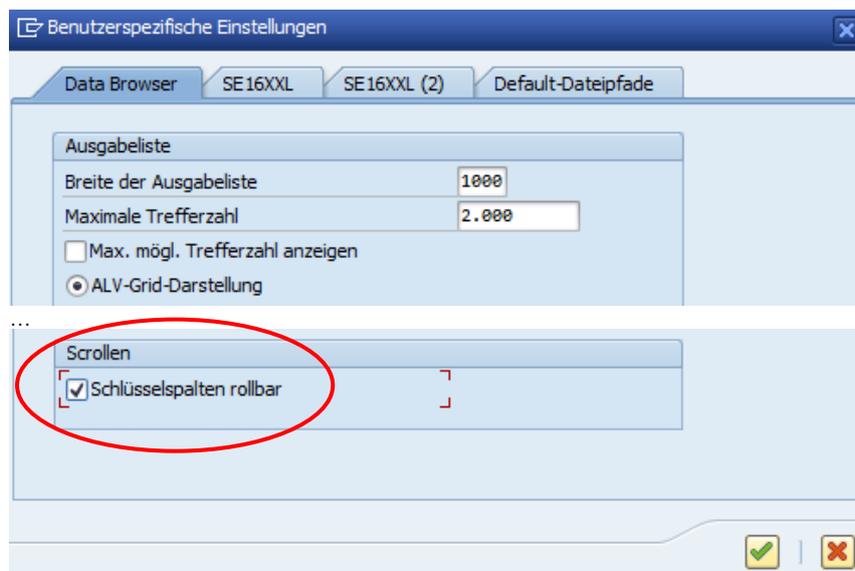
Tabelle S163 - Prüfergebnis quant. Lieferant

Mandant	Quelle	Version	Monat	Datum	Woche	Periode	Lieferant	Material	Werk/Arbpl	ArbPlatz	Werk/Stamm	Typ/Plangr	Knoten	Merkmal	Wer
800	000	199906	000000	000000	000000	000000	1000	300-130	1100	L001	11009-115	QPYRIDIN	00000001	0010	1100
800	000	199906	000000	000000	000000	000000	1000	300-130	1100	L001	11009-116	QPYRIDIN	00000001	0020	1100
800	000	199906	000000	000000	000000	000000	1000	300-130	1100	L002	11008-110	QPYRIDIN	00000002	0010	1100
800	000	199912	000000	000000	000000	000000	1014	AI-1201			11009-200		00000001	9501	1100
800	000	200004	000000	000000	000000	000000	1234	QS8X20	1000	1710	1000Q-1010	SQ-10	00000001	0010	1000
800	000	200004	000000	000000	000000	000000	1234	QS8X20	1000	1710	1000Q-1020	SQ-10	00000001	0020	1000
800	000	200004	000000	000000	000000	000000	1234	QS8X20	1000	1710	1000Q-1030	SQ-10	00000001	0030	1000

Die Scrollbalken erscheinen, wenn man folgende Menüfunktion verwendet:

Einstellungen → Spalten → Spalten entfixieren.

Das kann allerdings auf Dauer nervig sein, wenn man oft mit solchen Tabellen zu tun hat. Aus diesem Grund wurde eine **neue Benutzer-Einstellung** eingeführt:



Wenn diese Option aktiv ist, werden die Scrollbalken immer angezeigt, auch in den oben genannten Situationen.

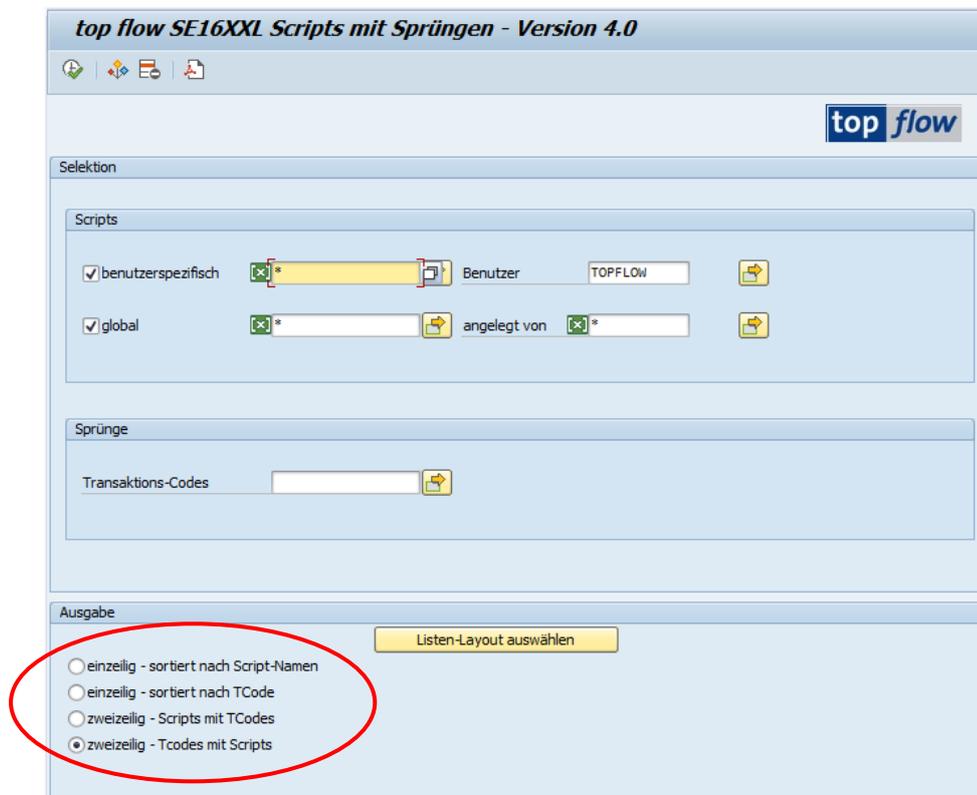
[Zum Anfang](#)

Liste von in Scripts definierten Sprüngen (TCodes)

Mit jeder neuen Version des SAP-Systems werden einige Transaktionscodes **obsolet** und müssen durch andere **ersetzt** werden. Im Falle eines **System-Upgrades** oder einer **Migration** ist es wichtig, Informationen über diese obsoleten TCodes einzuholen, um die notwendigen Anpassungen vornehmen zu können. In Bezug auf SE16XXL müssen alle Scripts, die Sprünge enthalten, auf mögliche TCodes überprüft werden, die ersetzt werden müssen. Bisher wäre diese Aufgabe recht mühsam gewesen, da der Script-Katalog nur anzeigt, welche Scripts einen oder mehrere definierte Sprünge haben, jedoch nichts über die verwendeten TCodes aussagt.

Aus diesem Grund wurde ab **Version 4.0** von SE16XXL ein Hilfsprogramm implementiert, das alle Scripts mit mindestens einem definierten Sprung auflistet, zusammen mit allen anderen Informationen zu den Sprüngen selbst. Das Programm erstellt **vier** Arten von Listen – **die letzte** bietet auch die Möglichkeit, die TCodes zu lokalisieren, die obsolet sind (oder werden). Für die alten TCodes werden auch die Ersatz-TCodes aufgelistet.

Die Selektionsmaske dieses Programms sieht folgendermaßen aus:



top flow SE16XXL Scripts mit Sprüngen - Version 4.0

Selektion

Scripts

benutzerspezifisch Benutzer TOPFLOW

global angelegt von

Sprünge

Transaktions-Codes

Ausgabe

Listen-Layout auswählen

einzeilig - sortiert nach Script-Namen

einzeilig - sortiert nach TCode

zweizeilig - Scripts mit TCodes

zweizeilig - Tcodes mit Scripts

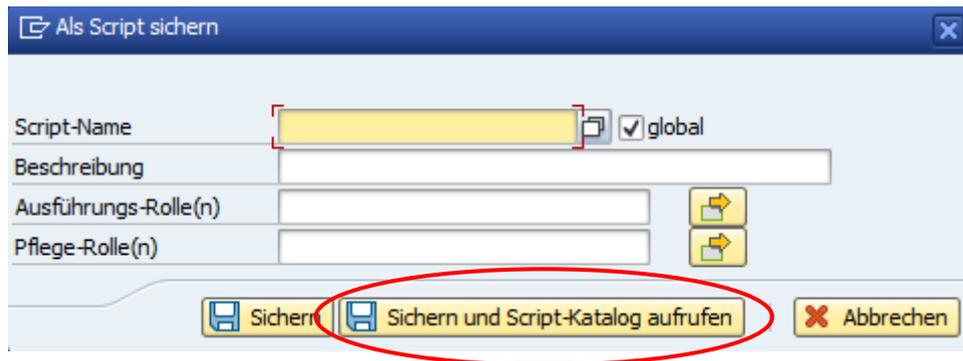
Für detailliertere Informationen lesen Sie bitte [Liste von Scripts mit Sprüngen](#).

[Zum Anfang](#)

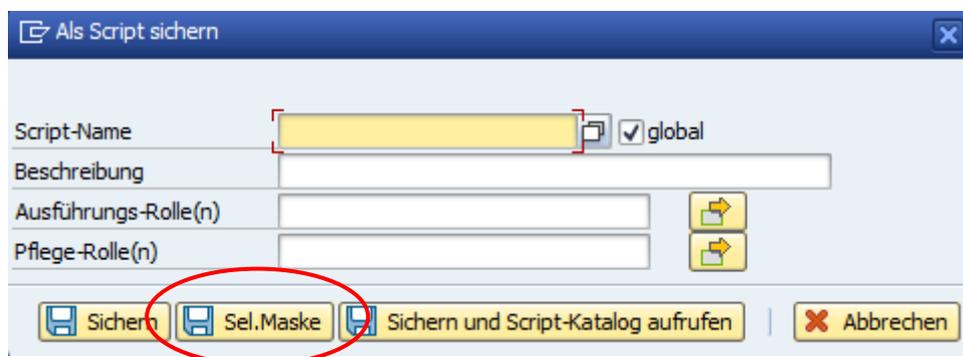
Zusätzliche Verbesserungen

Script Sichern und spezielle Selektionsmaske definieren

Bisher war es möglich, gleich nach dem Sichern eines Scripts zum Script-Katalog für das betreffende Script zu verzweigen. Das Dialogfenster zum Sichern eines Scripts war wie folgt konfiguriert:



Ab der Version 4.0 von SE16XXL ist es zusätzlich möglich, sofort zur Definition der **speziellen Selektionsmaske** des betreffenden Scripts zu verzweigen. Das Dialogfenster wurde entsprechend erweitert:



Nachdem die spezielle Selektionsmaske definiert wurde, ruft das Programm den Script-Katalog für das betreffende Script auf.

Server-Datei-Definition für Hintergrund-Job

Wenn ein Script mit der Option *“Ergebnisliste in Server-Datei speichern”* für die Ausführung in Hintergrund eingeplant wird, werden die Parameter, die die Eigenschaften der Server-Datei beschreiben, in einem **Set/Get-Parameter** hinterlegt. Auf diese Weise werden diese Parameter, wenn später ein weiteres Script auf die gleiche Weise eingeplant wird, als Standardwerte vorgeschlagen, in der Annahme, dass der zweite Job wahrscheinlich wie der erste konfiguriert wird. Bisher hatte diese Funktionalität jedoch einen **Nachteil**: Wenn in der **Übersicht der Hintergrund-jobs** für einen bestimmten Job die Funktion *Job(s) → Daten kopieren und erneut einplanen* aufgerufen wurde, wurden die ursprünglichen Parameter der Server-Datei-Definition **teilweise ignoriert** und an ihrer Stelle die Standardwerte aus dem **Set/Get-Parameter** präsentiert, eine unangenehme Verhaltensweise. Dasselbe geschah mit der Menüfunktion *Job(s) → Anford. eines periodischen Jobs ändern*.

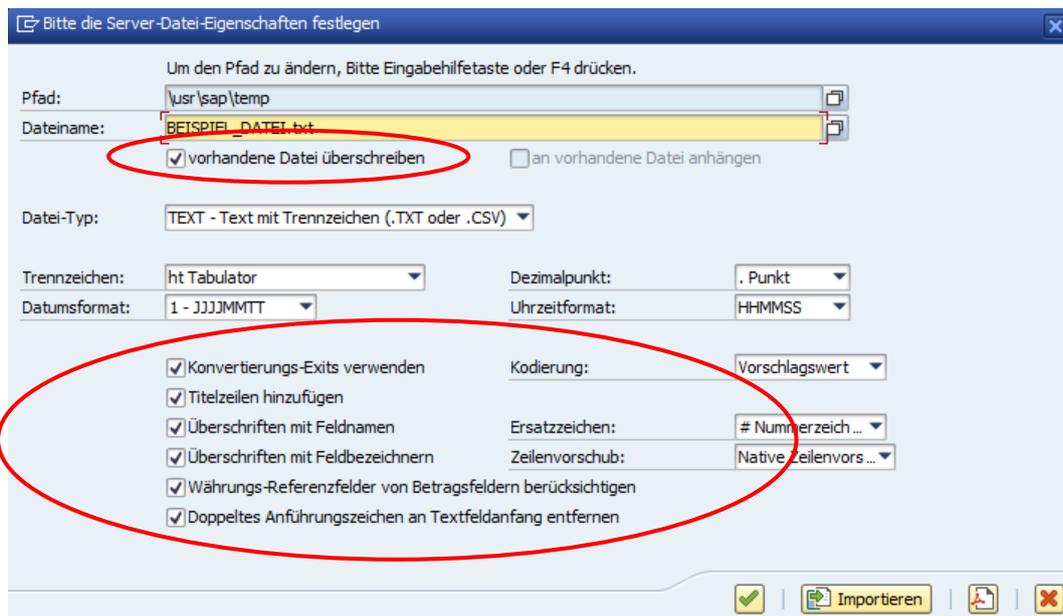
ANMERKUNG: Die Werte, die in einem Set/Get-Parameter gespeichert werden, bleiben bis zum Ende der aktuellen Dialogsitzung erhalten. Zu Beginn einer neuen Dialogsitzung werden sie initialisiert.

Um dieses negative Verhalten zu beseitigen, **lassen** ab Version 4.0 von SE16XXL die oben genannten Funktionen die Original-Definition der Server-Datei **unverändert**.

Ein **Beispiel** wird das Thema veranschaulichen.

Beispiel Teil 1 – Altes Verhalten

Wir planen ein Script für den Hintergrund ein. Dabei legen wir die Eigenschaften der Server-Datei wie folgt fest:



Bitte die Server-Datei-Eigenschaften festlegen

Um den Pfad zu ändern, Bitte Eingabehilfetaste oder F4 drücken.

Pfad: \\usr\sap\temp

Dateiname: BEISPIEL_DATEI.txt

vorhandene Datei überschreiben an vorhandene Datei anhängen

Datei-Typ: TEXT - Text mit Trennzeichen (.TXT oder .CSV)

Trennzeichen: ht Tabulator

Dezimalpunkt: . Punkt

Datumsformat: 1 - JJJJMMTT

Uhrzeitformat: HHMMSS

Konvertierungs-Exits verwenden

Titelzeilen hinzufügen

Überschriften mit Felddaten

Überschriften mit Feldbezeichnern

Währungs-Referenzfelder von Betragfeldern berücksichtigen

Doppeltes Anführungszeichen an Textfeldanfang entfernen

Kodierung: Vorschlagswert

Ersatzzeichen: # Nummerzeich...

Zeilenvorschub: Native Zeilenvors...

Importieren

Das Programm speichert die obige Definition im entsprechenden Set/Get-Parameter.

Nun möchten wir die Anforderung eines bestehenden Jobs kopieren und erneut einplanen. Die Server-Datei-Definition des ursprünglichen Jobs lautet wie folgt:



Server-Datei-Anforderung anzeigen

Pfad: \usr\sap\temp

Dateiname: FERTIGUNGSANFRAEGE_&DATUM_&ZEIT.txt

vorhandene Datei überschreiben an vorhandene Datei anhängen

Datei-Typ: TEXT - Text mit Trennzeichen (.TXT oder .CSV)

Trennzeichen: ht Tabulator Dezimalpunkt: . Punkt

Datumsformat: 1 - JJJJMMTT Uhrzeitformat: HHMMSS

Konvertierungs-Exits verwenden Titelzeilen hinzufügen

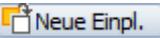
Überschriften mit Feldnamen Überschriften mit Feldbezeichnern

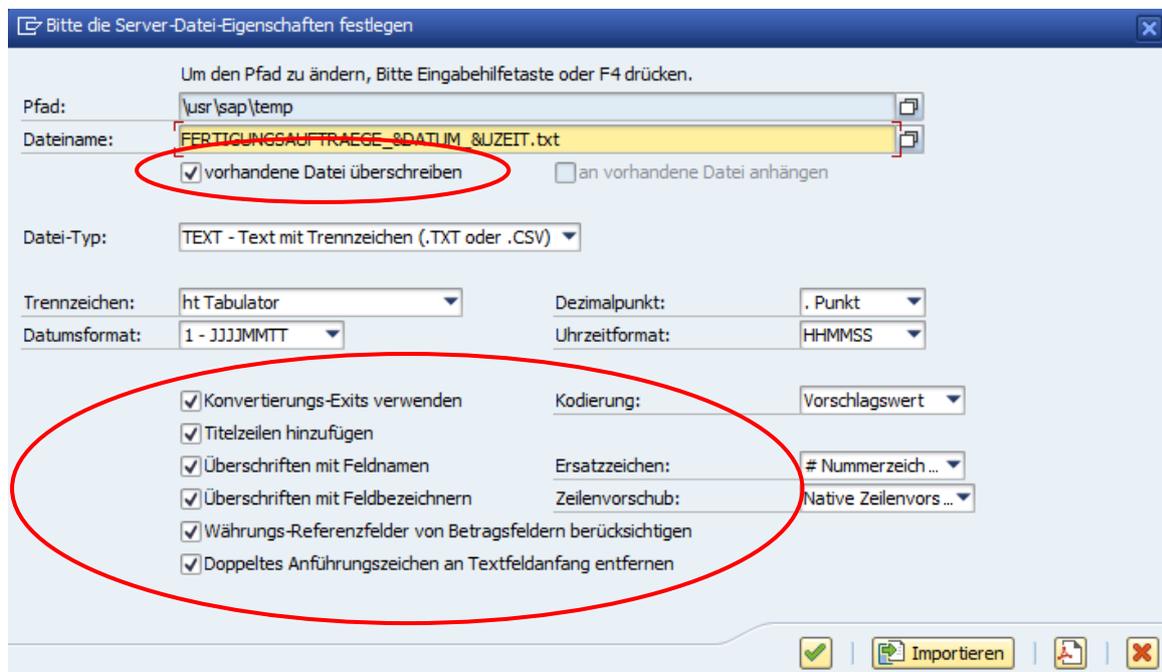
Währungs-Referenzfelder von Betragsfeldern berücksichtigen Doppeltes Anführungszeichen an Textfeldanfang entfernen

Kodierung: Vorschlagswert

Ersatzzeichen: # Nummerzeich ...

Zeilenvorschub: Native Zeilenvors ...

Wir rufen die Funktion  auf und erhalten nach Angabe der anderen Job-Optionen das folgende Dialogfenster zur Definition der Server-Datei:



Bitte die Server-Datei-Eigenschaften festlegen

Um den Pfad zu ändern, Bitte Eingabehilfetaste oder F4 drücken.

Pfad: \usr\sap\temp

Dateiname: FERTIGUNGSANFRAEGE_&DATUM_&ZEIT.txt

vorhandene Datei überschreiben an vorhandene Datei anhängen

Datei-Typ: TEXT - Text mit Trennzeichen (.TXT oder .CSV)

Trennzeichen: ht Tabulator Dezimalpunkt: . Punkt

Datumsformat: 1 - JJJJMMTT Uhrzeitformat: HHMMSS

Konvertierungs-Exits verwenden Titelzeilen hinzufügen

Überschriften mit Feldnamen Überschriften mit Feldbezeichnern

Währungs-Referenzfelder von Betragsfeldern berücksichtigen Doppeltes Anführungszeichen an Textfeldanfang entfernen

Kodierung: Vorschlagswert

Ersatzzeichen: # Nummerzeich ...

Zeilenvorschub: Native Zeilenvors ...

Importieren

Wie man sehen kann, unterscheidet sich die vorgeschlagene Definition von der ursprünglichen. Alle Optionen, die in der ursprünglichen Spezifikation inaktiv waren, wurden aus dem Set/Get-Parameter übernommen.

Beispiel Teil 2 – Neues Verhalten

Nun wiederholen wir die gesamte Prozedur in der Version **4.0** von SE16XXL. Die Server-Datei-Definition des ursprünglichen Jobs lautet in diesem Fall wie folgt:

Server-Datei-Anforderung anzeigen

Pfad: \usr\sap\ZES\tmp

Dateiname: KUNDENAUFTRAEGE_SDATUM_&UZEIT.txt

vorhandene Datei überschreiben an vorhandene Datei anhängen

Datei-Typ: TEXT - Text mit Trennzeichen (.TXT oder .CSV)

Trennzeichen: ht Tabulator Dezimalpunkt: . Punkt

Datumsformat: 1 - JJJJMMTT Uhrzeitformat: HHMMSS

Konvertierungs-Exits verwenden Kodierung: Vorschlagswert

Titelzeilen hinzufügen

Überschriften mit Feldnamen Ersatzzeichen: # Nummerzeich...

Überschriften mit Feldbezeichnern Zeilenvorschub: native Zeilenvors...

Währungs-Referenzfelder von Betragsfeldern berücksichtigen

Doppeltes Anführungszeichen an Textfeldanfang entfernen

Nun machen wir erneut von der Funktion Gebrauch und erhalten folgendes Definitionsfenster für die Eigenschaften der Server-Datei:

Bitte die Server-Datei-Eigenschaften festlegen

Um den Pfad zu ändern, Bitte Eingabehilfetaste oder F4 drücken.

Pfad: \usr\sap\ZES\tmp

Dateiname: KUNDENAUFTRAEGE_SDATUM_&UZEIT.txt

vorhandene Datei überschreiben an vorhandene Datei anhängen

Datei-Typ: TEXT - Text mit Trennzeichen (.TXT oder .CSV)

Trennzeichen: ht Tabulator Dezimalpunkt: . Punkt

Datumsformat: 1 - JJJJMMTT Uhrzeitformat: HHMMSS

Konvertierungs-Exits verwenden Kodierung: Vorschlagswert

Titelzeilen hinzufügen

Überschriften mit Feldnamen Ersatzzeichen: # Nummerzeich...

Überschriften mit Feldbezeichnern Zeilenvorschub: native Zeilenvors...

Währungs-Referenzfelder von Betragsfeldern berücksichtigen

Doppeltes Anführungszeichen an Textfeldanfang entfernen

Importieren Standardwerte

Dieses Mal wurde die ursprüngliche Definition **unverändert** gelassen.

[Zum Anfang](#)

Namen von Formelfeldern bis zu 30 Stellen lang

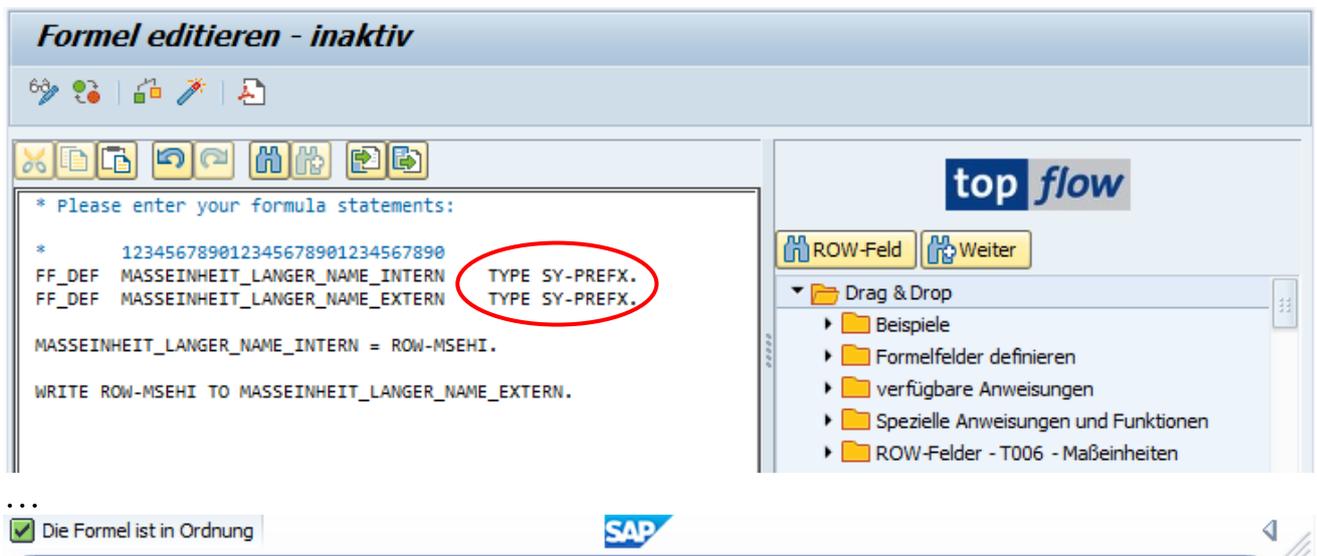
Bis dato konnten die Namen von Formelfeldern **maximal 16 Zeichen lang** sein. Dies war in älteren Versionen von SAP die maximale Länge von Feldnamen von Datenbanktabellen. Da diese Obergrenze jedoch in den neueren SAP-Versionen auf **30 Zeichen** angehoben wurde, wurde beschlossen, diese maximale Länge auch für SE16XXL Formelfeldnamen zuzulassen.

Diese Verbesserung wird nun anhand eines Beispiels verdeutlicht werden.

Beispiel

In diesem Beispiel erstellen wir ein Script, das die internen und externen Werte der **Maßeinheiten** anzeigt, die in der Tabelle **T006** enthalten sind. Der interne Wert ist der in der Datenbank gespeicherte (T006-MSEHI), der externe wird von der zugehörigen Konvertierungsroutine CUNIT erzeugt.

Wir beginnen mit der Selektion aller Einträge der Tabelle T006. Dann rufen wir den Formeleditor () auf und geben folgendes Coding ein:



Formel editieren - inaktiv

* Please enter your formula statements:

```
*
123456789012345678901234567890
FF_DEF MASSEINHEIT_LANGER_NAME_INTERN TYPE SY-PREFX.
FF_DEF MASSEINHEIT_LANGER_NAME_EXTERN TYPE SY-PREFX.

MASSEINHEIT_LANGER_NAME_INTERN = ROW-MSEHI.

WRITE ROW-MSEHI TO MASSEINHEIT_LANGER_NAME_EXTERN.
```

Die Formel ist in Ordnung

ANMERKUNG: Das Feld SY-PREFX wurde gewählt, weil es drei Zeichen lang ist und keine Konvertierungsroutine hat. Aber auch ein längeres Feld vom Typ CHAR wäre in Ordnung solange es keine Konvertierungsroutine enthält.

Nachdem wir die Formel () aktiviert haben und nur die beiden Formelfelder in der Ergebnisliste belassen haben, erhalten wir folgendes:

SE16XXL - Tabelle T006 - 248 Einträge selektiert

Tabelle T006 - Maßeinheiten

X~MASSEINHEIT_LANGER_NAME_INTERN	X~MASSEINHEIT_LANGER_NAME_EXTERN
%	%
%%	009
%O	%O
/MI	/MI
001	001

Da wir nur die Werte anzeigen wollen, die sich unterscheiden, nutzen wir die Menüfunktion

Zeilen → Sätze ausblenden wo Feld1 op Feld2

um die Zeilen zu eliminieren, in denen die beiden Werte identisch sind.
Das Endergebnis ist dann:

SE16XXL - Tabelle T006 - 59 Einträge selektiert

Tabelle T006 - Maßeinheiten

X~MASSEINHEIT_LANGER_NAME_INTERN	X~MASSEINHEIT_LANGER_NAME_EXTERN
%%	009
003	/NL
004	/PL
005	U/L
006	GDL
007	MGD
1	ONE
10	T
2M	CMS
2X	000
4G	µL
A93	GM3
B/H	018
R34	KN3

ANMERKUNG: Das Ergebnis hängt von der Anmeldesprache ab, da die externen Werte sprachabhängig sind.

[Zum Anfang](#)

Namen von V-Feldern bis zu 30 Stellen lang

Ähnlich wie bei den Formelfeldnamen können nun auch die **Namen der V-Felder** bis zu 30 Zeichen lang sein. Wir werden diese Verbesserung anhand eines Beispiels veranschaulichen.

Beispiel

Hier nehmen wir die Ergebnisliste des vorherigen Themas als Ausgangspunkt:

SE16XXL - Tabelle T006 - 59 Einträge selektiert

Tabelle T006 - Maßeinheiten

X~MASSEINHEIT_LANGER_NAME_INTERN	X~MASSEINHEIT_LANGER_NAME_EXTERN
%%%	009
003	/NL
004	/PI

Wir beginnen mit der Umwandlung der Formelfelder in V-Felder, indem wir die Menüfunktion **Bearbeiten** → **Formel** → **Formelfelder fixieren** verwenden. Das zugehörige Popup-Fenster wird angezeigt:

Bitte zu fixierende Formelfelder auswählen

Sel.	Formelfeld	V-Feld-Name	Tabelle/View	Feldname	Typ	Lng.	Ref.Feld
<input checked="" type="checkbox"/>	X~MASSEINHEIT_LANGER_NAME_INTERN	MASSEINHEIT_LANGER_NAME_INTERN	SYST	PREFX	CHAR	3	
<input checked="" type="checkbox"/>	X~MASSEINHEIT_LANGER_NAME_EXTERN	MASSEINHEIT_LANGER_NAME_EXTERN	SYST	PREFX	CHAR	3	

Wir akzeptieren die vorgeschlagenen Namen und erhalten:

SE16XXL - Tabelle T006 - 59 Einträge selektiert

Tabelle T006 - Maßeinheiten

V~MASSEINHEIT_LANGER_NAME_INTERN	V~MASSEINHEIT_LANGER_NAME_EXTERN
%%%	009
003	/NL
004	/PL
005	U/L
006	CFI

Die Ergebnisliste wäre aussagekräftiger, wenn für jede Maßeinheit auch eine Beschreibung zur Verfügung stünde. Um dies zu erreichen, fügen wir der Liste das **MSEHI**-Originalfeld hinzu und nutzen dann folgende Menüfunktion:

Extras → Zusatztexte einfügen.

Im Dialogfenster geben wir einen sehr langen Namen für das V-Feld ein:



Das Ergebnis ist:

Tabelle T006 - Maßeinheiten			
MSEHI	V~MASSEINHEIT_KURZE_BESCHREIBUNG	V~MASSEINHEIT_LANGER_NAME_INTERN	V~MASSEINHEIT_LANGER_NAME_EXTERN
%%%		%%%	009
003	pro Nanoliter	003	/NL
004	pro Pikoliter	004	/PL
005	Units/Liter	005	U/L
006	Gramm/Deziliter	006	G/D

Für das Endergebnis entfernen wir wieder das **MSEHI**-Feld und positionieren den Kurztext auf der rechten Seite der Liste:

SE16XXL - Tabelle T006 - 59 Einträge selektiert		
Tabelle T006 - Maßeinheiten		
V~MASSEINHEIT_LANGER_NAME_INTERN	V~MASSEINHEIT_LANGER_NAME_EXTERN	V~MASSEINHEIT_KURZE_BESCHREIBUNG
%%%	009	
003	/NL	pro Nanoliter
004	/PL	pro Pikoliter
005	U/L	Units/Liter
006	G/D	Gramm/Deziliter

[Zum Anfang](#)

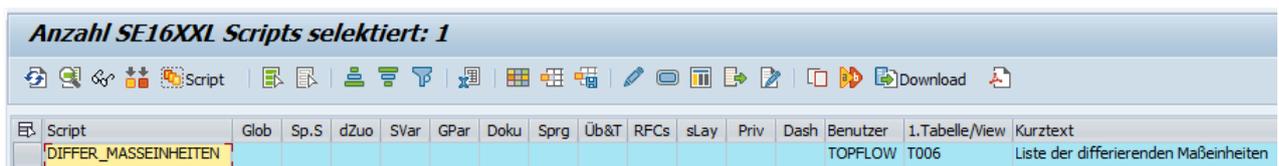
Anmerkungen zu Script-Operationen hinzufügen

Bis dato war die einzige Möglichkeit, die Funktionsweise eines SE16XXL Scripts zu erklären, entweder das Hinzufügen einer Doku zum Script oder das Einfügen von Dummy-Formeln, die irgendeine Art von Erklärung enthielten. Die erste Methode ist unzureichend, zumal die Dokumentation nicht sichtbar ist, wenn die Operationen angezeigt werden. Die zweite Methode ist unhandlich.

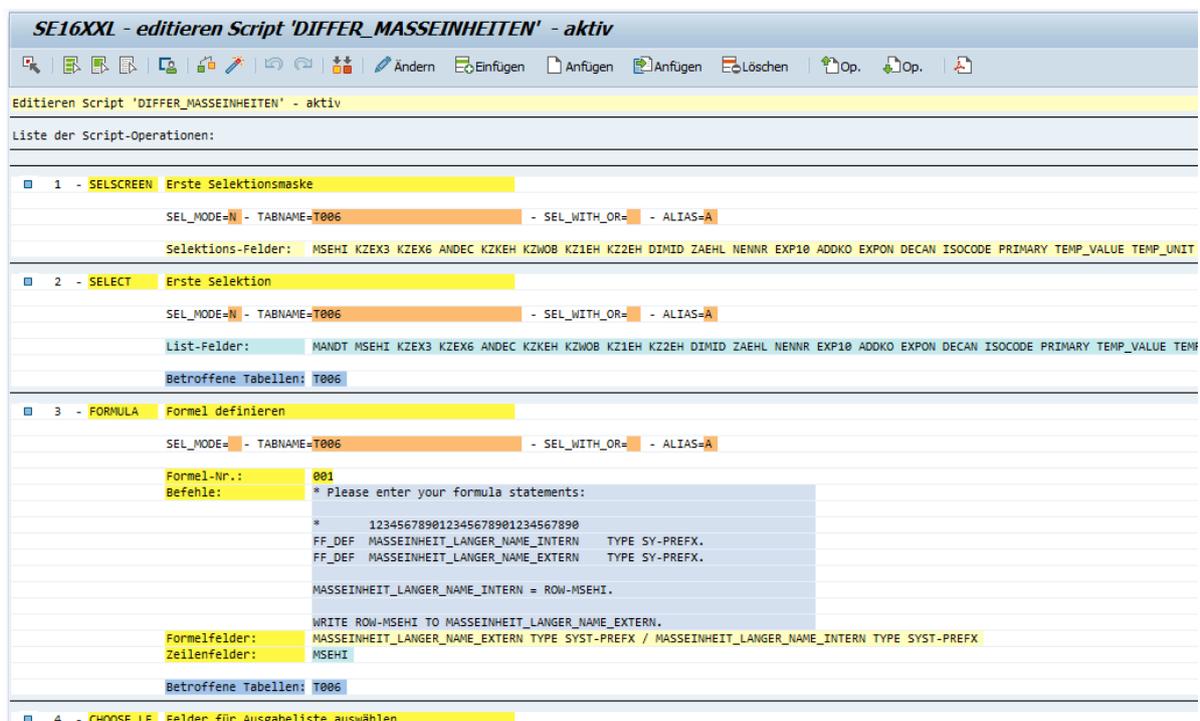
Ab Version 3.6E von SE16XXL ist es möglich, mit Hilfe des Script-Editors eine Anmerkung zu fast jeder Art von Operation eines Scripts hinzuzufügen. Ein Beispiel soll diese neue Funktion veranschaulichen.

Beispiel

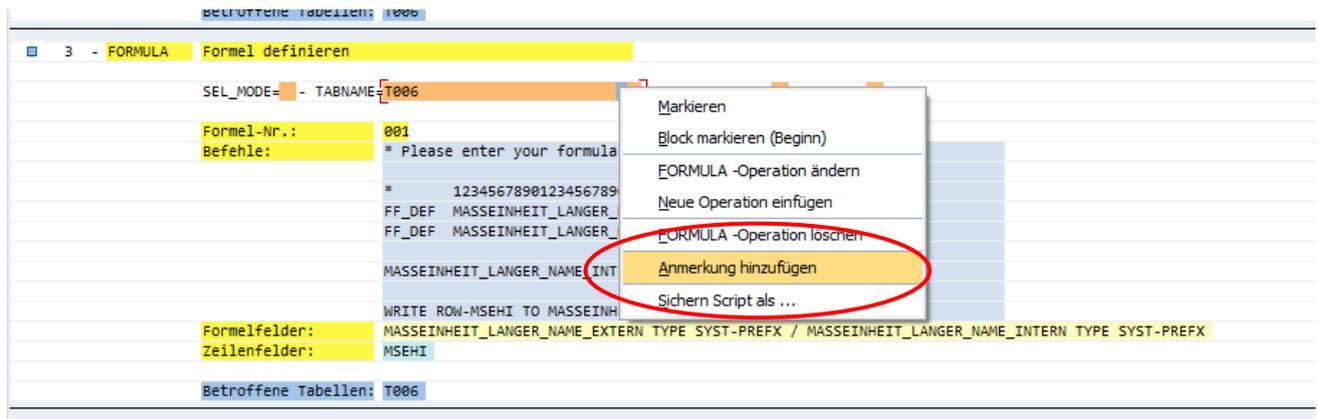
Als Ausgangspunkt nehmen wir das Script, das im vorherigen Thema erstellt wurde:



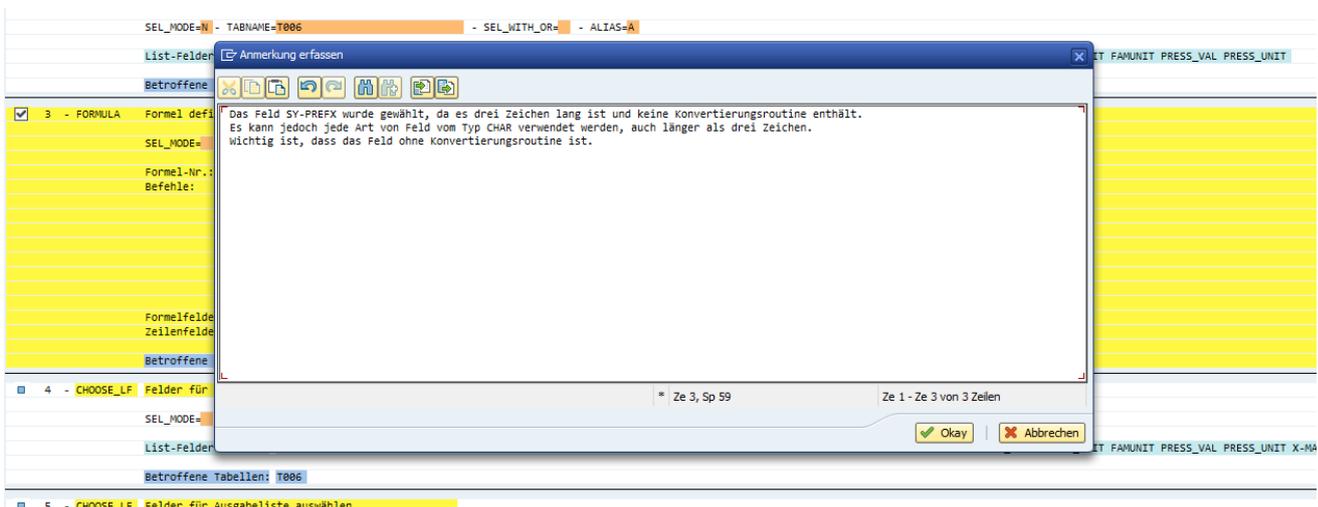
Wir rufen den Skript-Editor () auf und erhalten die Liste der Operationen:

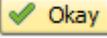


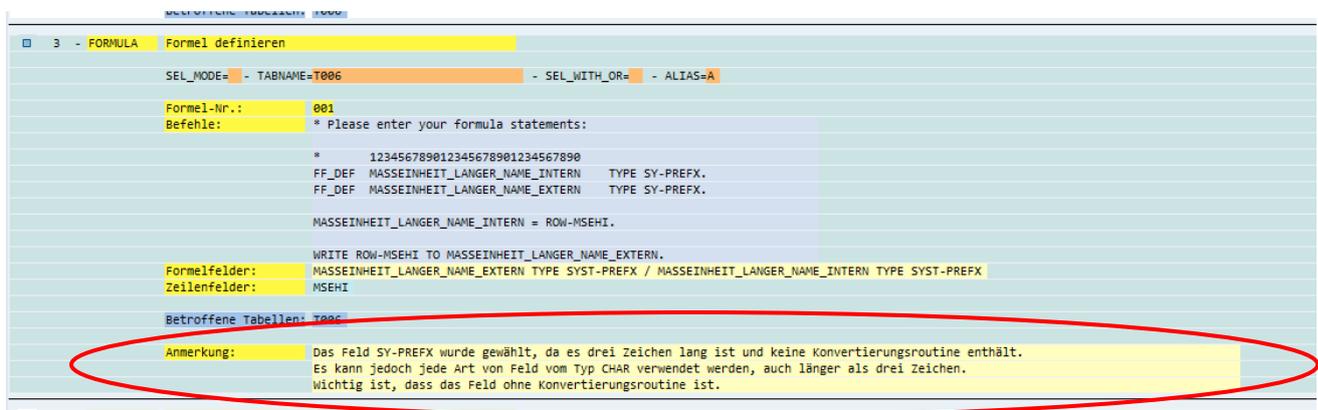
Wir verwenden das **Kontextmenü**, um eine Anmerkung zu einer bestimmten Operation hinzuzufügen:



Es öffnet sich nun ein Editor-Fenster, in dem wir unsere Anmerkung eingeben können:

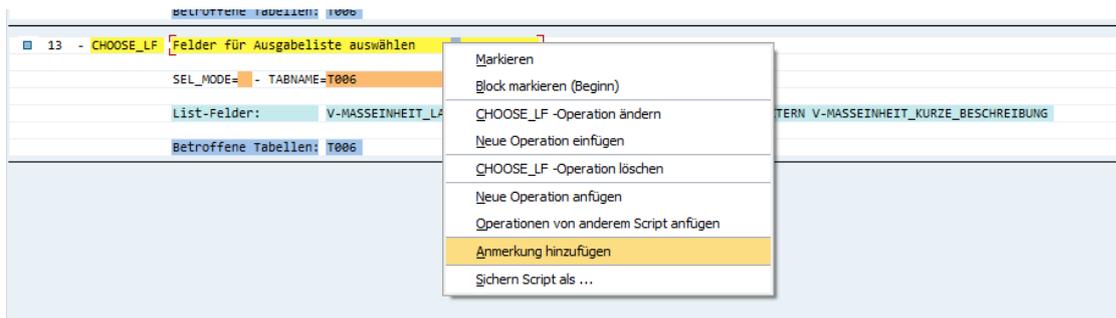


Wir schließen mit dem Betätigen von  und erhalten folgendes:

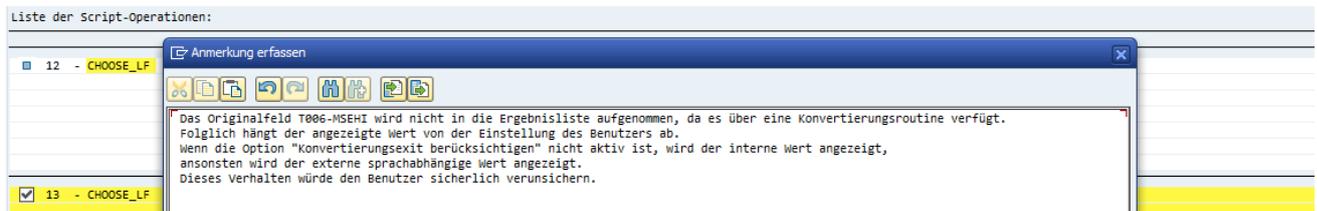


ANMERKUNG: Die Zeilen werden genau so angezeigt, wie sie eingegeben wurden.

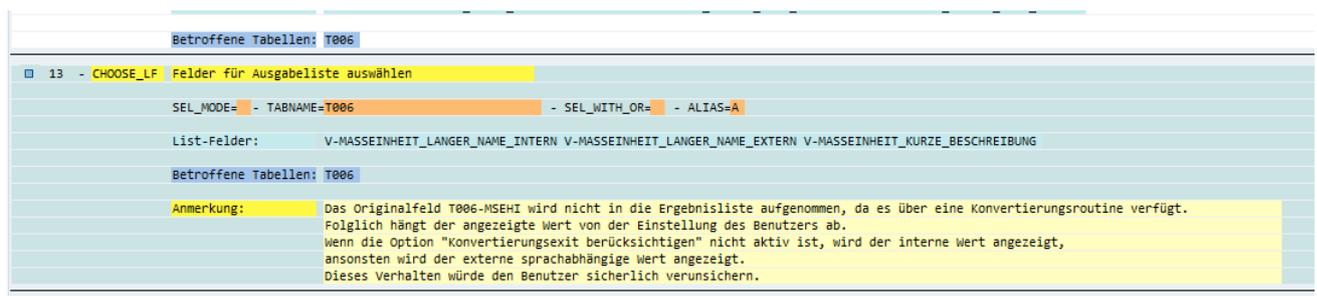
An dieser Stelle fügen wir eine weitere Anmerkung zur letzten Operation des Scripts hinzu:



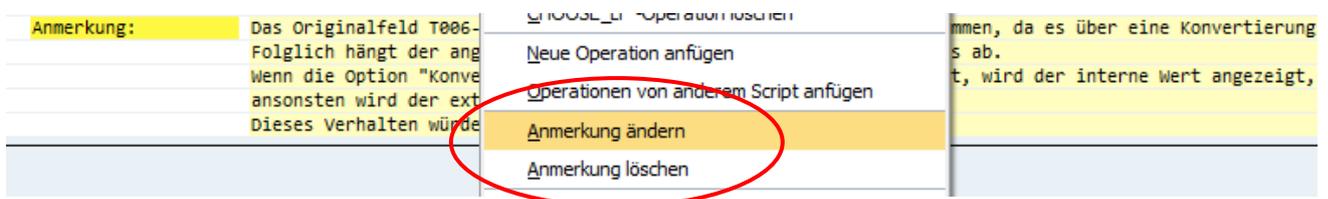
Hier erklären wir, warum das Originalfeld T006-MSEHI nicht angezeigt wird:



Das Endergebnis sieht dann wie folgt aus:



ANMERKUNG: Wenn eine bestimmte Operation eine Anmerkung enthält, bietet das Kontextmenü die Möglichkeit, sie entweder zu ändern oder zu löschen:



ANMERKUNG: Es sollte beachtet werden, dass die Operationen eines Scripts “komprimiert” werden, wenn das Script ausgeführt wird. Das bedeutet, dass unnötige oder irrelevante Operationen eliminiert werden. Von einer Reihe von SORT-Operationen wird z. B. nur die letzte beibehalten. Um sicherzustellen, dass die Anmerkungen nicht verschwinden, ist es ratsam, sie nur zu “wichtigen” Operationen wie SELECT usw. hinzuzufügen. Eine **CHOOSE_LF** Operation, die mit einer Anmerkung versehen ist, wird jedoch **nicht eliminiert**, auch wenn sie unnötig ist.

[Zum Anfang](#)

Klonen von langen zeichenartigen Feldern

Wenn in der Ergebnisliste eine bestimmte Zeile durch Doppelklick in Detail angezeigt wird, werden alle Felder vom Typ **CHAR**, die länger als 128 Zeichen sind, **abgeschnitten**, d.h. es werden nur die ersten 128 Zeichen angezeigt. Gleiches gilt für Felder vom Typ **LCHR**. Lediglich Felder vom Typ **STRING** werden vollständig angezeigt.

Um dieses Manko zu beheben, ist es ab Version 3.6E von SE16XXL möglich, die oben genannten Felder zu **klonen** und so ein V-Feld vom Typ **STRING** zu erstellen. Der Inhalt dieses V-Feldes ist dann, wenn die Zeile im Detail angezeigt wird, vollständig sichtbar.

Beispiel

Als Beispiel selektieren wir Einträge der Tabelle **USH10** (Änderungshistorie Berechtigungs-Profile), die das Feld **AUTHS** vom Typ **LCHR** enthält:

MANDT	PROFN	AKTIPS	MODDA	MODTI	MODBE	DIVIS	TYP	NRAUT	AUTHS
800	&_SAP_ALL	A	01.04.1997	15:15:57	SAP		G	3.302	C A_A_VIEW &_SAP_ALL A_B_ANLKL &_SAP_ALL A_B_BWART &_SAP_ALL A_C_AFAPL &_SAP_ALL A_IMPR_BUK&_SAP_ALL A_IM...
800	&_SAP_ALL	A	10.11.1997	17:09:42	SAP		G	3.302	C A_A_VIEW &_SAP_ALL A_B_ANLKL &_SAP_ALL A_B_BWART &_SAP_ALL A_C_AFAPL &_SAP_ALL A_IMA_ART &_SAP_ALL A_IM...
800	&_SAP_ALL_1	A	01.04.1997	15:15:57	SAP		G	3.302	C F_LFA1_BUK&_SAP_ALL F_MAHN_BUK&_SAP_ALL F_MAHN_KOAR&_SAP_ALL F_PAYR_BUK&_SAP_ALL F_REGU_BUK&_SAP_ALL F...
800	&_SAP_ALL_1	A	10.11.1997	17:09:52	SAP		G	3.302	C F_FICA_CCT&_SAP_ALL F_FICA_CTR&_SAP_ALL F_FICA_FCD&_SAP_ALL F_FICA_FMC&_SAP_ALL F_FICA_FTR&_SAP_ALL F_FI...
800	&_SAP_ALL_2	A	27.02.1997	09:44:48	SAP		G	3.302	C M_EINF_EKG&_SAP_ALL M_EINF_EKO&_SAP_ALL M_EINF_WRK&_SAP_ALL M_EINK_FRG&_SAP_ALL M_EVNT_ANL&_SAP_ALL M...
800	&_SAP_ALL_2	A	01.04.1997	15:15:57	SAP		G	3.302	C M_EINF_EKG&_SAP_ALL M_EINF_EKO&_SAP_ALL M_EINF_WRK&_SAP_ALL M_EINK_FRG&_SAP_ALL M_EVNT_ANL&_SAP_ALL M...
800	&_SAP_ALL_2	A	10.11.1997	17:09:52	SAP		G	3.302	C K_PCAB_DEL&_SAP_ALL K_PCAD_LUM &_SAP_ALL K_PCAF_UEB&_SAP_ALL K_PCAI_UEB&_SAP_ALL K_PCAI_GEN&_SAP_ALL K_P...
800	&_SAP_ALL_3	A	01.04.1997	15:15:57	SAP		G	1.608	C S_LOG_COM &_SAP_ALL S_NUMBER &_SAP_ALL S_OC_DOC &_SAP_ALL S_OC_FOLCR&_SAP_ALL S_OC_ROLE &_SAP_ALL S_O...
800	&_SAP_ALL_3	A	10.11.1997	17:09:52	SAP		G	3.302	C M_RECH_AKZ&_SAP_ALL M_RECH_BUK&_SAP_ALL M_RECH_SPIG&_SAP_ALL M_RECH_WRK&_SAP_ALL M_SKPF_VGA&_SAP_ALL ...
800	DIP_05	A	21.07.1997	12:49:12	CURA		S	1.652	C K_MLNUSER &_SAP_ALL K_MLPUSER &_SAP_ALL K_CBPR_SET&_SAP_ALL K_CKPH_SET&_SAP_ALL K_CBPR_PLA&_SAP_ALL K_CK...
800	DIP_05	A	21.07.1997	13:16:00	CJIRA		S	1.652	M K_MLNUSER &_SAP_ALL K_MLPUSER &_SAP_ALL K_CBPR_SFTR &_SAP_ALL K_CKPH_SFTR &_SAP_ALL K_CBPR_PLA &_SAP_ALL K_C...

Wenn wir auf eine der Zeilen doppelklicken, erhalten wir folgende Detailansicht:

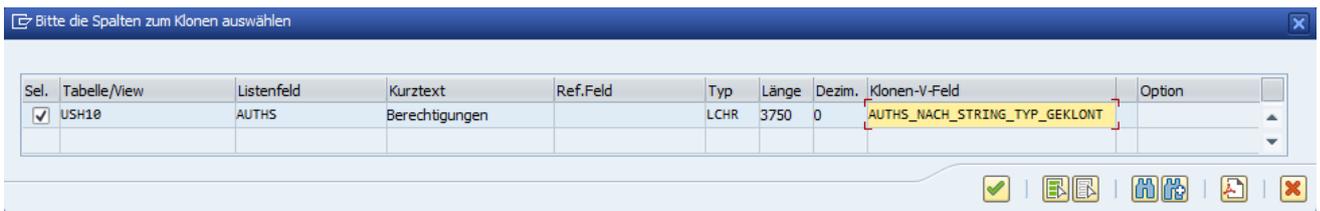
MANDT	PROFN	AKTIPS	MODDA	MODTI	MODBE	DIVIS	TYP	NRAUT	AUTHS
800	&_SAP_ALL	A	01.04.1997	15:15:57	SAP		G	3.302	C A_A_VIEW &_SAP_ALL A_B_ANLKL &_SAP_ALL A_B_BWART &_SAP_ALL A_C_AFAPL &_SAP_ALL A_IMPR_BUK&_SAP_ALL A_IMPR_GSB&_SAP_ALL A_IMPR_KOK&_SAP_ALL A_IMPR_PRC&_SAP...

Wie zu sehen ist, ist nur der erste Teil des **AUTHS**-Feldes sichtbar.

Um den gesamten Inhalt dieser Felder einsehen zu können, markieren wir die Spalte **'AUTHS'** und bedienen uns dann der folgenden Menüfunktion:

Extras → Spalten klonen → Normales Klonen.

Daraufhin öffnet sich folgendes Dialogfenster:



In der Ergebnisliste wird nun die neue Spalte angezeigt:

MANDT	PROFN	AKTIPS	MODDA	MODTI	MODBE	DIVIS	TYP	NRAUT	AUTHS	V~AUTHS_NACH_STRING_TYP_GEKLONT						
800	&_SAP_ALL	A	01.04.1997	15:15:57	SAP	G	3.302	C A_A_VIEW &_SAP_ALL	A_B_ANLKL &_SAP_ALL	A_B_BWART &_SAP_ALL	A_C_AFAPL &_SAP_ALL	A_IMPR_BUK &_SAP_ALL	A_IMPR_GSB &_SAP_ALL	A_IMPR_KOK &_SAP_ALL	A_IMPR_PRC &_SAP_ALL	A_IMPR_VER &_SAP_ALL

Ein Doppelklick auf eine Zeile gibt nun eine vollständige Ansicht des Inhalts von AUTHS:

Satz von Tabelle USH10 - Änderungshistorie Berechtigungs-Profile

Satz von Tabelle USH10 - Änderungshistorie Berechtigungs-Profile

MANDT	'800'	Mandant
PROFN	'&_SAP_ALL'	Berechtigungsprofil in Benutzerstammpflege
AKTIPS	'A'	Aktiv- oder Pflegeversion
MODDA	'01.04.1997'	Modifikationsdatum
MODTI	'15:15:57'	Modifikationszeit
MODBE	'SAP'	Letzter Änderer
DIVIS	'G'	Abteilung
TYP	'g'	Typ des Profils (Sammel- od. Einzelprofil)
NRAUT	'3.302'	Anzahl der Profiles oder Berechtigungen
AUTHS	'C A_A_VIEN &_SAP_ALL	A_B_ANLKL &_SAP_ALL

V~AUTHS_NACH_STRING_TYP_GEKLONT

```
'C A_A_VIEN &_SAP_ALL A_B_ANLKL &_SAP_ALL A_B_BWART &_SAP_ALL A_C_AFAPL &_SAP_ALL A_IMPR_BUK &_SAP_ALL A_IMPR_GSB &_SAP_ALL A_IMPR_KOK &_SAP_ALL A_IMPR_PRC &_SAP_ALL A_IMPR_VER &_SAP_ALL'
'A_INTP_ART &_SAP_ALL A_INT_EIS &_SAP_ALL A_M_ANLKL &_SAP_ALL A_PERI_BUK &_SAP_ALL A_S_ANLGR &_SAP_ALL A_S_ANLKL &_SAP_ALL A_S_GSBER &_SAP_ALL A_S_WERK &_SAP_ALL B_ALE_LSYS &_SAP_ALL'
'B_ALE_MAST &_SAP_ALL B_ALE_MODAL &_SAP_ALL B_ALE_RECV &_SAP_ALL B_ALE_RECV &_SAP_ALL B_USERSSTAT &_SAP_ALL C_AENR_BGR &_SAP_ALL C_AENR_RV &_SAP_ALL C_AENR_RV &_SAP_ALL C_AFFW_TWR &_SAP_ALL'
'C_AFKO_ACT &_SAP_ALL C_AFKO_ACT &_SAP_ALL C_AFKO_AKT &_SAP_ALL C_AFKO_DISE &_SAP_ALL C_AFRU_APL &_SAP_ALL C_AFRU_APL &_SAP_ALL C_AFRU_APL &_SAP_ALL C_AFRU_APL &_SAP_ALL C_AFRU_USR &_SAP_ALL'
'C_APPL_SOP &_SAP_ALL C_APPL_ART &_SAP_ALL C_APPL_WRK &_SAP_ALL C_BFLS &_SAP_ALL C_BFLS_I &_SAP_ALL C_BFLS_N &_SAP_ALL C_BFLS_R &_SAP_ALL C_CABN &_SAP_ALL C_CAPV_ANR &_SAP_ALL'
'C_CAPV_STA &_SAP_ALL C_CLAS_ER &_SAP_ALL C_CLAS_WNR &_SAP_ALL C_CLAS_USR &_SAP_ALL C_CLAS_UTI &_SAP_ALL C_CREC_WRK &_SAP_ALL C_CRFH_BGR &_SAP_ALL C_CRPT_BER &_SAP_ALL C_DRAM_BGR &_SAP_ALL'
'C_DRAM_DOK &_SAP_ALL C_DRAM_STA &_SAP_ALL C_DRAM_TCD &_SAP_ALL C_EVAL_WRK &_SAP_ALL C_KANBAN &_SAP_ALL C_KAPA_ABG &_SAP_ALL C_KAPA_PLA &_SAP_ALL C_KLAH_BK &_SAP_ALL C_KLAH_BKP &_SAP_ALL'
'C_KLAH_BSE &_SAP_ALL C_MESS_WRK &_SAP_ALL C_MLST_BGR &_SAP_ALL C_PDC &_SAP_ALL C_PLAN_SOP &_SAP_ALL C_PLKO_PLG &_SAP_ALL C_POI &_SAP_ALL C_PPBD &_SAP_ALL C_PPBD_REO &_SAP_ALL'
'C_PROJ_KOK &_SAP_ALL C_PROJ_PRC &_SAP_ALL C_PROJ_TCD &_SAP_ALL C_PROJ_VNR &_SAP_ALL C_PRPS_ART &_SAP_ALL C_PRPS_KOK &_SAP_ALL C_PRPS_KST &_SAP_ALL C_PRPS_PRC &_SAP_ALL C_PRPS_USR &_SAP_ALL'
'C_PRPS_VNR &_SAP_ALL C_PSTX_ART &_SAP_ALL C_ROUT &_SAP_ALL C_ROUT_MAT &_SAP_ALL C_SAFK &_SAP_ALL C_SHEG_G1 &_SAP_ALL C_SHEG_GEN &_SAP_ALL C_SHEP_TPB &_SAP_ALL C_SHEP_TPB &_SAP_ALL'
'C_SHEP_TRH &_SAP_ALL C_SHEP_TRH &_SAP_ALL C_STUE_BER &_SAP_ALL C_STUE_MAS &_SAP_ALL C_STUE_NOH &_SAP_ALL C_STUE_WRK &_SAP_ALL C_TCLA_BKA &_SAP_ALL C_TCLS_BER &_SAP_ALL C_TRAN &_SAP_ALL'
'C_VERS_ACT &_SAP_ALL F_AVIK_AVA &_SAP_ALL F_AVIK_BUK &_SAP_ALL F_BKPF_BER &_SAP_ALL F_BKPF_BER &_SAP_ALL F_BKPF_BER &_SAP_ALL F_BKPF_BER &_SAP_ALL F_BKPF_BER &_SAP_ALL F_BKPF_BER &_SAP_ALL'
'F_BKPF_GSB &_SAP_ALL F_BKPF_KO &_SAP_ALL F_BKPF_VW &_SAP_ALL F_BKPA_BUK &_SAP_ALL F_BKPA_MAR &_SAP_ALL F_FDSE_BUK &_SAP_ALL F_FDSE_GSB &_SAP_ALL F_FDSE_BUK &_SAP_ALL F_FDSE_GSB &_SAP_ALL'
'F_FDSE_BUK &_SAP_ALL F_FDSE_GSB &_SAP_ALL F_FDSE_BUK &_SAP_ALL'
'F_FICA_WTR &_SAP_ALL F_FICD_PSR &_SAP_ALL F_FICD_PSR &_SAP_ALL F_FICD_VER &_SAP_ALL F_KBE_RESE &_SAP_ALL F_KBE_TRAB &_SAP_ALL F_KMT_MGT &_SAP_ALL F_KNA1_AEN &_SAP_ALL F_KNA1_APP &_SAP_ALL'
'F_KNA1_BER &_SAP_ALL F_KNA1_BUK &_SAP_ALL F_KNA1_KG &_SAP_ALL F_KNA1_ANR &_SAP_ALL F_KNA1_AEN &_SAP_ALL F_KNA1_KKB &_SAP_ALL F_KNA1_MAR &_SAP_ALL F_LC_US &_SAP_ALL F_LC_COM &_SAP_ALL F_LC_ERH'
'&_SAP_ALL F_LC_LDR &_SAP_ALL F_LC_SUBD &_SAP_ALL F_LFA1_AEN &_SAP_ALL F_LFA1_APP &_SAP_ALL F_LFA1_BER &_SAP_ALL' Berechtigungen (Klone)
```

ANMERKUNG: Es werden maximal 30 Zeilen des Inhalts eines STRING-Feldes angezeigt. Die Anzahl der Zeilen hängt von den Abmessungen des Fensters ab.

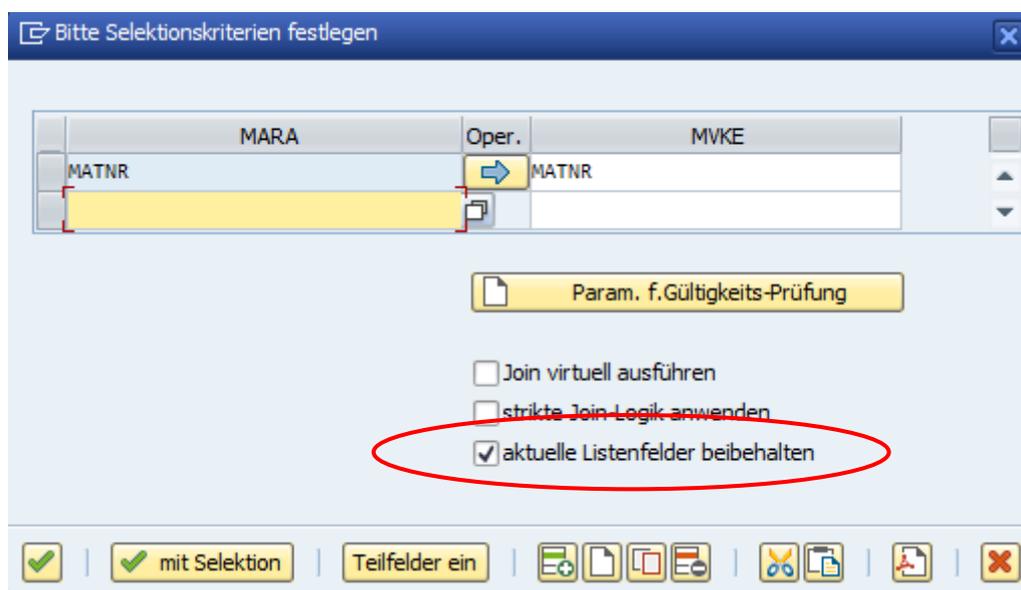
ANMERKUNG: Das gleiche Ergebnis könnte durch die Verwendung einer Formel erzielt werden. Jedoch haben viele Anwender, die mit ABAP nicht vertraut sind, ein Problem mit Formeln. Aus diesem Grund wurde es als benutzerfreundlicher angesehen, diese Funktionalität als separate Funktion anzubieten.

[Zum Anfang](#)

Join-Layouts

SE16XXL merkt sich für alle Join-Kombinationen von Tabellen/Views die resultierenden Listenfelder, die von einer bestimmten Logon-Kennung definiert wurden. Mit anderen Worten, wenn ein Join durchgeführt wird und diese Kombination von Tabellen/Views bereits von der aktuellen Logon-Kennung verwendet wurde, richtet SE16XXL die Ergebnisliste so ein, wie sie in der vorherigen Sitzung definiert wurde.

Die obige Logik gilt nicht, wenn bei der Festlegung der Join-Kriterien die Option *“aktuelle Listenfelder beibehalten”* aktiviert wird:



In diesem Fall oder wenn die Join-Kombination völlig neu ist, d.h. noch nie zuvor verwendet wurde, übernimmt das Programm **alle Schlüsselfelder** der neuen Tabelle/View oder, wenn es weniger als sechs sind, die **ersten sechs Felder**.

Angenommen, für eine gegebene Tabelle, z.B. **PA0002**, sind nur die Felder **VORNA** (Vorname) und **NACHN** (Nachname) erwünscht, kann es sehr unangenehm sein, jedes Mal die automatisch ausgewählten Listenfelder korrigieren zu müssen, um das gewünschte Ergebnis zu erhalten.

Um diese Unzulänglichkeit zu beseitigen, ist es ab Version 3.6E von SE16XXL möglich, für jede Tabelle/View ein **“Join-Layout”** zu definieren, d.h. die Standard-Listenfelder, die bei einem Join verwendet werden sollen.

ANMERKUNG: Diese Funktion gilt nur für lokale Datenbanktabellen und -Views. Sie gilt nicht für Joins mit einer Frontend-Datei oder mit einer entfernten (RFC) Tabelle/View.

Diese neue Funktionalität werden wir nun anhand eines Beispiels verdeutlichen.

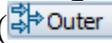
Beispiel

Wir beginnen mit einer Liste von Einträgen der Tabelle **PA0014**:

SE16XXL - Tabelle PA0014 - 998 Einträge selektiert

Tabelle PA0014 - Personal-Stammsatz Infotyp 0014 (Wiederk. Be- u. Abzüge)

MANDT	PERNR	SUBTY	OBJPS	SPRPS	ENDDA	BEGDA	SEQNR	LGART	BETRG	WAERS	ZANZL
800	00001000	M110			31.12.9999	01.01.2002	000	M110	1.515,13	EUR	12
800	00001000	M120			31.12.9999	01.01.2002	000	M120	4.545,38	EUR	12
800	00001000	M810			31.12.9999	01.01.2002	000	M810	0,00	EUR	0
800	00001001	M110			31.12.9999	01.01.2002	000	M110	542,14	EUR	12
800	00001001	M120			31.12.9999	01.01.2002	000	M120	1.626,42	EUR	12
800	00001001	M810			31.12.9999	01.01.2002	000	M810	0,00	EUR	0
800	00001001	M910			31.12.9999	01.01.2002	000	M910	511,29	EUR	0
800	00001002	M110			31.12.9999	01.01.2002	000	M110	1.533,88	EUR	12
800	00001002	M120			31.12.9999	01.01.2002	000	M120	4.601,63	EUR	12
800	00001003	M110			31.12.9999	01.01.2002	000	M110	571,62	EUR	12
800	00001003	M120			31.12.9999	01.01.2002	000	M120	1.714,87	EUR	12
800	00001004	M110			31.12.9999	01.01.2002	000	M110	514,02	EUR	12
800	00001004	M120			31.12.9999	01.01.2002	000	M120	1.542,06	EUR	12

Nun verknüpfen wir diese Tabelle mit **PA0002**, um die Ergebnisliste mit den Vor- und Nachnamen der beteiligten Personen zu ergänzen. Zu diesem Zweck führen wir einen Outer-Join durch ():

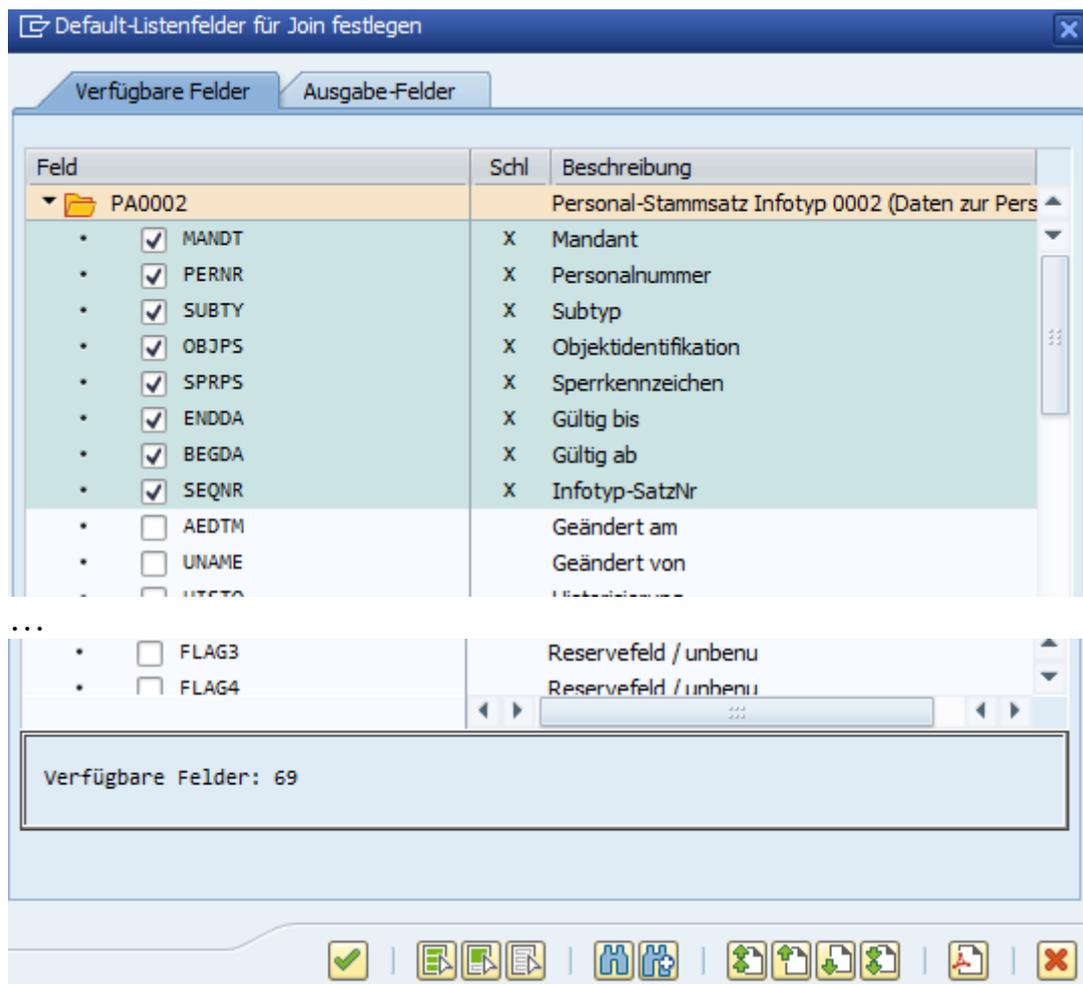
Outer-Join

Tabelle / View

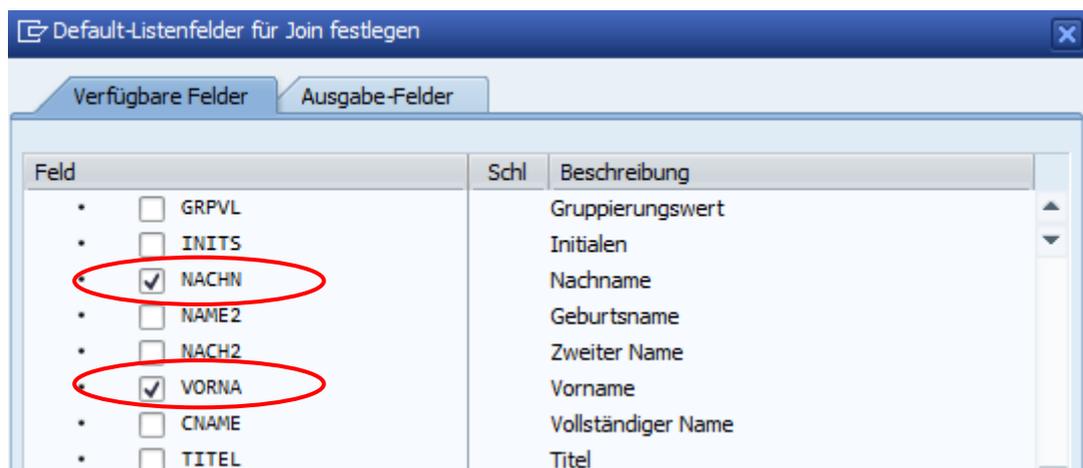
Frontend-Datei uploaden

RFC-Destination

An dieser Stelle, bevor wir fortfahren, betätigen wir die Schaltfläche  und definieren auf diese Weise die **Standardfelder** von PA0002, die für zukünftige Join-Operationen verwendet werden sollen. Das Programm reagiert mit einem Dialogfenster, ähnlich dem, in dem die Felder der Ergebnisliste festgelegt werden. In diesem Fall stehen jedoch nur die Felder der Tabelle PA0002 zur Verfügung:



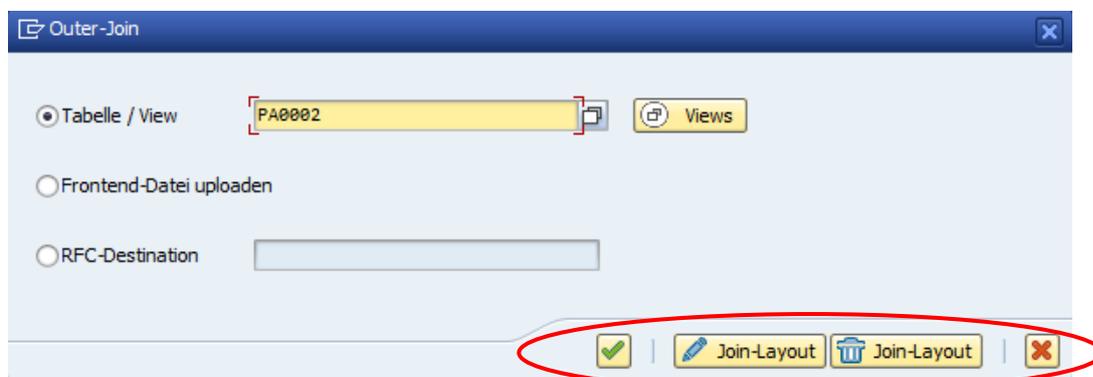
Zuerst entmarkieren wir sämtliche Felder (📄) und markieren dann die beiden gewünschten Felder:



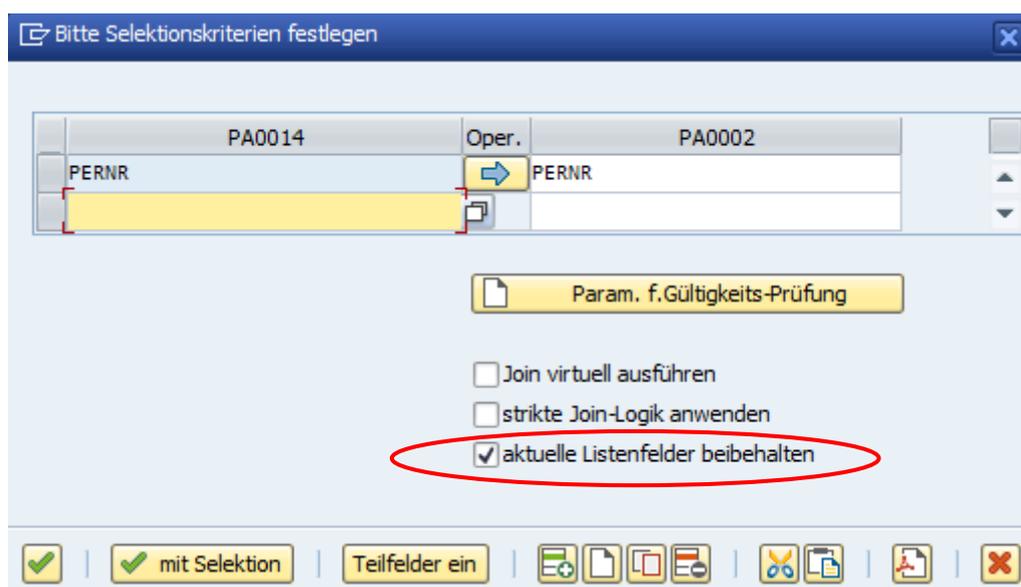
Dann wechseln wir zu den Ausgabe-Feldern und ändern die Reihenfolge der beiden Felder, indem wir VORNA vor NACHN setzen:



Beim Zurückkehren zum vorherigen Dialogfenster stellen wir fest, dass nun andere Schaltflächen verfügbar sind – sie zeigen, dass ein Join-Layout vorhanden ist:



Wir können nun mit der Definition der Join-Kriterien fortfahren:



Nach Festlegung der Parameter für die Gültigkeits-Prüfung erhalten wir folgende Ergebnisliste:

SE16XXL - Outer-Join - 1019 Ergebnis-Sätze

Join von PA0014(A) und PA0002(B)

A~MANDT	A~PERNR	A~SUBTY	A~OBJPS	A~SPRPS	A~ENDDA	A~BEGDA	A~SEQNR	A~LGART	A~BETRG	A~WAERS	A~ZANZL	B~VORNA	B~NACHN
800	00001000	M110			31.12.9999	01.01.2002	000	M110	1.515,13	EUR	12	Anja	Müller
800	00001000	M110			31.12.9999	01.01.2002	000	M110	1.515,13	EUR	12	Anja	Müller
800	00001000	M120			31.12.9999	01.01.2002	000	M120	4.545,38	EUR	12	Anja	Müller
800	00001000	M120			31.12.9999	01.01.2002	000	M120	4.545,38	EUR	12	Anja	Müller
800	00001000	M810			31.12.9999	01.01.2002	000	M810	0,00	EUR	0	Anja	Müller
800	00001000	M810			31.12.9999	01.01.2002	000	M810	0,00	EUR	0	Anja	Müller
800	00001001	M110			31.12.9999	01.01.2002	000	M110	542,14	EUR	12	Michaela	Maier
800	00001001	M120			31.12.9999	01.01.2002	000	M120	1.626,42	EUR	12	Michaela	Maier
800	00001001	M810			31.12.9999	01.01.2002	000	M810	0,00	EUR	0	Michaela	Maier
800	00001001	M910			31.12.9999	01.01.2002	000	M910	511,29	EUR	0	Michaela	Maier
800	00001002	M110			31.12.9999	01.01.2002	000	M110	1.533,88	EUR	12	Ulrike	Zaucker
800	00001002	M120			31.12.9999	01.01.2002	000	M120	4.601,63	EUR	12	Ulrike	Zaucker
800	00001003	M110			31.12.9999	01.01.2002	000	M110	571,62	EUR	12	Stefan	Pfändli
800	00001003	M120			31.12.9999	01.01.2002	000	M120	1.714,87	EUR	12	Stefan	Pfändli
800	00001004	M110			31.12.9999	01.01.2002	000	M110	514,02	EUR	12	Olaf	Paulsen
800	00001004	M120			31.12.9999	01.01.2002	000	M120	1.542,06	EUR	12	Olaf	Paulsen
800	00001005	M110			31.12.9999	01.01.2002	000	M110	709,84	EUR	12	Hanno	Gutjahr
800	00001005	M120			31.12.9999	01.01.2002	000	M120	2.129,53	EUR	12	Hanno	Gutjahr
800	00001006	M110			31.12.9999	01.01.2002	000	M110	672,69	EUR	12	Yasmin	Awad
800	00001006	M120			31.12.9999	01.01.2002	000	M120	2.018,07	EUR	12	Yasmin	Awad
800	00001007	M110			31.12.9999	01.01.2002	000	M110	789,09	EUR	12	Hanna	Ulrich
800	00001007	M120			31.12.9999	01.01.2002	000	M120	2.367,28	EUR	12	Hanna	Ulrich
800	00001008	M110			31.12.9999	01.01.2002	000	M110	1.619,09	EUR	12	Hilde	Müller
800	00001008	M120			31.12.9999	01.01.2002	000	M120	4.857,27	EUR	12	Hilde	Müller
800	00001009	M110			31.12.9999	01.01.2002	000	M110	2.812,11	EUR	12	Herbert	Braunstein
800	00001009	M120			31.12.9999	01.01.2002	000	M120	8.436,32	EUR	12	Herbert	Braunstein

Wie zu sehen ist, wurden anstelle der Schlüsselfelder die definierten Standardfelder (Join-Layout) übernommen.

[Zum Anfang](#)

Zusätzliche Verbesserungen

Spalten der Ergebnisliste in die Zwischenablage kopieren

Ab Version 3.6E von SE16XXL steht folgende Menüfunktion zur Verfügung:

Liste → *Spalten in die Zwischenablage kopieren*.

Sie kopiert die Werte der markierten Spalten in die Zwischenablage. Falls keine Zeilen markiert wurden, werden die Werte **aller Zeilen** kopiert, ansonsten nur die Werte der **markierten Zeilen**. Wenn mehr als eine Spalte beteiligt ist, werden die Werte einer Zeile durch einen **horizontalen Tabulator** getrennt. Zeilen, in denen alle beteiligten Werte initial sind, werden ignoriert.

Abhängig von der Benutzereinstellung “*Konvertierungsexit berücksichtigen*” wird entweder der **interne** oder der **externe Wert** eines bestimmten Feldes kopiert. Nicht alle Datentypen werden unterstützt, z. B. werden Spalten vom Typ RAW, LRAW oder RAWSTRING **nicht unterstützt**. Zeichentypen (CHAR, NUMC, LCHR, STRING) werden bis zu einer Länge von **255** Zeichen kopiert.

Ein paar Beispiele werden nun die Funktionsweise dieser neuen Funktion veranschaulichen.

Beispiel 1 – Spalten der Tabelle KNA1 kopieren

Wir beginnen mit einer Ergebnisliste von Einträgen der Tabelle **KNA1** und markieren die Spalten KUNNR, NAME1, PSTLZ und ORT01 (in dieser Reihenfolge):

SE16XXL - Tabelle KNA1 - 2000 Einträge selektiert

Tabelle KNA1 - Kundenstamm (allgemeiner Teil)

MANDT	KUNNR	LAND1	NAME1	NAME2	ORT01	PSTLZ	REGIO	SORTL	STRAS
800	0000000001	US	Nelson Tax & Associates		PHILADELPHIA	19115	PA		1 1 113 Germantown Ave
800	0000000002	DE	Wett		Walldorf	69190	08	WETT	Astorstrasse 34
800	0000000099	DE	Einmalkunde					HRTEM	
800	0000000110	DE	Auto Klement	Exclusive Automobile	München	81737	09	AM	Bert-Brecht-Allee 29
800	0000000149	CU	Cuban Automotive Supply SA		Havana	1111		GTS	789 Calle Obispo
800	0000000150	KP	Korean Automotive Supply		Pyongyang	8897		GTS	786 Taedong River Road
800	0000000175	MX	Californian Multi Chemical Company	Mexican Subsidiary	Mexico City	15530	MEX	CHEMICAL	Cd Mexico Piso 3
800	0000000224	US	Reference for R2C Customer		LOS ANGELES	90025	CA	R2C DEFEDE	Baker Street

Dann rufen wir die Menüfunktion *Liste* → *Spalten in die Zwischenablage kopieren* auf. Das Programm reagiert mit folgender Meldung:

Spalte(n) erfolgreich in die Zwischenablage kopiert

Um das Ergebnis zu verifizieren, rufen wir einen Texteditor auf und fügen die kopierten Werte aus der Zwischenablage ein:

```
000000001 Nelson Tax & Associates 19115 PHILADELPHIA
000000002 Wett 69190 Walldorf
000000009 Einmalkunde
000000110 Auto Klement 81737 München
000000149 Cuban Automotive Supply SA 1111 Havana
000000150 Korean Automotive Supply 8897 Pyonyang
000000175 Californian Multi Chemical Company 15530 Mexico City
000000224 Reference for B2C Customer 90025 LOS ANGELES
000000255 Emma Bull 80216 DENVER
000000256 Kenny A Chadburn 80303 BOULDER
000000257 John Evans 80466 NEDERLAND
000000258 Roger Zahn 87123 ALBUQUERQUE
000000259 Laurel K. Hardin 80631 GREELEY
000000260 Chelsea Quinn Yates 87109 ALBUQUERQUE
```

...

ANMERKUNG: Nachdem keine Zeilen markiert wurden, wurden die Werte aller Zeilen kopiert.

Beispiel 2 – Wie oben, jedoch nur für einige markierte Zeilen

In diesem Beispiel markieren wir die gleichen Spalten wie im vorherigen Beispiel und wählen zusätzlich nur **einige Zeilen** der Liste aus:

Tabelle KNA1 - Kundenstamm (allgemeiner Teil)										
MANDT	KUNNR	LAND1	NAME1	NAME2	ORT01	PSTLZ	REGIO	SORTL	STRAS	
800	000000001	US	Nelson Tax & Associates		PHILADELPHIA	19115	PA		1 1 113 Germantown Ave	
800	000000002	DE	Wett		Walldorf	69190	08	WETT	Astorstrasse 34	
800	000000009	DE	Einmalkunde					HRTEM		
800	000000110	DE	Auto Klement	Exclusive Automobile	München	81737	09	AM	Bert-Brecht-Allee 29	
800	000000149	CU	Cuban Automotive Supply SA		Havana	1111		GTS	789 Calle Obispo	
800	000000150	KP	Korean Automotive Supply		Pyonyang	8897		GTS	786 Taedong River Road	
800	000000175	MX	Californian Multi Chemical Company	Mexican Subsidiary	Mexico City	15530	MEX	CHEMICAL	Cd Mexico Piso 3	
800	000000224	US	Reference for B2C Customer		LOS ANGELES	90025	CA	B2C REFERE	Baker Street	
800	000000255	US	Emma Bull		DENVER	80216	CO	B2C BULL	145 145 West 51 th street (TE	
800	000000256	US	Kenny A Chadburn		BOULDER	80303	CO	B2C CHADBU	661 661 28 Street	
800	000000257	US	John Evans		NEDERLAND	80466	CO	B2C EVANS	60 60 Lakeview Drive	
800	000000258	US	Roger Zahn		ALBUQUERQUE	87123	NM	B2C ZAHN	12990 12990 Central Northea	



Markierte Werte erfolgreich in die Zwischenablage kopiert

In diesem Fall ist das Ergebnis:

```
000000110 Auto Klement 81737 München
000000149 Cuban Automotive Supply SA 1111 Havana
000000150 Korean Automotive Supply 8897 Pyonyang
000000255 Emma Bull 80216 DENVER
000000256 Kenny A Chadburn 80303 BOULDER
```

ANMERKUNG: Wie erwartet wurden nur die Werte der markierten Zeilen kopiert.

Beispiel 3 – Spalten der Tabelle CABN kopieren

In diesem Beispiel beginnen wir mit einer Liste der Tabelle **CABN** und markieren die Spalten **ATINN**, **ATNAM** und **V~ATINN_TXT** in dieser Reihenfolge:

Tabelle CABN - Merkmal												
MANDT	ATINN	V~ATINN_TXT	ADZHL	ATNAM	ATIDN	ATFOR	ANZST	ANZDZ	ATVOR	ATSCH	ATK	
800	0000000113	Land	0000	HD_COUNTRY		CHAR	3	0	X			
800	0000000114	Motor	0000	HD_MOTOR		CHAR	4	0	X			
800	0000000115	Farbe	0000	HD_COLOR		CHAR	2	0	X			
800	0000000116	Hinterrad	0000	HD_WHEEL		CHAR	1	0	X			
800	0000000117	Sitz	0000	HD_SEAT		CHAR	2	0	X			
800	0000000118	Zubehör	0000	HD_OPTIONS		CHAR	2	0	X			

Markierte Werte erfolgreich in die Zwischenablage kopiert

Da die Option “*Konvertierungsexit berücksichtigen*” nicht aktiv ist, ergibt sich folgendes Ergebnis des Kopiervorgangs:

```
0000000114      HD_MOTOR      Motor
0000000115      HD_COLOR      Farbe
0000000116      HD_WHEEL      Hinterrad
```

An dieser Stelle aktivieren wir die Option “*Konvertierungsexit berücksichtigen*” – das Feld ATINN zeigt nun den Namen des Merkmals an:

Tabelle CABN - Merkmal												
MANDT	ATINN	V~ATINN_TXT	ADZHL	ATNAM	ATIDN	ATFOR	ANZST	ANZDZ	ATVOR	ATSCH	ATK	
800	HD_COUNTRY	Land		HD_COUNTRY		CHAR	3	0	X			
800	HD_MOTOR	Motor		HD_MOTOR		CHAR	4	0	X			
800	HD_COLOR	Farbe		HD_COLOR		CHAR	2	0	X			
800	HD_WHEEL	Hinterrad		HD_WHEEL		CHAR	1	0	X			
800	HD_SEAT	Sitz		HD_SEAT		CHAR	2	0	X			
800	HD_OPTIONS	Zubehör		HD_OPTIONS		CHAR	2	0	X			

Markierte Werte erfolgreich in die Zwischenablage kopiert

Wenn wir nun die gleichen Werte wie im vorherigen Fall kopieren, erhalten wir:

```
HD_MOTOR      HD_MOTOR      Motor
HD_COLOR      HD_COLOR      Farbe
HD_WHEEL      HD_WHEEL      Hinterrad
```

ANMERKUNG: Wie erwartet, wurde in diesem Fall der **externe Wert** von ATINN kopiert.

Beispiel 4 – Die Felder BRGEW und NTGEW von MARA kopieren

In diesem Beispiel betrachten wir die numerischen Felder **BRGEW** und **NTGEW** der Tabelle MARA. Wir beginnen mit einer Liste von MARA-Einträgen und markieren die Spalten BRGEW und NTGEW, bevor wir die Kopierfunktion aufrufen:

Tabelle MARA - Allgemeine Materialdaten										
MATNR	ERSDA	LAEDA	VPSTA	MTART	MATKL	MEINS	BRGEW	NTGEW	GEWEI	DISST
88	27.05.1997	22.01.2003	KVB	FERT	02004	ST	0,200	0,200	KG	
89	27.05.1997	22.01.2003	KV	FERT	02004	ST	0,200	0,200	KG	
98	11.06.1997	11.06.1997	K	HALB	002	ST	1,120	1	KG	
288	16.11.1999	23.01.2003	A	HALB	00101	ST	5	4	KG	
358	22.02.2001	22.01.2003	KCVBEX	HAWA	002	ST	500	500	G	
359	22.02.2001	22.01.2003	KVEX	HAWA	002	ST	20	10	G	
578	21.05.2002	29.11.2002	KDEALBSVPCQGZX	FERT	00207	ST	16,800	16,800	KG	
598	13.06.2002		V	VERP		ST	4.000	3.999	KG	
599	13.06.2002		V	VERP		ST	4.000	3.999	KG	
679	25.10.2002	23.01.2003	S	FERT	012	ST	32	31	KG	

Das Ergebnis ist dann:

```
0,200    0,200
1,120    1,000
20,000   10,000
16,800   16,800
4000,000      3999,000
```

ANMERKUNG: Die Werte wurden ohne Berücksichtigung der Maßeinheit kopiert.

Nun markieren wir auch die Gewichtseinheit (**GEWEI**) und wiederholen die Operation:

Tabelle MARA - Allgemeine Materialdaten										
MATNR	ERSDA	LAEDA	VPSTA	MTART	MATKL	MEINS	BRGEW	NTGEW	GEWEI	DISST
88	27.05.1997	22.01.2003	KVB	FERT	02004	ST	0,200	0,200	KG	
89	27.05.1997	22.01.2003	KV	FERT	02004	ST	0,200	0,200	KG	
98	11.06.1997	11.06.1997	K	HALB	002	ST	1,120	1	KG	
288	16.11.1999	23.01.2003	A	HALB	00101	ST	5	4	KG	
358	22.02.2001	22.01.2003	KCVBFX	HAWA	002	ST	500	500	G	

Diesmal wurde die Maßeinheit berücksichtigt:

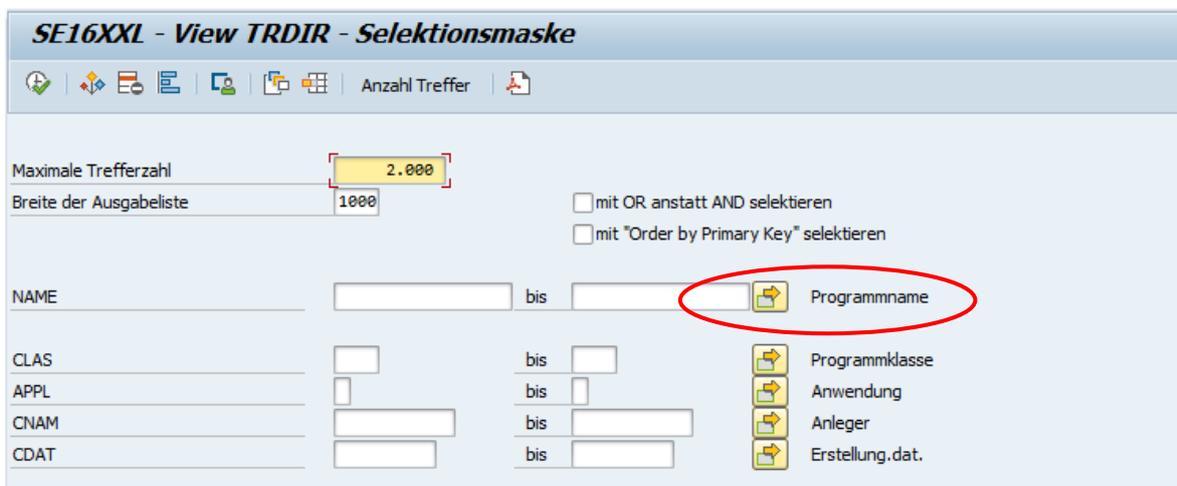
```
0,200    0,200    KG
1,120    1        KG
20        10        G
16,800   16,800   KG
4.000    3.999    KG
```

Mehrfachselektionen als Datei exportieren

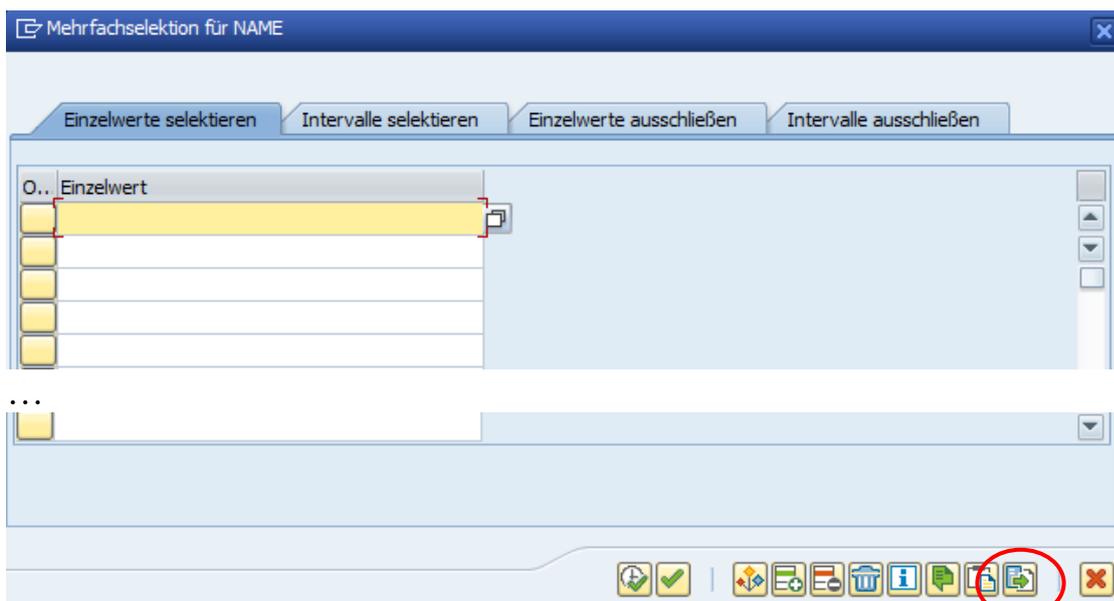
Im Dialogfenster “Mehrfachselektion” ist es nun möglich, die angegebenen Werte in eine Textdatei zu exportieren. Ein kurzes Beispiel wird diese neue Funktion verdeutlichen.

Beispiel

Angenommen, Sie selektieren Einträge der View **TRDIR** aus, die Informationen über Programme beinhaltet. Die Selektionsmaske könnte folgendermaßen aussehen:

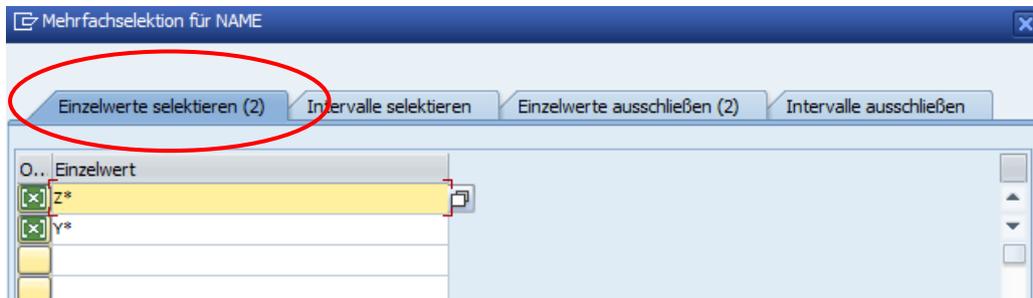


Wenn Sie auf die Schaltfläche  rechts neben NAME klicken, erhalten Sie das Dialogfenster **Mehrfachselektion**:

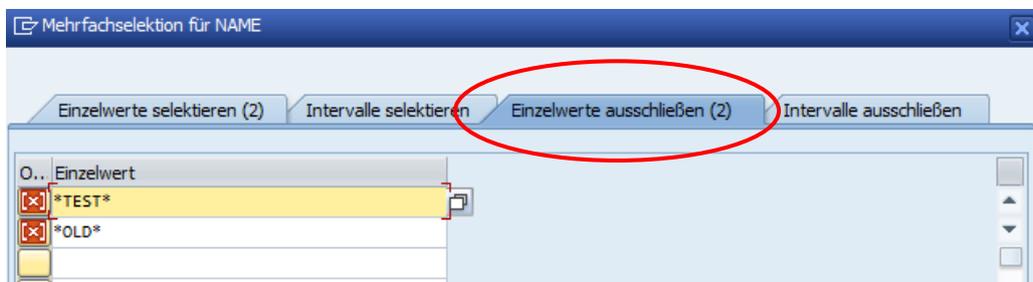


Die neue Funktion ist  (Als Datei exportieren).

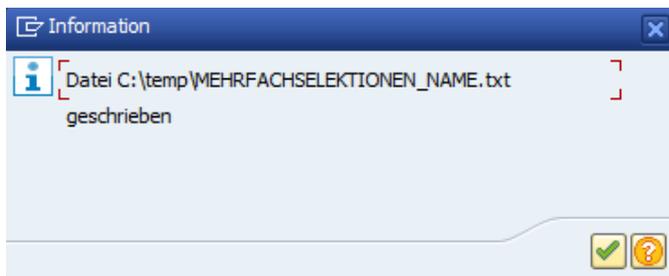
Nun geben wir einige Selektionswerte ein, um zu zeigen, wie die Exportfunktion arbeitet:



Und auch einige Ausschlusswerte:



Nun machen wir von der Funktion “*Als Datei exportieren*” () Gebrauch. Nach Angabe des Namens und des Verzeichnisses für die Datei erscheint folgende Meldung:



Der Inhalt der Datei in einem Texteditor lautet wie folgt:

```
I      CP      Z*
I      CP      Y*
E      CP      *TEST*
E      CP      *OLD*
```

Wie zu sehen ist, wurden nicht nur die Selektionswerte, sondern auch deren Attribute exportiert. Das heißt, wenn diese Datei mit der Funktion “Import aus Textdatei” () importiert wird, werden alle Arten von Selektionswerten geladen.

ANMERKUNG: Diese Funktion wurde als Support-Package für Version 3.6D freigegeben.

Programm /TFTO/SFMI_AUTO_REFRESH

Es kann Scripts geben, die eine Liste mit stark variablen Daten erzeugen. Mit anderen Worten, wenn die Ergebnisliste solcher Scripts nach kurzer Zeit aktualisiert wird, weicht das neue Ergebnis häufig vom vorherigen ab, weil zu den beteiligten Tabellen neue Einträge hinzugefügt oder bestehende Einträge geändert oder sogar gelöscht wurden.

Falls diese Änderungen überwacht werden sollen, kann man das entsprechende Script ausführen und dann die Ergebnisliste regelmäßig aktualisieren, um die neueste Situation zu erhalten. Dieses Verfahren ist jedoch zeitaufwändig und ineffizient.

Aus diesem Grund ist ein neues Programm (**/TFTO/SFMI_AUTO_REFRESH**) implementiert worden. Es führt über die SFMI-Schnittstelle (Script Function Module Interface) ein Script aus. Dabei kann festgelegt werden, dass die Ergebnisliste in regelmässigen Abständen zwischen **1** und **9999** Sekunden **automatisch** aktualisiert wird. Auf diese Weise ist es möglich, bestimmte Datensituationen zu überwachen, ohne manuell eingreifen zu müssen, um die Liste zu aktualisieren.

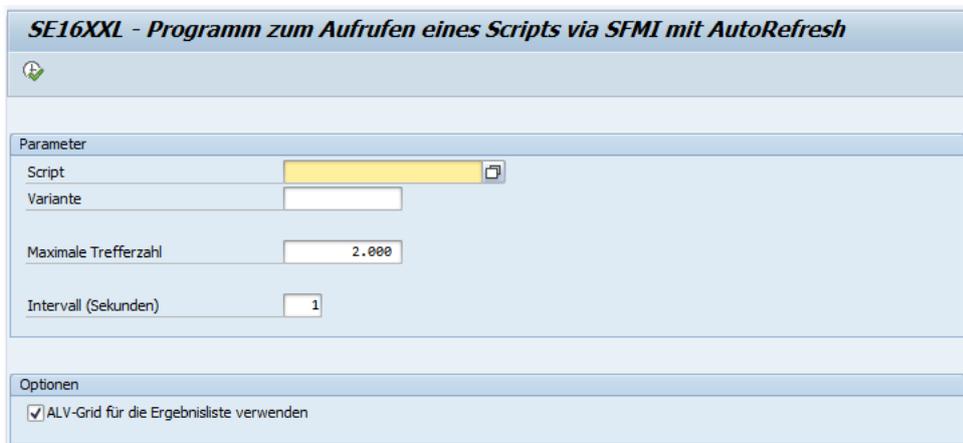
ANMERKUNG: Dieses Programm wurde als Support-Package für Version 3.6D freigegeben.

Aufruf des Programms

Verwenden Sie die Transaktion **/TFTO/SFMI_AUTO_REFR**, um das Programm aufzurufen.

Selektionsmaske

Die Selektionsmaske ist recht einfach:



The screenshot shows the selection mask for the program SE16XXL. The title bar reads "SE16XXL - Programm zum Aufrufen eines Scripts via SFMI mit AutoRefresh". Below the title bar, there is a "Parameter" section with the following fields:

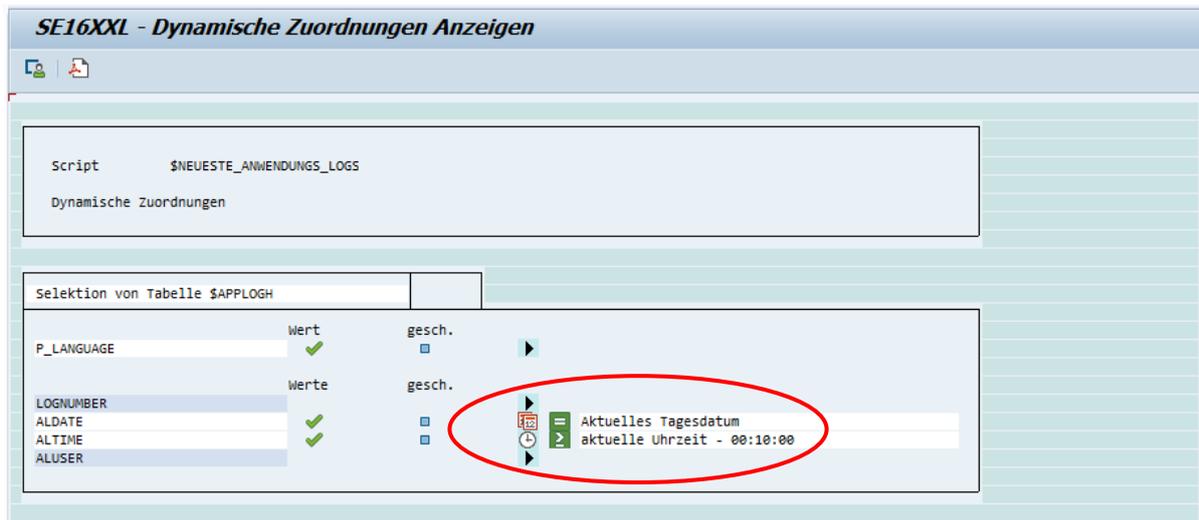
- Script: A text input field with a yellow background and a small icon to its right.
- Variante: A text input field.
- Maximale Trefferzahl: A text input field containing the value "2.000".
- Intervall (Sekunden): A text input field containing the value "1".

Below the "Parameter" section, there is an "Optionen" section with a single checked checkbox labeled "ALV-Grid für die Ergebnisliste verwenden".

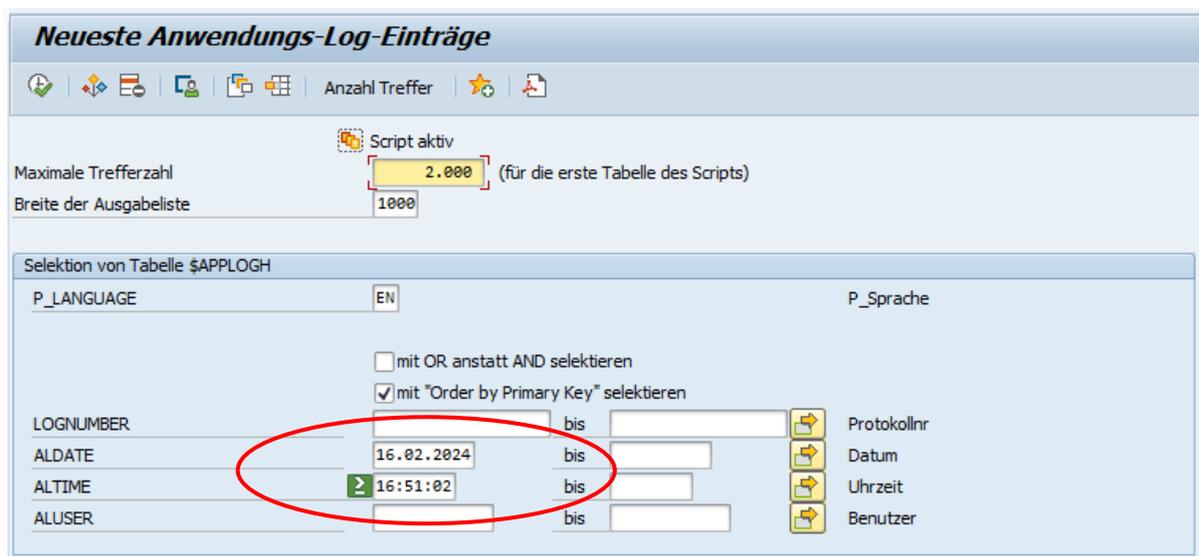
Die Funktionsweise dieses Programms soll nun anhand eines Beispiels veranschaulicht werden.

Beispiel

In diesem Beispiel verwenden wir das Script **\$NEUESTE_ANWENDUNGS_LOGS**, das die Anwendungs-Logs (**\$APPLOGH** und **\$APPLOGS**) der letzten 10 Minuten selektiert und anzeigt. Dies wird durch dynamische Zuordnungen erreicht:



Auf diese Weise selektieren die Selektionskriterien automatisch die Einträge des aktuellen Datums und der letzten 10 Minuten. Zum Beispiel:



ANMERKUNG: Das Programm /TFTO/SFMI_AUTO_REFRESH ruft das Script auf, **ohne** die Selektionsmaske anzuzeigen. Änderungen an den voreingestellten Selektionswerten können nur durch die Angabe einer entsprechenden **Variante** erreicht werden. In unserem Beispiel ist dies aufgrund der dynamischen Zuordnungen nicht notwendig.

Nun führen wir das Programm mit folgenden Parametern aus:

SE16XXL - Programm zum Aufrufen eines Scripts via SFMI mit AutoRefresh

Parameter

Script: \$NEUESTE_ANWENDUNGS_LOGS

Variante:

Maximale Trefferzahl:

Intervall (Sekunden):

Optionen

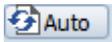
ALV-Grid für die Ergebnisliste verwenden

Die Ergebnisliste wird angezeigt:

Script \$NEUESTE_ANWENDUNGS_LOGS - 3 Einträge

Join von ... \$APPLOGS(B) Script: \$NEUESTE_ANWENDUNGS_LOGS - Neueste Anwendungs-Log-Einträge 16.02.2024 - 17:02:45

A~LOGNUMBER	A~OBJECT	A~SUBOBJECT	A~ALDATE	A~ALTIME	B~MSGNUMBER	B~MSGTY_ICON	B~MSG_TEXT
00000000000013912498	WF	WIERRE	16.02.2024	17:00:09	000001		'0' entries have been edited
00000000000013912497	WF	WIERRE		16:57:09			'0' entries have been edited
00000000000013912496	WF	WIERRE		16:54:09			'0' entries have been edited

Um die Funktion “**Auto-Refresh**” zu aktivieren, müssen wir die Schaltfläche  auf der Anwendungsleiste betätigen. Ab diesem Moment wird die Liste automatisch alle 10 Sekunden aktualisiert:

Script \$NEUESTE_ANWENDUNGS_LOGS - 3 Einträge

Join von ... \$APPLOGS(B) Script: \$NEUESTE_ANWENDUNGS_LOGS - Neueste Anwendungs-Log-Einträge 16.02.2024 - 17:05:41

A~LOGNUMBER	A~OBJECT	A~SUBOBJECT	A~ALDATE	A~ALTIME	B~MSGNUMBER	B~MSGTY_ICON	B~MSG_TEXT
00000000000013912499	WF	WIERRE	16.02.2024	17:03:09	000001		'0' entries have been edited
00000000000013912498	WF	WIERRE	16.02.2024	17:00:09	000001		'0' entries have been edited
00000000000013912497	WF	WIERRE	16.02.2024	16:57:09	000001		'0' entries have been edited

Eine weitere Ergebnisliste:

Script \$NEUESTE_ANWENDUNGS_LOGS - 4 Einträge								
 Stop								
Join von ... \$APPLG(S)			Script: \$NEUESTE_ANWENDUNGS_LOGS - Neueste Anwendungs-Log-Einträge				16.02.2024 - 17:06:55	
A~LOGNUMBER	A~OBJECT	A~SUBOBJECT	A~ALDATE	A~ALTIME	B~MSGNUMBER	B~MSGTY_ICON	B~MSG_TEXT	
00000000000013912500	WF	WIERRE	16.02.2024	17:06:09	000001		'0' entries have been edited	
00000000000013912499	WF	WIERRE	16.02.2024	17:03:09	000001		'0' entries have been edited	
00000000000013912498	WF	WIERRE	16.02.2024	17:00:09	000001		'0' entries have been edited	
00000000000013912497	WF	WIERRE	16.02.2024	16:57:09	000001		'0' entries have been edited	

Und noch eine:

Script \$NEUESTE_ANWENDUNGS_LOGS - 4 Einträge								
 Stop								
Join von ... \$APPLG(S)			Script: \$NEUESTE_ANWENDUNGS_LOGS - Neueste Anwendungs-Log-Einträge				16.02.2024 - 17:09:12	
A~LOGNUMBER	A~OBJECT	A~SUBOBJECT	A~ALDATE	A~ALTIME	B~MSGNUMBER	B~MSGTY_ICON	B~MSG_TEXT	
00000000000013912501	WF	WIERRE	16.02.2024	17:09:09	000001		'0' entries have been edited	
00000000000013912500	WF	WIERRE	16.02.2024	17:06:09	000001		'0' entries have been edited	
00000000000013912499	WF	WIERRE	16.02.2024	17:03:09	000001		'0' entries have been edited	
00000000000013912498	WF	WIERRE	16.02.2024	17:00:09	000001		'0' entries have been edited	

ANMERKUNG: Wenn das Script eine leere Liste zurückgibt, stoppt das Programm nicht, sondern zeigt stattdessen eine leere Ergebnisliste an. Dieses Verhalten ist notwendig, da “keine Treffer” ein mögliches Ergebnis in dieser Art von Umgebung ist, und es könnte danach von einer nicht leeren Liste gefolgt werden.

[Zum Anfang](#)

Werteilfe (F4-Hilfe) für Joins

Bisher gab es bei einem **Join** oder einer “**Selektieren für alle Einträge**”-Operation keine spezielle Werteilfe, um die “richtige” Datenbanktabelle oder View zu finden. Es war nur die Standard-Werteilfe verfügbar, und sie ist zu allgemein, um eine wirkliche Hilfe zu sein. Mit anderen Worten, es war mehr oder weniger notwendig, die Tabelle zu “kennen”, die für die Join-Operation ausgewählt werden sollte.

Um diese Situation zu entschärfen, wurde ab Version **3.6D** von SE16XXL ein neuer Satz von Wertehilfen (für **Datenbanktabellen, Views** und **CDS-View-Entitäten**) implementiert, um diese Art von Operation zu erleichtern. Diese Wertehilfen schlagen nur die Datenbanktabellen oder Views vor, die auf die eine oder andere Weise mit den Tabellen und/oder Views verbunden sind, die Teil der aktuellen Ergebnisliste sind. Neben dem Namen und der Kurzbeschreibung der verschiedenen Tabellen/Views listen sie auch die notwendigen **Join-Kriterien** auf und geben damit eine Vorstellung von deren Bedeutung.

Ein Beispiel wird diese neue Funktionalität veranschaulichen.

Beispiel

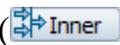
Wir beginnen mit einer Ergebnisliste von Tabelle **VBAK** (Verkaufsbeleg: Kopfdaten):

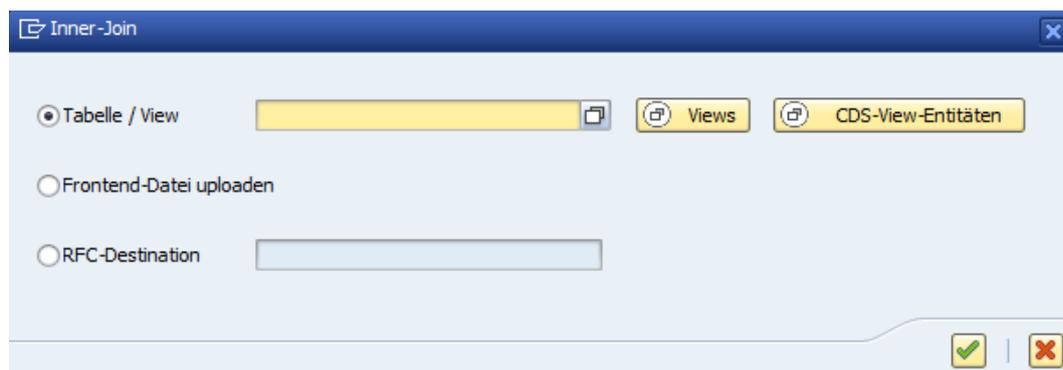


SE16XXL - Tabelle VBAK - 63 Einträge selektiert

Tabelle VBAK - Verkaufsbeleg: Kopfdaten

MANDT	VBELN	ERDAT	ERZET	VBTYP	TRVOG	AUART	AUGRU	WAERK	VKORG	VTWEG	SPART	VKGRP	VKBUR	GSBE
800	0000000012	04.05.2016	17:33:22	C	0	TA		EUR	0001	01	01			
800	0000000013	29.09.2016	16:27:39	C	0	TA		EUR	0001	01	01			
800	0000000014	26.06.2017	10:47:28	C	0	TA		EUR	0001	01	01	001	0001	
800	0000000016	13.03.2017	10:23:15	C	0	TA		EUR	0001	01	01			

Wir markieren die Spalte “**VBELN**” und führen dann einen Inner-Join durch (). Es erscheint das übliche Dialogfenster zum Festlegen der zu joinenden Tabelle/View:



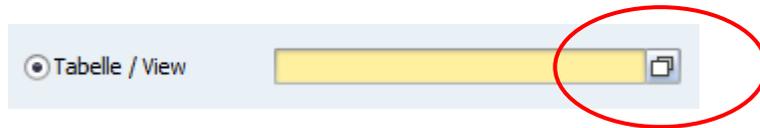
Inner-Join

Tabelle / View

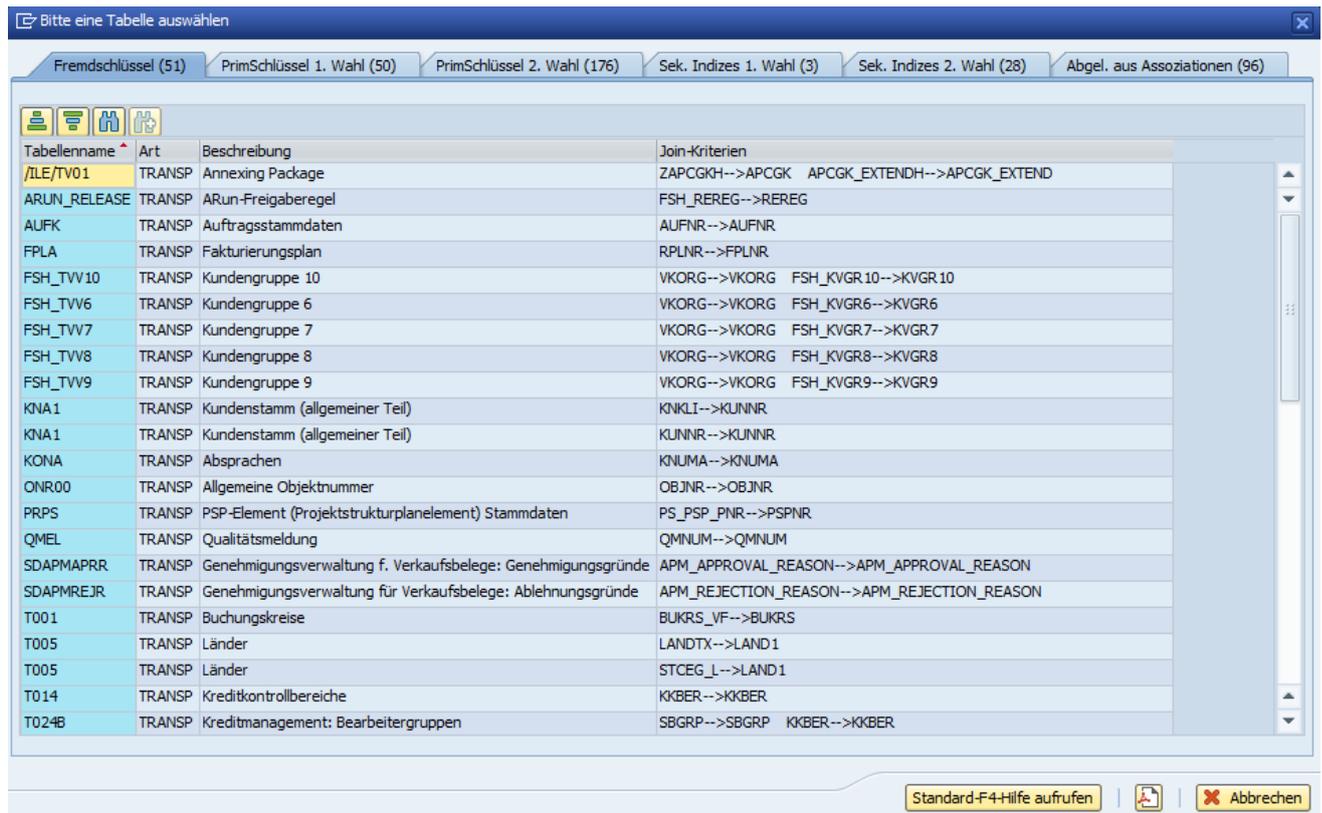
Frontend-Datei uploaden

RFC-Destination

Um eine passende **Datenbanktabelle** zu finden, bedienen wir uns der Wertehilfe, die direkt mit dem Eingabefeld verknüpft ist:



Das System reagiert mit der Anzeige folgendes Dialogfensters mit **Registerkarten**:



Von den sechs Registerkarten bieten die erste (**Fremdschlüssel**) und die letzte (**Abgel. aus Assoziationen**) eine Liste von Tabellen unabhängig von den markierten Spalten. Die übrigen Registerkarten sind nur vorhanden, wenn **mindestens eine Spalte** der ursprünglichen Liste markiert wurde. Dies geschieht, um die Zahl der Vorschläge auf eine überschaubare Zahl zu beschränken.

Je nach Situation können einige der Registerkarten fehlen, wenn keine geeigneten Tabellen dieser Art ermittelt werden konnten. Wenn überhaupt keine Tabellen gefunden werden, wird die Standard-Wertehilfe ausgegeben.

Weitere Informationen zur Wertehilfe für Joins mit Datenbanktabellen finden Sie in [F4-Hilfe für Joins mit Datenbanktabellen](#).

Wenn wir anstelle einer Datenbanktabelle eine geeignete **View** finden möchten, können wir die **zweite Schaltfläche** im Dialogfenster verwenden:



In diesem Fall würde das Dialogfenster mit Registerkarten wie folgt aussehen:

View-Name	Typ	Datendefinition	Beschreibung	Join-Kriterien	Datenbank-Tabellen
/CPD/WFC_SALES	D		Sicht für Kundenauftrag	VBELN-->VBELN	VBAP
/DMBE/CSSUBLOC	D	/DMBE/C_SALES_LOCS_AND_SUBLOCS	Sales contract locations, sublocations	VBELN-->VBELN	OIJ_EL_CP_LOC, OIJ_EL_DOC_MOT, VBAP
/DMBE/ISALESLOC	D	/DMBE/I_SALES_CONTRACT_LOC	Locations ID of sales contracts	VBELN-->VBELN	OIJ_EL_CP_LOC, OIJ_EL_DOC_MOT, VBAP
/DMBE/ISDOCITEM	D	/DMBE/I_SALES_DOC_ITEM_DATA	Sales Document: Item Data	VBELN-->VBELN	VBAP
/SAPAPO/V_TDL_SL	D	/SAPAPO/V_TDL_SO_SCHED_LINE	SAPAPO TDL: SO Scheduling line	VBELN-->DOC_NO	VBEP
/SAPCEM/V_VBMAN	D		Manuell erfasste Verkaufsbelege	VBELN-->VBELN	J_3GVBAP, VBKD
AARNPRIOBSC	D	A_ARUNPRIORITY	Item	VBELN-->REQUIREMENTDOCUMENTNUMBER	DDDDLCHARTYPES, VBAP
ACMRITMPCRGELMNT	D	A_CREDITMEMOREQITEMPCRGELMNT	Item Pricing Element	VBELN-->CREDITMEMOREQUEST	DDDDLCHARTYPES, PRCD_ELEMENTS, T006,
ACMPBPCRGELMNT	D	A_CREDITMEMOREFPBPCRGELMNT	Header Pricing Element	VBELN-->CREDITMEMOREQUEST	DDDDLCHARTYPES, PRCD_ELEMENTS, VBAP,

Wie man sehen kann, enthält die Liste nicht nur die **Join-Kriterien**, sondern auch den Namen der **Datendefinition** (wenn es sich um eine DDIC-basierte CDS-View handelt) und die **Datenbanktabellen**, auf denen die View basiert.

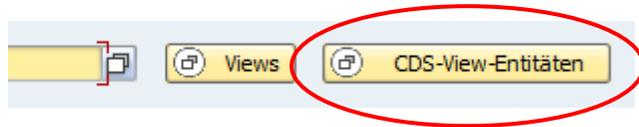
Wenn eine CDS-View mit Assoziationen Teil der vorliegenden Ergebnisliste ist, erscheint im Dialogfenster eine dritte Registerkarte mit allen **Assoziationen** dieser CDS-View. Beginnend mit einer Ergebnisliste von CDS-View **ISDSALESDOC** würde die F4-Hilfe für Views wie folgt aussehen:

Markiert	View-Name	Typ	Datendefinition	Assoziation	Beschreibung	Join-Kriterien
→	I_SALESDOCUMENTBILLINGPLAN	E	I_SALESDOCUMENTBILLINGPLAN	_BILLINGPLAN	Billing Plan of Sales Document	SALESDOCUMENT-
→	I_SALESDOCUMENTPRECEDPROCFLOW	E	I_SALESDOCUMENTPRECEDPROCFLOW	_PRECEDINGPROCFLOWDOC	Preceding Process Flow of Sales Document	SALESDOCUMENT-
→	I_SALESDOCUMENTSUBSQNTPROCFLOW	E	I_SALESDOCUMENTSUBSQNTPROCFLOW	_SUBSEQUENTPROCFLOWDOC	Subsequent Process Flow of Sales Doc	SALESDOCUMENT-
→	IEPPROJITEM	D	I_ENGAGEMENTPROJECTITEM	_ENGAGEMENTPROJECTITEM	Item of Engagement Project	SALESDOCUMENT-
→	ISDSALESDOCITEM	D	I_SALESDOCUMENTITEM	_ITEM	Sales Document Item	SALESDOCUMENT-
→	ISDSLSDOCPREL	D	I_SALESDOCUMENTPRICINGELEMENT	_PRICINGELEMENT	Sales Document Pricing Element	SALESDOCUMENT-
→	ISDSLSDOCPART	D	I_SALESDOCUMENTSTANDARDPARTNER	_STANDARDPARTNER	Sales Document Standard Partner	SALESDOCUMENT-
	I_CUSTOMER_CDS	D	I_CUSTOMER	_CUSTOMERCREDITACCOUNT	Customer	CUSTOMER.CREDIT
	I_CUSTOMER_CDS	D	I_CUSTOMER	_SOLDTOPARTY	Customer	SOLDTOPARTY-->
	ICONTRDPSTS	D	I_CONTRACTDOWNPAYMENTSTATUS	_DOWNPAYMENTSTATUS	Value help of contract down payment status	CONTRACTDOWNP
	ICUSTPOTYPE	D	I_CUSTOMERPURCHASEORDERTYPE	_CUSTOMERPURCHASEORDERTYPE	Customer Purchase Order Type	CUSTOMERPURCH

Die Views, die mit der markierten Spalte (**SALESDOCUMENT**) übereinstimmen, werden durch ein Pfeilsymbol (→) auf der linken Seite hervorgehoben.

Weitere Informationen zur Werthilfe für Joins mit Views finden Sie unter [F4-Hilfe für Joins mit Views](#).

Wenn wir schließlich eine geeignete **CDS-View-Entität** finden möchten, können wir die **dritte Schaltfläche** im Dialogfenster nutzen:



Diese letzte Werthilfe schlägt passende CDS-View-Entitäten vor, wie in folgender Abbildung dargestellt:

Bitte eine CDS-View auswählen

PrimSchlüssel 1. Wahl (57) PrimSchlüssel 2. Wahl (378)

CDS-View-Entität	Beschreibung	Join-Kriterien
A_DIGITALVEHICLEQUOTATIONITEM	Quotation	VBELN-->SALESQUOTATION
A_DIGITALVEHICLESALLESORDERITEM	Sales Order	VBELN-->SALESORDER
A_DIGITALVEHICLESORDITEM_2	Digital Vehicle Sales Order	VBELN-->SALESDOCUMENT
A_DIGITALVEHQTNCONFIGN	Digital Vehicle Quotation Config	VBELN-->SALESDOCUMENT
A_DIGITALVEHQTNCONFIGNVAL	Digital Vehicle Quotation Config	VBELN-->SALESDOCUMENT
A_DIGITALVEHQUTATIONITEM_2	Vehicle Quotation	VBELN-->SALESQUOTATION
A_DIGITALVEHLSORDCONFIGN	Digital Vehicle Sales Order Config	VBELN-->SALESDOCUMENT
A_DIGITALVEHLSORDCONFIGNVAL	Digital Vehicle Sales Order Config	VBELN-->SALESDOCUMENT
C_CUSTOMERCREDITLIMITDETAILS	Credit Limit Details in SalesDoc context	VBELN-->SALESDOCUMENT
C_CUSTRETITEMPRCGELMNTMNGTP	Returns Order Item Pricing Element	VBELN-->CUSTOMERRETURN
C_CUSTRETITEMPRCGELMNTMANAGETP	Returns Order Pricing Element - TP	VBELN-->CUSTOMERRETURN
C_RFM_PRIVSNLSLSCONTRHDR	Provisional Sales Contract Header	VBELN-->SALESORDER
C_RFM_PRIVSNLSLSCONTRREF	Provisional Sales Contract Reference	VBELN-->SALESORDER
C_RFM_PRIVSNLSLSCONTRVARITEMS	Provisional Sales Contract Variant Items	VBELN-->SALESORDER
C_RFM_SLSORDPRGRSMNTRHDR	Sales Order	VBELN-->SALESORDER
C_SALESORDERSCHEDULELINEMANAGE	Sales Order Schedule Line	VBELN-->SALESORDER
C_SLSDOCSTDPARTNERCONTACTINFO	Sales Doc Standard Partner Contact Info	VBELN-->SALESDOCUMENT
C_SLSVOLUMENETAMTYEAR20DATE	Sales Volume Net Amount Year To Date	VBELN-->SALESDOCUMENT
CIC_SALESDOCUMENTPARTNER	Projection on Sales Doc. Partner (Industry Cloud Solutions)	VBELN-->SALESDOCUMENT
CIC_SALESDOCUMENTSBLOCKEDBP	Sales Document with Blocked BP (Industry Cloud Solutions)	VBELN-->SALESDOCUMENT
I_CUSTRETITEMPRCINGELEMENTTP	Returns Order Item Pricing Element - TP	VBELN-->CUSTOMERRETURN
I_CUSTRETITEMPRCINGELEMENTTP	Returns Order Pricing Element - TP	VBELN-->CUSTOMERRETURN

Standard-F4-Hilfe aufrufen |  |  Abbrechen

Beachten Sie, dass weder der Name der Datendefinition noch die zugrunde liegenden Datenbanktabellen vorhanden sind. Dies liegt daran, dass bei CDS-View-Entitäten der Name der View und der Datendefinition identisch sind. Darüber hinaus sind keine Informationen über die Datenbanktabellen, auf denen sie basieren, ohne weiteres verfügbar.

Wie im Falle der F4-Hilfe für Views, falls die Ergebnisliste eine CDS-View wie ISDSALESDOC enthalten würde, wäre auch eine **dritte Registerkarte mit allen Assoziationen** vorhanden, wie auf der vorherigen Seite gezeigt.

Weitere Informationen zur F4-Hilfe für Joins mit CDS-View-Entitäten finden Sie in der [F4-Hilfe für Joins mit CDS-View-Entitäten](#).

[Zum Anfang](#)

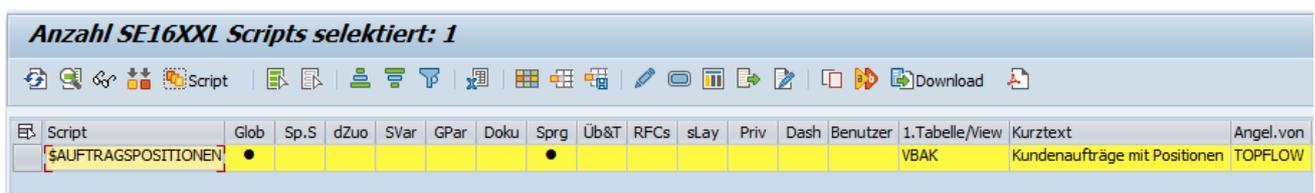
Werteilfe beim Ersetzen einer Tabelle im Script Editor

Wenn ein Script im Script-Editor bearbeitet wird, ist es unter anderem möglich, die Tabelle/View einer bestimmten SELECT-Operation zu **ersetzen**. Bis dato stand nur die Standard-Werteilfe für die Suche nach einem adäquaten Ersatz zur Verfügung. Ab Version **3.6D** von SE16XXL wurde **eine neue Werteilfe** implementiert, die nur solche Tabellen/Views vorschlägt, die mit der zu ersetzenden Tabelle/View **kompatibel** sind.

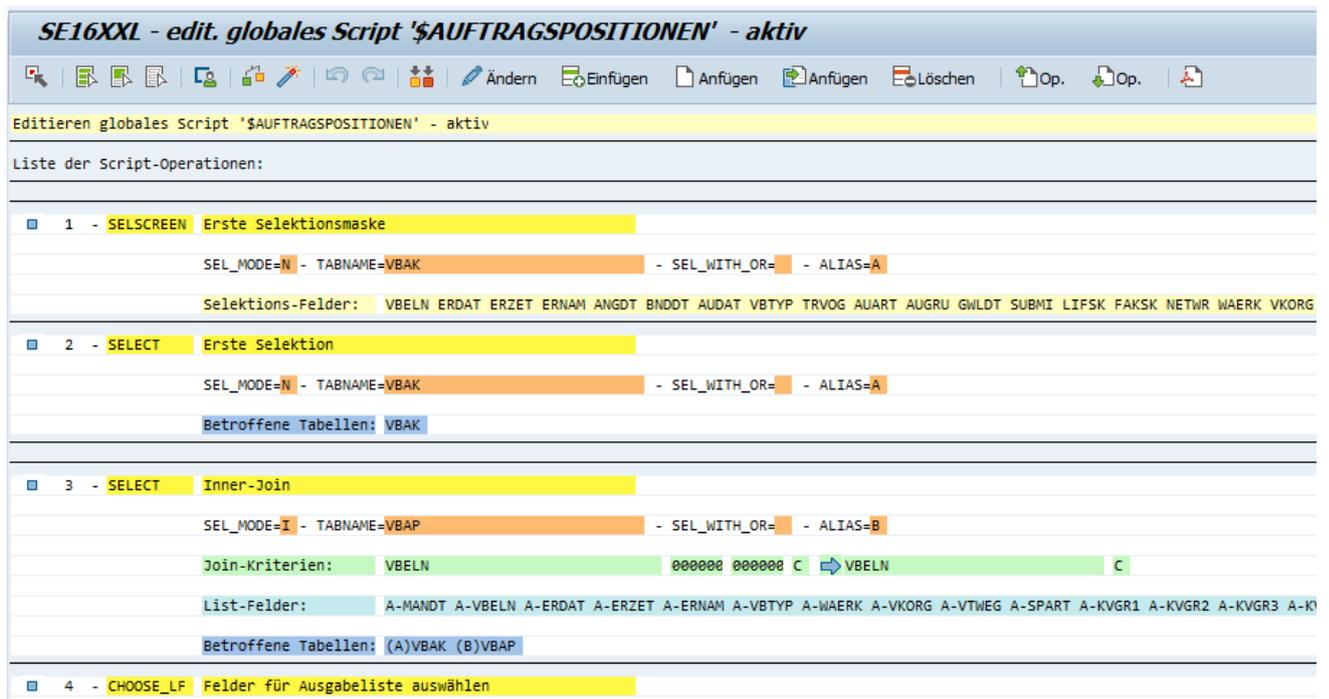
ANMERKUNG: Diese neue Werteilfe erscheint nur, wenn die **Primärschlüssel** der zu ersetzenden Tabelle/View als **Join-Kriterien** verwendet werden. Dies bedeutet auch, dass sie für die **initiale SELECT-Operation** eines Scripts **nicht verfügbar ist**. In all diesen Situationen steht nur die Standard-Werteilfe zur Verfügung.

Diese Funktionalität werden wir nun anhand eines Beispiels verdeutlichen.

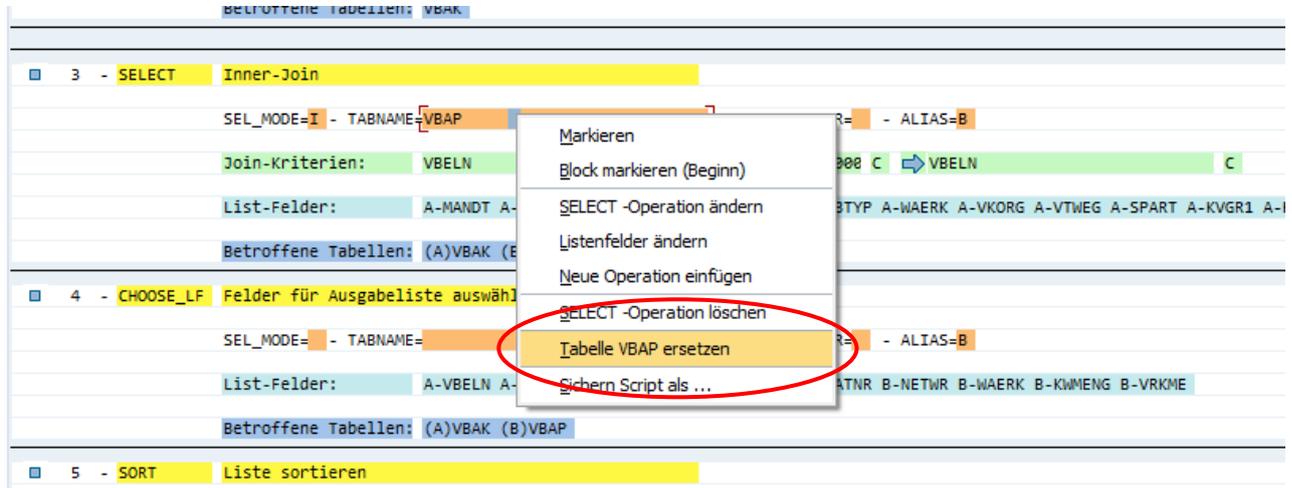
Wir beginnen mit der Bearbeitung von folgendem Scripts im Script-Katalog:



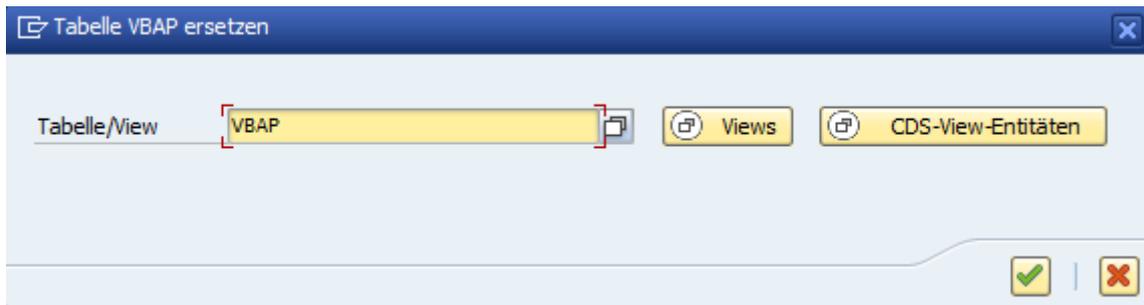
Wir betätigen  auf die Anwendungsleiste und erhalten eine Liste der Script-Operationen:



Unser Ziel ist es, die Tabelle **VBAP** in der **zweiten** SELECT-Operation zu ersetzen. Um dies zu erreichen, setzen wir den Cursor auf den Namen der Tabelle (VBAP) und wählen über das **Kontextmenü** die entsprechende Funktion:

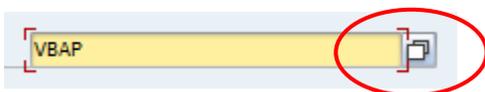


Das Programm reagiert mit folgendem Dialogfenster:



An dieser Stelle stehen **drei** Wertehilfen zur Verfügung:

Die erste ist diejenige, die direkt an das Eingabefeld angehängt ist. Sie liefert eine Reihe von Vorschlägen für Datenbanktabellen:



Die zweite ( Views) schlägt eine Reihe von Views vor.

Die dritte ( CDS-View-Entitäten) liefert eine Liste von CDS-View-Entitäten.

Auf den nächsten Seiten werden wir kurz auf alle drei Arten eingehen.

Wertehilfe für Ersatz-Datenbanktabellen

In diesem Fall erhalten wir für unser Beispiel folgende Liste von Vorschlägen:

Bitte eine Ersatz-Tabelle auswählen

PrimSchlüssel 1. Wahl (49) PrimSchlüssel 2. Wahl (176)

Tabellenname	Art	Beschreibung	Join-Kriterien
/DSD/ES_ORD	TRANSP	DSD Auftragssignaturen	VBELN-->VBELN
/DSD/SV_RL	TRANSP	DSD BT endgültige Abladebelege	VBELN-->VBELN
/SAPSL/SD0ASTS	TRANSP	SD-Beleg Statusaktualisierung von SAP GTS fehlgeschlagen	VBELN-->VBELN
ARBERP_D_SI_SO	TRANSP	Inbound Sales Order versioning for inbound messages	VBELN-->VBELN
ARBERP_D_SO_CONF	TRANSP	SD order confirmation versioning table for outbound messages	VBELN-->VBELN
BOSPS_ASSIGNMENT	TRANSP	LV-PS: PS-Objekt verweist auf spezifizierten Verkaufsbeleg	VBELN-->VBELN
CMM_MTM_SO	TRANSP	MTM pro Kundenauftrag	VBELN-->VBELN
CNVCMIS_M_VMPAB	TRANSP	CMIS: Temp. Generated Pooltable for Matchcode ID VMPA B	VBELN-->VBELN
COPA_GUID_MAP	TRANSP	Mapping table between CRM GUIDS and CO-PA database struct.	VBELN-->VBELN
DAC_D_SDOC	TRANSP	Digital access for sales document related instances	VBELN-->VBELN
EEWA_TCRB_P	TRANSP	Container Rental Billing Periods	VBELN-->VBELN
EEWA_TCRB_P_HIST	TRANSP	Container Rental Billing Periods History	VBELN-->VBELN
EWAPREMSD	TRANSP	Zuordnung Anfallstelle <-> Vertriebsbelegposition	VBELN-->VBELN
FINS_PR_ACNL_SO	TRANSP	Rechnungswesenmitteilung eines lokalen Vertriebsbelegs	VBELN-->VBELN
FINS_TRR_ALLOC	TRANSP	IFRS15: Zugeordneter Erlös	VBELN-->SALES_ORDER
FRE_SO_POINTER	TRANSP	Änderungszeiger für geänderte Kundenauftragsbelege	VBELN-->VBELN
FSH_MSO	TRANSP	MEA-Daten für Verkaufsbeleg	VBELN-->VBELN
FSH_SLOG_SD	TRANSP	Saisonermittlungsprotokoll: Ablage für Verkaufshelen	VBELN-->VBELN

Wir wählen folgende Tabelle als Ersatz für VBAP:

EWAPREMSD	TRANSP	Zuordnung Anfallstelle <-> Vertriebsbelegposition	VBELN-->VBELN
FINS_PR_ACNL_SO	TRANSP	Rechnungswesenmitteilung eines lokalen Vertriebsbelegs	VBELN-->VBELN
FINS_TRR_ALLOC	TRANSP	IFRS15: Zugeordneter Erlös	VBELN-->SALES_ORDER
FRE_SO_POINTER	TRANSP	Änderungszeiger für geänderte Kundenauftragsbelege	VBELN-->VBELN
FSH_MSO	TRANSP	MEA-Daten für Verkaufsbeleg	VBELN-->VBELN
FSH_SLOG_SD	TRANSP	Saisonermittlungsprotokoll: Ablage für Verkaufshelen	VBELN-->VBELN

Daraufhin ändert sich die zweite SELECT-Operation entsprechend:

Betroffene Tabellen: VBAP

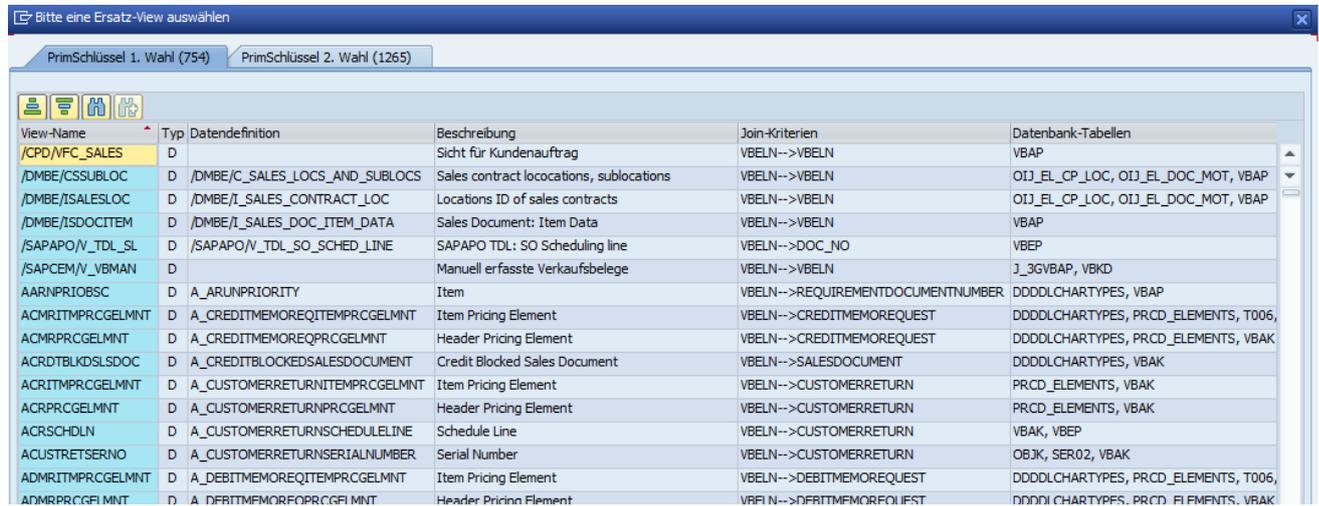
3	- SELECT	Inner-Join
		SEL_MODE=I - TABNAME=FINS_TRR_ALLOC - SEL_WITH_OR= - ALIAS=B
		Join-Kriterien: VBELN 000000 000000 C ⇨ SALES_ORDER C
		List-Felder: A-MANDT A-VBELN A-ERDAT A-ERZET A-ERNAM A-VBTYP A-WAERK A-VKORG A-VTWEG A-SPART A-KVGR1 A-KV
		Betroffene Tabellen: (A)VBAP (B)FINS_TRR_ALLOC
4	- CHOOSE_LF	Felder für Ausgabeliste auswählen

Beachten Sie, dass nicht nur die Tabelle, sondern auch die **Join-Kriterien** ersetzt werden.

Für eine Erläuterung der Bedeutung von 1. Wahl und 2. Wahl siehe [F4-Hilfe für Joins mit Datenbanktabellen](#).

Wertehilfe für Ersatz-Views oder Ersatz-CDS-View-Entitäten

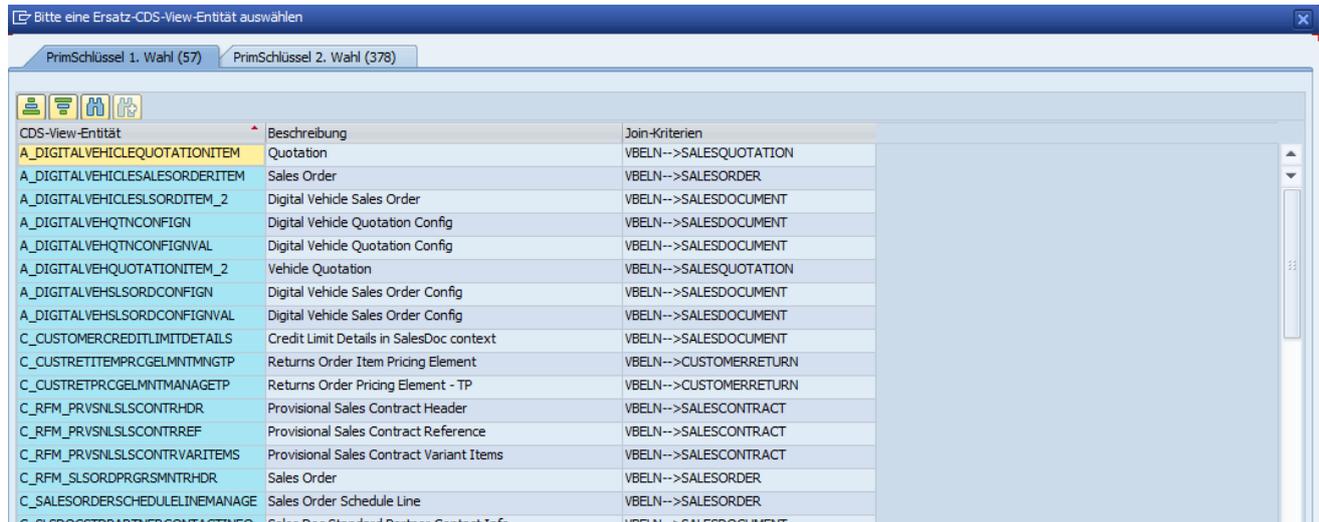
Falls wir die **zweite** Wertehilfe verwenden, erhalten wir folgende Liste mit Registerkarten:



View-Name	Typ	Datendefinition	Beschreibung	Join-Kriterien	Datenbank-Tabellen
/CPD/VFC_SALES	D		Sicht für Kundenauftrag	VBELN-->VBELN	VBAP
/DMBE/CSSUBLOC	D	/DMBE/C_SALES_LOCS_AND_SUBLOCS	Sales contract locations, sublocations	VBELN-->VBELN	OIJ_EL_CP_LOC, OIJ_EL_DOC_MOT, VBAP
/DMBE/ISALESLOC	D	/DMBE/I_SALES_CONTRACT_LOC	Locations ID of sales contracts	VBELN-->VBELN	OIJ_EL_CP_LOC, OIJ_EL_DOC_MOT, VBAP
/DMBE/ISDOCITEM	D	/DMBE/I_SALES_DOC_ITEM_DATA	Sales Document: Item Data	VBELN-->VBELN	VBAP
/SAPAPO/V_TDL_SL	D	/SAPAPO/V_TDL_SO_SCHED_LINE	SAPAPO TDL: SO Scheduling line	VBELN-->DOC_NO	VBEP
/SAPCEM/V_VBMAN	D		Manuell erfasste Verkaufsbelege	VBELN-->VBELN	J_3GVBAP, VBKD
AARNPRIOBSC	D	A_ARUNPRIORITY	Item	VBELN-->REQUIREMENTDOCUMENTNUMBER	DDDDLCHARTYPES, VBAP
ACMRITMPCRGELMNT	D	A_CREDITMEMOREQITEMPCRGELMNT	Item Pricing Element	VBELN-->CREDITMEMOREQUEST	DDDDLCHARTYPES, PRCD_ELEMENTS, T006,
ACMRPCRGELMNT	D	A_CREDITMEMOREQPCRGELMNT	Header Pricing Element	VBELN-->CREDITMEMOREQUEST	DDDDLCHARTYPES, PRCD_ELEMENTS, VBAK
ACRDTBLKDSLDOC	D	A_CREDITBLOCKEDSALESDOCUMENT	Credit Blocked Sales Document	VBELN-->SALESDOCUMENT	DDDDLCHARTYPES, VBAK
ACRITMPCRGELMNT	D	A_CUSTOMERRETURNITEMPCRGELMNT	Item Pricing Element	VBELN-->CUSTOMERRETURN	PRCD_ELEMENTS, VBAK
ACRPCRGELMNT	D	A_CUSTOMERRETURNPCRGELMNT	Header Pricing Element	VBELN-->CUSTOMERRETURN	PRCD_ELEMENTS, VBAK
ACRSCHDLN	D	A_CUSTOMERRETURNSCHEDULELINE	Schedule Line	VBELN-->CUSTOMERRETURN	VBAP, VBEP
ACLUSTRETSEKNO	D	A_CUSTOMERRETURNSERIALNUMBER	Serial Number	VBELN-->CUSTOMERRETURN	OBJK, SER02, VBAK
ADMRTMPCRGELMNT	D	A_DEBITMEMOREQITEMPCRGELMNT	Item Pricing Element	VBELN-->DEBITMEMOREQUEST	DDDDLCHARTYPES, PRCD_ELEMENTS, T006,
ADMRPCRGELMNT	D	A_DEBITMEMOREQPCRGELMNT	Header Pricing Element	VBELN-->DEBITMEMOREQUEST	DDDDLCHARTYPES, PRCD_ELEMENTS, VBAK

Weitere Informationen zu den verschiedenen Spalten der Vorschlagsliste finden Sie in der [F4-Hilfe für Joins mit Views](#).

Die **dritte** Wertehilfe liefert folgende Vorschläge:

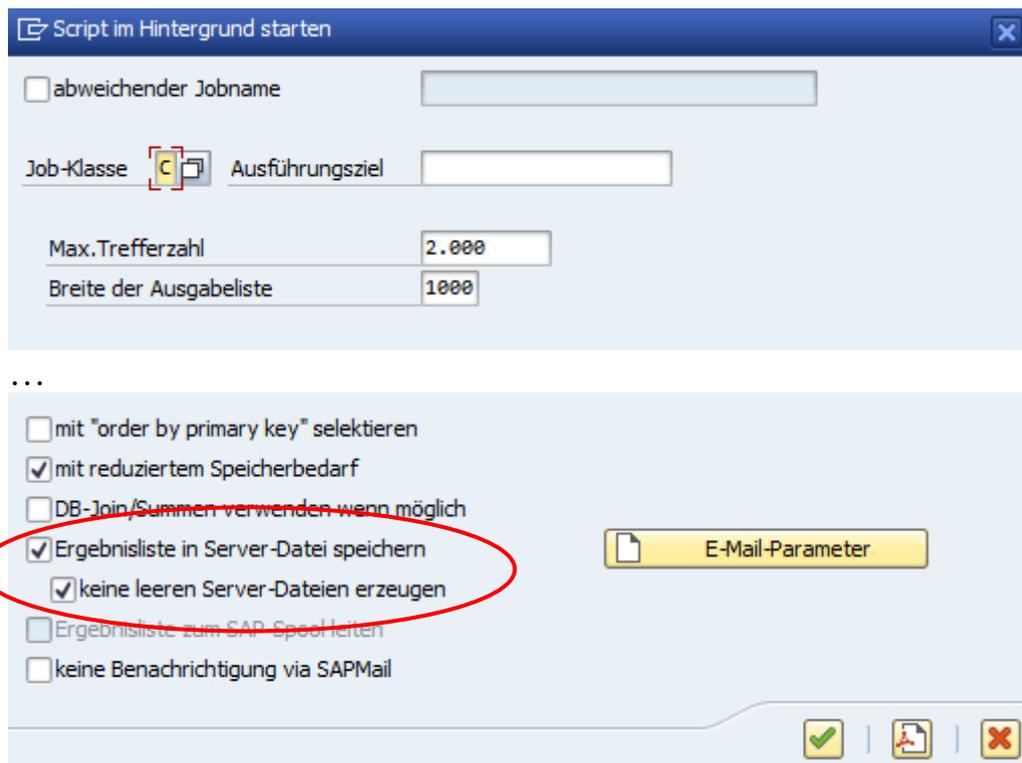


CDS-View-Entität	Beschreibung	Join-Kriterien
A_DIGITALVEHICLEQUOTATIONITEM	Quotation	VBELN-->SALESQUOTATION
A_DIGITALVEHICLESORDERITEM	Sales Order	VBELN-->SALESORDER
A_DIGITALVEHICLESORDITEM_2	Digital Vehicle Sales Order	VBELN-->SALESDOCUMENT
A_DIGITALVEHQTNCONFIG	Digital Vehicle Quotation Config	VBELN-->SALESDOCUMENT
A_DIGITALVEHQTNCONFIGVAL	Digital Vehicle Quotation Config	VBELN-->SALESDOCUMENT
A_DIGITALVEHQUTATIONITEM_2	Vehicle Quotation	VBELN-->SALESQUOTATION
A_DIGITALVEHLSORDCONFIG	Digital Vehicle Sales Order Config	VBELN-->SALESDOCUMENT
A_DIGITALVEHLSORDCONFIGVAL	Digital Vehicle Sales Order Config	VBELN-->SALESDOCUMENT
C_CUSTOMERCREDITLIMITDETAILS	Credit Limit Details in SalesDoc context	VBELN-->SALESDOCUMENT
C_CUSTRETITEMPCGELMNTMNGTP	Returns Order Item Pricing Element	VBELN-->CUSTOMERRETURN
C_CUSTRETPCRGELMNTMANAGETP	Returns Order Pricing Element - TP	VBELN-->CUSTOMERRETURN
C_RFM_PRIVSNLSLCONTRHDR	Provisional Sales Contract Header	VBELN-->SALESCONTRACT
C_RFM_PRIVSNLSLCONTRREF	Provisional Sales Contract Reference	VBELN-->SALESCONTRACT
C_RFM_PRIVSNLSLCONTRVARITEMS	Provisional Sales Contract Variant Items	VBELN-->SALESCONTRACT
C_RFM_SLSORDPRGRSMNTRHDR	Sales Order	VBELN-->SALESORDER
C_SALESORDERSCHEDULELINEMANAGE	Sales Order Schedule Line	VBELN-->SALESORDER
C_SLSORDPARTNERCONTACTINFO	Sales Doc Standard Partner Contact Info	VBELN-->SALESDOCUMENT

[Zum Anfang](#)

Hintergrund-Option “keine leeren Server-Dateien”

Wird ein Scripts im Hintergrund eingeplant, kann angegeben werden, dass das Ergebnis in eine Server-Datei geschrieben werden soll. Bis dato, wenn das Ergebnis des Scripts eine leere Liste war, erzeugte das Programm eine leere Server-Datei. Um diese Situation zu vermeiden, ist es möglich festzulegen, ab Version **3.6D** von SE16XXL, dass **keine leeren** Server-Dateien erstellt werden sollen. Das dazugehörige Dialogfenster wurde entsprechend erweitert:



Das Job-Log eines solchen Scripts mit leerem Ergebnis würde wie folgt aussehen:

Datum	Uhrzeit	Nachrichtentext
24.10.2023	16:18:05	Job wurde gestartet
24.10.2023	16:18:05	Step 001 gestartet (Programm /TFTO/TX_BATCH_SCRIPT_X, Variante &0000000005240, Benutzername TOPFLOW)
24.10.2023	16:18:05	Tabellenvariante 'NO_RESULTS' wird geladen
24.10.2023	16:18:05	Keine Sätze für die Selektion gefunden
24.10.2023	16:18:06	Ziel-Server-Dateipfad ist \usr\sap\ZE5\tmp
24.10.2023	16:18:06	Globales Script \$ALV_AFKO_AUFK_AFPO_AFVC wird ausgeführt
24.10.2023	16:18:06	Option "mit reduziertem Speicherbedarf" ist aktiv
24.10.2023	16:18:06	*****
24.10.2023	16:18:06	Die Option "keine leeren Server-Dateien erzeugen" ist aktiv.
24.10.2023	16:18:06	Das Ergebnis ist leer - keine Server-Datei erzeugt !
24.10.2023	16:18:06	*****
24.10.2023	16:18:06	Keine E-Mail-Empfänger definiert
24.10.2023	16:18:06	Job wurde beendet

[Zum Anfang](#)

Hintergrund-Übersicht – einige neue Funktionen

Zur Erinnerung: Die Hintergrund-Übersicht kann vom Einstiegsbild von SE16XXL über folgende Menüfunktion aufgerufen werden:

Springen → Übersicht der Hintergrund-Jobs

Wir werden die neuen Funktionen ausgehend von folgender Liste von periodischen Jobs besprechen:

8 SE16XXL Hintergrund-Jobs selektiert									
Job	Gepl.Datum	Gepl.Zeit	Status	Period.	Benutzer	Sc.Type	Script	Glob	
SCRIPT_\$AUFK_AFPO_UND_JEST	24.10.2023	18:00:00	Freigegeben	täglich	TOPFLOW	☰	\$AUFK_AFPO_UND_JEST	●	
	23.10.2023	18:00:00	Beendet	täglich	TOPFLOW	☰	\$AUFK_AFPO_UND_JEST	●	
	22.10.2023	18:00:00	Beendet	täglich	TOPFLOW	☰	\$AUFK_AFPO_UND_JEST	●	
SCRIPT_\$CDS_VIEW_ENT_BEISPIEL	29.10.2023	20:00:00	Freigegeben	Wöchent.	TOPFLOW	☰	\$CDS_VIEW_ENT_BEISPIEL	●	
	22.10.2023	20:00:00	Beendet	Wöchent.	TOPFLOW	☰	\$CDS_VIEW_ENT_BEISPIEL	●	
SCRIPT_\$MAT_BESTAND	24.10.2023	20:30:00	Freigegeben	täglich	TOPFLOW	☰	\$MAT_BESTAND	●	
	23.10.2023	20:30:00	Beendet	täglich	TOPFLOW	☰	\$MAT_BESTAND	●	
	22.10.2023	20:30:00	Beendet	täglich	TOPFLOW	☰	\$MAT_BESTAND	●	

Die Liste ist aufsteigend nach Jobname und absteigend nach Datum/Uhrzeit sortiert.

Anforderung eines periodischen Jobs ändern

Ab Version **3.6D** von SE16XXL ist es möglich, die zugrunde liegende Anforderung eines **periodischen Jobs** zu **ändern**, ohne den Job selbst zu ändern. Klarerweise ist dies nur für Jobs mit dem Status **“freigegeben”** oder **“geplant”** möglich.

Bevor wir diese Funktionalität anhand eines Beispiels veranschaulichen, wäre es vielleicht interessant, ein paar Worte auf die interne Implementierung dieser neuen Funktion zu verwenden.

Wenn ein Script periodisch im Hintergrund ausgeführt werden soll, wird eine entsprechende Hintergrund-Anforderung in die Datenbank geschrieben, die alle notwendigen Informationen über das auszuführende Script und die anderen Parameter wie Variante, Layout usw. beinhaltet. Dann wird ein periodischer Job angelegt und freigegeben. Dieser Job besteht aus einem einzigen Step mit einer Variante, die den Job mit der Hintergrund-Anforderung verknüpft. Jedes Mal, wenn der periodische Job ausgeführt wird, wird das Ergebnis entweder in die Datenbank, in den SAP-Spool oder in eine Server-Datei geschrieben. Die Hintergrund-Anforderung ist jedoch für alle diese Jobs gleich, für den periodisch geplanten und für all seine fertigen “Kinder”, die alle anhand ihrer Variante auf diese eine Anforderung zeigen.

Wenn die Hintergrund-Anforderung geändert wird, wird eigentlich eine Kopie davon erstellt, die die gewünschten Änderungen beinhaltet, und dann über eine neue Variante mit dem periodischen Job verknüpft. Nachdem es nicht möglich ist, die Variante eines Job-Steps zu ändern, wenn der geplante Startzeitpunkt in der Vergangenheit liegt, was sein kann, wenn sich der Job im Status “geplant” befindet, kann es notwendig werden, auch das geplante Datum/Uhrzeit zu ändern. Am Ende der Operation gibt es zwei Hintergrund-Anforderungen, die “alte”, die mit den bereits abgeschlossenen Jobs verknüpft ist, und die neue, die mit dem periodischen Job im Status “geplant” oder “freigegeben” verknüpft ist.

Nun können wir mit einem Beispiel fortfahren.

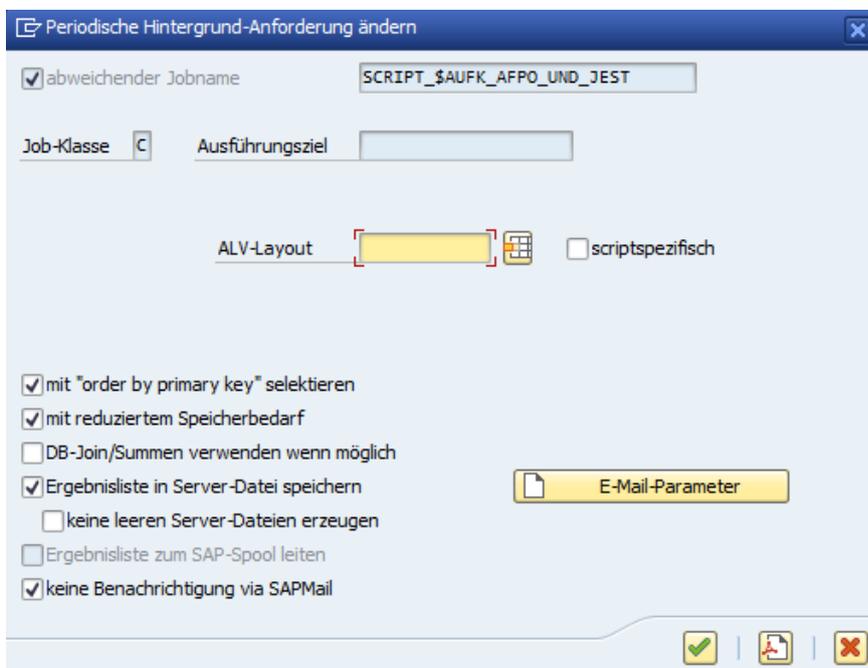
Wir setzen den Cursor auf den ersten periodischen Job mit dem Status “freigegeben”:

Job	Gepl. Datum	Gepl. Zeit	Status	Period.	Benutzer	Sc. Typ	Script
SCRIPT_\$AUFK_AFPO_UND_JEST	24.10.2023	18:00:00	Freigegeben	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST
	23.10.2023	18:00:00	Beendet	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST
	22.10.2023	18:00:00	Beendet	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST

Dann rufen wir folgende Menüfunktion auf:

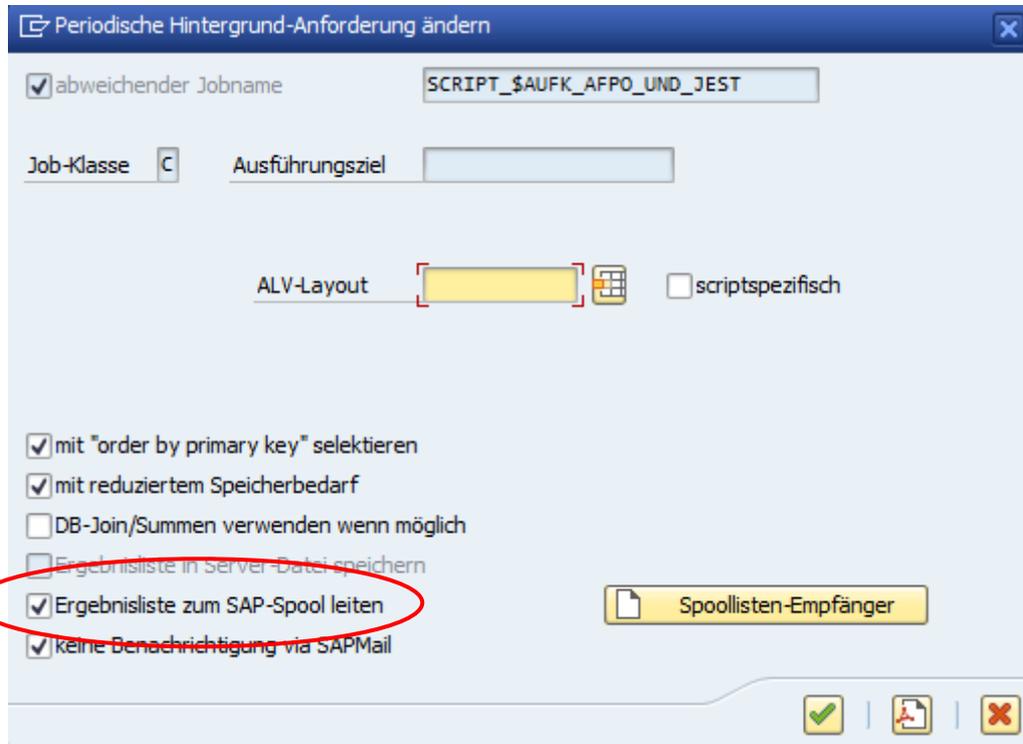
Job(s) → Anford. eines periodischen Jobs ändern

Das Programm reagiert mit der Anzeige des folgenden Dialogfensters:

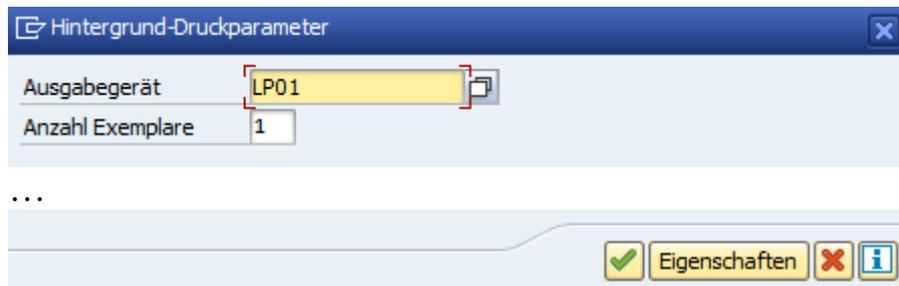


Dieses Dialogfenster ähnelt demjenigen, das bei der Einplanung des Hintergrund-Scripts verwendet wird, jedoch **ohne** zeitbezogene Parameter. Darüber hinaus sind der abweichende Jobname, die Job-Klasse usw. **geschützt**.

An dieser Stelle ändern wir die Parameter und leiten das Ergebnis des Scripts zum SAP-Spool:

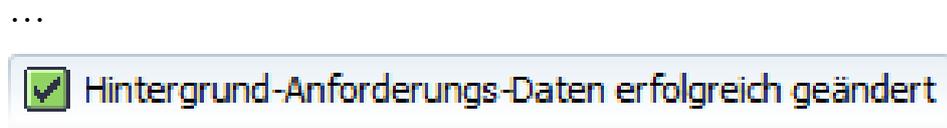


Nach Betätigung von müssen noch die Druckparameter angegeben werden:



Als Ergebnis ändert sich die Job-Übersicht entsprechend:

Job	Gepl. Datum	Gepl. Zeit	Status	Period.	eil	D.Par.	Sp.EMP	SDatei	K.I.
SCRIPT_\$AUFK_AFPO_UND_JEST	24.10.2023	18:00:00	Freigegeben	täglich					
	23.10.2023	18:00:00	Beendet	täglich					
	22.10.2023	18:00:00	Beendet	täglich					
SCRIPT_\$CDS_VIEW_ENT_BEISPIEL	29.10.2023	20:00:00	Freigegeben	Wöchent.					
	27.10.2023	20:00:00	Beendet	Wöchent.					



Wenn wir uns nun die entsprechenden Hintergrund-Anforderungen ansehen, sehen wir sowohl die **alte** als auch die **neue** Anforderung:

2 SE16XXL Hintergrund-Anforderungen selektiert

Benutzer	Datum	Uhrzeit	Nr.	Per.	Sc.Type	Script
TOPFLOW	24.10.2023	18:00:00	0000	●		\$AUFK_AFPO_UND_JEST
TOPFLOW	22.10.2023	18:00:00	0000	●		\$AUFK_AFPO_UND_JEST

...

Die alte Anforderung schreibt in eine Server-Datei, die neue in den SAP-Spool. Die alte Anforderung ist mit den bereits abgeschlossenen Jobs verknüpft, die neue ist mit dem periodischen Job mit Status “freigegeben” verknüpft. Dies kann leicht überprüft werden, indem man auf die Ikone ganz rechts klickt:

D.Par.	Sp.Emp	SDatei	K.leer.	E-Mail	Job(s)

Für die alte Anforderung erhalten wir:

Jobübersicht

Jobname	Spool	Job Dok	Job-Erstellung	Status	Startdatum	Startzeit	Dauer(sec.)	Verzög.	Man	Ver
SCRIPT_\$AUFK_AFPO_UND_JEST			TOPFLOW	fertig	22.10.2023	18:00:48	1	48	800	
SCRIPT_\$AUFK_AFPO_UND_JEST			TOPFLOW	fertig	23.10.2023	18:00:57	1	57	800	
*Zusammenfassung							2	105		

Für die neue Anforderung hingegen ist der zugehörige Job wie folgt:

Jobübersicht

Jobname	Spool	Job Dok	Job-Erstellung	Status	Startdatum	Startzeit	Dauer(sec.)	Verzög.	Man	Ver
SCRIPT_\$AUFK_AFPO_UND_JEST			TOPFLOW	freigegeben			0	0	800	
*Zusammenfassung							0	0		

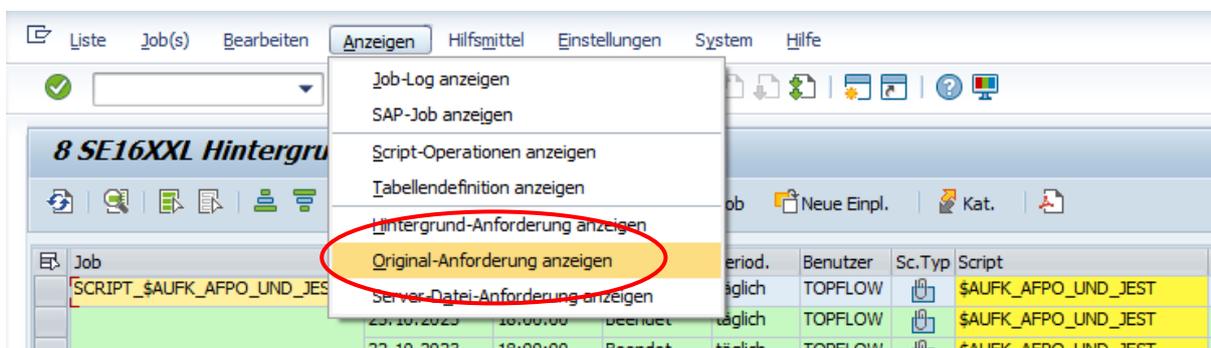
Original-Anforderung eines Hintergrund-Jobs anzeigen

Up to now only the “current” request associated with a job could be displayed in the Overview of Background Jobs. The related menu function was

Bis dato konnte in der Übersicht der Hintergrund-Jobs nur die “aktuelle” Anforderung eines Jobs angezeigt werden. Die zugehörige Menüfunktion ist

Anzeigen → Hintergrund-Anforderung anzeigen

Die notwendigen Informationen wurden der ursprünglichen Anforderung und der Job-Definition entnommen. Die Original-Anforderung konnte jedoch nicht angezeigt werden. Dieses Manko wurde nun beseitigt. Folgende Menüfunktion steht nun ebenfalls zur Verfügung:



Diese Funktion ist vor allem für periodische Jobs interessant, denn die Startbedingung dieser Jobs kann geändert werden, wobei nur der Job geändert wird, nicht aber die zugrunde liegende Hintergrund-Anforderung. Wenn z.B. die Periodizität eines Jobs von **einmal wöchentlich** auf **einmal täglich** geändert wird, zeigt die Original-Anforderung weiterhin “wöchentlich” an, während die aktuelle Anforderung “täglich” als Periodizität anzeigt.

Wir werden dieses Verhalten anhand eines Beispiels verdeutlichen.

Wir markieren den folgenden wöchentlichen Job und betätigen  auf der Anwendungsleiste:

8 SE16XXL Hintergrund-Jobs selektiert

Job	Gepl. Datum	Gepl. Zeit	Status	Period.	Benutzer	Sc. Typ	Script	Glob
SCRIPT_\$AUFK_AFPO_UND_JEST	24. 10. 2023	18:00:00	Freigegeben	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST	●
	23. 10. 2023	18:00:00	Beendet	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST	●
	22. 10. 2023	18:00:00	Beendet	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST	●
SCRIPT_\$CDS_VIEW_ENT_BEISPIEL	29. 10. 2023	20:00:00	Freigegeben	Wöchent.	TOPFLOW		\$CDS_VIEW_ENT_BEISPIEL	●
	22. 10. 2023	20:00:00	Beendet	Wöchent.	TOPFLOW		\$CDS_VIEW_ENT_BEISPIEL	●
SCRIPT_\$MAT_BESTAND	24. 10. 2023	20:30:00	Freigegeben	täglich	TOPFLOW		\$MAT_BESTAND	●
	23. 10. 2023	20:30:00	Beendet	täglich	TOPFLOW		\$MAT_BESTAND	●

Das Programm reagiert mit folgendem Dialogfenster:

Wir ändern das geplante Startdatum auf das heutige Datum:

Dann betätigen wir **Periodenwerte**, um die Periodizität auf "täglich" zu setzen:

Nach dem Sichern wird in der Jobübersicht die neue Periodizität angezeigt:

Job	Gepf.Datum	Gepf.Zeit	Status	Period.	Benutzer	Sc.Type	Script	Glob
SCRIPT_\$AUFK_AFPO_UND_JEST	24.10.2023	18:00:00	Freigegeben	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST	●
	23.10.2023	18:00:00	Beendet	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST	●
	22.10.2023	18:00:00	Beendet	täglich	TOPFLOW		\$AUFK_AFPO_UND_JEST	●
SCRIPT_\$CDS_VIEW_ENT_BEISPIEL	24.10.2023	20:00:00	Freigegeben	täglich	TOPFLOW		\$CDS_VIEW_ENT_BEISPIEL	●
	22.10.2023	20:00:00	Beendet	Wöchent.	TOPFLOW		\$CDS_VIEW_ENT_BEISPIEL	●

Über die Menüfunktion *Anzeigen* → *Hintergrund-Anforderung anzeigen* erhalten wir nun folgendes Bild:

Script im Hintergrund

abweichender Jobname

Job-Klasse Ausführungsziel

...

Start sofort Start verzögert Start an einem Werktag

Datum Uhrzeit

Jobausführung kalenderbezogen

Job immer ausführen

periodisch alle

Monat(e)

Woche(n)

Tag(e)

Stunde(n)

Minute(n)

Die Menüfunktion *Anzeige* → *Original-Anforderung anzeigen* liefert jedoch folgende Informationen:

Original-Hintergrund-Anforderung

abweichender Jobname

Job-Klasse Ausführungsziel

...

Start sofort Start verzögert Start an einem Werktag

Datum Uhrzeit

Jobausführung kalenderbezogen

Job immer ausführen

periodisch alle

Monat(e)

Woche(n)

Tag(e)

Stunde(n)

Minute(n)

[Zum Anfang](#)

Selektionswerte für die SFMI-Schnittstelle

Anhand der **SFMI**-Schnittstelle (Script Function Module Interface) ist es möglich, ein Script mithilfe eines ABAP-Programms auszuführen. Bis dato gab es zwei Möglichkeiten, das Ergebnis des Scripts zu beeinflussen: entweder durch die Ausgabe der Selektionsmaske (Parameter **I_WITH_SELSCREEN = 'X'**), wobei die Selektionswerte nach Belieben festgelegt werden konnten, oder durch die Angabe des Namens einer zu verwendenden Variante (Parameter **I_VARIANT**). In anderen Worten war es nicht möglich, ohne Selektionsmaske Selektionswerte zu übergeben, die nicht als Variante verfügbar waren. Es kann jedoch Situationen geben, in denen es nicht gewünscht oder nicht möglich ist, die Selektionsmaske eines Scripts auszugeben, und es besteht dennoch die Notwendigkeit, Ad-hoc-Selektionswerte anzugeben.

Um dieses Manko zu beseitigen, wurde die Schnittstelle des Funktionsbausteins **/TFTO/TX_SFMI_CALL_SCRIPT** um **zwei neue Parameter** erweitert:

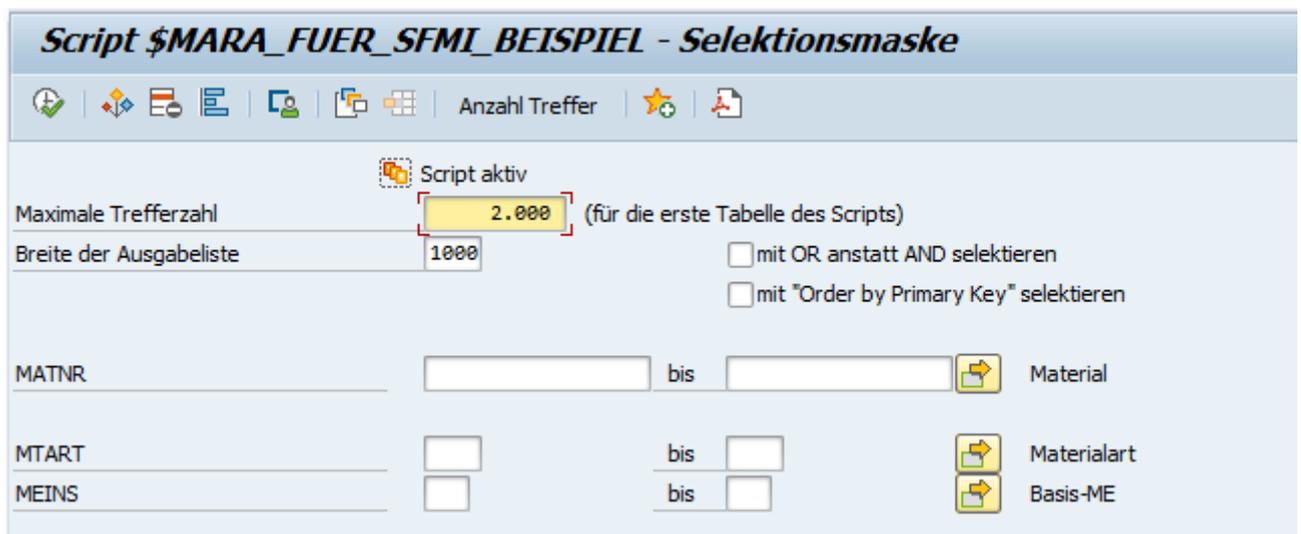
- **IT_SELVALS** (Selektionswerte)
- **IT_PARVALS** (Parameterwerte)

Sie wirken nur, wenn der Parameter **I_WITH_SELSCREEN** auf **SPACE** gesetzt ist.

Wir werden nun diese neue Funktion anhand einiger Beispiele veranschaulichen.

Beispiel 1 – ein Script mit normaler Selektionsmaske

Unser Beispielskript **\$MARA_FUER_SFMI_BEISPIEL** hat folgende Standard-Selektionsmaske:



Script \$MARA_FUER_SFMI_BEISPIEL - Selektionsmaske

Script aktiv

Maximale Trefferzahl: **2.000** (für die erste Tabelle des Scripts)

Breite der Ausgabeliste: 1000

mit OR anstatt AND selektieren

mit "Order by Primary Key" selektieren

MATNR: [] bis []  Material

MTART: [] bis []  Materialart

MEINS: [] bis []  Basis-ME

In unserem Beispielprogramm Z_SFMI_CALL_SCRIPT_BEISPIEL richten wir die interne Tabelle GT_SELVALS wie in der folgenden FORM-Routine ein:

```

*&-----*
*& Form U_GT_SELVALS_BEFUELLEN
*&-----*
FORM U_GT_SELVALS_BEFUELLEN
      CHANGING PT_SELVALS  TYPE /TFTO/TX_IT_SEL_VALS.

DATA: LS_SELVAL          TYPE /TFTO/TX_ST_SEL_VAL.

CLEAR PT_SELVALS[].

CLEAR LS_SELVAL.
LS_SELVAL-FIELDNAME = 'MATNR'.
LS_SELVAL-SIGN      = 'I'.
LS_SELVAL-OPTION    = 'CP'.
LS_SELVAL-LOW       = 'M*'.
APPEND LS_SELVAL TO PT_SELVALS.

CLEAR LS_SELVAL.
LS_SELVAL-FIELDNAME = 'MTART'.
LS_SELVAL-SIGN      = 'I'.
LS_SELVAL-OPTION    = 'EQ'.
LS_SELVAL-LOW       = 'FERT'.
APPEND LS_SELVAL TO PT_SELVALS.

ENDFORM.

```

Dann rufen wir den SFMI-Funktionsbaustein wie folgt auf:

```

CALL FUNCTION '/TFTO/TX_SFMI_CALL_SCRIPT'
  EXPORTING
    I_SCRIPT          = C_SCRIPT
    I_FOR_GRID        = 'X'
    I_WITH_SELSCREEN  = SPACE
    IT_SELVALS        = GT_SELVALS[]
  IMPORTING
    E_EXCEPTION       = G_EXCEPTION
    ET_MSG            = GT_MSG
    ED_SFMI_REF       = GD_SFMI_REF
    ED_T_SFMI_REF     = GD_T_SFMI_REF
    ET_SFMICOL        = GT_SFMICOL
    ET_DFIES          = GT_DFIES
    ET_ALV_FCAT       = GT_ALV_FCAT
    ET_ALV_GROUPS     = GT_ALV_GROUPS
    ET_ALV_SORT       = GT_ALV_SORT
    E_SOME_SKIPPED    = G_SOME_SKIPPED
  EXCEPTIONS
    SOMETHING_WRONG   = 1
    CANCELED          = 2
    NO_RESULTS        = 3
    OTHERS            = 99.

```

Wenn das Programm ausgeführt wird, erzeugt es sofort folgende Ergebnisliste:

Ergebnis von Script \$MARA_FUER_SFMI_BEISPIEL - 26 Zeilen



MATNR	ERSDA	LAEDA	VPSTA	MEINS	MTART	MATKL	BRGEW	GEWEI	DISST
M329-A	09.06.2000	29.01.2002	KVDBEZ	ST	FERT	00213	200	G	
MAT VTA 100	07.01.2002	22.01.2003	KVBZX	KG	FERT		0	KG	
MAT2401	21.02.2007	19.04.2007	KCV	ST	FERT	001	1	KG	
MAT2402	21.02.2007		K	ST	FERT	001	1	KG	
MAT2408	21.02.2007		K	ST	FERT	001	1	KG	
MD-FG1	27.02.2006	23.05.2006	KDBGXLEQ	EA	FERT	00107			000
MECHANICAL	16.05.2008		KV	EA	FERT		0	KG	
MED-1200	10.06.2004		KVDBG	EA	FERT	00204	1	KG	
MED-2000	11.06.2004		KVDBG	EA	FERT	00204	1	KG	

ANMERKUNG: Damit dieser Mechanismus funktioniert, ist es notwendig, dass sich die angegebenen Selektionswerte auf Felder beziehen, die auf der Selektionsmaske **vorhanden** sind. Wenn ein bestimmtes Feld nicht auf der Selektionsmaske vorhanden ist, werden die entsprechenden Werte **ignoriert**. Aus diesem Grund ist es ratsam, nur Scripts zu verwenden, die mit einer **speziellen Selektionsmaske** ausgestattet sind.

Beispiel 2 – ein Script mit einer speziellen Selektionsmaske

Diesmal basiert unser Beispiel auf einem Script mit spezieller Selektionsmaske:

Selektionsmaske von Script \$BEISPIEL_MIT_SELMASKE

Anzahl Treffer: (für die erste Tabelle des Scripts)

Breite der Ausgabeliste:

Script aktiv

Maximale Trefferzahl: (für die erste Tabelle des Scripts)

Breite der Ausgabeliste:

Selektion von Tabelle AUFK

mit "Order by Primary Key" selektieren

AUFNR bis Auftrag

AUART bis Auftragsart

Selektion von Tabelle \$JEST

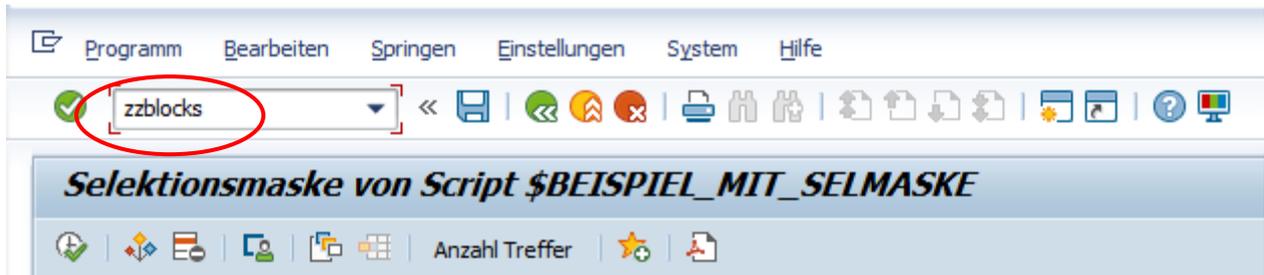
P_LANGUAGE P_Sprache

STAT bis Status

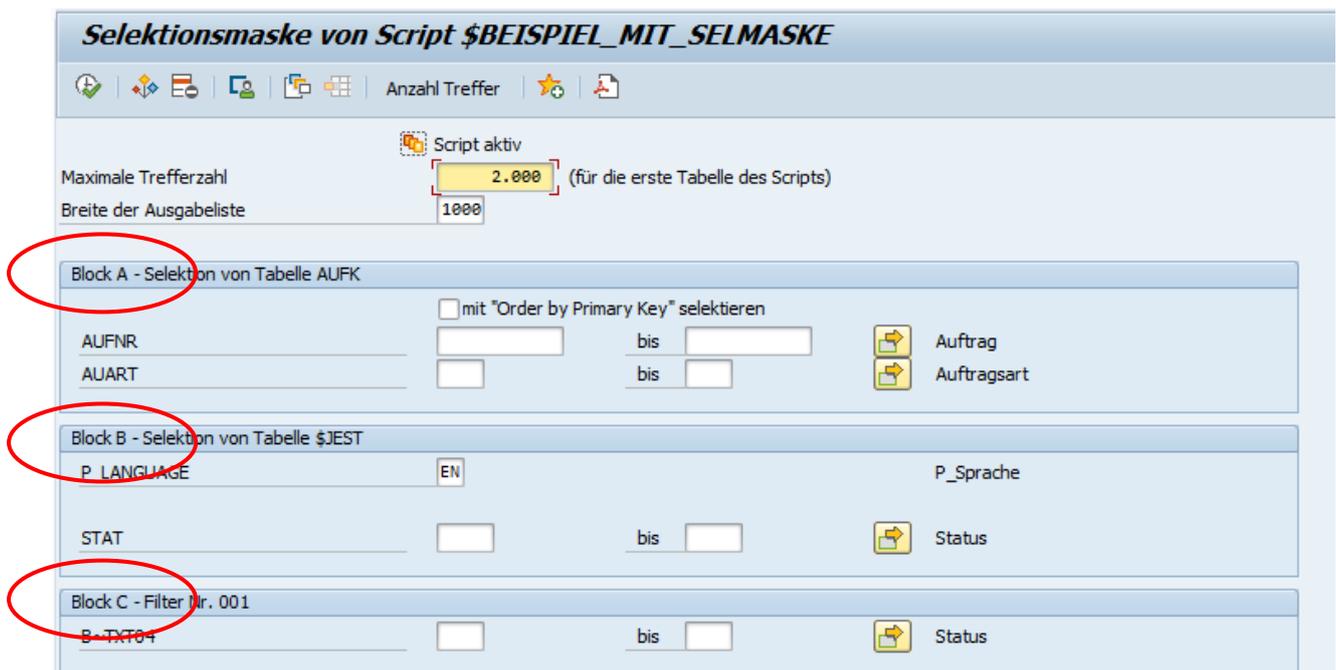
Filter Nr. 001

B~TXT04 bis Status

Wie im vorherigen Beispiel möchten wir dieses Script aufrufen, indem wir intern einige Selektionswerte übergeben, ohne die Selektionsmaske auszugeben. Die spezielle Selektionsmaske besteht aus drei Blöcken. Um die Selektionswerte korrekt anzugeben, müssen wir herausfinden, wie die einzelnen Blöcke angesprochen werden sollen. Dies geschieht durch Eingabe des OK-Codes “ZZBLOCKS”, wie in folgender Abbildung gezeigt:



Das Programm reagiert, indem es erneut die Selektionsmaske anzeigt, diesmal mit den Blocküberschriften versehen mit ihren **Blocknamen**:



Selektionsmaske von Script \$BEISPIEL_MIT_SELMASKE

Script aktiv

Maximale Trefferzahl: 2.000 (für die erste Tabelle des Scripts)

Breite der Ausgabeliste: 1000

Block A - Selektion von Tabelle AUFK

mit "Order by Primary Key" selektieren

AUFNR: bis Auftrag

AUART: bis Auftragsart

Block B - Selektion von Tabelle \$JEST

P_LANGUAGE: P_Sprache

STAT: bis Status

Block C - Filter Nr. 001

B-TXT01: bis Status

Wir sind nun in der Lage, die Selektionswerte (und Parameter) für den Aufruf unseres Scripts aufzubauen, ohne die Selektionsmaske auszugeben. Auf der nächsten Seite ist das Beispielcoding zu sehen. Beachten Sie, dass für den zweiten Block (B) nur der Parameter P_LANGUAGE angegeben wurde.

```

*&-----*
*&      Form   U_WERTE_VERSORGEN
*&-----*
FORM U_WERTE_VERSORGEN  CHANGING PT_SELVALS  TYPE /TFTO/TX_IT_SEL_VALS
                        PT_PARVALS  TYPE /TFTO/TX_IT_PAR_VALS.

DATA: LS_SELVAL          TYPE /TFTO/TX_ST_SEL_VAL.
DATA: LS_PARVAL          TYPE /TFTO/TX_ST_PAR_VAL.

CLEAR PT_SELVALS[].
CLEAR PT_PARVALS[].

CLEAR LS_SELVAL.
LS_SELVAL-BLOCK          = 'A'.
LS_SELVAL-FIELDNAME     = 'AUART'.
LS_SELVAL-SIGN           = 'I'.
LS_SELVAL-OPTION        = 'CP'.
LS_SELVAL-LOW            = 'PM*'.
APPEND LS_SELVAL TO PT_SELVALS.

CLEAR LS_PARVAL.
LS_PARVAL-BLOCK          = 'B'.
LS_PARVAL-PARNAME       = 'P_LANGUAGE'.
LS_PARVAL-VALUE         = 'E'.
APPEND LS_PARVAL TO PT_PARVALS.

CLEAR LS_SELVAL.
LS_SELVAL-BLOCK          = 'C'.
LS_SELVAL-ALIAS          = 'B'.
LS_SELVAL-FIELDNAME     = 'TXT04'.
LS_SELVAL-SIGN           = 'I'.
LS_SELVAL-OPTION        = 'EQ'.
LS_SELVAL-LOW            = 'CRTD'.
APPEND LS_SELVAL TO PT_SELVALS.

ENDFORM.                    " U_WERTE_VERSORGEN

```

Beachten Sie, dass für den Wert von **B~TXT04** im dritten Block auch der **Alias** angegeben wurde.

Der Funktionsaufruf lautet dann wie folgt:

```

CALL FUNCTION '/TFTO/TX_SFMI_CALL_SCRIPT'
  EXPORTING
    I_SCRIPT              = C_SCRIPT
    I_FOR_GRID            = 'X'
    I_WITH_SELSCREEN      = SPACE
    IT_SELVALS            = GT_SELVALS[]
    IT_PARVALS            = GT_PARVALS[]
  IMPORTING
    E_EXCEPTION           = G_EXCEPTION
    ET_MSG                = GT_MSG
    ED_SFMI_REF           = GD_SFMI_REF
  ...

```

Wenn das Programm ausgeführt wird, wird folgende Liste erstellt:

Ergebnis von Script \$BEISPIEL_MIT_SELMASKE - 2000 Zeilen

A~AUFNR	A~AUART	A~AUTYP	A~BUKRS	A~WERKS	A~GSBER	A~KOKRS	A~OBJNR	B~OBJNR	B~STAT	B~INACT	B~CHGNR	B~OBJTYP	B~TXT04	B~TXT30
00000800164	PM03	30	1000	1000	0001	1000	OR00000800164	OR00000800164	I0001	X	002	ORI	CRTD	Created
00000800165	PM03	30	1000	1000	0001	1000	OR00000800165	OR00000800165	I0001	X	001	ORI	CRTD	Created
00000800166	PM03	30	1000	1000	7000	1000	OR00000800166	OR00000800166	I0001	X	001	ORI	CRTD	Created
00000800167	PM03	30	1000	1000	0001	1000	OR00000800167	OR00000800167	I0001	X	001	ORI	CRTD	Created
00000800168	PM03	30	1000	1000	0001	1000	OR00000800168	OR00000800168	I0001	X	001	ORI	CRTD	Created
00000800000	PM06	30	1000	1000	1000	1000	OR00000800000	OR00000800000	I0001	X	000	ORI	CRTD	Created
00000800020	PM06	30	1000	1000	1000	1000	OR00000800020	OR00000800020	I0001	X	000	ORI	CRTD	Created
00000800040	PM06	30	1000	1000	7000	1000	OR00000800040	OR00000800040	I0001	X	000	ORI	CRTD	Created
00000800041	PM06	30	1000	1000	9900	1000	OR00000800041	OR00000800041	I0001	X	000	ORI	CRTD	Created
00000800042	PM06	30	1000	1000	7000	1000	OR00000800042	OR00000800042	I0001	X	000	ORI	CRTD	Created
00000800060	PM06	30	1000	1000	7000	1000	OR00000800060	OR00000800060	I0001	X	000	ORI	CRTD	Created
00000800080	PM06	30	1000	1000	7000	1000	OR00000800080	OR00000800080	I0001	X	001	ORI	CRTD	Created

ANMERKUNG: Falls die übergebenen Selektionswerte oder Parameter fehlerhaft sind, z. B. weil sie falsch geschrieben sind, oder weil der Blockname falsch ist oder die Select-Option oder der Parameter geschützt ist, gibt das Programm folgende Meldung aus:



WICHTIG: Die Blöcke auf einer Selektionsmaske mit Registerkarten heißen nicht **A, B, C**, sondern **001, 002, 003** usw., wie im folgenden Beispiel zu sehen ist:

Selektionsmaske von Script \$CLASSIF_PLUS_MARA_TBB

Anzahl Treffer

Script aktiv

Maximale Trefferzahl: 2.000 (für die erste Tabelle des Scripts)

Breite der Ausgabeliste: 1000

Block 001 - Selektion von Tabelle \$CLASSIF | Block 002 - Selektion von Tabelle MARA | Block 003 - Filter Nr. 001

P_CLASSYPE: 001 | P_Klassenart

P_OBJECTTABLE: MARA | P_ObjektTabelle

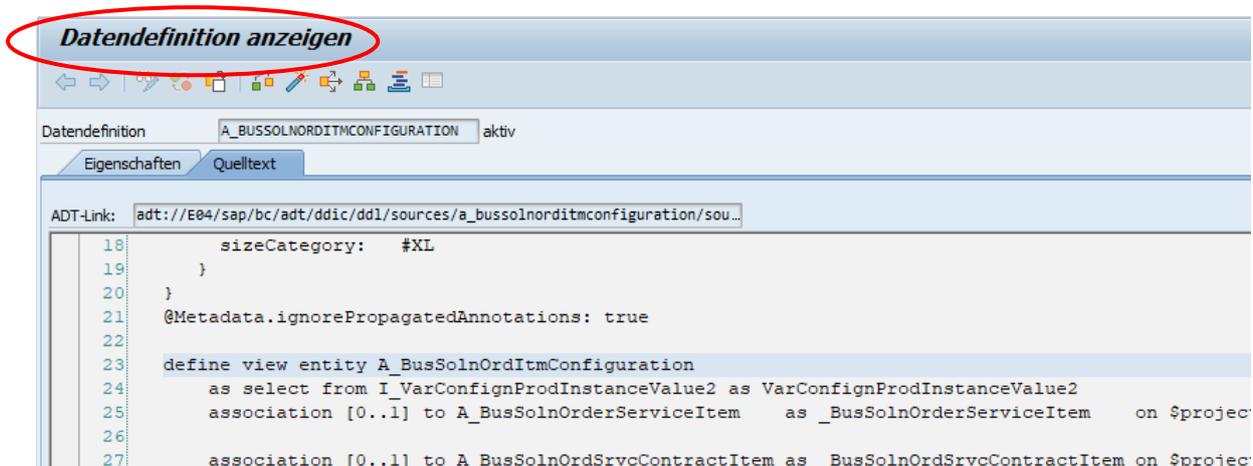
P_LANGUAGE: EN | P_Sprache

[Zum Anfang](#)

CDS-View-Entitäten unterstützt

Ab **Version 3.6C** unterstützt SE16XXL auch **CDS-View-Entitäten**. Es handelt sich dabei um eines der neuesten Features der Core Data Services von SAP. Sie zeichnen sich dadurch aus, dass nur eine Datendefinition, jedoch **keine Data Dictionary-Informationen** verfügbar sind. Dies ist der Hauptunterschied zwischen CDS-View-Entitäten und CDS-DDIC-basierten Views.

Beim Aufruf der Transaktion **SE11** (ABAP Dictionary) für eine CDS-View-Entität wird lediglich die **Datendefinition** angezeigt, wie im folgenden Beispiel:



In SE16XXL wurde die Einstiegsmaske wie folgt erweitert:



Das Feld für die Tabelle/View ist nun länger, um den bis zu 30 Zeichen langen Namen einer CDS-View-Entität zu akzeptieren. Darüber hinaus steht eine neue Schaltfläche für die F4-Hilfe für CDS-View-Entitäten zur Verfügung. Diese Schaltfläche ist nur sichtbar, wenn das aktuelle SAP-System CDS-View-Entitäten unterstützt.

Weitere Informationen finden Sie unter [CDS-View-Entitäten in SE16XXL](#).

[Zum Anfang](#)

Neue dynamische Datumsvariablen für Script-Varianten

Script-Varianten bieten unter anderem die Möglichkeit, bestimmten Select-Options eine dynamische Datumsvariable zuzuweisen. Zu der bereits langen Liste möglicher Datumsvariablen wurden **drei** neue Arten hinzugefügt:

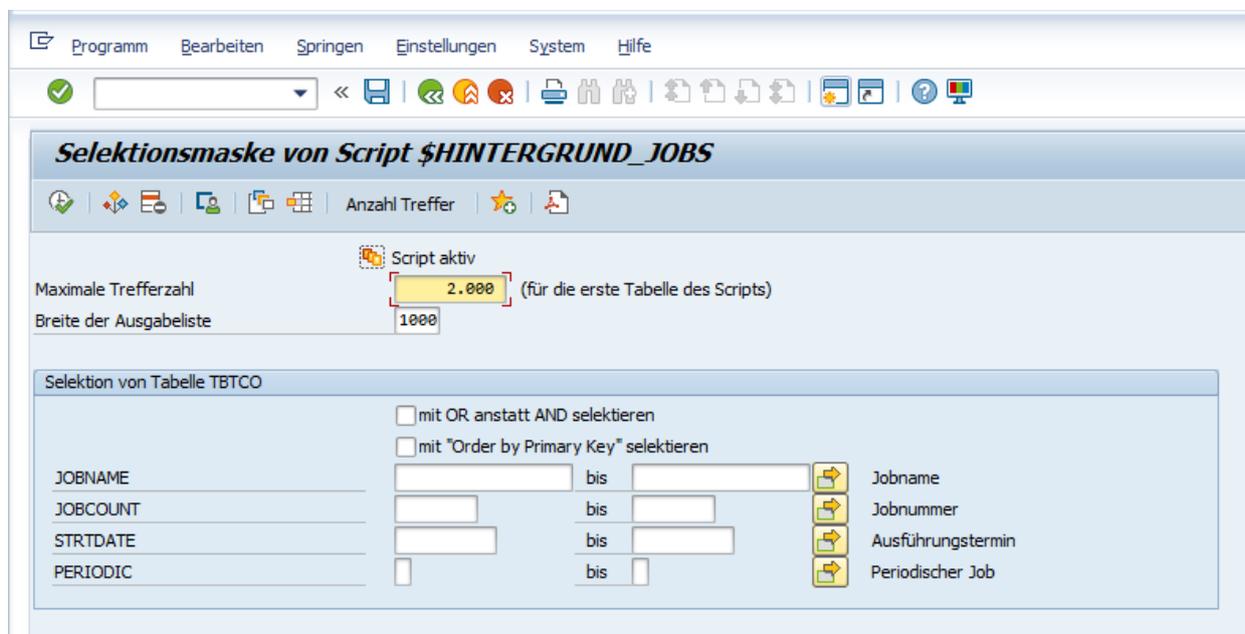
- **Akt. Jahr -/+ xx Jahre, akt. Jahr +/- yy Jahre**
- **Aktuelle Woche**
- **Akt. Woche -/+ xx Wochen, akt. Woche +/- yy Wochen**

Ein paar Beispiele werden diese neuen dynamischen Datumsvariablen illustrieren.

Beispiel 1 – Die Hintergrundjobs der letzten drei Wochen selektieren

Wir beginnen mit einem Script, das die Tabelle **TBTCO** (Job-Zustandsübersichtstabelle) selektiert. Es verfügt über eine spezielle Selektionsmaske, die die notwendige Voraussetzung für die Erstellung von Script-Varianten ist.

Das spezielle Selektionsmaske dieses Scripts sieht wie folgt aus:



Programm Bearbeiten Springen Einstellungen System Hilfe

Selektionsmaske von Script *\$HINTERGRUND_JOBS*

Script aktiv

Maximale Trefferzahl (für die erste Tabelle des Scripts)

Breite der Ausgabeliste

Selektion von Tabelle TBTCO

mit OR anstatt AND selektieren

mit "Order by Primary Key" selektieren

JOBNAME	<input type="text"/>	bis	<input type="text"/>		Jobname
JOBCOUNT	<input type="text"/>	bis	<input type="text"/>		Jobnummer
STRTDATE	<input type="text"/>	bis	<input type="text"/>		Ausführungstermin
PERIODIC	<input type="checkbox"/>	bis	<input type="checkbox"/>		Periodischer Job

Um eine Script-Variante zu erstellen, klicken wir auf die Schaltfläche  auf der Systemleiste. Es erscheint die entsprechende Definitionsmaske (wir haben bereits den Namen und die Beschreibung der Script-Variante eingetragen):

SE16XXL - Script-Variante definieren

Script: \$INTERGRUND_JOBS
 Variante: LETZTE_3_WOCHN geschützt
 Beschreibung: Die Jobs der letzten drei Wochen selektieren
 ALV-Layout: scriptspezifisch

Selektion von Tabelle TBTCO

	Werte	oblig.	gesch.	ausgb.
JOBNAME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
JOBCOUNT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
STRDATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
PERIODIC	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Unser Ziel ist es, der Select-Option **STRDATE** (Ausführungsdatum eines Jobs) eine dynamische Datumsvariable zuzuweisen. Dazu klicken wir auf das ► Symbol rechts daneben. Als Ergebnis wird eine Liste möglicher Variablen-Typen ausgegeben:

Selektion von Tabelle TBTCO

	Werte	oblig.	gesch.	ausgb.
JOBNAME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
JOBCOUNT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
STRDATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
PERIODIC	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bitte Variablen-Typ auswählen

- keine Variable
- dynamische Datumsvariable**
- TVARVC variable
- Set/Get-Parameter

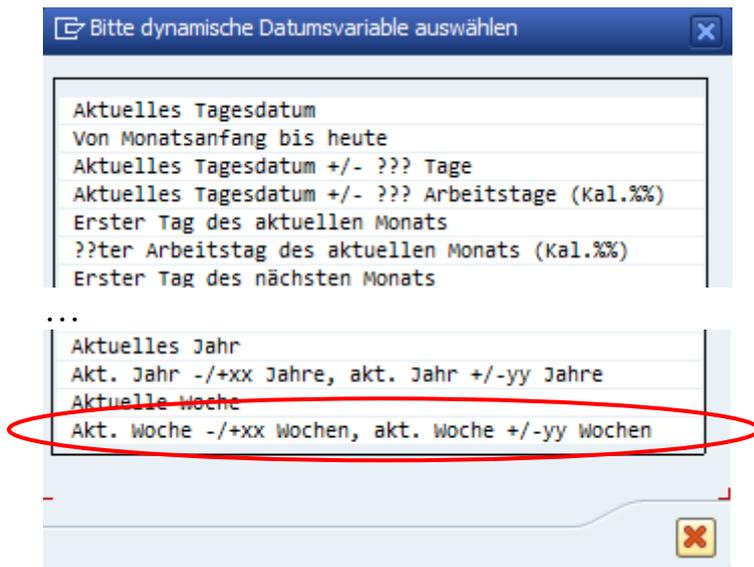
Wir wählen **dynamische Datumsvariable** aus. Die Anzeige ändert sich entsprechend:

Selektion von Tabelle TBTCO

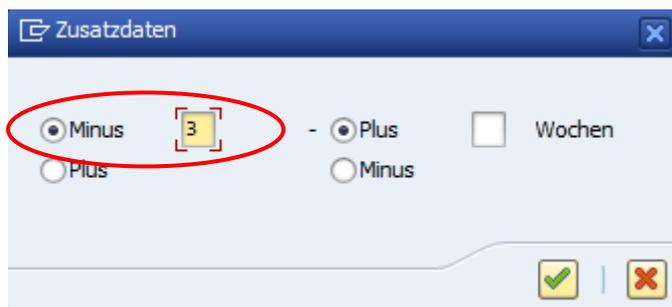
	Werte	oblig.	gesch.	ausgb.
JOBNAME	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
JOBCOUNT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
STRDATE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
PERIODIC	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

???

Nun klicken wir auf die drei Fragezeichen (???) , um eine Liste der verfügbaren Datumsvariablen zu erhalten:

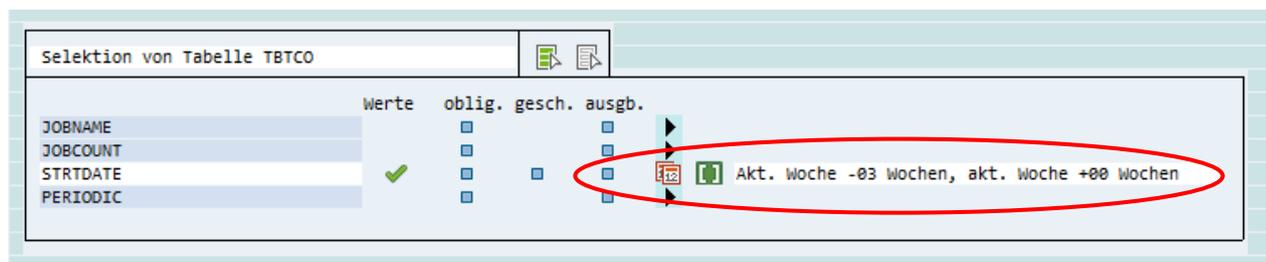


Der letzte Eintrag ist der, an dem wir interessiert sind, also klicken wir darauf. Daraufhin erscheint folgendes Dialogfenster zur Angabe der Zusatzdaten:



Wir haben “minus 3 Wochen – plus 0 Wochen” eingegeben. Das bedeutet drei Wochen zurück plus die aktuelle Woche. Je nachdem, wie “die letzten drei Wochen” interpretiert werden, wäre auch die Eingabe von “minus 2 Wochen – plus 0 Wochen” akzeptabel.

Die endgültige Anzeige sieht wie folgt aus:



Nach dem Sichern der Script-Variante durch Betätigen von  auf der Systemleiste erscheint erneut die spezielle Selektionsmaske mit dem definierten dynamischen Wert:

Selektion von Tabelle TBTCO

mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

JOBNAME bis  Jobname
 JOBCOUNT bis  Jobnummer
 STRTDATE bis  Ausführungstermin
 PERIODIC bis  Periodischer Job

Beispiel 2 – Die Hintergrundjobs von vor drei Wochen selektieren

Diesmal wollen wir nur die Jobs selektieren, die zur dritten Woche in der Vergangenheit gehören, d.h. die aktuelle Woche minus drei Wochen. Mit anderen Worten, nur eine Woche, jedoch in der Vergangenheit.

Die Prozedur ist ähnlich wie die vorherige, nur die Zusatzdaten sind anders:

Zusatzdaten

Minus - Plus Wochen
 Plus Minus

Die endgültige Anzeige ist:

Selektion von Tabelle TBTCO

	Werte	oblig.	gesch.	ausgb.	
JOBNAME		<input type="checkbox"/>		<input type="checkbox"/>	
JOBCOUNT		<input type="checkbox"/>		<input type="checkbox"/>	
STRTDATE	<input checked="" type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>	 Akt. Woche -03 Wochen, akt. Woche -03 Wochen
PERIODIC		<input type="checkbox"/>		<input type="checkbox"/>	

Und die spezielle Selektionsmaske:

Selektion von Tabelle TBTCO

mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

JOBNAME bis  Jobname
 JOBCOUNT bis  Jobnummer
 STRTDATE bis  Ausführungstermin
 PERIODIC bis  Periodischer Job

Die anderen dynamischen Datumsvariablen funktionieren ähnlich.

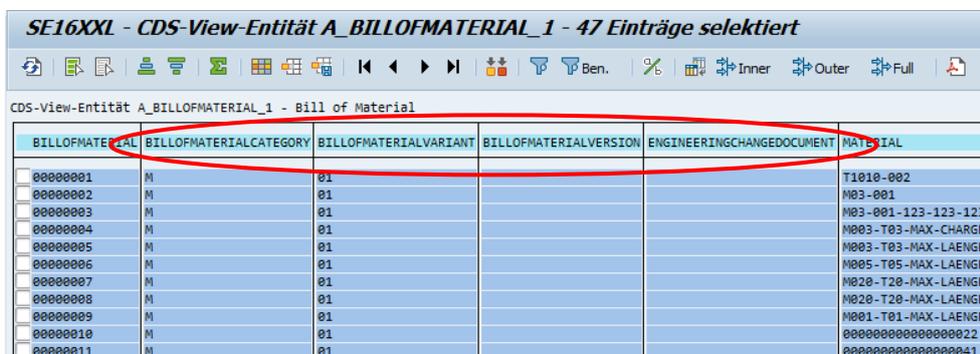
[Zum Anfang](#)

Längere Spaltennamen für Frontend-Dateien

Bisher durften die Namen der Spalten einer Frontend-Datei **16 Zeichen** nicht überschreiten. Da jedoch in den neuesten SAP-Versionen längere Feldnamen (bis zu 30 Zeichen) sogar für transparente Tabellen möglich geworden sind, wurde die oben genannte Einschränkung aufgehoben. Die Spaltennamen für Frontend-Dateien dürfen nun bis zu **30 Zeichen lang** sein.

Ein Beispiel wird diese Verbesserung verdeutlichen.

Angenommen, wir haben folgende Ergebnisliste mit sehr langen Feldnamen:

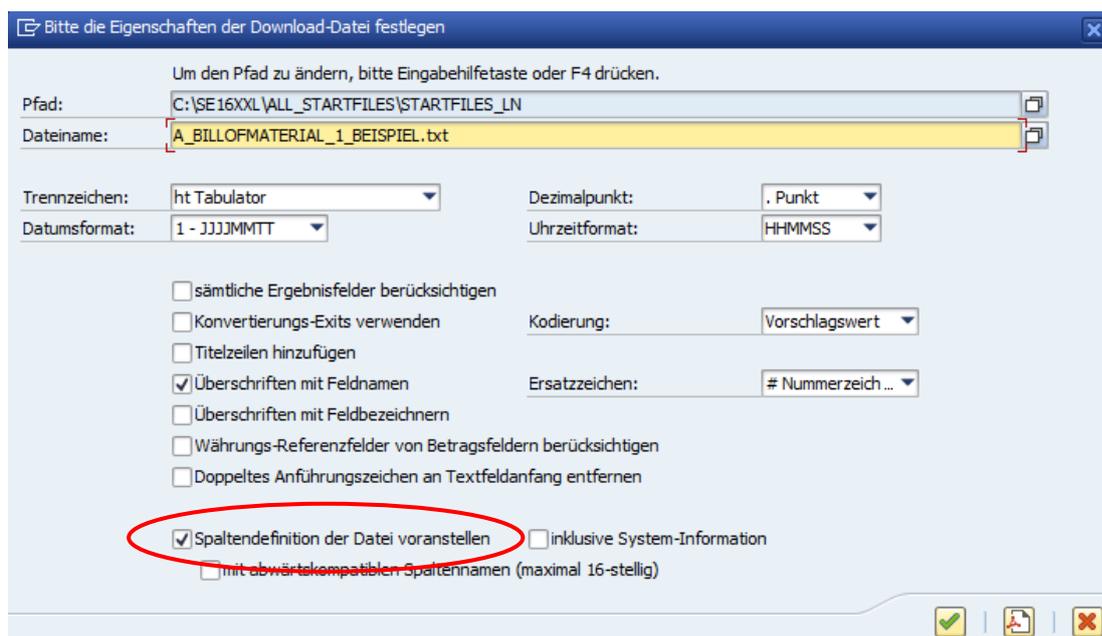


BILLOFMATERIAL	BILLOFMATERIALCATEGORY	BILLOFMATERIALVARIANT	BILLOFMATERIALVERSION	ENGINEERINGCHANGEDOCUMENT	MATERIAL
00000001	M	01			T1010-002
00000002	M	01			M03-001
00000003	M	01			M03-001-123-123-123-
00000004	M	01			M003-T03-MAX-CHARGE-
00000005	M	01			M003-T03-MAX-LAENGE-
00000006	M	01			M005-T05-MAX-LAENGE-
00000007	M	01			M020-T20-MAX-LAENGE-
00000008	M	01			M020-T20-MAX-LAENGE-
00000009	M	01			M001-T01-MAX-LAENGE-
00000010	M	01			000000000000000022
00000011	M	01			000000000000000041

Mithilfe folgender Menüfunktion

Liste → Download zum Frontend → als Text mit Trennzeichen

erstellen wir eine Frontend-Datei mit der Option **“Spaltendefinition der Datei voranstellen”**:



Bitte die Eigenschaften der Download-Datei festlegen

Um den Pfad zu ändern, bitte Eingabehilfetaste oder F4 drücken.

Pfad: C:\SE16XXL\ALL_STARTFILES\STARTFILES_LN

Dateiname: A_BILLOFMATERIAL_1_BEISPIEL.txt

Trennzeichen: ht Tabulator Dezimalpunkt: . Punkt

Datumsformat: 1 - JJJJMMTT Uhrzeitformat: HHMMSS

sämtliche Ergebnisfelder berücksichtigen

Konvertierungs-Exits verwenden Kodierung: Vorschlagswert

Titelzeilen hinzufügen

Überschriften mit Feldnamen Ersatzzeichen: # Nummerzeich...

Überschriften mit Feldbezeichnern

Währungs-Referenzfelder von Betragfeldern berücksichtigen

Doppeltes Anführungszeichen an Textfeldanfang entfernen

Spaltendefinition der Datei voranstellen inklusive System-Information

mit abwärtskompatiblen Spaltennamen (maximal 16-stellig)

Wenn wir die so erzeugte Frontend-Datei in SE16XXL hochladen, erscheint folgende Definitionsmaske:

C:|SE16XXL|...|STARTFILES_LN|A_BILLOFMATERIAL_1_BEISPIEL.txt

Akzeptieren
 Kopfzeile auswählen
 Definition importieren

Trennzeichen: ht Tabulator Erste relevante Zeile: 2
 Datumsformat: 1 - JJJJMMTT Dezimalpunkt: , Punkt
 Sprache: DE Deutsch

Sel.	Pos.	Name	DDIC	Dtel	Typ	KB	Länge	Dez.	Vorz.	F.N	Tabellenname	Feldname	Datenelement	Kurztext	Langtext	Wahr./Einh.
<input checked="" type="checkbox"/>	1	BILLOFMATERIAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	8	0	0			A_BILLOFMATERIA...	BILLOFMATERIAL	CS_STLNR	Stückliste	Stückliste	
<input checked="" type="checkbox"/>	2	BILLOFMATERIALCATEGORY	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	1	0	0			A_BILLOFMATERIA...	BILLOFMATERIALC...	STLTY	StlTyp	Stücklistentyp	
<input checked="" type="checkbox"/>	3	BILLOFMATERIALVARIANT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	2	0	0			A_BILLOFMATERIA...	BILLOFMATERIALV...	STALT	StlAlt	StücklAlternative	
<input checked="" type="checkbox"/>	4	BILLOFMATERIALVERSION	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	4	0	0			A_BILLOFMATERIA...	BILLOFMATERIALV...	CS_VERSN	Version	Stücklistenversion	
<input checked="" type="checkbox"/>	5	ENGINEERINGCHANGEDOCUMENT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	12	0	0			A_BILLOFMATERIA...	ENGINEERINGCHAN...	AENNR	ÄndNr	Änderungsnummer	
<input checked="" type="checkbox"/>	6	MATERIAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	40	0	0			A_BILLOFMATERIA...	MATERIAL	MATNR	Material	Material	
<input checked="" type="checkbox"/>	7	PLANT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	4	0	0			A_BILLOFMATERIA...	PLANT	WERKS_D	Werk	Werk	
<input checked="" type="checkbox"/>	8	BILLOFMATERIALVARIANTUSAGE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	1	0	0			A_BILLOFMATERIA...	BILLOFMATERIALV...	STLAN	StlVerw.	StücklVerwendung	
<input checked="" type="checkbox"/>	9	HEADervalIDITYSTARTDATE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DATS	8	0	0			A_BILLOFMATERIA...	HEADervalDITYS...	DATUV	Gültig	Gültig ab	
<input checked="" type="checkbox"/>	10	MATERIALTOLSIZEQUANTITY	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	QUAN	13	3	0			A_BILLOFMATERIA...	MATERIALTOLSI...	LOSBS	bis	Losgröße bis	BOMHEADERBASEUNIT
<input checked="" type="checkbox"/>	11	BOMHEADERBASEUNIT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	UNIT	3	0	0			A_BILLOFMATERIA...	BOMHEADERBASEUN...	BASME	Basis-ME	Basismengeneinheit	
<input checked="" type="checkbox"/>	12	LABORATORYDESIGNOFFICE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	3	0	0			A_BILLOFMATERIA...	LABORATORYORDES...	LABOR	Labor	Labor/Büro	
<input checked="" type="checkbox"/>	13	BILLOFMATERIALHDRDETAILTEXT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	255	0	0			A_BILLOFMATERIA...	BILLOFMATERIALH...		BILLOFMATERIALH...	BILLOFMATERIALHDRD...	
<input checked="" type="checkbox"/>	14	SELECTEDBILLOFMATERIALVERSION	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	4	0	0			A_BILLOFMATERIA...	SELECTEDBILLOFM...		SELECTEDBILLOFM...	SELECTEDBILLOFMATER...	

Wie man sehen kann, wurden die ursprünglichen Feldnamen nicht gekürzt.

Abwärtskompatible Spaltennamen

Eine Frontend-Datei mit Spaltennamen, die länger als 16 Zeichen sind, kann in einer älteren Version von SE16XXL nicht hochgeladen werden, da die vorangestellte Spaltendefinition aufgrund der längeren Namen ignoriert wird. Aus diesem Grund wurde eine **spezielle Option** eingeführt, die eine für ältere SE16XXL-Versionen geeignete Spaltendefinition erzeugt:

Spaltendefinition der Datei voranstellen
 inklusive System-Information
 mit abwärtskompatiblen Spaltennamen (maximal 16-stellig)

Die resultierende Frontend-Datei enthält wie in älteren Versionen gekürzte Namen:

C:|SE16XXL|...|STARTFILES_LN|A_BILLOFMATERIAL_1_ABW_KOMP.txt

Akzeptieren
 Kopfzeile auswählen
 Definition importieren

Trennzeichen: ht Tabulator Erste relevante Zeile: 2
 Datumsformat: 1 - JJJJMMTT Dezimalpunkt: , Punkt
 Sprache: DE Deutsch

Sel.	Pos.	Name	DDIC	Dtel	Typ	KB	Länge	Dez.	Vorz.	F.N	Tabellenname	Feldname	Datenelement	Kurztext	Langtext	Wahr./Einh.
<input checked="" type="checkbox"/>	1	BILLOFMATERIAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	8	0	0			A_BILLOFMATERIA...	BILLOFMATERIAL	CS_STLNR	Stückliste	Stückliste	
<input checked="" type="checkbox"/>	2	BILLOFMATERIALCATEGORY	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	1	0	0			A_BILLOFMATERIA...	BILLOFMATERIALC...	STLTY	StlTyp	Stücklistentyp	
<input checked="" type="checkbox"/>	3	BILLOFMATERIALVARIANT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	2	0	0			A_BILLOFMATERIA...	BILLOFMATERIALV...	STALT	StlAlt	StücklAlternative	
<input checked="" type="checkbox"/>	4	BILLOFMATERIALVERSION	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	4	0	0			A_BILLOFMATERIA...	BILLOFMATERIALV...	CS_VERSN	Version	Stücklistenversion	
<input checked="" type="checkbox"/>	5	ENGINEERINGCHANGEDOCUMENT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	12	0	0			A_BILLOFMATERIA...	ENGINEERINGCHAN...	AENNR	ÄndNr	Änderungsnummer	
<input checked="" type="checkbox"/>	6	MATERIAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	40	0	0			A_BILLOFMATERIA...	MATERIAL	MATNR	Material	Material	
<input checked="" type="checkbox"/>	7	PLANT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	4	0	0			A_BILLOFMATERIA...	PLANT	WERKS_D	Werk	Werk	
<input checked="" type="checkbox"/>	8	BILLOFMATERIALVARIANTUSAGE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	1	0	0			A_BILLOFMATERIA...	BILLOFMATERIALV...	STLAN	StlVerw.	StücklVerwendung	

[Zum Anfang](#)

Priorität für E-Mails in Hintergrund-Scripts

Bisher hatten die versendeten E-Mails eines Hintergrund-Scripts automatisch eine hohe Priorität. Dieses Verhalten verursachte einige Unannehmlichkeiten in Microsoft Outlook. Aus diesem Grund ist es nun möglich, die Priorität der zu versendenden E-Mails direkt festzulegen.

Um diese neue Funktion zu demonstrieren, planen wir ein Script im Hintergrund ein:

Script im Hintergrund starten

abweichender Jobname

Job-Klasse Ausführungsziel

Max.Trefferzahl

Breite der Ausgabeliste

Data-Aging-Einstellung Hot Cold Datum

...

mit "order by primary key" selektieren

mit reduziertem Speicherbedarf

Ergebnisliste in Server-Datei speichern

Ergebnisliste zum SAP-Spool leiten

keine Benachrichtigung via SAPMail

Durch Betätigen von erhalten wir:

E-Mail-Parameter

Empfänger

Verteilerliste Allgemeine Verteilerlis..

Liste von E-Mail-Adressen

...

Optionen

Leere Hintergrund-Ergebnisse ignorieren

Priorität

Wie zu sehen ist, stehen nun **drei Werte** für die **E-Mail-Priorität** zur Verfügung.

[Zum Anfang](#)

Zusätzliche Verbesserungen

Sprünge – Aufruf eines Scripts unter Angabe einer Script-Variante

Bisher war es möglich, einen Sprung zu Transaktion **/TFTO/SE16SCRIPT** zu definieren, die ein SE16XXL-Script aufruft. Es war jedoch **nicht möglich**, eine Script-Variante anzugeben, die für den Aufruf geladen werden soll. Dieses Manko wurde nun behoben. Ab Version 3.6C von SE16XXL **kann auch eine Script-Variante angegeben werden**. Diese Neuerung soll anhand eines Beispiels veranschaulicht werden.

Angenommen, wir haben eine Ergebnisliste mit Datensätzen der Tabelle **/TFTO/TXSV**, die die SE16XXL-Script-Varianten beinhaltet:

SE16XXL - Tabelle /TFTO/TXSV - 10 Einträge selektiert

Tabelle /TFTO/TXSV - SE16XXL - Script-Varianten

MANDT	RELID	SCRUSER	SCRIPT	VARIKEY	SRTF2	VARNAME	SHTEXT	XPRC
800	GV	GLOBAL	\$HINTERGRUND_JOBS	DREI_WOCHN_ZUR	0	DREI_WOCHN_ZUR	Die Jobs von drei Wochen zurück selektieren	
800	GV	GLOBAL	\$HINTERGRUND_JOBS	LETZTE_3_WOCHN	0	LETZTE_3_WOCHN	Die Jobs der letzten drei Wochen selektieren	
800	GV	GLOBAL	\$MARA_MTART	MEINS_M2	0	MEINS_M2	Materialien mit Basismengeneinheit M2	
800	GV	GLOBAL	\$MARA_MTART	MEINS_STUECK	0	MEINS_STUECK	Nur Basismengeneinheit Stück	
800	GV	GLOBAL	\$MARA_MTART	STANDARD	0	STANDARD	Standard Selektion	
800	GV	GLOBAL	\$MATERIALIEN	MTART FERT	0	MTART FERT	Fertigerzeugnisse	

Nun markieren wir die Spalte **“VARNAME”** und wählen die Menüfunktion

Extras → Feld-Sprünge definieren aus.

Im darauffolgenden Dialogfenster geben wir **“/TFTO/SE16SCRIPT”** als Ziel ein:

Definition von Sprüngen für Feld VARNAME

Feldname	Kriterium	Op.	Wert	Ziel	aktiv	Details
VARNAME				/TFTO/SE16SCRIPT	<input checked="" type="checkbox"/>	

An dieser Stelle drücken wir , um die Parameter des Sprungs angeben zu können:

Definition von Sprüngen für Feld VARNAME

Feldname	Kriterium	Op.	Wert	Ziel	aktiv	Details	Skri
VARNAME				/TFTO/SE16SCRIPT	<input checked="" type="checkbox"/>	Param.	

Ein Klick auf  öffnet folgendes Popup-Fenster, in dem die erforderlichen Parameter angegeben werden können (wir haben die entsprechenden Feldnamen bereits eingetragen):



Nachdem wir die Definition abgeschlossen haben, können wir in der Ergebnisliste überprüfen, ob der Sprung funktioniert. Wir klicken hierzu auf die Script-Variante in der zweiten Zeile:

SE16XXL - Tabelle /TFTO/TXSV - 10 Einträge selektiert

Tabelle /TFTO/TXSV - SE16XXL - Script-Varianten

MANDT	RELID	SCRUSER	SCRIPT	VARIKEY	SRTF2	VARNAME	SHTEXT
800	GV	GLOBAL	\$HINTERGRUND_JOBS	DREI_WOCHN_ZUR	0	DREI_WOCHN_ZUR	Die Jobs von drei Wochen zurück selektieren
800	GV	GLOBAL	\$HINTERGRUND_JOBS	LETZTE_3_WOCHN	0	LETZTE_3_WOCHN	Die Jobs der letzten drei Wochen selektieren
800	GV	GLOBAL	\$MARA_MTART	MEINS_M2	0	MEINS_M2	Materialien mit Basismengeneinheit M2
800	GV	GLOBAL	\$MARA_MTART	MEINS_STUECK	0	MEINS_STUECK	Nur Basismengeneinheit Stück
800	GV	GLOBAL	\$MARA_MTART	STANDARD	0	STANDARD	Standard Selektion

Die spezielle Selektionsmaske des entsprechenden Scripts erscheint:

Selektionsmaske von Script \$HINTERGRUND_JOBS

Anzahl Treffer: (für die erste Tabelle des Scripts)
 Breite der Ausgabeliste:

Selektion von Tabelle TBTCO

mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

JOBNAME bis Jobname
 JOBCOUNT bis Jobnummer
 STRTDAT bis Ausführungstermin
 PERIODIC bis Periodischer Job

Variante 'LETZTE_3_WOCHN' geladen

Wie zu sehen ist, wurde die gewünschte Script-Variante geladen.

Sprünge – /TFTO/SE11 für eine CDS-View-Entität aufrufen

Wie in [CDS-View-Entitäten in SE16XXL](#) beschrieben, ist eine spezielle Funktion in SE16XXL implementiert worden, um die **Feldstruktur** einer CDS-View-Entität so anzuzeigen, als ob sie im Data Dictionary verfügbar wäre. Diese Funktion kann auch bei der Definition eines Sprungs in einer Ergebnisliste genutzt werden. Wir werden diese Funktionalität anhand eines Beispiels demonstrieren.

Die in einem SAP-System verfügbaren CDS-View-Entitäten können ermittelt werden, indem die Einträge der Tabelle **DDLDEPENDENCY** mit **OBJECTTYPE** = “**STOB**” selektiert werden. Ein Outer-Join muss dann mit der gleichen Tabelle unter Verwendung von **DDLNAME** als Join-Kriterium durchgeführt werden, wobei diesmal die Einträge mit **OBJECTTYPE** = “**VIEW**” zu selektieren sind.

In der resultierenden Liste sind die STOB-Einträge mit einem VIEW-Pendant CDS-DDIC-basierte Views, die übrigen **ohne VIEW-Pendant sind CDS-View-Entitäten**.

Der Einfachheit halber beginnen wir mit einer Ergebnisliste, die nur CDS-View-Entitäten enthält. Hier ist es möglich, einen Sprung in die Transaktion “/TFTO/SE11” zu definieren, die für jede CDS-View-Entität die entsprechende Feldstruktur anzeigt, die in der Standardtransaktion SE11 (ABAP Dictionary) normalerweise nicht zur Verfügung steht.

SE16XXL - Outer-Join - 1782 Ergebnis-Sätze

Join von DDLDEPENDENCY(A) und DDLDEPENDENCY(B)

A~DDLNAME	A~OBJECTNAME	A~STATE	A~OBJECTTYPE	B~DDLNAME	B~OBJECTNAME	B~STATE	B~OBJECTTYPE
ABAP_DOCU_RELEASED_APIIS	ABAP_DOCU_RELEASED_APIIS	A	STOB				
ABAP_DOCU_TREE_SOURCE	ABAP_DOCU_TREE_SOURCE	A	STOB				
ABAP_LANGUAGE_VERSIONS	ABAP_LANGUAGE_VERSIONS	A	STOB				
ABAP_LANGUAGE_VERSIONS_SRC	ABAP_LANGUAGE_VERSIONS_SRC	A	STOB				
ABP_BEHV_DRAFT_USER_ACCESS	ABP_BEHV_DRAFT_USER_ACCESS	A	STOB				
ABP_BEHV_PAR_SHARE	ABP_BEHV_PAR_SHARE	A	STOB				
ACCTASGTBUSINESSTRANSACTION	ACCTASGTBUSINESSTRANSACTION	A	STOB				
ACMHEX_HIERS4DIM1_HIERARCHY	ACMHEX_HIERS4DIM1_HIERARCHY	A	STOB				
ACMRT_AUNIT_DDI_INHERIT	ACMRT_AUNIT_DDI_INHERIT	A	STOB				

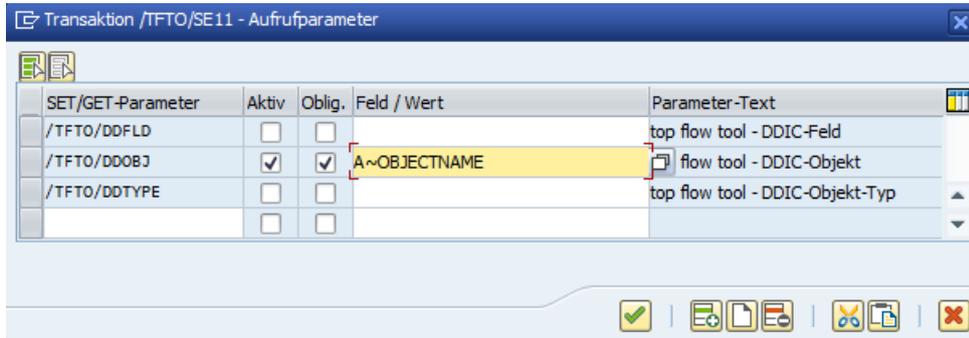
Auf dieser Ergebnisliste markieren wir die Spalte A~OBJECTNAME und wählen dann die Menüfunktion **Extras → Feld-Sprünge definieren** aus.

Im darauffolgenden Dialogfenster geben wir “/TFTO/SE11” in das Zielfeld ein:

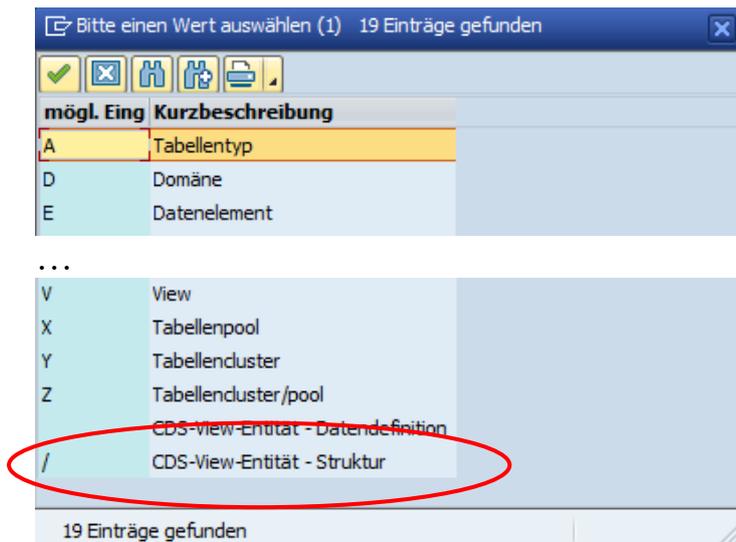
Definition von Sprüngen für Feld A~OBJECTNAME

Feldname	Kriterium	Op.	Wert	Ziel	aktiv	Details
A~OBJECTNAME				/TFTO/SE11	<input type="checkbox"/>	<input type="checkbox"/>

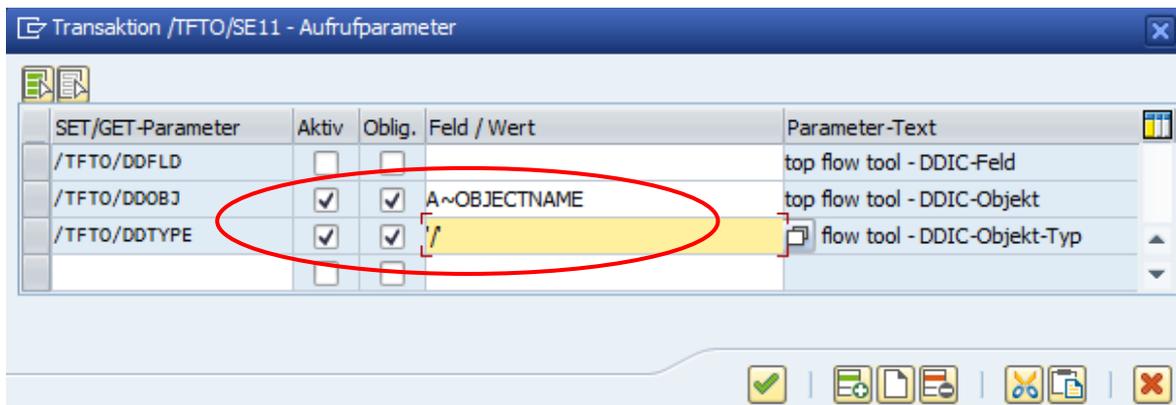
Im Parameter-Popup-Fenster tragen wir den Namen des entsprechenden Objekts ein:



An dieser Stelle müssen wir einen Wert für den **/TFTO/DDTYPE** angeben. Dieser Wert wäre normalerweise **“V”** für Views. Für CDS-View-Entitäten ist jedoch ein spezieller Wert **“/”** eingeführt worden, der bewirkt, dass /TFTO/SE11 die oben beschriebene gewünschte Feldstruktur anzeigt. Dieser spezielle Wert steht auch in der **F4-Hilfe für /TFTO/DDTYPE** zur Verfügung:



Nach Auswahl dieses Wertes sieht unsere Parameterdefinition wie folgt aus:



Nach Abschluss der Sprungdefinition können wir testen, ob der Sprung tatsächlich funktioniert. Also klicken wir auf den unten gezeigten OBJECTNAME:

SE16XXL - Outer-Join - 1782 Ergebnis-Sätze

Join von DDLDEPENDENCY(A) und DDLDEPENDENCY(B)

A~DDLNAME	A~OBJECTNAME	A~STATE	A~OBJECTTYPE	B~DDLNAME	B~OBJECTNAME	B~STATE	B~OBJECTTYPE
ABAP_DOCU_RELEASED_APIS	ABAP_DOCU_RELEASED_APIS	A	STOB				
ABAP_DOCU_TREE_SOURCE	ABAP_DOCU_TREE_SOURCE	A	STOB				
ABAP_LANGUAGE_VERSIONS	ABAP_LANGUAGE_VERSIONS	A	STOB				
ABAP_LANGUAGE_VERSIONS_SRC	ABAP_LANGUAGE_VERSIONS_SRC	A	STOB				
ABP_BEHV_DRAFT_USER_ACCESS	ABP_BEHV_DRAFT_USER_ACCESS	A	STOB				
ABP_BEHV_PAR_SHARE	ABP_BEHV_PAR_SHARE	A	STOB				
ACCTASGTBUSINESSTRANSACTION	ACCTASGTBUSINESSTRANSACTION	A	STOB				
ACMHEX_HIERS4DIM1_HIERARCHY	ACMHEX_HIERS4DIM1_HIERARCHY	A	STOB				
ACMRT_AINIT_DDI_INHERIT	ACMRT_AINIT_DDI_INHERIT	A	STOR				

Das Programm reagiert mit der Anzeige folgender Feldstruktur:

Struktur von CDS-View-Entität ACCTASGTBUSINESSTRANSACTION

CDS-View-Entität: ACCTASGTBUSINESSTRANSACTION

Kurzbeschreibung: FI Closing Valuation Discounting Run Result List Output

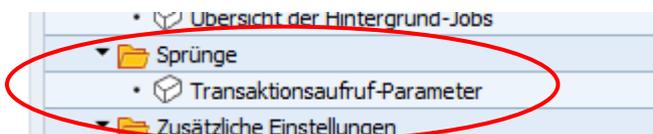
Datendefinition: ACCTASGTBUSINESSTRANSACTION

Felder: Währungs-/Mengenfelder

Viewfelder	Schl.	Datenelement	Datentyp	Länge	Dezima...	Kurzbeschreibung
BTTYPE_ITEM	<input checked="" type="checkbox"/>	FINS_BUS_TRANS_ITEM_CATEGORY	CHAR	10		Geschäftsvorfallspositionstyp
BTTYPE	<input checked="" type="checkbox"/>	FINS_BTTYPE	CHAR	4		Geschäftsvorfallstyp
CBTTYPE	<input checked="" type="checkbox"/>	FINS_ASGMT_CBTTYPE	CHAR	4		Geschäftsvorfallsart
CBTTYPE_TEXT	<input type="checkbox"/>	FINS_ASGMT_CBTTYPE_DESCRIPTION	CHAR	30		Geschäftsvorfallsartbeschreibung
BTTYPE_TEXT	<input type="checkbox"/>	FINS_ASGMT_BTTYPE_DESCRIPTION	CHAR	30		Beschreibung für Geschäftsvorfallstyp
BTTYPE_ITEM_TEXT	<input type="checkbox"/>	FINS_BUS_TRANS_ITEM_CAT_DESC	CHAR	50		Beschreibung für Geschäftsvorfallspositionstyp
IS_CONFIGURABLE	<input type="checkbox"/>	FINS_ASGMT_IS_CONFIGURABLE	CHAR	1		Kontierungsregel kann von dem Kunden konfiguriert werden
ASGMT_RULE	<input type="checkbox"/>	FINS_ASGMT_RULE_ID	CHAR	4		Zuordnungsregel

Der definierte Sprung scheint zu funktionieren!

ANMERKUNG: Um die oben besprochenen Sprünge definieren zu können, ist es notwendig, die Parameter-IDs zu aktualisieren, die den entsprechenden Transaktionscodes zugeordnet sind. Dies kann in den SE16XXL-Einstellungen unter dem Thema



Durchgeführt werden.

In der Liste der *Transaktionsaufruf-Parameter* nutzen Sie bitte die Menüfunktion *Einträge → Standardeinträge hinzufügen* für beide Transaktionscodes /TFTO/SE11 und /TFTO/SE16SCRIPT. Bevor Sie dies tun, löschen Sie bitte die aktuellen Einträge beider Transaktionscodes, um sicherzustellen, dass die neuen Set/Get-Parameter-IDs berücksichtigt werden.

[Zum Anfang](#)

Scriptspezifische ALV-Layouts

Bisher waren ALV-Layouts keinem bestimmten SE16XXL-Script zugeordnet. Vielmehr waren sie einer bestimmten Kombination von Tabellen oder Views in Bezug auf die Ergebnisliste zugeordnet. Mit anderen Worten, zwei SE16XXL-Scripts mit der gleichen Art von Ergebnisliste haben den gleichen Satz von ALV-Layouts verwendet.

Wäre z.B. die Ergebnisliste beider Scripts ein Join von VBAK + VBAP + VBEP, dann stünden dieser zwei Scripts alle ALV-Layouts zur Verfügung, die mit dieser Kombination von Tabellen verbunden sind. Diese ALV-Layouts würden auch dann zur Verfügung stehen, wenn eine solche Ergebnisliste in einer Dialogsitzung ohne Verwendung eines Scripts erstellt würde.

Diese scheinbare Flexibilität hat einen Nachteil: Sobald die Ergebnisliste eines Scripts geändert wird, z.B. durch Anfügen oder Einfügen eines neuen Joins, “verschwinden” sämtliche ALV-Layouts, die bis dahin für dieses Script zur Verfügung standen. Entweder ist nun ein neuer Satz von ALV-Layouts verfügbar, falls welche für die neue Kombination von Tabellen existieren, oder gar keine. Im Falle eines komplizierten globalen Scripts, für das zahlreiche spezialisierte ALV-Layouts erstellt wurden, würde eine solche Situation für viele Leute eine Menge zusätzlicher Arbeit bedeuten.

Um solche Probleme zu vermeiden, ist es ab der Version **3.6B** von SE16XXL möglich, **scriptspezifische ALV-Layouts** zu erstellen. Diese Layouts sind ausschließlich dem Script zugeordnet, für das sie erstellt wurden. Wird das Script geändert, werden alle seine scriptspezifischen ALV-Layouts entsprechend **angepasst** und stehen somit weiterhin zur Verfügung.

Beim Kopieren eines Scripts kann festgelegt werden, dass auch die zugehörigen scriptspezifischen ALV-Layouts berücksichtigt werden. Gleiches gilt für das Herunterladen oder Transportieren eines Scripts. Wenn ein Script umbenannt wird, werden dessen scriptspezifische Layouts automatisch umbenannt.

Von nun an werden die ALV-Layouts, die nicht scriptspezifisch sind, als “**generisch**” bezeichnet.

Außerdem wurde ein Tool zur Verwaltung von scriptspezifischen ALV-Layouts implementiert. Mit dessen Hilfe ist es unter anderem möglich, ausgewählte generische ALV-Layouts in scriptspezifische zu kopieren. Diese Funktionalität sollte den Übergang von den alten generischen Layouts zu den neuen scriptspezifischen Layouts erleichtern.

Weitere Informationen finden Sie unter “[Scriptspezifische ALV-Layouts](#)” sowie unter “[Tool zur Verwaltung scriptspezifischer ALV-Layouts](#)”.

[Zum Anfang](#)

Zusätzliche Verbesserungen

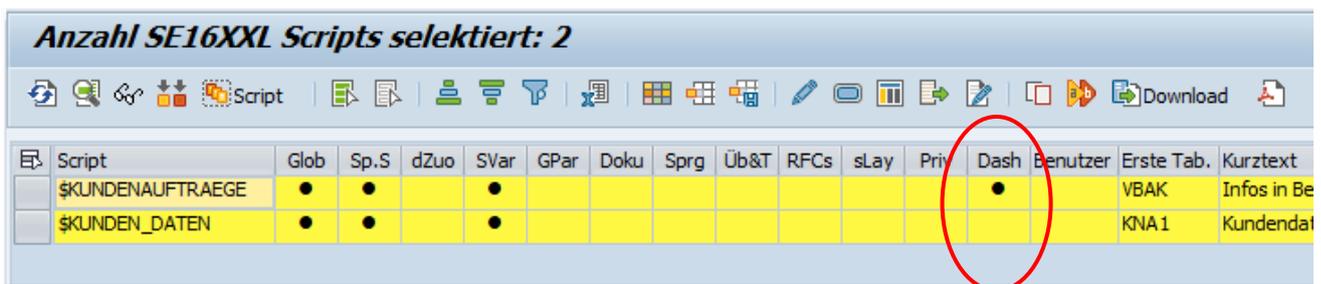
Flag “Relevant für Dashboarding” im Script-Katalog

Im Script-Katalog ist es nun möglich, das Flag “*relevant für Dashboarding*” für **ausgewählte globale Scripts** zu setzen bzw. zurückzusetzen. Dies kann durch die Verwendung folgender Menüfunktionen erreicht werden:

Ausgewählte Scripts → *Dashboarding* → “*relevant*” setzen
→ “*relevant*” zurücksetzen

Darüber hinaus wurde der Ausgabeliste eine **neue Spalte** hinzugefügt, die den Status dieses Flags für jedes globale Script darstellt:

Anzahl SE16XXL Scripts selektiert: 2

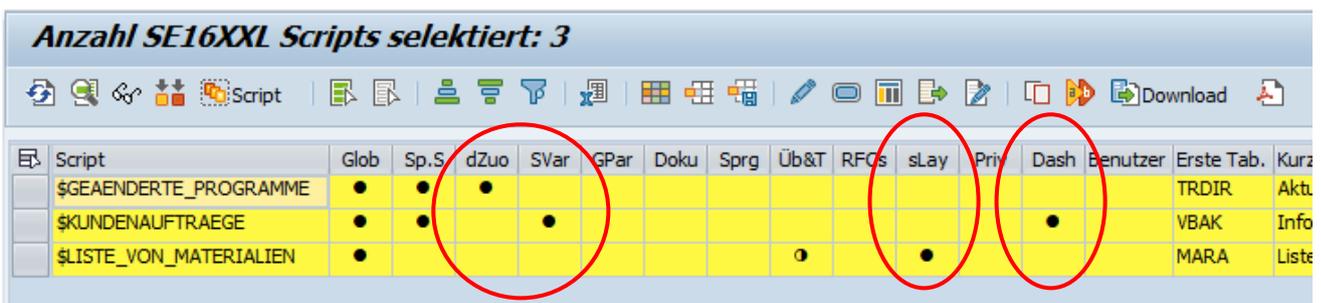


Script	Glob	Sp.S	dZuo	SVar	GPar	Doku	Sprg	Üb&T	RFCs	sLay	Pri	Dash	Benutzer	Erste Tab.	Kurztext
\$KUNDENAUFTRAEGE	•	•		•								•	VBAK		Infos in Be
\$KUNDEN_DATEN	•	•		•									KNA1		Kundendat

Zusätzliche Spalten in der Ausgabeliste des Script-Katalogs

Die Ausgabeliste des Script-Katalogs wurde um vier neue Spalten erweitert, wie in folgender Abbildung zu sehen ist:

Anzahl SE16XXL Scripts selektiert: 3



Script	Glob	Sp.S	dZuo	SVar	GPar	Doku	Sprg	Üb&T	RFCs	sLay	Pri	Dash	Benutzer	Erste Tab.	Kurz
\$GEAENDERTE_PROGRAMME	•	•	•											TRDIR	Aktu
\$KUNDENAUFTRAEGE	•	•		•								•	VBAK		Info
\$LISTE_VON_MATERIALIEN	•							•		•			MARA		Liste

Diese neuen Spalten haben informativen Charakter:

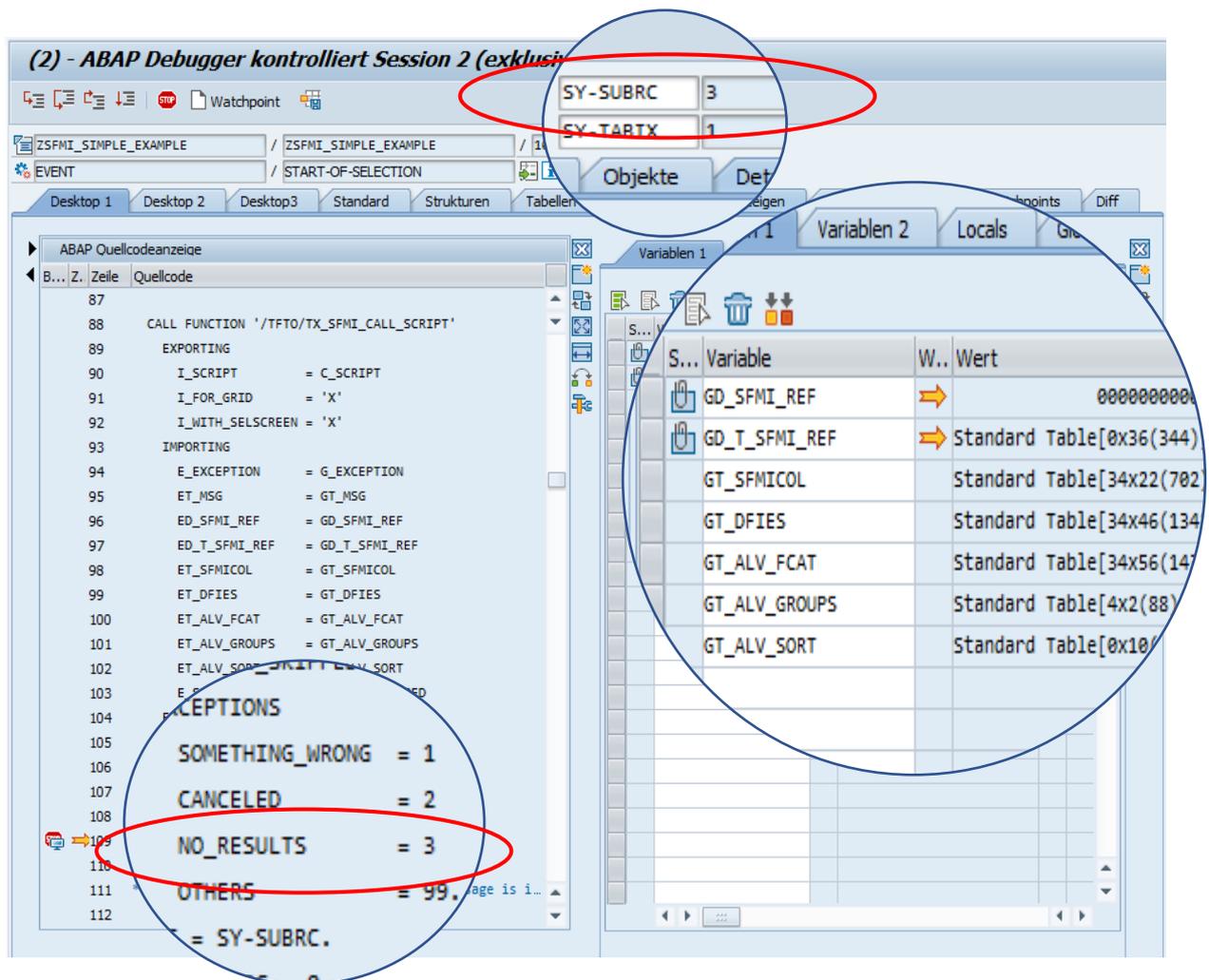
- **dZuo** Zeigt an, ob das Script mit dynamischen Zuordnungen ausgestattet ist.
- **SVar** Gibt an, ob Script-Varianten definiert sind.
- **sLay** Gibt an, ob scriptspezifische ALV-Layouts definiert sind.
- **Dash** Informiert darüber, ob das Script für das Dashboarding relevant ist.

Funktionsbaustein ‘/TFTO/TX_SFMI_CALL_SCRIPT’ verbessert

Der Funktionsbaustein '/TFTO/TX_SFMI_CALL_SCRIPT' kann aus einem ABAP-Programm heraus aufgerufen werden, um ein SE16XXL-Script auszuführen. Bisher, falls das Script keine Daten zurückgab, wurde die Ausnahme 'NO_RESULTS' ausgelöst und es wurden **keine zusätzlichen Metadaten** in Bezug auf die Struktur der Ergebnisse **zurückgeliefert**.

Ab der Version **3.6B** von SE16XXL wurde dieses Manko behoben. Auch im Falle dass das Script keine Daten selektiert (Ausnahme NO_RESULTS), werden die relevanten Metadaten **in jedem Fall** an das aufrufende Programm **zurückgeliefert**.

Dies kann folgendem Debugging-Beispiel entnommen werden:



The screenshot shows the ABAP Debugger interface. The top status bar indicates '(2) - ABAP Debugger kontrolliert Session 2 (exklusiv)'. The 'SY-SUBRC' variable is highlighted with a red circle and contains the value 3. The 'SY-TARTX' variable is highlighted with a blue circle and contains the value 1. The 'ABAP Quellcodeanzeige' window shows the function call and the 'EXCEPTIONS' section, where 'NO_RESULTS = 3' is highlighted with a red circle. The 'Variablen 1' window shows a list of variables including GD_SFMI_REF, GD_T_SFMI_REF, GT_SFMICOL, GT_DFIES, GT_ALV_FCAT, GT_ALV_GROUPS, and GT_ALV_SORT, all with their respective values.

S...	Variable	W.. Wert
	GD_SFMI_REF	0000000000
	GD_T_SFMI_REF	Standard Table[0x36(344)
	GT_SFMICOL	Standard Table[34x22(702)
	GT_DFIES	Standard Table[34x46(134)
	GT_ALV_FCAT	Standard Table[34x56(14)
	GT_ALV_GROUPS	Standard Table[4x2(88)
	GT_ALV_SORT	Standard Table[0x10(

[Zum Anfang](#)

Neue Art von globalem Parameter – Zeitraum

Bisher konnten **globale Parameter** in Formeln verwendet, auf der speziellen Selektionsmaske eines Scripts platziert und bei der Gültigkeitsprüfung während einer Join-Operation adressiert werden.

Ab Version 3.6A von SE16XXL können sie auch zu einzelnen **Select-Options** (und/oder Selektionsparametern) der speziellen Selektionsmaske **zugeordnet werden**. Auf diese Weise kann eine Reihe von Select-Options mit den gleichen Werten versehen werden, ohne dass diese Werte für jede Select-Option separat angegeben werden müssen (weitere Informationen finden Sie im [nächsten Thema](#)).

Um dieser Funktionalität mehr Flexibilität zu verleihen, wurde eine neue Art von globalem Parameter eingeführt, ein Zeitraum. Dies ist besonders nützlich bei der Selektion von Datensätzen, deren **Gültigkeit** durch **zwei Felder** definiert wird, wie z.B. BEGDA und ENDDA der Tabelle PA0000.

Die Definition dieser Art von globalen Parametern ist ganz einfach:

Extras → Globale Parameter definieren

Wählen Sie im nachfolgenden Dialogfenster **“Zeitraum”** aus der Dropdown-Liste:

Parameter	Art	Tabellenname	Feldname
ZEITRAUM	<div style="border: 1px solid black; padding: 2px;"> Parameter Select-Option Sprache für V-Texte Zeitraum </div>		

Sobald dies geschehen ist, füllt das Programm die Tabellen- und Feldnamen **automatisch** aus:

Definition von Globalen Parametern					
Parameter	Art	Tabellenname	Feldname		Initialwert(e)
ZEITRAUM	Zeitraum	SYST	DATUM		

Ein Beispiel finden Sie im [nächsten Thema](#).

[Zum Anfang](#)

Zuordnung von globalen Parametern zu Select-Options

Ab Version 3.6A von SE16XXL ist es möglich, einer oder mehreren Select-Options (bzw. Selektionsparametern) der speziellen Selektionsmaske eines Scripts einen globalen Parameter zuzuordnen.

Diese Funktion ist nützlich, wenn für zwei oder mehr Select-Options eines bestimmten Scripts dieselben Selektionswerte erforderlich sind. Oder wenn derselbe Wert für einige Selektionsparameter verschiedener Views (oder Pseudo-Tabellen) benötigt wird.

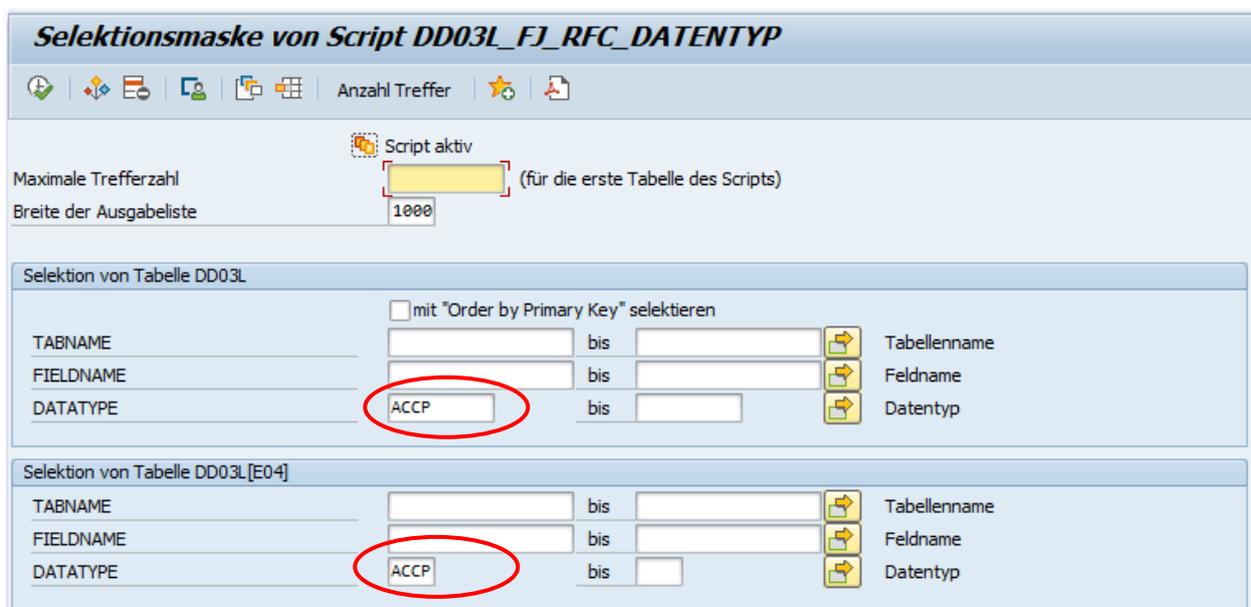
Darüber hinaus, wenn eine Tabelle Datensätze enthält, deren Gültigkeit durch zwei Felder bestimmt wird, wie z.B. BEGDA und ENDDA der PAXxxx HR-Tabellen, kann der neu eingeführte globale Parameter der Art "Zeitraum" verwendet werden. Dieser erleichtert die Handhabung dieser Art von Selektion.

Ein paar Beispiele werden diese neue Funktionalität veranschaulichen.

Beispiel #1

Angenommen, Sie haben ein Script, das einen Full-Join zwischen den Tabellen-feldern eines bestimmten Datentyps des lokalen Systems mit dem gleichen Typ eines entfernten Systems durchführt.

Bisher hätte die spezielle Selektionsmaske wie folgt ausgesehen:



Selektionsmaske von Script DD03L_FJ RFC_DATENTYP

Maximale Trefferzahl: (für die erste Tabelle des Scripts)

Breite der Ausgabeliste:

Script aktiv

mit "Order by Primary Key" selektieren

Parameter	Wert	Operator	Wert	Parameter
TABNAME	<input type="text"/>	bis	<input type="text"/>	Tabellenname
FIELDNAME	<input type="text"/>	bis	<input type="text"/>	Feldname
DATATYPE	<input type="text" value="ACCP"/>	bis	<input type="text"/>	Datentyp

Parameter	Wert	Operator	Wert	Parameter
TABNAME	<input type="text"/>	bis	<input type="text"/>	Tabellenname
FIELDNAME	<input type="text"/>	bis	<input type="text"/>	Feldname
DATATYPE	<input type="text" value="ACCP"/>	bis	<input type="text"/>	Datentyp

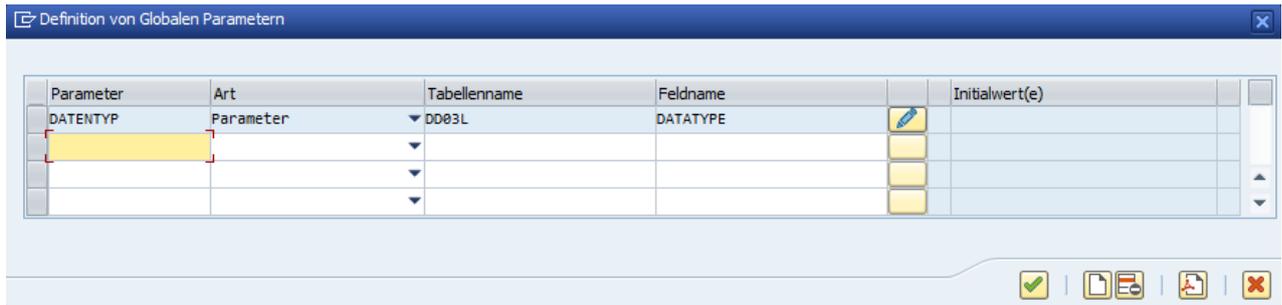
Damit das Script ein sinnvolles Ergebnis liefert, müssen beide DATATYPE-Select-Options mit demselben Wert versorgt werden.

Durch die Verwendung eines globalen Parameters kann das Script verbessert werden.

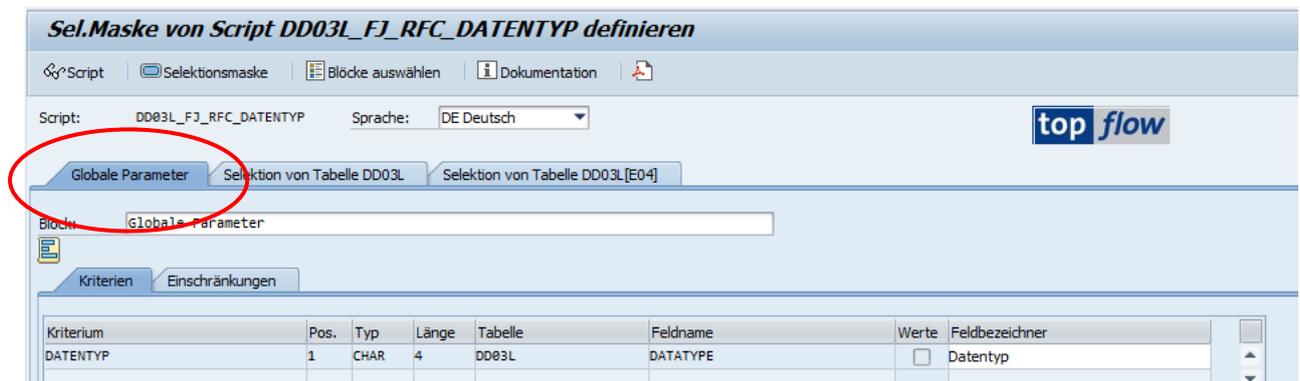
Wir beginnen mit der Definition des globalen Parameters. Dies kann direkt im **Script-Katalog** über folgende Menüfunktion erfolgen:

Script-Zusätze → Globale Parameter definieren

Die Definition könnte wie folgt lauten:

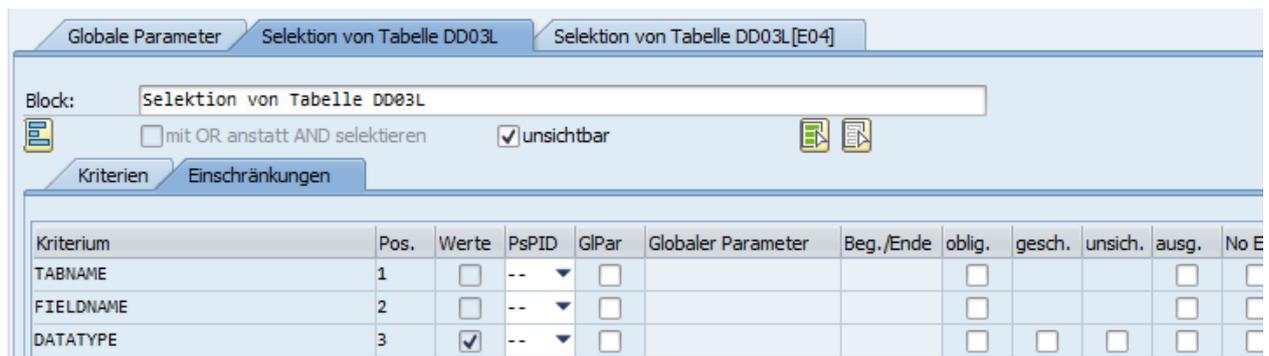


Nun wechseln wir zur Definition der speziellen Selektionsmaske ():



Wie man sehen kann, ist der Block mit den globalen Parametern automatisch aufgebaut worden.

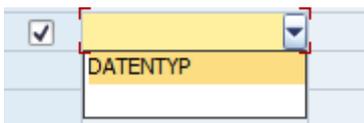
Um den globalen Parameter den Select-Options zuzuordnen, wechseln wir zu den Einschränkungen für die erste Tabelle:



Für die Select-Option DATATYPE aktivieren wir das Ankreuzfeld “GIPar”:

Kriterium	Pos.	Werte	PsPID	GIPar	Globaler Parameter	Beg./Ende	oblig.	gesch.	unsich.	ausg.	No E
TABNAME	1	<input type="checkbox"/>	-- ▾	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
FIELDNAME	2	<input type="checkbox"/>	-- ▾	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DATATYPE	3			<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>

Aus der Dropdown-Liste wählen wir unseren neu definierten globalen Parameter:



Das Ergebnis ist:

Kriterium	Pos.	Werte	PsPID	GIPar	Globaler Parameter	Beg./Ende	oblig.	gesch.	unsich.	ausg.	No E
TABNAME	1	<input type="checkbox"/>	-- ▾	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
FIELDNAME	2	<input type="checkbox"/>	-- ▾	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DATATYPE	3			<input checked="" type="checkbox"/>	DATENTYP ▾			<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>

ANMERKUNG: Die Select-Option ist als “geschützt” gekennzeichnet. Wird diese Option deaktiviert, schaltet die Select-Option automatisch auf “unsichtbar”. Mit anderen Worten, eine Select-Option, die einem globalen Parameter zugeordnet wurde, kann nur **entweder geschützt oder unsichtbar** sein. Dies ist notwendig, da die Werte nicht änderbar sein sollten.

Wir wiederholen die Prozedur für die zweite Selektion des Scripts.

Darüber hinaus, um die Anpassung der Selektionsmaske abzuschließen, setzen wir außerdem den globalen Parameter DATENTYP auf “obligatorisch”:

Sel.Maske von Script DD03L_FJ RFC_DATENTYP definieren

Script: DD03L_FJ RFC_DATENTYP Sprache: DE Deutsch 

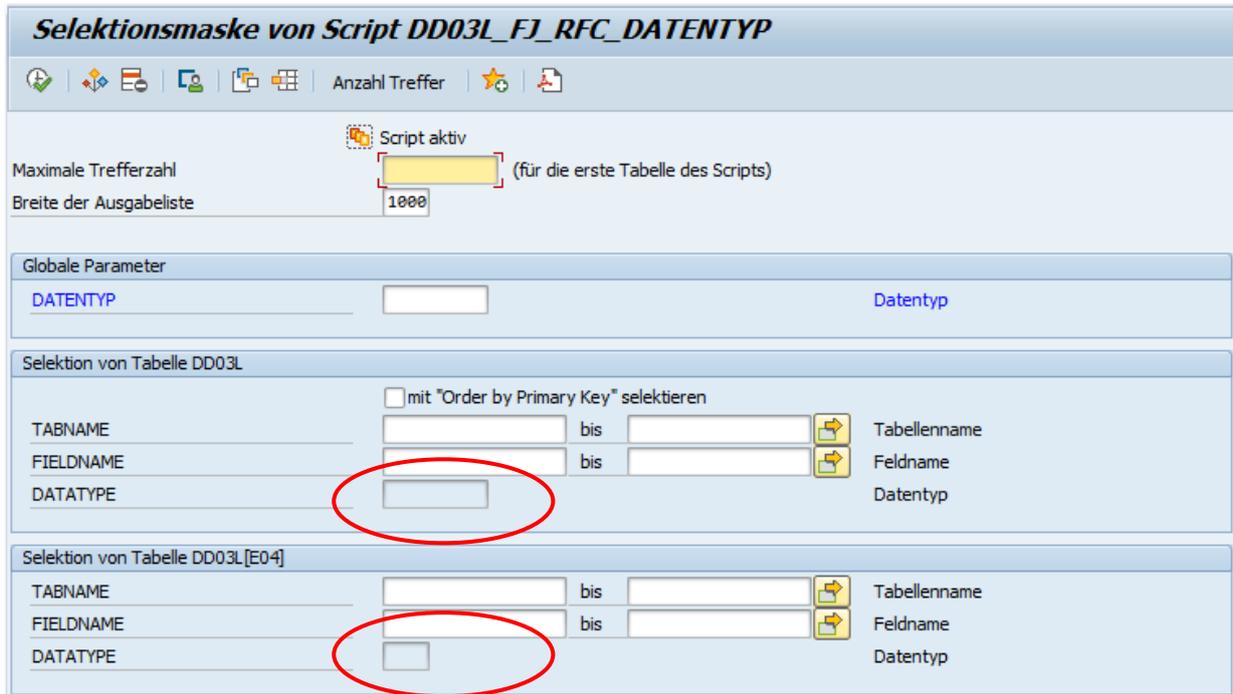
Globale Parameter | Selektion von Tabelle DD03L | Selektion von Tabelle DD03L[E04]

Block: Globale Parameter

Kriterien | Einschränkungen

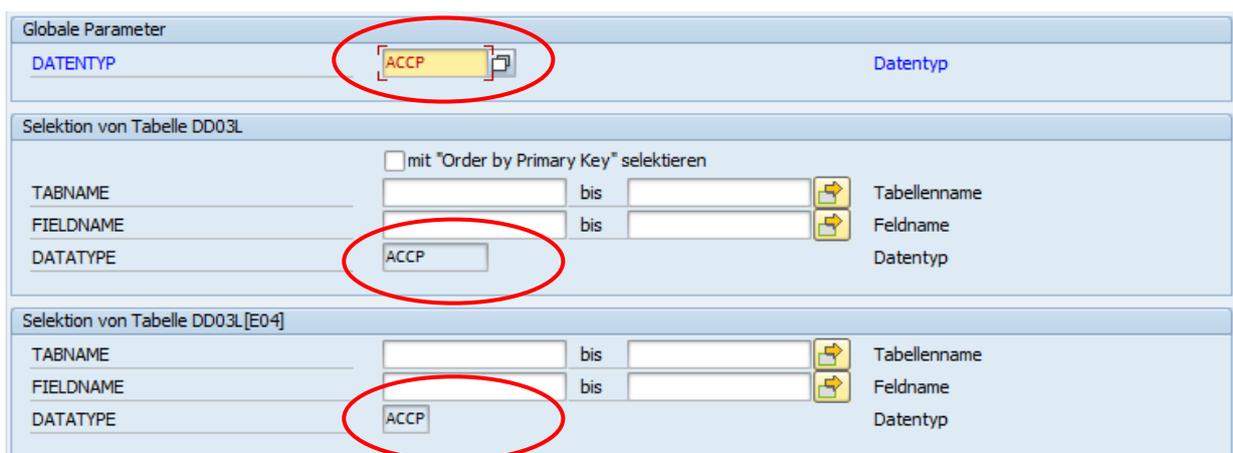
Kriterium	Pos.	Werte	PsPID	oblig.	gesch.	unsich.	ausg.	No E...	No Exc	Alle ...	EQ	NE	BT	NB	CP	NP	GE	GT	LE	LT
DATENTYP	1	<input type="checkbox"/>	▾	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>								

Die resultierende Selektionsmaske sieht wie folgt aus:



ANMERKUNG: Der Block mit dem globalen Parameter befindet sich oberhalb der anderen Blöcke. Bisher wäre dieser Block der zweite Block der Selektionsmaske gewesen. Aufgrund seiner neu gewonnenen Bedeutung wurde er jedoch ganz oben platziert.

Wenn wir nun über die F4-Hilfe einen DATENTYP auswählen, wird dieser Wert auch den beiden Select-Options der Selektionsblöcke zugewiesen:

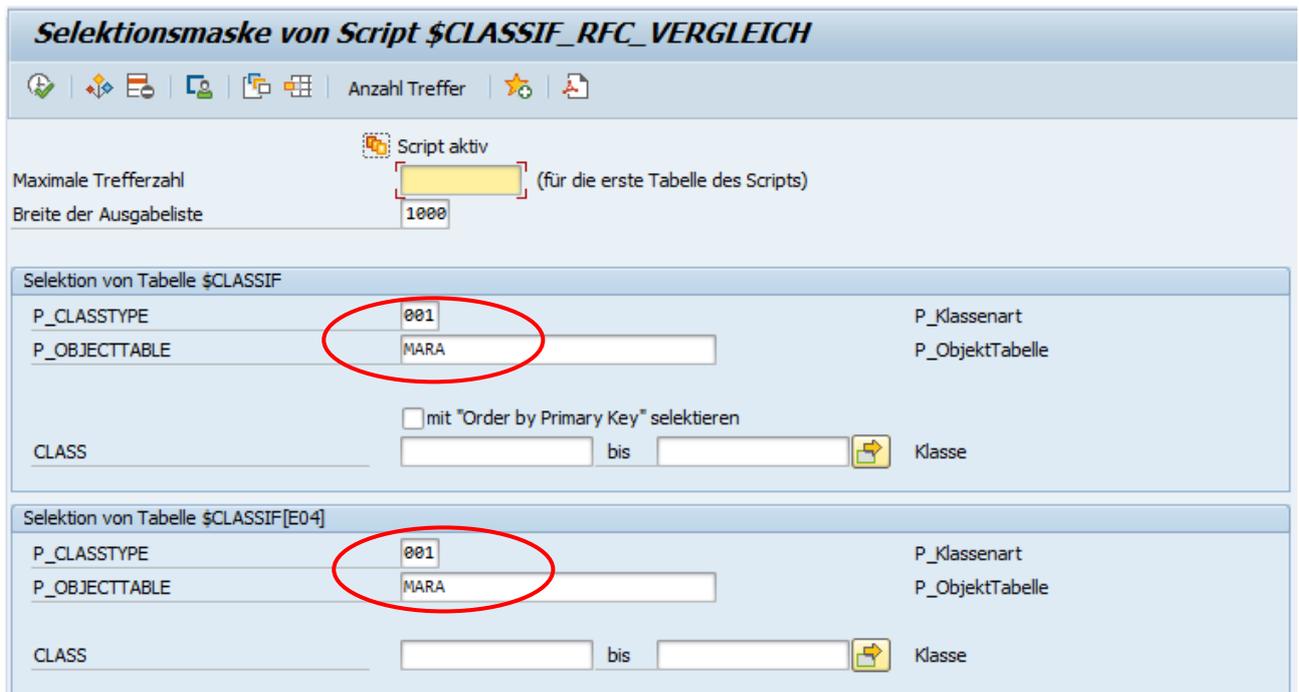


ANMERKUNG: Die Select-Option DATATYPE im RFC-Selektionsblock ist kürzer als die andere, da sie die internen Werte anstelle der externen anzeigt (siehe die Diskussion zu RFC-Select-Options mit einem Konvertierungsexit).

Beispiel #2

In diesem Beispiel haben wir es mit einem Script zu tun, das Einträge von Pseudo-Tabelle \$CLASSIF zwischen dem lokalen und einem entfernten System vergleicht.

Die traditionelle Selektionsmaske wäre wie im folgenden Bild:



Selektionsmaske von Script \$CLASSIF RFC VERGLEICH

Script aktiv

Maximale Trefferzahl: (für die erste Tabelle des Scripts)

Breite der Ausgabeliste:

Selektion von Tabelle \$CLASSIF

P_CLASSTYPE: P_Klassenart

P_OBJECTTABLE: P_ObjektTabelle

mit "Order by Primary Key" selektieren

CLASS: bis Klasse

Selektion von Tabelle \$CLASSIF[E04]

P_CLASSTYPE: P_Klassenart

P_OBJECTTABLE: P_ObjektTabelle

CLASS: bis Klasse

In diesem Fall sollten die beiden Parameter P_CLASSTYPE und P_OBJECTTABLE für beide Systeme auf identische Werte gesetzt werden, um ein sinnvolles Ergebnis zu erhalten. Die Select-Option CLASS sollte entweder gar nicht oder für beide Selektionen gleich ausgefüllt werden.

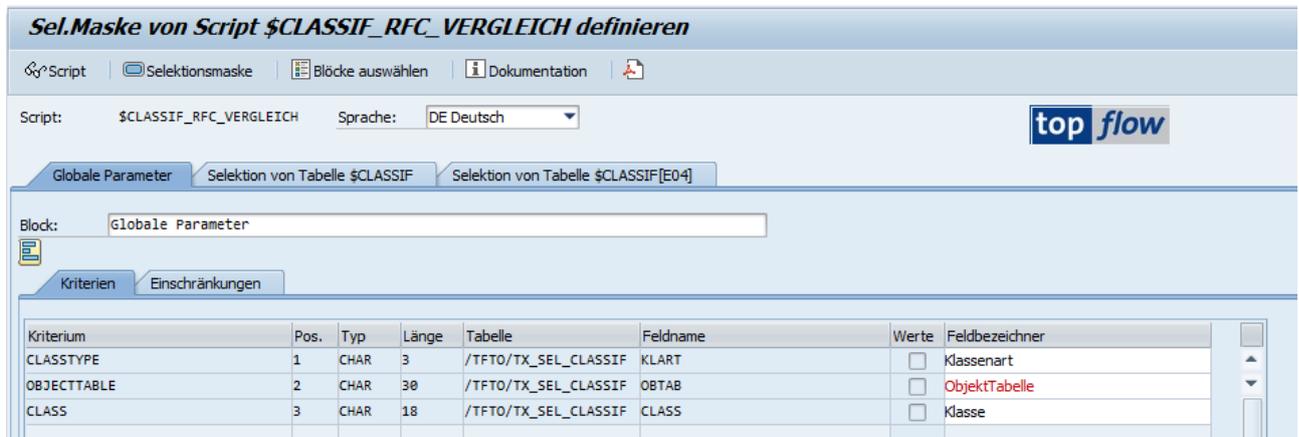
Nun verbessern wir dieses Script, indem wir globale Parameter einsetzen.

Wir beginnen mit der Definition der benötigten globalen Parameter:

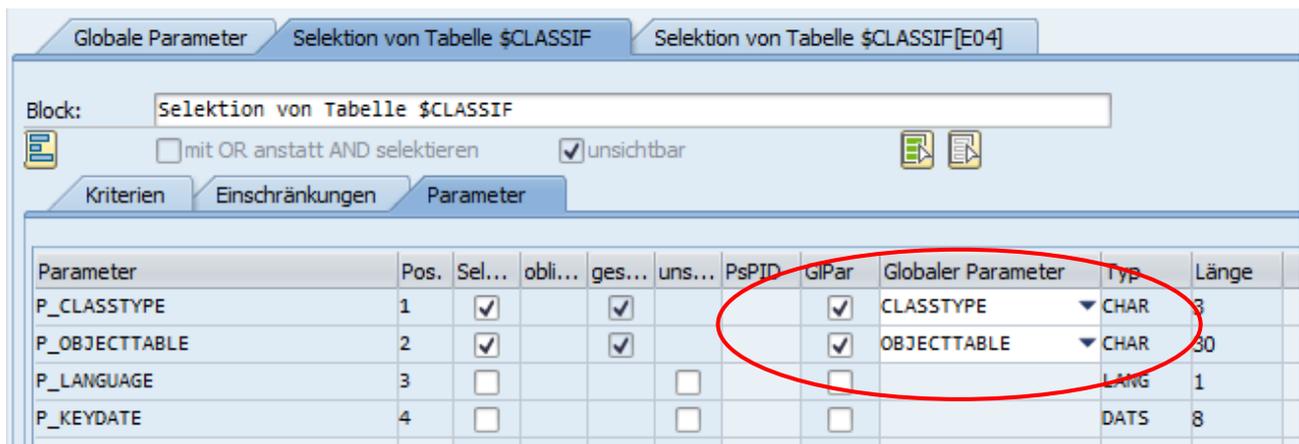
Definition von Globalen Parametern

Parameter	Art	Tabellenname	Feldname		Initialwert(e)
CLASSTYPE	Parameter	/TFTO/TX_SEL_CLASSIF	KLART		
OBJECTTABLE	Parameter	/TFTO/TX_SEL_CLASSIF	OBTAB		
CLASS	Select-Option	/TFTO/TX_SEL_CLASSIF	CLASS		
					

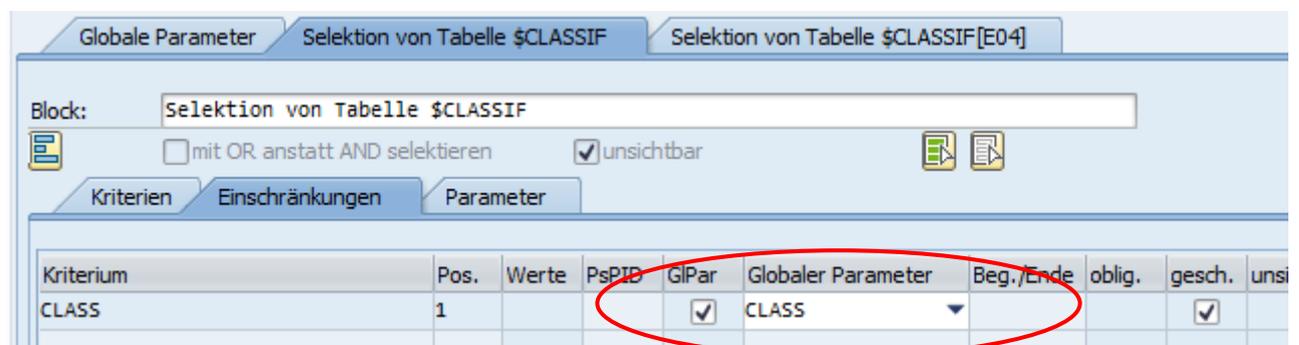
Dann rufen wir die Definition der speziellen Selektionsmaske auf und ändern dabei den Bezeichner für OBJECTTABLE von “Tabelle” nach “ObjektTabelle”:



Abschließend ordnen wir die relevanten globalen Parametern den Selektionsparametern zu:

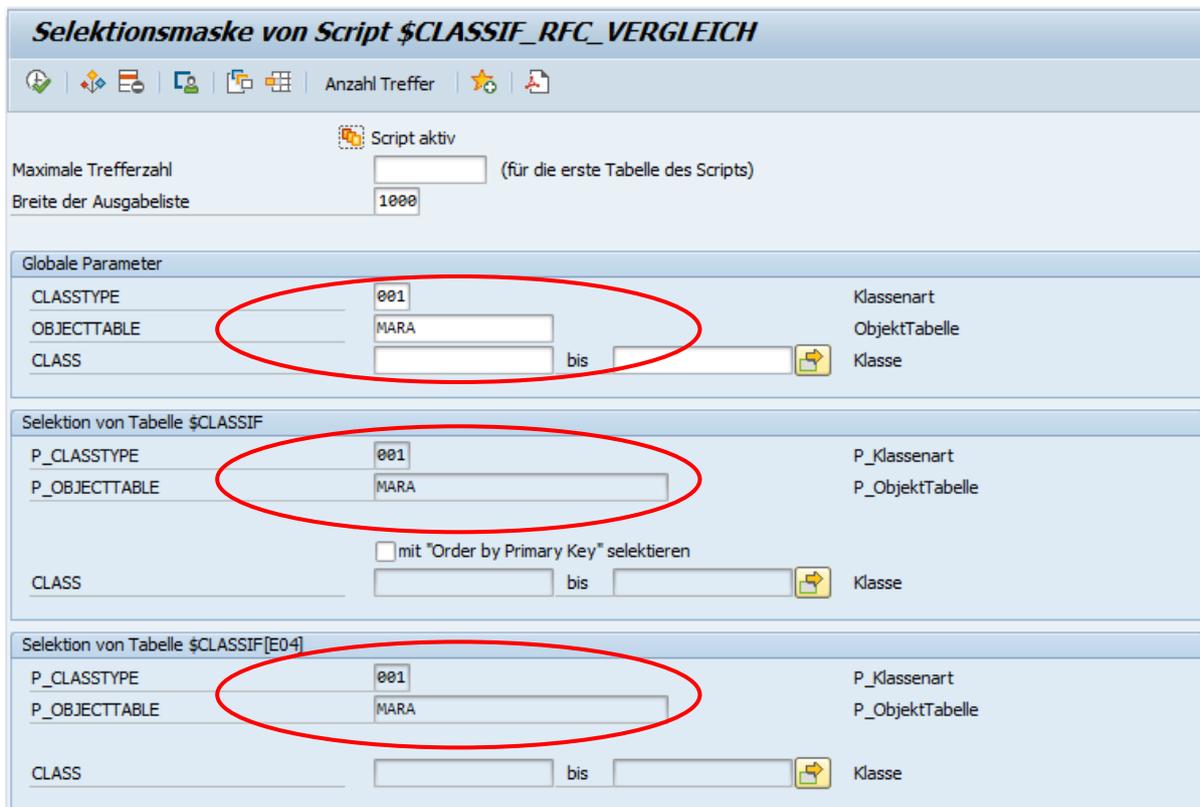


Und der Select-Option CLASS ebenfalls:



Wie im ersten Beispiel wird diese Prozedur für die zweite Selektion wiederholt.

Die resultierende spezielle Selektionsmaske sieht wie folgt aus:



Selektionsmaske von Script \$CLASSIF RFC VERGLEICH

Script aktiv

Maximale Trefferzahl (für die erste Tabelle des Scripts)

Breite der Ausgabeliste

Globale Parameter

CLASSTYPE	<input type="text" value="001"/>	Klassenart
OBJECTTABLE	<input type="text" value="MARA"/>	ObjektTabelle
CLASS	<input type="text"/> bis <input type="text"/>	Klasse

Selektion von Tabelle \$CLASSIF

P_CLASSTYPE	<input type="text" value="001"/>	P_Klassenart
P_OBJECTTABLE	<input type="text" value="MARA"/>	P_ObjektTabelle
CLASS	<input type="text"/> bis <input type="text"/>	Klasse

mit "Order by Primary Key" selektieren

Selektion von Tabelle \$CLASSIF[E04]

P_CLASSTYPE	<input type="text" value="001"/>	P_Klassenart
P_OBJECTTABLE	<input type="text" value="MARA"/>	P_ObjektTabelle
CLASS	<input type="text"/> bis <input type="text"/>	Klasse

ANMERKUNG: Die Verwendung der Struktur `/TFTO/TX_SEL_CLASSIF` als Tabellename für die globalen Parameterfelder stellt sicher, dass die F4-Hilfe für diese Felder genauso funktioniert wie für die Selektionsparameter der Pseudo-Tabelle `$CLASSIF` selbst.

Beispiel #3

In diesem letzten Beispiel werden wir uns den neuen globalen Parameter der Art **“Zeitraum”** genauer ansehen. Wie bereits in der Einleitung angedeutet, ist ein solcher globaler Parameter nützlich, wenn die betreffende Tabelle Datensätze enthält, deren Gültigkeit durch zwei Felder definiert wird, wie zum Beispiel die Felder `DATAB` und `DATBI` der Konditionstabellen **Annn** oder **BEGDA** und **ENDDA** der HR-Tabellen **PAnnnn**.

Unser Ausgangspunkt ist ein Script, das Datensätze der Tabelle `PA0000` (Personal-Stammsatz Infotyp 0000 (Massnahmen)) selektiert. Das Ergebnis sollte nur die Sätze umfassen, die im ersten Quartal 2022, d.h. vom 1.1.2022 bis zum 31.3.2022, gültig waren.

Die traditionelle Selektionsmaske müsste wie auf der nächsten Seite gezeigt ausgefüllt werden:

Selektionsmaske von Script BEISPIEL_MIT_PA0000

Script aktiv

Maximale Trefferzahl: 2.000 (für die erste Tabelle des Scripts)

Breite der Ausgabeliste: 1000

Selektion von Tabelle PA0000

mit OR anstatt AND selektieren

mit "Order by Primary Key" selektieren

PERNR		bis			Personalnr
SUBTY		bis			Subtyp
OBJPS		bis			Objekt-ID
SPRPS		bis			Sperrknz
BEGDA	31.03.2022	bis			Gültig ab
ENDDA	01.01.2022	bis			Gültig bis
SEQNR		bis			IT-SatzNr
AEDTM		bis			Geändert am
UNAME		bis			Geändert von

ANMERKUNG: Auf der obigen Selektionsmaske wurde BEGDA vor ENDDA platziert. Im Data Dictionary kommt ENDDA zuerst.

Um die gewünschten Sätze zu selektieren, ist es notwendig, die Werte für BEGDA und ENDDA wie oben gezeigt anzugeben. Dies ist nicht ohne weiteres verständlich und sicherlich fehleranfällig. Wie wir jetzt zeigen werden, erleichtert ein globaler Parameter die Handhabung dieser Situation erheblich.

Zunächst definieren wir unseren globalen Parameter:

Definition von Globalen Parametern

Parameter	Art	Tabellename	Feldname
GUELTIGKEIT	Zeitraum	SYST	DATUM

Dann wechseln wir zur Definition der speziellen Selektionsmaske und ändern dabei den Feldbezeichner von GUELTIGKEIT vom Standardwert "aktuelles Datum" auf "Gültigkeitszeitraum":

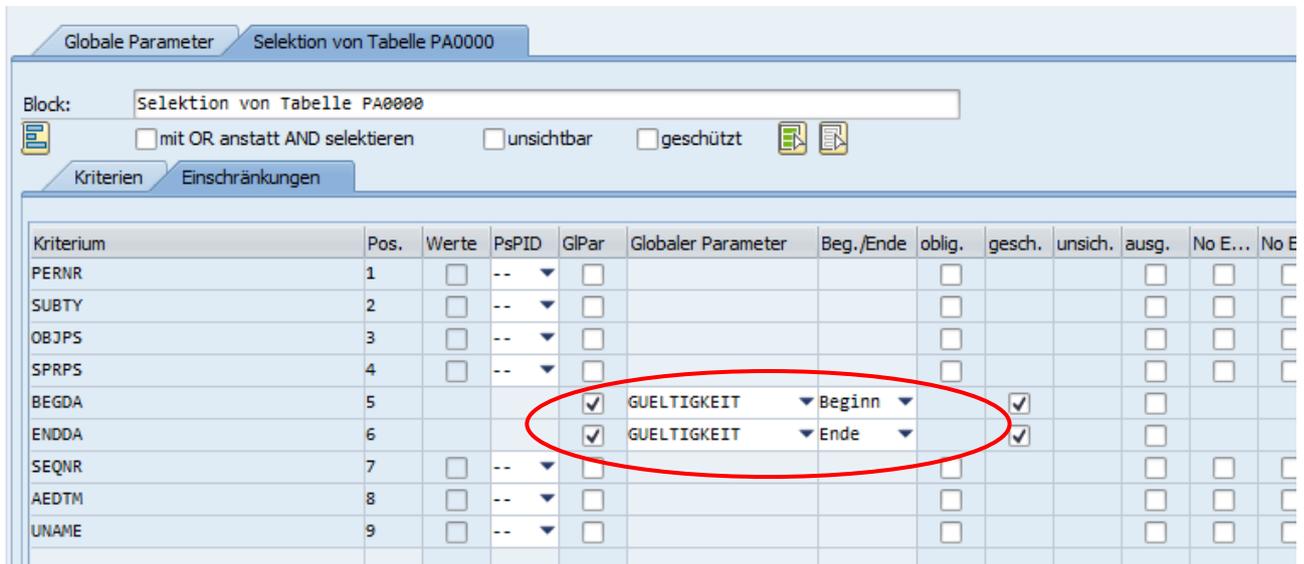
Globale Parameter Selektion von Tabelle PA0000

Block: Globale Parameter

Kriterien

Kriterium	Pos.	Typ	Länge	Tabelle	Feldname	Werte	Feldbezeichner
GUELTIGKEIT	1	DATS	8	SYST	DATUM	<input checked="" type="checkbox"/>	Gültigkeitszeitraum

An dieser Stelle ordnen wir unseren beiden Select-Options BEGDA und ENDDA den globalen Parameter zu:



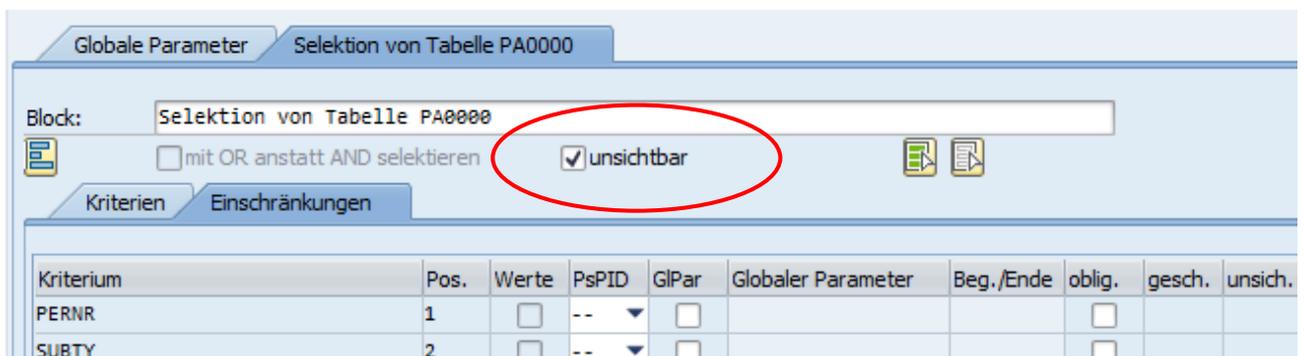
Kriterium	Pos.	Werte	PsPID	GLPar	Globaler Parameter	Beg./Ende	oblig.	gesch.	unsich.	ausg.	No E...	No E
PERNR	1	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SUBTY	2	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
OBJPS	3	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SPRPS	4	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BEGDA	5			<input checked="" type="checkbox"/>	GUELTIGKEIT	Beginn	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ENDDA	6			<input checked="" type="checkbox"/>	GUELTIGKEIT	Ende	<input type="checkbox"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SEQNR	7	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
AEDTM	8	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UNAME	9	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

ANMERKUNG: Neben dem Namen des globalen Parameters (GUELTIGKEIT) wurde auch das Flag **“Beg./Ende”** angegeben. Das Programm benötigt diese Informationen, um die Selektionswerte korrekt einzustellen.

Wenn wir versuchen, die Definition zu sichern, reagiert das Programm mit folgender Fehlermeldung:

 Option "Select mit OR" nicht erlaubt mit Beg/Ende-Flag für Glob.Params !

Der Grund dafür ist, dass es bei der Versorgung der beiden Select-Options wie zuvor beschrieben keinen Sinn macht, die Option **“mit OR anstatt AND selektieren”** zu aktivieren. Würde dies getan, wäre das Ergebnis anders als erwartet. Daher muss die Option deaktiviert werden, indem sie auf **“unsichtbar”** gesetzt wird, wie unten gezeigt:



Kriterium	Pos.	Werte	PsPID	GLPar	Globaler Parameter	Beg./Ende	oblig.	gesch.	unsich.
PERNR	1	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>		
SUBTY	2	<input type="checkbox"/>	--	<input type="checkbox"/>			<input type="checkbox"/>		

Unsere Selektionsmaske ist nun vollständig und wir können das Script ausführen.

Die spezielle Selektionsmaske erscheint wie im folgenden Bild:

Selektionsmaske von Script BEISPIEL_MIT_PA0000


 Anzahl Treffer  

 Script aktiv
 Maximale Trefferzahl (für die erste Tabelle des Scripts)
 Breite der Ausgabeliste

Globale Parameter

GUELTIGKEIT bis Gültigkeitszeitraum

Selektion von Tabelle PA0000

mit "Order by Primary Key" selektieren

PERNR	<input type="text"/>	bis	<input type="text"/>	 Personalnr
SUBTY	<input type="text"/>	bis	<input type="text"/>	 Subtyp
OBJPS	<input type="text"/>	bis	<input type="text"/>	 Objekt-ID
SPRPS	<input type="text"/>	bis	<input type="text"/>	 Sperrknz
BEGDA	<input type="text"/>			Gültig ab
ENDDA	<input type="text"/>			Gültig bis
SEQNR	<input type="text"/>	bis	<input type="text"/>	 IT-SatzNr
AEDTM	<input type="text"/>	bis	<input type="text"/>	 Geändert am
UNAME	<input type="text"/>	bis	<input type="text"/>	 Geändert von

Sobald wir die beiden Werte eingeben, die unseren Gültigkeitszeitraum definieren, nehmen die beiden abhängigen Select-Options ihre gewünschten Werte an:

Globale Parameter

GUELTIGKEIT bis Gültigkeitszeitraum

Selektion von Tabelle PA0000

mit "Order by Primary Key" selektieren

PERNR	<input type="text"/>	bis	<input type="text"/>	 Personalnr
SUBTY	<input type="text"/>	bis	<input type="text"/>	 Subtyp
OBJPS	<input type="text"/>	bis	<input type="text"/>	 Objekt-ID
SPRPS	<input type="text"/>	bis	<input type="text"/>	 Sperrknz
BEGDA	 31.03.2022			Gültig ab
ENDDA	 01.01.2022			Gültig bis
SEQNR	<input type="text"/>	bis	<input type="text"/>	 IT-SatzNr
AEDTM	<input type="text"/>	bis	<input type="text"/>	 Geändert am
UNAME	<input type="text"/>	bis	<input type="text"/>	 Geändert von

Das Spezifizieren eines Gültigkeitszeitraums als **von ... bis...** ist viel einfacher, als in den beiden Select-Options die erforderlichen Werte einzugeben.

[Zum Anfang](#)

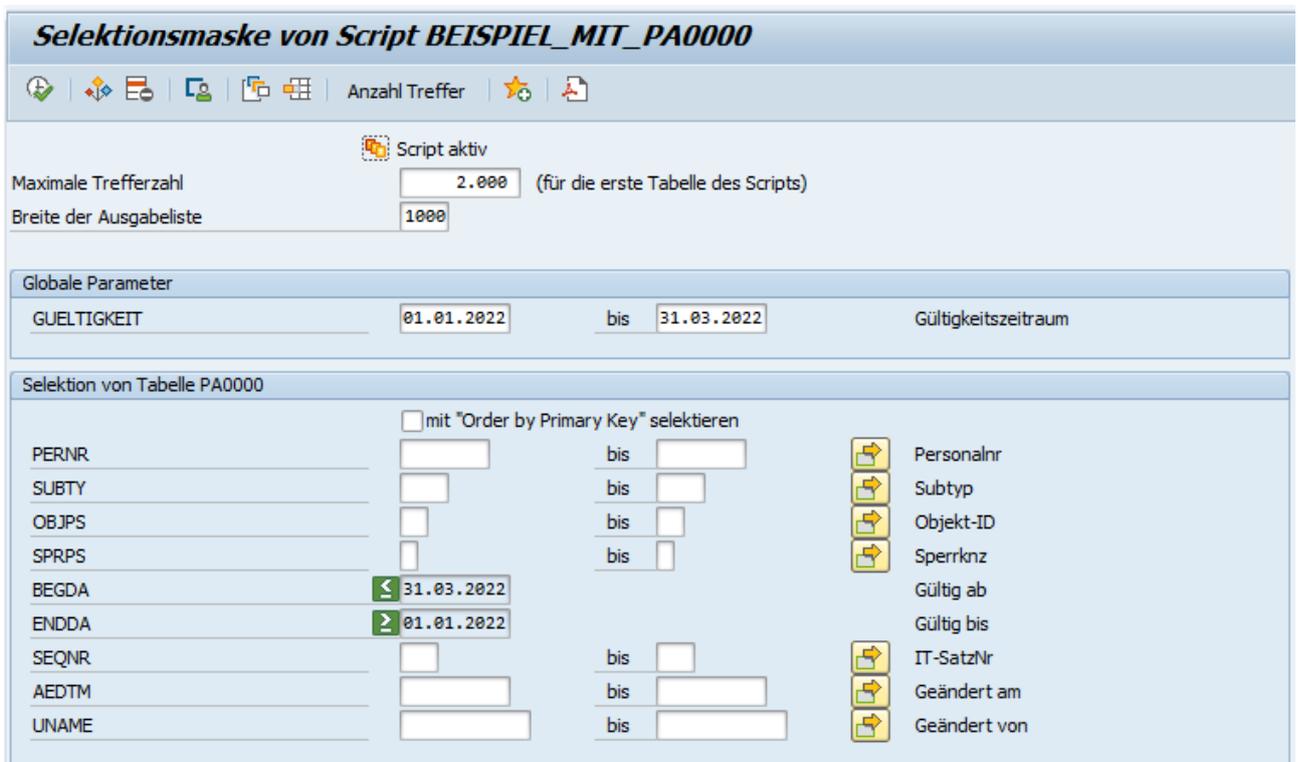
Spezielle Selektionsmaske – dynamische Zuordnungen

Bisher war es möglich, dynamische Variablen (wie Datumsvariablen etc.) einzelnen Select-Options der speziellen Selektionsmaske zuzuordnen. Dies konnte jedoch nur mithilfe einer **Script-Variante** erfolgen. Script-Varianten sind für die meisten Zwecke praktisch, sie sind jedoch immer **optional**, d.h. das Script kann auch ohne Verwendung einer Variante ausgeführt werden. Selbst eine Standard-Scriptvariante kann übersteuert werden.

Ab Version 3.6A von SE16XXL ist es möglich, dynamische Variablen **direkt** der speziellen Selektionsmaske eines Scripts **zuzuordnen, ohne** dass eine Variante erforderlich ist. Auf diese Weise ist es z.B. möglich, das Ergebnis des Scripts auf Datensätze **zu beschränken**, die im aktuellen Monat erstellt wurden. Man bedient sich dabei einerseits der dynamischen Werte einer Datumsvariablen und andererseits der Möglichkeit, die betreffende Select-Option vor Änderungen zu schützen.

Um diese neue Funktionalität zu veranschaulichen, nehmen wir das letzte Beispiel aus dem vorherigen Thema und passen das Script entsprechend an.

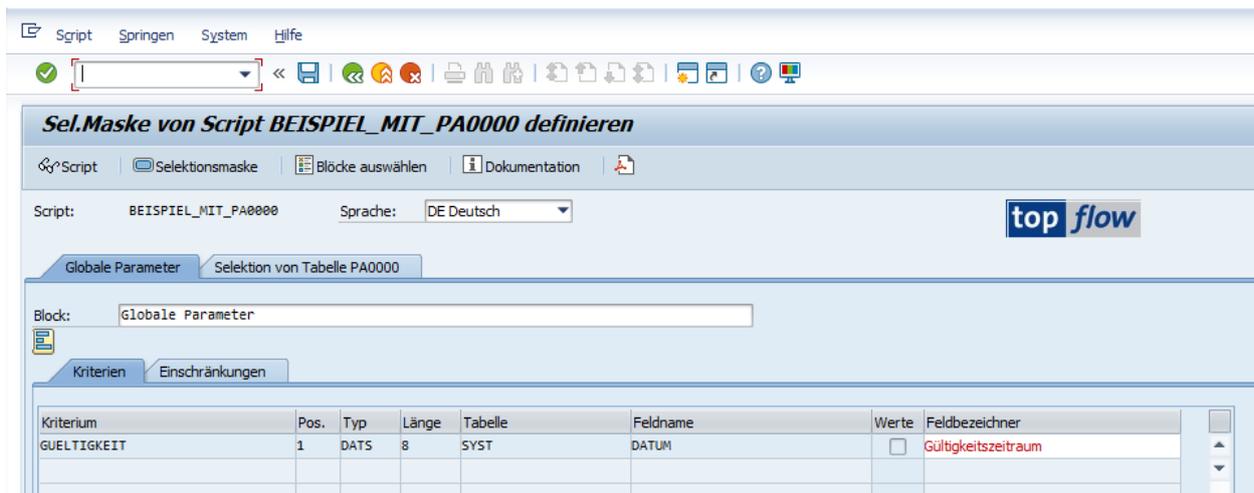
Die Selektionsmaske des betreffenden Scripts sah wie folgt aus:



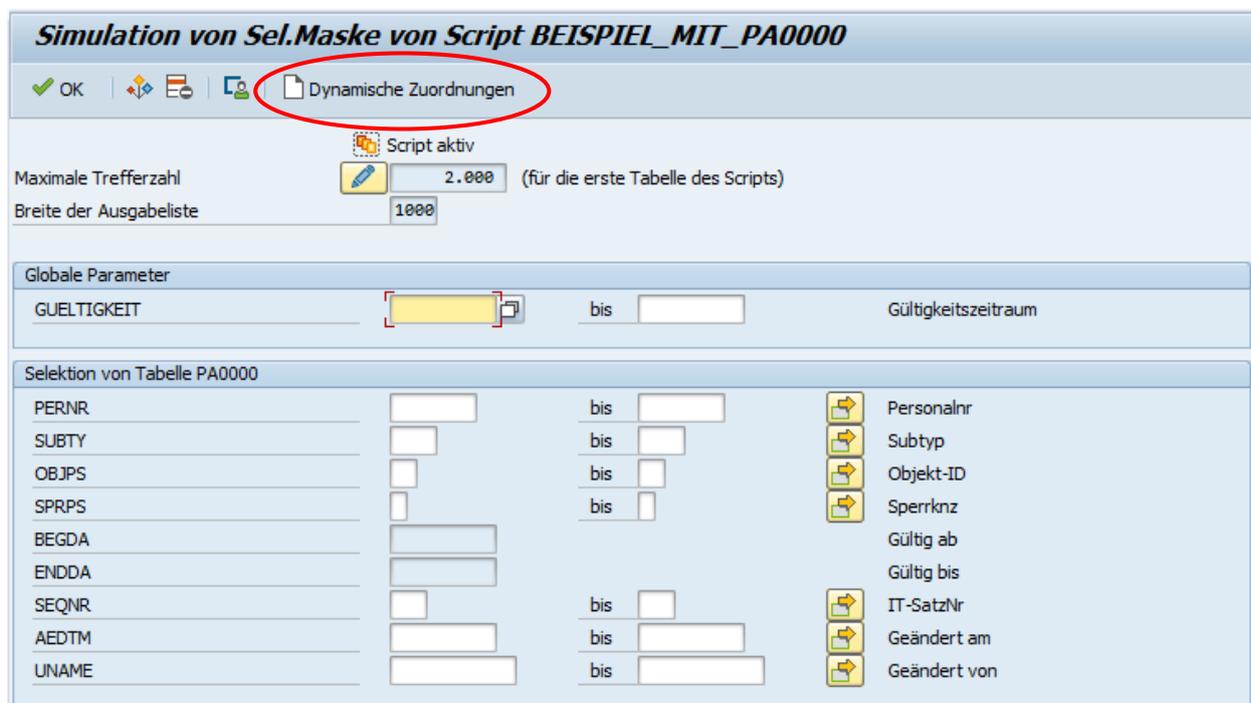
Der globale Parameter GUELTIGKEIT kann nach Belieben geändert werden.

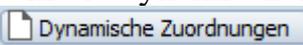
Unser Ziel ist es nun, dieses Script so anzupassen, dass nur der **aktuelle Monat** für die Selektion verwendet werden kann. Darüber hinaus sollte es **nicht** möglich sein, diese Einschränkung zu umgehen.

Um unser Ziel zu erreichen, rufen wir zunächst den Script-Katalog für unser Script auf. Dann rufen wir die Definition der speziellen Selektionsmaske () auf:

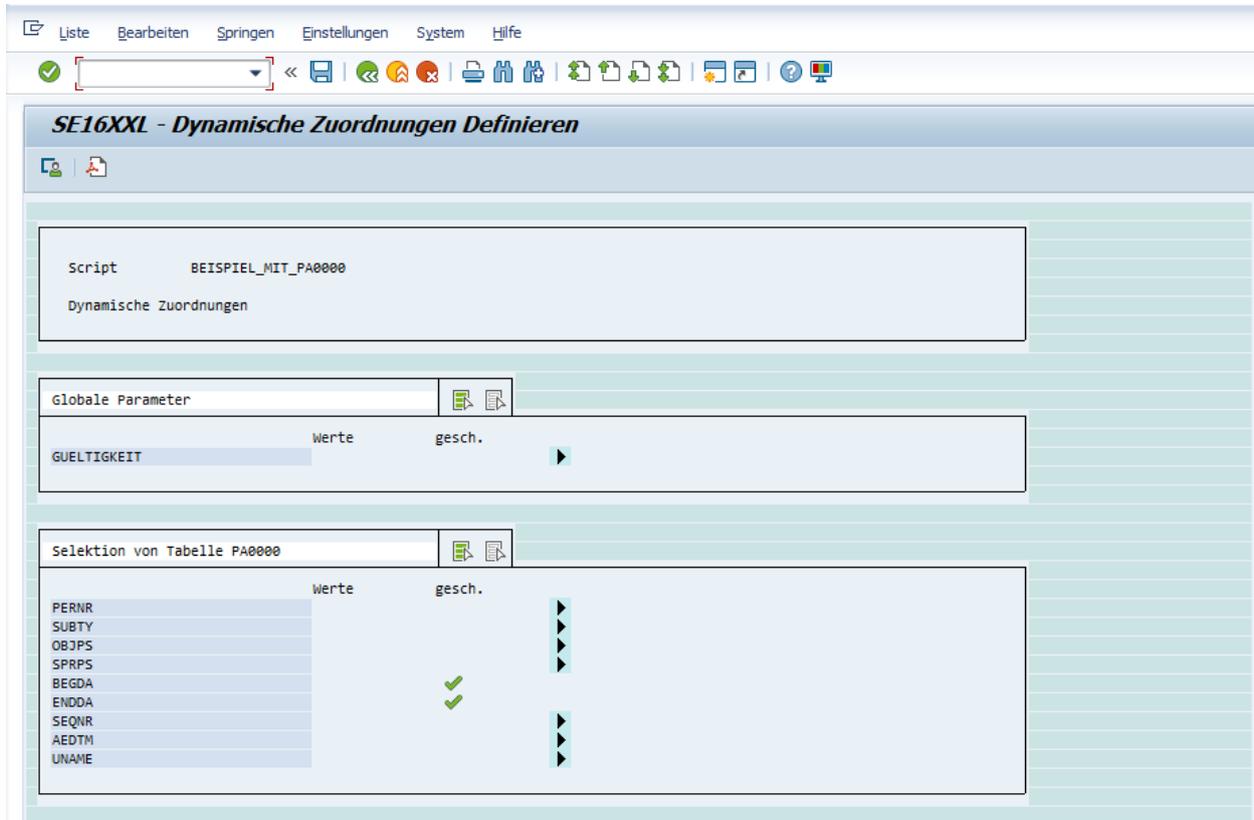


An dieser Stelle rufen wir die **Simulation der Selektionsmaske** auf, indem wir die Schaltfläche  betätigen:



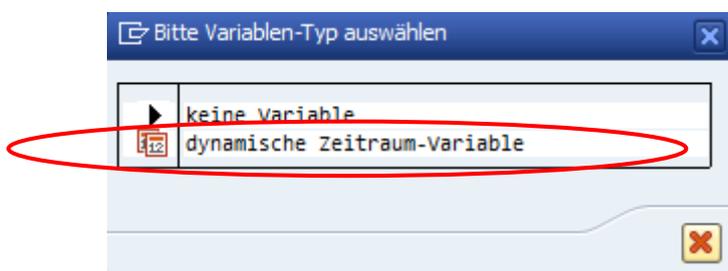
Um die dynamischen Zuordnungen zu definieren, betätigen wir die Schaltfläche  auf der Anwendungsleiste.

Als Ergebnis wird folgende Maske angezeigt:

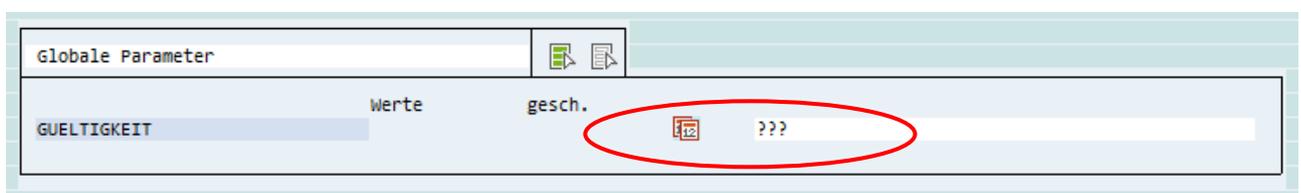


ANMERKUNG: Dieser Dialog ist dem Dialog, der zum Erstellen einer Script-Variante verwendet wird, sehr ähnlich, und in der Tat ist die interne Implementierung ähnlich. In diesem Fall jedoch werden nur dynamische Variablen berücksichtigt.

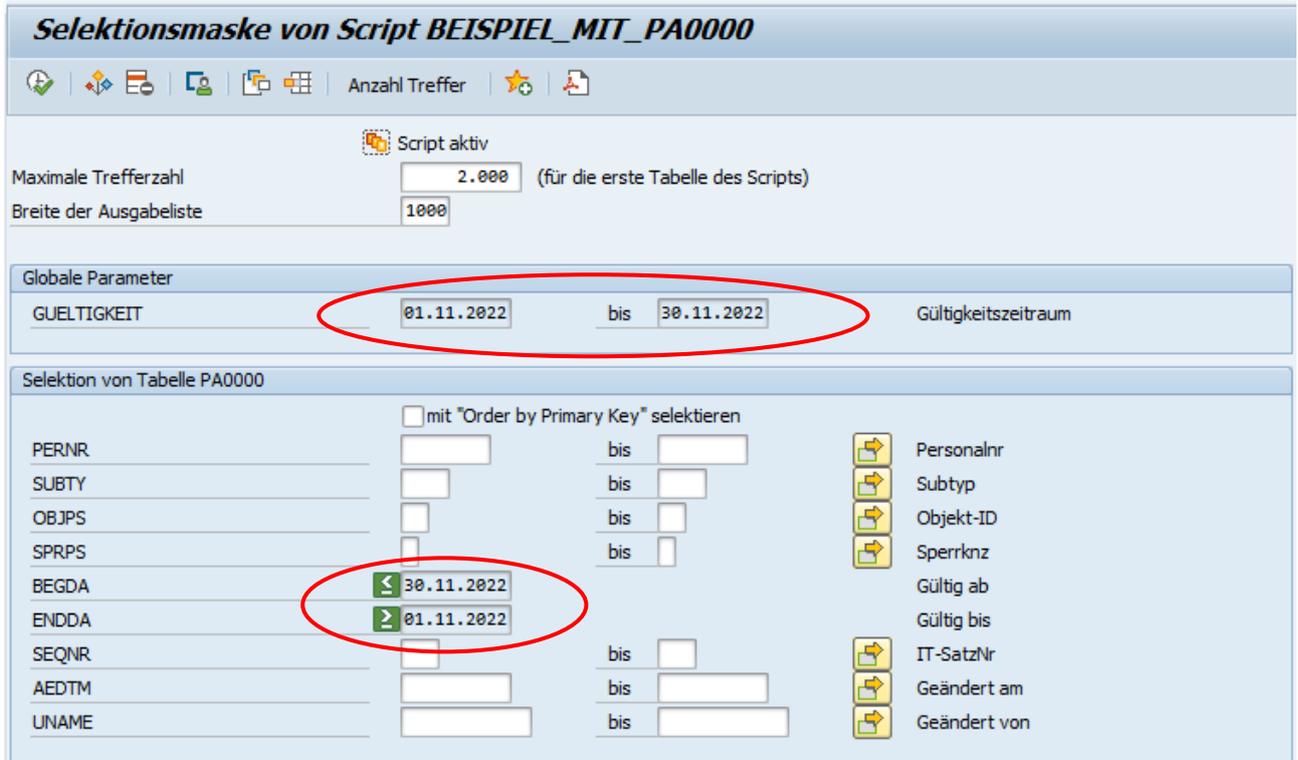
Wie im Falle einer Script-Variante klicken wir auf die ▶ Ikone rechts neben GUELTIGKEIT und wählen “dynamische Zeitraum-Variable”:



Das Layout ändert sich entsprechend:



Wenn wir nun das Script ausführen, wird die Selektionsmaske wie folgt aussehen:



Selektionsmaske von Script BEISPIEL_MIT_PA0000

Anzahl Treffer

Script aktiv

Maximale Trefferzahl: 2.000 (für die erste Tabelle des Scripts)

Breite der Ausgabeliste: 1000

Globale Parameter

GUELTIGKEIT: 01.11.2022 bis 30.11.2022 (Gültigkeitszeitraum)

Selektion von Tabelle PA0000

mit "Order by Primary Key" selektieren

PERNR	<input type="text"/>	bis	<input type="text"/>	 Personlnr
SUBTY	<input type="text"/>	bis	<input type="text"/>	 Subtyp
OBJPS	<input type="text"/>	bis	<input type="text"/>	 Objekt-ID
SPRPS	<input type="text"/>	bis	<input type="text"/>	 Sperrknz
BEGDA	<input type="text" value="30.11.2022"/>			 Gültig ab
ENDDA	<input type="text" value="01.11.2022"/>			 Gültig bis
SEQNR	<input type="text"/>	bis	<input type="text"/>	 IT-SatzNr
AEDTM	<input type="text"/>	bis	<input type="text"/>	 Geändert am
UNAME	<input type="text"/>	bis	<input type="text"/>	 Geändert von

Der globale Parameter GUELTIGKEIT ist nun auf die Grenzen des aktuellen Monats (November 2022) gesetzt und geschützt. Die Select-Options BEGDA und ENDDA wurden mit den notwendigen Werten entsprechend diesem globalen Parameter versehen.

WICHTIGER HINWEIS: Wenn ein mit dynamischen Zuordnungen versehenes Script in ein anderes SAP-System transportiert wird, entweder per Download/Upload oder über einen Transportauftrag, werden die dynamischen Zuordnungen ebenfalls transportiert. Wenn das Zielsystem jedoch eine **ältere Version von SE16XXL** enthält, **funktioniert das Script nicht**. Wenn Sie versuchen, es auszuführen, wird eine Fehlermeldung ausgegeben. Dies ist notwendig, **um unerwartete Ergebnisse zu vermeiden**.

[Zum Anfang](#)

Spalten Einfügen mit datumsbezogenen Feldern

Im SAP-System gibt es viele Datenbanktabellen, die **Zeitstempel** verschiedener Art enthalten, einige vom Typ NUMC, andere CHAR und wieder andere vom Typ DEC mit oder ohne Dezimalstellen. Auf neueren Systemen gibt es auch Zeitstempel vom Typ UTCLONG. Einige dieser Zeitstempel sind mit einer Konvertierungsexit versehen, andere nicht.

In den meisten Fällen ist es nicht einfach, das zugrunde liegende Datum zu erkennen, insbesondere wenn der Zeitstempel vom Typ DEC ist, der mit Punkten und Kommas als Trennzeichen angezeigt wird.

Theoretisch wäre es möglich, eine Formel zu definieren, die diese Informationen extrahiert und ein Feld vom Typ DATS mit dem gewünschten Wert zurückgibt. Diese Möglichkeit stünde allerdings nur ABAP-Kennern zur Verfügung und ist jedenfalls recht umständlich.

Daher wurde ab Version 3.6A von SE16XXL eine neue Menüfunktion implementiert:

Extras → Spalten einfügen → mit datumsbezogenen Feldern

Diese neue Operation extrahiert nicht nur das Datum aus einem Zeitstempel, sondern auch weitere Informationen wie das **Jahr**, den **Monat**, die **Woche**, die **Tageszeit** usw. Diese Werte können somit als separate Spalten in der Ergebnisliste eingefügt werden. Auf diese Weise ist es viel einfacher, auf einen Blick die gewünschten Informationen zu erhalten.

Folgende Tabelle zeigt, welche Feldtypen als **Originalfeld** (Quelle) für diese Operation Verwendung finden:

Art	Beschreibung
DATS	Datum (JJJJMMTT)
ACCP	Buchungsperiode (JJJJXX)
N14	Zeitstempel (NUMC 14)
C14	Zeitstempel (CHAR 14)
C16	Zeitstempel (CHAR 16)
D150	Zeitstempel (DEC 15,0)
D217	Zeitstempel (DEC 21,7)
UTCL	Zeitstempel (UTCLONG)
YYMM	Jahr + Monat (JJJJMM)
YYWW	Jahr + Woche (JJJJWW)
YYQ	Jahr + Quartal (JJJJQ)
INVD	Invertiertes Datum

Die datumsbezogenen Felder, die abgeleitet werden können, sind folgende:

Abgeleitetes Feld	Beschreibung
YYYY	Jahr (JJJJ)
MON	Monat (MM)
DD	Tag des Monats (TT)
QUART	Quartal (Q)
WDAY	Wochentag (W)
YYYYMM	Jahr + Monat (JJJJMM)
YYYYWW	Jahr + Woche (JJJJWW)
YYYYQ	Jahr + Quartal (JJJJQ)
DATS	Datum (JJJJMMTT)
TIMS	Uhrzeit (HHMMSS)

Es ist klar, dass nicht alle der oben aufgeführten Felder von allen Quellfeldern abgeleitet werden können. Beispielsweise kann die Uhrzeit (HHMMSS) nur aus den Zeitstempeln extrahiert werden und nicht aus einem Feld vom Typ DATS oder ACCP.

Ein kurzes Beispiel soll nun verdeutlichen, was besprochen wurde.

Beispiel

Tabelle **TSP01** (Spool-Aufträge) enthält drei Zeitstempel vom Typ C16:

- **RQCRETIME** - Erzeugungszeit eines Spool-Auftrags
- **RQDELTIME** - Löschezit eines Spool-Auftrags
- **RQMODTIME** - Letzte Änderungszeit eines Spool-Auftrags

Anhand des ersten (**RQCRETIME**) werden wir zeigen, wie sich die neue Operation verhält.

Zunächst selektieren wir einige Datensätze dieser Tabelle:

SE16XXL - Tabelle TSP01 - 663 Einträge selektiert

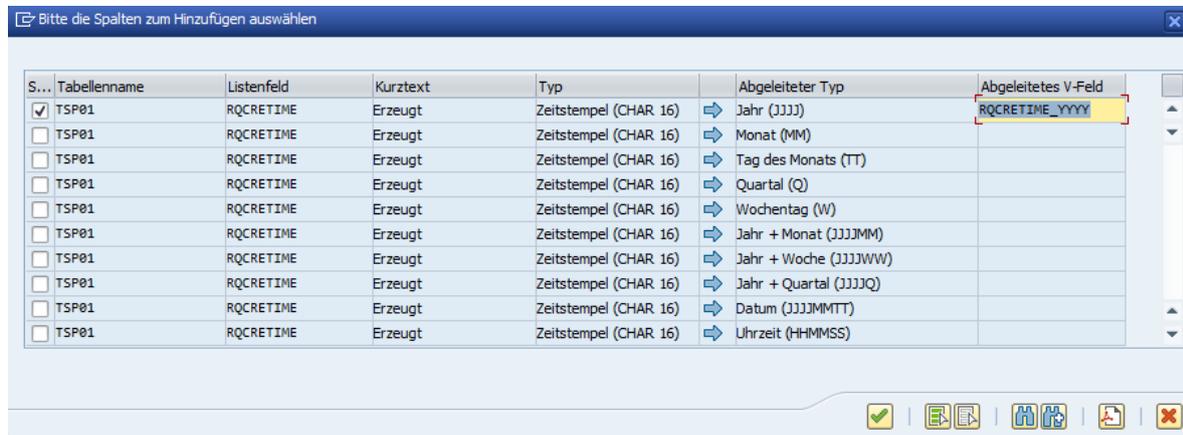
Tabelle TSP01 - Spool Requests

	RQIDENT	RQCLIENT	RQ0NAME	RQ1NAME	RQ2NAME	RQOWNER	RQCMODE	RQFINAL	RQCRETIME	RQDELTIME	RQREARULE	RQDELRULE	RQAF
<input type="checkbox"/>	97.521	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010103320200	2022010903320200	0	0	
<input type="checkbox"/>	97.522	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010115330300	2022010915330300	0	0	
<input type="checkbox"/>	97.523	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010203340300	2022011003340300	0	0	
<input type="checkbox"/>	97.524	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010215350300	2022011015350300	0	0	
<input type="checkbox"/>	97.525	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010303360500	2022011103360500	0	0	
<input type="checkbox"/>	97.526	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010315370300	2022011115370300	0	0	
<input type="checkbox"/>	97.527	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010403380600	2022011203380600	0	0	
<input type="checkbox"/>	97.528	800	LIST15	LOCL	RSLDAGDS_BEL	BELZ	1	.	2022010415390400	2022011215390400	0	0	

Nun markieren wir die Spalte RQCRETIME und wählen dann die Menüfunktion

Extras → Spalten einfügen → mit datumsbezogenen Feldern.

Das folgende Dialogfenster erscheint:



Alle möglichen abgeleiteten Werte werden angezeigt, aber nur das Jahr (JJJJ) wird ausgewählt. Man kann dann entscheiden, was man benötigt. In diesem Beispiel wählen wir alle Felder aus, um ihre Ergebnisse anzuzeigen. Die neu angelegten Spalten werden direkt nach dem Originalfeld eingefügt:

FINAL	RQCRETIME	V~RQCRETIME_YYYY	V~RQCRETIME_MON	V~RQCRETIME_DD	V~RQCRETIME_QUART	V~RQCRETIME_WDAY	V~RQCRETIME_YYYYMM	V~RQCRETIME_YYYYWW	V~RQCRETIME_YYYYQ	V~RQCRETIME_DATS	V~RQCRETIME_TIMS
	2022010103320200	2022	01	01	1	6	202201	202152	20221		
	2022010115330300	2022	01	01	1	6	202201	202152	20221		
	2022010203340300	2022	01	02	1	7	202201	202152	20221		
	2022010215350300	2022	01	02	1	7	202201	202152	20221		
	2022010303360500	2022	01	03	1	1	202201	202201	20221		
	2022010315370300	2022	01	03	1	1	202201	202201	20221		
	2022010403380600	2022	01	04	1	2	202201	202201	20221		

Die einzelnen Werte können am besten in der Detailansicht inspiziert werden:

RQFINAL	','	Spool-Auftrag ist abgeschlossen
RQCRETIME	'2022010415390400'	Erzeugungszeit eines Spool-Auftrags
RQDELTIME	'2022011215390400'	Löschzeit eines Spool-Auftrags
RQREARULE	' 0 '	TemSe: Nummer einer Lese-Schutzregel

...

V~RQCRETIME_YYYY	'2022'	Erzeugungszeit eines Spool-Auftrags (Jahr)
V~RQCRETIME_MON	'01'	Erzeugungszeit eines Spool-Auftrags (Mon)
V~RQCRETIME_DD	'04'	Erzeugungszeit eines Spool-Auftrags (Tag)
V~RQCRETIME_QUART	'1'	Erzeugungszeit eines Spool-Auftrags (Qu)
V~RQCRETIME_WDAY	'2'	Erzeugungszeit eines Spool-Auftrags (WTag)
V~RQCRETIME_YYYYMM	'202201'	Erzeugungszeit eines Spool-Auftrags (Monat)
V~RQCRETIME_YYYYWW	'202201'	Erzeugungszeit eines Spool-Auftrags (Woche)
V~RQCRETIME_YYYYQ	'20221'	Erzeugungszeit eines Spool-Auftrags (Quartal)
V~RQCRETIME_DATS	'04.01.2022'	Erzeugungszeit eines Spool-Auftrags (Datum)
V~RQCRETIME_TIMS	'15:39:04'	Erzeugungszeit eines Spool-Auftrags (Uhrzeit)

[Zum Anfang](#)

Summieren auf der DB mithilfe eines Jahres-V-Feldes

Bisher war es möglich, ein Script mit der Option “**DB-Summen verwenden wenn möglich**” auszuführen. Unter den verschiedenen Einschränkungen gab es eine, die besagt, dass die Comparing-Kriterien der Summierungs-Operationen **keine Teilfelder enthalten sollten**.

Ab Version 3.6A von SE16XXL gilt diese Einschränkung weiterhin. Es gibt jetzt jedoch eine **Ausnahme** von dieser Regel: Wenn das Teilfeld ein **Jahres-V-Feld** ist, abgeleitet von einem **DATS**-Originalfeld, dann kann die Summierung auf der Datenbank stattfinden, obwohl die Datenbank dieses V-Feld gar nicht kennt. Mit anderen Worten, wenn die Datenbanktabelle ein Feld vom Typ DATS (Datum) enthält, kann die Summierung der zugehörigen Jahreswerten durchgeführt werden.

Wie wird das erreicht? Das Programm beginnt mit der Selektion **aller beteiligten Datumswerte**. Aus diesen Informationen wird eine Reihe von **Jahreswerten** extrahiert. **Pro Jahr** wird dann eine Summierungs-Operation auf der Datenbank durchgeführt, die auf die Datensätze beschränkt ist, deren ursprüngliches Datumfeld zwischen Anfang und Ende dieses Jahres liegt.

Ist die Anzahl der beteiligten Datensätze sehr hoch, sollte die Durchführung der DB-Summierung auf diese Weise dennoch zu einer deutlichen Verbesserung der Performance führen.

ANMERKUNG: Die Operation *Extras* → *Spalten einfügen* → *mit datums-bezogenen Feldern*, die zum Ableiten des Jahreswerts verwendet wird, darf diesen einen Wert nur aus einem Quellfeld vom Typ DATS ableiten. Falls das Quellfeld von anderer Art ist oder wenn zusätzliche V-Felder abgeleitet werden, findet keine Summierung auf der Datenbank statt.

Beispiele

Mit dieser neuen Funktion könnte eine Statistik erstellt werden, wie viele Änderungsbelege (Tabelle **CDHDR** - Änderungsbelegkopf) pro Jahr erzeugt wurden, wobei man das Datumfeld **UDATE** als Originalfeld verwenden würde.

Ein weiteres Beispiel wäre, eine Liste der Kundenauftrags-Nettowerte pro Jahr zu erstellen, indem man das Feld **NETWR** der Tabelle **VBAK** (Verkaufsbeleg: Kopfdaten) mit dem Datumfeld **AUDAT** als Originalfeld aufsummieren würde.

[Zum Anfang](#)

Einfärben einzelner Spalten der Ergebnisliste

Bisher hatten nur die **Schlüsselfelder** und die **Sortierkriterien** der Ergebnisliste eine spezielle Farbe, um sie von den übrigen zu unterscheiden.

Ab Version 3.6A von SE16XXL ist es möglich, einzelnen Spalten der Ergebnisliste eine **individuelle Farbe** zuzuweisen. Auf diese Weise können “wichtige” Spalten hervorgehoben werden, um Aufmerksamkeit zu erregen.

Ein kurzes Beispiel soll verdeutlichen, wie diese Funktion genutzt werden kann.

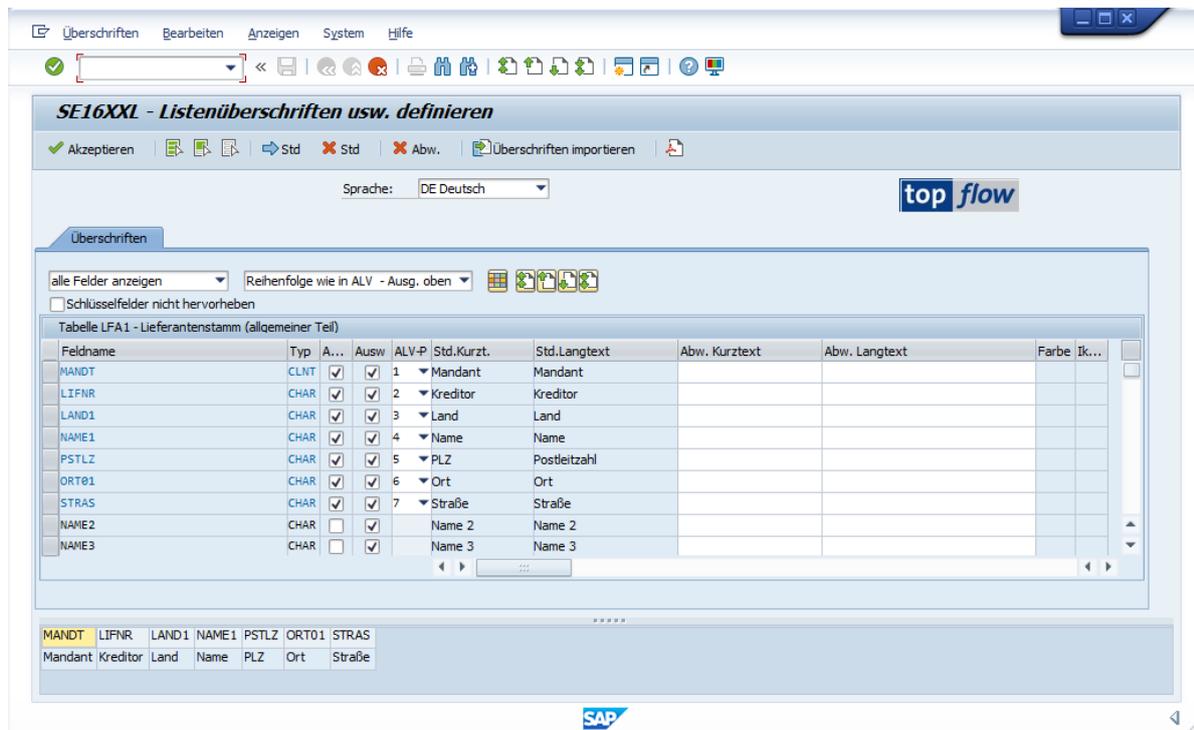
Beispiel

Angenommen, Sie haben eine Liste von Lieferanten (Tabelle **LFA1** – Lieferantenstamm (allgemeiner Teil)). In der Ergebnisliste sollten die Felder **PSTLZ** (Postleitzahl) und **ORT01** (Ort) hervorgehoben werden.

Wir beginnen mit dem Aufruf der Menüfunktion

Extras → Überschriften usw. definieren

Die Definitionsmaske erscheint:



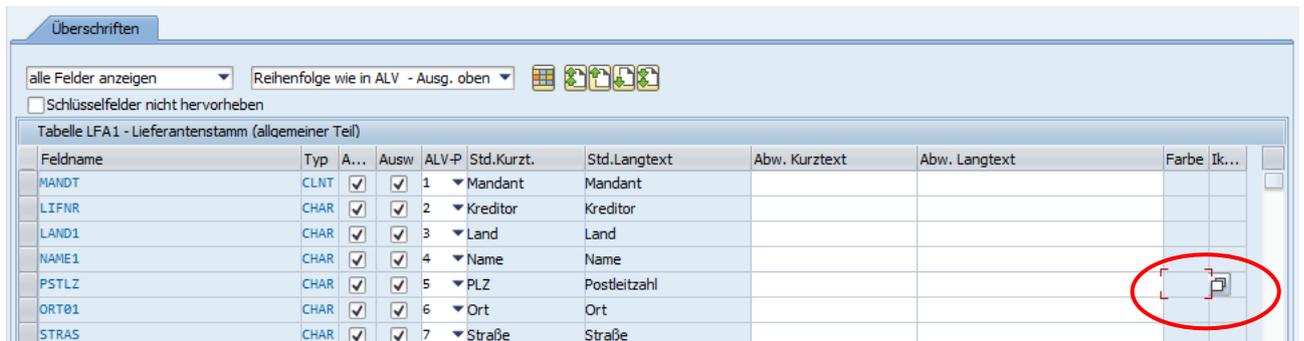
The screenshot shows the SAP SE16XXL 'Listenüberschriften usw. definieren' dialog box. The table below the dialog shows the following data:

Feldname	Typ	A...	Ausw	ALV-P	Std.Kurztext	Std.Langtext	Abw. Kurztext	Abw. Langtext	Farbe	Ik...
MANDT	CLNT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	▼ Mandant	Mandant				
LIFNR	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2	▼ Kreditor	Kreditor				
LAND1	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	3	▼ Land	Land				
NAME1	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	4	▼ Name	Name				
PSTLZ	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	5	▼ PLZ	Postleitzahl				
ORT01	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	6	▼ Ort	Ort				
STRAS	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	7	▼ Straße	Straße				
NAME2	CHAR	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Name 2	Name 2				
NAME3	CHAR	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Name 3	Name 3				

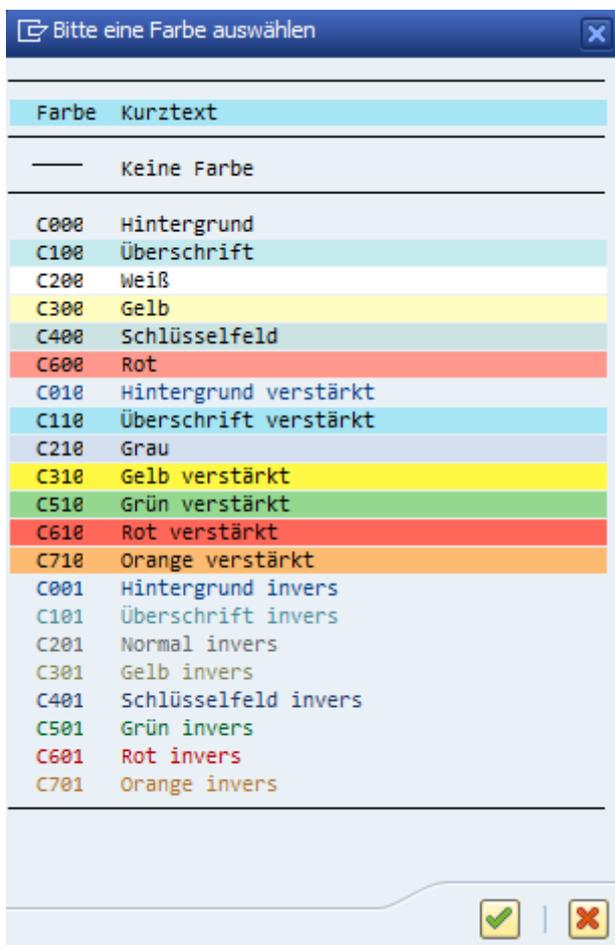
The table below the dialog shows the following data:

MANDT	LIFNR	LAND1	NAME1	PSTLZ	ORT01	STRAS
Mandant	Kreditor	Land	Name	PLZ	Ort	Straße

Um einer bestimmten Spalte eine individuelle Farbe zuzuordnen, müssen Sie die F4-Hilfe für die Spalte “Farbe” rechts verwenden:



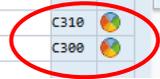
Daraufhin zeigt Ihnen das Programm eine Liste möglicher Farben an:



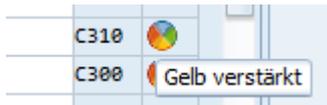
ANMERKUNG: Die Farben, die für Schlüsselfelder und Sortierkriterien verwendet werden, sind reserviert und stehen daher nicht für andere Zwecke zur Verfügung.

Wir wählen unsere Farben aus und die Definitionsmaske ändert sich entsprechend:

LAND1	CHAR	✓	✓	3	Land	Land		
NAME1	CHAR	✓	✓	4	Name	Name		
PSTLZ	CHAR	✓	✓	5	PLZ	Postleitzahl		
ORT01	CHAR	✓	✓	6	Ort	Ort		
STRAS	CHAR	✓	✓	7	Straße	Straße		



Wenn der Cursor auf der Ikone positioniert wird, zeigt die Quickinfo die ausgewählte Farbe an:



Nach der Rückkehr zur Ergebnisliste können wir feststellen, wie sich das Layout geändert hat:

SE16XXL - Tabelle LFA1 - 1766 Einträge selektiert

Tabelle LFA1 - Lieferantenstamm (allgemeiner Teil)

MANDT	LIFNR	LAND1	NAME1	PSTLZ	ORT01	STRAS
800	000000001	DE	Forks Manufacturing GmbH	27563	Hamburg	Heinerstraße 98-100
800	000000002	US	Electronic Components Distributor	94404	FOSTER CITY	Tower Lane 1082
800	000000005	DE	Safety Clean Inc.	73693	New York	Desoto Road
800	000000008	MX	Jose Fernandez	11111	Mexico City	Via Rioja 1
800	000000010	FR	Dupont de la Rivière	75000	Paris	rue des fleuves
800	000000015	DE	Tiedemeier Entsorgung GmbH	11576	Berlin	Westendstraße
800	000000025	DK	Metropol	1000	Copenhagen	Kronprinsensgade 11
800	000000050	DE	Central Logistics Inc.	73701	New York	Greenhill Road

Eine weitere Möglichkeit, ein ähnliches Ergebnis zu erzielen, wäre, allen Feldern außer denen, die hervorgehoben werden sollen, eine unauffällige Farbe zuzuweisen.

In diesem Fall markieren wir alle Felder () auf der Definitionsmaske für Überschriften:

Überschriften

alle Felder anzeigen | Reihenfolge wie in ALV - Ausg. oben

Schlüsselfelder nicht hervorheben

Tabelle LFA1 - Lieferantenstamm (allgemeiner Teil)

Feldname	Typ	A...	Ausw	ALV-P	Std.Kurztext	Std.Langtext	Abw. Kurztext	Abw. Langtext	Farbe	Ik...
MANDT	CLNT	✓	✓	1	Mandant	Mandant				
LIFNR	CHAR	✓	✓	2	Kreditor	Kreditor				
LAND1	CHAR	✓	✓	3	Land	Land				
NAME1	CHAR	✓	✓	4	Name	Name				
PSTLZ	CHAR	✓	✓	5	PLZ	Postleitzahl			C310	
ORT01	CHAR	✓	✓	6	Ort	Ort			C110	
STRAS	CHAR	✓	✓	7	Straße	Straße				
NAME2	CHAR	✓	✓		Name 2	Name 2				
NAME3	CHAR	✓	✓		Name 3	Name 3				

Wenn wir nun einer der Spalten eine Farbe zuweisen, erhalten auch alle anderen Spalten diese Farbe.

Da wir eine unauffällige Farbe benötigen, wählen wir Grau. Das Ergebnis:

Überschriften

alle Felder anzeigen Reihenfolge wie in ALV - Ausg. oben

Schlüsselfelder nicht hervorheben

Tabelle LFA1 - Lieferantenstamm (allgemeiner Teil)

Feldname	Typ	A...	Ausw	ALV-P	Std.Kurztext	Std.Langtext	Abw. Kurztext	Abw. Langtext	Farbe	Ik...
MANDT	CLNT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	Mandant	Mandant			C210	
LIFNR	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2	Kreditor	Kreditor			C210	
LAND1	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	3	Land	Land			C210	
NAME1	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	4	Name	Name			C210	
PSTLZ	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	5	PLZ	Postleitzahl			C210	
ORT01	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	6	Ort	Ort			C210	
STRAS	CHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	7	Straße	Straße			C210	
NAME2	CHAR	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Name 2	Name 2			C210	
NAME3	CHAR	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Name 3	Name 3			C210	

Nun können wir unseren beiden Feldern PSTLZ und ORT01 die gewünschten Hervorhebungsfarben zuweisen – zusätzlich setzen wir auch die Farbe des Schlüsselfeldes LIFNR zurück.

Das Resultat wäre dann wie im folgenden Bild:

SE16XXL - Tabelle LFA1 - 1766 Einträge selektiert

Ben. Inner Outer Full

Tabelle LFA1 - Lieferantenstamm (allgemeiner Teil)

MANDT	LIFNR	LAND1	NAME1	PSTLZ	ORT01	STRAS
800	000000001	DE	Forks Manufacturing GmbH	27563	Hamburg	Heinerstraße 98-100
800	000000002	US	Electronic Components Distributor	94404	FOSTER CITY	Tower Lane 1082
800	000000005	DE	Safety Clean Inc.	73693	New York	Desoto Road
800	000000008	MX	Jose Fernandez	11111	Mexico City	Via Rioja 1
800	000000010	FR	Dupont de la Rivière	75000	Paris	rue des fleuves
800	000000015	DE	Tiedemeier Entsorgung GmbH	11576	Berlin	Westendstraße
800	000000025	DK	Metropol	1000	Copenhagen	Kronprinsensgade 11
800	000000050	DE	Central Logistics Inc.	73701	New York	Greenhill Road
800	000000075	DE	Meier Logistics GmbH	11588	Berlin	Industriestrasse 19
800	000000100	DE	C.E.B. BERLIN	12001	Berlin	Kolping Str. 15
800	000000111	DE	KBB Schwarze Pumpe	67210	Frankenthal/Pfalz	Fasanenstr. 8
800	000000200	US	SMP	30309	ATLANTA	1730 Industrial Plaza
800	000000222	US	Express Vendor Inc	60614	CHICAGO	2550 North Racine Ave
800	000000300	US	AluCast	08844	HILLSBOROUGH	900 Edison Ave. Suite 300
800	000000424	US	Sedona Suppliers	86335	RIMROCK	3456 Red View Way
800	000000500	DE	IDES AG New GL - (InterCompany Acco		Berlin	1 Unter den Linden
800	000000603	CA	Intercompany Resources US			
800	000000604	CA	Intercompany Resources to Belgium			

ANMERKUNG: Wenn die Ergebnisliste Zeilen mit einzelnen Farben enthält (bewirkt durch ein COLOR-Formelfeld oder ein COLOR-V-Feld), funktioniert die Färbung einzelner Spalten im ALV-Modus nicht, da in ALV die Zeilenfarbe die Spaltenfarben übersteuert.

[Zum Anfang](#)

Gültigkeits-Prüfung – Aktuelles Datum als Ref.-Datum

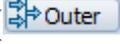
Bisher konnte bei der Verwendung der Gültigkeits-Prüfung während einer Join-Operation das **Referenz-Datum** nur entweder als **Listenfeld** oder als **globaler Parameter** angegeben werden. Es stellt sich heraus, dass in den meisten Fällen das aktuelle Datum die beste Wahl wäre. Dies konnte allerdings nur dadurch erreicht werden, indem ein globaler Parameter definiert und auf der Selektionsmaske platziert wurde, um das Datum entsprechend ändern zu können. In der Tat eine relativ aufwendige Prozedur.

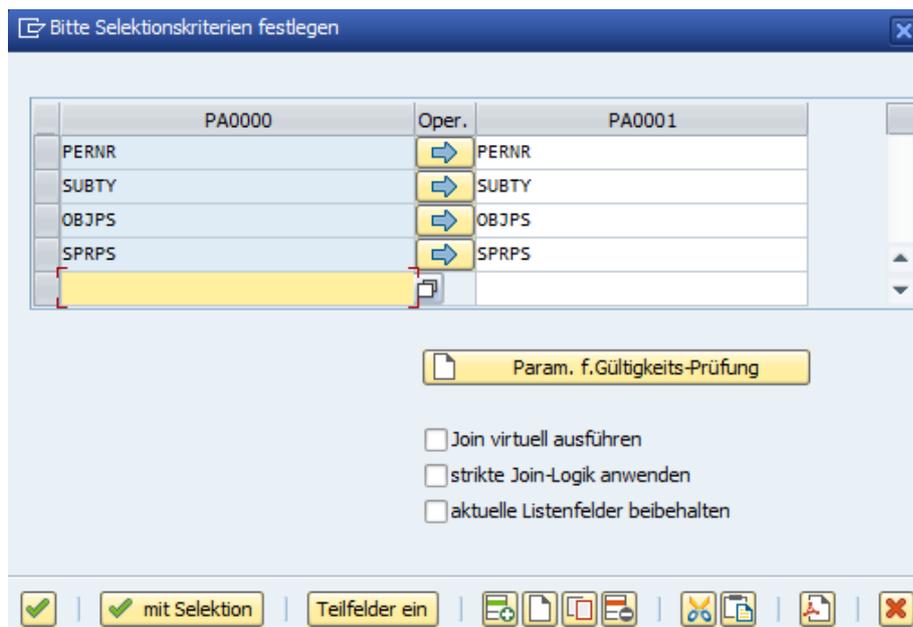
Ab Version 3.6A von SE16XXL ist es möglich, den speziellen Parameter **“CURRENT_DATE”** zu verwenden, der **immer verfügbar** ist und immer das aktuelle Datum (**SY-DATUM**) enthält. Dadurch dürfte es viel einfacher werden, die Funktionalität der Gültigkeits-Prüfung zu verwenden.

Wir werden diese neue Funktionalität anhand eines Beispiels veranschaulichen.

Beispiel

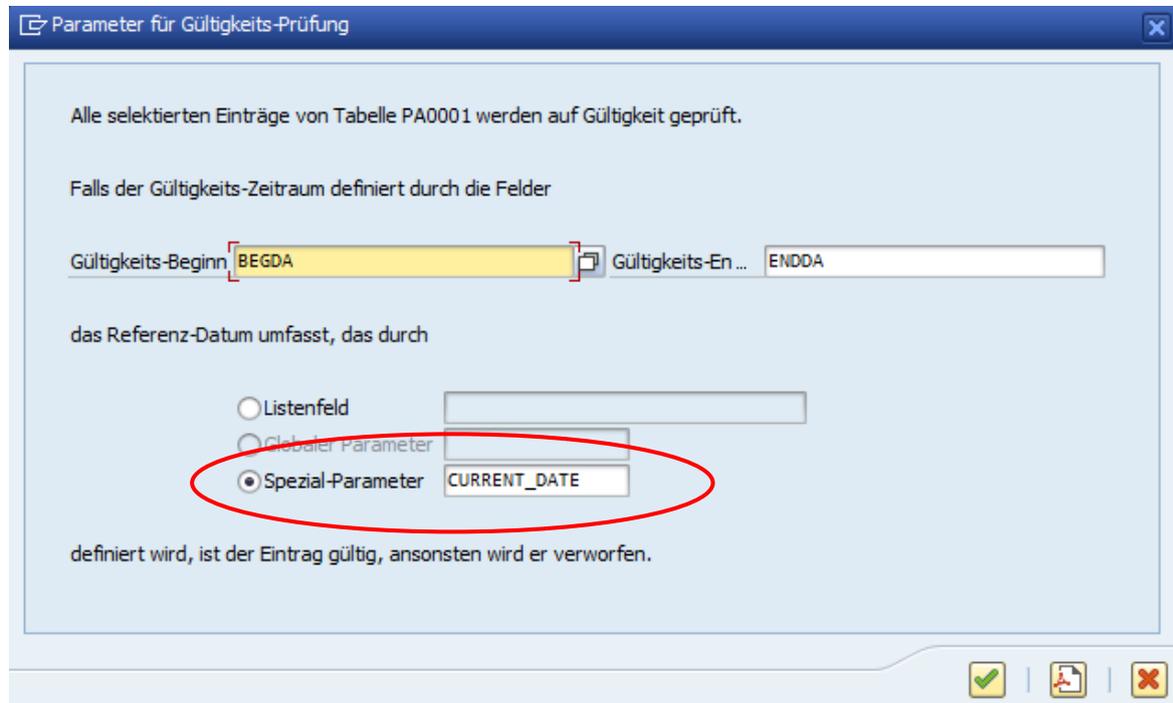
Angenommen, wir haben eine Liste der Datensätze der HR-Tabelle **PA0000** (Massnahmen). Wir möchten einen Join mit der HR-Tabelle **PA0001** (Org. Zuordnung) durchführen. Allerdings sollten nur die Datensätze selektiert werden, die zum aktuellen Datum gültig sind.

Ausgehend von der Ergebnisliste mit PA0000-Datensätzen führen wir einen Outer-Join mit PA0001 durch ( Outer). Die Join-Kriterien könnten wie folgt festgelegt werden:



An dieser Stelle konfigurieren wir die Parameter der Gültigkeits-Prüfung durch Betätigung der Schaltfläche .

Das entsprechende Dialogfenster erscheint:



Wie man sehen kann, ist der spezielle Parameter `CURRENT_DATE` bereits markiert. Wir müssen nur mit  bestätigen und der Join wird durchgeführt.

ANMERKUNG: Die Gültigkeits-Prüfung kann nur genutzt werden, wenn ein Join oder ein “Selektieren für alle Einträge” durchgeführt wird. Für die erste Selektion steht diese Funktion nicht zur Verfügung. Jedoch mit einem kleinen Trick kann dieses Problem wie folgt gelöst werden:

- 1) Selektieren Sie zunächst die gewünschten PA0000-Datensätze, ohne weder BEGDA noch ENDDA als Selektionskriterien anzugeben.
- 2) Führen Sie nun eine Operation “Selektieren für alle Einträge” durch, um PA0000 erneut zu selektieren, diesmal jedoch unter Verwendung der Gültigkeits-Prüfung mit `CURRENT_DATE`.

Auf diese Weise sind die PA0000-Datensätze in der Ergebnisliste diejenigen, die wir gesucht haben.

[Zum Anfang](#)

E-Mails mit Server-Datei-Anhang – mehr Flexibilität

Bisher standen bei der Einplanung eines Scripts im Hintergrund, das eine Server-Datei erzeugt, drei Alternativen zur Angabe der E-Mail-Empfänger zur Verfügung:

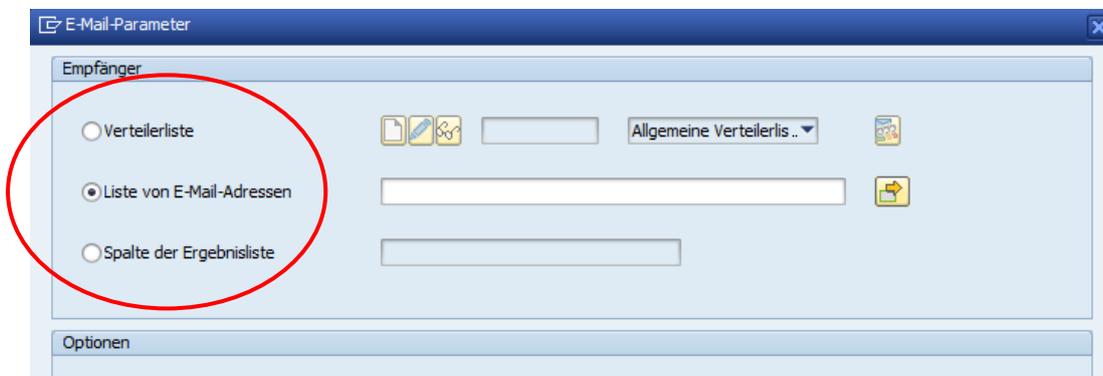
- Als Verteilerliste
- Als Liste von E-Mail-Adressen
- Als Spalte der Ergebnisliste

Es konnte **nur eine** dieser drei Optionen aktiviert werden, wodurch die anderen ausgeschlossen wurden.

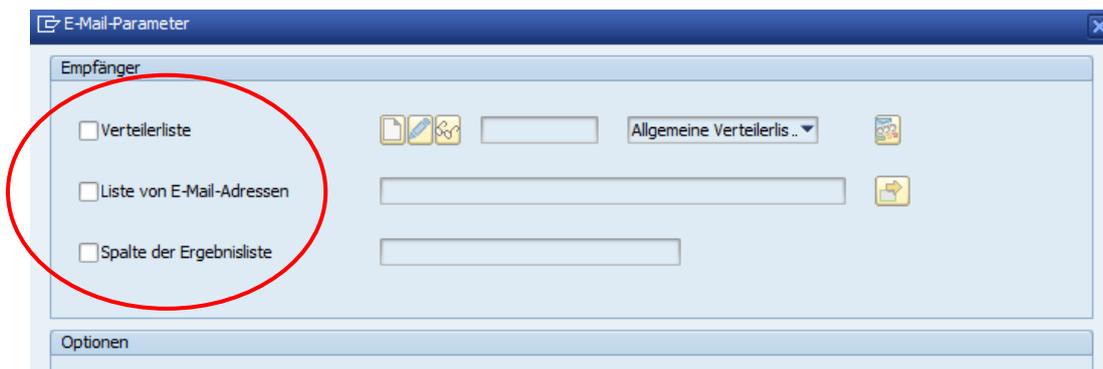
Ab Version 3.6A von SE16XXL wurde diese Unannehmlichkeit beseitigt. Es ist nun möglich, **alle drei Optionen gleichzeitig zu verwenden**. Mit anderen Worten, es kann ein Script eingeplant werden, das E-Mails an alle Empfänger sendet, die in einer Verteilerliste enthalten sind, zusammen mit den Empfängern, die explizit angegeben wurden und zusätzlich an diejenigen, die in einer Spalte der Ergebnisliste selbst enthalten sind.

Das Dialogfenster zur Definition der E-Mail-Empfänger wurde entsprechend angepasst.

Bisher waren die Optionen wie folgt:

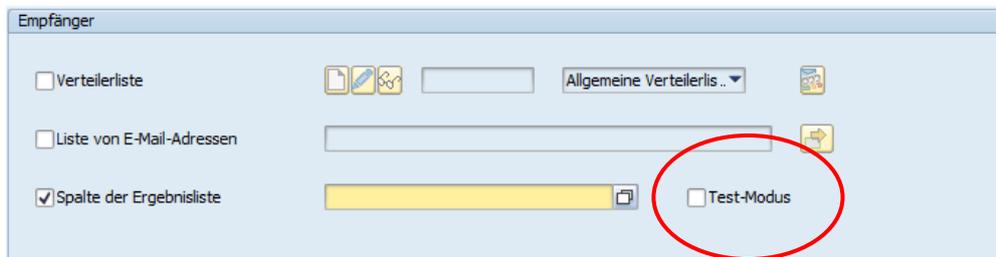


Jetzt wurden die **Auswahlknöpfe** durch **Ankreuzfeldern** ersetzt:

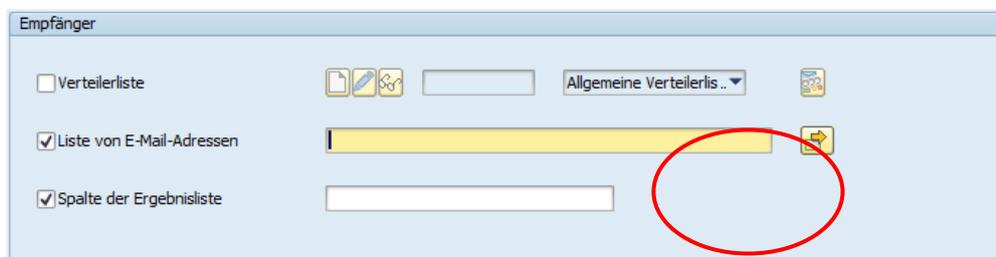


ANMERKUNG #1

In Kombination mit der Option “Spalte der Ergebnisliste” gibt es eine spezielle Option “Test-Modus”:



Diese spezielle Option ist nur verfügbar, wenn keine weiteren Optionen aktiviert werden, wie aus dem folgenden Bild ersichtlich ist:



ANMERKUNG #2

In Kombination mit der Option “Spalte der Ergebnisliste” gibt es noch eine weitere Option, “**Nur relevante Sätze an jeden Empfänger senden**”. Diese Option ist nur für Empfänger relevant, die aus der Spalte der Ergebnisliste entnommen wurden. Alle explizit oder über eine Verteilerliste angegebenen Empfänger erhalten eine **vollständige** Server-Datei.

ANMERKUNG #3

If more than one option is selected, the e-mails are sent to the various recipients in the following order:

Wenn mehr als eine Option aktiviert wird, werden die E-Mails in der folgenden Reihenfolge an die verschiedenen Empfänger gesendet:

- Empfänger aus der Spalte der Ergebnisliste
- Empfänger aus der Verteilerliste
- Empfänger explizit angegeben

Kommt dieselbe E-Mail-Adresse mehr als einmal vor, wird nur das erste Vorkommen verwendet.

[Zum Anfang](#)

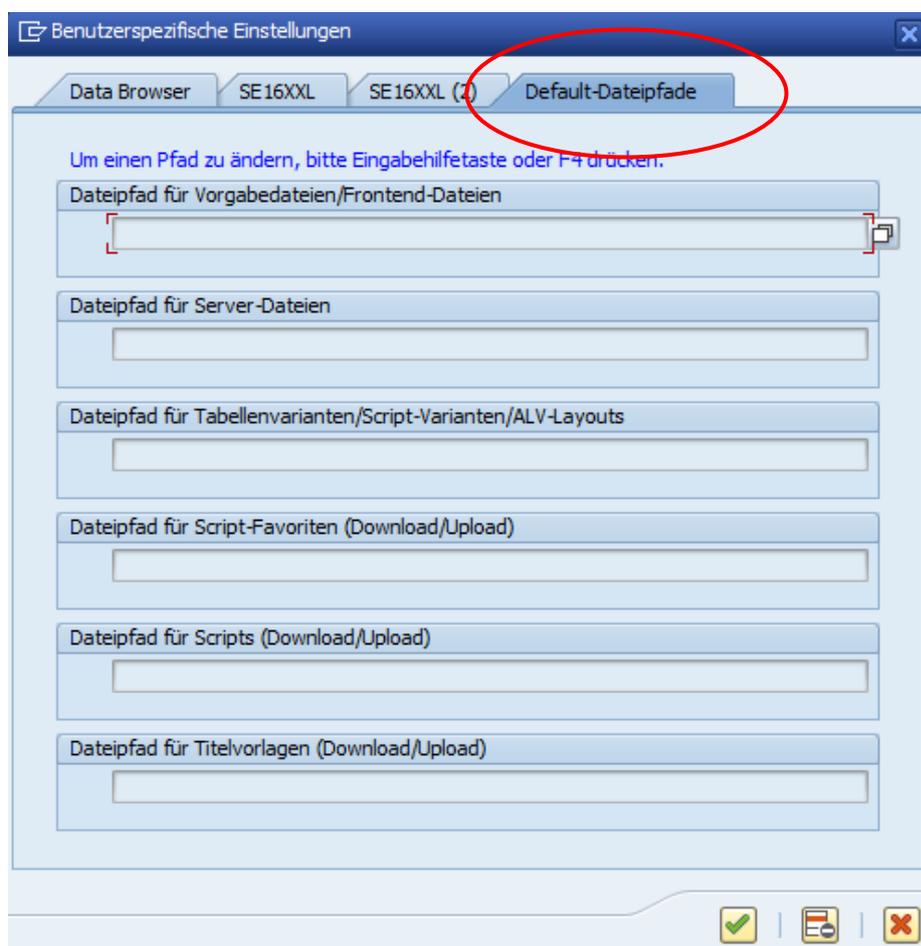
Benutzer-Einstellungen – Default-Dateipfade

Bisher, wenn eine Operation mit einer Datei durchgeführt wurde, wie z.B. das Herunterladen der Ergebnisliste in eine Frontend-Datei oder das Speichern in einer Server-Datei oder das Herunterladen/Hochladen von Scripts im Script-Katalog usw., schlug SE16XXL immer den gleichen Standardpfad **C:\temp** für Frontend-Dateien und den **ersten zulässigen Dateipfad** für Server-Dateien vor.

Dieses Verhalten seitens SE16XXL kann ziemlich frustrierend sein, wenn der vorgeschlagene Dateipfad nicht derjenige ist, der normalerweise verwendet wird.

Um diese Unannehmlichkeiten zu beseitigen, ist es ab Version 3.6A von SE16XXL möglich, in den Benutzer-Einstellungen eine Reihe von **Default-Pfaden** für die am häufigsten auftretenden Situationen festzulegen. Darüber hinaus kann die Systemadministration für einige oder alle dieser Default-Pfade **Initialwerte** festlegen, die dann angewendet werden, wenn keine individuelle Einstellung angegeben wurde.

Zu diesem Zweck wurde im Dialogfenster der benutzerspezifischen Einstellungen eine neue Registerkarte hinzugefügt:



Um eine falsche Schreibweise der Pfadnamen zu verhindern, sind die Felder **geschützt**. Sie müssen die **F4-Hilfe** verwenden, um einen Pfad anzugeben.

Wenn links neben einem Pfadnamen eine  Ikone zu sehen ist, bedeutet dies, dass der Pfad von der Systemadministration als **initialer Pfadname** vorgeschlagen wurde. Dieser wird verwendet, wenn kein individueller Name festgelegt wurde:



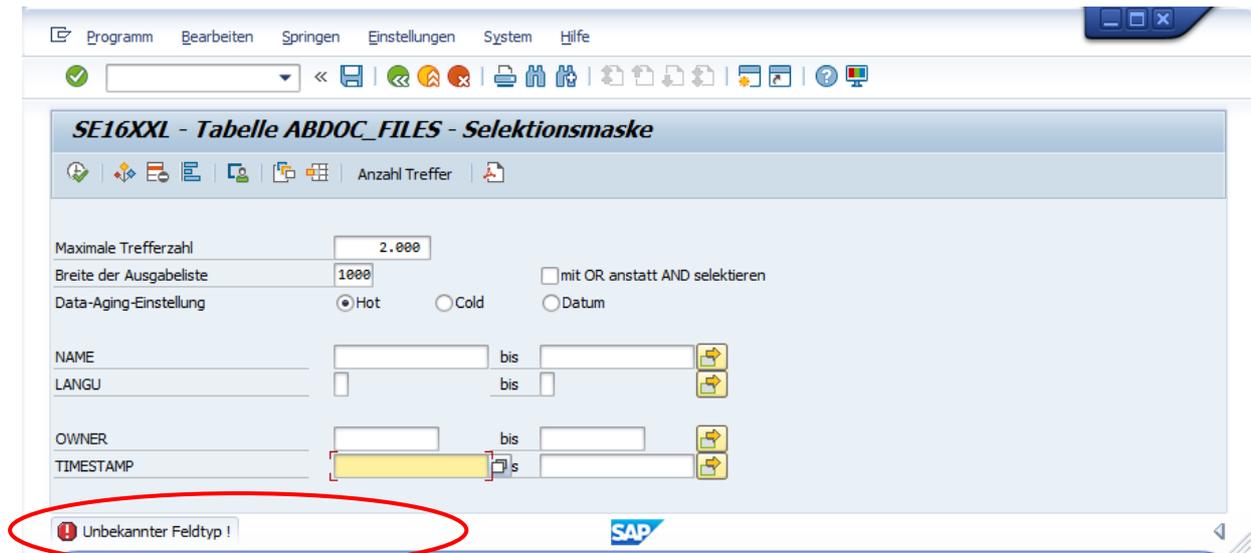
Sobald ein individueller Pfadname gewählt wird, verschwindet die Ikone mit der Flagge. Sie erscheint wieder, wenn die Einstellung zurückgesetzt wird ().

ANMERKUNG: Die angegebenen Default-Pfadnamen werden verwendet, wenn eine Datei-Operation zum ersten Mal während einer **Logon-Sitzung** durchgeführt wird. Wird zu einem anderen Pfad gewechselt, wird dieser neu angegebene Pfad für die nachfolgenden Datei-Operationen derselben Art verwendet. Der Default-Pfad wird erst zu Beginn einer neuen Logon-Sitzung erneut vorgeschlagen.

[Zum Anfang](#)

Elementartyp UTCLONG unterstützt

Bisher wurde der elementare Typ **UTCLONG**, der auf neueren SAP-Systemen verfügbar ist, nicht explizit unterstützt. Ein Feld dieser Art auf der Selektionsmaske führte dazu, dass SE16XXL sich über einen unbekanntes Feldtyp beschwerten würde:



Durch Entfernung des inkriminierten Feldes aus der Selektionsmaske war es jedoch möglich, die Selektion durchzuführen und somit eine Ergebnisliste zu erhalten.

Ab Version 3.6A von SE16XXL wurde diese Unannehmlichkeit beseitigt. UTCLONG-Felder werden jetzt wie alle anderen Arten von Feldern unterstützt.

In der Ergebnisliste werden solche Felder entsprechend den Benutzervorgaben formatiert, wie unten gezeigt:

LANGU	OWNER	TIMESTAMP	CONTENTS
	SAP	14.02.2011 07:24:59,0000000 AM	000001000A003030
	SAP	16.02.2011 06:20:54,0000000 AM	03504E478D0A1A0A
D	SAP	11.12.2020 02:37:19,7705820 PM	474946383961D007

In der Detailansicht wird das interne Format verwendet:

NAME	NAME
'ABAPIcon.ico'	'NAME
'LANGU	'LANGU
'OWNER	'OWNER
'2011-02-14 07:24:59.0000000'	'TIMESTAMP
'000001000A00303010000100040068060000A600000020201	

[Zum Anfang](#)

Zusätzliche Verbesserungen

Transaktionscode /TFTO/SHOW_URL

Mithilfe dieses Transaktionscodes in einer Sprungdefinition ist es möglich, eine **Internetseite** ausgehend von einer SE16XXL-Ergebnisliste anzuzeigen.

Diese Funktionalität wird anhand eines einfachen Beispiels veranschaulicht.

Beispiel

Wir beginnen mit der Selektion der Datensätze der Tabelle **SCARR**, die Informationen über verschiedene Fluggesellschaften enthält. Die Ergebnisliste könnte wie folgt aussehen:

SE16XXL - Tabelle SCARR - 18 Einträge selektiert

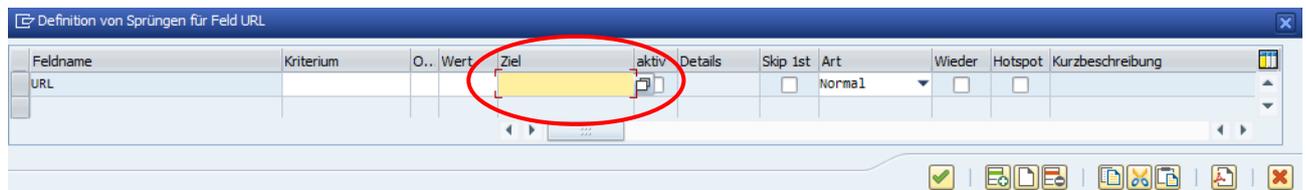
Tabelle SCARR - Fluggesellschaft

MANDT	CARRID	CARRNAME	CURRCODE	URL	
<input type="checkbox"/>	800	AA	American Airlines	USD	http://www.aa.com
<input type="checkbox"/>	800	AB	Air Berlin	EUR	http://www.airberlin.de
<input type="checkbox"/>	800	AC	Air Canada	CAD	http://www.aircanada.ca
<input type="checkbox"/>	800	AF	Air France	EUR	http://www.airfrance.fr
<input type="checkbox"/>	800	AZ	Alitalia	EUR	http://www.alitalia.it
<input type="checkbox"/>	800	BA	British Airways	GBP	http://www.british-airways.com
<input type="checkbox"/>	800	CO	Continental Airlines	USD	http://www.continental.com
<input type="checkbox"/>	800	DL	Delta Airlines	USD	http://www.delta-air.com
<input type="checkbox"/>	800	FJ	Air Pacific	USD	http://www.airpacific.com

Nach dem Markieren der **URL**-Spalte rufen wir folgende Menüfunktion auf:

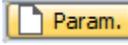
Extras → Feld-Sprünge definieren

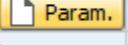
Das zugehörige Dialogfenster erscheint:



Wir geben **/TFTO/SHOW_URL** in das Feld "Ziel" ein:

O.. Wert	Ziel	aktiv	Details
	/TFTO/SHOW_URL	<input type="checkbox"/>	

Sobald ENTER gedrückt wird, erscheint die  Schaltfläche und es ist somit möglich, die notwendigen Parameter anzugeben:

O..	Wert	Ziel	aktiv	Details	Skip	1st	Art	Wieder
		/TFTO/SHOW_URL	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		Normal	<input type="checkbox"/>

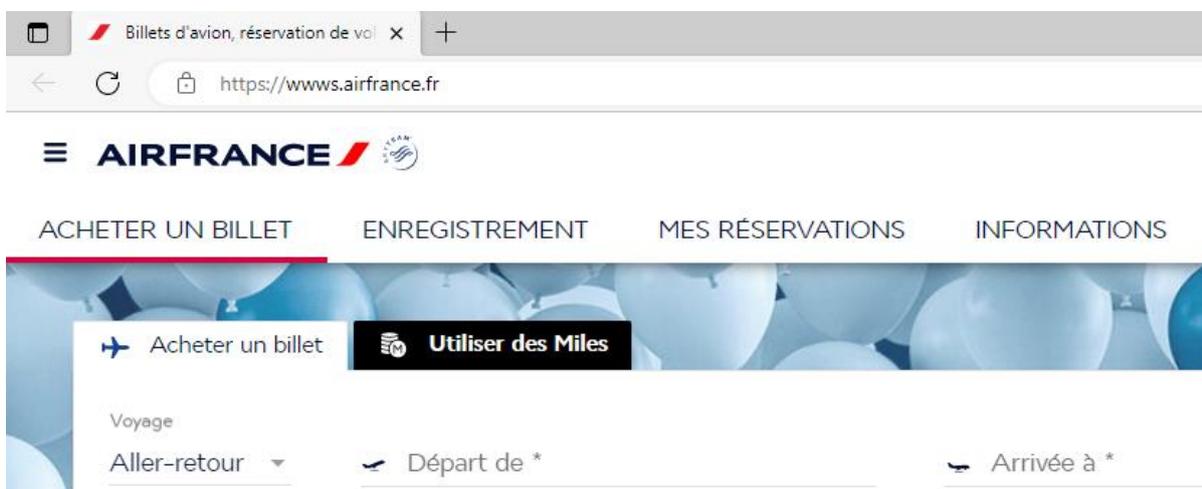
Im darauffolgenden Dialogfenster weisen wir das Feld mit der URL dem SET/GET-Parameter /TFTO/TX_URL zu:

SET/GET-Parameter	Aktiv	Oblig.	Feld / Wert	Parameter-Text
***	<input type="checkbox"/>	<input type="checkbox"/>		Dummy MEMID
/TFTO/TX_URL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	URL	 .6XXL - Internet URL
	<input type="checkbox"/>	<input type="checkbox"/>		

Die Sprungdefinition ist somit vollständig und wir können zur Ergebnisliste zurückkehren.

ANMERKUNG: Der SET/GET-Parameter *** wird nur benötigt, wenn die URL als Literal angegeben wird, z.B. **‘www.top-flow.de’**. In diesem Fall beschwert sich das Programm, dass kein Feld der Ergebnisliste zugeordnet wurde. Um diese formale Anforderung des Dialogs zu erfüllen, weisen Sie einfach ein beliebiges Feld der Ergebnisliste dem Dummy-SET/GET-Parameter *** zu.

Um nun zu sehen, ob der Sprung tatsächlich funktioniert, klicken wir auf **“http://www.airfrance.fr”**. Als Ergebnis erscheint die Homepage von Airfrance:



The screenshot shows a web browser window with the URL <https://www.airfrance.fr>. The page displays the Air France logo and navigation menu with options: ACHETER UN BILLET, ENREGISTREMENT, MES RÉSERVATIONS, and INFORMATIONS. Below the menu, there is a search bar with a dropdown menu showing 'Voyage' and 'Aller-retour'. There are also input fields for 'Départ de *' and 'Arrivée à *'.

Es scheint zu funktionieren!

Script-Katalog – Definition von Globalen Parametern

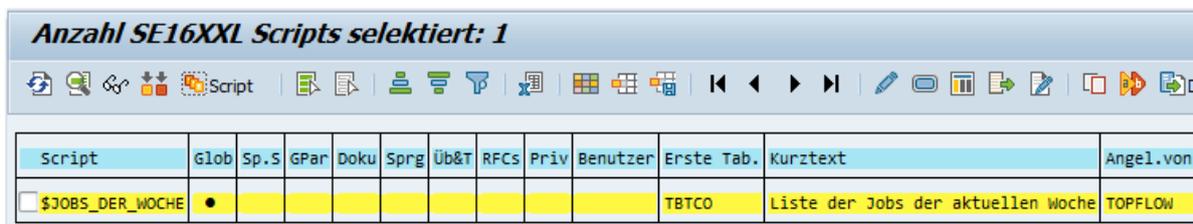
Bisher konnten die globalen Parameter nur während einer SE16XXL-Sitzung oder mithilfe des Script-Editors definiert werden.

Ab Version 3.6A von SE16XXL ist es auch möglich, diese Definition aus dem Script-Katalog durchzuführen, ebenso wie andere Zusätze eines Scripts, wie die spezielle Selektionsmaske oder die Überschriften.

Ein kurzes Beispiel wird diese neue Funktionalität veranschaulichen.

Wir beginnen mit einem Script, das Datensätze der Tabelle **TBTCO** (Job-Zustandsübersichtstabelle) selektiert. Wir rufen den Script-Katalog für dieses Script auf:

Anzahl SE16XXL Scripts selektiert: 1

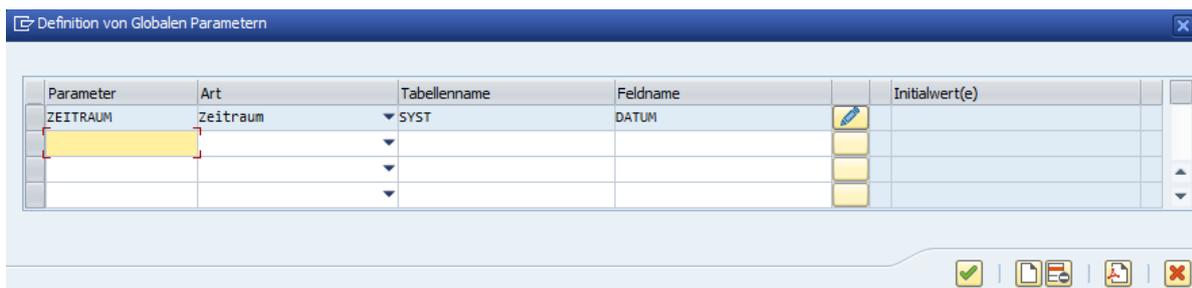


Script	Glob	Sp.S	GPar	Doku	Sprg	Üb&T	RFCs	Priv	Benutzer	Erste Tab.	Kurztext	Angekl. von
<input type="checkbox"/> \$JOBS_DER_WOCHE	•									TBTCO	Liste der Jobs der aktuellen Woche	TOPFLOW

Nun wählen wir die Menüfunktion

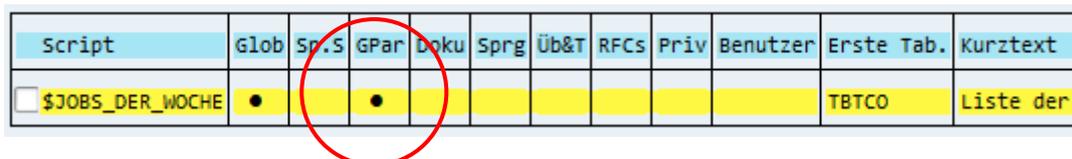
Script-Zusätze → Globale Parameter definieren

Das zugehörige Dialogfenster wird angezeigt. Wir definieren einen globalen Parameter vom Typ **Zeitraum** wie folgt:



Parameter	Art	Tabellenname	Feldname	Initialwert(e)
ZEITRAUM	Zeitraum	SYST	DATUM	

Zurück zum Script-Katalog zeigt unser Scripteintrag nun an, dass globale Parameter definiert wurden (GPar):



Script	Glob	Sp.S	GPar	Doku	Sprg	Üb&T	RFCs	Priv	Benutzer	Erste Tab.	Kurztext
<input type="checkbox"/> \$JOBS_DER_WOCHE	•		•							TBTCO	Liste der

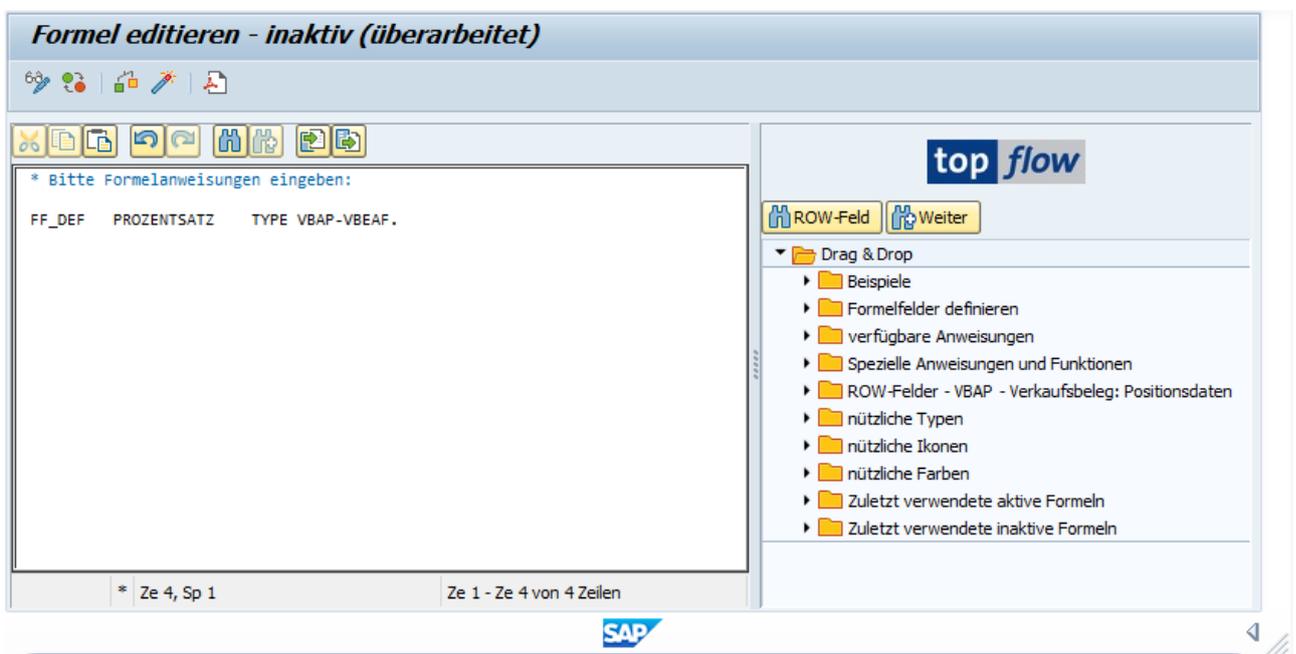
An dieser Stelle würden wir mit dem Aufbau der speziellen Selektionsmaske usw. fortfahren, wie im [zweiten](#) und [dritten](#) Thema dieses Dokuments beschrieben.

Formeleditor – Wort durch Doppelklick markieren

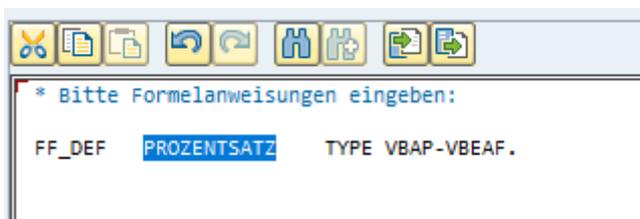
Um mehr Flexibilität bei der Nutzung des Formeleditors zu bieten, ist es nun möglich, ein bestimmtes Wort der Formel durch **Doppelklick** zu **markieren**. Das markierte Wort kann dann per **Drag & Drop** verschoben oder durch Drücken von STRG beim Drag & Drop kopiert werden. Oder das Kontextmenü kann durch einen Rechtsklick mit der Maus abgerufen werden.

Beispiel

Angenommen, Sie bearbeiten folgende Formel:



Mit einem **Doppelklick** auf PROZENTSATZ können Sie dieses Wort **markieren**:



Jetzt können Sie mit **CNTL+C** (oder über das Kontextmenü) das Wort kopieren und in zusätzliche Anweisungen einfügen. Oder Sie verschieben per Drag & Drop – mit gedrücktem STRG – eine Kopie davon an eine andere Stelle in der Formel.

[Zum Anfang](#)

Neue Arten von Berechtigungsprüfungen

Berechtigungen für Primärtabellen

Bisher wurden die Berechtigungsprüfungen auf Satzebene für einzelne Datenbanktabellen definiert und von SE16XXL bei der Selektion von Datensätzen einer dieser Tabellen durchgeführt. Ab **Version 3.6** von SE16XXL ist es dem Administrator möglich, eine neue Art von Berechtigungsprüfungen zu definieren, die anhand der Datensätze sogenannter **Primärtabellen** durchgeführt werden. Für eine gegebene Datenbanktabelle (in diesem Zusammenhang als **„Sekundärtabelle“** bezeichnet) ist es möglich, eine oder mehrere **„Primärtabellen“** zu definieren, die durch festgelegte Join-Kriterien mit der ersteren verbunden sind. Wenn Datensätze einer solchen Sekundärtabelle aus der Datenbank selektiert werden, werden **intern** auch die relevanten Datensätze der zugehörigen Primärtabellen **selektiert** und die definierten Berechtigungsprüfungen an diesen Primärdatensätzen durchgeführt. Nur wenn alle Primärdatensätze die Prüfungen bestehen, zeigt SE16XXL den entsprechenden Sekundärdatensatz an. Andernfalls wird der Sekundärdatensatz verworfen. Werden einige Datensätze aufgrund dieser Berechtigungsprüfungen verworfen, wird die übliche Meldung ausgegeben, so dass an der Oberfläche alles das gewohnte Aussehen hat. Nur im Falle einer **View** kann eine Meldung ähnlich der folgenden ausgegeben werden:

 Notwendige Felder für Primär-Berechtigungsprüfungen fehlen in View U_16022 !

Diese Meldung wird ausgegeben, wenn einer View einige oder alle Felder fehlen, die für den Zugriff auf die definierten Primärtabellen erforderlich sind, die für die oben beschriebenen Berechtigungsprüfungen notwendig sind.

Strenge Berechtigungsprüfungen für Views

Im Falle einer View werden die entsprechenden Berechtigungsprüfungen auf Datenebene aus den zugrunde liegenden Datenbanktabellen übernommen. Allerdings **enthält** eine View **nicht unbedingt** alle Felder, die zur Durchführung der definierten Berechtigungsprüfungen erforderlich sind.

In solchen Fällen werden nur die möglichen Berechtigungsprüfungen durchgeführt. Falls jedoch die globale Einstellung **„Strenge Berechtigungsprüfungen für Views“** aktiviert ist, ermittelt das Programm, ob der Benutzer eine allgemeine Berechtigung (**) für alle fehlenden Felder hat. Ist dies der Fall, werden die möglichen Prüfungen durchgeführt. Andernfalls **weigert** sich das Programm, die View zu prozessieren, und es wird eine Meldung wie folgende ausgegeben:

 Notwendige Felder f. strenge Berechtigungsprüfungen fehlen in View U_25800 !

[Zum Anfang](#)

Die Selektionskriterien der Pseudo-Tabelle \$APPLOGH sind wie folgt:

SE16XXL - Tabelle \$APPLOGH - Selektionsmaske

mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

P_LANGUAGE: P_Sprache:
 Maximale Trefferzahl:
 Breite der Ausgabeliste:

LOGNUMBER	<input type="text"/>	bis	<input type="text"/>		Protokollnr
EXTNUMBER	<input type="text"/>	bis	<input type="text"/>		Ext. Identif.
OBJECT	<input type="text"/>	bis	<input type="text"/>		Objekt
SUBOBJECT	<input type="text"/>	bis	<input type="text"/>		Unterojekt
ALDATE	<input type="text"/>	bis	<input type="text"/>		Datum
ALUSER	<input type="text"/>	bis	<input type="text"/>		Benutzer
ALTCODE	<input type="text"/>	bis	<input type="text"/>		Transakt.Code
ALPROG	<input type="text"/>	bis	<input type="text"/>		Programm
ALMODE	<input type="text"/>	bis	<input type="text"/>		Betriebsmodus
PROBCLASS	<input type="text"/>	bis	<input type="text"/>		Max. Probl.kl.
LOG_HANDLE	<input type="text"/>	bis	<input type="text"/>		Handle
MSG_CNT_AL	<input type="text"/>	bis	<input type="text"/>		Gesamt
MSG_CNT_A	<input type="text"/>	bis	<input type="text"/>		A
MSG_CNT_E	<input type="text"/>	bis	<input type="text"/>		E
MSG_CNT_W	<input type="text"/>	bis	<input type="text"/>		W
MSG_CNT_I	<input type="text"/>	bis	<input type="text"/>		I
MSG_CNT_S	<input type="text"/>	bis	<input type="text"/>		S
MSG_CNT_P1	<input type="text"/>	bis	<input type="text"/>		A
MSG_CNT_P2	<input type="text"/>	bis	<input type="text"/>		A
MSG_CNT_P3	<input type="text"/>	bis	<input type="text"/>		A
MSG_CNT_P4	<input type="text"/>	bis	<input type="text"/>		A
TABNAME	<input type="text"/>	bis	<input type="text"/>		Struktur

Die wichtigsten Kriterien sind **OBJECT**, **SUBOBJECT**, **ALDATE**, **ALUSER** und **ALTCODE**. Der einzige Primärschlüssel, **LOGNUMBER**, ist nur eine laufende Nummer, die verwendet werden kann, um diese Einträge mit den entsprechenden Einträgen der Pseudo-Tabelle **\$APPLOGS** zu verbinden.

Mittels \$APPLOGH wären die gleichen Daten wie im vorherigen Beispiel wie folgt:

SE16XXL - Tabelle \$APPLOGH - 97 Einträge selektiert

Tabelle \$APPLOGH - SE16XXL - Anwendungs-Log-Kopf

MANDT	LOGNUMBER	EXTNUMBER	OBJECT	OBJ_TEXT	SUBOBJECT	SUB_TEXT	ALDATE	ALTIME	ALUSER
800	00000000000002945867	0000000000000058-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	Protokoll BOMBOS-Schnittstelle	23.06.2017	12:35:43	BAUR
800	00000000000002945868	0000000000000058-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	Protokoll BOMBOS-Schnittstelle	23.06.2017	12:35:56	BAUR
800	00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	Protokoll BOMBOS-Schnittstelle	27.06.2017	08:11:25	BAUR
800	00000000000002946051	100-100-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	Protokoll BOMBOS-Schnittstelle	27.06.2017	08:21:11	BAUR
800	00000000000002946052	100-100-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	Protokoll BOMBOS-Schnittstelle	27.06.2017	09:18:49	BAUR
800	00000000000002946053	100-100-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	Protokoll BOMBOS-Schnittstelle	27.06.2017	09:19:14	BAUR

Nun verbinden wir diese Datensätze mit den zugehörigen Meldungen (\$APPLOGS).

Das Ergebnis könnte wie im folgenden Bild aussehen:

SE16XXL - Outer-Join - 860 Ergebnis-Sätze

Join von \$APPLGH(A) und \$APPLGS(B)

A~LOGNUMBER	A~EXTNUMBER	A~OBJECT	A~OBJ_TEXT	A~SUBOBJECT	A~ALDATE	A~ALTIME	B~MSGNUMBER	B~MSGTY_ICON	B~MSG_TEXT
00000000000002945868	0000000000000058-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	23.06.2017	12:35:56	11	▲	Positionsmaterial 100-100 ist keiner Spezifikation zugeordnet
00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:11:25	1	■	Generierungsprozess der Stückliste ist erfolgreich gestartet
00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:11:25	2	■	Zusammensetzung wird generiert
00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:11:25	3	■	Dem Material 100-300 ist keine Spezifikation zugeordnet
00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:11:25	4	■	Daten werden in den Puffer geschrieben
00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:11:25	5	■	Daten wurden erfolgreich gesichert
00000000000002946038	100-300-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:11:25	6	▲	Positionsmaterial 100-310 ist keiner Spezifikation zugeordnet
00000000000002946051	100-100-1000-1-01	EHDI	EHS: Import	IMP_BOMBOS	27.06.2017	08:21:11	1	■	Generierungsprozess der Stückliste ist erfolgreich gestartet

Manchmal enthalten einige der Log-Meldungen zusätzliche Umfeldinformationen, die in der Standardtransaktion SLG1 über die Funktion “Techn. Infomationen” (🔗) angezeigt werden können:

Performance Assistant

Meldungstext

Fall 'Test: Manual log of cash receipts' für 01M250033206 wurde eingeplant

Technische Daten

Meldungstyp _____ S (Erfolgsmeldung)

Meldungsklasse _____ FOPC_CASE (Beurteilung, Schwachstelle, Schwachstellenbehebung und Fall)

Meldungsnummer _____ 111

...

Umfeldinformationen

Schlüssel des Falls___ 5C196712D9CA6F4E99AA828D118D7EDE

Planvariante _____ 01

Objekttyp _____ M2

ObjektId _____ 50033206

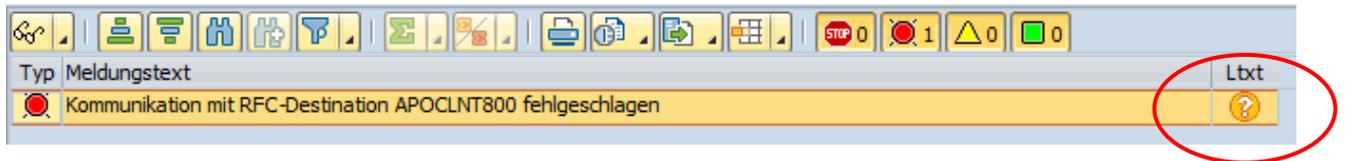
Fallart _____

Dieselben Informationen sind in \$APPLGS in den Feldern **CONTEXT** und **CONTEXT_X** sichtbar, wie im folgenden Beispiel:

B~CONTEXT_TAB	'FOPC_S_LOG_MESSAGE	' Anwendungs-Log: Kontext: Name der DDIC-Struktur
B~CONTEXT	'CASE_GUID=5C196712D9CA6F4E99AA828D118D7EDE PLVAR=01 OTYPE=M2 OBJID=50033206 CASE_TYPE=' SE16XXL - Meldungs-Kontext	
B~CONTEXT_X	'Schlüssel des Falls=5C196712D9CA6F4E99AA828D118D7EDE Planvariante=01 Objekttyp=M2 ObjektId=50033206 Fallart=' SE16XXL - Meldungs-Parameter	
B~PARAMS	'CASE_GUID=5C196712D9CA6F4E99AA828D118D7EDE' SE16XXL - Meldungs-Parameter	

Da sich die Umfelddaten von Meldung zu Meldung unterscheiden können, werden die einzelnen Felder durch zwei vertikale Balken ”||” getrennt, um sie leichter auseinanderzuhalten.

Einige Log-Meldungen haben einen Langtext, der in SLG1 durch die Ikone  gekennzeichnet ist:



\$APPLLOGS hat ein entsprechendes Feld mit dem Namen **DOCU_ICON**:

B~MSGV4					' Nachrichtenvariable
B~MSGTY_ICON					SE16XXL - Ikone für Meldungstyp
B~DOCU_FLAG	'X'				SE16XXL - Flag "Meldung hat Langtext"
B~DOCU_ICON					SE16XXL - Ikone für Meldungslangtext
B~DETLEVEL	'1'				Anwendungs-Log: Detaillierungsgrad
B~DETLEVEL_TEXT					Detaillierungslevel 1
B~PROBCLASS	'1'				Anwendungs-Log: Problemklasse einer Meldung
B~PROBCLASS_TEXT					'sehr wichtig
B~PROBCLASS_ICON					SE16XXL - Ikone für Problemklasse
B~MSG_TEXT					'Kommunikation mit RFC-Destination APOCLNT800 fehlgeschlagen'

Dieses Feld kann verwendet werden, um einen Sprung in der Ergebnisliste zu definieren, mit dem der entsprechenden Langtext angezeigt werden kann, genau wie in der Transaktion SLG1.

Die erforderliche Transaktion ist **/TFTO/SHOW_MSG_DOCU**:

Feldname	Kriterium	O..	Wert	Ziel	aktiv	Details	Skip 1st	Art	Wieder	Hotspot	Kurzbeschreibung
B~DOCU_ICON				/TFTO/SHOW_MSG_DOCU	<input checked="" type="checkbox"/>	Param.	<input checked="" type="checkbox"/>	Normal	<input type="checkbox"/>	<input checked="" type="checkbox"/>	SE16XXL - Meldungs-Doku anze

Mit folgenden Parametern:

SET/GET-Parameter	Aktiv	Oblig.	Feld / Wert	Parameter-Text
MAG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	B~MSGID	Message-Arbeitsgebiet
MSN	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	B~MSGNO	SE91-Message-Nr
	<input type="checkbox"/>	<input type="checkbox"/>		

[Zum Anfang](#)

Neue Start-Optionen für Scripts im Hintergrund

Bisher, wenn ein Script im Hintergrund eingeplant wurde, konnten nur das Datum und die Uhrzeit für den Start des Jobs festgelegt werden. Es waren keine kalenderbezogene Optionen verfügbar. Falls eine solche Option gewünscht wurde, war es notwendig, das Einplanungsprogramm /TFTO/TX_SCHED_SCRIPT zu verwenden.

Ab **Version 3.6** von SE16XXL stehen **zwei** Arten von zusätzlichen Start-Optionen zur Verfügung.

Kalenderbezogene Jobausführung

Wenn die Start-Option **“Start verzögert”** gewählt wird, sieht das Dialogfenster zur Einplanung eines Scripts im Hintergrund nun wie folgt aus:

The screenshot shows the 'Script im Hintergrund starten' dialog box. The 'abweichender Jobname' field is checked and contains 'SCRIPT_\$BEISPIEL_SCRIPT'. The 'Job-Klasse' is 'C'. The 'Ausführungsziel' field is empty. The 'Max. Trefferzahl' is '2.000' and the 'Breite der Ausgabeliste' is '1000'. The 'mit Variante' field is empty. The 'Start sofort' radio button is unselected, the 'Start verzögert' radio button is selected and circled in red, and the 'Start an einem Werktag' radio button is unselected. The 'Datum' is '14.05.2022' and the 'Uhrzeit' is '23:00:00'. The 'Jobausführung kalenderbezogen' section is expanded and circled in red, showing the 'Job immer ausführen' checkbox checked. The 'periodisch' checkbox is unselected. The 'mit "order by primary key" selektieren', 'mit reduziertem Speicherbedarf', 'Ergebnisliste in Server-Datei speichern', and 'Ergebnisliste zum SAP-Spool leiten' checkboxes are unselected. The 'Ergebnis-Teilnehmer' button is visible. The dialog box has a blue title bar and a standard Windows-style window control bar.

Beachten Sie den neuen Kasten **“Jobausführung kalenderbezogen”**.

Die Option “Job immer ausführen” entspricht dem bisherigen Verhalten. Wenn diese Option deaktiviert wird, werden die neuen Optionen sichtbar:

The screenshot shows a software interface for job scheduling. At the top, there are input fields for 'Datum' (14.05.2022) and 'Uhrzeit' (23:00:00). Below these is a 'Fabrikkalender' dropdown menu which is currently empty. Underneath is a field for 'Errechnetes Startdatum'. A blue-bordered menu titled 'Jobausführung kalenderbezogen' is open, containing four radio button options: 'Job immer ausführen' (unchecked), 'Ausfallenlassen an Sonn- und Feiertagen' (selected), 'Auf den vorigen Werktag vorziehen' (unchecked), and 'Auf den folgenden Werktag verschieben' (unchecked). At the bottom of the menu is a checkbox for 'periodisch' which is also unchecked. A red oval highlights the entire 'Jobausführung kalenderbezogen' menu.

Um eine dieser neuen Optionen auszuwählen, ist es zwingend erforderlich, einen Fabrikkalender anzugeben:



Nach Angabe des Fabrikkalenders und der gewünschten Option könnte das Ergebnis wie folgt aussehen:

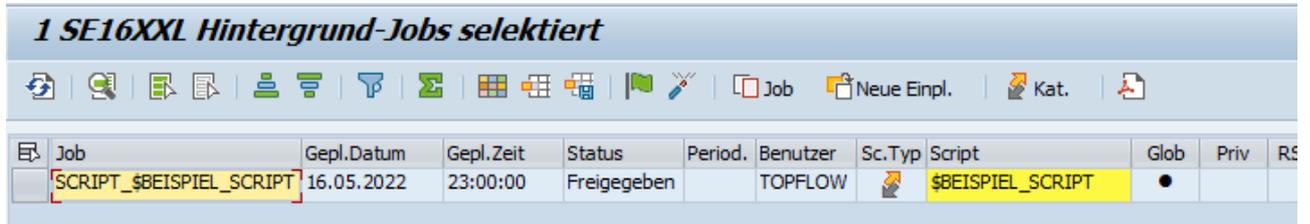
This screenshot shows the same job scheduling interface as before, but with changes. The 'Datum' (14.05.2022) and 'Uhrzeit' (23:00:00) fields are circled in red. The 'Fabrikkalender' dropdown now shows the value '01'. The 'Errechnetes Startdatum' field now displays '16.05.2022', which is also circled in red. The 'Jobausführung kalenderbezogen' menu is still open, and the 'Auf den folgenden Werktag verschieben' option is now selected. The 'periodisch' checkbox remains unchecked.

Da das angegebene Datum kein Werktag gemäß dem gewählten Kalender ist, wird das errechnete Startdatum auf den nächsten Werktag verschoben:

OK - Script '\$BEISPIEL_SCRIPT' eingeplant - Start am 16.05.2022 - 23:00:00 Uhr

In der *Übersicht der Hintergrund-Jobs* wird der neu eingeplante Job wie folgt dargestellt:

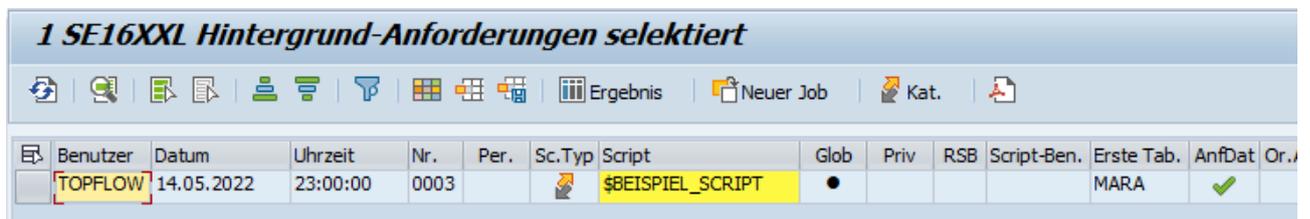
1 SE16XXL Hintergrund-Jobs selektiert



Job	Gepl. Datum	Gepl. Zeit	Status	Period.	Benutzer	Sc. Typ	Script	Glob	Priv	RS
SCRIPT_\$BEISPIEL_SCRIPT	16.05.2022	23:00:00	Freigegeben		TOPFLOW		\$BEISPIEL_SCRIPT	●		

Ähnlich verhält es sich mit der entsprechenden Anforderung:

1 SE16XXL Hintergrund-Anforderungen selektiert

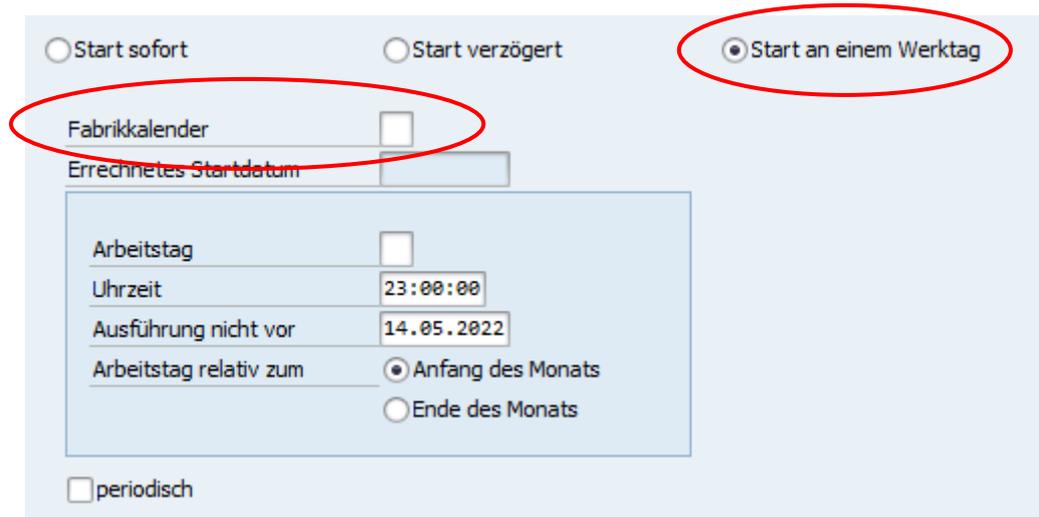


Benutzer	Datum	Uhrzeit	Nr.	Per.	Sc. Typ	Script	Glob	Priv	RSB	Script-Ben.	Erste Tab.	AnfDat	Or.
TOPFLOW	14.05.2022	23:00:00	0003			\$BEISPIEL_SCRIPT	●				MARA	✓	

Der Job weist das geplante Startdatum (16.05.2022) auf, während die Anforderung das ursprüngliche (14.05.2022) Datum anzeigt.

Start an einem Werktag eines Monats

Die zweite Option bezieht sich auf die Arbeitstage des Monats. Das Dialogfenster nimmt in diesem Fall folgendes Aussehen an:



Start sofort
 Start verzögert
 Start an einem Werktag

Fabrikkalender

Errechnetes Startdatum:

Arbeitstag

Uhrzeit:

Ausführung nicht vor:

Arbeitsstag relativ zum:
 Anfang des Monats
 Ende des Monats

periodisch

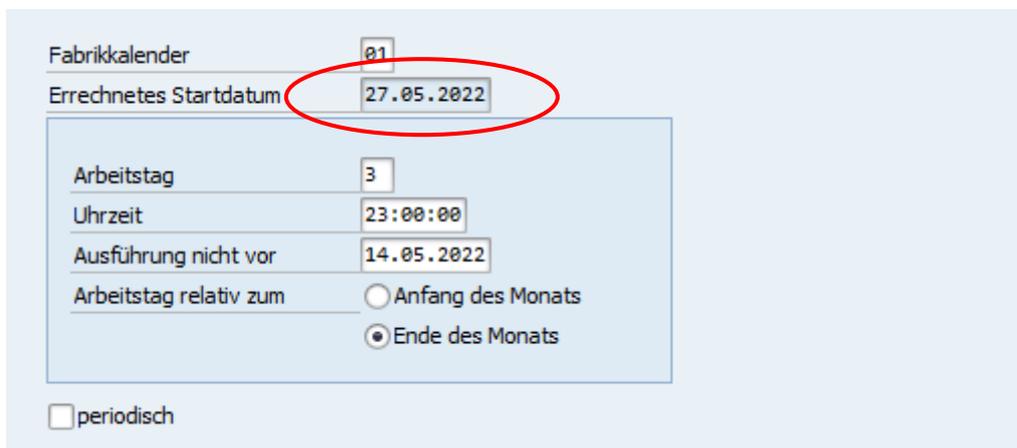
Es ist klar dass diese Option ebenfalls einen Fabrikkalender erfordert.

Außerdem muss ein Arbeitstag angegeben werden (im **Bereich 01 – 19**):



Zusätzlich sollte der Bezugspunkt des Monats angegeben werden, d.h. ob der Arbeitstag relativ zum Anfang oder zum Ende des Monats ist.

In diesem Beispiel wählen wir den **dritten Arbeitstag ab dem Ende des Monats**. Das Ergebnis ist dann wie folgt:



Das errechnete Startdatum ist dann der 27.05.2022:

 OK - Script '\$BEISPIEL_SCRIPT' eingeplant - Start am 27.05.2022 - 23:00:00 Uhr

Die entsprechenden Einträge in der *Übersicht der Hintergrund-Jobs* lauten:

1 SE16XXL Hintergrund-Jobs selektiert											
Job	Gepl.Datum	Gepl.Zeit	Status	Period.	Benutzer	Sc.Type	Script	Glob	Priv	RS	
SCRIPT_\$BEISPIEL_SCRIPT	27.05.2022	23:00:00	Freigegeben		TOPFLOW		\$BEISPIEL_SCRIPT	●			

1 SE16XXL Hintergrund-Anforderungen selektiert													
Benutzer	Datum	Uhrzeit	Nr.	Per.	Sc.Type	Script	Glob	Priv	RSB	Script-Ben.	Erste Tab.	AnfDat	Or
TOPFLOW	27.05.2022	23:00:00	0000			\$BEISPIEL_SCRIPT	●				MARA		✓

[Zum Anfang](#)

Standard-Scriptvarianten U_<benutzer> unterstützt

Bisher war es in SE16XXL möglich, für ein mit einer **speziellen Selektionsmaske** ausgestattetes Script eine Standard-Variante zu definieren. Diese Art der Zuweisung erfolgt im Script-Katalog wie im folgenden Beispiel:



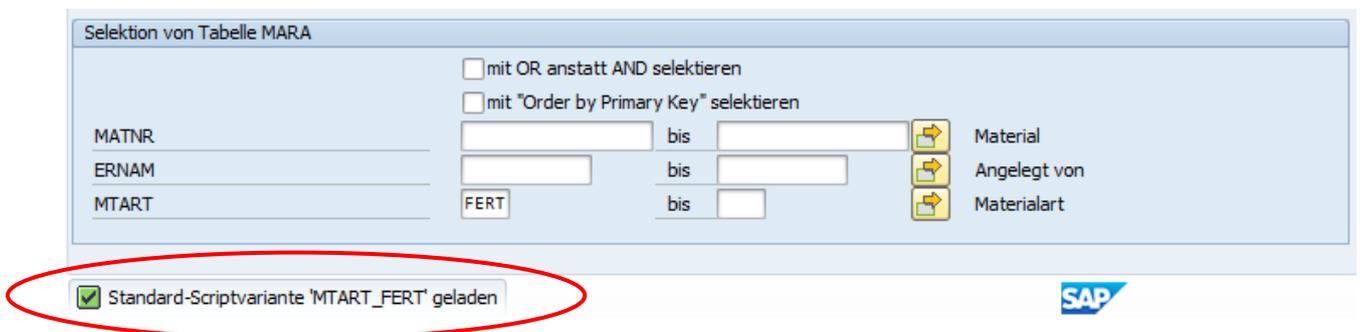
Mithilfe der Menüfunktion

Script-Zusätze → *Std.Scriptvariante setzen/zurücksetzen*

ist es möglich, die gewünschte Variante wie folgt zuzuordnen:



Diese Standard-Scriptvariante wird bei jeder Ausführung des Scripts **automatisch geladen**:

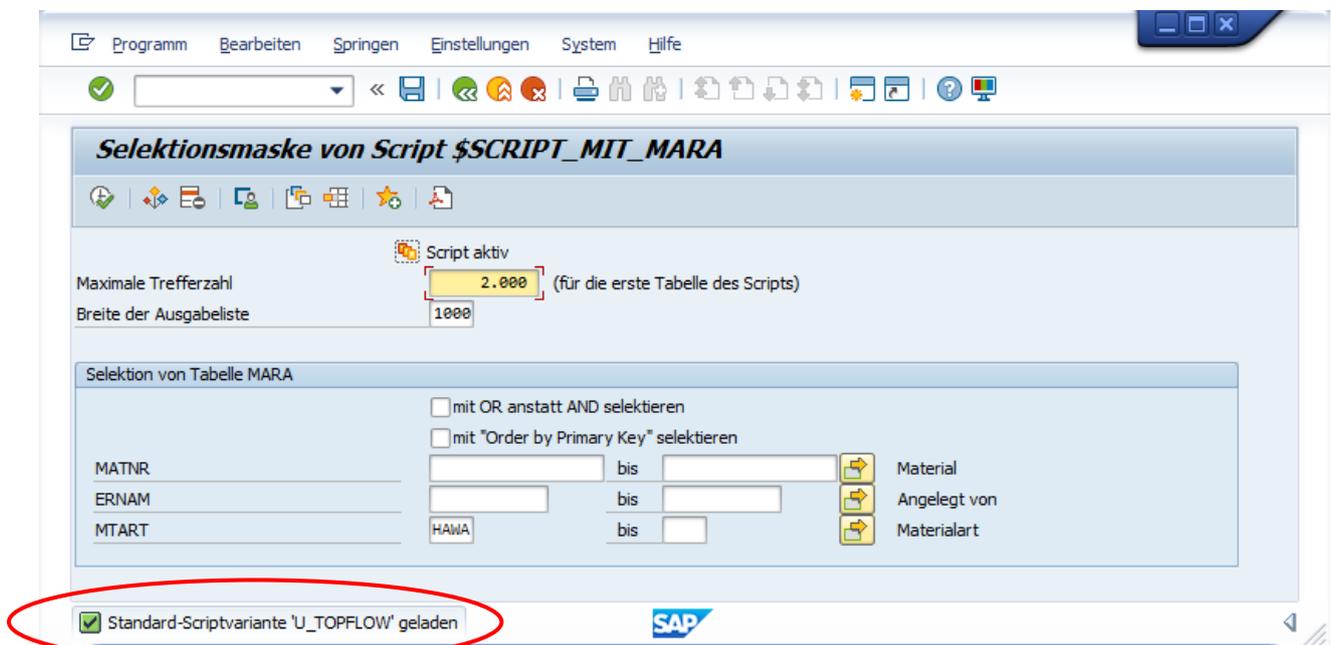


All dies ist bei benutzerspezifischen Scripts in Ordnung, da diese meist von einer Person verwendet werden. Bei **globalen Scripts** ist die Situation jedoch anders. Ein globales Script kann von **vielen Personen** verwendet werden, und es könnte sein, dass jede von ihnen es vorziehen würde, eine **individuelle Standard-Variante** anstelle der bisherigen allgemeinen Standard-Variante zu haben.

Um diese Unannehmlichkeiten zu beseitigen, ist es in SE16XXL ab **Version 3.6** möglich, dass jede Person ihre **eigene individuelle Standard-Scriptvariante** für ein bestimmtes globales Script definiert. Der Mechanismus ist recht einfach und wurde nach ähnlichen Funktionalitäten implementiert, die in Standard-SAP zu finden sind.

Eine Scriptvariante mit dem Namen **U_<benutzer>**, wobei <benutzer> der Logon-Name des Benutzers ist, wird **automatisch** geladen, wenn diese Person das zugehörige globale Script ausführt. Diese individuelle Standardvariante hat die **höchste Priorität**, d.h. auch wenn das Script bereits eine allgemeine Standard-variante hat, wird die individuelle Variante zuerst in Betracht gezogen.

Im obigen Beispiel erstellt der Benutzer TOPFLOW eine Scriptvariante mit dem Namen **U_TOPFLOW**. Wenn das Script von TOPFLOW ausgeführt wird, wird diese individuelle Variante anstelle der allgemeinen Standard-Scriptvariante geladen:



Alle anderen Personen, die keine individuelle Standardvariante haben, erhalten bei der Ausführung dieses Scripts weiterhin die allgemeine Standardvariante.

ANMERKUNG: Dieser Mechanismus gilt nur für **globale Scripts**.

[Zum Anfang](#)

Zusätzliche Verbesserungen

Felder vom Typ 'CHAR' länger als 255 als Selektionskriterien

Bisher konnte in SE16XXL ein Feld vom Typ 'CHAR', das länger als **255** Zeichen ist, **nicht** als Selektionskriterium **verwendet werden**.

Beispiel: Tabelle **APS_ODA_W_URL** mit folgender Struktur:

Dictionary: Tabelle anzeigen

Transp.Tabelle: APS_ODA_W_URL aktiv
Kurzbeschreibung: URL für Open Discovery API

Eigenschaften | Auslieferung und Pflege | **Felder** | Eingabehilfe/-prüfung | Währungs-/Mengenfelder | Indizes

Suchhilfe | Eingebauter Typ | 1 / 7

Feld	K...	In...	Datenelement	Datentyp	Länge	Dez...	KoordSys...	Kurzbeschreibung	Gruppe
PACKAGE_ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	APS_ODA_PACKAGE...	CHAR	30	0		Ø ODA - Paket-ID	
ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	UUID	RAW	16	0		Ø Globale eindeutige Id für Dictionary-Objekt	
TECHNICAL_NAME	<input type="checkbox"/>	<input type="checkbox"/>	APS_ODA_URL TEC...	CHAR	128	0		Ø Technischer Name	
DISPLAY_NAME	<input type="checkbox"/>	<input type="checkbox"/>	APS_ODA_URL DIS...	CHAR	128	0		Ø Name anzeigen	
URL	<input type="checkbox"/>	<input type="checkbox"/>	APS_ODA_URL URL	CHAR	512	0		Ø URL	
VERSION	<input type="checkbox"/>	<input type="checkbox"/>	APS_ODA_URL VER	CHAR	20	0		Ø Version	
TEXT	<input type="checkbox"/>	<input type="checkbox"/>	APS_ODA_URL TEXT	STRING	0	0		Ø URL-Text	

Das Feld **URL** konnte aufgrund seiner Länge von **512** Zeichen nicht als Selektionskriterium verwendet werden:

APS_ODA_W_URL - Felder für Selektion auswählen

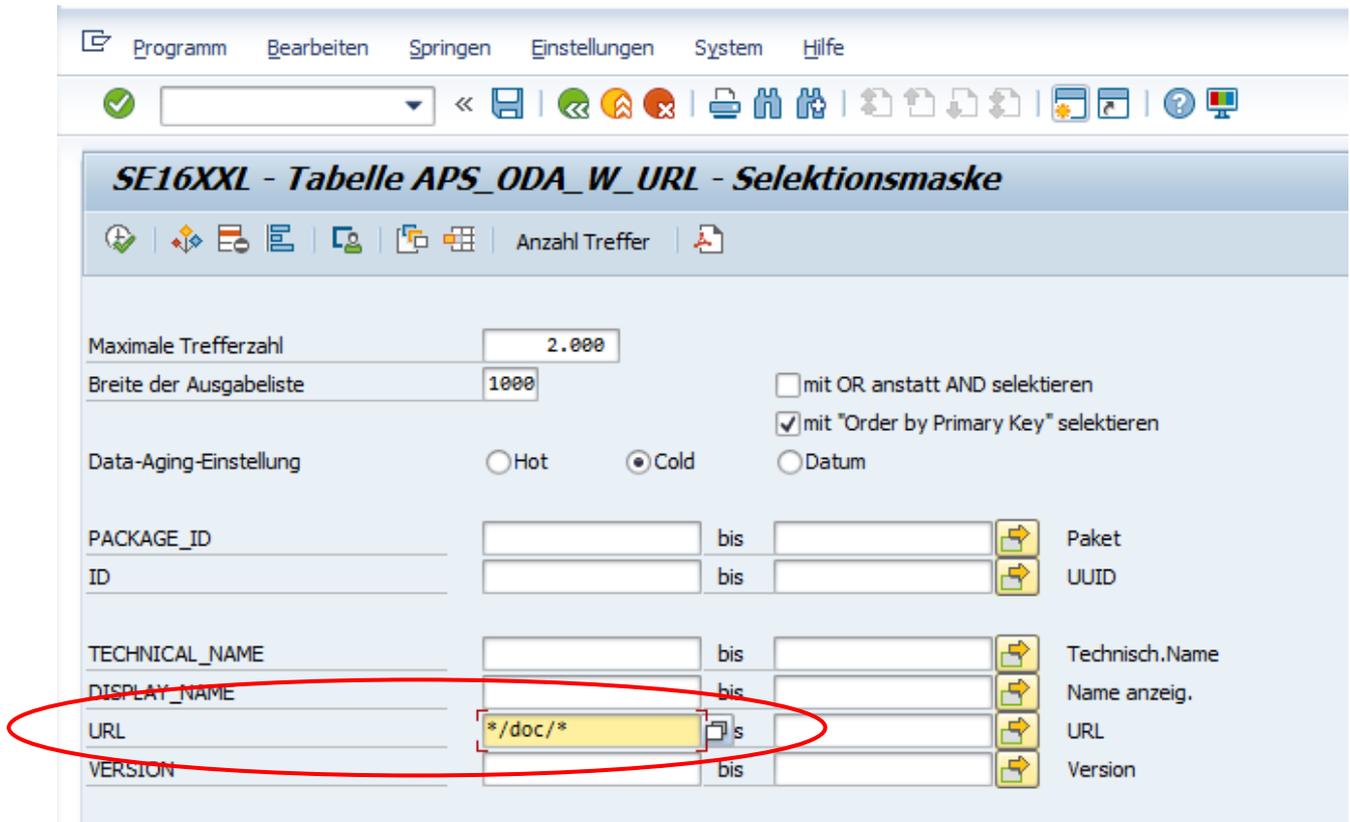
Feld	Schl.	Beschreibung
<input checked="" type="checkbox"/> PACKAGE_ID	X	ODA - Paket-ID
<input checked="" type="checkbox"/> ID	X	Globale eindeutige Id für Dictionary-Objekt
<input checked="" type="checkbox"/> TECHNICAL_NAME		Technischer Name
<input checked="" type="checkbox"/> DISPLAY_NAME		Name anzeigen
<input checked="" type="checkbox"/> VERSION		Version

Dieses Manko wurde nun beseitigt. Ab **Version 3.6** von SE16XXL ist es möglich, solche langen Felder als Selektionskriterien auszuwählen:

APS_ODA_W_URL - Felder für Selektion auswählen

Feld	Schl.	Beschreibung
<input checked="" type="checkbox"/> PACKAGE_ID	X	ODA - Paket-ID
<input checked="" type="checkbox"/> ID	X	Globale eindeutige Id für Dictionary-Objekt
<input checked="" type="checkbox"/> TECHNICAL_NAME		Technischer Name
<input checked="" type="checkbox"/> DISPLAY_NAME		Name anzeigen
<input checked="" type="checkbox"/> URL		URL
<input checked="" type="checkbox"/> VERSION		Version

In diesem Beispiel verwenden wir den folgenden Selektionswert:

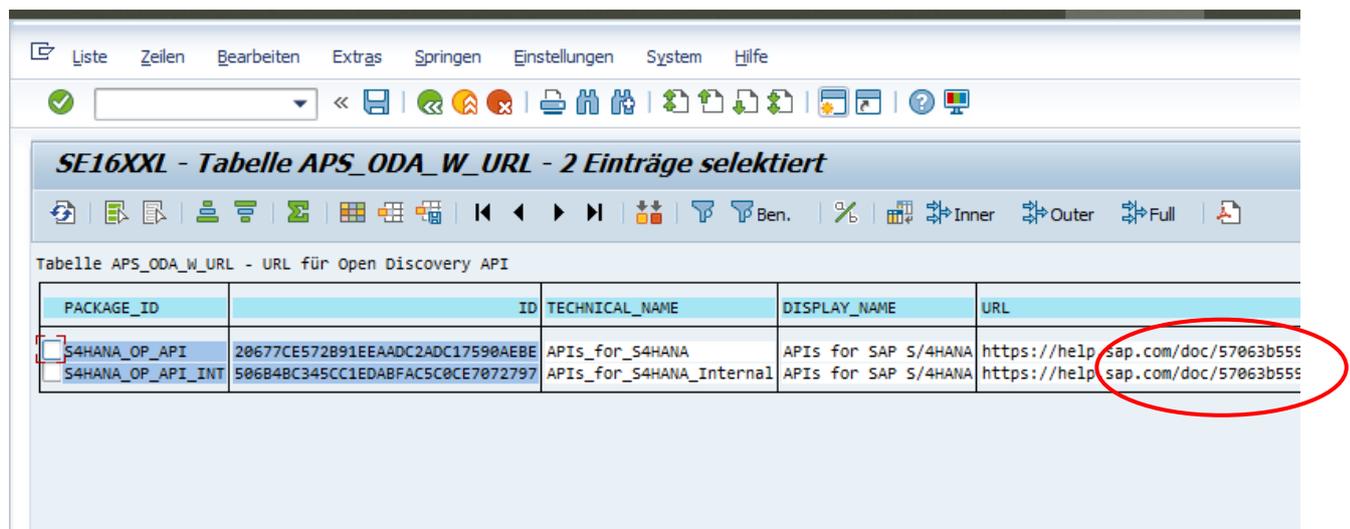


SE16XXL - Tabelle APS_ODA_W_URL - Selektionsmaske

Maximale Trefferzahl: 2.000
 Breite der Ausgabeliste: 1000
 mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren
 Data-Aging-Einstellung: Hot Cold Datum

PACKAGE_ID bis Paket
 ID bis UUID
 TECHNICAL_NAME bis Technisch.Name
 DISPLAY_NAME bis Name anzeig.
 URL bis URL
 VERSION bis Version

Die Ergebnisliste lautet dann wie folgt:



SE16XXL - Tabelle APS_ODA_W_URL - 2 Einträge selektiert

Tabelle APS_ODA_W_URL - URL für Open Discovery API

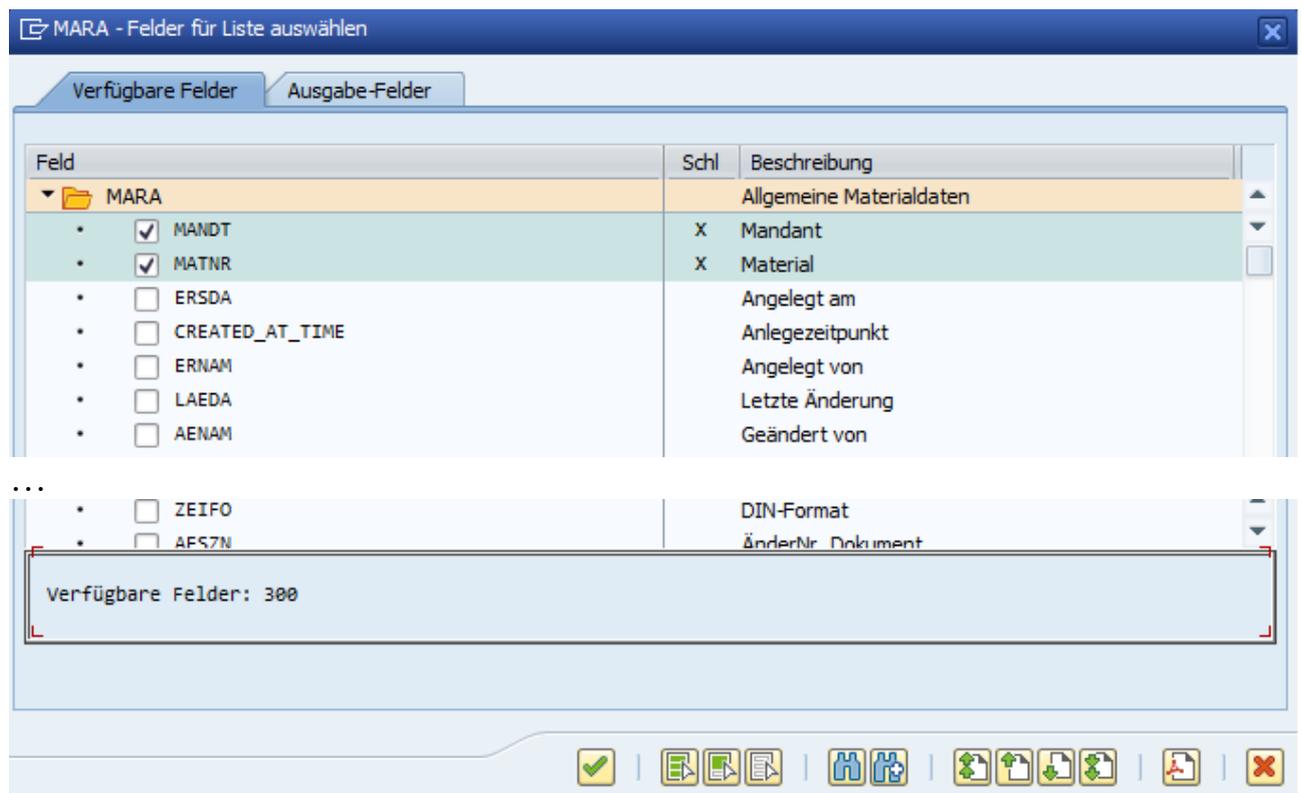
PACKAGE_ID	ID	TECHNICAL_NAME	DISPLAY_NAME	URL
<input checked="" type="checkbox"/> S4HANA_OP_API	20677CE572B91EEAADC2ADC17590AE8E	APIs_for_S4HANA	APIs for SAP S/4HANA	https://help.sap.com/doc/57063b55
<input checked="" type="checkbox"/> S4HANA_OP_API_INT	506B4BC345CC1EDABFAC5C0CE7072797	APIs_for_S4HANA_Internal	APIs for SAP S/4HANA	https://help.sap.com/doc/57063b55

Neue Sortieroptionen bei die Auswahl der Listenfelder

Bei der Auswahl der Felder für die Ergebnisliste ist es nun möglich, die **Sortierreihenfolge** der **Feldnamen** oder der **Feldbeschreibungen** zu ändern, wodurch es für weniger erfahrene Anwender einfacher wird, “ihre” Felder zu finden.

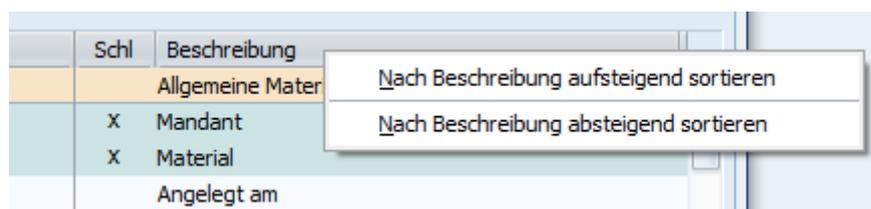
Dieses neue Feature wird anhand eines Beispiels mit der Tabelle MARA erläutert.

Das Dialogfenster hat normalerweise folgendes Aussehen:



Die Felder sind entsprechend der **DDIC-Definition** angeordnet.

Wenn nun der Mauszeiger auf die Spaltenüberschrift “**Beschreibung**” gesetzt wird und die rechte Maustaste gedrückt wird, erscheint folgendes **Kontextmenü**:

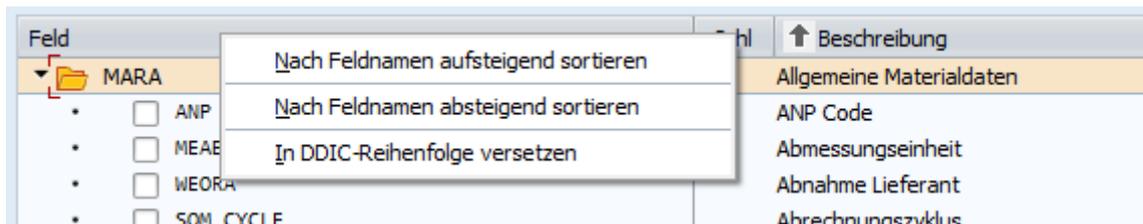


Mithilfe der ersten Funktion erhalten wir entsprechend den Feldsektor **sortiert** nach der **Feldbeschreibung** in aufsteigender Reihenfolge:

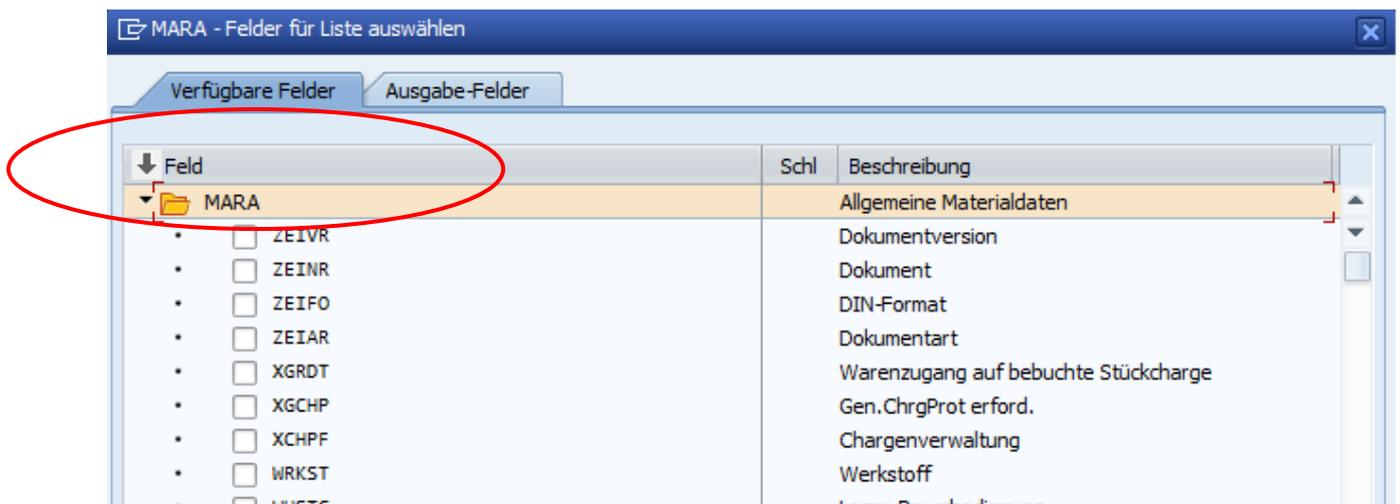


Beachten Sie den kleinen Pfeil () , der die Sortierrichtung anzeigt.

Gleiches gilt auch für die Spaltenüberschrift “**Feld**”:



Wenn wir dieses Mal die zweite Funktion wählen, würden wir Folgendes erhalten:



[Zum Anfang](#)

Summieren/Zählen auf der Datenbank

Die SQL **SELECT**-Anweisung der Datenbank bietet verschiedene Arten von **Aggregatausdrücken** wie **SUM()**, **COUNT()**, **AVG()** und ähnliches. Mithilfe dieser Ausdrücke ist es möglich, Summierungs- und/oder Zählungs-Operationen **direkt** auf der Datenbank durchzuführen.

Bisher machte SE16XXL von diesen Möglichkeiten der Datenbank keinen Gebrauch. Operationen mit großen Datenmengen, die summiert oder gezählt werden mussten, führten entweder zu sehr langen Antwortzeiten oder kamen wegen Speichermangel nicht zu einem Ergebnis.

Ab **Version 3.5B** wurde diese Lücke geschlossen. Es ist nun unter bestimmten Bedingungen möglich, das Summieren/Zählen von der Datenbank durchführen zu lassen. Ein einfaches Beispiel wird zeigen, wovon die Rede ist.

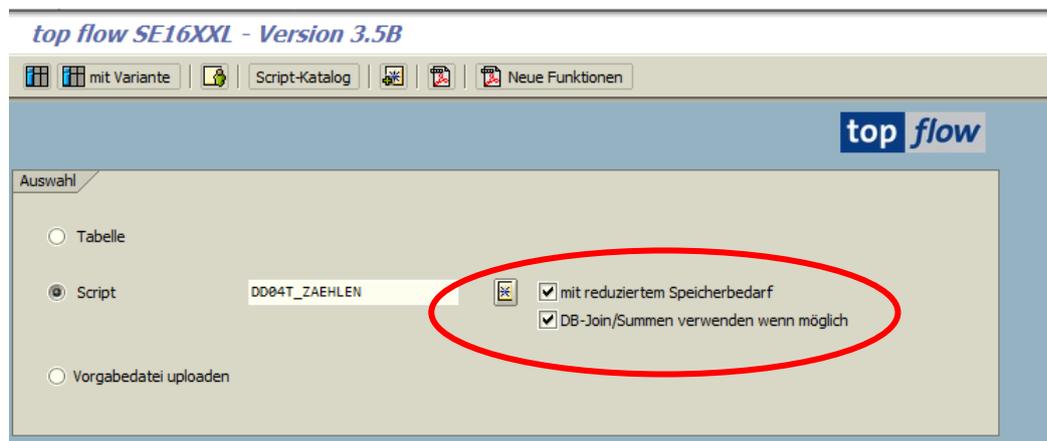
In diesem Beispiel möchten wir herausfinden, **wie viele Einträge pro Sprache** in der Tabelle **DD04T** (R/3-DD: Texte der Datenelemente) gespeichert sind. Diese Tabelle enthält mehr als **sieben Millionen** Datensätze. Um unser Ziel zu erreichen, erstellen wir ein Script, das aus folgenden Operationen besteht:

- 1) Selektion von Tabelle **DD04T** mit **AS4LOCAL = 'A'**.
- 2) Sortierung der resultierenden Liste nach **DDLLANGUAGE**.
- 3) **Extras** → **Eine Spalte einfügen** → mit **Anzahl sortierter Sätze**.
- 4) **Zeilen** → **Benachbarte Duplikate ausblenden**.
- 5) **Zeilen** → **Ausgeblendete Sätze verwerfen**.
- 6) **Extras** → **Zusatztexte einfügen** für **DDLLANGUAGE**.
- 7) Als Listenfelder **DDLLANGUAGE**, den Text dazu und das V-Feld mit der Anzahl festlegen.

Die letzten beiden Operationen dienen lediglich kosmetischen Zwecken.

Wir speichern unser Script unter dem Namen **“DD04T_ZAEHLEN”**.

An dieser Stelle führen wir das Script mit folgenden Optionen durch:



Nach Betätigung von  auf der darauffolgenden Selektionsmaske erscheint für ein paar Sekunden folgende Meldung am unteren Bildschirmrand:

**** DB-Summieren/Zählen verwendet ****

Sie weist darauf hin, dass die Zählung direkt auf der Datenbank durchgeführt wird.

Schließlich erscheint die Ergebnisliste:

Script DD04T_ZAEHLEN - 31 Einträge

Tabellenwerkzeug-Schnittstelle mit einer Liste von Sprachkategorien und deren Anzahlen.

DDLLANGUAGE	V~DDLLANGUAGE_TXT	V~ANZAHL
<input type="checkbox"/> W	Bulgarisch	199.933
<input type="checkbox"/> C	Tschechisch	274.646
<input type="checkbox"/> K	Dänisch	236.293
<input type="checkbox"/> D	Deutsch	356.533
<input type="checkbox"/> G	Griechisch	217.098
<input type="checkbox"/> E	Englisch	379.813
<input type="checkbox"/> S	Spanisch	306.207
<input type="checkbox"/> U	Finnisch	227.258
<input type="checkbox"/> F	Französisch	316.975
<input type="checkbox"/> B	Hebräisch	5
<input type="checkbox"/> 6	Kroatisch	213.750
<input type="checkbox"/> H	Ungarisch	268.066
<input type="checkbox"/> I	Italienisch	299.752
<input type="checkbox"/> J	Japanisch	307.807
<input type="checkbox"/> 3	Koreanisch	260.279
<input type="checkbox"/> N	Niederländisch	295.331
<input type="checkbox"/> O	Norwegisch	228.254
<input type="checkbox"/> L	Polnisch	266.101
<input type="checkbox"/> P	Portugiesisch	285.543
<input type="checkbox"/> 4	Rumänisch	203.875
<input type="checkbox"/> R	Russisch	270.422
<input type="checkbox"/> d	Serbisch(latein)	167.301
<input type="checkbox"/> Q	Slowakisch	251.223
<input type="checkbox"/> 5	Slowenisch	196.695
<input type="checkbox"/> V	Schwedisch	215.378
<input type="checkbox"/> 2	Thailändisch	203.071
<input type="checkbox"/> T	Türkisch	227.551
<input type="checkbox"/> 8	Ukrainisch	5
<input type="checkbox"/> Z	Kundenreserve	1.441
<input type="checkbox"/> M	Chinesisch trad.	227.711
<input type="checkbox"/> 1	Chinesisch	251.725

ANMERKUNG: Das oben genannte Script muss nicht unbedingt mit der Option *“DB-Join/Summen verwenden wenn möglich”* ausgeführt werden. Es kann auch auf herkömmliche Weise laufen – in diesem Fall dauert es jedoch länger, bis das Ergebnis vorliegt.

Weitere Informationen zu dieser interessanten Funktionalität finden Sie unter [Summieren/Zählen auf der Datenbank](#).

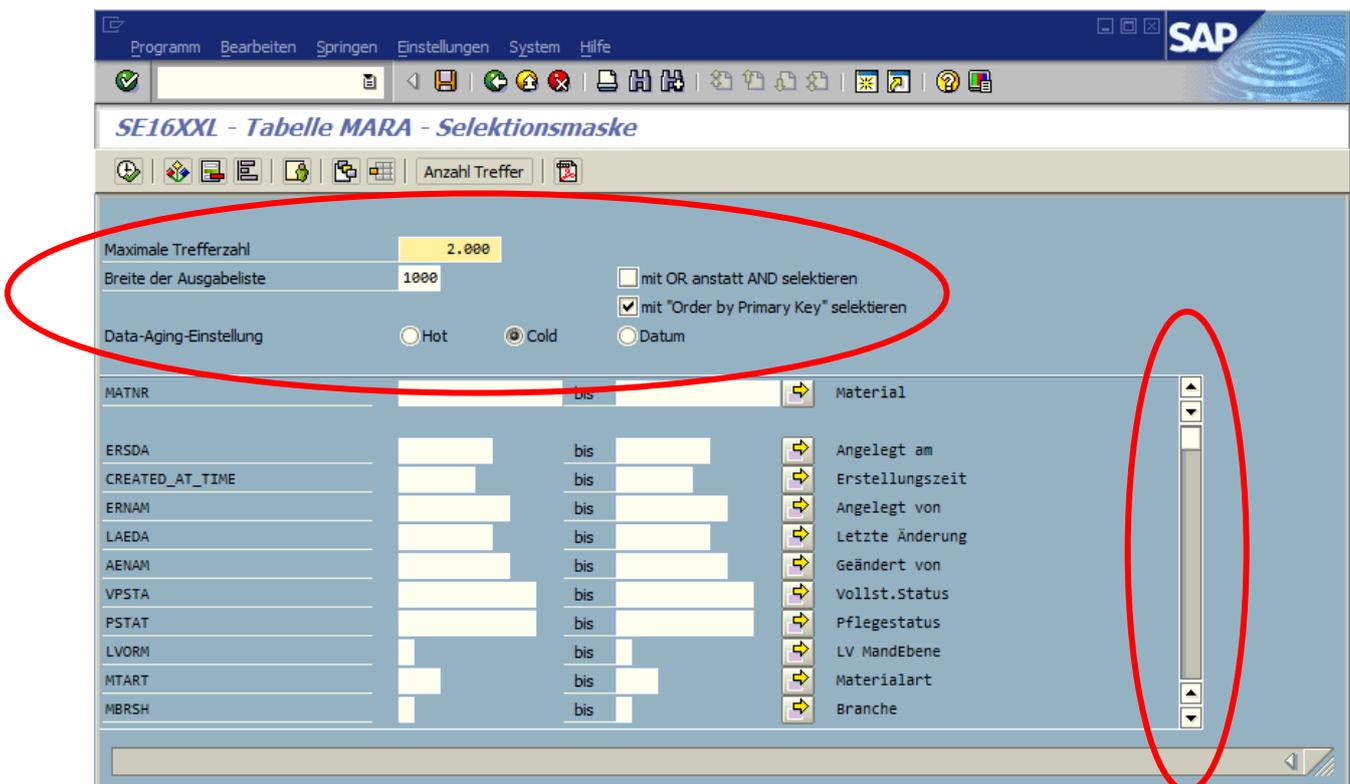
[Zum Anfang](#)

Standard-Selektionsmaske neu gestaltet

Bisher konnte die Standard-Selektionsmaske von SE16XXL maximal **30** Selektionsparameter und **40** Selektionskriterien gleichzeitig anzeigen. Diese Grenzwerte wurden oft als unzureichend empfunden. Um dieses Manko zu beseitigen, wurde die Standard-Selektionsmaske **komplett neu gestaltet**, um bis zu **60** Selektionsparameter und **199** Selektionskriterien gleichzeitig zu akzeptieren. Es sind 199 und nicht 200, weil eine Zeile benötigt wird, um die Schlüsselfelder von den verbleibenden zu trennen (200 ist die maximale Anzahl von Zeilen, die für ein Dynpro möglich ist).

Zudem basiert die neu gestaltete Standard-Selektionsmaske nun auf der **Dynpro-Funktionalität von SAP**. Dies hat unter anderem den SAP-Hinweis **1099527** überflüssig gemacht.

Die neue Selektionsmaske sieht zum Beispiel wie folgt aus:



Übersteigt die Anzahl der Selektionskriterien den sichtbaren Bereich der Maske, erscheint eine Scrollbar – die wichtigsten Parameter und Optionen hingegen bleiben oben sichtbar.

Falls die Tabelle oder View Selektionsparameter aufweist, werden diese **oben** auf der Maske platziert, wie auf der nächsten Seite zu sehen ist:

SE16XXL - DDL SQL View CARUNQCUSTOMER - Selektionsmaske

Maximale Trefferzahl: 2.000
Breite der Ausgabeliste: 1000

mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

Data-Aging-Einstellung: Hot Cold Datum

Parameter	Wert	bis	Wert	Attribut
CUSTOMER		bis		Debitor
COMPANYCODE		bis		Buchungskreis
ACCOUNTINGCLERK		bis		Sachbearbeiter
PAYMENTTERMS		bis		ZahlBeding. Finanzen
DUNNINGPROCEDURE		bis		Mahnverfahren
DUNNINGBLOCK		bis		Mahnsperre

In der Regel sind maximal **acht** Parameter sichtbar. Wenn der Bereich jedoch relativ klein ist, wird diese Zahl wie folgt auf **vier** verringert:

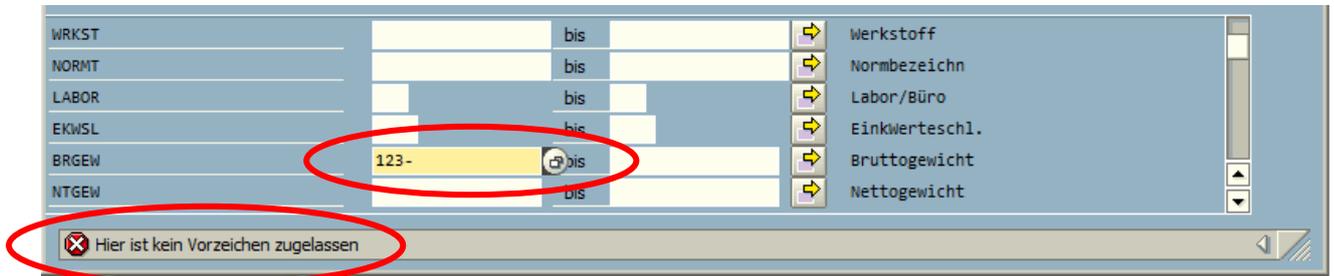
Maximale Trefferzahl: 2.000
Breite der Ausgabeliste: 1000

mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

Data-Aging-Einstellung: Hot Cold Datum

Parameter	Wert	bis	Wert	Attribut
CUSTOMER		bis		Debitor
COMPANYCODE		bis		Buchungskreis
ACCOUNTINGCLERK		bis		Sachbearbeiter
PAYMENTTERMS		bis		ZahlBeding. Finanzen
DUNNINGPROCEDURE		bis		Mahnverfahren
DUNNINGBLOCK		bis		Mahnsperre

Ein weiteres Merkmal der neuen Selektionsmaske ist, dass bei einem fehlerhaften Eingabewert alle übrigen Selektionskriterien **ungeschützt** bleiben:



Dadurch ist es möglich, den Fehler zu korrigieren **und** auch andere Selektionswerte in die restlichen Kriterien einzugeben, ohne die EINGABETASTE betätigen zu müssen.

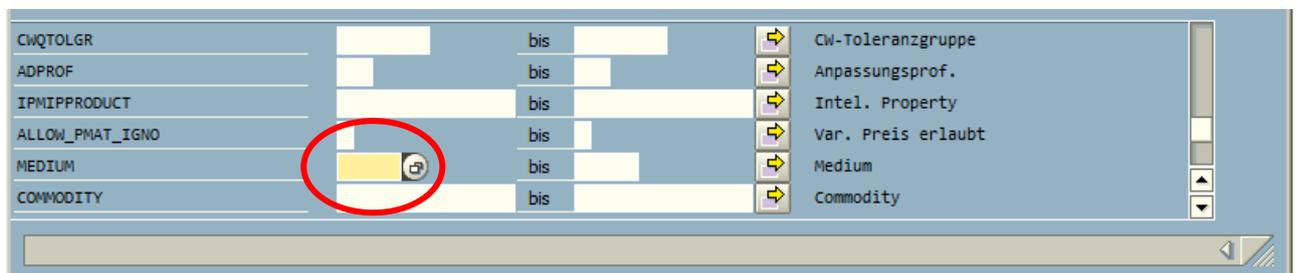
Zum Schluss noch ein kleines Feature. Die Systemfunktionsleiste verfügt nun über zwei Symbolen zur Suche nach Selektionskriterien:



Durch Betätigen von  wird ein entsprechendes Such-Dialogfenster angezeigt:



Nach Betätigung von  setzt das Programm den Fokus auf das **erste** Selektionskriterium, das den Suchparametern entspricht:



Für weitere Informationen lesen Sie bitte [Standard-Selektionsmaske](#).

[Zum Anfang](#)

Script-Selektionsmaske mit Registerkarten

Bisher bestand die spezielle Selektionsmaske eines Scripts aus einer Reihe von **Blöcken**, jeder Block einer bestimmten Selektions- oder Filter-Operation des Scripts zugeordnet. Die folgende Tabelle zeigt die Einschränkungen im Zusammenhang mit dieser Art von Selektionsmaske:

Beschreibung	Maximum
Anzahl Blöcke	12
Anzahl Select-Options für den ersten Block	40
Anzahl Select-Options für die restlichen Blöcke	10

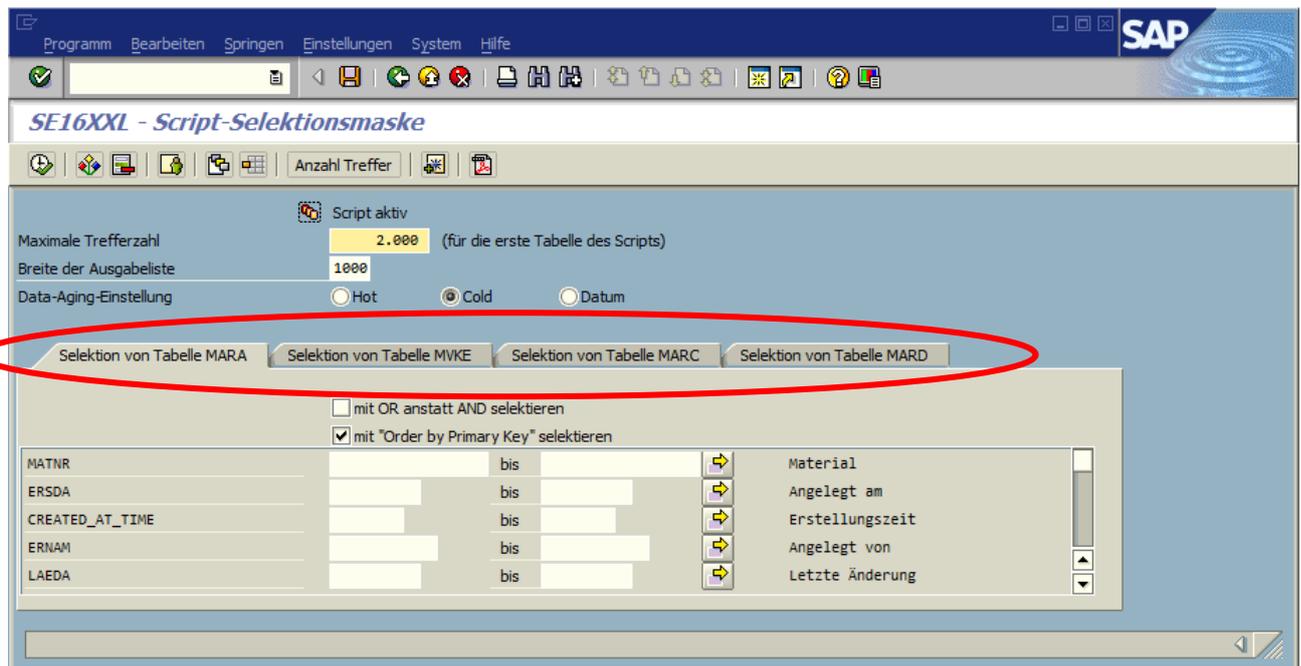
Falls mindestens ein Block mit **Selektionsparametern** vorhanden war, waren die Einschränkungen wie folgt:

Beschreibung	Maximum
Anzahl Blöcke	8
Anzahl Selektionsparameter für den ersten Block	16
Anzahl Selektionsparameter für die restlichen Blöcke	4
Anzahl Select-Options für den ersten Block	40
Anzahl Select-Options für die restlichen Blöcke	10

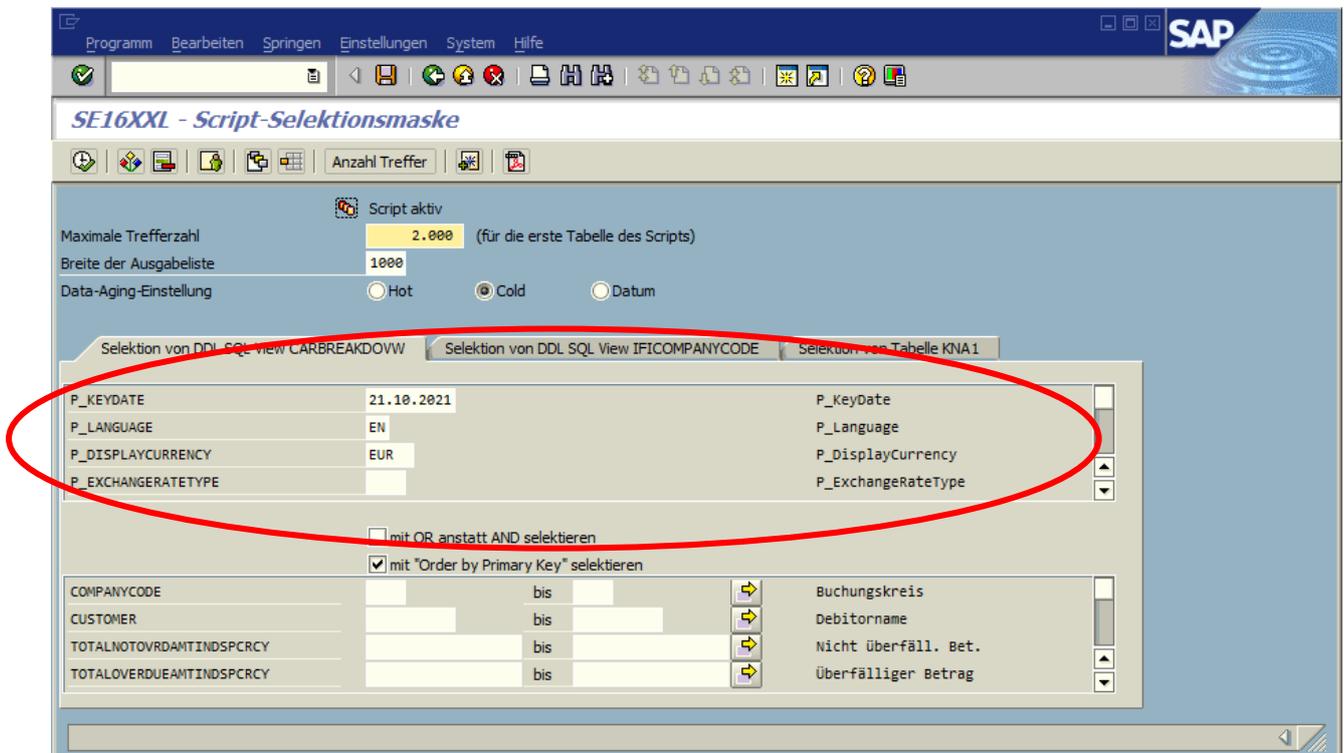
Diese Grenzwerte, die auf Designbeschränkungen von Dynpros basieren, wurden in einigen Situationen als unzureichend empfunden. Um diese Unzulänglichkeit zu überwinden, wurde daher eine neue Art von **Script-Selektionsmaske mit Registerkarten** implementiert. Auf dieser neuen Maske werden die traditionellen Blöcke durch **Registerkarten** ersetzt. Diese neue Art von Selektionsmaske ersetzt **automatisch** die herkömmliche, falls die oben aufgeführten Grenzwerte überschritten werden. Für die neue Art gelten folgende Einschränkungen:

Beschreibung	Maximum
Anzahl Registerkarten	40
Anzahl Selektionsparameter für eine Registerkarte	60
Anzahl Select-Options für eine Registerkarte	200

Auf der nächsten Seite ist ein Beispiel für eine solche Selektionsmaske dargestellt.



Falls eine Registerkarte auch Selektionsparameter aufweist, werden diese ganz oben platziert, wie im folgenden Beispiel zu sehen ist:



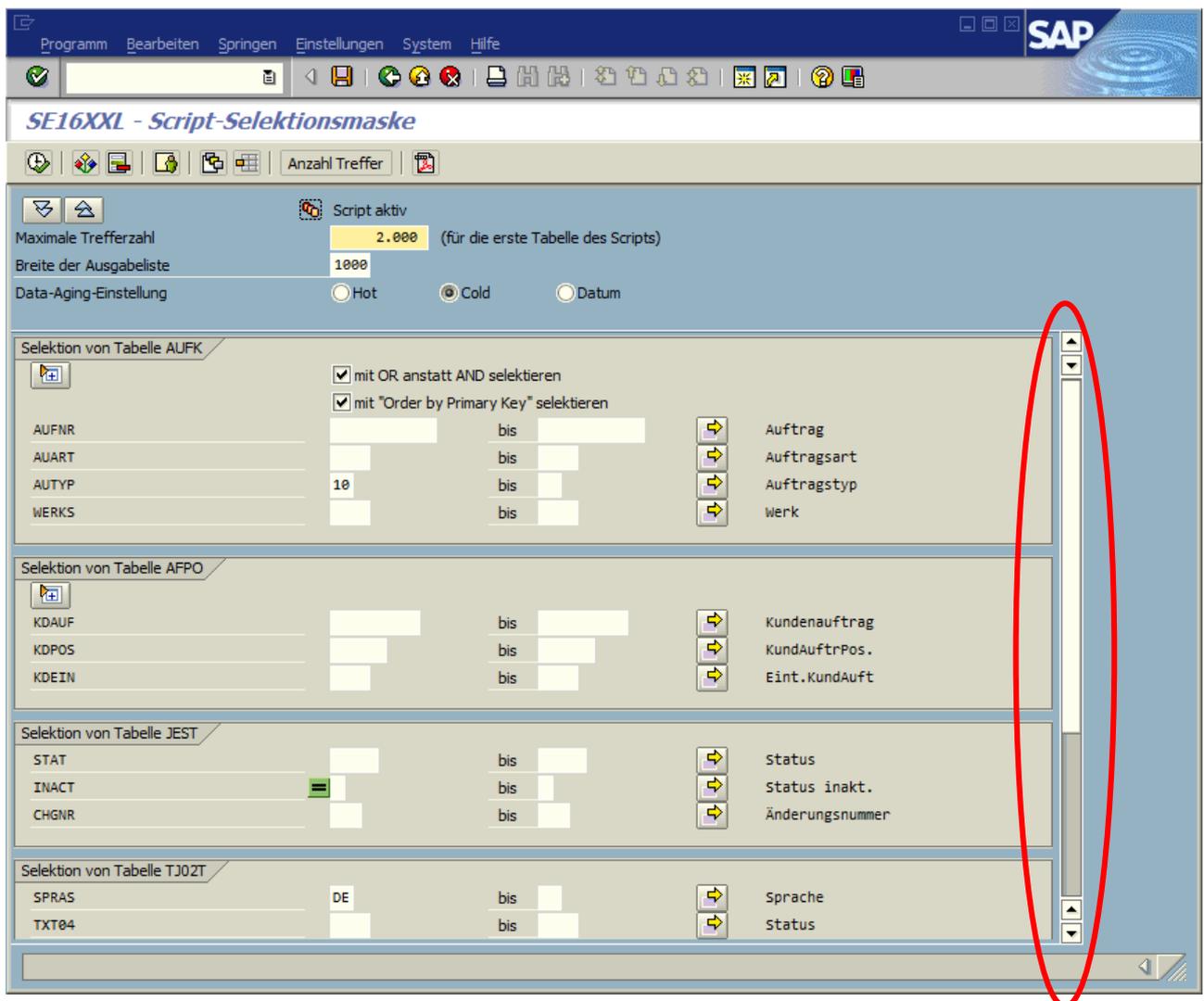
Für weitere Informationen lesen Sie bitte [Script-Selektionsmaske mit Registerkarten](#).

[Zum Anfang](#)

Zusätzliche Verbesserungen

Herkömmliche Script-Selektionsmaske neu gestaltet

Um eine einheitliche Benutzeroberfläche für alle Arten von Selektionsmasken zu bieten, nachdem die Standard-Selektionsmasken komplett neu gestaltet wurde, wurde es als sinnvoll erachtet, auch die herkömmliche Script-Selektionsmaske **neu zu gestalten**. Die Einschränkungen für die Anzahl der Blöcke usw. konnten nicht geändert werden, jedoch das Gesamtverhalten ähnelt nun dem der anderen Selektionsmasken. Die Haupt-Parameter und -Optionen bleiben oben unbeweglich, während die verschiedenen Blöcke mit ihren Select-Options in einem **Subscreen-Bereich** (bei Bedarf mit einer Scrollbar) eingeschlossen sind, wie im nachfolgenden Beispiel zu sehen ist:

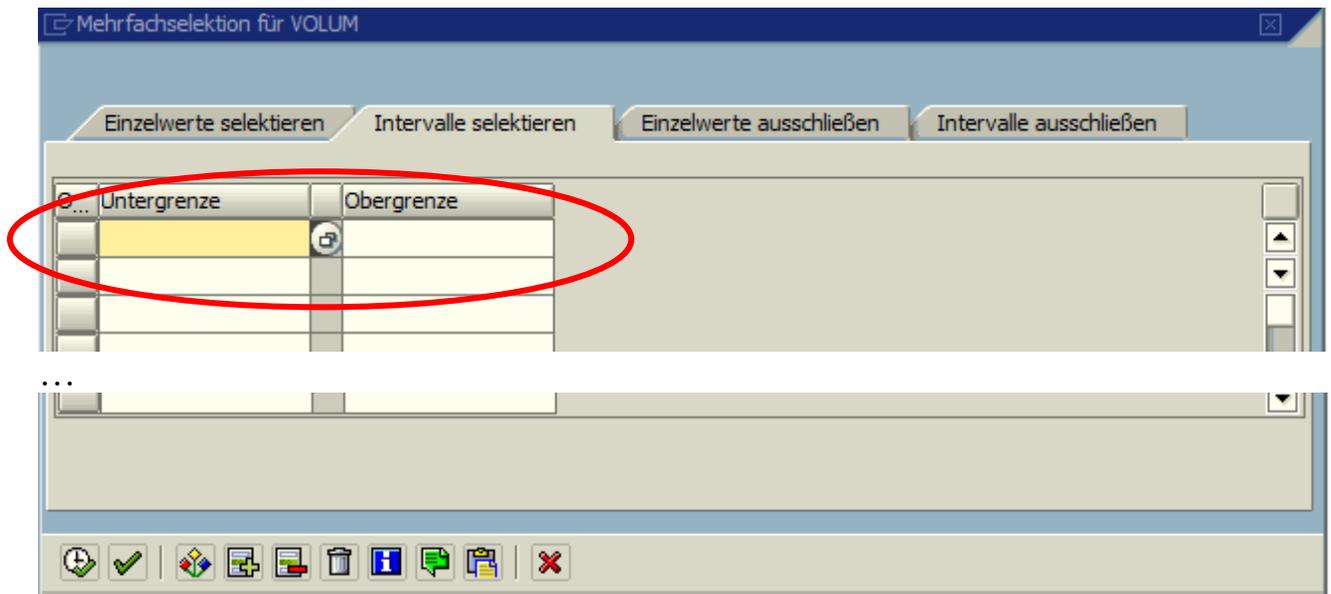


Für mehr Informationen lesen Sie bitte [Spezielle Selektionsmaske](#).

Mehrfachselektions-Dialog neu gestaltet

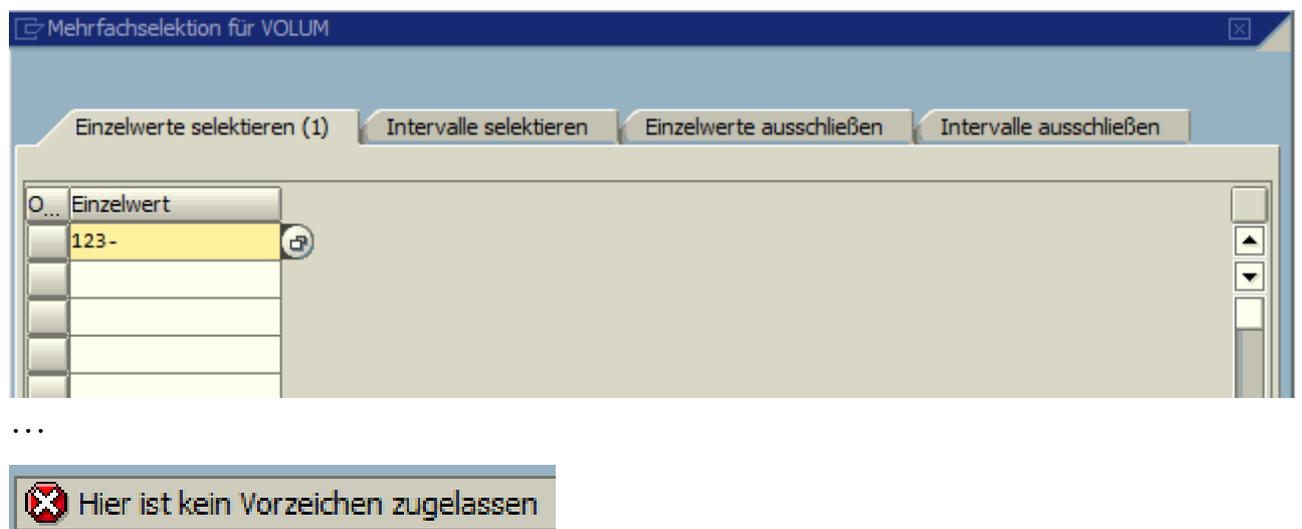
Nicht nur alle Selektionsmasken wurden neu gestaltet. Auch das Dialogfenster für die Mehrfachselektion wurde durch eine interne Entwicklung ersetzt.

Das neue Dialogfenster weist unter anderem eine **Lücke** zwischen der Unter- und Obergrenze auf, wie im folgenden Bild zu sehen ist.



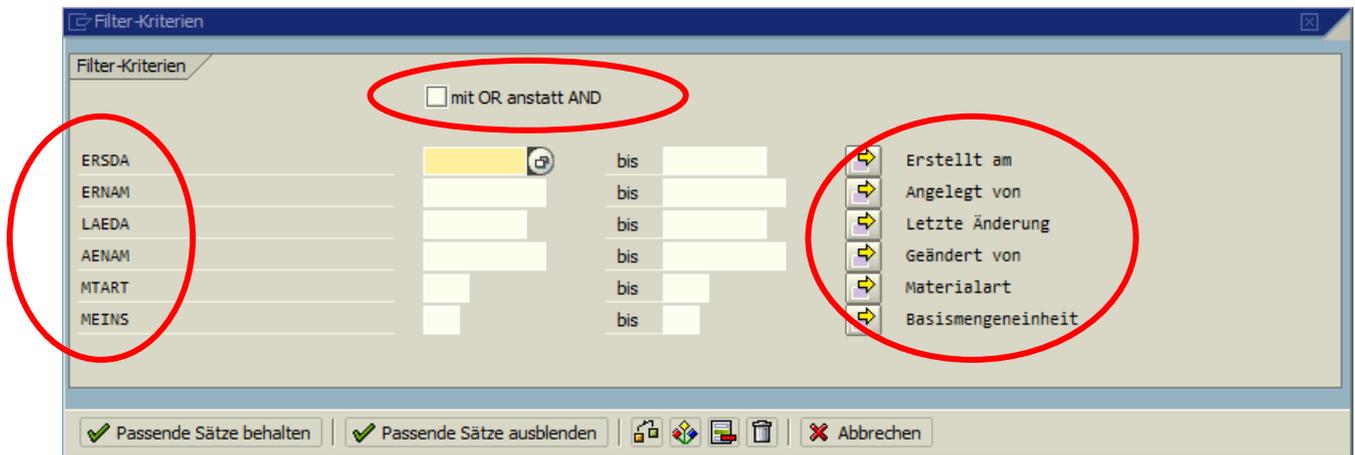
Auf diese Weise **überlappt** das Symbol für die F4-Hilfe () **nicht** die linke Seite der Obergrenze.

Darüber hinaus bleiben im Falle eines fehlerhaften Eingabewertes die anderen Eingabefelder **ungeschützt** und ermöglichen somit die Korrektur des Fehlers **und** die Eingabe zusätzlicher Werte in einem Durchgang, ohne die EINGABETASTE betätigen zu müssen:



Filter-Dialogfenster neu gestaltet

Auch das Dialogfenster zur Angabe der Filterkriterien wurde komplett neu gestaltet. Ein typisches Beispiel könnte wie folgt aussehen:

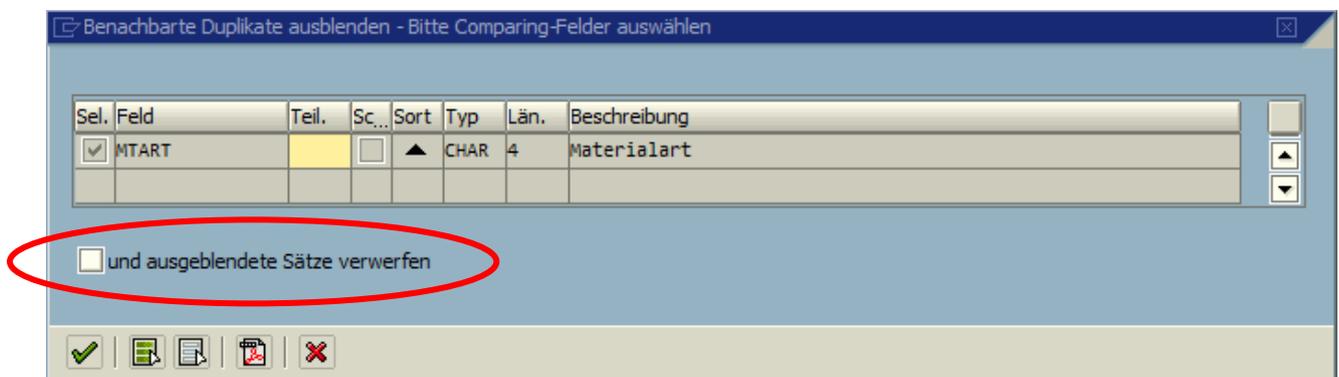


Nun wird die Option “*mit OR anstatt AND*” durch eine echte Checkbox dargestellt. Zusätzlich werden sowohl der **Name** als auch der **Kurztext** der Filterkriterien entsprechend den Benutzereinstellungen wie auf den Selektionsmasken angezeigt.

Ähnlich wie beim Verhalten der Selektionsmasken bleiben alle Filterkriterien **ungeschützt**, wenn ein fehlerhafter Wert erkannt wird.

Benachbarte Duplikate ausblenden – Neue Option

Auf dem Dialog-Fenster zur Festlegung der Comparing-Felder für die Funktion “*Zeilen → Benachbarte Duplikate ausblenden*” befindet sich nun die Option “*und ausgeblendete Sätze verwerfen*”:



[Zum Anfang](#)

Unterstützung der SAP® Data-Aging-Funktionalität

Einleitung

Data Aging ist eine Funktionalität, die auf SAP HANA® basierten Systemen verfügbar ist. Data Aging teilt die Zeilen einiger Datenbank-Tabellen in zwei Bereiche, **aktuell** (hot) und **historisch** (cold). Die Daten des aktuellen Bereichs können effizienter (d.h. schneller) abgerufen werden als die historischen Daten. Eine Reihe von sogenannten **Data-Aging-Objekten** ist vordefiniert, wobei jedes Objekt eine Reihe logisch verbundener Datenbank-Tabellen darstellt. Jede dieser Tabellen enthält ein spezielles Datumsfeld **_DATAAGING**, das verwendet wird, um zwischen den aktuellen Daten (Initialwert) und den historischen Daten (Erstellungsdatum) zu unterscheiden. Durch die Aktivierung bestimmter Data-Aging-Objekte legt die System-Administration fest, welche Datenbank-Tabellen auf diese Weise verarbeitet werden sollen.

Es muss bedacht werden, dass jedes Data-Aging-Objekt **ein eigenes Trenndatum** hat (in Bezug auf was aktuell und was historisch ist). Zwei Jahre alte Bestellanforderungen können noch aktuell sein, während drei Monate alte Anwendungs-Log-Einträge bereits historisch sein können.

Weitere Informationen zum Thema Data Aging finden Sie in der entsprechenden SAP®-Dokumentation.

SE16XXL-Unterstützung

Bis dato unterstützte SE16XXL die Data-Aging-Funktionalität nicht – nur hot Daten (d.h. aktuelle Daten) konnten aus der Datenbank selektiert werden.

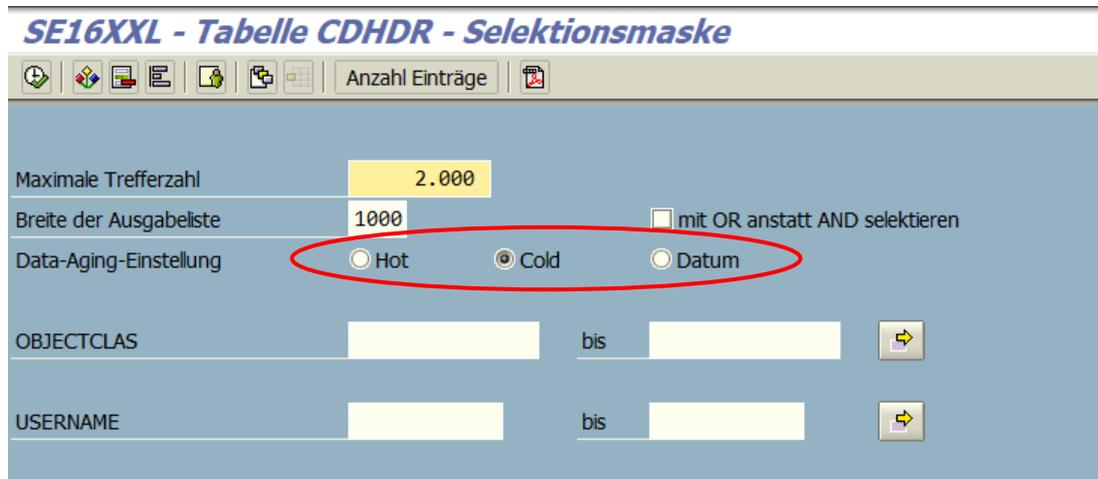
Ab **Version 3.5A** ist es möglich, auch **cold Daten (d.h. historische)** zu selektieren.

Folgende Voraussetzungen sind erforderlich, um cold Daten aus der Datenbank selektieren zu können,

- Das System und die Datenbank müssen Data Aging unterstützen;
- Data Aging muss von der System-Administration aktiviert worden sein;
- In den SE16XXL-Einstellungen kann der Administrator festlegen, dass nur Benutzer mit einer speziellen Rolle auf cold Data-Aging-Daten zugreifen dürfen. In diesem Fall benötigt der Anwender diese Rolle oder eine gleichwertige Berechtigung.

Wenn die oben genannten Voraussetzungen nicht erfüllt sind, kann der Benutzer nur auf hot Daten zugreifen.

Falls alle oben genannten Voraussetzungen erfüllt sind, enthält die SE16XXL-Selektionsmaske **drei Auswahlknöpfe** zur Angabe der Data-Aging-Einstellung:



SE16XXL - Tabelle CDHDR - Selektionsmaske

Anzahl Einträge

Maximale Trefferzahl: 2.000

Breite der Ausgabeliste: 1000 mit OR anstatt AND selektieren

Data-Aging-Einstellung: Hot Cold Datum

OBJECTCLAS: [] bis []

USERNAME: [] bis []

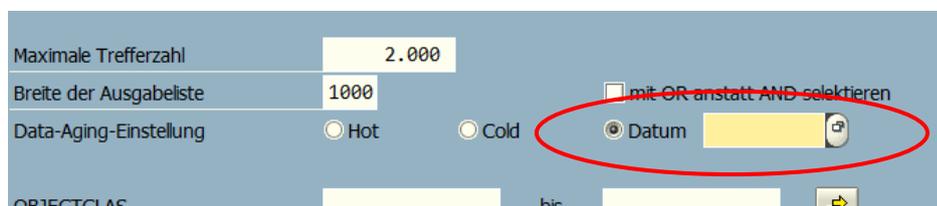
WICHTIG: Die Einstellung “Cold” beinhaltet sowohl **hot** als auch **cold Daten**.

Diese Auswahlknöpfe erscheinen nur auf der **ersten** Selektionsmaske, d. h. auf der Selektionsmaske, die sich auf die erste zu selektierende Tabelle bezieht. Nachfolgende Joins erben implizit diese Einstellung, um konsistente Daten zu liefern.

ANMERKUNG: Die Data-Aging-Auswahlknöpfe sind immer auf der ersten Selektionsmaske vorhanden, auch wenn die zugehörige Datenbank-Tabelle nicht Data-Aging-relevant ist. Dies ist notwendig, weil ein nachfolgender Join eine relevante Tabelle beinhalten könnte.

Darüber hinaus werden sie nicht in einer Variante gespeichert, denn die gleiche Variante könnte von Benutzern verwendet werden, die über keine Data-Aging-Berechtigung verfügen.

Wenn “Datum” gewählt wird, ist es möglich, eine Data-Aging-“Temperatur” anzugeben, d.h. ein Datum als Untergrenze für den Zugriff auf cold Daten:



Maximale Trefferzahl: 2.000

Breite der Ausgabeliste: 1000 mit OR anstatt AND selektieren

Data-Aging-Einstellung: Hot Cold Datum []

OBJECTCLAS: [] bis []

In diesem Fall werden nur cold Daten selektiert mit einem Datum größer oder gleich dem angegebenen Datum. Ältere cold Daten stehen nicht zur Verfügung.

Diese Wahl sollte jedoch vermieden werden, da, wie bereits in der Einleitung erwähnt, verschiedene Data-Aging-Objekte unterschiedliche Trenndaten haben können. Dadurch könnten Daten selektiert werden, die nicht miteinander konsistent sind.

Das **_DATAAGING-Feld**

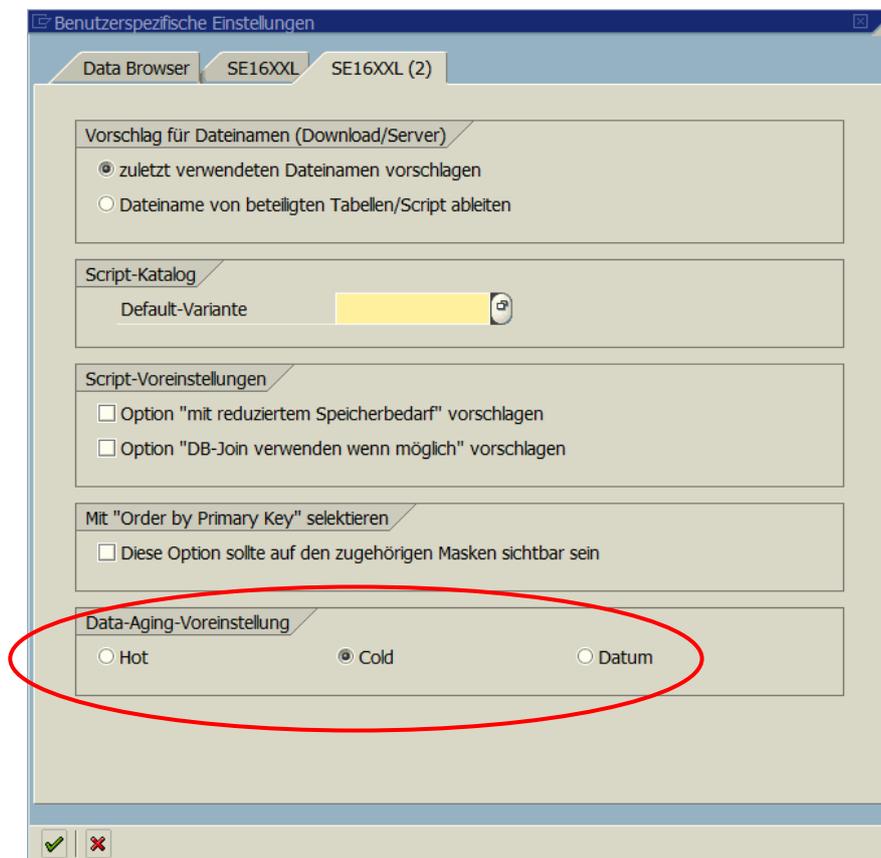
Wie bereits in der Einleitung erwähnt, enthält jede Datenbank-Tabelle, die potenziell Data-Aging-relevant ist, ein Feld namens **_DATAAGING** (oder ähnlich), das anzeigt, ob ein Datensatz aktuell (Initialwert) oder historisch (Erstellungsdatum) ist.

Anstatt die oben erwähnten Auswahlknöpfe zu verwenden, könnte man versucht sein, das **_DATAAGING** Feld auf der Selektionsmaske zu nutzen, um entweder hot oder cold Daten zu selektieren. **Leider funktioniert dieser Ansatz nicht:** Die Datenbank **ignoriert** dieses Selektionskriterium. Um cold Daten zu selektieren, ist es intern notwendig, eine spezielle Methode der Klasse **CL_ABAP_STACK_TEMPERATURE** aufzurufen. Andernfalls liefert die Datenbank nur hot Daten zurück.

Um solche Missverständnisse zu vermeiden, wurde das Feld **_DATAAGING** **gänzlich aus der Selektionsmaske verbannt**.

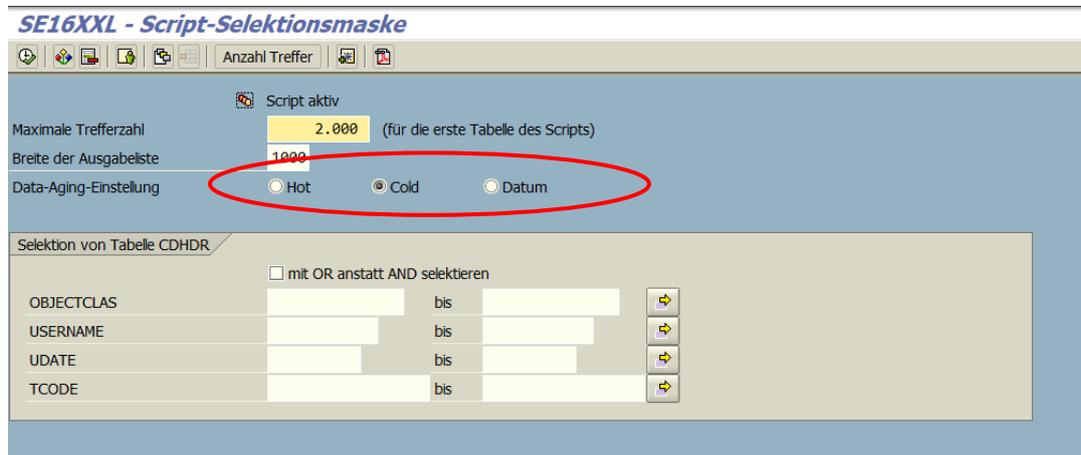
Standard-Voreinstellung

Benutzer ausgestattet mit der erforderlichen Berechtigung können in ihren eigenen benutzerspezifischen Einstellungen eine Data-Aging-Voreinstellung festlegen:



Spezielle Script-Selektionsmaske

Die drei Auswahlknöpfe sind auch auf der speziellen Selektionsmaske eines Scripts verfügbar:



SE16XXL - Script-Selektionsmaske

Anzahl Treffer

Script aktiv

Maximale Trefferzahl: 2.000 (für die erste Tabelle des Scripts)

Breite der Ausgabeliste: 1000

Data-Aging-Einstellung: Hot Cold Datum

Selektion von Tabelle CDHDR

mit OR anstatt AND selektieren

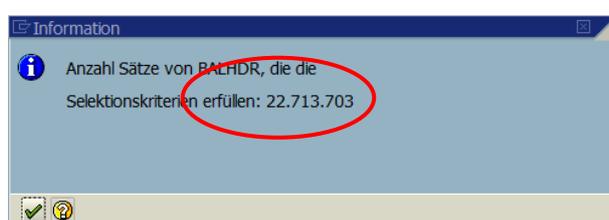
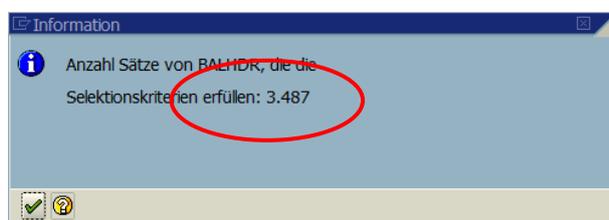
OBJECTCLAS		bis		
USERNAME		bis		
UDATE		bis		
TCODE		bis		

Wie bei einer normalen Selektionsmaske stehen sie nur berechtigten Benutzern zur Verfügung.

ANMERKUNG: Die Data-Aging-Einstellung wird weder mit dem Script noch in einer Script-Variante gespeichert. Der Grund dafür ist, dass das Script und die Varianten auch von Benutzern verwendet werden können, die nicht mit der erforderlichen Berechtigung ausgestattet sind.

Prüfen, ob eine Tabelle oder View über Cold Daten verfügt

Eine einfache Möglichkeit, herauszufinden, ob eine Datenbank-Tabelle oder View für das Data Aging behandelt wurde (d. h., ob ihre Einträge in hot (aktuell) und cold (historisch) unterteilt sind, besteht darin, die Anzahl der Einträge zuerst mit der Einstellung "hot" und dann mit "cold" zu zählen. Wenn sich die zwei Werte **unterscheiden**, wurde die Tabelle dem Data Aging unterzogen:



Scripts im Hintergrund

Wenn ein Script im Hintergrund ausgeführt werden soll, bietet das entsprechende Popup-Fenster nun auch die oben genannten drei Auswahlknöpfe:

The screenshot shows the 'Script im Hintergrund starten' dialog box with the following settings:

- abweichender Jobname: SCRIPT_ \$AENDERUNGSBELEGE
- Job-Klasse: C
- Ausführungsziel: [Empty]
- Max. Trefferzahl: 9999
- Breite der Ausgabeliste: 1000
- Data-Aging-Einstellung: Hot, Cold, Datum
- Sprache für Zusatztexte: [Empty]
- Sekundärsprache: [Empty]
- Start sofort
- Start verzögert: Datum 29.03.2021, Uhrzeit 11:05:00
- mit Variante: [Empty]
- mit reduziertem Speicherbedarf
- Ergebnisliste in Server-Datei speichern
- Ergebnisliste zum SAP-Spool leiten
- Ergebnis-Teilnehmer: [Empty]

Dieses Script wird in der SE16XXL-Hintergrund-Übersicht wie folgt angezeigt:

Job	Gep1.Datum	Gep1.Zeit	Status	Period.	Benutzer	Sc.Type	Script	Glob	Priv	RSB	Script-Ben.	Erste Tax.	DAG	DAG	Anf.Da
<input type="checkbox"/> SCRIPT_ \$AENDERUNGSBELEGE	29.03.2021	11:05:00	Freigegeben		TOPFLOW		\$AENDERUNGSBELEGE	•				COHDR	<input checked="" type="checkbox"/>		

Beachten Sie die erste Spalte mit **DAG (Data-Aging-Einstellung (angefordert))**.

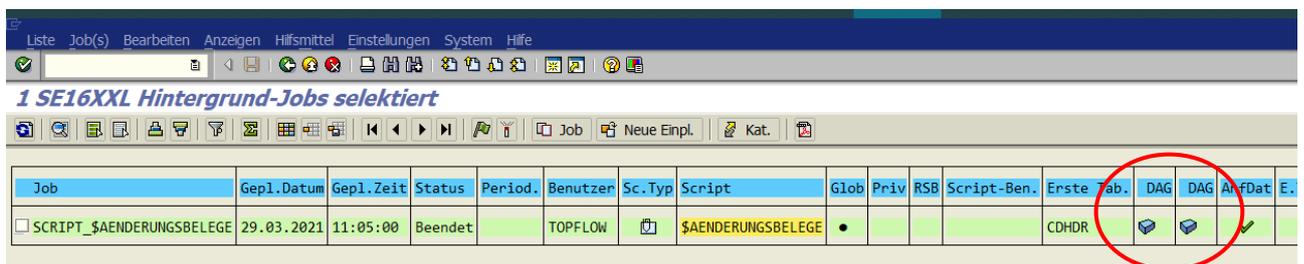
Folgenden Ikonen werden verwendet, um zu signalisieren, welche Einstellung angefordert wurde:

Nur hot Daten

 Sowohl hot als auch cold Daten

Hot plus cold Daten mit einem Datum \geq dem angegebenen Datum

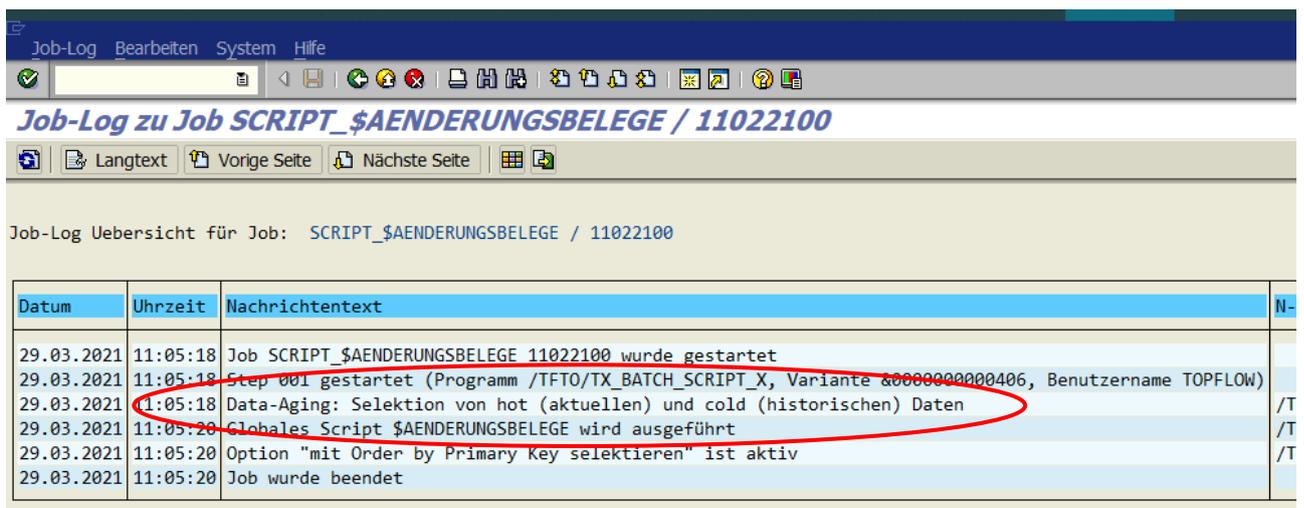
Wenn der Hintergrund-Job beendet ist, wird er wie folgt angezeigt:



Job	Gepf.Datum	Gepf.Zeit	Status	Period.	Benutzer	Sc.Type	Script	Glob	Priv	RSB	Script-Ben.	Erste Tab.	DAG	DAG	AnfDat	E.
<input type="checkbox"/> SCRIPT_ \$AENDERUNGSBELEGE	29.03.2021	11:05:00	Beendet		TOPFLOW		\$AENDERUNGSBELEGE	•				CDHDR				

Beachten Sie, dass nun auch die zweite **DAG**-Spalte (**Data-Aging-Einstellung (verwendet)**) gefüllt wurde. Falls nach einiger Zeit die Berechtigung dem Hintergrund-Benutzer entzogen wird, kann der Auftrag die angeforderte Data-Aging-Einstellung nicht erfüllen, und der Job selektiert nur hot Daten anstelle von hot und cold. Deshalb sind beide Spalten notwendig.

Das entsprechende Job-Log zeigt im Detail an, welche Einstellung verwendet wurde:



Datum	Uhrzeit	Nachrichtentext	N-
29.03.2021	11:05:18	Job SCRIPT_ \$AENDERUNGSBELEGE 11022100 wurde gestartet	
29.03.2021	11:05:18	Step 001 gestartet (Programm /TFTO/TX_BATCH_SCRIPT_X, Variante 0000000000406, Benutzername TOPFLOW)	
29.03.2021	11:05:18	Data-Aging: Selektion von hot (aktuellen) und cold (historischen) Daten	/T
29.03.2021	11:05:20	Globales Script \$AENDERUNGSBELEGE wird ausgeführt	/T
29.03.2021	11:05:20	Option "mit Order by Primary Key selektieren" ist aktiv	/T
29.03.2021	11:05:20	Job wurde beendet	

[Zum Anfang](#)

Spezielle Behandlung von Invertierten Datumsfeldern

Einige Datenbank-Tabellen, wie z. B. **TCURR** (Umrechnungskurse), enthalten ein invertiertes Datumsfeld (**TCURR-GDATU**). Solche Felder werden intern durch folgenden Wert dargestellt:

Invertiertes Datum = 99999999 – Datum.

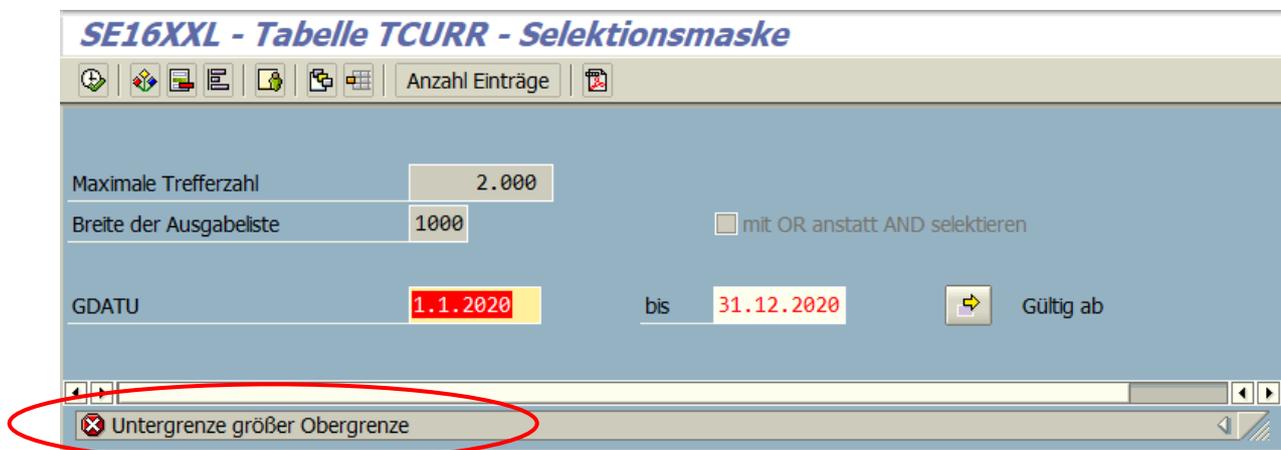
Zum Beispiel würde das Datum 25.3.2021 wie folgt dargestellt:

Invertiertes Datum = 99999999 – 20210325 = 79789674.

Mittels eines Konvertierungs-Exits wird der Wert extern als Datum dargestellt.

Diese Tabellen wurden zu einem Zeitpunkt definiert, zu dem es nicht möglich war, Sätze in absteigender Reihenfolge zu sortieren. Mithilfe eines invertierten Datums ist es möglich, durch Sortieren der internen Werte in aufsteigender Reihenfolge die entsprechenden externen Datumswerte in absteigender Reihenfolge zu sortieren.

Bisher wurden invertierte Datumsfelder in SE16XXL nicht sonderlich behandelt und durften daher nichtsahnende Benutzer mit ihrem seltsamen Verhalten verwirren. Ein solches merkwürdiges Verhalten kann auf einer Selektionsmaske beobachtet werden:



Wenn der Selektionswert als Intervall angegeben wird, meldet das System, dass die Untergrenze größer als die Obergrenze ist, obwohl das Intervall korrekt erscheint. Um das System zu befriedigen, muss das Intervall wie folgt angegeben werden:



Dieses verwirrende Verhalten ist auf die interne Darstellung von invertierten Datumsfeldern zurückzuführen.

Ein weiteres seltsames Verhalten kann beobachtet werden, wenn die Ergebnisliste nach dem invertierten Datumfeld **in aufsteigender Reihenfolge** sortiert wird:

SE16XXL - Tabelle TCURR - 3984 Einträge selektiert

Tabelle TCURR - Umrechnungskurse

MANDT	KURST	FCURR	TCURR	GDATU	UKURS	FFACT	TFACT	
<input type="checkbox"/>	800	EURB	USD	EUR	01.09.2020	/1,24170	1	1
<input type="checkbox"/>	800	EURG	USD	EUR	01.09.2020	/1,24110	1	1
<input type="checkbox"/>	800	EURX	USD	EUR	01.09.2020	/1,24140	1	1
<input type="checkbox"/>	800	EURX	ARS	EUR	31.08.2020	/3,55120	1	1
<input type="checkbox"/>	800	EURX	AUD	EUR	31.08.2020	/1,63280	1	1
<input type="checkbox"/>	800	EURX	BRL	EUR	31.08.2020	/2,90790	1	1
<input type="checkbox"/>	800	EURX	CHF	EUR	31.08.2020	/1,54855	1	1
<input type="checkbox"/>	800	EURX	CLP	EUR	31.08.2020	/668,80850	1	1
<input type="checkbox"/>	800	EURX	COP	EUR	31.08.2020	/2.815,39500	1	1
<input type="checkbox"/>	800	EURX	CZK	EUR	31.08.2020	/29,58800	1	1
<input type="checkbox"/>	800	EURX	GBP	EUR	31.08.2020	/68,35500	1	100

Anstatt in aufsteigender Reihenfolge erscheinen die Werte **in absteigender Reihenfolge**.

Außerdem, wenn die Konvertierungs-Exits deaktiviert wurden, werden die **internen Werte** angezeigt, Werte, die niemand leicht interpretieren kann:

SE16XXL - Tabelle TCURR - 3984 Einträge selektiert

Tabelle TCURR - Umrechnungskurse

MANDT	KURST	FCURR	TCURR	GDATU	UKURS	FFACT	TFACT	
<input type="checkbox"/>	800	EURB	USD	EUR	79799098	1,24170-	1	1
<input type="checkbox"/>	800	EURG	USD	EUR	79799098	1,24110-	1	1
<input type="checkbox"/>	800	EURX	USD	EUR	79799098	1,24140-	1	1
<input type="checkbox"/>	800	EURX	ARS	EUR	79799168	3,55120-	1	1
<input type="checkbox"/>	800	EURX	AUD	EUR	79799168	1,63280-	1	1
<input type="checkbox"/>	800	EURX	BRL	EUR	79799168	2,90790-	1	1
<input type="checkbox"/>	800	EURX	CHF	EUR	79799168	1,54855-	1	1
<input type="checkbox"/>	800	EURX	CLP	EUR	79799168	668,80850-	1	1
<input type="checkbox"/>	800	EURX	COP	EUR	79799168	2.815,39500-	1	1
<input type="checkbox"/>	800	EURX	CZK	EUR	79799168	29,58800-	1	1
<input type="checkbox"/>	800	EURX	GBP	EUR	79799168	68,35500-	1	100

Um diese Probleme zu vermeiden, behandelt SE16XXL nun invertierte Datumfelder auf besondere Weise, damit sie wie normale Datumfelder erscheinen. Diese Sonderbehandlung erfolgt nur auf externer Ebene – intern sind die Werte weiterhin invertiert. Dies ist für die Abwärtskompatibilität notwendig.

Auf den folgenden Seiten werden wir zeigen, wie sich invertierte Datumfelder jetzt verhalten.

Auf der Selektionsmaske kann das Intervall jetzt normal angegeben werden:

SE16XXL - Tabelle TCURR - Selektionsmaske

Anzahl Einträge

Maximale Trefferzahl

Breite der Ausgabeliste 1000 mit OR anstatt AND selektieren
 mit "Order by Primary Key" selektieren

GDATU 1.1.2020 bis 31.12.2020 Gültig ab

Die Ergebnisliste zeigt nun immer das normale Datum an und das Sortierverhalten ist ebenfalls normal:

SE16XXL - Tabelle TCURR - 3985 Einträge selektiert

Tabelle TCURR - Umrechnungskurse

MANDT	KURST	FCURR	TCURR	GDATU	UKURS	FFACT	TFACT
<input type="checkbox"/> 800	B	AUD	EUR	01.01.2020	0,62490	0	0
<input type="checkbox"/> 800	B	AUD	GBP	01.01.2020	0,42800	0	0
<input type="checkbox"/> 800	B	AUD	NZD	01.01.2020	1,08510	0	0
<input type="checkbox"/> 800	B	AUD	USD	01.01.2020	0,76060	0	0
<input type="checkbox"/> 800	B	NZD	EUR	01.01.2020	0,57660	0	0
<input type="checkbox"/> 800	B	NZD	GBP	01.01.2020	0,39430	0	0
<input type="checkbox"/> 800	B	NZD	USD	01.01.2020	0,70050	0	0
<input type="checkbox"/> 800	EURB	JPY	EUR	01.01.2020	138,94500-	0	0
<input type="checkbox"/> 800	EURB	USD	EUR	01.01.2020	1,35660	0	0

In der Detailansicht eines Satzes wird das invertierte Datum weiterhin unkonvertiert angezeigt:

Satz von Tabelle TCURR - Umrechnungskurse

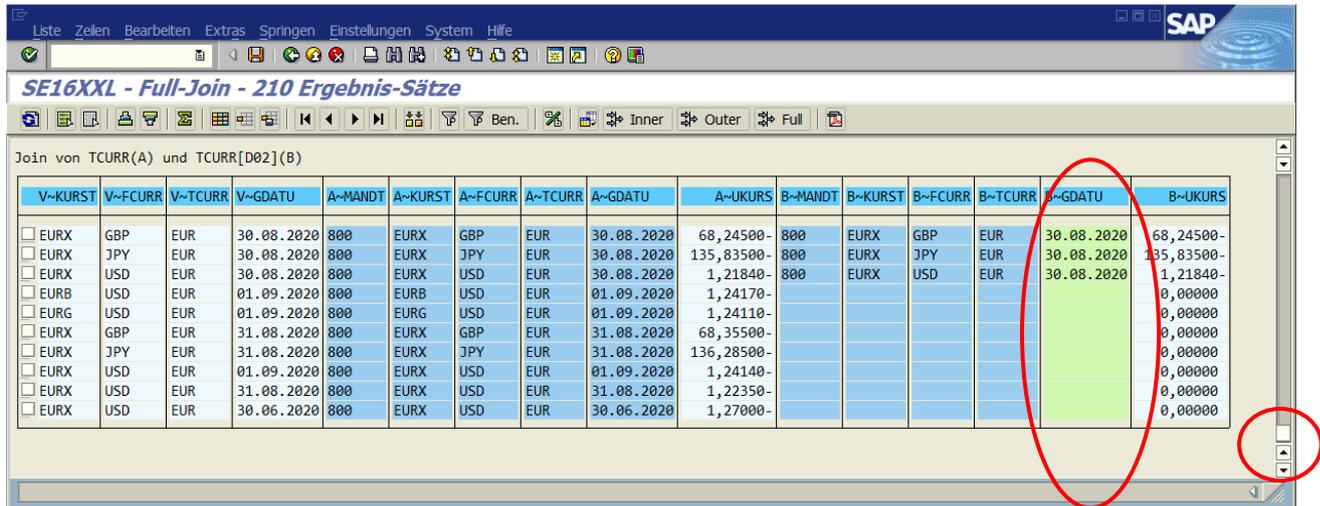
Voriger Eintrag Nächster Eintrag

Satz von Tabelle TCURR - Umrechnungskurse

MANDT	'800'	Mandant
KURST	'B'	Kurstyp
FCURR	'AUD'	Von-Währung
TCURR	'EUR'	Nach-Währung
GDATU	'79799898'	Datum, ab dem der Kurs gültig ist
UKURS	'01.01.2020'	Umrechnungskurs
FFACT	'0'	Faktor für die Einheiten der Von-Währung
TFACT	'0'	Faktor für die Einheiten der Nach-Währung

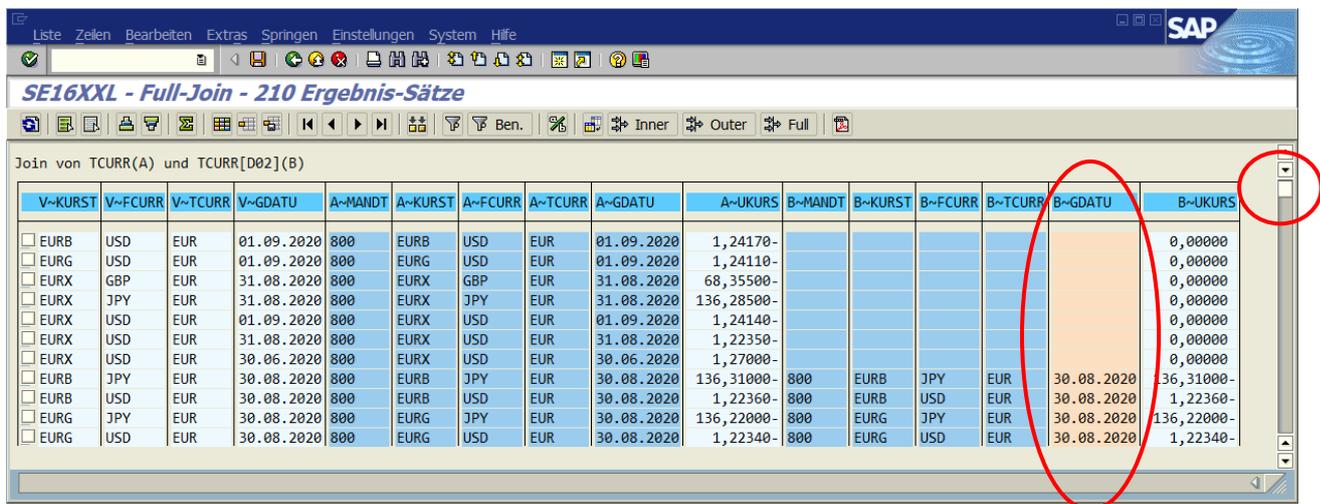
Allerdings wird das entsprechende Datum als **Quickinfo** angezeigt.

Es gibt nur eine Situation, in der sich ein invertiertes Datum nicht wie ein normales Datumsfeld verhält. Dies geschieht, wenn **Initialwerte** in der Ergebnisliste vorhanden sind, z. B. bei einem Full-Join, und die Liste nach dieser Spalte **sortiert** wird:



V~KURST	V~FCURR	V~TCURR	V~GDATU	A~MANDT	A~KURST	A~FCURR	A~TCURR	A~GDATU	A~UKURS	B~MANDT	B~KURST	B~FCURR	B~TCURR	B~GDATU	B~UKURS	
<input type="checkbox"/>	EURX	GBP	EUR	30.08.2020	800	EURX	GBP	EUR	30.08.2020	68,24500-	800	EURX	GBP	EUR	30.08.2020	68,24500-
<input type="checkbox"/>	EURX	JPY	EUR	30.08.2020	800	EURX	JPY	EUR	30.08.2020	135,83500-	800	EURX	JPY	EUR	30.08.2020	135,83500-
<input type="checkbox"/>	EURX	USD	EUR	30.08.2020	800	EURX	USD	EUR	30.08.2020	1,21840-	800	EURX	USD	EUR	30.08.2020	1,21840-
<input type="checkbox"/>	EURB	USD	EUR	01.09.2020	800	EURB	USD	EUR	01.09.2020	1,24170-					0,00000	
<input type="checkbox"/>	EURG	USD	EUR	01.09.2020	800	EURG	USD	EUR	01.09.2020	1,24110-					0,00000	
<input type="checkbox"/>	EURX	GBP	EUR	31.08.2020	800	EURX	GBP	EUR	31.08.2020	68,35500-					0,00000	
<input type="checkbox"/>	EURX	JPY	EUR	31.08.2020	800	EURX	JPY	EUR	31.08.2020	136,28500-					0,00000	
<input type="checkbox"/>	EURX	USD	EUR	01.09.2020	800	EURX	USD	EUR	01.09.2020	1,24140-					0,00000	
<input type="checkbox"/>	EURX	USD	EUR	31.08.2020	800	EURX	USD	EUR	31.08.2020	1,22350-					0,00000	
<input type="checkbox"/>	EURX	USD	EUR	30.06.2020	800	EURX	USD	EUR	30.06.2020	1,27000-					0,00000	

In diesem Fall erscheinen die Initialwerte, anstatt am Anfang, **am Ende der Liste**. In ähnlicher Weise, wenn die Liste in absteigender Reihenfolge sortiert ist, werden die Initialwerte **am Anfang** und nicht am Ende der Ergebnisliste angezeigt:



V~KURST	V~FCURR	V~TCURR	V~GDATU	A~MANDT	A~KURST	A~FCURR	A~TCURR	A~GDATU	A~UKURS	B~MANDT	B~KURST	B~FCURR	B~TCURR	B~GDATU	B~UKURS	
<input type="checkbox"/>	EURB	USD	EUR	01.09.2020	800	EURB	USD	EUR	01.09.2020	1,24170-					0,00000	
<input type="checkbox"/>	EURG	USD	EUR	01.09.2020	800	EURG	USD	EUR	01.09.2020	1,24110-					0,00000	
<input type="checkbox"/>	EURX	GBP	EUR	31.08.2020	800	EURX	GBP	EUR	31.08.2020	68,35500-					0,00000	
<input type="checkbox"/>	EURX	JPY	EUR	31.08.2020	800	EURX	JPY	EUR	31.08.2020	136,28500-					0,00000	
<input type="checkbox"/>	EURX	USD	EUR	01.09.2020	800	EURX	USD	EUR	01.09.2020	1,24140-					0,00000	
<input type="checkbox"/>	EURX	USD	EUR	31.08.2020	800	EURX	USD	EUR	31.08.2020	1,22350-					0,00000	
<input type="checkbox"/>	EURX	USD	EUR	30.06.2020	800	EURX	USD	EUR	30.06.2020	1,27000-					0,00000	
<input type="checkbox"/>	EURB	JPY	EUR	30.08.2020	800	EURB	JPY	EUR	30.08.2020	136,31000-	800	EURB	JPY	EUR	30.08.2020	36,31000-
<input type="checkbox"/>	EURB	USD	EUR	30.08.2020	800	EURB	USD	EUR	30.08.2020	1,22360-	800	EURB	USD	EUR	30.08.2020	1,22360-
<input type="checkbox"/>	EURG	JPY	EUR	30.08.2020	800	EURG	JPY	EUR	30.08.2020	136,22000-	800	EURG	JPY	EUR	30.08.2020	136,22000-
<input type="checkbox"/>	EURG	USD	EUR	30.08.2020	800	EURG	USD	EUR	30.08.2020	1,22340-	800	EURG	USD	EUR	30.08.2020	1,22340-

Dieses seltsame Verhalten ist auf die Tatsache zurückzuführen, dass die Initialwerte mit Leerzeichen gefüllt sind, die niedrigsten invertierten Datumsfelder jedoch Zahlen sind, die von 999999999 abwärts beginnen.

[Zum Anfang](#)

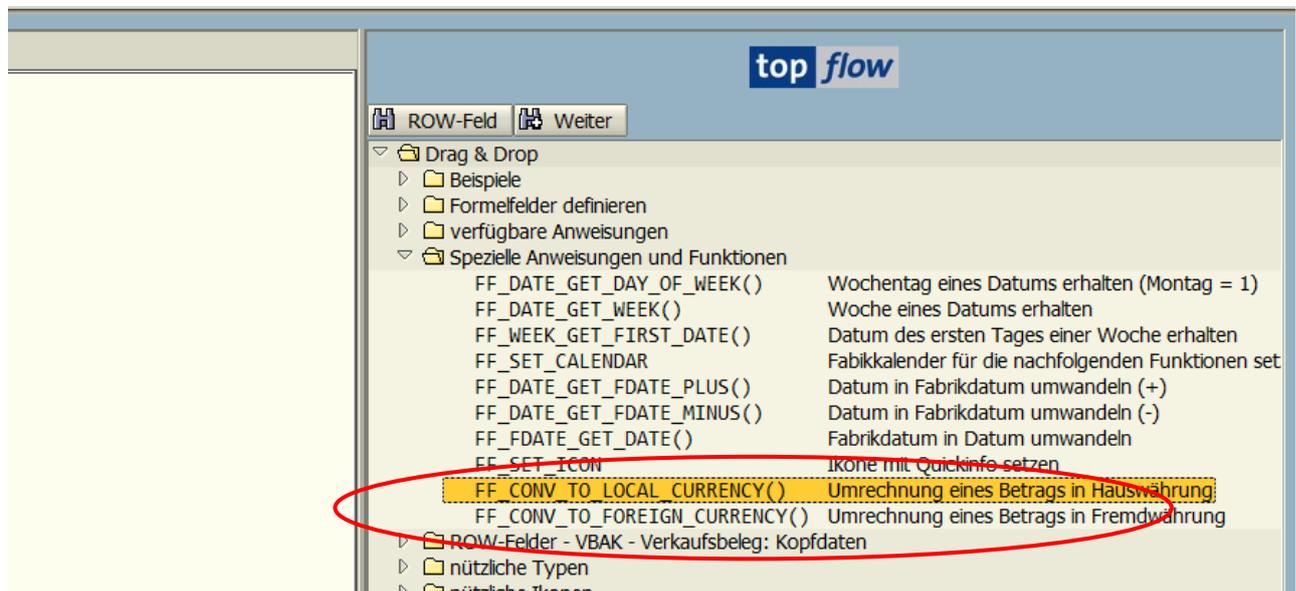
Währungsumrechnungen in Formeln

In einer SE16XXL-Formel ist es nun möglich, Währungsumrechnungen durchzuführen. Diese Funktionalität basiert auf folgenden Funktionsbausteinen:

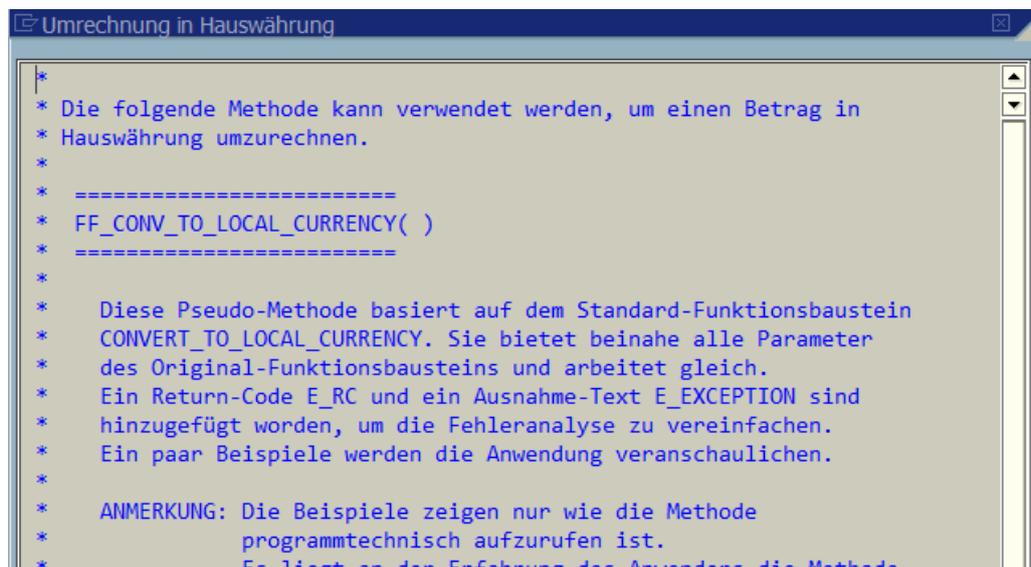
CONVERT_TO_LOCAL_CURRENCY

CONVERT_TO_FOREIGN_CURRENCY

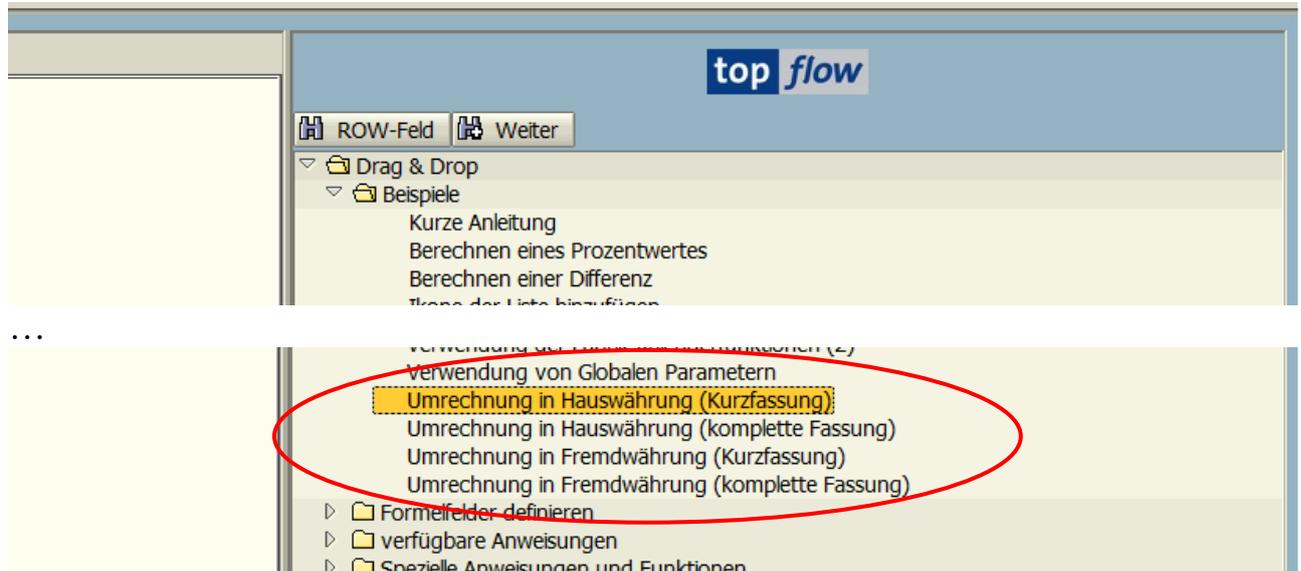
Im Formeleditor sind die entsprechenden Pseudo-Methoden rechts unter „Spezielle Anweisungen und Funktionen“ zu finden:



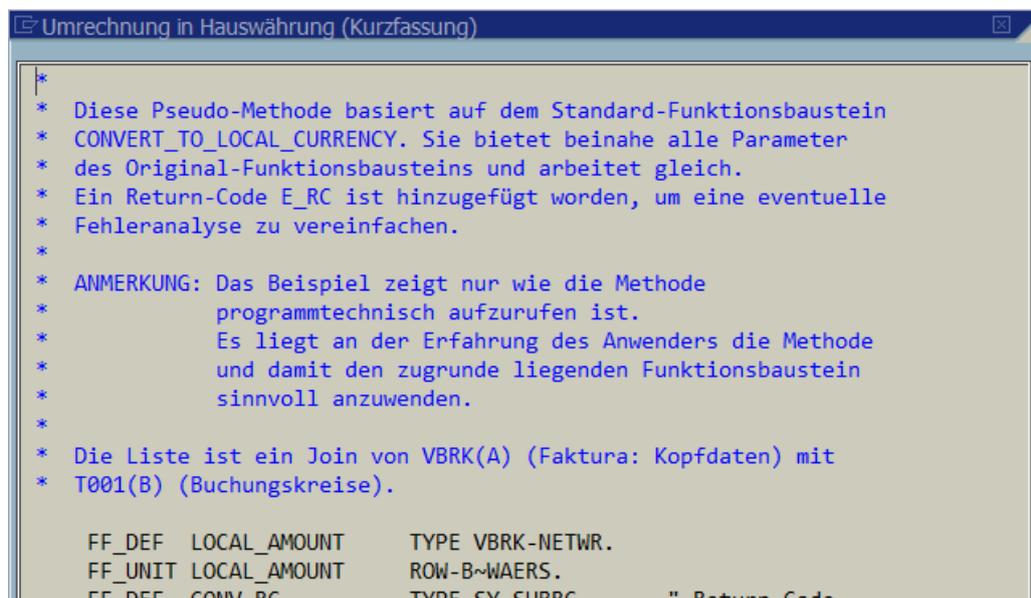
Durch Doppelklick auf einen Knoten erhalten Sie die zugehörige Dokumentation:



Unter den Beispielen gibt es auch einige in Bezug auf die Währungsumrechnungen:



Mit einem Doppelklick können Sie das Beispiel-Coding im Detail anschauen:



Um sich mit diesen beiden Pseudo-Methoden vertraut zu machen, kann es hilfreich sein, ein Beispiel per **Drag & Drop** vom Baum rechts in den Editorbereich auf der linken Seite zu kopieren und dann zu beobachten, wie sich die Formel verhält.

[Zum Anfang](#)

Upload von Frontend-Dateien vom Typ .XLS, .XLSX usw.

Ab SE16XXL Version 3.5A ist es möglich, eine Datei vom Typ .XLS, .XLSX, .XML oder .MHTML hochzuladen (als Vorgabedatei oder Frontend-Datei).

Anhand eines kurzen Beispiels wird diese neue Option erläutert.

Nach dem Upload einer Datei von einem der oben genannten Typen erhalten wir das Definitions-Fenster:

D: | SE16XXL | BEISPIELE | VBAK.xlsx

Akzeptieren
 Kopfzeile auswählen
 Definition importieren

Trennzeichen: ht Tabulator Erste relevante Zeile: 1
 Datumsformat: 3 - MM/TT/JJJJ Dezimalpunkt: . Punkt
 Sprache: DE Deutsch

Sel.	Pos.	Name	DDIC	Dtel	Typ	KB	Länge	Dez.	Vorz.	F.N	Tabellenname	Feldname	Datenelement	Kurztext	Langtext	Währ./Einh.
<input checked="" type="checkbox"/>	1	COL1	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	5	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 1	Spalte 1	
<input checked="" type="checkbox"/>	2	COL2	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	10	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 2	Spalte 2	
<input checked="" type="checkbox"/>	3	COL3	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	10	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 3	Spalte 3	
<input checked="" type="checkbox"/>	4	COL4	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	5	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 4	Spalte 4	
<input checked="" type="checkbox"/>	5	COL5	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	5	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 5	Spalte 5	
<input checked="" type="checkbox"/>	6	COL6	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	10	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 6	Spalte 6	
<input checked="" type="checkbox"/>	7	COL7	<input type="checkbox"/>	<input type="checkbox"/>	CHAR	<input type="checkbox"/>	5	0	<input type="checkbox"/>	<input type="checkbox"/>				Sp. 7	Spalte 7	

COL1	COL2	COL3	COL4	COL5	COL6	COL7	
0001	MANDT	VBELN	ERDAT	VBTYP	AUART	NETWR	WAERK
0002	800	0000005178	9/5/1997	C	TA	102,546,01	USD
0003	800	0000005205	9/26/1997	C	TA	1,759,18	USD
0004	800	0000005207	10/9/1997	C	TA	92,330,70	USD
0005	800	0000005244	11/6/1997	C	TA	29,723,56	USD
0006	800	0000005289	12/10/1997	C	TA	133,696,50	USD
0007	800	0000005308	1/9/1998	C	TA	45,019,37	USD

Nachdem diese Art von Datei nicht mit Metadaten versehen werden kann, müssen wir die Definition manuell angeben:

D: | SE16XXL | BEISPIELE | VBAK.xlsx

Akzeptieren
 Kopfzeile auswählen
 Definition importieren

Trennzeichen: ht Tabulator Erste relevante Zeile: 2
 Datumsformat: 3 - MM/TT/JJJJ Dezimalpunkt: . Punkt
 Sprache: DE Deutsch

Sel.	Pos.	Name	DDIC	Dtel	Typ	KB	Länge	Dez.	Vorz.	F.N	Tabellenname	Feldname	Datenelement	Kurztext	Langtext	Währ./Einh.
<input checked="" type="checkbox"/>	1	MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CLNT	<input type="checkbox"/>	3	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	MANDT	MANDT	Mandant	Mandant	
<input checked="" type="checkbox"/>	2	VBELN	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	<input type="checkbox"/>	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	VBAK	VBELN	VBELN_VA	Verkaufsb.	Verkaufsbeleg	
<input checked="" type="checkbox"/>	3	ERDAT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DATS	<input type="checkbox"/>	8	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	ERDAT	ERDAT	Am	Angelegt am	
<input checked="" type="checkbox"/>	4	VBTYP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	<input checked="" type="checkbox"/>	1	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	VBTYP	VBTYP	V-Belegtyp	Vertriebsbelegtyp	
<input checked="" type="checkbox"/>	5	AUART	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	<input type="checkbox"/>	4	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	AUART	AUART	V.belegart	Verkaufsbelegart	
<input checked="" type="checkbox"/>	6	NETWR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CURR	<input type="checkbox"/>	15	2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	VBAK	NETWR	NETWR_AK	Nettowert	Nettowert	WAERK
<input checked="" type="checkbox"/>	7	WAERK	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CUKY	<input type="checkbox"/>	5	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	WAERK	WAERK	Währung	Belegwährung	

MANDT	VBELN	ERDAT	VBTYP	AUART	NETWR	WAERK	
0001	MANDT#VBELN#ERDAT#VBTYP#AUART#NETWR#WAERK						
0002	800	0000005178	19970905	C	TA	102.546,01	USD
0003	800	0000005205	19970926	C	TA	1.759,18	USD
0004	800	0000005207	19971009	C	TA	92.330,70	USD
0005	800	0000005244	19971106	C	TA	29.723,56	USD
0006	800	0000005289	19971210	C	TA	133.696,50	USD
0007	800	0000005308	19980109	C	TA	45.019,37	USD

Es ist besonders darauf zu achten, dass das richtige Datums- und Dezimalpunkt-Format gewählt wird:

D: \SE16XXL \BEISPIELE \VBAK.xlsx

Akzeptieren
 Kopfzeile auswählen
 Definition importieren

Trennzeichen: ht Tabulator Erste relevante Zeile: 2
 Datumsformat: 3 - MM/TT/JJJJ Dezimalpunkt: . Punkt
 Sprache: DE Deutsch

Sel.	Pos.	Name	DDIC	Dtel	Typ	KB	Länge	Dez.	Vorz.	F.N	Tabellenname	Feldname
<input checked="" type="checkbox"/>	1	MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CLNT	<input type="checkbox"/>	3	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	MANDT
<input checked="" type="checkbox"/>	2	VBELN	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CHAR	<input type="checkbox"/>	10	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	VBAK	VBELN
<input checked="" type="checkbox"/>	3	FRDAT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	DATS	<input type="checkbox"/>	8	0	<input type="checkbox"/>	<input type="checkbox"/>	VBAK	FRDAT

Schließlich wird die Ergebnisliste in der üblichen Weise angezeigt:

SE16XXL - Vorgabedatei - 2000 Einträge

Vorgabedatei - D: \SE16XXL \BEISPIELE \VBAK.xlsx

MANDT	VBELN	ERDAT	VBTYP	AUART	NETWR	WAERK
<input type="checkbox"/> 800	0000005178	05.09.1997	C	TA	102.546,01	USD
<input type="checkbox"/> 800	0000005205	26.09.1997	C	TA	1.759,18	USD
<input type="checkbox"/> 800	0000005207	09.10.1997	C	TA	92.330,70	USD
<input type="checkbox"/> 800	0000005244	06.11.1997	C	TA	29.723,56	USD
<input type="checkbox"/> 800	0000005289	10.12.1997	C	TA	133.696,50	USD
<input type="checkbox"/> 800	0000005308	00.01.1998	C	TA	15.010,37	USD

Nun sichern wir unsere Arbeit als Script:

Als Script sichern

Script-Name: BEISPIEL_M_VORGABEDATEI global privat
 Beschreibung: Beispiel-Script mit XLSX Vorgabedatei

OK - Script BEISPIEL_M_VORGABEDATEI gesichert

An dieser Stelle führen wir unser neu gespeichertes Script aus. Wir wählen jedoch eine andere Vorgabedatei mit gleicher Struktur, aber von einem anderen Typ.

Als Ergebnis erhalten wir:

Script BEISPIEL_M_VORGABEDATEI - 2000 Einträge

Vorgabedatei - D:\SE16XXL\BEISPIELE\VBAK.mhtml
Script: BEISPIEL_M_VORGABEDATEI

MANDT	VBELN	ERDAT	VBTYP	AUART	NETWR	WAERK
<input type="checkbox"/> 800	0000005178	05.09.1997	C	TA	102.546,01	USD
<input type="checkbox"/> 800	0000005205	26.09.1997	C	TA	1.759,18	USD
<input type="checkbox"/> 800	0000005207	09.10.1997	C	TA	92.330,70	USD
<input type="checkbox"/> 800	0000005244	06.11.1997	C	TA	29.723,56	USD
<input type="checkbox"/> 800	0000005289	10.12.1997	C	TA	133.696,50	USD
<input type="checkbox"/> 800	0000005308	09.01.1998	C	TA	15.019,37	USD

Dieses Beispiel zeigt, dass man einen beliebigen Typ von Vorgabedatei auswählen kann, solange sie die vom Script erwartete Spaltenstruktur aufweist.

Nun ein paar Worte zur Implementierung.

Um den Inhalt einer Frontend-Datei eines der oben genannten Typen einzulesen, verwendet das Programm die **OLE2-Schnittstelle**. Es wird ein OLE-Objekt des Typs “*Excel.Application*” erstellt und die ausgewählte Datei geöffnet. Der Inhalt des **ersten Blattes** der Datei (es könnten mehrere sein) wird in einer provisorischen Datei vom Typ “**Unicode-Text (*.txt)**” gespeichert. Dadurch wird sichergestellt, dass alle möglichen Zeichen, z. B. japanische oder kyrillische Zeichen, korrekt übernommen werden. Zum Schluss wird die Originaldatei geschlossen und die provisorische Datei in der üblichen Weise hochgeladen, d.h. als Textdatei mit horizontalen Tabs als Trennzeichen. Die provisorische Datei wird im SAPGUI-Arbeitsverzeichnis mit einem speziellen Namen erstellt, der eine 32-stellige GUID enthält, um die Eindeutigkeit zu gewährleisten. Am Ende der Verarbeitung wird die provisorische Datei gelöscht.

Es wurde zunächst versucht, eine einfachere Lösung mithilfe der Zwischenablage zu implementieren. Aufgrund von sporadisch auftretenden Problemen wurde dieser Ansatz jedoch zugunsten der oben beschriebenen Logik aufgegeben, die robuster zu sein scheint.

[Zum Anfang](#)

Neue Option für Server-Datei als E-Mail-Anhang

Bisher war es in einem im Hintergrund laufenden Script möglich, eine Server-Datei zu erstellen und als E-Mail-Anhang an eine Reihe von E-Mail-Empfängern zu versenden. Allerdings mussten die Empfänger **von vornherein** bekannt sein, d.h. entweder in Form einer Verteilerliste oder direkt in der Hintergrund-Anforderung angegeben werden. Dieses Verfahren ist in der Regel ausreichend. Es gibt jedoch Situationen, die einen anderen Ansatz erfordern. Dies ist der Fall, wenn die zu erstellende Server-Datei Informationen enthält, die von einem Lauf zum nächsten unterschiedliche Empfänger betreffen. Wenn z.B. die Server-Datei eine Liste von Produkten enthält, deren Status von “in Entwicklung” in “frei verwendbar” geändert werden muss, kommen nur die verantwortlichen Personen als Empfänger der Datei infrage.

Die Produkte in der Liste variieren von einem Lauf zum nächsten. Es ist daher **nicht möglich**, die richtigen Empfänger im Voraus anzugeben.

Um diese Lücke zu schließen, steht jetzt eine neue Option zur Verfügung.

In diesem Fall muss die Ergebnisliste **in einer bestimmten Spalte die E-Mail-Adresse des Empfängers** enthalten. Alle E-Mail-Adressen in der Liste werden gesammelt. Die resultierende Server-Datei kann dann entweder **an jede in der Liste gefundene E-Mail-Adresse** versendet werden, oder es kann festgelegt werden, **jedem Empfänger nur die Einträge der Liste zu senden, die die entsprechende E-Mail-Adresse beinhalten**.

Ein paar Beispiele werden diese neue Funktionalität veranschaulichen.

Das Script `$AUFTRAEGE_MIT_EMAIL_ADR` erstellt eine Liste von Kundenaufträgen. Jede Zeile enthält die E-Mail-Adresse der Person, die den Auftrag erstellt hat:

SE16XXL - Inner-Join - 139 Ergebnis-Sätze

Join von VBAK(A) USR21(B) und ADR6(C)

A~VBELN	A~VBTYPE	A~AUART	A~NETWR	A~WAERK	A~VKORG	A~VTWEG	A~SPART	A~ERNAM	C~SMTP_ADDR
<input type="checkbox"/> 0000011892	C	ZSAA	20,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0000011893	C	ZSAA	10,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0000011933	C	ZSAA	20,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0060000081	H	ZSEA	200,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0060000082	H	ZSEA	200,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0060000083	H	ZSEA	200,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0060000084	H	ZSEA	200,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0060000085	H	ZSEA	200,00	USD	3000	10	00	C5055253	joris.sikking@sap.com
<input type="checkbox"/> 0000005205	C	TA	1.759,18	USD	1000	12	00	CURA	sergio.marcelo.cura.daball@sap.com
<input type="checkbox"/> 0000011934	C	TA	3,90	EUR	1000	10	00	D027220	Hans-Joerg.Kersten@sap.com
<input type="checkbox"/> 0000011935	C	TA	3,90	EUR	1000	10	00	D027220	Hans-Joerg.Kersten@sap.com
<input type="checkbox"/> 0000011936	C	TA	3,90	EUR	1000	10	00	D027220	Hans-Joerg.Kersten@sap.com
<input type="checkbox"/> 0000011941	C	VPO	3,90	EUR	1000	10	00	D027220	Hans-Joerg.Kersten@sap.com

Diese unspektakuläre Liste wurde nur gewählt, um zu zeigen, wie die neue Option funktioniert.

Nun soll dieses Script im Hintergrund ausgeführt werden. Das Ergebnis soll in einer **Server-Datei** gespeichert werden, die als E-Mail-Anhang an alle Personen gesendet werden soll, deren E-Mail-Adresse in der Ergebnisliste zu finden ist:

Script im Hintergrund starten

abweichender Jobname SCRIPT_\$AUFTRAEGE_MIT_EMAIL_ADR

Job-Klasse C Ausführungsziel

Max. Trefferzahl

...

mit reduziertem Speicherbedarf

Ergebnisliste in Server-Datei speichern E-Mail-Parameter

Ergebnisliste zum SAP-Spool leiten

keine Benachrichtigung via SAPMail

✓ 🗑️ ✖

Jetzt betätigen wir die E-Mail-Parameter Schaltfläche, um die E-Mail und ihre Empfänger anzugeben. Auf dem Popup-Fenster ist die neue Option bereits ausgewählt:

E-Mail-Parameter

Empfänger

Verteilerliste Allgemeine Verteilerliste

Liste von E-Mail-Adressen

Spalte der Ergebnisliste C~SMTP_ADDR Test-Modus

Optionen

Leere Hintergrund-Ergebnisse ignorieren Nur relevante Sätze zu den Empfängern versenden

Betreff Kundenaufträge

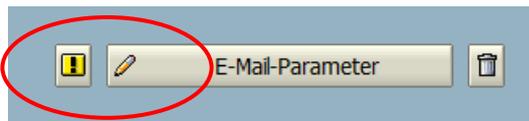
ANMERKUNG: Die Spalte mit der E-Mail-Adresse muss nicht nur in der Liste vorhanden, sondern auch **sichtbar sein**, d. h. zu den Ausgabefeldern zählen. Dies ist notwendig, denn sonst können die Empfänger der E-Mail die für sie relevanten Datensätze nicht erkennen.

Um zu testen, ob das Ganze funktioniert, wurde eine spezielle Test-Option eingeführt:

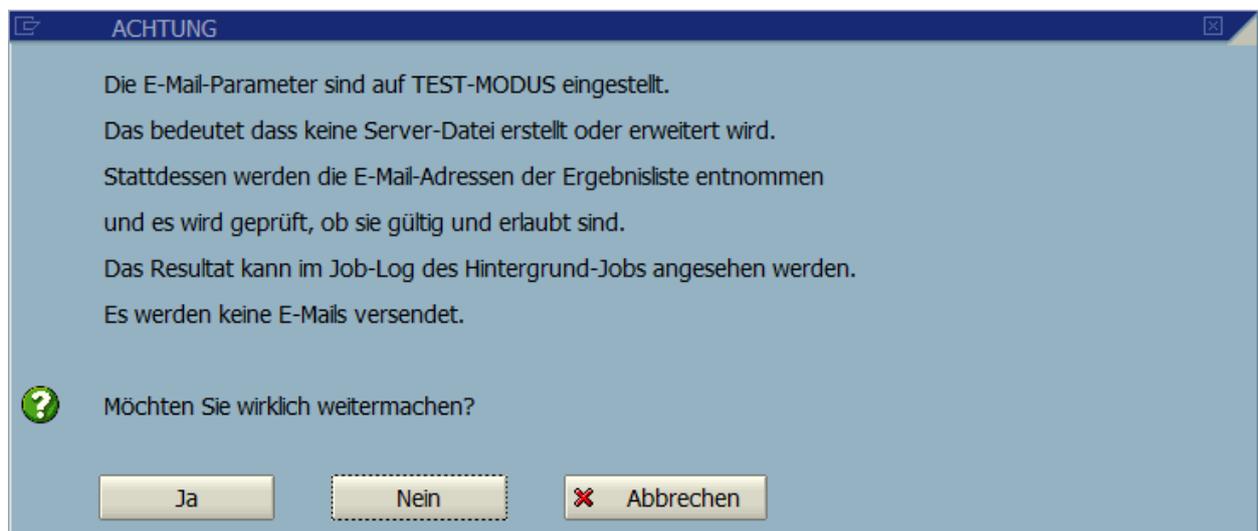


Wenn dieses Flag aktiviert wird, wird keine Server-Datei erstellt, überschrieben oder erweitert. Stattdessen werden die E-Mail-Adressen aus der angegebenen Spalte der Ergebnisliste auf Gültigkeit und Berechtigung überprüft.

Zurück zum Haupt-Popup-Fenster wird die oben genannte Schaltfläche durch eine **Warn-Ikone** flankiert, um zu signalisieren, dass der Test-Modus aktiviert wurde:



Wenn das Script schließlich eingeplant wird, wird folgende Warnung ausgegeben:



Falls Sie mit Ja antworten, wird der Job eingeplant. Das resultierende Job-Log informiert Sie über das Ergebnis des Tests. Auf der folgenden Seite wird ein solches Job-Log angezeigt.

Job-Log zu Job SCRIPT_ \$AUFTRAEGE_MIT_EMAIL_ADR / 16490600

Langtext Vorige Seite Nächste Seite

Job-Log Uebersicht für Job: SCRIPT_ \$AUFTRAEGE_MIT_EMAIL_ADR / 16490600

Datum	Uhrzeit	Nachrichtentext	N-Klasse	N-Nummer	N-Typ
29.03.2021	16:49:06	Job wurde gestartet	00	516	S
29.03.2021	16:49:06	Step 001 gestartet (Programm /TFTO/TX_BATCH_SCRIPT_X, Variante &000000004844, Benutzername TOPFLOW)	00	550	S
29.03.2021	16:49:07	Globales Script \$AUFTRAEGE_MIT_EMAIL_ADR wird ausgeführt	/TFTO/TX	750	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	***** Test-Modus: keine Server-Datei erstell *****	/TFTO/TXBT	180	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	***** TEST MODUS *****	/TFTO/TXBT	175	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: joris.sikking@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: sergio.marcelo.cura.daball@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: Hans-Joerg.Kersten@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: priyangshu.kakati@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: karen.lindholm@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: ides@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: susan.pitten@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: elizabeth.stoddard@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	Gültige E-Mail-Adresse: james.thomas@sap.com	/TFTO/TXBT	174	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	Anzahl gültiger E-Mail-Adressen: 9	/TFTO/TXBT	176	S
29.03.2021	16:49:07	Anzahl ungültiger E-Mail-Adressen: 0	/TFTO/TXBT	177	S
29.03.2021	16:49:07	Anzahl nicht erlaubter E-Mail-Adressen: 0	/TFTO/TXBT	178	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	***** TEST MODUS *****	/TFTO/TXBT	175	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	*****	/TFTO/TXBT	998	S
29.03.2021	16:49:07	Job wurde beendet	00	517	S

Im Echtfall hingegen würde das Job-Log wie folgt aussehen:

Job-Log zu Job SCRIPT_ \$AUFTRAEGE_MIT_EMAIL_ADR / 16514900

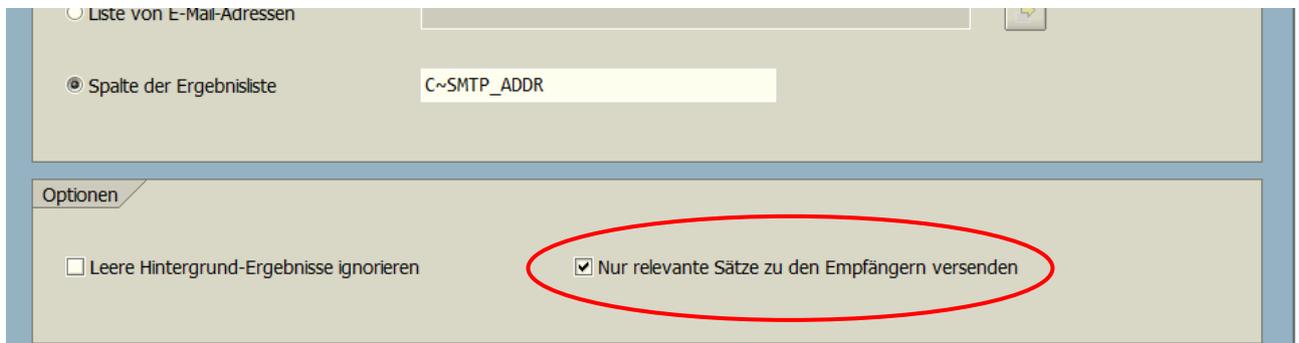
Langtext Vorige Seite Nächste Seite

Job-Log Uebersicht für Job: SCRIPT_ \$AUFTRAEGE_MIT_EMAIL_ADR / 16514900

Datum	Uhrzeit	Nachrichtentext	N-Klasse	N-Nummer	N-Typ
29.03.2021	16:51:49	Job wurde gestartet	00	516	S
29.03.2021	16:51:49	Step 001 gestartet (Programm /TFTO/TX_BATCH_SCRIPT_X, Variante &000000004845, Benutzername TOPFLOW)	00	550	S
29.03.2021	16:51:52	Ziel-Server-Datei-Pfad ist \usr\sap\ZE5\tmp	/TFTO/TX2	026	S
29.03.2021	16:51:52	Globales Script \$AUFTRAEGE_MIT_EMAIL_ADR wird ausgeführt	/TFTO/TX	750	S
29.03.2021	16:51:52	Datei \usr\sap\ZE5\tmp\Kundenaufraege.txt geschrieben	/TFTO/TX	621	S
29.03.2021	16:51:52	*****	/TFTO/TXBT	998	S
29.03.2021	16:51:52	Anzahl gültiger E-Mail-Adressen: 9	/TFTO/TXBT	176	S
29.03.2021	16:51:52	Anzahl ungültiger E-Mail-Adressen: 0	/TFTO/TXBT	177	S
29.03.2021	16:51:52	Anzahl nicht erlaubter E-Mail-Adressen: 0	/TFTO/TXBT	178	S
29.03.2021	16:51:52	*****	/TFTO/TXBT	998	S
29.03.2021	16:51:52	Die E-Mail-Adressen sind der Ergebnisliste entnommen worden	/TFTO/TXBT	173	S
29.03.2021	16:51:52	E-Mails an 9 Empfängern versendet	/TFTO/TXBT	134	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: joris.sikking@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: sergio.marcelo.cura.daball@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: Hans-Joerg.Kersten@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: priyangshu.kakati@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: karen.lindholm@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: ides@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: susan.pitten@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: elizabeth.stoddard@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	E-Mail mit Anhang gesendet an: james.thomas@sap.com	/TFTO/TXBT	139	S
29.03.2021	16:51:52	Job wurde beendet	00	517	S

In diesem Beispiel wurde dieselbe Server-Datei mit der kompletten Ergebnisliste als Anhang an alle beteiligten Empfänger versendet. Jeder muss dann seine eigenen relevanten Einträge aus der Liste herausuchen.

Es gibt jedoch eine bessere Lösung. Im E-Mail-Popup-Fenster kann festgelegt werden, dass jeder Empfänger nur die relevanten Sätze erhalten soll:



In diesem Fall erstellt das Programm zusätzlich zur kompletten Server-Datei **eine spezielle Server-Datei für jeden Empfänger**. Die Dateien werden separat an jede E-Mail-Adresse gesendet und anschließend gelöscht. Um Namenskonflikte zu vermeiden, werden die Dateinamen mithilfe einer 32-stelligen GUID aufgebaut.

Das resultierende Job-Log informiert über diese Vorgehensweise:

Job-Log zu Job SCRIPT_ \$AUFTRAEGE_MIT_EMAIL_ADR / 16552000

Langtext | Vorige Seite | Nächste Seite

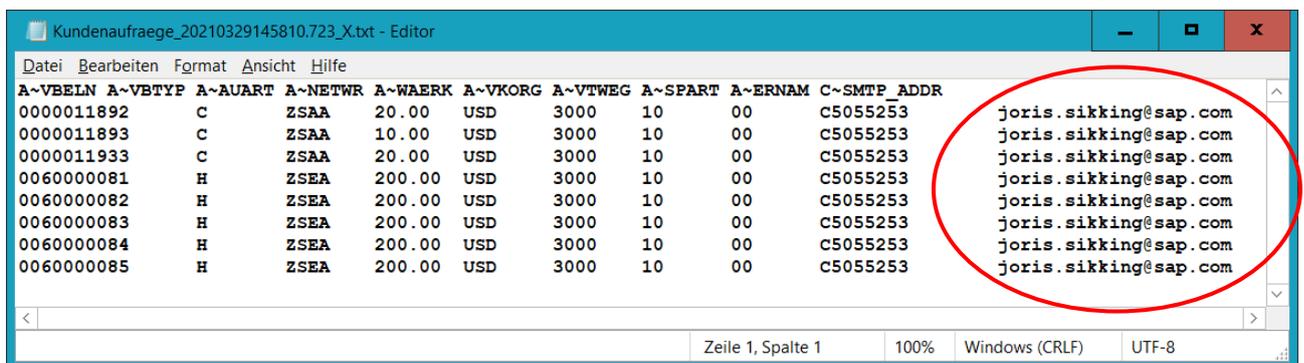
Job-Log Uebersicht für Job: SCRIPT_ \$AUFTRAEGE_MIT_EMAIL_ADR / 16552000

Datum	Uhrzeit	Nachrichtentext	N-Klasse	N-Nummer	N-Typ
29.03.2021	16:55:20	Job wurde gestartet	00	516	S
29.03.2021	16:55:20	Step 001 gestartet (Programm /TFTO/TX_BATCH_SCRIPT_X, Variante &0000000004846, Benutzername TOPFLOW)	00	550	S
29.03.2021	16:55:22	Ziel-Server-Datei-Pfad ist \usr\sap\ZES\tmp	/TFTO/TX2	026	S
29.03.2021	16:55:22	Globales Script \$AUFTRAEGE_MIT_EMAIL_ADR wird ausgeführt	/TFTO/TX	750	S
29.03.2021	16:55:22	Datei \usr\sap\ZES\tmp\Kundenaufraege.txt überschrieben	/TFTO/TX	664	S
29.03.2021	16:55:22	*****	/TFTO/TXBT	998	S
29.03.2021	16:55:22	Anzahl gültiger E-Mail-Adressen: 9	/TFTO/TXBT	176	S
29.03.2021	16:55:22	Anzahl ungültiger E-Mail-Adressen: 0	/TFTO/TXBT	177	S
29.03.2021	16:55:22	Anzahl nicht erlaubter E-Mail-Adressen: 0	/TFTO/TXBT	178	S
29.03.2021	16:55:22	*****	/TFTO/TXBT	998	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse joris.sikking@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse sergio.marcelo.cura.daball@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse Hans-Joerg.Kersten@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse priyangshu.kakati@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse karen.lindholm@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse ides@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse susan.pitten@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse elizabeth.stoddard@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Spezifische Server-Datei erstellt für E-Mail-Adresse james.thomas@sap.com !	/TFTO/TXBT	187	S
29.03.2021	16:55:22	Die E-Mail-Adressen sind der Ergebnisliste entnommen worden	/TFTO/TXBT	173	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: joris.sikking@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: sergio.marcelo.cura.daball@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: Hans-Joerg.Kersten@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: priyangshu.kakati@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: karen.lindholm@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: ides@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: susan.pitten@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: elizabeth.stoddard@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	E-Mail mit spezifischem Anhang gesendet an: james.thomas@sap.com	/TFTO/TXBT	189	S
29.03.2021	16:55:22	Job wurde beendet	00	517	S

Wenn wir uns eine der zu sendenden E-Mails anschauen (mithilfe der Transaktion SOST), werden wir feststellen, dass die angehängte Datei nur Sätze enthält, die für den Empfänger relevant sind:



Ein Doppelklick auf den Dateinamen unten links zeigt den Inhalt des Anhangs an:



[Zum Anfang](#)

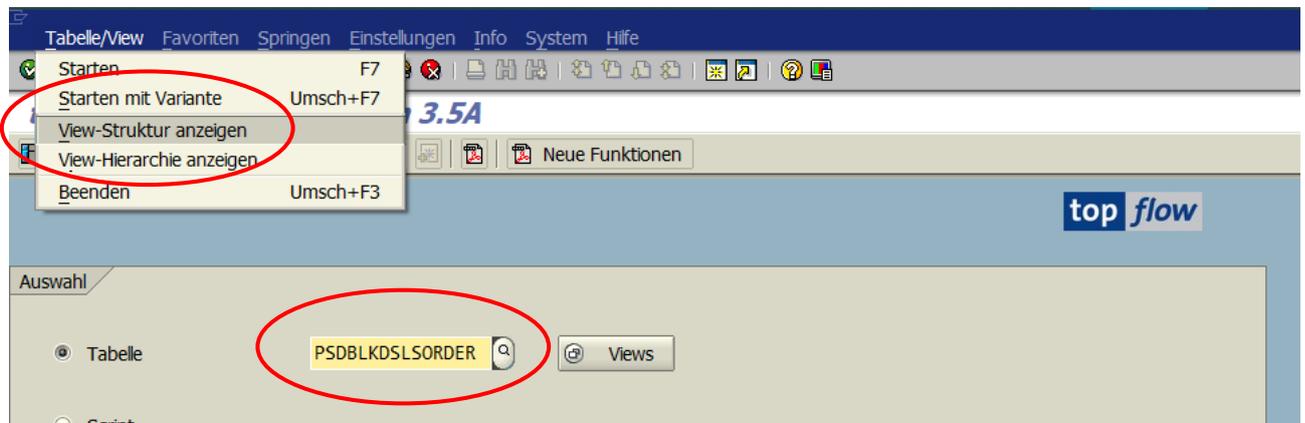
Anzeige der Struktur eines mehrstufigen Views

Normalerweise werden die Felder eines Datenbank-Views direkt aus Datenbank-Tabellen abgeleitet. In den neuesten SAP®-Systemen gibt es jedoch Datenbank-Views, die selbst **auf anderen Datenbank-Views basieren**. Bei einigen Feldern kann sich diese Situation mehrmals wiederholen – in diesem Fall liegt das Feld der “echten” Datenbank-Tabelle **unter vielen Ebenen** von Zwischen-Views verborgen. Dadurch wird es schwierig herauszufinden, auf welchen Datenbank-Tabellen die Viewfelder tatsächlich basieren. Dieses Wissen ist jedoch notwendig, wenn Berechtigungen auf Satzebene definiert werden sollen.

Um diese Probleme zu überwinden, sind in SE16XXL **zwei** Anzeigefunktionen implementiert worden.

Erste Funktion – View-Struktur Anzeigen

Die erste Funktion, die aus der Startmaske von SE16XXL zu erreichen ist, erstellt eine Liste aller Felder des Views mit den zugehörigen Datenbank-Tabellenfeldern:



Das Ergebnis ist wie folgt:

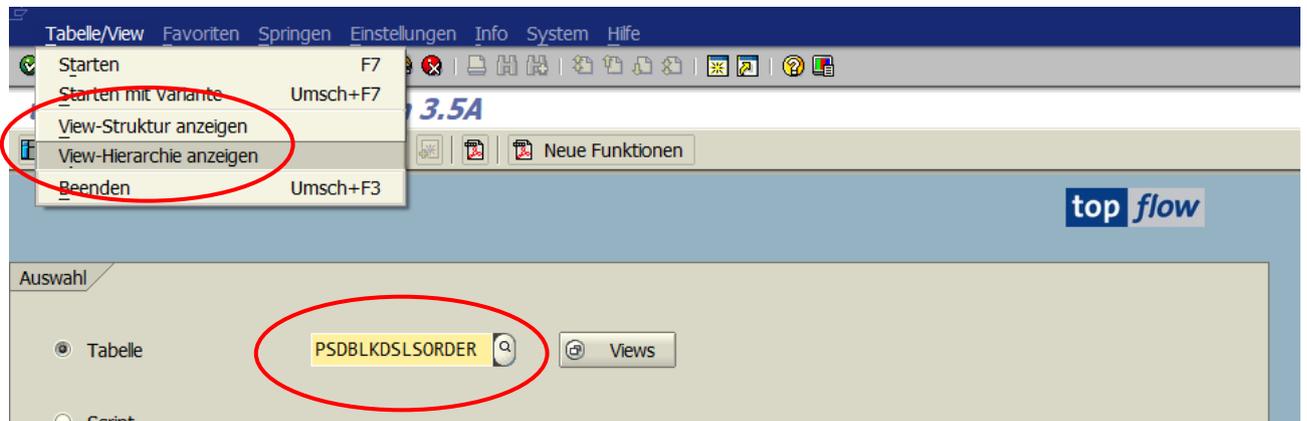
CDS-View PSDBLKDSLORDER -

Pos.	View	Viewfeld	Tabelle	Tabellenfeld	Mand.	Schl.	Kurztext	Dtel.	Domäne	Typ	Lng.	Ber.-Objekt	Ber.-Feld
1	PSDBLKDSLORDER	MANDT	VBAK	MANDT	X	X	Mandant	MANDT	MANDT	CLNT	3		
2	PSDBLKDSLORDER	SALESORDER	VBAK	VBELN	X	X	Kundenauftrag	MANDT	VBELN	CHAR	10		
3	PSDBLKDSLORDER	SALESORDERTYPE	VBAK	AUART	X	X	Verkaufbelegart	VDM_SALES_ORDER	AUART	CHAR	4	V_VBAK_AAT	AUART
4	PSDBLKDSLORDER	SALESORGANIZATION	VBAK	VKORG	X	X	Verkaufsforg.	VKORG	VKORG	CHAR	4	V_VBAK_VKO	VKORG
5	PSDBLKDSLORDER	DISTRIBUTIONCHANNEL	VBAK	VTWEG	X	X	Vertriebsweg	VTWEG	VTWEG	CHAR	2	V_VBAK_VKO	VTWEG
6	PSDBLKDSLORDER	ORGANIZATIONDIVISION	VBAK	SPART	X	X	Sparte	SPART	SPART	CHAR	2	V_VBAK_VKO	SPART
7	PSDBLKDSLORDER	ISSUE	DDDLCHARTYPES	CCHAR4	X	X	Problem	ISSUE_SOF	ISSUE	CHAR	4		
8	PSDBLKDSLORDER	NUMBEROFSALESORDERS	DDDLNUMTYPES	CINT4	X	X	Anzahl Aufträge	NUMBER_OF_SALES_ORDERS	INT4	INT4	10		
9	PSDBLKDSLORDER	NMBROFISSUESINCOMPLETE	DDDLNUMTYPES	CINT4	X	X	Unvollst. Kundenauf.	NMBR_OF_ISSUES_FOR_S001	NMBR_OF_ISSUES_PER_ISSUETYPE	INT4	10		
10	PSDBLKDSLORDER	NMBROFISSUESDELIVERYBLKD	DDDLNUMTYPES	CINT4	X	X	Liefersperre	NMBR_OF_ISSUES_FOR_S002	NMBR_OF_ISSUES_PER_ISSUETYPE	INT4	10		
11	PSDBLKDSLORDER	NMBROFISSUESCREDITBLOCKED	DDDLNUMTYPES	CINT4	X	X	Kreditsperre Kdauf	NMBR_OF_ISSUES_FOR_S003	NMBR_OF_ISSUES_PER_ISSUETYPE	INT4	10		
12	PSDBLKDSLORDER	NMBROFISSUESBILLINGBLOCKED	DDDLNUMTYPES	CINT4	X	X	Fakturasperre	NMBR_OF_ISSUES_FOR_S004	NMBR_OF_ISSUES_PER_ISSUETYPE	INT4	10		

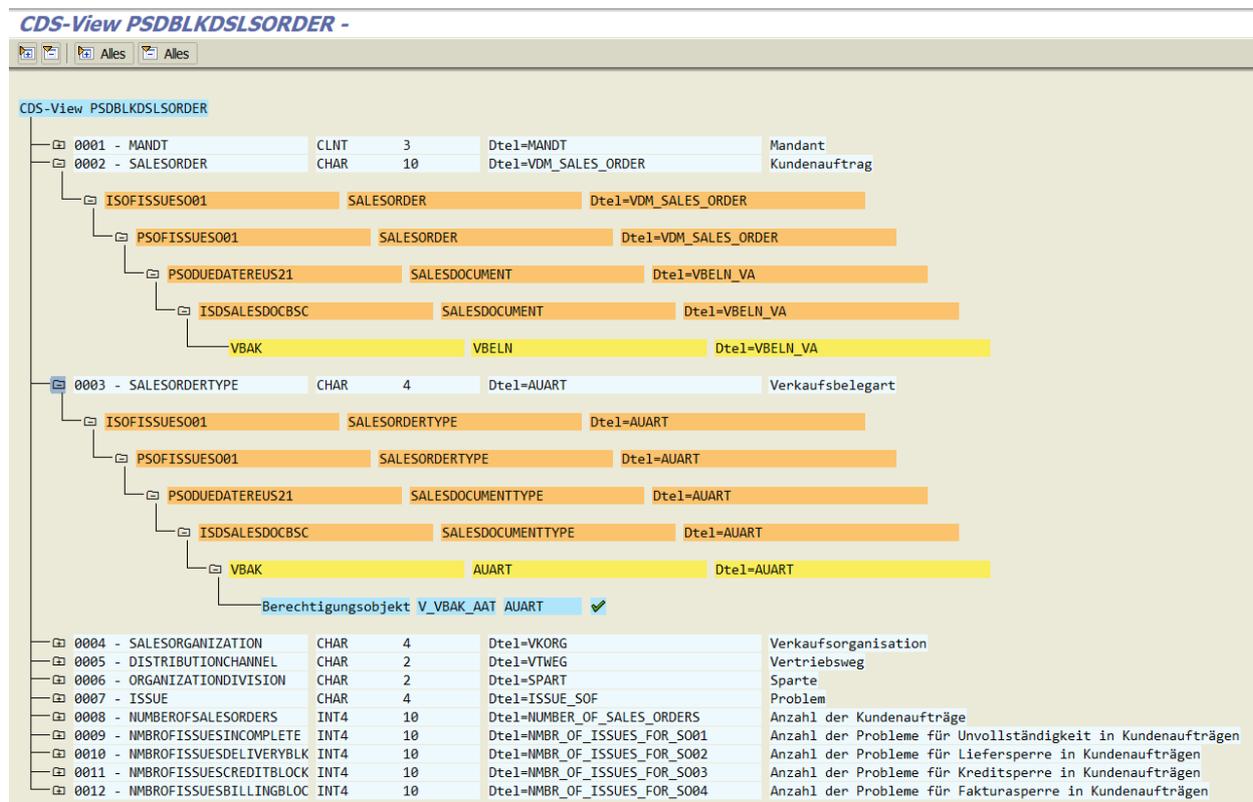
Beachten Sie die definierten Berechtigungsobjekte, die einigen Feldern von VBAK zugeordnet sind.

Zweite Funktion – View-Hierarchie Anzeigen

Die zweite Funktion, die ebenfalls aus der Startmaske von SE16XXL zu erreichen ist, zeigt alle Zwischen-Views hierarchisch an:



Die resultierende Anzeige, die hier mit einigen expandierten Knoten zu sehen ist, ist wie folgt:



[Zum Anfang](#)