

Technical Solution Guide

The Smith Consulting Group, Inc. PO Box 703 Flanders, NJ 07836-0703 973-713-5846 SAPtechsolutions.com

SAP and R/3 are the trademarks or registered trademarks of SAP AG in Germany and in several other countries. Copyright © 2005 The Smith Consulting Group, Inc. All Rights Reserved.

Subsequent Function Code3Table TVFO4Function Group V07A5Processing Function Modules6SD_ORDER_SUBSEQUENT_ALLOWED6SD_DELIVERY_SUBSEQUENT_ALLOWED7Multiple Calls8Custom Goods Issue Requirement Example9Business Requirement9Solution9Step 1Clone the standard routine10Step 2Activate the new routine13Step 3Assign the new routine14Step 4Code the routine16Step 5Test the new routine20	Subsequent Function Requirement Routines	3
Function Group V07A5Processing Function Modules6SD_ORDER_SUBSEQUENT_ALLOWED6SD_DELIVERY_SUBSEQUENT_ALLOWED7Multiple Calls8Custom Goods Issue Requirement Example9Business Requirement9Solution9Step 1Clone the standard routine10Step 2Activate the new routine13Step 3Assign the new routine14Step 4Code the routine16	bubsequent Function Code	3
Processing Function Modules6SD_ORDER_SUBSEQUENT_ALLOWED6SD_DELIVERY_SUBSEQUENT_ALLOWED7Multiple Calls8Custom Goods Issue Requirement Example9Business Requirement9Solution9Step 1Clone the standard routineStep 2Activate the new routineStep 3Assign the new routineStep 4Code the routine	Table TVFO	4
SD_ORDER_SUBSEQUENT_ALLOWED6SD_DELIVERY_SUBSEQUENT_ALLOWED7Multiple Calls8Custom Goods Issue Requirement Example9Business Requirement9Solution9Step 1Clone the standard routineStep 2Activate the new routineStep 3Assign the new routineStep 4Code the routine	unction Group V07A	5
SD_DELIVERY_SUBSEQUENT_ALLOWED	Processing Function Modules	6
Multiple Calls.8Custom Goods Issue Requirement Example.9Business Requirement.9Solution.9Step 1Clone the standard routine	SD_ORDER_SUBSEQUENT_ALLOWED	6
Custom Goods Issue Requirement Example.9Business Requirement.9Solution9Step 1Clone the standard routineStep 2Activate the new routineStep 3Assign the new routineStep 4Code the routineCode the routine16	SD_DELIVERY_SUBSEQUENT_ALLOWED	7
Custom Goods Issue Requirement Example.9Business Requirement.9Solution9Step 1Clone the standard routineStep 2Activate the new routineStep 3Assign the new routineStep 4Code the routineCode the routine16	Multiple Calls	8
Business Requirement.9Solution.9Step 1Clone the standard routine		
Step 1Clone the standard routine10Step 2Activate the new routine13Step 3Assign the new routine14Step 4Code the routine16		
Step 2Activate the new routine13Step 3Assign the new routine14Step 4Code the routine16	Solution	9
Step 3Assign the new routine	Step 1 Clone the standard routine	0
Step 4 Code the routine	Step 2 Activate the new routine	3
1	Step 3 Assign the new routine	4
Step 5 Test the new routine	Step 4 Code the routine	6
	Step 5 Test the new routine	0
Step 6 Running the VOFM regeneration	Step 6 Running the VOFM regeneration	4

Subsequent Function Requirement Routines are used to control the subsequent processing of sales and delivery documents within the Sales and Distribution Module. The purpose of these routines is to prevent specific functions from occurring if certain conditions exist. These routines do not prevent subsequent documents from being created and they should not be confused with Copy Control. For example, these routines will not prevent an invoice from being created from a delivery, but they can prevent the picking of an existing delivery.

Subsequent Function Code

The system uses Subsequent Function Codes to determine which routines to call. Table TVFO contains all subsequent functions defined in the system. The following are the current functions.

Function	Description
01	Requirements from sales document (availability)
02	Purchase requisition from sales document
03	Requirements from sales document (procurement)
04	Create delivery due index
05	Production order/plan order from sales document
06	Release/Block-Assembly Order From Sales Document
11	Picking from delivery
12	Packing from delivery
13	Goods issue from delivery

Table TVFO

Table TVFO contains the routine mapping for each subsequent function code. For each function there are two routines specified. The System Routine (SyFORM routine) is the standard R/3 routine and should not be modified. The User Routine (Routine number) is the user defined requirement.

¢	<u>T</u> able en	try <u>E</u> dit <u>G</u> oto <u>S</u>	ettings <u>U</u> tilities E	<u>n</u> vironment S <u>y</u> ste	m <u>H</u> elp		SAP
•	2		🛯 🛛 😓 I 😋 🤅	9 😢 I 🗅 🛱 🕷	3000	🐹 🗾 🔞 📑	
D	ata B	rowser: Table	e TVFO Select	t Entries	10		
6	s 🕄 4	s a e e q	Check table]			
	ole :)		E. Edward and we				▲ ▼
		fields: 5 of	5 Fixed colum		2 List widt	in 0250	
			Group indicator		SyFORM routine		
	400 400	01 02	FOFU FOFU	101 102	001 002		
	400 400	03 04	FOFU FOFU	103 104	003 004		
	400 400	05 06	FOFU	105	005 000		
	400	11 12	FOFU	111 112	011 012		
	400	13	FOFU	113	013		
	400	14	FOFU	114	014		
							•
L							<u> </u>

In the example above, function code 11 is the picking function code. Routine 011 is the standard system routine and routine 111 is the user routine.

Page 4 of 25

Function Group V07A

Function group V07A contains the processing function modules and all of the requirement routines. Each routine is contained in a separate include. Both the include file and routine name contain the routine number.

Function group Function module SD_DELIVERY_SUBSEQUENT_ALLOWED Active	Function Builder: Display SD_DELIV	
Additional problem Import Exception Source code Import Exception Source code Import Import Exception Import Import Import Import Import Import Import Import Import Import Import Import Import Import	╞╺┥╱╩╓╵╔╶┿╕╧┺┇╙	
Prinction modules Princtin modules Prinction modules Prinction modules		
<pre>indextrained intervery subsequent_ALLOWED intervery subsequent_ALLOWE</pre>	107A 🗸 😪	Import Export Changing Tables Exceptions Source code
Jupit name Image: Stand		
▼ Punction modules ▼ Importing BD DELEMERY SUBSEQUENT_ALLOWED SD_PRODUCT_SELECTION_MANUELL VALUE (BLIPS) LIKE LIPS STRUCTURE LIPS VB SD_PRODUCT_SELECTION_MANUELL ↓ Didionary structures VALUE (BLIPS) LIKE LIPS STRUCTURE TVLX DEFAULT SPACE ↓ Macros VALUE (BTVLP) LIKE TVLX STRUCTURE TVLX DEFAULT SPACE ↓ Macros VALUE (BTVLP) LIKE TVLX STRUCTURE TVLX DEFAULT SPACE ↓ Macros VALUE (BVLP) LIKE TVLX STRUCTURE TVLX DEFAULT SPACE ↓ Macros VALUE (BVL) LIKE TVLX STRUCTURE TVLX DEFAULT SPACE ↓ Macros VALUE (BVBN) LIKE TVP STRUCTURE TVLX DEFAULT SPACE ↓ MALUE (BVBN) LIKE TVP STRUCTURE VBLYB VALUE (BOLLER) OPTIONAL BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_011 MOVE BLIP TO LIPS. MOVE BLINF TO LIPS. WARNING WARNING BEDINGUNG_PRUEFEN_011 MOVE BLIPS TO LIPS. MOVE BLIPS TO LIPS. MOVE BLIPS TO LIPS. MOVE BLIPS TO TVLP. MULE (BALLER) OPTIONAL BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_103 * MIT 14. * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_101 STRUETURE	ject name	
RV_WEIGHT_OROUP_DETERMINE IMPORTING RV_WEIGHT_OROUP_DETERMINE IMPORTING SD_DELWERY_SUBSEQUENT_ALLOWE VALUE (BLIPS) LIKE LIPSYB STRUCTURE LIPSYB SD_DEDRE_SUBSEQUENT_ALLOWE VALUE (BLIPS) LIKE VIX STRUCTURE LIPSYB SD_DEDRE_SUBSEQUENT_ALLOWE VALUE (BLIPS) LIKE VIX STRUCTURE VERAULT SPACE VALUE (BVBUP) LIKE VIX STRUCTURE VERAUE VALUE (BVBUP) LIKE VIX STRUCTURE VERAUE VALUE (BVBUP) LIKE VIPO-FOFUN VALUE (BVBUP) LIKE VIPO-FOFUN ASSEMBLY_ORDER_CHECK_ACTIVITY VALUE (BCALLER) OPTIONAL BEDINGUNG_PRUEFEN_003 DEFAULT SPACE BEDINGUNG_PRUEFEN_003 VALUE (BCALLER) OPTIONAL BEDINGUNG_PRUEFEN_103 WALUE (BCALLER) OPTIONAL BEDINGUNG_PRUEFEN_104 MOVE BLIPS TO LIPS: BEDINGUNG_PRUEFEN_103 MOVE BLIPS TO LIPS: BEDINGUNG_PRUEFEN_104 MOVE BLIPS TO LIPS: BEDINGUNG_PRUEFEN_103 MOVE BLIPS TO LIPS: BEDINGUNG_PRUEFEN_104 MOVE BUPA TO VALUE BEDINGUNG_PRUEFEN_105 MOVE BUPA TO CALLER: BEDINGUNG_PRUEFEN_104 FROMO BEDINGUNG_PRUEFEN_105 FROMO <td< td=""><td></td><td>*"</td></td<>		*"
SD_DELWERY_SUBSEQUENT_ALLOWE SD_ORDER_SUBSEQUENT_ALLOWE SD_PRODUCT_SELECTION_MANUELL ************************************		*" IMPORTING
SD_ORDER_SUBSEQUENT_ALLOWED SD_PRODUCT_SELECTION_MANUELL P Dictionary structures P ALUE (STULY) LIKE TYLK STRUCTURE TVLY DEFAULT SPACE VALUE (STULY) LIKE TYLK STRUCTURE TVLY DEFAULT SPACE VALUE (BYLP) LIKE TYLK STRUCTURE TVLY DEFAULT SPACE VALUE (BYDP) LIKE TYLK STRUCTURE VBUYB DEFAULT SPACE VALUE (BYDP) LIKE VBUYS STRUCTURE VBPA DEFAULT SPACE VALUE (BYDP) LIKE VFLK STRUCTURE VBPA DEFAULT SPACE VALUE (BYDPA) STRUCTURE VBPA DEFAULT SPACE VALUE (BYDPA) LIKE VFLK STRUCTURE VBPA DEFAULT SPACE VERTOR VFLK STRUCTURE VBPA DEFAULT SPACE VALUE (BYDPA) STRUCTURE VBPA DEFAULT SPACE VERTOR VFLK STRUCTURE VBPA DEFAULT SPACE VERTOR		VALUE (DETKE) ETKE ETKE STRUCTURE ETKE DEFAULT STAGE
 Dictional Structures Dictional Structures VALUE (SPRUK) Like Tructure TWP DEFAult SPACE VALUE (SPRUK) Like VOLVE STRUCTURE VOLVE VOLVE Marcos Subroutines ASSEMBLY_ORDER_CHECK_ACTIVITY ASSEMBLY_ORDER_ONDER BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_003 BEDINGUNG_PRUEFEN_013 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_110 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_115 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_119 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_111 <l< td=""><td>SD_ORDER_SUBSEQUENT_ALLOWED</td><td>*" DEFAULT SPACE</td></l<>	SD_ORDER_SUBSEQUENT_ALLOWED	*" DEFAULT SPACE
 Autorial products Fields Matros Subroutines Assembly_ORDER_OHECK_ACTIVITY Assembly_ORDER_OHECK_ACTIVITY Assembly_ORDER_OHECK_STATUS BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_003 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_012 BEDINGUNG_PRUEFEN_013 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_119 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_115 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRU		VALUE (DIVER) LIKE IVER STRUGTURE IVER DELADET STAGE
 Marros Warros Warros WALUE (BYBUP) LIKE WBUP STRUCTURE VBUPYB VALUE (BYBUP) LIKE WBUP STRUCTURE VBUPYB WALUE (FOFUN) LIKE TVPO-FOFUN VALUE (BYBPA) LIKE VBPA STRUCTURE VBPA DEFAULT SPACE WALUE (BYBA) LIKE VBPA STRUCTURE VBA DEFAULT SPACE WOYE BUBPA TO LIKE WOYE BUBPA TO LIKE MOYE BUBPA TO VBUR. MOYE BUBPA TO VBUR. MOYE BUBPA TO VBUR. MOYE BUBPA TO VBPA. MOYE BUBPA TO VBPA. MOYE BUBPA TO VSPAL. MOYE BUBPA		*" VALUE(BVBUK) LIKE VBUKVB STRUCTURE VBUKVB
Subroutines ASSEMBLY_ORDER_CHECK_ACTWITY ASSEMBLY_ORDER_CHECK_ACTWITY ASSEMBLY_ORDER_CHECK_ACTWITY BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_002 BEDINGUNG_PRUEFEN_003 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_015 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_015 BEDINGUNG_PRUEFEN_016 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_102 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_1168 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_1168 BEDINGUNG_PRUEFEN_1174 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108		
ASSEMBLY_ORDER_CHECK_ACTIVITY ASSEMBLY_ORDER_CHECK_STATUS BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_002 BEDINGUNG_PRUEFEN_003 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_012 BEDINGUNG_PRUEFEN_012 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_08 IF NOT TVFO-GRND_SY IS INITIAL. MOVE BYBPA TO EXP_CTR_VBPA. ENDIF. * USEPPOTUNG IF NOT TVFO-GRND_SY IS INITIAL. MOVE FORM (FORMAME_GRND. ENDIF.		*" DEFAULT SPACE
BEDINGUNG_PRUEFEN_001 BEDINGUNG_PRUEFEN_002 BEDINGUNG_PRUEFEN_003 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_005 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_012 BEDINGUNG_PRUEFEN_013 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_015 BEDINGUNG_PRUEFEN_016 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_1108 BEDINGUNG_PRUEFEN_1108 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_115 BEDINGUNG_PRUEFEN_116 BEDINGUNG_PRUEFEN_117 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 BEDINGUNG_PRUEFEN_118 <		VALUE(FUFUN) LIKE IVFU-FUFUN
BEDINGUNG_PRUEFEN_002 ** ERROR BEDINGUNG_PRUEFEN_003 ** WARNING BEDINGUNG_PRUEFEN_014 MOVE BLIKP TO LIKP. BEDINGUNG_PRUEFEN_013 MOVE BLIKP TO LIVP. BEDINGUNG_PRUEFEN_013 MOVE BVUP TO VBUK. BEDINGUNG_PRUEFEN_014 MOVE BVUP TO VBUK. BEDINGUNG_PRUEFEN_013 MOVE BVUP TO VBUK. BEDINGUNG_PRUEFEN_014 MOVE BVUP TO VBUK. BEDINGUNG_PRUEFEN_014 MOVE BVUP TO VBUK. BEDINGUNG_PRUEFEN_014 MOVE BVD.TO TVLP. BEDINGUNG_PRUEFEN_102 *1f blips-vbBr.eq (908000'. * Aufruf aus WA-Buchung BEDINGUNG_PRUEFEN_103 0R (BVPA-PARW E0 'WE' * Aufruf aus Kommiss. * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_105 0R (BVPA-PARW E0 'WE' * Aufruf aus Kommiss. * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_106 0R (BVPA-RAW E0 'WE' * Aufruf aus Kommiss. * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_112 * Systemprüfung * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_113 * Systemprüfung IF NOT TYFO-GRPNO_SY TO FORMAME-GRPNO. * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_113 * Systemprüfung IF NOT TYFO-GRPNO_SY TO FORMAME-GRPNO. * Systemprüfung		*" VALUE (BCALLER) OPTIONAL
BEDINGUNG_PRUEFEN_003 BEDINGUNG_PRUEFEN_004 BEDINGUNG_PRUEFEN_005 BEDINGUNG_PRUEFEN_011 BEDINGUNG_PRUEFEN_012 BEDINGUNG_PRUEFEN_013 BEDINGUNG_PRUEFEN_014 BEDINGUNG_PRUEFEN_013 BEDINGUNG_PRUEFEN_101 BEDINGUNG_PRUEFEN_102 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 <td< td=""><td></td><td>EAGEFITONO</td></td<>		EAGEFITONO
BEDINGUNG_PRUEFEN_005MOVE BLIKP TO LIKP. MOVE BLIPS TO LIPS.BEDINGUNG_PRUEFEN_011MOVE BULIPS TO LIPS.BEDINGUNG_PRUEFEN_013MOVE BVUK TO VBUK. MOVE BVUK TO VBUK.BEDINGUNG_PRUEFEN_013MOVE BULLER TO CALLER.BEDINGUNG_PRUEFEN_104MOVE BULK TO TVLK. MOVE BULK TO TVLK.BEDINGUNG_PRUEFEN_103* 1f blips-vosnr eq '000000'. * Aufruf aus WA-BuchungBEDINGUNG_PRUEFEN_104* 1f blips-vosnr eq '000000'. * Aufruf aus WA-BuchungBEDINGUNG_PRUEFEN_105* 1f blips-vosnr eq '000000'. * Aufruf aus Kommiss.BEDINGUNG_PRUEFEN_106* 1f blips-vosnr eq '000000'. * Aufruf aus Kommiss.BEDINGUNG_PRUEFEN_105* 1f blips-vosnr eq '000000'. * Aufruf aus Kommiss.BEDINGUNG_PRUEFEN_105* NAD NOT (BVPA-KUNNR IS INITIAL)).BEDINGUNG_PRUEFEN_106MOVE BVPA TO EXP_CTR_VBPA. ENDIF.BEDINGUNG_PRUEFEN_111ENDIF.BEDINGUNG_PRUEFEN_112* Systemprufung * SystemprufungBEDINGUNG_PRUEFEN_114* Systemprufung * SystemprufungBEDINGUNG_PRUEFEN_114* Systemprufung * SystemprufungBEDINGUNG_PRUEFEN_114* ON TVFO-GRPNO_SY TO FORNNAME-GRPNO. * SystemprufungBEDINGUNG_PRUEFEN_114* Userprufung * IF NOT TVFO-GRPNO_SY TO FORNNAME-GRPNO. * PERFORM (FORNNAME) IN PROGRAM (PRO6_ORDER). * UserprufungCHECK_RMBA_FOR_DELETION CHECK_RMBA_FOR_DELETION LUST_ANDE* USERPRUFUNG TO NO * USERPROM (FORNNAME) IN PROGRAM (PRO6_ORDER). * USERPROM (FORNNAME) IN PROGRAM (PRO6_ORDER). * ENDIF.LIST_INITENDELMICTION		
BEDINGUNG_PRUEFEN_101MOVE BLIPS TO LIPS. MOVE BVBUK TO VBUK.BEDINGUNG_PRUEFEN_101MOVE BVBUK TO VBUK. MOVE BVBUT TO VBUP.BEDINGUNG_PRUEFEN_103MOVE BTULP TO TVLK. MOVE BTULE TO CALLER.BEDINGUNG_PRUEFEN_101MOVE BULR TO CALLER.BEDINGUNG_PRUEFEN_102'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_103'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_104'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_105'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_106'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_107'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_108'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_112'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_113'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_114'11 blips-vbeln is initial.BEDINGUNG_PRUEFEN_115'11 blips.''BEDINGUNG_PRUEFEN_116'11 blips.''BEDINGUNG_PRUEFEN_117'11 blips.''BEDINGUNG_PRUEFEN_118'11 blips.''BEDINGUNG_PRUEFEN_114'11 blips.''BEDINGUNG_PRUEFEN_115'11 blips.''BEDINGUNG_PRUEFEN_116'11 blips.''BEDINGUNG_PRUEFEN_117'11 blips.''BEDINGU		*"MOVE BLIKP TO LIKP
BEDINGUNG_PRUEFEN_1012BEDINGUNG_PRUEFEN_1013BEDINGUNG_PRUEFEN_1014BEDINGUNG_PRUEFEN_1012BEDINGUNG_PRUEFEN_102BEDINGUNG_PRUEFEN_103BEDINGUNG_PRUEFEN_103BEDINGUNG_PRUEFEN_104BEDINGUNG_PRUEFEN_105BEDINGUNG_PRUEFEN_106BEDINGUNG_PRUEFEN_106BEDINGUNG_PRUEFEN_107BEDINGUNG_PRUEFEN_108BEDINGUNG_PRUEFEN_108BEDINGUNG_PRUEFEN_111BEDINGUNG_PRUEFEN_112BEDINGUNG_PRUEFEN_113BEDINGUNG_PRUEFEN_113BEDINGUNG_PRUEFEN_114BEDINGUNG_PRUEFEN_113BEDINGUNG_PRUEFEN_114 <t< td=""><td></td><td>MOVE BLIPS TO LIPS.</td></t<>		MOVE BLIPS TO LIPS.
BEDINGUNG_PRUEFEN_013BEDINGUNG_PRUEFEN_014BEDINGUNG_PRUEFEN_014BEDINGUNG_PRUEFEN_012BEDINGUNG_PRUEFEN_103BEDINGUNG_PRUEFEN_103BEDINGUNG_PRUEFEN_104BEDINGUNG_PRUEFEN_105BEDINGUNG_PRUEFEN_106BEDINGUNG_PRUEFEN_107BEDINGUNG_PRUEFEN_108BEDINGUNG_PRUEFEN_111BEDINGUNG_PRUEFEN_112BEDINGUNG_PRUEFEN_113BEDINGUNG_PRUEFEN_114BE		
BEDINGUNG_PRUEFEN_101 HOVE BCALLER TO CALLER. BEDINGUNG_PRUEFEN_102 * 1f bilps-vbeln is initial. BEDINGUNG_PRUEFEN_103 * 1f bilps-vbeln is initial. BEDINGUNG_PRUEFEN_104 * 1f bilps-vbeln is initial. BEDINGUNG_PRUEFEN_105 * 080000' BEDINGUNG_PRUEFEN_106 0R (BVBPA-PARVW EQ 'WE' * Aufruf aus WA-Buchung BEDINGUNG_PRUEFEN_108 0R (BVBPA-NARWW EQ 'WE' * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_118 NOVE BVBPA TO VBPA. BEDINGUNG_PRUEFEN_112 MOVE BVBPA TO VBPA. BEDINGUNG_PRUEFEN_113 PERFORM TVFO_READ USING FOFUN. BEDINGUNG_PRUEFEN_114 * OT TVFO-GRPNO_SY IS INITIAL. BEDINGUNG_PRUEFEN_114 * NOT TVFO-GRPNO_SY IS INITIAL. BEDINGUNG_PRUEFEN_114 * OT TVFO-GRPNO_SY IS INITIAL. BEDINGUNG_PRUEFEN_100 * Userprüfung IF NOT TVFO-GRPNO_SY IS INITIAL. MOVE TVFO-GRPNO_SY IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOYE TVFO-GRPNO TO FORMMAME-GRPNO. PERFORM (FORMNAME) IN PROGRAM (PROG_ORDER). LIST_AUSGABE UST_INT LIST_INIT ENDEFUNCTION		MOVE BTVLP TO TVLP.
BEDINGUNG_PRUEFEN_102 BEDINGUNG_PRUEFEN_103 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_104 BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_107 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_114 <td< td=""><td></td><td></td></td<>		
BEDINGUNG_PRUEFEN_104 IF BLIPS-POSKE E0'060000' * Aufruf aus WA-Buchung BEDINGUNG_PRUEFEN_105 OR (BVBA-PARVW E0 'WE' * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_106 OR (BVBA-PARVW E0 'WE' * Aufruf aus Kommiss. BEDINGUNG_PRUEFEN_108 MOVE BVBA TO VBPA. BEDINGUNG_PRUEFEN_108 MOVE BVBA TO VBPA. BEDINGUNG_PRUEFEN_112 MOVE BVBA TO VBPA. BEDINGUNG_PRUEFEN_113 PERFORM TYFO_READ USING FOFUN. BEDINGUNG_PRUEFEN_114 FORTUREFEN_900 BEDINGUNG_PRUEFEN_900 CHECK_PMMA_FOR_DELETION CHECK_PRMA_FOR_DELETION CHECK_RMBA_FOR_DELETION CHECK_RMBA_FOR_DELETION FNOT TYFO-GRPNO_SY IS INITIAL. LIPS_AUSFUHR_KONTROLLE USERPTOTUNG LIST_AUSGABE USET_ORME LIST_INIT ENDIF. ENDIF. ENDIF. ENDIF. ENDIF. ENDIF. USERPTOTUNG INST_INIT ENDIF.	BEDINGUNG_PRUEFEN_102	* if blips-vbeln is initial.
BEDINGUNG_PRUEFEN_105 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_106 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_108 BEDINGUNG_PRUEFEN_111 BEDINGUNG_PRUEFEN_112 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_113 BEDINGUNG_PRUEFEN_114 BEDINGUNG_PRUEFEN_106 CHECK_RMBA_FOR_DELETION CHECK_RMBA_FOR_DELETION CHECK_RMBA_FOR_DELETION LIST_AUSGABE LIST_AUSGABE LIST_INIT LIST_INIT DIST_INT		
BEDINGUNG_PRUEFEN_106 AND NOT (SPRA-RUNRE IS INITIAL)). BEDINGUNG_PRUEFEN_108 MOVE BVBPA TO VEPA. BEDINGUNG_PRUEFEN_112 MOVE BVBPA TO EXP_CTR_VBPA. BEDINGUNG_PRUEFEN_112 PERFORM TVF0_READ USING FOFUN. BEDINGUNG_PRUEFEN_113 Systemprutung BEDINGUNG_PRUEFEN_114 MOVE BVBPA. BEDINGUNG_PRUEFEN_114 MOVE OF RND_SY IS INITIAL. BEDINGUNG_PRUEFEN_114 MOVE VF0-GRPN0_SY TO FORNAME-GRPN0. BEDINGUNG_PRUEFEN_100 PERFORM (FORNAME) IN PROGRAM (PRO6_ORDER). CHECK_PMBA_FOR_DELETION ENDIF. CHECK_RMBA_FOR_DELETION IF NOT TVF0-GRPN0 IS INITIAL. LIPS_AUSFUHR_KONTROLLE IF NOT TVF0-GRPN0 IS INITIAL. LIST_AUSGABE PERFORM (FORMAME) IN PROGRAM (PRO6_ORDER). LIST_INIT ENDIF.		OR (BVBPA-PARVW EQ 'WE' "Aufruf aus Kommiss."
BEDINGUNG_PRUEFEN_108 MOVE BVBPA TO EXP_CTR_VBPA. BEDINGUNG_PRUEFEN_111 ENDIF. BEDINGUNG_PRUEFEN_112 PERFORM TVFO_READ USING FOFUN. BEDINGUNG_PRUEFEN_113 Systemprüfung BEDINGUNG_PRUEFEN_114 IF NOT TVFO-GRPNO_SY IS INITIAL. BEDINGUNG_PRUEFEN_100 IF NOT TVFO-GRPNO_SY TO FORMNAME-GRPNO. CHECK_PMA_FOR_DELETION PERFORM (FORNAME) IN PROGRAM (PRO6_ORDER). CHECK_RMBA_FOR_DELETION USEPDITUNG IJF_AUSFUHR_KONTROLLE IF NOT TVFO-GRPNO IS INITIAL. UIST_AUSGABE MOVE TVFO-GRPNO TO FORMNAME GRPNO. UIST_INIT PERFORM (FORMAME) IN PROGRAM (PRO6_ORDER). ENDIF. ENDIF.		
BEDINGUNG_PRUEFEN_112 PERFORM TVF0_READ USING FOFUN. BEDINGUNG_PRUEFEN_113 * Systemprufung BEDINGUNG_PRUEFEN_114 IF NOT TVF0_GRPN0_SY IS INITIAL. BEDINGUNG_PRUEFEN_100 * OFFORM (FORMNAME) IN PROGRAM (PROG_ORDER). CHECK_PPMA_FOR_DELETION ENDIF. CHECK_RMBA_FOR_DELETION * Userprufung IF FOFUN_TEXT_READ IF NOT TVF0-GRPN0 IS INITIAL. LIPS_AUSFUHR_KONTROLLE IST_AUSGABE LIST_AUSGABE PERFORM (FORMNAME) IN PROGRAM (PROG_ORDER). LIST_INIT ENDIF.		MOVE BVBPA TO EXP_CTR_VBPA.
BEDINGUNG_PRUEFEN_113 * Systemprüfung BEDINGUNG_PRUEFEN_114 IF NOT TVF0-6RPN0_SY IS INITIAL. BEDINGUNG_PRUEFEN_000 IF NOT TVF0-6RPN0_SY ID FORMNAME-6RPN0. CHECK_PPMA_FOR_DELETION PERFORM (FORMNAME) IN PROGRAM (PROG_ORDER). CHECK_RMAA_FOR_DOLETION ENDIF. FOFUN_TEXT_READ IF NOT TVF0-6RPN0 IS INITIAL. LIPS_AUSFUHR_KONTROLLE MOVE TVF0-6RPN0 IS INITIAL. LIST_ENDE ENDIF. LIST_INIT ENDELWCTION		
BEDINGONG_RRUEFEN_300 MOVE TVF0-GRPNO_SY TO FORMAME-GRPNO. CHECK_PPMA_FOR_DELETION FORDUN_TEXT_READ CHECK_RMBA_FOR_DELETION IN Serpoiding FOFUN_TEXT_READ ISSCHURK_KONTROLLE LIST_AUSGABE PERFORM (FORMAME) IN PROGRAM (PROG_ORDER). LIST_INNT ENDIF.		* Systemprüfung
CHECK_PPMA_FOR_DELETION CHECK_PRMA_FOR_DELETION CHECK_RMBA_FOR_DELETION FOFUN_TEXT_READ LIPS_AUSFUHR_KONTROLLE LIST_AUSGABE LIST_ENDE LIST_INIT LI		
CHECK_RMBA_FOR_DELETION FOFUN_TEXT_READ LIPS_AUSFUHR_KONTROLLE LIST_AUSGABE LIST_ENDE LIST_INIT LIST_INI		PERFORM (FORMNAME) IN PROGRAM (PROG_ORDER).
FOFUN_TEXT_READ LIPS_AUSFUHR_KONTROLLE LIST_AUSGABE LIST_ENDE LIST_INIT		
LIST_AUSGABE PERFORM (FORMNAME) IN PROGRAM (PROG_ORDER). LIST_ENDE ENDIF. LIST_INIT ENDELINGTION		IF NOT TVFO-GRPNO IS INITIAL.
LIST_ENDE ENDIF. LIST_INIT ENDELINGTION		
		ENDIF.
MAKT_ERMITTELN		
SALES HEADER CHECK STATUS		
SALES_HEADER_CHECK_STATUS		Li 1, Co 1 Ln 1 - Ln 49 of 49 lines

Page 5 of 25

Processing Function Modules

There are two function modules in this function group that process requirements. One function is for sales documents, and the other is for delivery documents. These function modules are called from various points in the Order and Delivery programs. The calling programs pass the appropriate function code to the function module. The logic reads table TVFO with the passed function code to determine which routines to call. Each function executes the non-modifiable system routine first, and then executes the user defined routine.

SD_ORDER_SUBSEQUENT_ALLOWED

Passes all relevant sales document structures and the function code.

FUNCTION SD ORDER SUBSEQUENT ALLOWED **IMPORTING** TVAP BTVAP LIKE BTVEP LIKE TVEP LIKE VBAK **BVBAK BVBAP** LIKE VBAP **BVBEP** LIKE VBEP BVBEPVB LIKE **VBEPVB** LIKE BVBUK VBUK **BVBUP** LIKE **VBUP** FOFUN LIKE **TVFO-FOFUN** I SIMUL MODE DEFAULT SPACE I_CHECK_CYCLE_FOFUN_04 DEFAULT SPACE **EXPORTING** NO CREATE LIKE **TPTEST-STATUS** NO CHANGE LIKE **TPTEST-STATUS EXCEPTIONS** ERROR WARNING ERROR BUT DO NOT DELETE.

SD_DELIVERY_SUBSEQUENT_ALLOWED

Passes all relevant delivery document structures and the function code.

FUNCTION SD_DELIVERY_SUBSEQUENT_ALLOWED IMPORTING BLIKP LIKE LIKP LIKE **LIPSVB** BLIPS BTVLK LIKE TVLK BTVLP LIKE TVLP BVBUK LIKE VBUKVB **BVBUP** LIKE VBUPVB FOFUN LIKE **TVFO-FOFUN** LIKE **BVBPA** VBPA BCALLER **OPTIONAL** EXCEPTIONS ERROR WARNING.

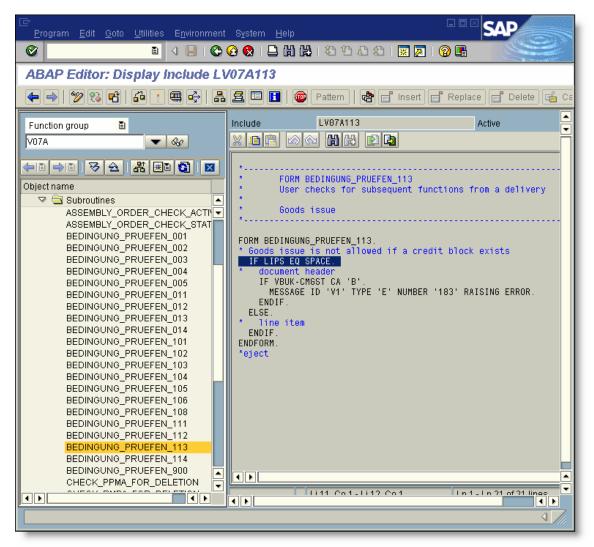
6/19/2005

Page 7 of 25

Technical Solution Guide

Multiple Calls

These functions are called multiple times from the calling program during the processing of a document. Each time the functions are called, the data contained in the structures may be different. In the case of delivery processing, the function is first called with only the header data populated. During subsequent calls, the header data is not passed, but item data is present.



The above example is the standard logic from the goods issue routine. The logic checks a field in VBUK only when LIPS is initial. The ELSE side of the IF statement is for checking item fields. This is because when LIPS is initial, the header data is being passed.

We suggest setting break points in the functions and routines to understand exactly how they work before implementing your own logic.

Page 8 of 25

Custom Goods Issue Requirement Example

The following example demonstrates how to implement a custom goods issue requirement routine. The basic process is the same for the other available functions.

Business Requirement

For a specific plant, inventory balances are maintained in a custom 'Z' table in addition to the standard R/3 inventory. This was done as part of a larger project to track the quantities of materials imported into Mexico for re-sale. Without elaborating on the design of the entire project, the purpose of this portion is to ensure that inventory exists in the 'Z' table during the goods issue process. If there is insufficient inventory in the table, the goods issue will not be posted.

Solution

To solve this problem, a custom goods issue requirement routine will be implemented. The logic in this routine will check the inventory balance in the 'Z' table and prevent the goods issue if the quantity in the table is less than the delivery quantity.

Step 1 Clone the standard routine

Using transaction VOFM, navigate to the subsequent function requirement routine for goods issue. Alternatively, you can go directly to the good issue requirement using transaction OVB7.

「 able <u>E</u> dit <u>C</u> opying requirements <u>D</u> ata transfer 」	<u>R</u> equirements <u>F</u> ormulas Syst	em <u>H</u> elp SAP
Maintain: Requirements & Formulas	Output co <u>n</u> trol A <u>c</u> count determination <u>M</u> aterial determination Listing/exclusion Free <u>g</u> oods <u>B</u> onus buy Reta <u>i</u> l pricing <u>R</u> isk management Cr <u>e</u> dit checks	
	Card authorization Subsequent functions	Reqs.availability Purch.req./assembly Delivery Due Index Picking Packing Ooods issue Distribution
		47

Page 10 of 25

There is only one routine allowed for each function even though the screen implies that multiple routines can be used. On this screen, press the 'Req. maintenance' button at the bottom of the screen.

C T	able view	Edit Goto Se	election criteria	<u>U</u> tilities S <u>y</u> stem	Help		SAP
8					18 2 C C 2		
CI	hange ¹	View "Requ	irement for	r Goods Issu	e from a Delive	ery": Overview	
62	∞ 🖪						
Goo	ods issue :	for delivery					
		Requirmnt	SVEORM rou	System requir.	Subs.fu	1	
	113	Goods Issue		System: Goods 1	Issue 13 🔺		
					•		
	ļ						
	ļ						
	<u> </u>						
	Dog moint	ananca					
	Req.maint						
	Only one	entry chosen					<u> </u>

Page 11 of 25

To clone the standard routine (113), type over the existing routine number and press enter. The system will prompt you for an object key. Obtain and specify the object key, then press ENTER.

Routine number	Description	Active	Application			
	System: BANF	V				
I	System: Requirements	V			-	
	System: DlvDueIndex	V				
i	System: Assembly	¥				
1	System: Picking	V				
2	System: Packing	¥				
3	System: Goods Issue	V				
4	System: Distribution	V				
01	Reqs.availability	V				
02	PReq	V				
03	Requirements	V				
04	Delivery Due Index	1				
05	Assembly	1				
06	Block construction	4				
08	Ship. due date index	V				
11	Picking	1				
12	Packing	¥				
113	Custom Goods Issue	~			▲	
14	Distribution	×			-	
• •				• •		

In this example, we typed routine 913 over 113. It is a common practice to simply replace the first digit of these routines with a '9'. Using this standard makes it obvious which routine was used for cloning.

Step 2 Activate the new routine

VOFM routines need to be activated before they can be used.

년 <u>T</u> able <u>E</u>	dit System	<u>H</u> elp								SA	
Ø	A <u>c</u> tivate all		0	(8 🕐	ا 🕄 🗘	₩ 2	2 🖪		
Mainta	Activate		<u> </u>		inctions						
	<u>D</u> eactivate		ooquo								
19 🖬 [De <u>l</u> ete line	Shift+F2									
Routine n	umber	Description	ŀ	Active	Application			[===		
111		Picking		1							
112		Packing		1					-		
113		Goods Issue		4							
114		Distribution		1							
þ 13		Custom Goods	Issue								
						_					
						_					
						_					
						_					
						_					
						_					
						_					
						-					
						-					
						-					
						-					
						-					
						-					
					1						
83	Positio	on									
											4 7/

Page 13 of 25

Step 3 Assign the new routine

Only one routine can be active in the system. There is no other configuration required.

⊡ Ta	able view	Edit Go	oto Sele	ection cr	riteria <u>U</u> tiliti	es Systi	em He	qlp						SAP	
0			Ē		8 00				0 0	8	💥	21	2 🖪		
Ch	ange	View "F	Requir	reme	nt for Go	ods Is	sue f	from a	a De	live	ery"	: Ove	erview		
62	∞ 🖪	B													
Goo	ids issue	for delivery	./							_					
	Routin	Requirmn	nt		SyFORM ro	System r	equir.		Su	11					
	913	Custom G				System:		Issue			i I				
										•					
	ļ														
	ļ														
	ļ														
	ļ														
	<u> </u>														
	<u> </u>														
	<u> </u>														
	<u> </u>														
	<u> </u>														
	<u> </u>														
										Þ					
)							_		_					
			1												
	Req.maint		J												
	Only one	entry chose	en									_			

Page 14 of 25

Examining the TVFO table, we now see that subsequent function 13 points to routine number 913.

C		try <u>E</u> dit <u>G</u> oto <u>S</u> i	ettings <u>U</u> tilities E	<u>n</u> vironment S <u>y</u> ste	m <u>H</u> elp		SAP
(🗈 🔍 🔛 I 😋 🤅) 🔇 I 🗅 🕅 🕊	8008	🐹 🗾 🔞 📑	
1	Data B	rowser: Table	TVFO Select	t Entries	10		
	& 🕄 🕹	3 7 B B 3	Check table]			
	able : 1 splayed	FVFO fields: 5 of	5 Fixed colum	ns:	2 List widt	:h 0250	
	Client	Subseq.function	Group indicator	Routine number	SyFORM routine		
	400 400	01 02	FOFU FOFU	101 102	001 002		
	400 400	03 04	FOFU FOFU	103 104	003 004		
	400 400	05 06	FOFU FOFU	105 106	005 000		
		11 12 13	FOFU FOFU FOFU	111 112 913	011 012 013		
		14	FOFU	114	014		

Page 15 of 25

Step 4 Code the routine

To access the source code, pull down the list of valid routines (F4), select the new routine and press the source code icon at the bottom of the dialog box.

Image: Selection criteria Utilities System Help Image: Selection criteria Utilities System Help Image: Selection criteria Utilities System Help Image: Selection criteria Utilities System Help Selection Image: Selection criteria Utilities System Help Selection Selection
Change View "Requirement for Goods Issue from a Delivery": Overview
Goods issue for delivery
Routin Requirmnt SyFORM ro Su 913 Custom Goods Issue 13 System: Goods Issue 13
Routines 🗵
R Description
184 Delivery Due Index
105 Assembly
106 Block construction
198 Ship. due date index
111 Picking
112 Packing
113 Goods Issue
114 Distribution
913 Custom Goods Issue
Req.maintenance

Page 16 of 25

When a routine is cloned, the system automatically inserts comments at the top of the include denoting that it was copied from another routine. Since the include generated (RV07A913) is within the R/3 name range, the system protects the code as if we were installing a source code modification.

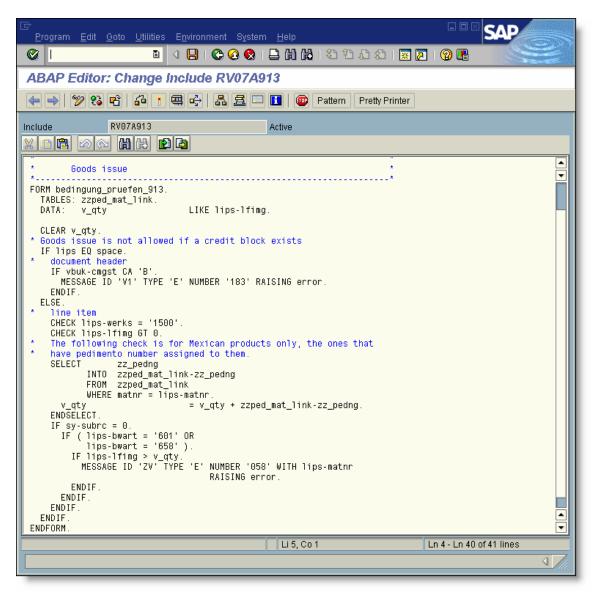
它 Program <u>E</u> dit <u>G</u> oto <u>U</u> tilities E <u>n</u> vironment S <u>v</u> stem <u>H</u> elp	
🖉 🔲 🖞 💾 🚱 🚱 😫 🗒 🖓 🔛 🖉	8 G 🧲
ABAP Editor: Modify Include RV07A913	
🔄 🔿 🦻 🚱 T 🚍 🚭 🛃 🔲 🚺 🎯 Pattern 🖑 🗗 Insert 📻	Replace 📑 Delete 🖷 C
Include RV07A913 Active	
*** Attention: copied routine! *** Attention: *** Character string 113 is replaced by 913 everywhere ! ***	
* FORM BEDINGUNG_PRUEFEN_913 * * User checks for subsequent functions from a delivery * * * * Goods issue *	
FORM BEDINGUNG_PRUEFEN_913. * Goods issue is not allowed if a credit block exists IF LIPS EQ SPACE. * document header IF VBUK-CMGST CA 'B'. MESSAGE ID 'V1' TYPE 'E' NUMBER '183' RAISING ERROR. ENDIF. ELSE. * line item ENDIF. ENDFORM.	
*eject	- Ln 27 of 27 lines
	1

Page 17 of 25

For custom routines, it is acceptable to turn off the modification assistant as long as the routine number begins with a '9'.

Program F	<u>adit G</u> oto <u>U</u> tilities	Environment	System Heln			
	Pretty Printer		e 😒 i 🖵 🖨 🖧 i 🏖	ነ 🕰 🖧 🕄 🔀		
ABAP EC	<u>P</u> attern	Ctrl+F6	07A913			ALCONY MACRONICS
	<u>C</u> lipboard	► Ctrl+F		ern 🕬 🗗 Insert	t 📑 Replace 📑 Del	lete 🕞 C
	Searc <u>h</u> /replace Find ne <u>x</u> t	Ctrl+G				
Include	Con <u>t</u> inue replace		Active			
	D <u>e</u> select		* * * * * * * * * * * * * * * * * * * *	:		
*** Attent	More functions	+	***	1		
*** Attent *** Charac	Modification operati		<u>I</u> nsert line area Replace line area	Ctrl+Shift+F8 Ctrl+Shift+F6		
	C <u>a</u> ncel	F12	Delete line area	Ctrl+Shift+F7		
*			— Undo modifi <u>c</u> ation	Shift+Ctrl		
	RM BEDINGUNG_PRUEF er checks for subs		Modification overview	Ctrl+Shift+F2		
* * Goo	ods issue		Switch off <u>a</u> ssistant			
*			<u>^</u> %	*		
* Goods iss	GUNG_PRUEFEN_913. sue is not allowed	l if a credi	t block exists			
IF LIPS E * documer	EQ SPACE. nt header					
	<-CMGST CA 'B'. AGE ID 'V1' TYPE '	E' NUMBER '	183' RAISING ERROR.			
ENDIF. ELSE.						
* line it ENDIF.	tem					
ENDFORM.						
*eject						
			Li 1, Co 1		Ln 1 - Ln 27 of 27 lines	
						4 7//

Our custom logic is added to the routine. The standard logic that was already present is kept.



Step 5 Test the new routine

We have selected a delivery and pressed the Post goods issue button.

Cuthound delivery Edit	t Goto Extras Environment	Subsequent <u>f</u> unctions S <u>y</u> stem <u>H</u> elp	
		日本 (1997年) (1997	
Change Outboun	d Delivery		
9 6 1 B 2 4	3 🔄 🏷 🖫 🍙 🛛 Post g	oodş issue	
Outbound delivery	80231444	3	

6/19/2005

Page 20 of 25

Technical Solution Guide

A break point was set in the SD_DELIVERY_SUBSEQUENT_ALLOWED function module.

Debugging Edit Goto Breakpoint Settings Development System Help
◎
ABAP Debugger
The second secon
Fields Table Breakpoints Watchpoints Calls Overview Settings Main program SAPLV07A Fwed pt anthmetic Image: Sape and the settings Source code of LV07AU03 32 48 Image: Sape and the settings FUNCTION SD_DELIVERY_SUBSEQUENT_ALLOWED AND NOT (BVBPA-KUNNR IS INITIAL)). MOVE BVBPA TO VBPA. Image: Sape and the settings MOVE BVBPA TO EXP_CTR_VBPA. ENDIF. PERFORM TVFO_READ USING FOFUN. Systemprüfung IF NOT TVFO-GRPNO_SY IS INITIAL. MOVE TVFO-GRPNO_SY TO FORMAME-GRPNO. PERFORM (FORMAME) IN PROGRAM (PRO6_ORDER). ENDIF. Userprüfung IF NOT TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IS INITIAL. MOVE TVFO-GRPNO IN PROGRAM (PRO6_ORDER). ENDIF. ENDIF. ENDIF. ENDIF. ENDIF.
Field names 1 - 4
FOFUN 13 Image: Contract of the second seco
TVF0-GRPNO 913
SY-SUBRC 0 SY-TABIX 1 SY-DBCNT 1

We can see that the function was called with a function code of '13'. The logic read table TVFO and determined the system and user defined routine numbers. The logic will build the form names with these numbers and execute the system form first and then the user form.

Page 21 of 25

The program is now in our custom routine. The inventory in our 'Z' table is not enough to cover the goods issue. The logic will issue an error using the MESSAGE statement. This will be trapped and the message will be displayed on the error log.

ট Debugging <u>E</u> dit <u>G</u> oto B <u>r</u> eakpoint	<u>S</u> ettings De <u>v</u> elopme	nt S <u>y</u> stem <u>H</u> elp	[
🕑 🛛 🖉	I I 😋 🙆 😒 I 🖴 () (2 (3 (3 (🕱 🗾 🛛 🕅	
ABAP Debugger				
🖼 属 🚺 🗋 Watchpoint				
FROM zzped_ma WHERE matnr = v_qty ENDSELECT. IF sy-subrc = 0. IF (lips-bwart = ^ lips-bwart = ^ IF lips-lfimg > v	g nt_link-zz_pedng nt_link lips-matnr. = v_qty + z; 601' OR 658'). /_qty.	alls Overview Fixed ptarithm Fixed ptarithm 24 - 2ped_mat_link-zz_pedn '058' WITH lips-matnr error.	40 ▼ 🚛	
ENDFORM.				
Field names 1 - sy-subrc	4 🔽 🖸	Field con	tents	
lips-lfimg	12.000		•	
v_qty	0.000		•	
SY-SUBRC 0 SY-	TABIX 3	SY-DBCNT 0		
Sield copied into the display				۹ <i>7//</i> /

Page 22 of 25

The error generated in the routine displays on the goods movement log. The goods issue did not post.

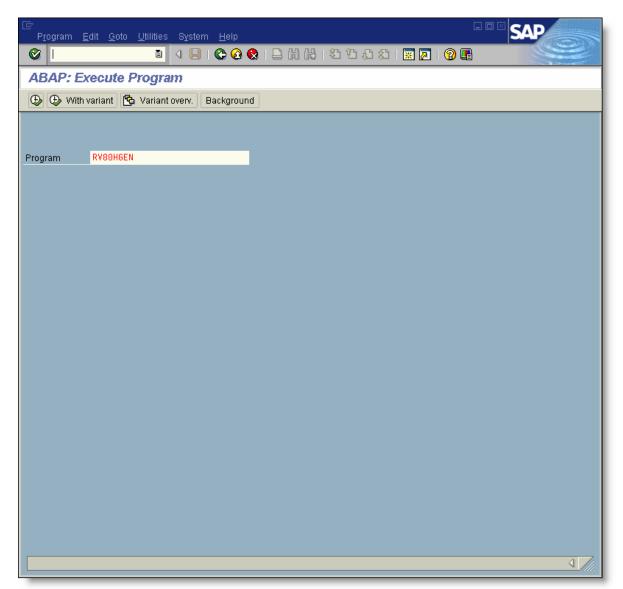
⊡ <u>E</u> dit	<u>G</u> oto S <u>y</u> stem <u>H</u> el;	J				SAP
Ø I		i 🗸 📙 i 🗞 🚱 i 🖨 🛗 🛗 i 🏵 😷 🕰 🏵	Ж 🚬	3	•	
🕄	Display Log					
Goods m	novement: Error lo	3				[
(tem	Material	Description	Delivery	qty	UoM	
	8001927 E ZV 058	PEL GEN 5 GRD 355MMX60M SP829		12	EA	686021
	Quantity exceeds :	a Pedimento inventory for material 8001927				
						4

6/19/2005

Page 23 of 25

Step 6 Running the VOFM regeneration

Whenever a VOFM routine is created, a special generation program (RV80HGEN) should be run in each system that the routine is moved to.



Note: This is a critical step that should not be overlooked. If this step is not executed, short dumps can result.

Page 24 of 25

The RV80HGEN completed.

C Log Edit Goto System Help	
Log Display	
😼 🛃 🚘 🔯 🔚 🍫 Long text	
All routines have been successfully activated	ň
	7/.

Page 25 of 25

Technical Solution Guide