

Reading Sample

This sample chapter discusses executing, configuring, and using pricing reports. It also covers the worklist provided for the internal sales representative, the use of the net price list, and the performance-optimized price list.

-  **"Condition Lists"**
-  **Contents**
-  **Index**
-  **The Authors**

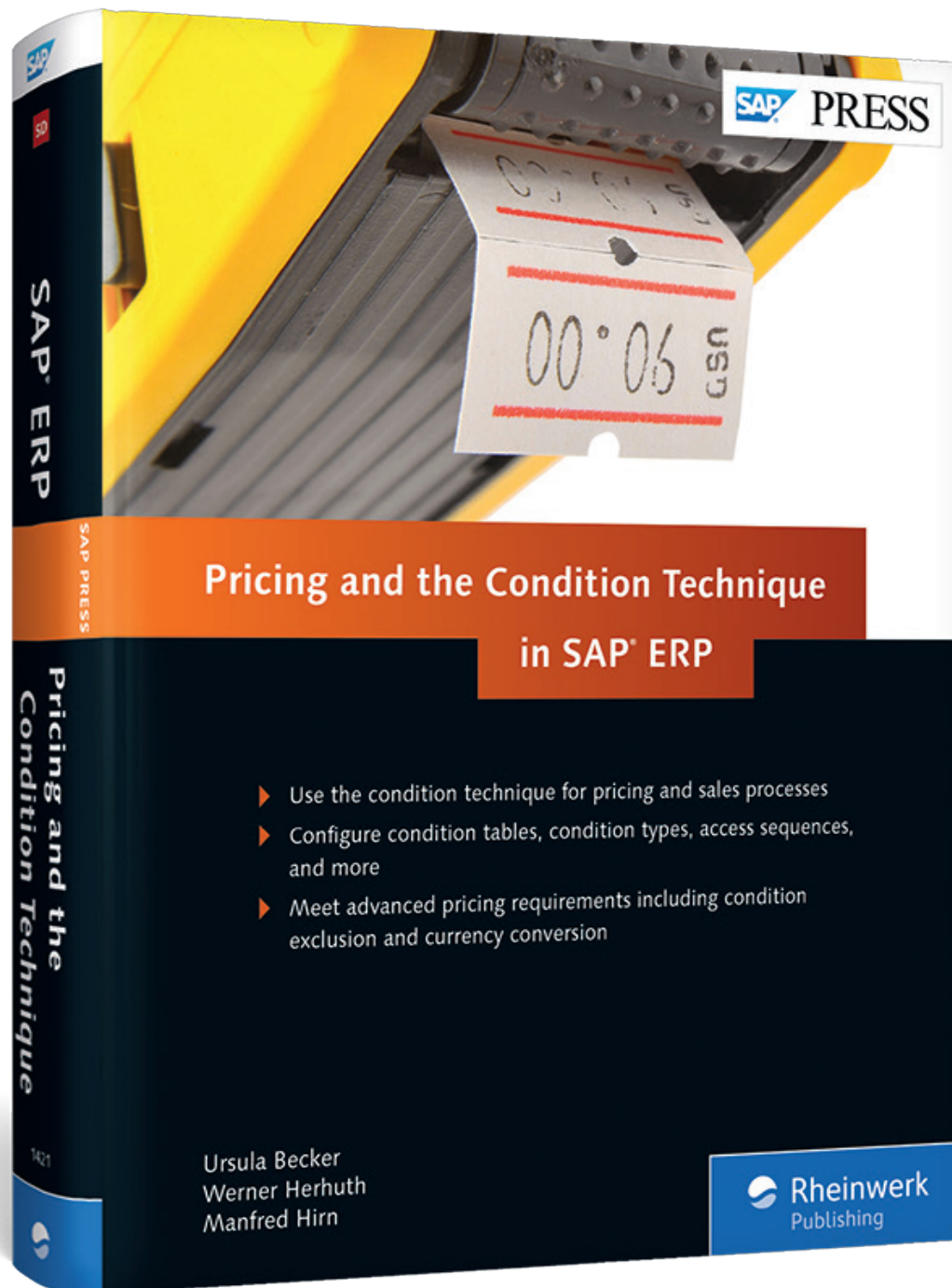
Ursula Becker, Werner Herhuth, Manfred Hirn

Pricing and the Condition Technique in SAP ERP

665 Pages, 2017, \$89.95

ISBN 978-1-4932-1421-1

 www.sap-press.com/4173



Contents

Preface by Manfred Hirn	19
Preface by Werner Herhuth	21
Preface by Ursula Becker	23
Introduction	25

PART I Foundations

1	Application Areas and Elements of the Condition Technique ...	33
1.1	Condition Records	34
1.2	Application Areas (Using Condition Tables)	35
1.3	Selected Application Areas (Usages)	37
1.3.1	Usage A: Pricing	37
1.3.2	Usage B: Output	37
1.3.3	Usage C: Account Determination	38
1.3.4	Usage D: Material Determination	38
1.3.5	Usage E: Rebate	39
1.3.6	Usage G: Material Listing and Exclusion	39
1.3.7	Usage N: Free Goods	39
1.3.8	Usage 3: Campaign Determination	40
1.3.9	Usage T: Data Collection	40
1.4	Applications of the Condition Technique	40
1.5	Elements of the Condition Technique	44
1.6	Field Catalog and Communication Structures	46
1.6.1	Field Catalog	46
1.6.2	Communication Structures	47
1.7	Condition Tables	47
1.8	Access Sequences	50
1.8.1	Access Sequence with Different Condition Tables	50
1.8.2	Access Sequence with a Single Condition Table (Access with Partial Key)	53
1.8.3	Exclusive or Additive Access	56
1.8.4	Direct Values	57
1.8.5	Data Determination During Access	57
1.9	Condition Types	58
1.10	Condition Master Data	58
1.10.1	Data Model	59

- 1.10.2 Maintenance Dialog for Simple Usages (Application Areas) 60
- 1.10.3 Maintenance Dialog for Complex Usages 61
- 1.10.4 Validity Periods 62
- 1.10.5 Release Status and Processing Status 63
- 1.11 Determination Procedures 63
- 1.12 Determination Analysis 64
- 1.13 Requirements 65
- 1.14 The Complete Process at a Glance 67
- 1.15 Summary 68

2 Pricing Condition Master Data 69

- 2.1 Condition Maintenance via Condition Type 69
 - 2.1.1 Details Functionality 71
 - 2.1.2 Additional Data Functionality 72
 - 2.1.3 Scales Functionality 73
 - 2.1.4 Condition Supplement Functionality 74
 - 2.1.5 Validity Periods Functionality 74
 - 2.1.6 Free Goods Functionality 74
 - 2.1.7 Status Information Functionality 74
 - 2.1.8 Cumulative Values Functionality 75
 - 2.1.9 Other Condition Type Functionality 75
- 2.2 Condition Maintenance via Area Menu 75
- 2.3 Condition Maintenance via Index 79
- 2.4 Agreements 80
 - 2.4.1 Sales Deals 80
 - 2.4.2 Rebate Agreements 83
- 2.5 Price Agreements 84
- 2.6 Summary 84

3 Condition Lists 85

- 3.1 Pricing Reports 85
 - 3.1.1 Execution of Pricing Reports 86
 - 3.1.2 Configuration of a Pricing Report 87
 - 3.1.3 Pricing Report for Condition Maintenance via Area Menu 91
- 3.2 Worklists for the Internal Sales Representative Role 92
 - 3.2.1 Change Query 93
 - 3.2.2 Personalize Query 95

- 3.2.3 Define New Query 96
- 3.3 Net Price List 97
- 3.4 Performance-Optimized Price List 99
 - 3.4.1 Creation of Price Lists 100
 - 3.4.2 Customizing and Customer Enhancements 105
 - 3.4.3 Registration of Fields 105
 - 3.4.4 Define the Data Determination 108
 - 3.4.5 Enhancements of the Selection Screen 114
 - 3.4.6 Enhancements of the Price List Result 118
 - 3.4.7 Customizing Overview 122
 - 3.4.8 Important Programs and Function Modules 124
- 3.5 Summary 125

PART II Standard Configuration

4 Pricing Configuration 129

- 4.1 Implementation of an Initial Customer-Specific Scenario 130
 - 4.1.1 Creating a Condition Table 130
 - 4.1.2 Creating an Access Sequence 135
 - 4.1.3 Customizing a Condition Type 138
 - 4.1.4 Creating a Condition Type 141
 - 4.1.5 Inserting a Condition Type into a Pricing Procedure 143
 - 4.1.6 Defining the Pricing Procedure Determination 144
 - 4.1.7 Testing the Pricing Scenario 145
- 4.2 Elements of Pricing in Detail 146
 - 4.2.1 Condition Table 146
 - 4.2.2 Access Sequence 148
 - 4.2.3 Condition Type 149
 - 4.2.4 Pricing Procedure 161
- 4.3 Customer-Specific Adaptations and Customer Namespaces 166
 - 4.3.1 Creative Use of the Standard Configuration 166
 - 4.3.2 New Fields for Pricing 167
 - 4.3.3 Requirements and Formulas (Routines) 168
 - 4.3.4 User Exits for Pricing 169
- 4.4 Example for the Implementation of Complex Pricing Scenarios 170
- 4.5 Configurable Parameters and Formulas 172
 - 4.5.1 Implementing an Initial Scenario with Configurable Parameters and Formulas 173
 - 4.5.2 Elements of CPF in Detail 185
 - 4.5.3 Customer-Specific Adaptations in CPF 191

4.6	Interface to Profitability Analysis	198
4.7	Summary	200

5 Working with Condition Records 201

5.1	Release Status	201
5.1.1	Available Release Statuses	202
5.1.2	Release Status in Condition Records	202
5.1.3	Release Status in Conjunction with Agreements	203
5.2	Mass Change/Mass Creation with Template	206
5.3	Copying Condition Records	210
5.3.1	Prerequisites	210
5.3.2	Application Example	213
5.4	Condition Maintenance via Area Menus	215
5.5	Other Functionalities in Condition Maintenance	217
5.5.1	Change Documents	217
5.5.2	Texts	218
5.5.3	Condition Supplements	220
5.5.4	Deletion of a Condition Record	220
5.6	Summary	222

6 Pricing in the Sales Document 223

6.1	The Item Conditions Screen	223
6.1.1	Condition Detail	226
6.1.2	Insert Row	229
6.1.3	Delete Row	229
6.1.4	Master Data Condition Record	229
6.1.5	Analysis (Pricing Log)	229
6.1.6	Update Prices	230
6.1.7	Configurable User Interface for Pricing	230
6.2	Header Conditions Screen	232
6.3	Predefined Price Elements in the Item Overview Screen	236
6.4	Price Agreements	237
6.4.1	Creating Price Agreements	237
6.4.2	Searching and Displaying Price Agreements Using the Condition Index	240
6.5	Pricing During Creation of Documents with Reference	240
6.6	Pricing for Special Billing Scenarios	241
6.6.1	Pricing in Rental and Maintenance Contracts (Periodic Billing Plan)	242

6.6.2	Pricing in Fixed Price Contracts (Milestone Billing)	246
6.6.3	Pricing in Resource-Related Billing	252
6.7	Summary	253

7 Special Functions in Pricing 255

7.1	Group Conditions	255
7.1.1	Header Conditions with a Fixed Amount	256
7.1.2	Percentage-Based Header Conditions	256
7.1.3	Scale-Based Conditions without Group Key	257
7.1.4	Scale-Based Conditions without Group Key	257
7.1.5	Percentage-Based and Fixed Amount Group Conditions with Condition Records	257
7.2	Condition Exclusion	258
7.2.1	Simple Condition Exclusion Using Requirements	258
7.2.2	Condition Exclusion Using Exclusion Groups	259
7.2.3	Condition Exclusion Using Formulas	261
7.3	Condition Update	264
7.4	Tax Determination in Sales	267
7.4.1	Simple Tax Determination	267
7.4.2	Tax Determination via Accounting (Tax Trigger)	269
7.4.3	Tax Determination via External Tax Interface	272
7.4.4	Tax Increase	273
7.5	Free Goods in Sales	273
7.6	Cost of Sales	276
7.6.1	Stock Sale (No Cross-Company)	277
7.6.2	Drop Shipment (Third Party)	278
7.6.3	Cross-Company Stock Sale	278
7.6.4	Make-To-Order with/without Production Order	280
7.6.5	Actual-Cost Accounting	280
7.7	Currency Conversion in Sales	281
7.7.1	Local Currency	282
7.7.2	Document Currency and Exchange Rates	282
7.7.3	Exchange Rate for Accounting	283
7.7.4	Condition Currency and Exchange Rates	285
7.7.5	Currency Customizing and Exchange Rates	286
7.7.6	Currency Conversion	286
7.8	Print Formatting of the Pricing Result	289
7.8.1	Functional Scope of the Output Processing	289
7.8.2	Print Formatting of Order Item Values and Document Footer	291
7.9	Summary	293

8 Pricing Procedures and Condition Types in Sales and Distribution	295
8.1 Selected Pricing Procedures	297
8.1.1 Pricing Procedure RVAA01 (Standard B2B)	298
8.1.2 Pricing Procedure RVAA02 (Standard with Price Book) ...	303
8.1.3 Pricing Procedure RVAB01 (Tax Included in Price)	306
8.1.4 Pricing Procedure PSER01 (Periodic Billing)	307
8.1.5 Pricing Procedure PSER02 (Resource Related Billing)	308
8.1.6 Pricing Procedure ICAA01 (Intercompany Billing)	309
8.1.7 Pricing Procedure WK0001 (Value Contract Procedure)	311
8.1.8 Pricing Procedure RVWIA1 (Plants Abroad/ Tax Handling)	311
8.1.9 Pricing Procedure RVACRN (Contract Billing)	311
8.2 Selected Condition Types of the Standard Pricing Procedures	314
8.2.1 Condition Types EDI1/EDI2 (Customer Expected Price) ...	314
8.2.2 Condition Types RLOO/MW15 (Invoice List Conditions) ...	315
8.2.3 Condition Types SKTV/SKTO (Cash Discount)	315
8.2.4 Condition Type HM00 (Order Value)	316
8.2.5 Condition Types AMIW/AMIZ (Minimum Sales Order Value/Minimum Value Surcharge)	317
8.2.6 Condition Type AZWR (Down Payment/Settlement)	318
8.2.7 Condition Type GRWR (Statistical Value (Cross-Border))	318
8.3 Summary	319

PART III Advanced Techniques, Tips, and Tricks

9 Special Features of the Condition Technique in Pricing	323
9.1 Interfaces, Tables, and Interrelationships	324
9.1.1 Interfaces to Condition Tables and Master Data Maintenance	324
9.1.2 Interfaces to the Pricing Modules	325
9.1.3 Internal Structures and Tables of the Pricing Programs ...	326
9.1.4 Field Sequence in the Condition Table	328
9.2 Data Determination via Condition Technique	329
9.2.1 Variant 1: Data Determination Within the Access: Used in Subsequent Access	330

9.2.2 Variant 2: Data Determination in the Access, Transfer to Document Item via Requirement	335
9.2.3 Transfer from Condition Record via Formula or Requirement	344
9.2.4 Variant 3: Transfer to XKOMV from Condition Record via Formula	347
9.2.5 Variant 4: Transfer to KOMK from Condition Record via Requirement	348
9.2.6 Variant 5: Utilization of Data of Other Document Conditions in Formula	349
9.3 Fields with Multiple Values	350
9.3.1 The KOMPLOOP Variant for Multiple Value Fields	350
9.3.2 The Repeating Field Variant for Multiple Value Fields	353
9.3.3 Variant Using BADL SD_COND_LOAD_MVA	353
9.4 Summary	354

10 Important Programs for Pricing **355**

10.1 The PRICING Function Module	356
10.1.1 Routine PREISFINDUNG_VORBEREITEN (Pricing Prepare)	358
10.1.2 Routine KONDITIONSVORSTEP (Condition Prestep)	359
10.1.3 Routine KONV_EINLESEN (Read KONV)	363
10.1.4 Routine XKOMV_AUFBAUEN_AUS_KOMT1 (Set Up XKOMV from KOMT1)	363
10.1.5 Routine XKOMV_AUFBAUEN_AUS_TKOMV (Set Up XKOMV from TKOMV)	366
10.1.6 Routine XKOMV_BEWERTEN (Evaluate XKOMV)	366
10.1.7 Routine XKOMV_AUSSCHLUSS (XKOMV Exclusion)	367
10.1.8 Routine XKOMV_UEBERTRAGEN_NACH_TKOMV (Transfer XKOMV to TKOMV)	367
10.2 The PRICING_COMPLETE Function Module	367
10.2.1 Types of Group Conditions	368
10.2.2 Execution of Function Module PRICING_COMPLETE during Order Processing	370
10.2.3 Program Flow of the PRICING_COMPLETE Function Module	372
10.3 The PRICING_COPY Function Module	378
10.4 Data Storage of the Pricing Result in Clustered Table KONV	379
10.5 The PRICING_REFRESH Function Module	380

- 10.6 Dialog Connection/Other Function Modules 381
 - 10.6.1 Function Module PRICING_LOAD_CONDITION_TYPS ... 381
 - 10.6.2 Dialog Connection 381
 - 10.6.3 The PRICING_DETERMINE_DATES Function Module 382
- 10.7 Relationship of Tables and Structures 383
- 10.8 Summary 384

11 System Adaptation Using Requirements, Formulas, and User Exits 385

- 11.1 Pricing Types 387
 - 11.1.1 Control via KOMP-KAEND_TYP 387
 - 11.1.2 Control via KOMP-KONAU_TYP 388
- 11.2 Requirements 388
 - 11.2.1 Manipulation of KOMK Fields and KOMP Fields via Requirements 388
 - 11.2.2 Overriding Pricing Types via Requirements 389
 - 11.2.3 Example 1: Using Table TKSCHLEXCL 389
 - 11.2.4 Example 2: Using Tables TKSCHL and TKSCHLEXCL 390
 - 11.2.5 Processing Time of the Prestep-Requirement (KOBEP Part) 390
 - 11.2.6 Processing Time of the Definite Requirement (KOBED Part) 391
 - 11.2.7 Behavior for Copied Conditions 392
- 11.3 An Example Using Formulas 392
- 11.4 Formulas for Pricing Conditions 398
 - 11.4.1 Condition Base Value Formulas 398
 - 11.4.2 Scale Base Formulas 400
 - 11.4.3 Condition Value Formulas 402
 - 11.4.4 Formulas for the Structure of Group Keys (Group Key Routines) 404
- 11.5 Special Logic for Pricing Type F in Routine XKOM_BEWERTEN ... 408
 - 11.5.1 Point in Time "A" 409
 - 11.5.2 Point in Time "B" 409
 - 11.5.3 Point in Time "C" 410
 - 11.5.4 Point in Time "D" 410
- 11.6 Pricing Result in KOMP 410
 - 11.6.1 KOMP-FXMSG (Message Number) 412
 - 11.6.2 KOMP-PRSOK (Pricing Is OK) 412
 - 11.6.3 KOMP-CEPOK (Status Expected Price) 413
 - 11.6.4 KOMP-MPROK (Status Manual Price Change) 413
 - 11.6.5 KOMP-KOUPD (Condition Update) 414

- 11.6.6 KOMP-NETPR (Net Price) 414
- 11.6.7 KOMP-NETWR (Net Value of the Item) 414
- 11.6.8 KOMP-MWSBP (Tax Amount of the Item) 414
- 11.6.9 KOMP-WAVWR (Cost of the Item) 414
- 11.6.10 KOMP-BRTWR, KZWI1-KZWI6, Etc. (Subtotals) 415
- 11.7 Control Flag XKOMV-KSTEU 416
- 11.8 User Exits 418
 - 11.8.1 User Exits of Function Group V61A 419
 - 11.8.2 User Exits of Function Group V69A 424
- 11.9 Enhancement of KONV 426
- 11.10 Performance Aspects 427
- 11.11 Error Messages/Error Handling 428
- 11.12 Summary 429

12 Typical Practice Demands on Pricing and Their Solutions 431

- 12.1 Budgeting Requirements 432
 - 12.1.1 Statistics Tables S060 and S469 433
 - 12.1.2 Statistics Table S071 434
 - 12.1.3 Budgeting Requirements 434
- 12.2 Why Is the Condition Not Found? 442
- 12.3 Rounding 445
- 12.4 Prices with More Than Two Decimal Places 446
- 12.5 Handling Freight Surcharges 447
 - 12.5.1 Complete Billing of Freight Surcharges with the First Delivery 447
 - 12.5.2 No Freight Surcharges for Free Goods and Returns 448
- 12.6 Authorizations for the Pricing Screen 449
- 12.7 Adding Subtotal Fields 449
- 12.8 Date-Dependent Maintenance of Master Data Fields 450
 - 12.8.1 Data Determination for the Sales Document Item VBAP ... 451
 - 12.8.2 Data Determination for the Sales Document Header VBAK/Business Data VBKD 453
 - 12.8.3 Example: Substitution of a Customer Group (VBKD-KDGRP) 453
 - 12.8.4 Pros and Cons of Data Determination 454
 - 12.8.5 Data Determination for the Billing Documents 455
- 12.9 Copied Conditions and Subsequent Quantity Changes 455
 - 12.9.1 Simple Solution for Fixed Amount Condition Types 456
 - 12.9.2 Complex Solution for Any Condition Type 458
- 12.10 Increased Prices in Returns and Credit Notes 462

- 12.11 Key Figures for Reporting and Analysis 463
 - 12.11.1 Implementation of a Consolidated Pricing Procedure 465
 - 12.11.2 Key Figure 1: The Initial Value (KZWI1) 470
 - 12.11.3 Key Figure 2: The Standard Condition Types (KZWI2) 472
 - 12.11.4 Key Figure 3: The Customer Specific Condition Types (KZWI3) 472
 - 12.11.5 Key Figure 4: Manual Condition Types (KZWI4) 474
 - 12.11.6 Key Figure 5: The Rebate Condition Types (KZWI5) 475
 - 12.11.7 Key Figure 6: The Net Value (NETWR) 475
 - 12.11.8 Key Figure 7: The Cost (WAVWR) 475
- 12.12 Condition Record Depending Requirements 479
 - 12.12.1 Creation of a New Usage Task 479
 - 12.12.2 Exclusion of Particular Material Groups in a Condition Record 485
- 12.13 Summary 492

13 Pricing in Selected Applications 493

- 13.1 Pricing in Sales Orders 494
 - 13.1.1 Routine USEREXIT_PRICING_PREPARE_TKOMK 496
 - 13.1.2 Routine USEREXIT_PRICING_PREPARE_TKOMP 497
 - 13.1.3 Reacting to the Changes of Document Contents 497
 - 13.1.4 Manual Triggering of New Pricing in the Sales Order 499
- 13.2 Pricing in Billing Documents 500
- 13.3 Pricing in Purchase Orders 501
- 13.4 Pricing in Accounting 505
- 13.5 Tax Determination in Accounting 506
- 13.6 Pricing in Transport Management (Shipment Cost Calculation) 513
- 13.7 Summary 514

14 Performance and Testing 515

- 14.1 SAP Table Buffering 516
- 14.2 Prestep Conditions and Access Optimization 517
- 14.3 Field Sequence in the Condition Tables 520
- 14.4 Usage of Requirements 521
- 14.5 Group Conditions and Formulas 521
- 14.6 Peculiarities in Sales Order and Billing Document 522
 - 14.6.1 Peculiarities in the Sales Order 522
 - 14.6.2 Peculiarities in the Billing Document 523
 - 14.6.3 Filling the Communication Structure KOMK 526

- 14.7 Analytical Tools 529
- 14.8 Testing 531
- 14.9 Summary 533

PART IV Rebate Processing in Sales

15 Rebate Processing in Sales 537

- 15.1 Overview of Rebate Processing 537
 - 15.1.1 Creating a Rebate Agreement 538
 - 15.1.2 Sales Volume: Rebate Condition Types in the Sales Document 541
 - 15.1.3 Settlement of Rebate Agreements 547
 - 15.1.4 Deletion of Rebate Agreements 555
 - 15.1.5 Retroactive Changes and Rebate Index 555
- 15.2 Elements of Rebate Processing in Detail 560
 - 15.2.1 Condition Technique for Rebate Processing 560
 - 15.2.2 Rebate Agreements 563
 - 15.2.3 Activation of Rebate Processing 569
 - 15.2.4 Document Types for Rebate Processing 570
 - 15.2.5 Activation of the New Rebate Method 571
 - 15.2.6 Important Reports 571
 - 15.2.7 Scale Base Determination Based on Grouped Condition Records 579
 - 15.2.8 Replacement of the Rebate Index Table 580
- 15.3 Extended Rebate Processing 581
 - 15.3.1 Implementation of a First Customer-Specific Scenario 582
 - 15.3.2 Customizing the Extended Rebate Processing 588
- 15.4 System Adaptation in Rebate Processing 594
 - 15.4.1 Business Transaction Event 00503201 594
 - 15.4.2 BAdI BADI_SD_REBATES 595
 - 15.4.3 BAdIs of Extended Rebate Processing 597
- 15.5 Typical Practice Demands 600
 - 15.5.1 Exclusion 600
 - 15.5.2 Settlement Material and Tax Classification 601
 - 15.5.3 External Control of Rebate Processing 601
- 15.6 Summary 603

- A Condition Type Profiles 605
- B The Authors 653

- Index 655

Like any other master data, condition records must also be maintained by the respective departments of the company. Efficient tools can select the relevant subset from a large population of condition records to assist in this task.

3 Condition Lists

In addition to the different maintenance transactions, additional tools are provided to maintain condition records and to evaluate them in order to answer different questions. For example, you can use pricing reports or personalized worklists to get an overview of condition records or use price lists for price information.

In the following sections, we will take a look at executing, configuring, and using pricing reports. We will also discuss worklist provided for the internal sales representative role (SAP_BPR_INT_SALES_REP_14 or SAP_SR_INT_SALES_REP_5), the use of the net price list, and the performance-optimized price list.

3.1 Pricing Reports

To get an overview of the condition records in the system, you can use special pricing reports. For example, the condition records can be searched to answer the following questions:

- ▶ Which customer-specific price agreements exist in my sales organization for specific products?
- ▶ Which condition records are stored in the system for certain *incoterms*?
- ▶ What conditions have been created for a specific customer?

In the standard SAP system, there are a number of preconfigured pricing reports, such as pricing report 16 (INDIVIDUAL PRICES). In addition, you can configure further pricing reports by a report generator within Customizing (see Section 3.1.2)

to cover any customer requirements. Let us now take a look at how to handle pricing reports.

3.1.1 Execution of Pricing Reports

To call a pricing report, choose the menu path SAP EASY ACCESS • LOGISTICS • SALES AND DISTRIBUTION • MASTER DATA • CONDITIONS • LIST • PRICING REPORT (Transaction V/LD).

Let's look at this tool using the example of pricing report 16 (see Figure 3.1): Initially, after starting a pricing report, a selection screen appears to restrict the query. The number of selection fields can be configured, as you will also see in Section 3.1.2, when creating the pricing report in Customizing. You can also configure the default for the attributes in the field group LIST SCREEN.

Individual Prices

Sales Organization: 1000 to []

Distribution Channel: [] to []

Customer: [] to []

Material: [] to []

Release status: [] to []

Condition Type: [] to []

Validity period

Validity range: 16.04.2016 to 31.12.9999

Condition records exceeding interval named above

at start of validity period

at end of validity period

List screen

Display scales

Display validity period

Additional condition fields

Cond. marked for deletion

Exclusive

Max. hits per table: 500

Figure 3.1 Calling the Individual Prices Pricing Report (Transaction V/LD)

After entering the selection criteria, you can start the pricing report by selecting EXECUTE or using **F8**. The results will be displayed (see Figure 3.2). Within the possibilities offered, the appearance of the result list (e.g., positioning a field on the page header, group header, or item level) can be configured in the Customizing of the pricing report.

Individual Prices

Sales Org. 1000 Germany Frankfurt
Distr. Channel 10 Final customer sales

Customer	CnTy	Material	ReSt	S	Scale	qty	UoM	Amount	Unit	per	UoM	Valid From	Valid to
1234		K.F.W. Berlin											
PRO0	QS8X35					0,10	EUR	1	PC			15.11.2000	31.12.9999
1235		K.F.W. London											
PRO0	QS8X30					0,10	EUR	1	PC			16.11.2000	31.12.9999
PRO0	QS8X35					0,10	EUR	1	PC			15.11.2000	31.12.9999
1250		LCH Markt											
PRO0	T-FV100-EM-01					9,00	EUR	1	KG			30.01.2003	31.12.9999
1410		PILAR am Neckar											
PRO0	P-109					3.323,00	EUR	1	PC			26.07.2000	31.12.9999
CUSTOMER00	Becker 00												
PRO0	P-W0-200					5,00	EUR	1	PC			20.05.2010	31.12.9999

Figure 3.2 Individual Prices: Result List

From the results list, you can branch to the condition maintenance Transactions VK11 (Create), VK12 (Change), and VK13 (Display). In the case of display and change, you jump directly to the condition record marked in the list.

3.1.2 Configuration of a Pricing Report

The configuration of pricing reports cannot be done in the application menu of the end user, but it is part of Customizing. The settings for the pricing reports can be accessed via the menu path IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • MAINTAIN PRICING REPORT.

To create a new pricing report, you first select a two-digit abbreviation from the customer namespace (these are, by definition, all two-digit combinations that begin with a letter) and a title. In our example report, we will choose the abbreviation X0 and the title CONDITIONS PER CUSTOMER (see Figure 3.3).

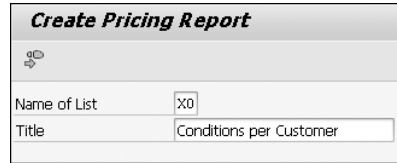


Figure 3.3 Create Pricing Report: Initial Screen

Next, a screen appears with all fields that are part of a condition table in the underlying development system (see Figure 3.4).

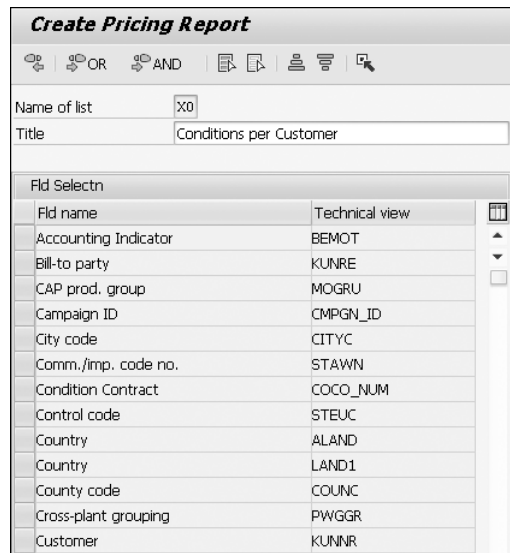


Figure 3.4 Create Pricing Report: Field Selection

When selecting the fields, the challenge is to make a preliminary decision: Which condition tables and condition types are to be evaluated in the resulting pricing report? Here, a combination of the fields with AND or with OR is possible. Each option has an impact on the selection of the condition tables. We opt for the CUSTOMER (KUNNR) field. As a result, we receive a listing of all condition tables in which field KUNNR is included, as shown in Figure 3.5.

We select condition tables 005, 007, 305, and 307 and come to the last screen in the generation sequence (see Figure 3.6), where we can influence the selection screen and the appearance of the result list.

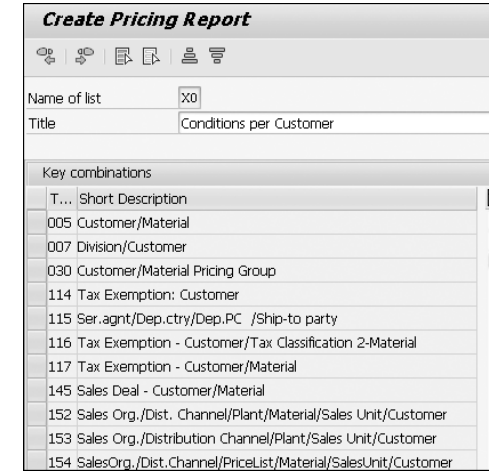


Figure 3.5 Create Pricing Report: Selection of Condition Tables

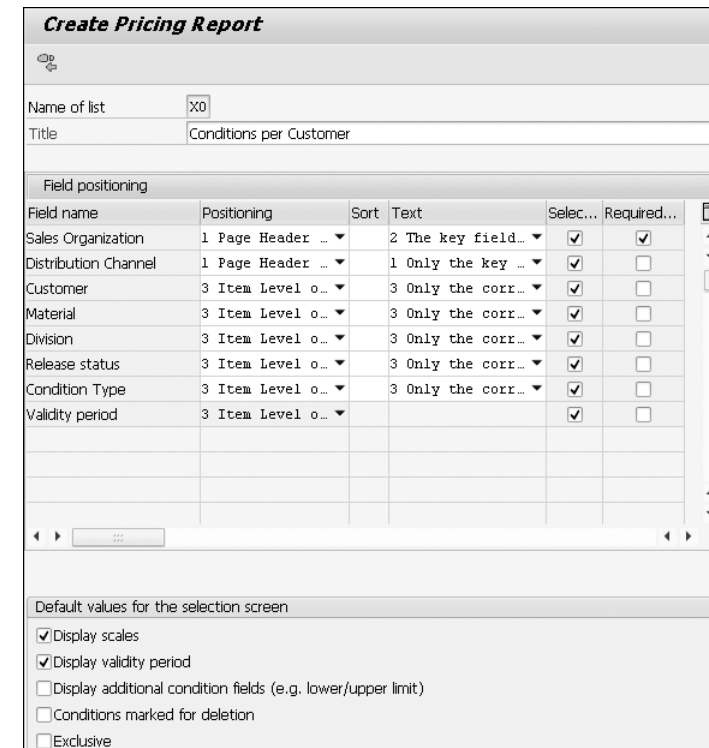


Figure 3.6 Create Pricing Report: Appearance of the Result List

The system offers the key fields of all previously selected condition tables as selection fields. In the POSITIONING column, you can choose between three options:

- ▶ 1: PAGE HEADER OF PRICING REPORT
- ▶ 2: GROUP HEADER OF PRICING REPORT
- ▶ 3: ITEM LEVEL OF PRICING REPORT

In the SORT column, the order of the fields in a level (page header, group header, or item level) can be set. The sort order does not affect the order of the page header, group header, and item level itself.

In the TEXT column, you also have three options available:

- ▶ 1: ONLY THE KEY FIELD IS DISPLAYED
- ▶ 2: THE KEY FIELD AND THE CORRESPONDING TEXT ARE DISPLAYED
- ▶ 3: ONLY THE CORRESPONDING TEXT IS DISPLAYED

These options apply only to the levels of group header and item level. In contrast, in the page header, the field name and description are always displayed. Due to lack of space, either the field name or the associated text should be displayed at the item level.

In the SELECTION column, you can deselect some of the selection fields as necessary, though we have not done so in our example. In the REQUIRED INPUT column, the SALES ORGANIZATION field has been marked as a required entry field. Click SAVE to generate the new pricing report X0. A first test run of the new pricing report shows the result displayed in Figure 3.7.

Conditions per Customer												
Sales Org. 1000 Germany Frankfurt Distr. Channel 12 Sold for resale												
Condition type	Dv	Customer	\$	Scale qty	UoM	Amount	Unit	per	UoM	Valid From	Valid to	
Customer Discount	Cross-division	Motomarkt Stuttgart GmbH				10,000-	%			01.03.2015	31.12.9999	
Sales Org. 1000 Germany Frankfurt Distr. Channel 12 Sold for resale												
CnTy	Customer	Material	ReSt	\$	Scale qty	UoM	Amount	Unit	per	UoM	Valid From	Valid to
Price	Motomarkt Stuttgart GmbH	CrossFun / 350 cm3				6.500,00	EUR	1	PC	01.04.2016	31.12.9999	

Figure 3.7 Conditions per Customer: First Test Run

Since the appearance of the result list is not completely in line with our expectations, we edit the pricing report again in change mode and replace the attributes in the TEXT column labeled 3 - ONLY THE CORRESPONDING TEXT IS DISPLAYED with the attribute 1 - ONLY THE KEY FIELD IS DISPLAYED. A new call of the pricing report X0 shows the result displayed in Figure 3.8.

Conditions per Customer												
Sales Org. 1000 Germany Frankfurt Distr. Channel 12 Sold for resale												
CnTy	Dv	Customer	\$	Scale qty	UoM	Amount	Unit	per	UoM	Valid From	Valid to	
K007		1174				10,000-	%			01.03.2015	31.12.9999	
Sales Org. 1000 Germany Frankfurt Distr. Channel 12 Sold for resale												
CnTy	Customer	Material	ReSt	\$	Scale qty	UoM	Amount	Unit	per	UoM	Valid From	Valid to
PR00	1174	1400-310				6.500,00	EUR	1	PC	01.04.2016	31.12.9999	

Figure 3.8 Conditions per Customer: Second Test Run

Usually several change rounds will be necessary to customize the options of the result list according to your needs.

3.1.3 Pricing Report for Condition Maintenance via Area Menu

As described in Chapter 2, the pricing reports play an important role in the condition maintenance via area menus. They act as filters, which are assigned to a node of the area menu in the form of a variant of a pricing report (e.g., in area menu COND_AV).

If you want to use a pricing report as a filter in an area menu, no specific settings are required when generating the report. However, the condition maintenance Transactions VK31 to VK34 provide an adapted selection screen that is controlled by them. Therefore, the selection screen shown in Figure 3.1 of pricing report 16 (INDIVIDUAL PRICES) looks slightly different when used as a filter in the condition maintenance (see Figure 3.9).

Figure 3.9 Pricing Report as a Filter in the Condition Maintenance

Pricing reports are by themselves very useful and flexible tools made even more valuable by their use as a filter element in the condition maintenance via area menus.

The latest development in condition lists are the *worklists* introduced for the internal sales representative role, which are presented next.

3.2 Worklists for the Internal Sales Representative Role

Since SAP ERP 6.0 EHP 2, the *Internal Sales Representative* business package has been available. To use it, however, you must activate the business function SD_01 (Logistics: S&D Simplification) with Transaction SFW5 (Switch Framework).

In the accompanying supplied sample user role SAP_BPR_INT_SALES_REP_14 (or SAP_SR_INT_SALES_REP_5 later with EHP 5), a personal worklist for condition information is contained. This worklist provides the sales staff with a flexible overview of the different types of conditions as well as extensive search facilities to identify condition master data according to different selection criteria. The individual queries are provided in the form of personal worklists—also called POWLs (personal object worklists). The salesperson receives different lists with the customer-specific agreements such as customer-specific prices, discounts, and free goods. In addition, lists are available with material prices, freight costs, and price lists.

This POWL can be adjusted individually by the user with regard to their use of selection criteria (CHANGE QUERY), the visibility and the name of the queries (PERSONALIZE), and the presentation of the search result (SETTINGS). In addition, users can even create their own queries (DEFINE NEW QUERY).

Initially, the functionality of the internal sales representative role was exclusively available via portal usage. Starting with SAP ERP 6.0 EHP 4, there is also the option to use the role with the SAP Business Client. The functionality relating to the worklist for condition information is identical. Figure 3.10 shows the worklist for condition information after accessing the system with the SAP Business Client and choosing menu node PRICES.

Condition type	Sales Org.	Distr. Channel	Material	Release status	Proc. status	Amount	Unit	per	UoM	Valid From	Valid to
FR00	1000	10	DFC1002			220,90	EUR	1	PC	14.09.2002	31.12.9999
FR00	1000	10	DFC1003			280,90	EUR	1	PC	14.09.2002	31.12.9999
FR00	1000	10	DFC1004			310,20	EUR	1	PC	14.09.2002	31.12.9999
FR00	1000	10	DFC1005			259,60	EUR	1	PC	14.09.2002	31.12.9999

Figure 3.10 Screen of the Internal Sales Representative Role

To ensure that you receive insight into the capabilities of POWL, we will discuss the features of this list in the following sections, using a practical example.

3.2.1 Change Query

The queries in the personal worklist should of course reflect the specific information needs of the user. You can adjust existing queries by using the function CHANGE QUERY (see Figure 3.10). In our example, we select the query SEARCH MATERIAL PRICES in the screen area ACTIVE QUERIES and choose the function

CHANGE QUERY. As a result, a selection screen opens with a large number of possible selection fields, which you can fill in with any combinations (see Figure 3.11).

Figure 3.11 Selection Fields of the Search Prices Query

The available selection fields are defined in the configuration of the underlying POWL. This configuration can be reached via the menu path IMG • SALES AND DISTRIBUTION • "INTERNAL SALES REPRESENTATIVE" ROLE • PERSONAL WORKLIST (NEW) • PERSONAL WORKLIST FOR CONDITION INFORMATION.

The main worklist types for sales employees' work areas are predefined and assigned to SAP ERP Sales and Distribution (SD):

- ▶ SALES: CUSTOMER REBATES
- ▶ SALES: CUSTOMER FREE GOODS
- ▶ SALES: CUSTOMER PRICES
- ▶ SALES: DOCUMENT PRICES
- ▶ SALES: FREIGHT
- ▶ SALES: MATERIAL PRICES
- ▶ SALES: PRICE LIST

Customer-Specific Worklist Types

In addition to the predefined worklist types, customer-specific worklist types can be created. Nevertheless, for this purpose, Business Add-ins (BADIs) must be programmed!

As you can see on closer inspection in Figure 3.11, the underlying worklist type for the query SEARCH PRICES is the type SALES: CUSTOMER PRICES.

3.2.2 Personalize Query

To personalize a query, select the PERSONALIZE function in the initial screen of the query (see Figure 3.10). This will open the screen shown in Figure 3.12. You can enable or disable the available queries and design your worklist.

Figure 3.12 Personalize Query

3.2.3 Define New Query

To search condition records according to other aspects, you can create new queries. After selecting the DEFINE NEW QUERY function, a wizard starts that guides the user in three steps through the creation of the new query.

First, select the object type of the query (see Figure 3.13).

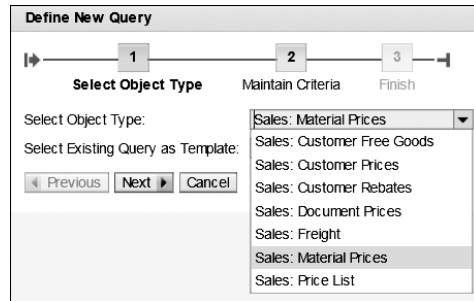


Figure 3.13 Define New Query: Step 1

The second step is in the input of the selection criteria (see Figure 3.14). As already mentioned, the selected query type defines these.

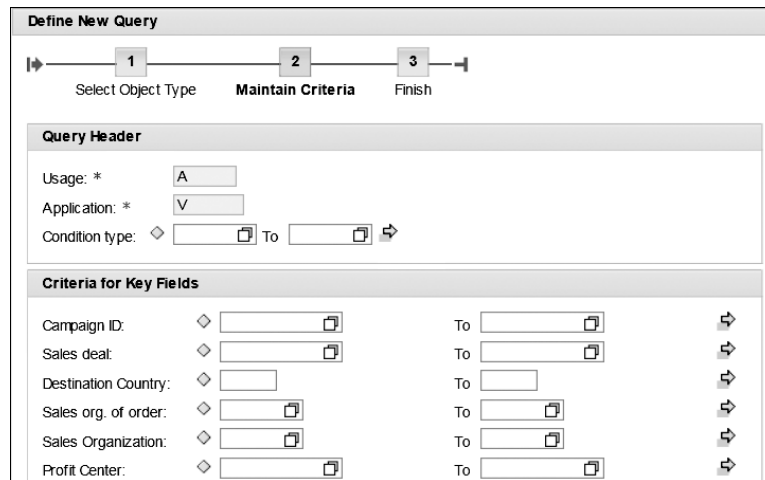


Figure 3.14 Define New Query: Step 2

The third and final step is to specify the description and the category under which the query should appear in the worklist (see Figure 3.15).

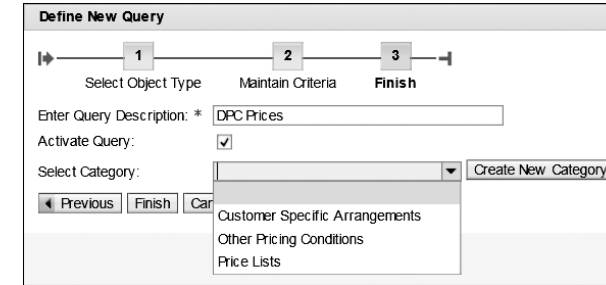


Figure 3.15 Define New Query: Step 3

After you click on the FINISH button, you can use the new query in the worklist (see Figure 3.16).

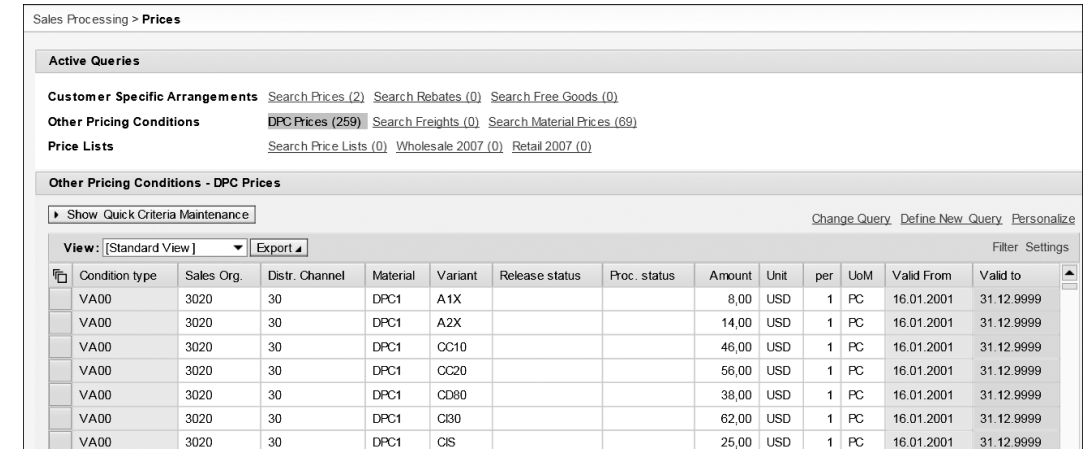


Figure 3.16 New Query in the Worklist

As you can see, it can be worthwhile to use the Internal Sales Representative business package. In addition to the worklist for price data, it contains many other interesting areas, such as the customer cockpit. For more information on this, we recommend a look at the SAP ERP documentation at <http://help.sap.com>.

3.3 Net Price List

The net price list provides price information on a selected set of products for a specific customer. The net prices cannot simply be read from a master data table,

but instead must be calculated. For this purpose, an invoice is simulated (using function module `GN_INVOICE_CREATE`) and a corresponding pricing run is performed. Therefore, in the selection screen of the net price list, all information must be entered that is necessary for a specific pricing procedure determination: sales area, sales document type (from which the `DOCUMENT PRICING PROCEDURE` field is derived), and the customer (from which the `CUSTOMER PRICING PROCEDURE` field is derived).

The net price list can be accessed via the menu path `SAP EASY ACCESS • LOGISTICS • SALES AND DISTRIBUTION • MASTER DATA • CONDITIONS • LIST • NET PRICE LIST` (Transaction `V_NL`).

A selection screen is displayed (see Figure 3.17), and the previously mentioned selection criteria must be entered for the pricing procedure determination.

Create net price list

Organizational Data

Sales Organization

Distribution Channel

Division

Customer Data

Sold-To Party

Material Data

Plant

Material to

Default Data

Pricing Date Price Simulation

Billing Type

Order Type

Item Category

Figure 3.17 Net Price List: Selection Screen (Transaction `V_NL`)

You also need the `PLANT` field to identify the country of departure (impact on the tax determination). The `PRICE SIMULATION` attribute determines whether conditions with release status `B – RELEASED FOR PRICE SIMULATION` or `C - RELEASED FOR PLANNING AND PRICE SIMULATION` are to be taken into account.

As a result, an SAP List Viewer (ALV) is displayed with a selected set of fields that can be edited with the usual procedure in the ALVs (menu path `SETTINGS • LAYOUT`) (see Figure 3.18).

Create net price list

Material	Description	Net price	Curr.	Unit	U...	Tax	Cost	Pricing date
DPC1	Desktop PC1/...	1.060,65	EUR	1	PC	106,07	0,00	18.04.2016
DPC1002	Harddisk 10.8...	183,60	EUR	1	PC	18,36	51,53	18.04.2016
DPC1003	Harddisk 180 ...	365,85	EUR	1	PC	36,59	73,91	18.04.2016
DPC1004	Harddisk 42.9...	750,96	EUR	1	PC	75,10	259,44	18.04.2016
DPC1005	Harddisk 21.1...	193,77	EUR	1	PC	19,38	75,01	18.04.2016
DPC1009	Standard Key...	19,08	EUR	1	PC	1,91	4,72	18.04.2016
DPC1010	Standard Key...	19,98	EUR	1	PC	2,00	2,84	18.04.2016
DPC1011	Professional k...	24,12	EUR	1	PC	2,41	4,66	18.04.2016
DPC1012	Professional k...	31,32	EUR	1	PC	3,13	7,11	18.04.2016
DPC1013	Professional k...	39,42	EUR	1	PC	3,94	8,84	18.04.2016
DPC1014	SIM-Modul MB...	51,57	EUR	1	PC	5,16	9,55	18.04.2016

Figure 3.18 Net Price List: Result Screen

The net price list, which can also contain information such as transfer prices, is primarily an internal information tool for the department. If such a net price list is forwarded to the customer, of course, that information must be removed from the list. Since it is an ALV, a personalization of the columns displayed is easily possible by choosing menu path `SETTINGS • LAYOUT • CURRENT`.

3.4 Performance-Optimized Price List

Starting with SAP ERP 6.0 EHP 7, Support Package 3, a performance-optimized price list is offered with business function `LOG_SD_PRICE_LIST` and enhanced with `LOG_SD_PRICE_LIST_2` (Support Package 4). Unlike the original net price list, the performance-optimized price list can handle multiple customers at a time and has, in general, more options. Its purpose is not so much to provide information to the department (although this is of course still possible) but to create lists that, for example, can be sent automatically and on a regular basis to your customers. The technical realization of this new list is quite different and no longer uses any document simulation. We will comment on this later when discussing the various possibilities of customer enhancements in Section 3.4.2, Section 3.4.5, and Section 3.4.6.

Let us begin with the creation of an optimized price list and then have a closer look on the configuration and enhancement possibilities.

3.4.1 Creation of Price Lists

If the relevant business functions are switched on, the optimized price list can be accessed via the menu path SAP EASY ACCESS • LOGISTICS • SALES AND DISTRIBUTION • MASTER DATA • CONDITIONS • LIST • NET PRICE LIST (Transaction V_NLN), thereby replacing the original list that you can still invoke with Transaction V_NL.

At first glance, the upper part of the selection screen in Figure 3.19 looks very similar to the traditional price list from Figure 3.17.

Figure 3.19 Selection Screen of the Performance Optimized Price List

If you look closer, you will see that you can now select several customers at once and that the selection of the products is more flexible. Net prices are calculated for each possible customer/material combination.

The lower part of the selection screen offers several new options. There are three radio buttons in the OUTPUT section of the selection screen, as follows:

▶ DISPLAY LIST

If you choose this button, the prices will be displayed on your screen in the form of one ALV, including all calculated customer/material combinations. You can personalize the list to your needs, create different layouts, filter, etc. You can enter an existing layout in the LAYOUT AND LANGUAGE section. We will describe later how to control and enhance the available fields in the output list. A possible result could look like the one shown in Figure 3.20.

Pricing date	Customer	Material	Description	Quantity	Sales	Net Value	Doc. ...
14.12.2015	NPL_TEST1	TEST_PL_01	Folder	1	ST	1,49	EUR
14.12.2015	NPL_TEST1	TEST_PL_02	Loose-leaf Folder	1	ST	0,99	EUR
14.12.2015	NPL_TEST1	TEST_PL_03	Print-out Paper 10...	1	PAC	3,99	EUR
14.12.2015	NPL_TEST2	TEST_PL_01	Folder	1	ST	2,09	EUR
14.12.2015	NPL_TEST2	TEST_PL_02	Loose-leaf Folder	1	ST	0,99	EUR
14.12.2015	NPL_TEST2	TEST_PL_03	Print-out Paper 10...	1	PAC	3,69	EUR
14.12.2015	NPL_TEST3	TEST_PL_01	Folder	1	ST	2,99	EUR
14.12.2015	NPL_TEST3	TEST_PL_02	Loose-leaf Folder	1	ST	0,99	EUR
14.12.2015	NPL_TEST3	TEST_PL_03	Print-out Paper 10...	1	PAC	3,49	EUR

Figure 3.20 Example Output of Performance-Optimized Price List

▶ UPLOAD TO FTP SERVER

Alternatively, you can upload the price list as a CSV file to an FTP server. The server address, user, and password are defined in the Customizing of the price list function (IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULT • DEFINE FTP SETTINGS). Price lists for different customers are stored in different files, with the customer number being part of the filename. You may define a prefix for the files in the LAYOUT AND LANGUAGE section of the report. The system will then put the prefix in front of the file-names that are composed as follows: <Prefix>_<Customer Number>_<Time-stamps>.

The files can be stored with or without the column headers (checkbox in the LAYOUT AND LANGUAGE section).

► SEND BY E-MAIL

It is also possible to send the price lists via e-mail. The report allows you to select the e-mail addresses from different sources such as, for example, the communication data of the customer or the contact person in the customer master data. You can also set the e-mail addresses manually for each customer (the SELECT E-MAIL ADDRESSES button) and store them in a report's variant, or send all lists to one or several internal e-mail addresses. SAP provides the `PROCESS_CUSTOMER_EMAIL` method in the BAdI `BADI_PIQ_SDPRICELIST`, where you can program your own logic if necessary.

Price lists can be sent in CSV or Microsoft Excel format. For all output options, the system will take the denoted layout from the LAYOUT AND LANGUAGE section into account and send one list per customer. You can specify a default language for all texts or decide to retrieve the language for each customer separately from the sold-to master data record. You can also create language-dependent SAPscript texts for the e-mail header and body in Transaction SO10 and specify them on the selection screen. The texts can also be filled in using the method `PROVIDE_EMAIL_TEXT_NAME` of the BAdI `BADI_PIQ_SDPRICELIST`.

The advanced settings section offers options to influence the result of the price list, as follows:

► FILTER PRICING PROCEDURE

When creating a price list, you may not need all lines of your pricing procedure. Maybe you do not care about the freight cost or you are not interested in the cost conditions. The new price list Customizing offers the definition of *filter variants* for pricing procedures where you mark non-relevant steps. These steps are then ignored or skipped during the price calculation, thus improving performance. You can maintain one or more filters for a pricing procedure here: IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULT • DEFINE FILTER FOR PRICING PROCEDURE. Just flag the rows in the procedure that you do not need in the price list calculation (see Figure 3.21).

► DISPLAY PRICING ELEMENTS

The previous net price list provides net price and tax information but no information about individual price elements or subtotal lines, for example the overall discount or a special sales deal. In a similar way to filtering some lines from the pricing procedure, you may define *display variants* where you mark those lines in the procedure that will be available for output in the price list result.

Change View "Skipped Conditions During Processing": Overview

Procedure: ZZAA01
Filter Variant: NOFC

Step	Cntr	CTyp Description	From	To	Man.	Mdt	Stat	S Reqt	AltCTy	AltCBV	Skip
107	0	K029 Material pricing grp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	0	0	<input type="checkbox"/>
108	0	K030 Customer/Mat.Pr.Grp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	0	0	<input type="checkbox"/>
109	0	K031 Price Grp/Mat.Pr.Grp	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	0	0	<input type="checkbox"/>
300	0	Discount Amount	101	299	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	0	0	<input type="checkbox"/>
302	0	NETP Price	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	6	3	<input checked="" type="checkbox"/>
310	0	PN00 Price (net)	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	6	0	<input checked="" type="checkbox"/>
800	0	Net Value for Item	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	0	2	<input type="checkbox"/>
801	0	MRAB Free goods	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	59	0	29	<input checked="" type="checkbox"/>
815	0	KF00 Freight	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	4	2	0	<input checked="" type="checkbox"/>
820	0	HM00 Order value	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	0	0	<input checked="" type="checkbox"/>
895	0	PDIF Diff.value (own)	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	0	0	<input checked="" type="checkbox"/>
900	0	Net Value 2	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	3	0	2	<input type="checkbox"/>
914	0	SKTV Cash Discount	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	D 14	0	2	<input checked="" type="checkbox"/>
915	0	MWST Output Tax	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	10	0	16	<input type="checkbox"/>
919	0	DIFF Rounding Off	0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	13	16	4	<input type="checkbox"/>
920	0	Total	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A 0	4	0	<input type="checkbox"/>
930	0	SKT0 Cash Discount	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	9	0	11	<input checked="" type="checkbox"/>
932	0	RL00 Factoring Discount	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	23	0	2	<input checked="" type="checkbox"/>
933	0	MW15 Factoring Disc. Tax	932	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	21	0	0	<input type="checkbox"/>
935	0	GRWR Statistical value	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	C 8	0	2	<input checked="" type="checkbox"/>
940	0	VPRS Internal price	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	B 4	0	0	<input checked="" type="checkbox"/>
950	0	Profit Margin	0	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0	11	0	<input checked="" type="checkbox"/>

Figure 3.21 Definition of a Filter Variant: Rows in the Pricing Procedure Irrelevant for the Net Price List Result Are Marked to be Skipped

The relevant Customizing is located in IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULT • DEFINE PRICING ELEMENTS DISPLAY. We will explain the exact setting needed to make the flagged elements part of the output list in Section 3.4.6.

► DISPLAY SCALES

The performance-optimized price list supports the display of scale information in the result. If you flag the DISPLAY SCALES checkbox, the system will evaluate all scales involved in the calculation of the net price and will deduce the quantities where it expects the effective net price to change. For these quantities, the system recalculates the price and adds corresponding lines to the result (see Figure 3.22). You can thereby inform your customer about the opportunity to get better prices if they purchase higher quantities.

The number of additional quantities can be limited by the SCALE LEVEL LIMIT field. If you leave it empty, the price list will calculate and display all relevant quantities. If you enter "4", for example, you will receive up to four additional lines or five in total.

The net price list supports any mixture of value, quantity, weight, and volume scales and has a from-scale notation in the output; i.e., the listed price is valid from the indicated sales quantity upwards. Graduated scales are not permitted since they result in different prices for each sales quantity. Scale-base formulas may interfere with the calculation of the sales quantities in an unpredictable way and are similarly not allowed. However, BAdI BADI_PRC_NPL_SCALES allows you to overrule these restrictions and to give your own set of sales quantities to be calculated.

Pricing date	Customer	Material	Description	Quantity	Sales ...	Net price	Doc. ...	Pricin...	Unit ...
14.12.2015	NPL_TEST1	TEST_PL_01	Folder	1	ST	1,49	EUR		1 ST
14.12.2015	NPL_TEST1	TEST_PL_02	Loose-leaf Folder	1	ST	0,99	EUR		1 ST
14.12.2015	NPL_TEST1	TEST_PL_03	Print-out Paper 100 pc	1	PAC	3,99	EUR		1 PAC
14.12.2015	NPL_TEST1	TEST_PL_03	Print-out Paper 100 pc	10	PAC	3,49	EUR		1 PAC
14.12.2015	NPL_TEST1	TEST_PL_03	Print-out Paper 100 pc	50	PAC	2,99	EUR		1 PAC

Figure 3.22 Output of the Price List with Scales Displayed

- ▶ **CONSIDER MINIMUM ORDER QUANTITY**
Usually the price list will calculate the net price and value for the quantity 1 in sales units of measure. If you mark this checkbox, the prices will be calculated for the minimum order quantity instead.
- ▶ **HIDE ITEM WHEN PRICE IS ZERO**
Customer/material combinations with a zero net price will be excluded from the result list.
- ▶ **HIDE ITEMS WITH PRICING ERROR**
Customer/material combinations with errors in the price calculation will be excluded from the result list.
- ▶ **DISPLAY MESSAGE LOG**
We will explain in the next section how the necessary data for the pricing call is gathered. Especially in the setup and test phases, the message log can give you valuable hints if the required Customizing settings for the data retrieval are not complete.

3.4.2 Customizing and Customer Enhancements

We already mentioned that the performance-optimized price list does not use any kind of business document simulation, so user exits, like those found in the sales order or billing document, are not called. Instead, the price list coding analyzes the access sequences, user exits, formulas, and requirements used in the relevant pricing procedures to determine the actual list of required fields for the pricing communication structures KOMK and KOMP. Only the relevant fields are then determined, with as few database accesses as possible.

While the system can identify fields needed in access sequences reliably, this is not possible for fields used in formulas or requirements. That is why this information is stored in a dedicated customizing table. To allow an effective and flexible bundling of database selects, the necessary table accesses for data retrieval are also stored in customizing tables in form of metadata.

If you use customer-specific fields in access sequences, user exits, formulas, or requirements, you need to include the same information (which fields are needed and how they are determined) in the customer areas in Customizing.

Note

We will now explain the necessary settings with an example. In doing so, we have to assume some technical knowledge and need to delve into some details and enhancement concepts described later in the book, especially topics covered in Chapters 9 and 11. If you are more of a novice in pricing, you may want to skip the following sections for now and continue with Chapter 4.

You will find all necessary activities under the node IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS.

3.4.3 Registration of Fields

Let's assume that we have added the fields ZZMVGR1, ZZMVGR2, ZZSIZECAT, and ZZPROMO to the item communication structure of pricing KOMP. ZZMVGR1 and ZZMVGR2 correspond to the fields MVGR1 (material group 1) and MVGR2 (material group 2) from the sales data in the material master record. ZZSIZECAT is stored in our own table ZZTSIZECAT with the material number as the key field. ZZPROMO is derived by some dedicated business logic. We assume that we use ZZMVGR1 in one of our condition accesses, while ZZMVGR2,

ZZSIZECAT, and ZZPROMO are used only in our requirement 999 (see Figure 3.23). ZZPROMO is used inside function module Z_CHECK_RELEVANCE.

```

Include          RV61A999          Active

1  FORM koted_999.
2  DATA lv_subrc TYPE sysubrc.
3  sy-subrc = 4.
4  IF komp-kposn NE 0.
5  CHECK: komp-prsfd CA 'BX'.
6  CHECK: komp-zzmvr2 NE '001'.
7  CHECK: komp-zzsizecat NE 'XL'.
8  CALL FUNCTION 'Z_CHECK_RELEVANCE'
9  EXPORTING
10     is_komk = komk
11     is_komp = komp
12  IMPORTING
13     ev_subrc = lv_subrc.
14
15  CHECK lv_subrc IS INITIAL.
16  ENDIF.
17  sy-subrc = 0.
18  ENDFORM.
  
```

Figure 3.23 Requirement 999 Using Fields ZZMVR2, ZZSIZECAT, and ZZPROMO

The performance-optimized price list tries to identify all needed fields for a specific price calculation. Therefore, it scans all access sequences of the pricing procedure involved and thus becomes aware of ZZMGR1. However, fields required in the execution of user exits, customer-specific formulas, or requirements must be explicitly published in Customizing. To do so, choose the IMG activity IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • EDIT FIELDS USED IN PRICING ENHANCEMENTS.

On the entry screen, we restrict the view to the relevant pricing procedure and decide that we want to see only customer-developed enhancements. We navigate to requirement 999. A double-click on include RV61A999 would lead us to the coding, but we double-click the node 999 instead.

We now have the list of fields automatically found by means of a simple code scans: ZZMVR2 and ZZSIZECAT. However, the field KOMP-ZZPROMO is “hidden” behind a function module call and hence was missed. Therefore, we register it manually by clicking the ADD FIELD MANUAL button and entering the necessary data. You can see the result in Figure 3.24, where the found fields have not yet been confirmed and the ZZPROMO field has been added manually.

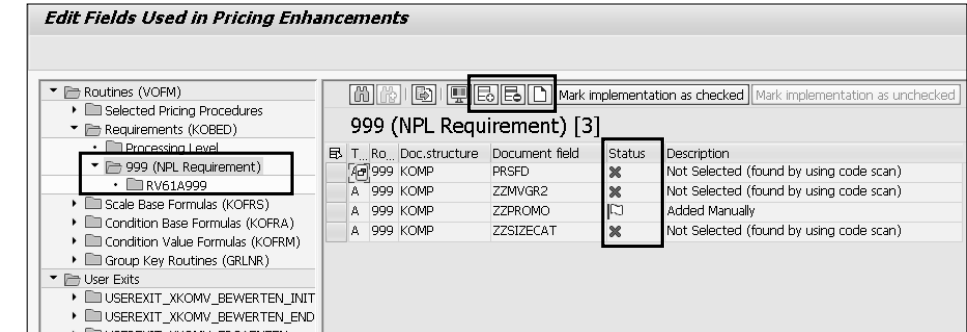


Figure 3.24 Result of the Generic Code Scan for Requirement 999 Searching for Relevant Fields

In the second step, we have to select all fields that must be filled in a price list run (there may be fields that are not relevant) and confirm them explicitly. For this, we select the field(s) and click the SELECT FIELDS button . The status of the fields then changes to SELECTED (Figure 3.25). You can remove fields that are not relevant for the price lists calculation again with the Deselect button .

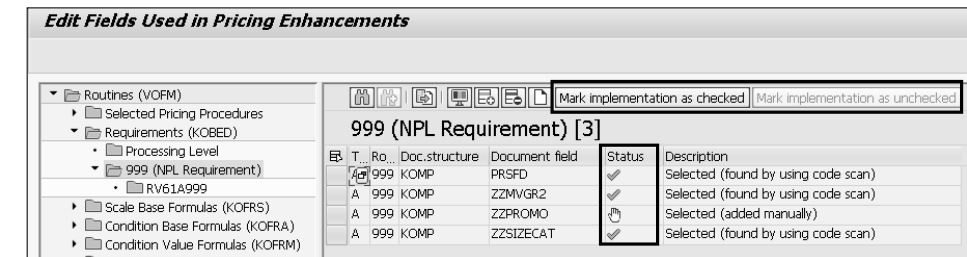


Figure 3.25 Fields Have Been Selected as Relevant

Since the automatic code scan is quite basic, you should always check the system's proposal for your routines and user exits.

After reviewing and completing all entries, you finalize and document your settings by clicking on the MARK IMPLEMENTATION AS CHECKED button. Only now will the net price list take your settings into account.

Note

You can undo your settings any time by clicking the MARK IMPLEMENTATION AS UNCHECKED button. Manually added fields will automatically be removed.

In addition to specifying which fields are required in your pricing exits, you may also specify a *processing level* for your requirements (Figure 3.26). If, for example, a requirement uses only fields from the header communication structure KOMK (so-called header fields), it is usually sufficient to test it only once per pricing header and buffer the result for the following items.

The available processing levels are HEADER, ITEM, CONDITION TYPE (the hat, list, and coin icons, respectively) and STANDARD PROCESSING LEVEL (the green square icon). The STANDARD PROCESSING LEVEL corresponds to an unmaintained processing level. Here nothing is buffered and the requirement is processed each time it's necessary. In simple cases, this is often faster than writing and reading a buffer. That is why SAP has not maintained any level for many standard requirements.

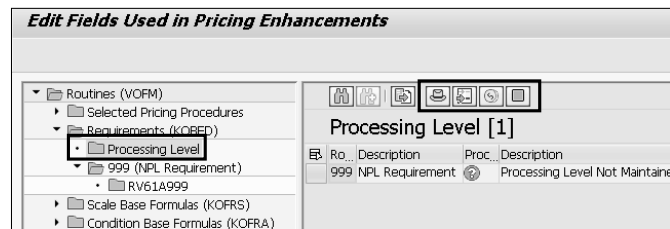


Figure 3.26 Setting the Processing Level of a Requirement

When setting the processing level, you need to be aware that, for performance reasons, the flow logic of pricing is slightly different in the price list processing. The condition accesses are not performed on each item separately in the pricing function module, but for all items together prior to pricing. Consequently, the call of the requirements is brought forward too, and takes place outside of pricing. Hence, if you do more in your requirement than simply check the relevance of a condition, you should assign the standard level or no level at all. By doing so, you ensure that the requirement is called again at the usual place. An example of such a routine is the standard requirement 62, which fills the internal table XKOMPLOOP.

3.4.4 Define the Data Determination

We have already mentioned that the new price list tries to bundle database accesses. This applies in particular when filling the communication structures in pricing. Thus, in order to enable the system to identify collectable accesses in a generic way, the necessary information is stored in the form of metadata. It is

important that you understand how the price list determines its data since you must provide the same information in your own fields. For this purpose, we will have a closer look now at the definition of the table accesses and the corresponding source fields (including customer-specific fields).

Define Table Access

Setting up the metadata consists of two steps. First, you need to specify how a database table will be accessed. This happens in IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • PRICE INQUIRY • DEFINE TABLE ACCESS.

In Figure 3.27 you will see that for the table containing the sales data of the customer master KNVV, three DB or *table aliases* are defined. For table KNVV, three different aliases are specified: KNVV, KNVV_RG, and KNVV_WE. Each of them accesses the table with different fields.

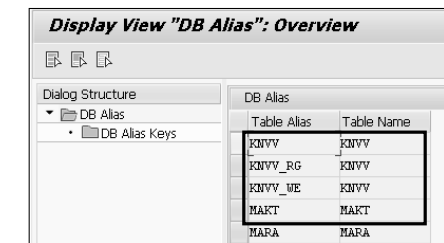


Figure 3.27 Definition of Table Alias

If we navigate to the DB ALIAS KEY node, we realize that the three aliases differ in their access fields. The alias KNVV accesses the table with the contents of the source fields KUNNR, SPAKU, VKORG, and VTW KU, reading data from the sold-to party. The alias KNCC_RG, on the other hand, reads data from the payer, accessing the same table as the KUNRG, SPAKU, VKORG, and VTW KU fields (Figure 3.28). The DB alias thus defines the table and the *Where* clause of the required *Select* statement.

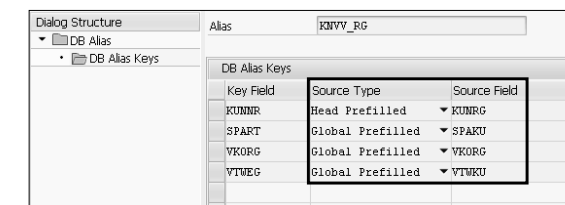


Figure 3.28 The DB Alias KNVV_RG Accesses the Sales Data of the Payer

What does the SOURCE TYPE in Figure 3.28 represent? It defines where the source field comes from, or technically speaking, in what internal structure the field resides. We distinguish the following fields:

- ▶ GLOBAL PREFILLED data is relevant for the whole price list, such as organizational data or the price date
- ▶ HEAD PREFILLED data is usually derived from the customer and relevant for all products
- ▶ ITEM PREFILLED data is usually derived from the material
- ▶ HEADITEM PREFILLED data depends on customer and material combination
- ▶ DIRECT VALUE is a constant that can be set in Customizing

Simply put, prefilled data is data whose determination is hard-coded in the price list. It is also possible to preempt your own data via a BAdI implementation. In this case, you have to choose the additional data source types (GLOBAL, HEAD, ITEM, or HEADITEM).

Finally, there is *caller data*. Roughly speaking, this is additional data coming from the outside, especially if the price determination is called outside the net price list via the dedicated function `PIQ_CALCULATE`. We will come back to this very technical detail in Section 3.4.8. For the moment, it is sufficient to know that the `F4` help on the source field will provide you with all possible fields for the selected source type.

To summarize, the DB alias `KNVV_RG` can be translated into as follows:

```
select <?> from KNVV into <?> where KUNNR = (Head Prefilled)-KUNRG.
```

Maintain Sources for Communication Structures

In IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • PRICE INQUIRY • SOURCES FOR COMMUNICATION STRUCTURES we define how the fields relevant for pricing are determined in detail. In Figure 3.29 you can see the fields of the header communication structure KOMK. The distribution channel VTWEG is transferred from prefilled data. Other fields like the payment terms ZTERM are determined using a DB alias. Let's have a closer look at the ZTERM field. The line you see in Figure 3.29, together with the definition of the DB alias `KNVV_RG`, means that the field ZTERM is filled from the field ZTERM from table `KNVV` using

the payer `KUNRG` to access the table, i.e., the payment terms are read from the sales area data of the payer.

Target Structure	Target Field	Source Type	Source Alias	Source Field
KOMK	VTWEG	Global Prefilled		VTWKO
KOMK	WAERK	Database Table Alias	KNVV	WAERS
KOMK	WERKS	HeadItem Prefilled		WERKS
KOMK	WKCOU	HeadItem Prefilled		COUNC
KOMK	WKCTY	HeadItem Prefilled		CITYC
KOMK	ZTERM	Database Table Alias	KNVV_RG	ZTERM
KOMP	ATTYP	Database Table Alias	MARA	ATTYP

Figure 3.29 Sources of the Fields from the Pricing Communication Structures

Together with the DB alias definition, the following `select` will be performed:

```
select ZTERM from KNVV into KOMK-ZTERM where KUNNR = (Head Prefilled)-KUNRG.
```

After evaluating all metadata, the price list can optimize the access and read all required fields from a database table together.

Available Fields with Standard Sources

The price list report does not yet support all fields from the communication structure. You can find out which fields are available by displaying the standard sources in the IMG. If the required field is not listed there, you need to add a corresponding entry in the customer section. Customer entries will always overrule standard settings.

Add Customer Fields

It is time to come back to our example. With what you have just learned, you can now enter the necessary data to determine the fields.

ZZMVGR1 and ZZMVGR2 correspond to fields MVGR1 and MVGR2 from table `MVKE`. We quickly find a suitable DB alias: Alias `MVKE` reads from table `MVKE` with material number and organizational data. Therefore, all you need to tell the system is that you want to select material groups 1 and 2 and have them transferred into the fields ZZMVGR1 and ZZMVGR2 (see Figure 3.30).

Target Structure	Target Field	Source Type	Source Alias	Source Field
KOMP	ZZMVGR1	Database Table Alias	MVKE	MVGR1
KOMP	ZZMVGR2	Database Table Alias	MVKE	MVGR2
KOMP	ZZSIZECAT	Database Table Alias	ZZMYALIAS	ZZSIZECAT
KOMP	ZZPROMO	Postprocessing BAdI		

Figure 3.30 Define the Sources of Customer Fields

Table Alias

All data derived by the same table alias is selected together. You should therefore always check if you can reuse an existing DB alias to avoid additional, unnecessary database accesses.

We store the field ZZSIZECAT field in our own table so we need to create a new DB alias first. We define that we want to access table ZZTSIZECAT (Figure 3.31) with the prefilled material number (Figure 3.32).

Table Alias	Table Name
ZZMYALIAS	ZZTSIZECAT

Figure 3.31 Creating an DB Alias for Accessing Table ZZTSIZECAT

Next, we need to request the selection of field ZZSIZECAT. Go back to the activity IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • PRICE INQUIRY • DEFINE SOURCES FOR PRICING COMMUNICATION STRUCTURE and enter the third line shown in Figure 3.30.

We have assumed that our last field ZZPROMO cannot be determined using a simple select. For such cases, SAP provides the BAdI BADI_PIQ_PREPARE, where you can program your own logic. We nevertheless maintain an entry in Customizing for ZZPROMO with the source type POSTPROCESSING BAdI (see Figure 3.30). This will prevent the system from issuing warnings indicating that the source of the field is unknown.

Key Field	Source Type	Source Field	Dir
MATNR	Item Prefilled	MATNR	

Figure 3.32 Defining How to Access Table ZZTSIZECAT

You find BAdI BADI_PIQ_PREPARE in IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • BUSINESS ADD-INS (BAdIs) • BAdI: PRICE INQUIRY PREPARATION. As mentioned, the underlying price inquiry function can run in different scenarios. Each scenario has a specific caller ID (see Figure 3.33) that defines how to access table ZZTSIZECAT.

Filter Combinations	Value 1	Com...	Filter	Com...
Combination 1	PL	=	CALLER_ID	

Figure 3.33 BAdI Implementations Need Filter Value PL to be Relevant for Net Price List Calculations

The method ADAPT_KOMK_KOMP_DATA substantially corresponds to the known user exits in the various documents and allows you to change the content of the pricing communication structures. You will find some tips how to reuse your user exit coding in SAP Note 2019242. Looking at the signature of the method, you can see that apart from the header and item table, there is an additional table called CT_KOMP_HEADFIELDS. It contains all fields that simultaneously can depend on the customer and the material. The plant is a prominent example. The plant is therefore empty in CT_KOMK and filled in CT_KOMP_HEADFIELDS instead for each customer/material combination. So, if you have added fields to KOMK or KOMP whose content also depends on header data, you need to add these fields to KOMP_HEADFIELDS.

The BAdI interface has a second method called ADAPT_HEAD_ITEM_DATA. You can add fields to the already mentioned source ADDITIONAL DATA here. It is useful if you have fields you want to use in DB aliases because ADAPT_HEAD_ITEM_DATA is processed before the database selects.

Similar to user exits and formulas, you can request the determination of additional fields from KOMK and KOMP for your implementation of method ADAPT_KOMK_KOMP_DATA. To do so, go to IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • BUSINESS ADD-INS (BADIs) • DEFINE FIELDS USED IN BADI: PRICE INQUIRY PREPARATION and enter the required fields (see Figure 3.34).

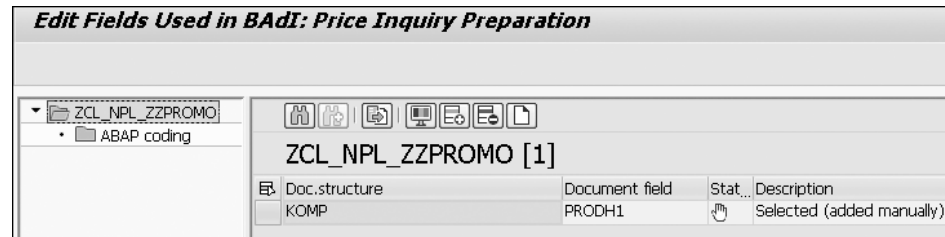


Figure 3.34 Publishing Fields Needed in the Implementation of Method ADAPT_KOMK_KOMP_DATA

The activity works very similarly to EDIT FIELDS USED IN PRICING ENHANCEMENTS; however, there is no code scan. The system will fill the fields requested here, even if the price determination does not need them.

As you can see, the setup of the new price list function requires a certain amount of effort. However, you are usually rewarded with significantly shorter run times.

Check Reports

Currently, two reports exist that allow some control of the settings:

- ▶ SD_PRC_MULTI_ITEM_CHECK
Examines the new bundled access to condition tables, but not the data determination for the KOMK and KOMP fields and is therefore of limited use to the end user. It may be helpful in isolating the cause if the result of the price list is not as expected. It has a very detailed protocol that makes it easy to find inconsistent conditions.
- ▶ SDPIQAPICOMPARE
Compares the price list result with that from an order simulation or the traditional net price list run. The protocol is not as detailed as in SD_PRC_MULTI_ITEM_CHECK.

3.4.5 Enhancements of the Selection Screen

You can extend the selection criteria and the result fields of the price list. Let's assume that we want to use the customer price group KONDA as additional selection criteria. To enhance the selection screen, we will have to enhance the price

list report SDPIQPRICELIST using enhancement spot ES_PIQ_SDPRICELIST. In enhancement point ES_PIQ_SDPRICELIST_EXT1, we add the field to the screen. We use enhancement point ES_PIQ_SDPRICELIST_EXT7 to add a field label and enhancement point ES_PIQ_SDPRICELIST_EXT6 to transfer our selection option to the program using ABAP macro transfer_select_option. Listing 3.1 shows the necessary coding.

```
REPORT sdpiqpricelist.
...
ENHANCEMENT-POINT ES_PIQ_SDPRICELIST_EXT1 SPOTS
                        ES_PIQ_SDPRICELIST STATIC .
***$-Start: ES_PIQ_SDPRICELIST_EXT1-----
ENHANCEMENT 4 ZZ_NPLTEST.  "active version
  Select-OPTIONS: konda for knvv-konda.
ENDENHANCEMENT.
***$-End:  ES_PIQ_SDPRICELIST_EXT1-----
...
Customer extension: change of selection field texts
ENHANCEMENT-POINT ES_PIQ_SDPRICELIST_EXT7
                        SPOTS ES_PIQ_SDPRICELIST .
***$-Start: ES_PIQ_SDPRICELIST_EXT7-----
ENHANCEMENT 3 ZZ_NPLTEST.  "active version
  loop at screen.
    if screen-name = '%_KONDA_%_APP_%-TEXT'.
      %_KONDA_%_APP_%-TEXT = 'Customer Price Group'.
      exit.
    endif.
  endloop.
ENDENHANCEMENT.
***$-End:  ES_PIQ_SDPRICELIST_EXT7-----
...
transfer the search criteria of customer extension fields to price list
API
ENHANCEMENT-POINT ES_PIQ_SDPRICELIST_EXT6 SPOTS ES_PIQ_SDPRICELIST .
***$-Start: ES_PIQ_SDPRICELIST_EXT6-----
ENHANCEMENT 2 ZZ_NPLTEST.  "active version
  transfer-select_option 'KNVV' 'KONDA' konda.
ENDENHANCEMENT.
***$-End:  ES_PIQ_SDPRICELIST_EXT6-----
```

Listing 3.1 Adding the Customer Price Group KONDA as Selection Criteria

Finally we make an entry in IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULTS • DEFINE FIELDS because we need to tell the system from which master data table the KONDA value can be derived and how it is connected to the customer or material data (see Figure 3.35).

New Entries: Overview of Added Entries							
Define Customer-Specific Fields							
Field Name	Select From	Source Table	Source Field	Table Alias	Output	Reference Field	Seq...
KONDA	Customers	KNVV	KONDA	KNVV	<input type="checkbox"/>		
	Not Relevant				<input type="checkbox"/>		

Figure 3.35 Inform the Price List That Field KONDA Can Be Selected from Table KNVV with Alias KNVV

Similar to the data determination for the pricing communication structure, the information for the generic selection of customers and materials is stored in the form of metadata. The table entry in Figure 3.35 tells the system that, while selecting the customer, it can find the relevant customer group information in field KONDA of table KNVV. The table alias KNVV is the DB alias you saw in Figure 3.27.

The KONDA field is now available as selection criteria on the selection screen (see Figure 3.37) of the price list.

Additional Fields for Selection and Output

The price list report uses tables KNVV and MVKE to retrieve all necessary data for selection and output. You can only add fields from related tables such as KNA1 or MARA. The program creates a generic `select` statement joining all necessary tables during runtime. BAdIs exist for more complex cases.

In a very similar way, you can extend the selection screen by your own input parameter. Let's assume you want to provide the price list currency. In this case, you add the code in Listing 3.2.

```
REPORT sdpiqpricelist.
...
* Customer extension: further criteria in control block
ENHANCEMENT-POINT ES_PIQ_SDPRICELIST_EXT3 SPOTS
                    ES_PIQ_SDPRICELIST STATIC .
*$$$-Start: ES_PIQ_SDPRICELIST_EXT3-----
ENHANCEMENT 8 ZZ_NPLTEST. "active version
PARAMETERS Waerk like komk-waerk OBLIGATORY.
ENDENHANCEMENT.
*$$$-End: ES_PIQ_SDPRICELIST_EXT3-----
SELECTION-SCREEN END OF BLOCK control.
...
Customer extension: change of selection field texts
```

```
ENHANCEMENT-POINT ES_PIQ_SDPRICELIST_EXT7 SPOTS ES_PIQ_SDPRICELIST .
*$$$-Start: ES_PIQ_SDPRICELIST_EXT7-----
ENHANCEMENT 3 ZZ_NPLTEST. "active version
loop at screen.
  if screen-name = '%_WAERK_%_APP_%-TEXT'.
    %_WAERK_%_APP_%-TEXT = 'Price List Currency'.
    exit.
  endif.
endloop.
ENDENHANCEMENT.
*$$$-End: ES_PIQ_SDPRICELIST_EXT7-----
...
transfer the search criteria of customer extension fields to price list
API
ENHANCEMENT-POINT ES_PIQ_SDPRICELIST_EXT6 SPOTS ES_PIQ_SDPRICELIST .
*$$$-Start: ES_PIQ_SDPRICELIST_EXT6-----
ENHANCEMENT 2 ZZ_NPLTEST. "active version
add_name_value 'WAERK' WAERK 'KOMK'.
ENDENHANCEMENT.
*$$$-End: ES_PIQ_SDPRICELIST_EXT6-----
```

Listing 3.2 Adding the Document Currency as Control Parameter

To inform the price list that, from now on, it has to read the currency from the caller data and not from the customer master data, you add the entry shown in Figure 3.36 in IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • PRICE INQUIRY • SOURCES FOR COMMUNICATION STRUCTURES.

You might have noticed the addition of `OBLIGATORY` in Listing 3.2. This is necessary because with the entry from Figure 3.36, the currency is no longer read from the customer master. Therefore, you must require that the user supply a currency on the selection screen. Alternatively, we could fill the currency in method `ADAPT_KOMK_KOMP_DATA` of BAdI `BADI_PIQ_PREPARE` in case it has not been set by the user.

Change View "Define Customer-Specific Sources": Overview					
New Entries					
Define Customer-Specific Sources					
Target Structure	Target Field	Source Type	Source Alias	Source Field	Dir
KOMK	WAERK	Global Caller Data		WAERK	

Figure 3.36 Retrieve the Currency from the Caller Data

You can see the result of these enhancements in Figure 3.37.

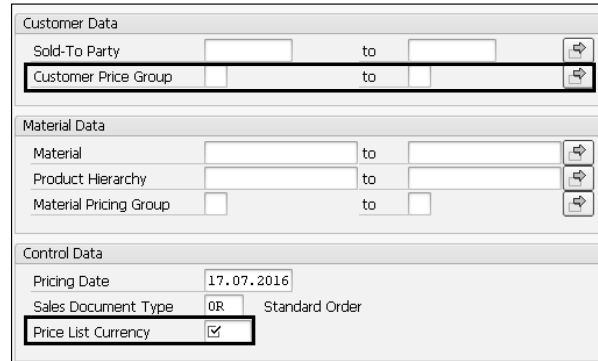


Figure 3.37 Selection Screen Now Enhanced with Customer Price Group and Currency

In addition to the enhancement spot `ES_PIQ_SDPRICELIST`, SAP provides the BAdI `BADI_PIQ_SDPRICELIST` to influence the selection of the price list. Here, the methods `GET_CUSTOMER_DATA` and `GET_MATERIAL_DATA` are particularly worth mentioning. They allow you to add or remove customers and materials from the price list processing. Method `PREPARE_PRICING` permits final changes to global, control, head, and item data before the underlying price inquiry API is called.

3.4.6 Enhancements of the Price List Result

You can also adjust the output to your needs. Suppose that in addition to the customer number, we also want to display the customer's name. We know that the customer's name is stored in the field `NAME1` of the customer master table `KNA1`. Selecting the path `IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULTS • DEFINE TABLE MAPPING`, we find that an access to table `KNA1` with the customer number is already available as the DB alias `KNA1`. Since only the tables `KNVV` and `MVKE` (sales area data for customer and material) are used for the selection and the output of the price list, we need to tell the system how `KNA1` relates to `KNVV`. The system can then select the data with one access by joining the two tables. We choose the customer section using `IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULTS • DEFINE TABLE MAPPING` and specify that the customer number `KUNNR` of `KNVV` must be equal to the customer number `KUNNR` of `KNA1` (see Figure 3.38).

Next, we have to define that the field `NAME1` is retrieved from `KNA1`. We go to Customizing activity `IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING •`

`PRICE LISTS • SELECTION AND RESULTS • DEFINE FIELDS` and enter the line shown in Figure 3.39 in the customer section.

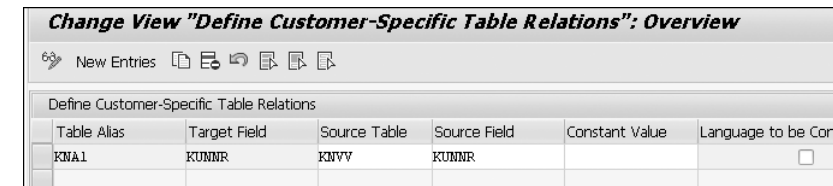


Figure 3.38 Specifying the Relation between Two Tables

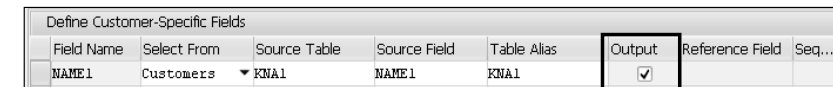


Figure 3.39 Selecting the Customer Name for Display in the Price List Result

Now the price list will read field `NAME1` from table `KNA1` with the key field `KUNNR` being the same as in `KNVV`. We flag the checkbox `OUTPUT` to add the field to the output data.

In addition, we want to display a particular discount in the price list. As discussed earlier in the chapter, we need to create a display variant to make the selected lines of the pricing result available.

We create a variant and select the rows in the pricing procedure in `IMG Customizing activity IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULTS • DEFINE PRICING ELEMENTS DISPLAY` (see Figure 3.40).

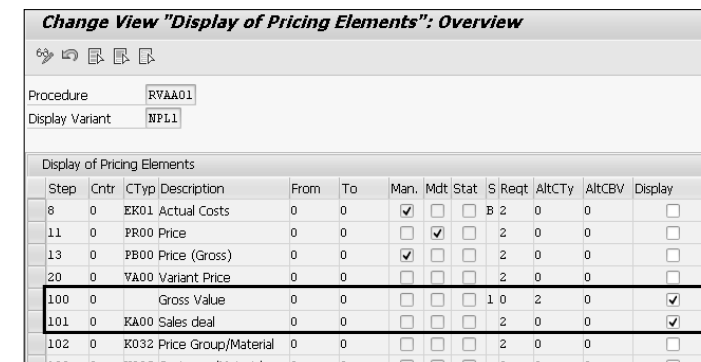
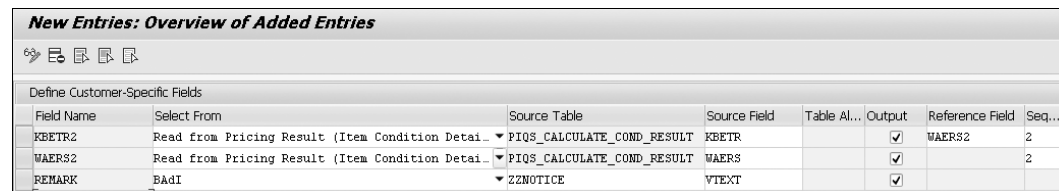


Figure 3.40 Display Variant: Data from the Gross Price Subtotal Line and the Condition Line KA00 Are Provided for the Result List

Similar to the NAME1 fields, we need entries in the customer-specific fields list to make the pricing element details available in the output. First, we have to realize what fields we need. In our case, we want the condition amount KBETR and its currency WAERS from the KAOO condition that is the second price element in our display variant (see Figure 3.40). Therefore, we add the lines shown in Figure 3.41. Price elements are always read from the price result (item condition details) and the corresponding source table is PIQS_CALCULATE_COND_RESULT (the exception to this rule is the description of a price element or subtotal line, which is stored in PIQS_CALCULATE_STEP_DESCR). Since we need the second line of the variant, we add the SEQUENCE NUMBER 2. To ensure the correct formatting of the amount, we also enter a reference field.



Field Name	Select From	Source Table	Source Field	Table Al...	Output	Reference Field	Seq...
KBETR2	Read from Pricing Result (Item Condition Detai...	PIQS_CALCULATE_COND_RESULT	KBETR		<input checked="" type="checkbox"/>	WAERS2	2
WAERS2	Read from Pricing Result (Item Condition Detai...	PIQS_CALCULATE_COND_RESULT	WAERS		<input checked="" type="checkbox"/>		2
REMARK	BAdI	ZZNOTICE	VTEXT		<input checked="" type="checkbox"/>		

Figure 3.41 Adding Fields to the Output

Last but not least, we want to add a text field where we can write a short notice. For more complex field determinations, SAP provides the method PROCESS_RESULTS in BAdI BADI_PIQ_SDPRICELIST where you can fill your output fields dependent on other result data. We add the third line from Figure 3.41, indicating that we will fill the fields in the BAdI. Nevertheless, we have to supply a source table and field to define the technical properties of our field.

We decide to add text if the discount exceeds a certain limit (see Listing 3.3).

```
METHOD if_badi_piq_sdpricelist~process_result.
  FIELD-SYMBOLS: <ft_result> TYPE STANDARD TABLE,
                 <fs_result> TYPE any,
                 <fv_kbetr> TYPE kbetr,
                 <fv_remark> TYPE text30.
  DATA: lv_limit TYPE kbetr VALUE '-25.00'.
  DATA: comp1 TYPE komp_type VALUE 'KBETR2',
         comp2 TYPE komp_type VALUE 'REMARK'.
  ASSIGN cr_result->* TO <ft_result>.

  LOOP AT <ft_result> ASSIGNING <fs_result>.
    ASSIGN COMPONENT comp1 OF STRUCTURE <fs_result> TO
      <fv_kbetr>.
```

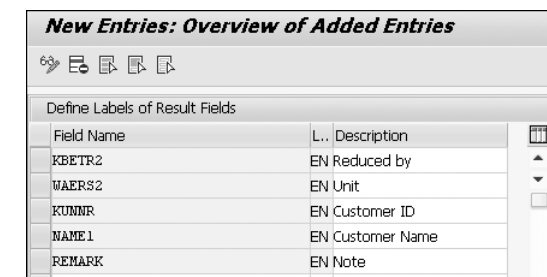
```
CHECK <fv_kbetr> IS ASSIGNED.
IF <fv_kbetr> LE lv_limit.
  ASSIGN COMPONENT comp2 OF STRUCTURE <fs_result>
    TO <fv_remark>.

  CHECK <fv_remark> IS ASSIGNED.
  <fv_remark> = 'greatly reduced'.
ENDIF.
ENDLOOP.

ENDMETHOD.
```

Listing 3.3 Adding Text if the Discount Exceeds 25%

Finally, we want to adjust the labels of our new columns. To do so, we go to IMG • SALES AND DISTRIBUTION • BASIC FUNCTIONS • PRICING • PRICE LISTS • SELECTION AND RESULTS • DEFINE LABELS OF RESULT FIELDS and enter suitable descriptions (see Figure 3.42). We can define the labels in general or depending on the particular pricing procedure. However, be aware that procedure-related labels do not appear in the DISPLAY LIST option. This output option can combine several customers in one list and there may be more than one pricing procedure involved. The column label needs to be unique, however.



Field Name	L...	Description
KBETR2	EN	Reduced by
WAERS2	EN	Unit
KUNNR	EN	Customer ID
NAME1	EN	Customer Name
REMARK	EN	Note

Figure 3.42 Adjusting the Column Header of the Price List Result

After adjusting the layout of the price list report, the final list will look something like the one shown in Figure 3.43 (CUSTOMER NAME, REDUCED BY, and NOTE have been added).

We do not want to finish this section without pointing out another BAdI. The method PROCESS_OUTPUT of BAdI BADI_PIQ_SDPRICELIST_OUTPUT allows you to program your own output options. You can, for example, decide to store the price list result in a database table for analytical purposes.

Pricing date	Customer ID	Customer Name	Material	Description	Quantity	Sales	Net price	Doc. ...	Reduced by	Unit	Note
14.12.2015	NPL_TEST1	Max Mustermann	TEST_PL_01	Folder	1	ST	1,40	USD			
14.12.2015	NPL_TEST1	Max Mustermann	TEST_PL_02	Loose-leaf Folder	1	ST	0,93	USD			
14.12.2015	NPL_TEST1	Max Mustermann	TEST_PL_03	Print-out Paper 10...	1	PAC	3,75	USD			
14.12.2015	NPL_TEST2	Lieschen Müller	TEST_PL_01	Folder	1	ST	1,97	USD	30,00-	%	greatly reduced
14.12.2015	NPL_TEST2	Lieschen Müller	TEST_PL_02	Loose-leaf Folder	1	ST	0,93	USD			
14.12.2015	NPL_TEST2	Lieschen Müller	TEST_PL_03	Print-out Paper 10...	1	PAC	3,47	USD			
14.12.2015	NPL_TEST3	Hans Beispiel	TEST_PL_01	Folder	1	ST	2,81	USD			
14.12.2015	NPL_TEST3	Hans Beispiel	TEST_PL_02	Loose-leaf Folder	1	ST	0,93	USD			
14.12.2015	NPL_TEST3	Hans Beispiel	TEST_PL_03	Print-out Paper 10...	1	PAC	3,28	USD			

Figure 3.43 The Final ALV Output of Price List Example

SAP Note 1949280 provides some detailed descriptions if you want to engage more deeply in selection and output enhancements.

3.4.7 Customizing Overview

This section serves as a recap and brief overview of the Customizing and Enhancement options of the performance-optimized price list. Figure 3.44 shows the relevant part of the IMG.

Price Lists
• Edit Fields Used in Pricing Enhancements
Selection and Result
• Define Table Mapping
• Define Filter for Pricing Procedure
• Define Pricing Elements Display
• Define Fields
• Define Labels of Result Fields
• Define FTP Settings
Price Inquiry
• Define Table Access
• Define Sources for Pricing Communication Structures
• Check Consistency of Customizing
Business Add-Ins (BAdIs)
Price Inquiry Preparation
• BAdI: Price Inquiry Preparation
• Define Fields Used in BAdI: Price Inquiry Preparation
Price List
• Notes on Implementation
• BAdI: Price List Processing
• BAdI: Price List Output
• BAdI: Control of Calculation Regarding Scale Information

Figure 3.44 IMG Activities for the Performance-Optimized Price List

It helps to understand the structuring, if you know that the price list technically has two layers. The outer one, the *price list layer*, processes the data from the

selection screen and builds up the caller data we have seen before. It calls the inner layer, the *price inquiry layer*, and processes the result of the price calculation, for example, by sending the lists to the customers via e-mail. The inner layer is responsible for taking the caller data, deriving from it all necessary information for pricing, performing the call to pricing, and passing the result back to the caller. The price inquiry layer can be called stand-alone and distinguishes its callers by their caller IDs. The price list has the PL caller ID.

The most important activities in the Customizing are as follows:

- ▶ **EDIT FIELDS USED IN PRICING ENHANCEMENTS**
In this activity, you register fields you need in formulas, requirements, and user exits. With this information, the price inquiry will identify the relevant fields for the price calculation.
- ▶ **DEFINE TABLE MAPPING**
Either display the standard table aliases and relations, or register your own tables for database selects. You need to define how they relate to `KNVV` or `MVKE`. `KNVV` and `MVKE` are the core tables for the selection of customers and products for the price list processing. The defined relation allows a `Join` statement to select all necessary data.
- ▶ **DEFINE FILTER FOR PRICING PROCEDURE**
In a filter variant, you select lines in a pricing procedure to be skipped during the pricing call. This reduces the runtime of the price list.
- ▶ **DEFINE PRICING ELEMENTS DISPLAY**
Lines selected in a display variant are available for output in the price list processing.
- ▶ **DEFINE FIELDS**
Display standard fields or register additional fields for selection or output.
- ▶ **DEFINE LABELS OF RESULT FIELDS**
Adjust the column labels in the output list.
- ▶ **DEFINE FTP SETTINGS**
Register the server address, user, and password for storing the price list result on an FTP server.
- ▶ **DEFINE TABLE ACCESS**
Display the DB aliases of standard table accesses or define new ones. DB aliases define how to access a database table.

- ▶ **DEFINE SOURCES FOR PRICING COMMUNICATION STRUCTURE**
Display the sources of standard fields or define how your own fields can be determined.
- ▶ **CHECK CONSISTENCY OF CUSTOMIZING**
Check the consistency of the Customizing.
- ▶ **BADI: PRICE INQUIRY PREPARATIONS**
BAdI to fill fields in the pricing communication structures.
- ▶ **DEFINE FIELDS USED IN BADI**
Price Inquiry Preparation: Register fields you need in your BAdI implementation. The system will then provide the necessary information if possible.
- ▶ **BADI: PRICE LIST PROCESSING**
Allows influencing parameters for the price inquiry call. Allows influencing the result.
- ▶ **BADI: PRICE LIST OUTPUT**
You can program your own output options here.
- ▶ **BADI: CONTROL OF CALCULATION REGARDING SCALE INFORMATION**
Influence the standard algorithm and how the relevant quantities are calculated.

3.4.8 Important Programs and Function Modules

Following are the most important programs and function modules of the performance-optimized price list:

- ▶ **SDPIQPRICELIST**
This is the standard report for the creation of price lists. It has a dedicated selection screen allowing you to select customers and products and to choose from various options. You can enhance the report's screen by implementing enhancement spot `ES_PIQ_SDPRICELIST`.
- ▶ **API_PIQ_SDPRICELIST**
You can use this function module if you want to access the price list functionality from within another program. The interface corresponds more or less to the selection screen of the price list. SAP Note 1949280 provides a detailed description.

- ▶ **BAPI_PIQ_SDPRICELIST**
With `BAPI_PIQ_SDPRICELIST`, you can call the price list functionality remotely. It is very similar to the function module `API_PIQ_SDPRICELIST` but allows you to disable the consistency check on the input table for performance reasons. In addition, SAP note 1949280 gives a more detailed description of `BAPI_PIQ_SDPRICELIST`.
- ▶ **PIQ_CALCULATE**
`PIQ_CALCULATE` is an RFC-enabled function module and can be useful if you need to calculate prices outside a document simulation with good performance. It requires some control data and a list of customers and materials. It reads metadata from Customizing to optimize the database accesses. Different callers can register different caller IDs in table `PIQC_API_CALLER` or use the standard ID `STD`. When using your own caller ID, you need to define the sources for the pricing communication structure for your fields and your new ID in table `PIQC_PREP_FLD_C`. Unlike the price list APIs, the price inquiry can trigger a group processing for items belonging together.

If you're using `PIQ_CALCULATE`, keep in mind that it does not create any document simulation. Hence, not all functionality related to pricing is available. Examples are free goods, product substitution, down payment functionality, EAN numbers, batch determination, etc.

3.5 Summary

With the ability to configure pricing reports, you can ensure, for example, as department supervisor, that pricing reports are provided for your departments that are adapted to their specific information needs. Pricing reports can be created without programming by a kind of list generator in Customizing for pricing.

In addition to the traditional pricing reports, which have also been introduced as a filter medium in the condition maintenance via area menus, we have—starting with SAP ERP 6.0 (but subject to the activation of the business function `SD_01`)—the worklist for price data from the Internal Sales Representative role as a new development.

With the net price list or its successor, the performance-optimized price list, you can give your customers individual price information for a large number of products.

This chapter marks the end of the first part of this book about the condition technique. We now move on to the second and central part of this book, standard pricing configuration.

Index

A

ABAP editor, 67
Access
 additive, 56
 exclusive, 56
Access optimization, 349
Access sequence, 34, 45, 129, 148, 518
 creation, 135
 customer namespace, 136
 definition, 50
 H102, 616
 MWM1, 57
 PR01, 305, 333, 519
 PR02, 50
 Z901, 55
 Z902, 338
 Z905, 339
 ZASH, 60
 ZPPB, 331
Account determination for cash settlement, 38
Account determination for payment cards, 38
Account key, 165, 563
Accounting settlement period, 574
Accruals, 158, 165
 amount for accruals, 539
 indicator, 562
 manual, 567
 reverse, 540, 568
Accruals correction procedure, 561
Actual costs, 276
Actual-cost accounting, 280
Agreement, 80
 price book, 83
 rebate agreement, 83
 sales deal, 80
Agreement currency (rebate), 578
APPEND structure, 329
Application
 TX (tax), 267
 V (sales), 494, 500
Archiving run, 155, 222
Area menu, 215

Area menu COND_AV, 75, 76, 91
Arrangement calendar, 567
Arrangement calendar (rebate), 578
Automatic code scan, 107

B

BAdI, 206
 BADI_PIQ_PREPARE, 112, 117
 BADI_PIQ_SDPRICELIST, 102, 120
 BADI_PIQ_SDPRICELIST_OUTPUT, 121
 SD_COND_LOAD_MVA, 353
Base condition type, 510
Base formula, 400
Base price, 301
Base value formula, 398
 002 (net value), 457
 028, 638
 029, 629
 110, 313
 202, 305
 28, 275
Billing calendar, 525
Billing date, 525
Billing document, 265, 277, 501, 523
 automatic new pricing, 501
 pricing, 500
Billing plan, 242, 245, 247, 307, 382
 milestone, 246
 number fields, 382
 periodic, 242
Billing request, 252
Breakpoint, 524, 526
BRFplus, 180
 creating an application, 185
Budget condition, 160
Budgeting, 264, 432
Business function
 LOG_MM_COMMODITY_02, 172
 LOG_SD_COMMODITY_02, 230
 LOG_SD_PRICE_LIST, 99
 LOG_SD_REBATE_INDEX, 557, 580

Business function (Cont.)
LOG_SD_REBATE_INDEX_SBS, 557, 580
LOG_SD_SIMP_02, 224, 236
SD_01, 92, 125
 Business transaction event, 206
 Business-to-business (B2B), 295, 298, 501
 Business-to-customer (B2C), 295

C

Calculation schema, 502
 Calculation type, 139, 151
 B - fixed amount, 152, 256
 H, 307
 percentage, 256
 Caller data, 110
 Central table control, 77
 CEPOK (expected price), 413, 436, 646
 Change document, 217
 Closing invoice, 251
 Cluster table, 379
 Collective billing, 523
 Collective billing document, 524
 Commodity Pricing Engine (CPE), 232
 Communication structure, 36, 47
 Composite function, 166
 Composite tax, 269
 Condition amount, 70
 Condition application, 35, 41
 Condition base formula, 393, 398
 Condition base value formula, 276
 Condition basis value, 266
 Condition category, 151, 302
 Condition class, 150
 C, 302, 561
 H, 304, 305, 630
 Condition currency, 285
 Condition exclusion, 72, 167, 258
 procedure, 260
 using formulas, 261
 Condition for invoice list, 159
 Condition formula for basis, 165
 Condition index, 79, 156
 Condition list, 85
 Condition maintenance, 69, 217, 265
 deletion of a condition record, 220
 via area menu, 75
 via area menus, 215
 via index, 79
 Condition master data, 45, 58
 definition, 59
 Condition master record, 218
 Condition prestep, 359
 Condition record, 34, 201, 229
 copy, 210
 definition, 59
 delete, 155, 222
 exclusion indicator, 259
 fast entry screen, 70
 for data determination, 335
 maintenance, 62
 maintenance screen, 237
 release status, 202
 transfer via formula, 344
 Condition supplement, 220, 502
 pricing procedure, 154
 Condition table, 34, 45, 47, 88, 129, 146, 202, 507, 521
 creation, 130
 data field in the, 134
 definition, 48
 field sequence, 328
 Condition technique
 application areas, 33
 data collection, 40
 elements, 33
 special features, 323
 Condition type, 45, 58, 129, 149, 295
 902 (*data determination material*), 451
 ACRL, 313
 ACRN, 313
 ACRR, 313
 AMIW, 170, 317, 605
 AMIZ, 318, 606
 AZWR, 318
 BASB, 508
 create, 141
 customer namespace, 141
 Customizing, 138
 definition, 58

Condition type (Cont.)
 DIFF, 302
 down payment/settlement, 607
 DUPL, 610
 EDI1, 610
 EDI1/EDI2, 314
 EK01, 309, 470, 612
 EK02, 280
 EK02/EK03, 612
 GRWR, 318, 613
 HA00, 369, 614
 HB00, 615
 HD00, 616
 HI02, 616
 HM00, 316, 532, 617
 ITD2, 389
 IV01, 618
 IV01/IV02, 310
 K004, 213
 K005, 264
 KP00, 619
 KP01, 620
 KP02, 621
 KP03, 369, 622
 KUMU, 623
 KW00, 310, 624
 LCFR (license - France), 401, 433
 MW01, 272
 MW15, 315, 626
 MWAA, 269, 508
 MWAS, 269, 508
 MWSI, 627
 MWST, 267
 NETP, 301, 627
 NETW, 307, 628
 NRAB, 275, 628
 PB00, 502
 PB1, 305
 PB1D, 305
 PBBS, 304, 331, 345, 629
 PBU, 344
 PBUD, 304, 344, 631, 632
 PBUP, 345, 472
 PC00, 633
 PI01, 634
 PI01/PI02, 310
 PMIN, 635

Condition type (Cont.)
 PN00, 301, 635
 PPAR, 309
 PPSV, 239, 308
 PR00, 50, 227, 470
 PR01, 636
 PR02, 636, 649
 PTVO, 302, 638
 R100, 638
 RA01, 262
 recalculation, 497
 RL00, 57, 315
 SKTO, 315, 522, 640
 SKTV, 315, 640
 statistical flag, 315
 UTXJ, 273
 VA00, 641
 VPRS, 288, 643
 WK00, 311
 XR1, 273
 Z030/Z031, 405
 Z902, 338, 344
 Z905, 338
 ZB00, 475
 ZBUP, 472, 645
 ZDB1, 646
 ZH01, 435, 650
 ZH02, 435, 652
 ZPBP, 647
 ZPRA, 470, 471, 647
 ZPRB, 470, 472, 648
 ZPRC, 473
 ZPRD, 473, 650
 Condition type group, 538, 563
 Condition update, 156, 264
 Condition value formula, 165, 348
 011, 303, 476
 013, 318
 015, 302
 017 (*Rounding as per T001R*), 393
 018 (*profit margin %*), 476
 025, 307, 628
 036, 623
 112, 314, 389
 113, 313
 114, 313
 329 (*indirect sales vol.*), 592
 905, 477, 646

Condition value formula (Cont.)
 906 (*initial price PR02*), 478
 913, 316
 936, 623
 953 (*rebate exclusion*), 600
 Condition control, 227
 Configurable user interface, 230
 Consumer goods industry, 515
 Contract with resource-related billing, 296
 Control information, 226
 Control run (rebate), 576
 Copy operation, 379
 Copying control, 241
 Copying rule, 210
 Cost, 231, 276
 treatment, 277
 Costing-based CO-PA, 198
 Counter, 162
 CPF, 160, 172, 386, 479
 condition record assignment, 181
 creation, 173
 data source routine, 485
 parameter catalog, 480
 scales, 487
 usage task, 479
 CPF formula, 182, 184, 188, 490
 scales, 189
 CPF routines, 192
 Create with template, 206
 Credit memo request, 285
 Cross-company process, 309
 Cross-company stock sale, 278
 Cross-condition table, 75
 Cross-condition type, 75
 Cumulative condition, 152
 Cumulative graduated scale, 197
 Currency conversion, 157, 281, 286
 Custom structures, 187
 Customer hierarchy, 54
 Customer invoice, 279
 Customer namespace, 44
 for pricing reports, 87
 rounding rules, 210
 Customer-specific condition type, 498, 522
 Customizing table, 105

D

Data determination, 323, 329
 during access, 83
 in pricing, 343
 Data Dictionary (DDIC), 516
 Data model, 59
 free goods, 59
 pricing, 59
 Data retrieval, 194
 Data source, 176
 Data source routine, 192
 Database access, 516
 Date dependency, 49
 Date-dependent relationship, 58
 DB alias, 110
 Del credere commission, 315
 Deletion indicator, 220
 Determination analysis, 64
 Determination procedure, 46, 63
 definition, 64
 Different validity period (rebate), 566
 Direct values, 57
 Discount, 263, 521
 Discount condition type, 301
 Display scale information, 103
 Display variant, 102
 Document condition, 531
 Document currency, 282
 Document item field, 343
 Document pricing procedure, 592
 Drop shipment, 278
 Duplicated condition, 152

E

Electronic Data Interchange (EDI), 37,
 501, 533
 Error handling, 428
 Error message, 428
 301 (*fields missing for access sequence*), 138
 Exchange rate, 281, 283
 currency customizing, 286
 date, 282
 document chain, 285
 for accounting, 283

Exclusion, 158
 rebate, 600
 Exclusion group, 259
 Exclusive free goods, 274
 Expense accounting, 280
 Extended rebate processing, 582
 External tax interface, 272

F

Fast entry screen
 additional data, 72
 condition supplements, 74
 details, 71
 free goods, 74
 scales, 73
 status information, 74
 validity periods, 74
 Field catalog, 46
 and communication structures, 45
 definition, 46
 Field order, 521
 Field sequence, 520
 Filter variant, 102
 Final settlement, 568
 Final settlement (rebate), 553
 Fixed price contract, 246
 Formulas, 521
 Formulas (CPF), 232
 Free goods, 273
 FTP server, 101
 Function module, PRICING_COPY, 378

G

Global data, 497
 Graduated scales, 195
 Graduated-to interval scale, 157
 Group condition, 152, 255, 368, 374, 521
 definition, 256
 header condition with fixed amount, 256
 percentage based header condition, 256
 scale based condition with group key, 257
 scale based condition without group key, 257
 with condition record, 257
 Group key routine, 152, 370

H

Header condition, 232, 256
 percentage based, 256
 screen, 232
 with fixed amount, 256
 Header discount, 234
 Horizon, 243

I

Inactivity indicator, 226, 228
 Inclusive free goods
 with item generation, 274
 without item generation, 274
 Incoterms, 85, 151
 Indirect sales volume (rebate), 582
 Initial indicator, 149
 Internal Sales Representative business
 package, 92
 Internal table
 GKOMV (*group conditions*), 372
 KONDTAB, 364
 STEU (*pricing types*), 359
 TKOMV (*pricing result*), 357
 TKSCHL (*conditions to be redetermined*), 381
 XKNUMA, 305
 XKOMPLOOP, 305
 Invoice list conditions, 315
 Item category, 523
 TAO, 246
 WVN, 242

K

KAEND_TYP (condition categories), 387
 KKURS (conversion rate), 422
 KNPRS (Pricing Types), 359
 KONAU_TYP (condition categories), 388
 KOUPD (condition update), 414
 KZWIW (indicator
 condition subtotal), 415

L

Limit, 264
 condition basis value, 264
 condition value, 264
 number of orders, 264
 Linear tax function, 511
 Local currency, 282

M

Maintain conditions
 transfers, 205
 Maintenance contract, 307
 Maintenance dialog, 60
 Maintenance task, 70
 Make-to-order, 280
 Mass change, 206
 Master data maintenance, 324
 Material group hierarchy, 336, 337, 342
 definition, 336
 nodes, 336
 Milestone billing, 246, 251
 Minimum order quantity, 275
 Missing field assignment, 138
 Module pool, 36
 Multidimensional scale, 513
 Multiple fields, 324

N

Name range for routines, 168
 Net item value, 301
 Net price, 101
 Net price list, 97, 104
 NETWR (net value), 414, 463, 475
 Newton iteration, 511
 NRFAKTOR (factor for free goods
 [inclusive]), 276

O

Oil & Gas industry solution, 493
 Origin of the condition, 228

P

Parameter catalog, 187
 Parent material group, 337
 Partial buffering, 516
 Payment procedure, 568
 Payment procedure (rebate), 547, 574
 Payout currency (rebate), 538
 Percentage based header condition, 256
 Percentage tax, 510
 Performance, 427, 515
 Performance analysis, 529
 Performance-optimized price list, 99, 105
 additional fields, 116
 creation, 100
 customization and enhancements, 105
 Customizing, 122
 data determination, 108
 enhancements, 114, 118
 programs and function modules, 124
 registration of fields, 105
 Planned costs, 276
 Plant determination, 455
 Plus/minus sign, 150
 Points in time, 385
 POWL (personal object worklist), 92
 Predefined price element, 224, 236
 Prefilled data, 110
 Prestep, 43, 391, 517, 523
 Price agreement, 84, 237, 245, 309, 382
 condition types, 238
 creation, 237
 Price book, 83, 303
 Price element, 231
 Price increase, 245
 Price inquiry, 113
 Price inquiry layer, 123
 Price list, 101, 119
 advanced settings, 102
 layer, 122
 Price list report, 111, 121
 Price list run, 107
 Pricing
 complex pricing, 170
 configuration, 129
 for industry solutions, 493
 in accounting, 505

Pricing (Cont.)

in selected applications, 493
 new fields, 167
 programs, 355
 shipment costs, 513
 special functions, 255
 user exits, 169
 Pricing analysis, 220
 Pricing communication structure, 116
 Pricing date, 159, 282
 PRICING function module, 326
 Pricing log, 229
 Pricing modules, 325
 Pricing procedure, 67, 129, 161
 business processes, 295
 ICAA01, 310
 PABR02, 238
 production system, 297
 PSER01, 307
 PSER02, 308
 RM0000, 502
 RVAA01, 143, 268, 298
 RVAA02, 83, 303, 344
 RVAB02, 306
 RVAXUS, 272
 ZVAA02 (*standard consolidated*), 465
 ZVAAEU, 271
 ZZVBKD (*data determination*
 VBAK/VBKD), 453
 Pricing procedure determination, 129, 144
 Pricing procedures, 295
 Pricing program, 326
 Pricing report, 75, 85
 configuration, 87
 execution, 86
 for condition maintenance, 91
 Pricing result, 379
 Pricing type, 356, 363, 387, 389, 498
 B, 230
 C, 498
 customer-specific, 241
 F (group valuation), 408, 428
 I (redetermine rebate conditions), 559
 Print formatting, 289
 order item values, 291
 output processing, 289
 Printing indicator, 164, 292

Processing level, 108
 Processing status, 63, 202
 material, 204
 Profit margin, 231, 475
 Profitability Analysis (CO-PA), 198, 202, 280,
 310, 522, 526
 Program
 BADI_CPF_SCALE_EVALUATION, 487
 BADI_PR_CPF_BASE, 193
 BADI_PR_CPF_DATA_RTRVL, 193
 BADI_PR_CPF_DATA_SOURCE, 187, 192
 BADI_PR_CPF_DATASOURCE, 485
 BADI_PR_CPF_RATE, 193
 BADI_PR_CPF_VALUE, 193
 BADI_PRICING_CPF_UI_FIELD, 198
 BADI_SD_REBATES, 595
 BON_ENH_SETTLE, 586
 CALCULATE_TAX_ITEM, 508
 GN_INVOICE_CREATE, 98
 PRICING, 356
 PRICING_COMPLETE, 256, 258, 367, 404
 PRICING_COPY, 252, 318, 360, 392, 456
 PRICING_GET_KOMK_KOMP_XKOMV, 198
 PRICING_LOAD_CONDITION_TYPS, 381
 PRICING_REFRESH, 380
 RBT_ENH_RMCCS469, 588
 REBATE_GENERAL_ANALYZE, 579
 RSNAST00, 289
 RV_PRICE_PRINT_HEAD, 293
 RV_PRICE_PRINT_ITEM, 293
 RV15B001, 571
 RV15B003, 571
 RV15B004, 578
 RV15C005, 578
 RV15C006, 578
 RV15CC01, 213
 SD_COND_ACCESS, 63, 361, 518
 SD_COND_READ_FROM_MEMORY, 364
 SD_COND_STATISTICS_READ, 400, 433
 SD_MOVE_A004_TO_A304, 202
 SDBONT06, 559, 576
 SDNETPRO (*Net Price List*), 202
 SDS060RB, 572
 SDVPRSUPDATE, 278
 TP_MATERIAL_PRICES_GET, 624
 USEREXIT_XKOMV_FUELLEN, 484
 Promotion, 80

Proofreading, 520
 PRSDT (pricing date), 523

Q

Quantity contract processing, 296

R

Rebate, 83
 Rebate condition type, 560
 Rebate index, 555, 595
 Rebate method, 552
 Rebate processing, 537
 Rebate recipient, 538, 590
 Rebate relevance, 542, 569
 Reconciliation account determination, 38
 Reference condition type, 155
 Release status, 63, 132, 201
 defined values, 202
 in agreements, 203
 in condition record, 202
 Rent processing, 296
 Rental contract, 307
 Report
 SD_MOVE_A004_TO_A304, 201
 SD_PRC_MULTI_ITEM_CHECK, 114
 Requirement
 002 (item with pricing), 557
 003, 65
 006, 502
 008, 318
 009, 315, 640
 014, 315, 640
 024, 302, 541
 055, 302
 057 (not for free goods), 448
 062, 305
 063, 352
 106, 389
 202 (price book - base price), 305
 55, 275
 902, 338, 341
 957, 616
 definition, 66
 Requirement routine, 35

Requirements, 388
 Resource-related billing, 252
 Result list, 91
 Retroactive rebate agreements, 551
 Revenue account determination, 38
 Rounding, 445
 Rounding difference, 288, 373
 Rounding difference comparison, 153,
 256, 257
 Rounding effect, 288
 Rounding rule, 151, 208, 210
 Routine
 GKOMV_bewerten, 375
 komt2_aufbauen, 362, 391
 kondition_pruefen (check condition), 426
 konditionen_lesen, 363, 391
 konditionsvorstep, 362, 391, 419, 523
 konp_2_xkonp, 363
 kwert_ermitteln, 402
 preisfindung_vorbereiten, 371
 pricing, 530
 provide_condition_data, 436
 userexit_change_pricing_rule, 424
 userexit_field_modification, 425
 userexit_move_field_to_vbap, 451
 userexit_move_field_to_vbkd, 453
 userexit_new_pricing_vbap, 498
 userexit_new_pricing_vbkd, 499
 userexit_pricing_check, 428
 userexit_pricing_copy, 419, 457, 532
 userexit_pricing_prepare_tkomk, 167,
 496, 529
 userexit_pricing_prepare_tkomp, 167, 359,
 381, 497
 userexit_pricing_rule, 419
 userexit_print_head, 420
 userexit_print_item, 421
 userexit_xkomv_bewerten_end, 421
 userexit_xkomv_bewerten_init, 421, 450
 userexit_xkomv_ergaenzen, 422
 userexit_xkomv_ergaenzen_manu, 423
 userexit_xkomv_fuellen, 365, 423
 userexit_xkomv_fuellen_o_konp, 423
 xkomv_aufbauen_aus_komt1, 391
 xkomv_aufbauen_pruefen, 363, 366
 xkomv_bewerten, 281, 398
 xkomv_ergaenzen, 363

Routine (Cont.)
 xkomv_fuellen, 365
 xkomv_kkurs_ermitteln, 285
 Routine number, 187

S

Sales deal, 204
 Sales document
 pricing, 223
 type, 236
 Sales document category, 496
 Sales order, 522
 SAP Business Client, 93
 SAP Customer Relationship Management
 (SAP CRM), 40
 SAP ERP Financials (FI), 206, 284
 SAP ERP Materials Management (MM), 158
 SAP ERP Sales and Distribution (SD), 95, 198,
 206, 505
 SAP List Viewer (ALV), 99
 SAP R/3, 206
 SAPscript, 102
 Scale base, 197, 257
 Scale base formula, 400
 023 (Partial Quantity), 393, 400
 043 (tax license France), 433
 Scale evaluation, 185, 194
 Scale evaluation routine, 194
 Scale group, 579
 Scale rebate, 601
 Scale-based conditions
 with group key, 257
 without group key, 257
 Service charge settlement, 160
 Service processing, 296
 Settlement material, 540, 549, 601
 Settlement period (rebate), 547, 568
 Shipping costs, 447
 Ship-to party, 336
 Simple condition exclusion using
 requirements, 258
 Sold-to party, 336
 Source (condition) table, 211
 Source condition type, 210
 Special condition type, 302

Statistical group condition, 317
 Status (rebate agreement), 553
 Stock sale (no cross-company), 277
 Stock sale/drop shipment, 295
 Structure

 KOMG, 324
 KOMK, 43, 325, 387, 522, 526
 KOMKAZ, 167
 KOMP, 43, 325
 KOMPAX, 327, 344
 KOMPAX (pricing result), 410
 KOMPAZD, 43, 58, 327, 331
 KOMPLOOP, 327
 KOMV, 325
 PRCS_PRICING_CPF_CONTEXT, 178
 RKOMV, 349
 RKONP, 348
 SAP_REFERENCE_RATE, 186
 Structure condition, 151
 Sub-parameters, 186
 Switch Framework, 92

T

Table
 A053, 507
 conditions not to be redetermined, 389
 CPFC_USAGETASK, 489
 FPGA/FPLT (billing plan), 382
 GKOMV, 328
 KOMT2, 328
 KONP, 328
 KONV, 426
 KOTE001, 565
 MBEW, 277
 S060, 264, 302, 543
 S071, 264, 434
 S136, 560, 571, 577
 S469, 302, 583
 STEU, 328
 T001, 282
 T052, 315
 T681Z, 41
 T682V, 518
 T685, 58
 TKOMVD, 293
 TKSCHL, 328, 389

- Table (Cont.)
 - TKSCHLEXCL, 328
 - VBOX, 555, 569, 571
 - XKOMPLOOP, 350
 - XKOMV, 326
- Table alias, 109, 112
- Table buffering, 516
- Tax calculation, 510
- Tax calculation procedure
 - TAXEUR, 506
 - TAXUSX, 508
- Tax classification, 269
- Tax determination, 267
 - in accounting, 506
 - in sales document, 508
 - sales pricing, 270
- Tax determination procedure
 - TAXD, 267
 - TAXES, 269
 - TAXUSX, 272
- Tax increase, 273
- Tax jurisdiction code, 273
- Tax trigger, 269
- Terms of payment, 450
- Testing, 515, 531
 - pricing scenario, 145
- Text determination, 161
- Text determination procedure, 218
- Time-dependent calculation types, 308
- Transaction
 - 8KEV, 505, 506
 - DP90, 252, 280, 308, 312, 612
 - FTXP, 267, 507
 - OKOZ, 505
 - OMO1, 560
 - OV77, 61
 - OVB3, 571
 - OVG1, 571
 - RBT_ENH_VB7, 586
 - SE11, 516
 - SE30, 530
 - SE43, 75
 - SE80, 480
 - SFW5, 92
 - SM30, 41, 216
- Transaction (Cont.)
 - SO10, 102
 - ST05, 531
 - V_I7, 79
 - V_NL, 98
 - V_NLN, 100
 - V.25, 314, 413, 436, 441, 611
 - V/16, 240
 - V/15, 79
 - V/16, 79
 - V/LD, 86
 - VA01, 529
 - VA01, VA02, 499
 - VA05, 454
 - VB(7, 547, 553, 573
 - VB(8, 579
 - VB(D, 578
 - VB22, 203
 - VCN1, 274
 - VBO1, 538
 - VBOE, 578
 - VBOF, 552
 - VF01, 524
 - VF31, 289
 - VFSN, 572
 - VFTA and VTFL, 283
 - VK01, 50
 - VK03, 48
 - VK05, 48
 - VK11, 62, 69, 87, 335
 - VK12, 69, 87, 204, 208
 - VK13, 69, 87
 - VK14, 69, 206
 - VK31, 62, 75
 - VK31 to VK34, 91, 215
 - VK32, 75, 77
 - VK33, 75
 - VK34, 75
 - VOB3, 573
 - VOBF, 576
 - VOFM, 65, 168, 210, 385, 525
- Transaction-specific pricing procedure, 550, 563
- Turnover update, 83

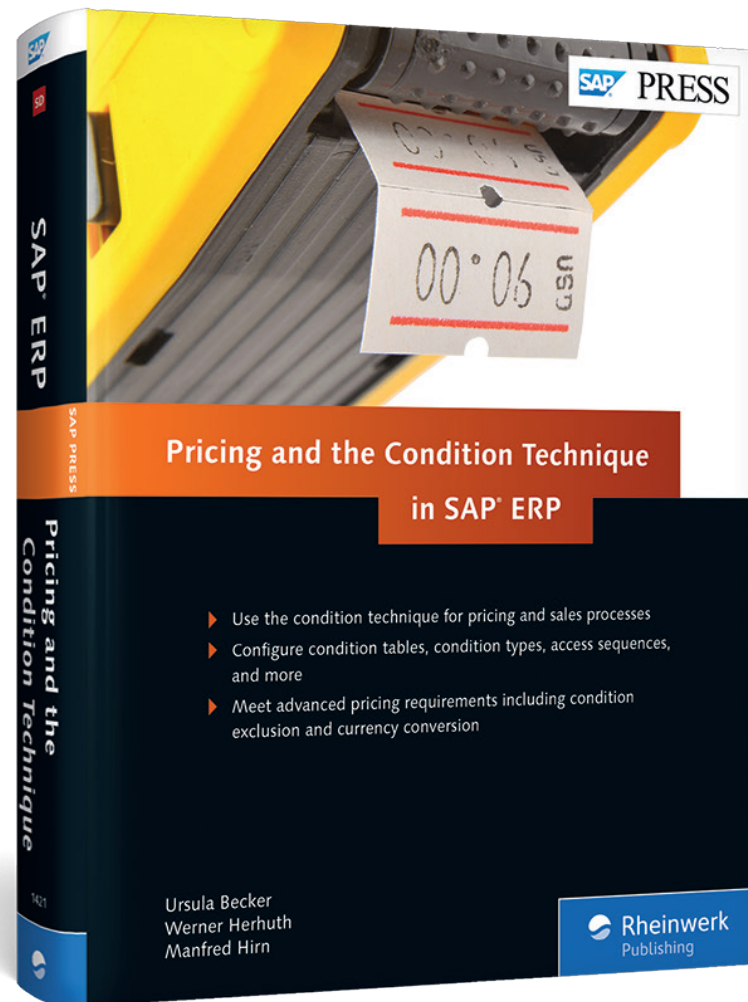
- U**

 - Update prices, 235
 - Usage, 37
 - 3: Campaign Determination, 40
 - A: Pricing, 37
 - B: Output, 37
 - C: Account Determination, 38
 - D: Material Determination, 38
 - E: Rebate, 39, 537
 - G: Material Listing and Exclusion, 39
 - N: Free Goods, 39
 - T: Data Collection, 40
 - Usage of condition tables, 35
 - Usages of conditions, 41
 - User exit, 113, 418
- V**

 - VAKEY, 49, 596
 - Validity period, 62, 80, 132, 206, 207
 - rebate, 538
 - rebate agreement, 566
 - rebate condition record, 539
 - Validity period split, 207, 208
- Valuation, 376
- Value contract processing, 296
- Value Formula 038, 261
- Variant, 77
 - condition, 158
 - definition, 77
- VBOX, 555
- Verification level, 544, 566
- View V_T683V, 499
- VPRS (Cost), 415
- W**

 - WAVWR (cost in document currency), 277, 414, 463
 - Worklist, 92
 - change query, 93
 - customer-specific type, 95
 - define new query, 96
 - personalize query, 95
- Z**

 - ZTERM (terms of payment), 323



Ursula Becker, Werner Herhuth, Manfred Hirn

Pricing and the Condition Technique in SAP ERP

665 Pages, 2017, \$89.95

ISBN 978-1-4932-1421-1

 www.sap-press.com/4173



Dr. Ursula Becker has been the development architect responsible for the pricing functionality in SAP ERP since 2009. She studied physics and astronomy at the University of Heidelberg and joined SAP in 1998, after completing her doctorate and two years of research. In the following years she worked in support before moving on to development—with a focus on pricing, billing, and rebate processing. She has also worked in the areas of SAP Customer Relationship Management (SAP CRM) and SAP Business ByDesign.



Werner Herhuth is a certified consultant in the area of order fulfillment (in SAP ERP) and the author of several SAP courses. He studied mathematics and business administration with specialization in business informatics at the University of Mannheim. He worked for 13 years as a systems analyst and head of application programming in the engineering industry before joining SAP in 1996.



Manfred Hirn was responsible for the development of the condition technique, as well as the billing and pricing functionalities which he helped program, in SAP R/3. He studied mathematics with a minor in business administration at the University of Würzburg. In his first job at a brewery, he was involved in the development and introduction of an order and trip management system. In 1984, he joined the SAP development team and supervised the SAP R/2 billing program. Later he worked as a development manager in the areas of sales, pricing, and billing.

We hope you have enjoyed this reading sample. You may recommend or pass it on to others, but only in its entirety, including all pages. This reading sample and all its parts are protected by copyright law. All usage and exploitation rights are reserved by the author and the publisher.